

M-Commerce를 위한 XML문서 변환 도구 개발

윤석용, 하상호
순천향대학교 정보처리학과

Development of A Tool for Transformation XML Document for M-commerce

Seokyoung Yoon, Sangho Ha
Dept. of Information Technology, SoonChunHyang University
E-mail: fire400@nownuri.net, hsh@sch.ac.kr

요 약

XML은 웹문서 기술을 위한 표준과 모든 유형의 정보에 대한 데이터 기술 언어로서 점차 그 사용 폭을 넓혀가고 있고 특히 E-commerce(인터넷 전자상거래)를 위한 웹문서 표준으로 자리를 잡아가고 있다. 또한 최근에는 이동전화를 사용하여 데이터 통신을 하는 WAP 즉 WML 역시 이동전화의 보급으로 인해 사용 폭이 매우 급속히 증가되고 있고 이로 인해 M-commerce(모바일 전자상거래)의 관심 역시 커지면서 WML이 M-commerce를 위한 표준이 되고 있다. 하지만 웹에서 작성된 XML데이터를 WML로 변환하는데 있어서 수작업으로 변환해야 하는 문제점을 안고 있다. 본 논문에서는 이러한 문제점을 해결하기 위한 방안으로써 모바일 인터넷 통신을 위해 임의의 XML문서를 자동으로 WML문서로 변환하여 즉시 원하는 XML데이터를 WML로 변환 가능하여 E-commerce와 M-commerce간에 원활한 데이터 변환이 가능한 시스템을 구현한다.

1. 서론

XML[4]가 웹 상의 데이터 표현하고 교환하기 위한 새로운 표준으로 빠른 속도로 받아들여지고 있다. XML 데이터는 그 구조를 자체적으로 기술 가능하며, 따라서 XML 문서를 해석하고 그 문서를 조작하는 응용 프로그램을 작성하는 것이 가능 할 정도로 매우 유용성 있는 문서이며 현재 특히 E-commerce[7]을 위한 웹 문서 표준으로써 자리를 잡아가고 있다. 또한 WML[6]은 역시 최근 이동전화의 보급으로 인해 모바일 인터넷 통신에 있어서 급속도로 사용량이 많아지고 있으며 각 이동통신 회사 역시 앞다투어 새로운 WML서비스를 제공하고 있다. 이중 모바일 전자상거래(M-commerce)[6]는 원하는 구매할 때 상품을 선택한 후 바로 이동통신요금으로 결제가 가능하여 M-commerce 비중이 높아지고 있고 이로 인해 현재 구축되어 있는 XML데이터들이 WML로 변환에 대한 필요성이 커지고 있다.

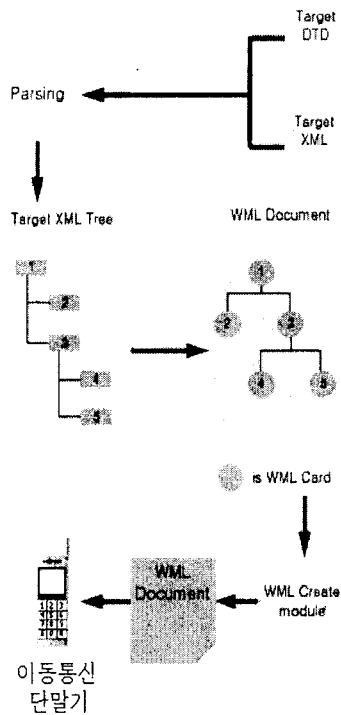
현재 많은 양의 XML데이터가 WML데이터로 변환되고 있으며 모바일 이동통신의 성장에 따라 앞으로는 지금 보다 더 많은 양의 XML데이터들이 WML로 변환 될 것이다. 하지만 현재 문서 변환하는데 있어서 사람이 수작업으로 XML의 문서를 하나 하나 분석하여 WML로 변환하고 있다. XML문서는 하나의 문서로 이루어진 HTML문서와는 달리 XML문서와 그 문서의 구조를 나타내는 DTD[4]로 이루어져 있어 데이터량이 큰 XML문서를 다루기 위해서는 XML문서를 분석하는데 많은 시간을 소비할 수밖에 없고 이 분석 내용을 XML문서에 익숙하지 못한 WML 프로그래머한테 완벽히 내용을 전달하기는 매우 힘들다. 또한 방대한 양의 데이터를 가지고 있는 XML을 WML로 변환하기 위해서는 디스플레이에 문제점이 있다. WML은 단지 디스플레이 창에 1줄에 최대 12 문자, 최소 4줄 최대64줄의 텍스트가 표현 가능해 많은 양을 데이터를 하나의 창에 표시하기에는 불가능

한 문제점이 있다.[9]

본 논문에서는 이러한 디스플레이의 문제점을 해결하기 위해 XML에 한 계층을 WML에 한 Card와[9] 동일하게 하여 하나의 XML문서를 여러 개의 계층으로 분해해서 데이터량을 줄임으로 디스플레이의 문제점을 해결하였다. 자세한 내용은 2장에서 언급하기로 한다. XML2WML 애플리케이션에 대해서는 3장에서 살펴보고 4장은 예제를 통해 변환되는 과정을 알아보고 모바일 에뮬레이터에서 어떻게 표현되는지 확인해보겠다. 마지막 5장에서는 결론을 언급하기로 한다.

2. 시스템

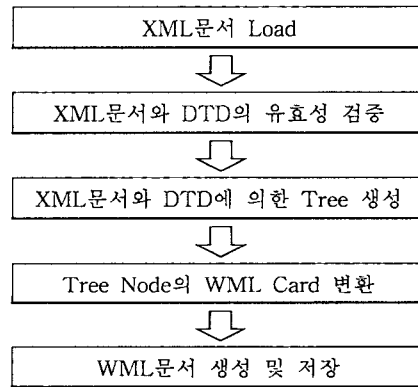
여기에서는 본 시스템의 전체 흐름도와 주요 모듈별의 특징에 대하여 살펴본다. 아래 그림 1을 의 시스템의 전체흐름도를 보여준다.



<그림 1. 시스템의 전체 흐름도>

이 변환 애플리케이션을 구현하는데 있어서는 크게 5단계로 나눌수 있다. 첫 단계로 XML문서와 DTD문서를 읽어들이고, 두 번째 단계로 읽어들이는 문서를 파

서에 넣고 XML문서와 DTD문서가 유효한지를 검사한다. 세번째 단계에서는 유효성이 검증된 XML문서와 DTD가 파서에 의해 XML Tree[4]로 생성되며 XML문서와 DTD문서의 유효성을 검증한다. 다음 이 유효성을 확인한 XML문서를 XML Tree로 만들기 위해서 IBM XML DOM Parser[5]을 사용하여 트리 변환을 하였다. 네 번째 단계에서 트리의 상위계층부터 하나씩 WML문서의 한 페이지를 나타내는 Card로 변환하고 마지막 단계에서 이 문서를 WML문서로 저장한다. 이렇게 어플리케이션 구조를 크게 5단계로 나눌수 있다.



<그림 2. 어플리케이션 구조도>

이중 4단계에서 XML Tree의 상위계층부터 WML의 Card로 변환하는 것이 이 애플리케이션 핵심이라고 볼 수 있다.

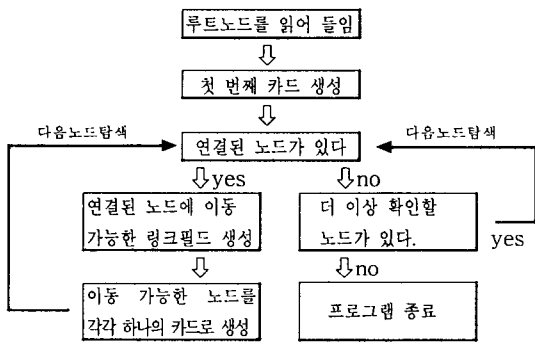
Parsing 모듈

먼저 사용자는 변환하기 원하는 Target XML을 불러 들여 파싱하는데 이 XML문서가 내부 DTD를 사용하면 바로 파싱에 들어가고, XML문서가 외부 DTD를 사용하면 XML문서에서 선택한 Target DTD를 불러들이고 이 XML문서와 DTD를 파서에서 유효성을 검사를 한다. 이때 먼저 XML문서 어휘분석을 하여 문법에 적절한지 확인하고 XML문서와 DTD문서가 서로의 규칙에 맞는 유효한 문서인지 구문분석한다. 유효성이 확인되면 파서에서 이 XML DOM트리를 만들게 되는데 여기서 사용한 XML파서는 IBM사에서 제공하는 XML4JAVA 1.1.9버전[5]을 사용하여 XML DOM트리를 만들었다.

Transformation 모듈

이 IBM파서는 만들어진 XML DOM Tree를 사용하여 문서 변환 이루어진다. 트리에서 만들어진 노트

들이 WML의 Card로 변환되고 이 Card들이 모여서 하나의 WML문서가 된다. 여기서 WML문서의 Card란 모바일 창에 한 페이지를 나타내는 부분을 말한다. 이 Card가 각각 한 페이지를 나타내고 이 Card들이 모여서 여러 페이지를 나타내는 하나의 WML문서가 나오는 것이다. XML문서는 많은 노드를 가지고 있는데 이 노드들을 하나의 Card로 표현하였다. 더 많은 정보를 한 Card에 넣지 않고 한 Card에 한 노드의 정보를 넣은 이유는 모바일 창은 단지 한 줄에 12글자를 나타낼 수 있으며 현재 모바일 창에서 한번에 보이는 줄은 8줄이 최대이기 때문에 많은 내용을 한 창에 보이는 데에는 문제가 있다. 물론 8줄은 한 창에서 보이는 최대의 줄이지 Card 한 페이지에서 나타낼 수 있는 최대 줄의 크기는 아니다. 예를레이터에서 화살표 버튼을 사용하면 최대 64줄까지 표현 가능하다. 본 논문의 XML 트리에서 WML로 변환되는 과정은 다음과 같다. 먼저 루트 노드를 읽어들이고 그 읽어들이는 루트노드로 첫 번째 카드로 만든다. 이후 연결된 노드가 있는지 확인하고 있을 경우 연결된 노드로 이동 가능한 링크필드가 생성된다. 이 이동 가능한 노드는 다시 하나의 카드로 생성되며 다음 노드를 찾게 되는데 여기서 탐색 방법은 Pre_order 탐색을 선택하였다. 노드를 생성한 다음 이와 같은 작업을 계속적으로 반복 수행하다가 연결된 노드가 없으면 더 이상 탐색할 노드가 있는지 확인하고 없으며 변환 프로그램은 종료한다.



<그림 3> XML트리에서 WML문서로 변환과정

WML 모듈 생성 모듈

변환하여 만들어진 만들어진 하나의 카드가 생성될 때마다 한 Card씩 WML문서는 저장되며 모든 변환이 끝나면 이 문서는 완성된 하나의 WML문서가 되기 위해 WML태그를 넣어주는 등 몇 가지 수정이 이루어지고 이 작업이 끝나면 완벽한 WML문서가 만들어지고 이 완성된 WML문서는 View창에 저장된다.

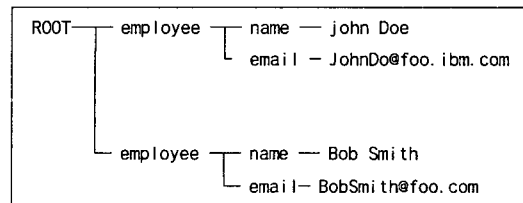
3. 예제

여기서는 하나의 XML파일을 사용하여 WML로 변환하는 과정을 살펴보기로 하겠다. 뒤에 나오는 XML문서는 사용한 XML문서는 간단한 인사관리 XML파일이다.

```

<?xml version="1.0"?>
<!DOCTYPE TERMDIC SYSTEM "Info_emp.dtd">
<department xml:space="preserve">
  <employee id="J.D" sd="jk" tt="aa">
    <name>John Doe</name>
    <email>John.Doe@foo.ibm.com</email>
  </employee>
  <employee id="B.S">
    <name>BobSmith</name>
    <email>Bob.Smith@foo.com</email>
  </employee>
</department>
  
```

위에 Info_emp.xml 파일을 불러들여 파싱 작업이후 아래 그림.4 과 같은 트리가 만들어진다.



<그림 4> Info_emp.xml XML Tree>

그림 4의 트리를 예들 들어 변환하면 먼저 루트 노드를 읽어 들인 후 첫 번째 Card를 생성되고 이 생성된 Card의 ID는[9] Default로 name이라고 붙게 된다. 이 노드가 연결된 노드가 있는지 확인한 후 연결된 노드가 아니므로 루트에 연결된 노드의 링크 필드를 만들고 연결된 노드에 해당하는 Card의 ID는 자동으로 name_1, name_2 만들어진다. 루트에 연결된 노드는 다음 계층이 되고 위의 트리에서 루트노드 다음에 나오는 두 개의 employee 노드가 다음 계층이 된다. 이 계층을 표현하기 위해 이미 생성된 ID의

Name이 Card의 ID가 되며 여기서는 name_1이라고 생성된 ID를 불러오게 된다. 다음 계층에서 같은 일이 반복되며 모든 노드는 Pre_order탐색 순서에 따라 각각의 Card가 생성되고 생성 될 때마다 저장작업이 일어난다. 더 이상이 탐색할 노드가 나오지 않으면 저장된 Card들은 모이고 WML태그작업이후 완벽한 WML문서가 된다. 아래의 WML소스는 변환된 소스 중 일부를 나타낸 것이다.

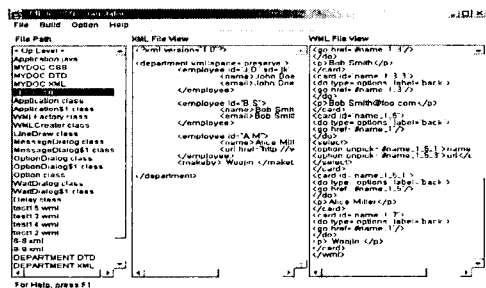
```

<wml xml:lang="lang">
<card id="name">
<p>#document</p>
<do type="accept" label="ok">
<go href="#name_1"/>
</do>
</card>

<card id="name_1">
<do type="options" label="back">
<go href="#name"/>
</do>
<select>
<option onpick="#name_1_1">employee</option>
<option onpick="#name_1_3">employee</option>
</select>
</card>

<card id="name_1_1">
<do type="options" label="back">
<go href="#name_1"/>
</do>
<select>
<option onpick="#name_1_1_1">name</option>
<option onpick="#name_1_1_3">email</option>
</select>
</card>
이후생략
    
```

4. 애플리케이션 구현



<그림 6: WML로 변환된 상태>

창은 크게 File Path창, XML File View창, WML File View창 3개로 나누어지며 File Path창은 선택한

XML 파일을 불러 올 수 있게 하고, XML File View 창은 불러온 파일의 소스를 볼 수 있게 하고 WML File View창은 변환된 WML File 소스를 볼 수 있게 한다.

메뉴바는 아래와 표와 같다

FILE	BUILD	OPTION	HELP
-OPEN XML	-RELOAD	-CONVERTER	-MAKE BY
-SAVE XML	-CONVERTER	OPTION	-ABOUT
-SAVE WML	WML		CONVERT
-EXIT			

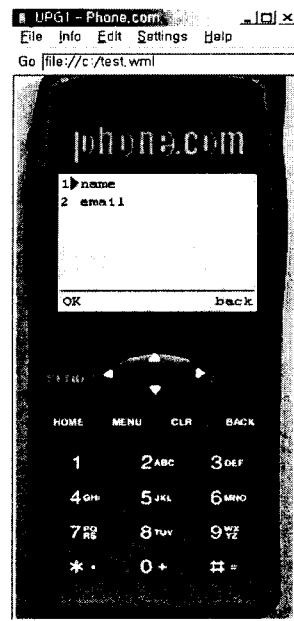
-File바에서는 각각의 XML과 WML의 불러오고 저장할 수 있게 한다.

-BUILD바에서는 XML을 WML 변환시켜준다

-OPTION바에서는 변환되는 내용에 옵션을 지정한다.

-HELP에서는 제작자 및 프로그램 내용을 표시한다.

위의 WML파일은 원하는 이름으로 File바의 SAVE WML를 사용하여 저장 가능하고 위의 파일을 폰넛컴에서 제공하는 UP.SDK 4.0[9]버전을 WML에블레이터를 사용하여 확인하면 아래와 같다.



<그림 7: WML 애플레이터로 문서확인 >

5. 결론 및 향후 연구과제

이동통신의 발달로 인해 우리는 직간접적으로 WML을 사용한 WAP을 접하고 있으며 효율적인 웹문서인

XML 역시 인터넷상에서 많이 쓰이고 있다. 하지만 이 문서들의 상호 교환을 수작업으로 밖에 할 수 없다는 문제로 인해서 많은 XML문서가 WML로 변환되지 못하고 있고 XML 문서로 만든 E-commerce 데이터를 M-commerce로 변환하기 위해서는 XML2WML은 필수적이다. 이 어플리케이션에서는 임의의 XML을 WML로 변환시켜 상호 호환 가능하므로써 XML문서를 모바일에서 사용 가능하도록 만들었으며 XML에 대한 전반적인 지식이 없는 WML 프로그래머도 XML문서를 작업하도록 하였다.

향후 작업은 모바일은 이동전화에서 보다 참이 크고 더 방대한 데이터 전송 가능한 PDA로 넘어 갈 것으로 전망되고 있다. 현재 PDA에서 많이 사용되는 OS PAM에서[9] 어떻게 XML을 표현 가능하게 만드는 방법에 대한 연구가 수행될 것이다.

[참고 문헌]

- [1]Cay S. Horstmann, Gary Cornell "Core java 2nd Edition, sun soft Press, Prentice Hall PTR, 1997
- [2]Cay S. Horstmann. Grat Cornell "Core java 2nd Edition, Prentice Hall PTR, 2000
- [3] Simon St. Laurent, Ethan Cerami : Building XML Applications, McGraw-Hill Professional Publishing, 1999
- [4] Benoit Marchal : Applied XML Solutions (Sams Professional Publishing), Sams, 2000
- [5] Hiroshi Maruyama, Kent Tamura, Naohiko Uramoto : XML and Java: Developing Web Applications, Addison-Wesley Pub Co, 1999
- [6] Martin Frost : Learning WML & WMLScript : Developing Web Applications, O'Reilly, 2000
- [7] Elliotte Rusty Harold, Elliotte Rusty Harold : XML Bible 2nd Edition, Hungry Minds, Inc, June 2001
- [8]Frank Boumphrey의 11명, XML Applications, Wrox Press Inc, 1999
- [9]LG-EDS SYSTEM I&TEC Team : 무선인터넷 어플리케이션 프로그램: 삼양출판사, 2000