

XML/EDI, WML을 사용한 시스템 연계에 관한 연구^{†*}

안동률^{**}, 박정선^{***}

A Research on System Interconnection Using XML/EDI, WML

Dong-Ryul Ahn, Jung-sun Park

Abstract

As the population of the Internet increases, so does that of mobile Internet. Currently, the use area of mobile Internet is rather limited due to the low speed of transmission and lack of applications. However, it is evident that the speed will be improved and more applications will be developed.

In this paper, we propose: i) to use WAP in SCM, ii) to interconnect WAP and XML/EDI system, iii) to interconnect WAP, XML/EDI, Legacy system in SCM. We implemented a prototype which can show the interconnection among WAP, XML/EDI, and Legacy system. We hope the concept could be used in real applications in the near future.

Key Words : WAP, XML/EDI, Legacy system, interconnection, SCM

† 본 논문은 제6권 3호에 실릴 논문이었으나 사정에 의해 못 실려 이번 호에 함께 실었습니다.

* 본 연구는 명지대학교 신진교수 교내연구비 지원을 받아 수행되었습니다.

** 명지대학교 산업시스템공학부

*** 명지대학교 산업시스템공학부 부교수

1. 서론

세계는 개방화되어 가고 있으며, 따라서 정보의 가치가 더욱 중요하게 되고 있다. 국가와 기업이 경쟁에서 살아남기 위해서는 정보의 효율적 관리와 공유 그리고 가치 있는 정보로 가공하는 것이 필요하다. 이러한 환경에서 등장한 것이 EDI(Electronic Data Interchange)이다. 오늘날 많은 주문서, 송장과 같은 B2B 문서들은 EDI 메시지로 교환된다. XML/EDI는 전통적인 EDI가 안고 있는 여러 가지 문제점을 해결하기 위한 접근방법으로 XML/EDI 그룹에 의하여 제안되었다. 이러한 XML이 EDI에 도입됨으로써 대규모의 전자 상거래 자동화를 가능하게 해주며 전통적인 EDI의 문제점을 해결할 수 있으며 차세대 EDI의 주요 기술로 자리 잡고 있다.

유선망 못지 않게 무선 인터넷(mobile internet)의 영역 또한 빠르게 확장되어가고 있는데 휴대용 개인 단말기를 통한 무선 인터넷은 시간과 장소의 제약 없이 정보의 검색 및 처리가 가능하여 근래에 많은 주목을 받고 있는 부문이다. 무선 인터넷은 이동통신과 인터넷이라는 두 가지 기술을 필요로 하며 두 망 사이의 연동을 위하여 새로운 방식이 필요하게 되었다. 이후 프로토콜의 통일을 위하여 WAP 포럼에서 무선 인터넷 프로토콜인 WAP(Wireless Application Protocol) 1.0을 정의하였다. WAP은 1997년 WAP Forum에서 결성된 이후, 그 사용언어로 무선 단말기의 특성과 낮은 대역폭을 갖는 이동통신에 적합하게 WML(Wireless Markup Language)를 정의하여 개발하였으며 현재 세계 모바일 언어의

90%를 점유할 정도로 대표적으로 인식되어 본 논문에서 사용하게 되었다. 기술적으로는 WAP은 현재 이동통신망에서 인터넷 서비스를 제공할 수 있는 표준으로 인터넷 브라우저가 탑재되어 있는 휴대폰으로 접속이 가능하며 HTTP와 별도의 WAP 프로토콜이 기존의 기술과의 호환성을 제공하고 어플리케이션의 개발이 가능하기 때문에 다른 방식에 비하여 많은 유연성을 갖고 있으며 기존의 서비스와 차별화된 서비스를 개발하기에 유리하다. 현재까지는 무선 인터넷과 SCM(Supply Chain Management)의 연계가 부족하며, XML/EDI 망과 연동시켰을 때 큰 시너지 효과를 볼 수 있으나 구현 사례가 거의 없는 실정이다.

따라서 본 논문에서는 WML과 XML/EDI 기술을 이용하여:

- WML의 전자상거래에 대한 응용 가능성,
- WML, XML/EDI의 연계에 의한 응용 범위 확대 가능성,
- WML, XML/EDI, Legacy 시스템의 연계 가능성

을 보여주는 프로토타입을 제시하고자 한다.

고객은 인터넷 망을 통해 XML/EDI를 통하여 판매자의 DB에 접근하여 제품의 조회, 주문, 주문처리 과정의 확인, 주문내역서의 열람 및 고객 DB 수정 등이 가능하다. 무선 인터넷 망(WAP)으로 접속시 WML 코드로 정보를 주고 받으며 Email을 통한 XML/EDI로 판매자는 고객에게 주문내역서를 보내준다. 본 논문의 구성은 다음과 같다. 2장에서는 WAP과 WML의 소개와 XML/EDI의 개념설명 및 필요한 기술에 대해 알아본다. 3장에서는 WML과 XML/EDI를 사용하여 Legacy 시스템과 연동되는 프로토타입을 제시한다. 4장에서는 결

론 및 향후 연구방향에 대해 제시하였다.

2. WML과 XML/EDI

2.1. WAP과 WML

2.1.1 WAP과 WML 현황

WAP은 Phone.com(현 OpenWave)을 주축으로 대표적인 휴대폰 제조업체가 공통 규격의 제정을 위해 만든 표준화 단체인 WAP 포럼에서 제정한 무선망과 인터넷 연동을 위한 프로토콜이다. WML은 XML을 기반으로 하는 WML(Wireless Markup Language)과 WMLScript을 사용하여 개발하며 휴대용 단말기에 적합하도록 설계된 언어이다. WML은 하나의 DECK와 여러 개의 CARD로 구성되며, 핸드폰 사양에 맞게 개발된 언어이고 <TAG>로 구성되어 있어서 기존 HTML 및 XML 코딩과 유사하다. 대표적인 특징으로는 XML 문서의 문자 집합을 사용하고 대소문자를 구분, 대역폭(Bandwidth)이 좁고 상대적으로 가벼운 트랜잭션을 지원한다[1]. WML 문서구조를 보면 <그림 1>과 같다.



<그림 1> WML문서의 구조

한 개의 Deck은 서버가 한 사용자에게 보낼 수 있는 WML의 가장 작은 단위를 나타낸다. 따라서 WML 문서 하나를 하나의 Deck이

라고 생각하면 된다. Deck 안에는 모든 형태의 WML 구성요소가 들어간다. Card는 WML을 이루는 가장 기본적인 단위이다. 그러므로 하나의 Card로 이루어진 WML 문서 또는 Deck이 당연히 존재할 수도 있다. 또한 여러 개의 Card로 이루어진 WML 문서도 한번에 단 하나의 Card만 보일 수 있다. 하나의 Card 안에서 Navigation, Variable, Input Elements 등과 같은 요소들이 존재하는데, 이러한 요소들은 WML에서 UI와 Control을 표현하는 요소들이다. Card 사이에서의 이동은 링크문을 사용하는데 Card id="이름"에 의해 각각 지정된 이름으로 이동하게 된다[2].

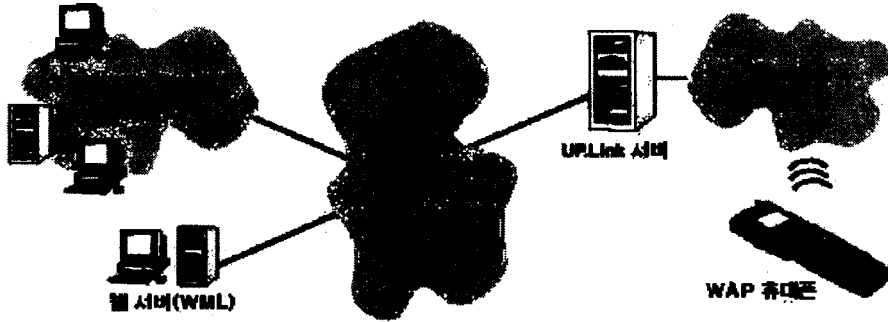
```
<?xml version="1.0" encoding="ks_c_5601-1987"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM/DTD WML 1.2//EN"
"http://www.wapforum.org/DTD/wml_1.2.xml"?>
```

<그림 2> WML문서의 서문

```
<?xml version="1.0" encoding="EUC-KR"?>
<?xml:stylesheet type="text/xsl" href="phone.xsl"?>
<!DOCTYPE phone SYSTEM "phone.dtd">
```

<그림 3> XML 문서의 서문

WML 문서는 XML을 기반(<그림 2>, <그림 3> 참조)으로 하고 있으며 문서 정의를 위해 XML의 DTD(Document Type Definition)를 사용한다. WAP은 XML을 사용하여 개발할 수 있는데 핸드폰 특성에 맞게 개발할 수 있다는 장점과 WAP 게이트웨이를 사용한다는 단점이 있으며, Microsoft사의 휴대용 단말기 언어인 ME는 기존 인터넷 환경을 그대로 사용할 수 있다는 데에 장점이 있다. ME는 기존 유선 체계를 그대로 수용, 반영하려는 MS의 의도로 빠른 성장을 하고 있다. 하지만, 아직까지는 통신 사업자와 인터넷폰 제조업체



<그림 4> WML 시스템 구성도

등의 약 90%는 WAP을 사용하고 있어 ME의 시장확대가 쉽지 않다.

<그림 4>는 WML 시스템 구성을 보여주는 것으로 고객은 WAP 휴대폰을 통해 무선망에 접속하고, UPLink 서버(이동통신사업자)를 거쳐 인터넷과 인트라넷 등에 연결이 가능하다.

2.1.2 무선인터넷 구성요소



<그림 5> WAP 기반 무선인터넷 구성요소

위의 그림은 WAP 기반 무선 인터넷 구성요소들간의 상호작용을 제시하고 있다. CP 서버로부터 텍스트 형태의 WML 파일들이 WAP 게이트웨이로 HTTP망을 통하여 전송되고 WAP 게이트웨이는 이를 이진(binary) 형태로 변환하여 무선망을 통해 휴대폰으로 전송한다. 무선망의 전송 방식은 Bearer 타입

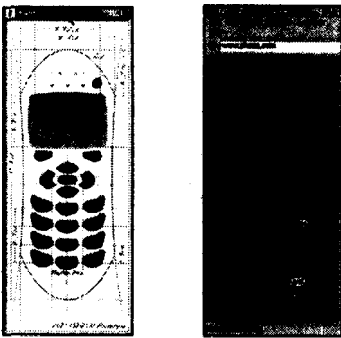
에 의해 결정된다. Bearer는 WAP Gateway와 휴대폰을 연결하는 통신망 또는 전송 방식을 의미한다. 국내의 이동 통신은 CDMA 방식이고, 유럽의 경우 GSM 방식이다. WAP 기기는 WAP을 지원하는 장치로 휴대폰, PDA(Personal Digital Assistant) 등을 말한다[4].

2.1.3 WAP 애플레이터

휴대폰이 아닌 PC에서 WML파일을 실행하고 결과값을 볼 수 있도록 지원하는 애플레이터가 여러 종류 공개 혹은 비공개로 오픈되어 있다. 그 중에 대표적인 것을 소개하자면 노키아(Nokia)사에서 제공하는 WAP Development Tool Kit이 있다. <http://forum.nokia.com>에서 다운로드받을 수 있다. Nokia 전용 게이트웨이를 사용하여 셋팅방법이 좀 까다롭고 한글이 지원되지 않아 사용이 불편하다. 반면 작업환경이 편리하며 디자인이 상당히 미려하다.

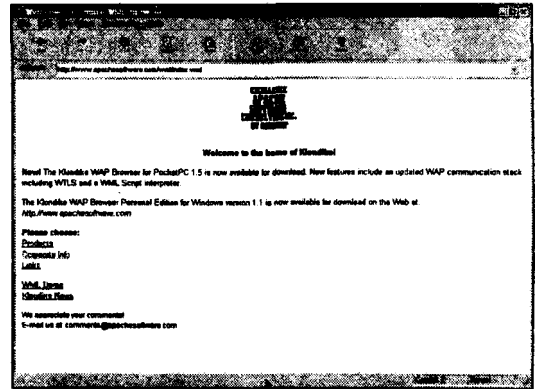
다음으로 많이 사용되고 있는 것은 Phone.com의 UP 애플레이터이다. 애플레이션할 경우 정상적으로 작동되어야 하나 현재 Phone.com 사이트가 openwave.com으로 전환 작업중이라

WAP 게이트웨이가 정상적으로 작동되지 않아 사용할 수 없었다. 현재 4.2 버전까지 나와 있으며 4.0 버전만 한글의 입·출력을 지원한다. 다운로드드는 www.openwave.com에서 받을 수 있다. 장점은, 가장 대표적으로 사용되는 에뮬레이터이기 때문에 실제 작동유무를 확실하게 검증할 수 있고 폰의 스킨을 다양한 제품으로 바꿀 수 있다는 점이며, 단점으로는 한글 지원이 잘 안되는 점이 있다.



<그림 6> Nokia Wap SDK와 UP 에뮬레이터

마지막으로 본 논문에 사용하고 있는 에뮬레이터로 Apache Software Consulting에서 개발한 Klondike WML 브라우저에 대해 소개하겠다. 이 브라우저는 다른 에뮬레이터와는 다르게 Internet Explorer처럼 브라우저 형태로 표현한다. 또한 한글을 잘 지원하여 에뮬레이터로 적합하다. 현재 1.5버전까지 나와 있으며 Micro Browser(ME지원)용도 있고 포켓 PC용 버전도 있다. www.apachsoftware.com에서 다운로드 받을 수 있다. 단점이라면 <Input> tag로 입력할 때 한글이 입력되지 않는다는 점이다. 영문만 입력이 가능하다.

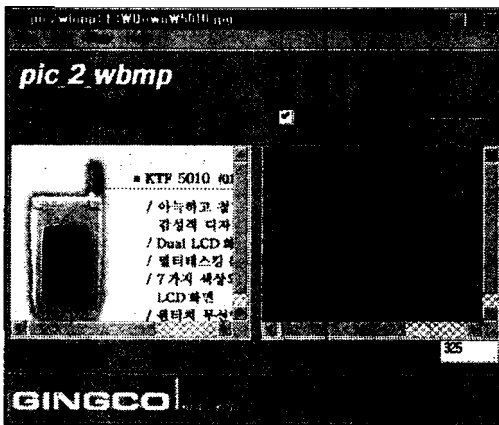


<그림 7> Klondike WAP Browser

2.1.4 Wbmp 이미지 포맷

WML에서 그래픽 이미지를 텍스트와 함께 사용할 수 있다. 그래픽 이미지를 사용하게 되면 접속자가 WML 응용 서비스를 쉽게 보고 느낄 수 있다. 그렇지만, WML을 사용하는 경우, 이미지는 어디까지나, 텍스트의 내용을 도와주기 위한 보조수단이고 또한, 모든 휴대폰에서 이미지가 가능한 것이 아니므로 텍스트와 함께 사용하여야 한다. 텍스트에 임의로 이미지를 포함하기 위해서는 다음과 같이 element를 사용하면 된다. 현재 사용할 수 있는 이미지는 흑백 BMP 이미지와 WBMP 이미지이다. 흑백 BMP의 경우는 기본적인 이미지 에디터에서 모두 지원이 되고, WBMP는 이러한 파일을 변환하는 프로그램을 이용하게 된다. Gingco사(<http://www.gingco.de/wap>)에서 제작한 변환프로그램을 이용하거나, 온라인 변환프로그램을 이용해서 GIF/JPEG/BMP파일을 WBMP파일로 전환하는 것도 하나의 방법이 될 수 있다. 본 논문의 경우

아래의 프로그램으로 이미지를 변환해서 사용하였다. 주의할 점은 이미지 작성시 1492byte를 넘으면 안 된다는 점이다. 이는 휴대폰의 자원문제로 야기되는 아쉬운 점이라고 할 수 있다.



<그림 8> pic2wbmp 이미지 변환틀

2.1.5 무선 인터넷을 위한 서버구축

기존 웹서버(IIS, APACHE)에서 웹 서비스를 하려면, MIME 타입을 선언해야 하듯이 WAP 서비스를 하기 위해선 기존 웹서버에 MIME TYPE을 선언함으로써 기존 유선과 같이 서비스 할 수 있다.

MIME TYPE을 선언해 주고 기존 SSL (Server-side Scripting Language) 언어와 연동시켜주면 기존 유선과 같은 서비스를 할 수 있게 된다. SSL언어로는 ASP, PHP, CGI, JSP 등이 있다. 본 논문은 이중에 Microsoft사의 ASP(Active Server Page)를 사용하여 구현할 것이다.

<표 1> WML을 위한 MIME TYPE 내용

확장명	MIME TYPE	용도
wbmp	image/wbmp	wml 그림서식
wml	text/vnd.wap.wml	wml text서식
wmlc	application/vnd.wap.wmlc	wml 응용서식
wmls	text/vnd.wap.wmlscript	wmlscript text 서식
wmlsc	application/vnd.wap.wmlscript	wmlscript 응용서식

2.2 XML/EDI

2.2.1 XML/EDI의 필요성

XML/EDI는 인터넷 기반의 EDI를 구현하기 위한 개방 표준의 한 가지 접근 방법으로 XML/EDI 그룹에 의하여 제안되었다. 전통적인 EDI의 단점을 극복 할 수 있는 XML의 장점은 다음과 같다.

- (1) XML은 필요한 요소를 추출하여 DTD를 작성함으로써 다양한 형태의 EDI 문서도 교환이 가능하다.
- (2) EDI 표준의 변화 또는 사실 표준의 변화에 따라 추가적인 엘리먼트가 요구될 경우 DTD를 수정함으로써 쉽게 해결할 수 있다.
- (3) 교환을 위해 필요한 엘리먼트를 추출하여 자신에게 적절한 DTD를 개발함으로써 XML은 문서 구조정보를 가질 수 있다. 이는 EDI 시스템과 데이터베이스와의 상호 연계를 위한 경우에도 XML/EDI 태그가 바로 데이터베이스의 스키마로 매칭될 수 있기 때문에 직접적으로 연계될 수 있어 중요한 의미를 가지며, 검색의

경우에도 강력한 기능을 발휘할 수 있다.

2.2.2 XML/EDI

XML/EDI는 기존의 EDI 문서가 가지고 있는 거래정보에 대하여 필요한 엘리먼트를 추출하여 XML의 DTD로 정의하고, XML의 특성인 인터넷을 통신 수단으로 사용하여 EDI를 구현하는 것이다. 기존의 EDI에서는 전송되는 메시지 중에서 데이터 항목을 분리하고 식별하기 위해서 독특한 세그먼트 식별자를 사용하여 왔다. XML/EDI에서는 이러한 세그먼트 식별자를 XML DTD로 정의하여 태깅된 데이터를 교환하도록 하는 것이다.

XML/EDI 시스템은 3-tier 구조로 이루어져 있다. 중간계층(XML 데이터 파일들)에서 XML 데이터는 Web 기반의 응용 프로그램에 XML 데이터를 제공해, 응용 어플리케이션이 XML이라는 표준화된 방식으로 데이터를 주고 받는 것을 가능하게 한다[11].

2.2.3 각 모듈별 특징

① 질의처리모듈

JAVA, ASP, VB, VC 등을 사용하여 DB에 있는 자료들을 XML 파일로 변화하여 저장하는 기능을 수행한다. 이때 저장된 파일들은 중간 계층의 데이터로 사용된다.

② 글로벌 저장소(Global Repository)

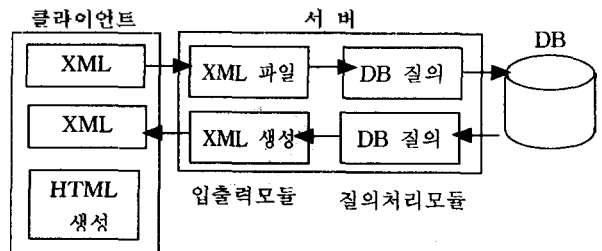
정보 교환에 필요한 문서의 DTD, 문서의 형태를 정의하는 XSL(eXtensible Style Language), EDI 파일의 데이터 위치정보를 가지고 있는 XML 템플릿 등을 담고 있으며, 수신자와 송신자 사이에 존재하여 송신측과 수

신측이 공유하여 사용하게 된다.

③ 입출력모듈

Client가 XML/EDI 문서를 조회, 작성, 편집할 수 있는 환경을 제공한다. Thin 클라이언트가 가능하도록 Java 또는 ASP 등을 사용하여 웹 브라우저만으로도 작업이 가능하다. 클라이언트는 입출력모듈을 통해서 XML 문서에 접근할 수 있으며 글로벌 저장소에 있는 DTD, XSL 등을 참조하여 각각의 클라이언트가 원하는 방식으로 작업을 할 수 있도록 한다.

DB에 저장되어 있는 제품 데이터는 XML 파서인 MSXML 파서를 통해 분석이 되고 XML의 표준 API인 DOM(Data Object Model)을 사용하여 각각의 XML 태그에 담긴 정보를 분석하게 된다. 이렇게 분석된 데이터는 질의처리 모듈에 의해서 데이터베이스에 질의를 수행하여 결과값을 XML 형태로 전달한다. 입출력된 모듈은 XML로 전달된 결과값을 클라이언트에게 전달하며 클라이언트는 브라우저에 내장된 파서를 사용하여 XML을 분석하여 사용자가 볼 수 있는 형태의 HTML로 변환하여 보여준다(<그림 9> 참조).



<그림 9> XML/EDI를 사용한 클라이언트/서버 질의처리 과정

3. WML, XML/EDI를 통한 Legacy 시스템과의 연동 프로토타입

3.1 WML, XML/EDI 프로토타입 시스템

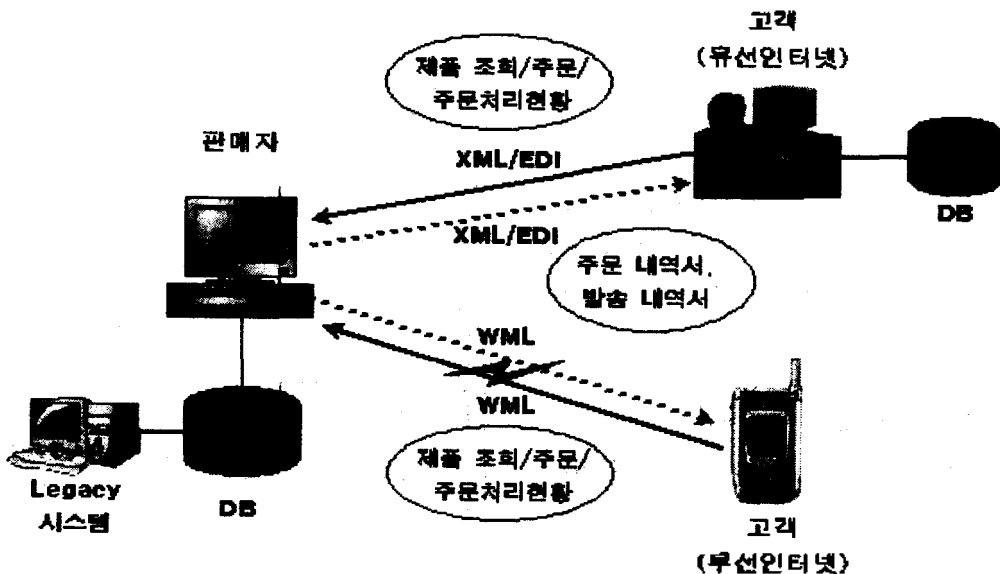
무선망으로 접속시 고객과 판매자는 WML 코드를 통하여 각각의 DB로 연결이 되며 유선망의 경우에는 XML/EDI를 통해 상호 연동이 이루어진다(<그림 10> 참조).

고객과 판매자는 휴대폰을 사용하여 무선 인터넷망에 접속할 경우 WML 코드로 데이터를 주고 받을 수 있다. 제품을 주문할 경우 XML/EDI를 사용하여 XML파일을 첨부한 Email을 판매자로부터 받는다. 유선 인터넷으로 접속할 경우 XML/EDI로 데이터를 주고받게 된다. 본 연구에서는 WML, XML/EDI를 통한 Legacy 시스템과의 연동프로토타입을 구현했는데 구현 환경은 다음과 같다.

- 하드웨어 : Pentium3 933Mhz, 384M
- 운영체제 : Windows 2000 Professional
- 개발도구 : Visual InterDev 6.0 SP5
- Web Server : IIS 5.0
- DBMS : MS-Access 2000
- 애플레이터 : Internet Explorer 6.0, Klondike WAP Browser
- 개발언어 : ASP(Active Server Page) XML, XSL, HTML, WML Visual Basic 6.0

3.1.1 고객의 주문과정

고객이 접근할 수 있는 메뉴는 판매자나 고객간의 정보교환 및 질문과 답변 게시판으로 제품에 관한 문의를 할 수 있는 곳이며 판매자가 게시판의 글을 읽고 답변을 남길 수 있다. 무선 인터넷망을 통해 질문과 답변이 이루어지기 때문에 고객과 판매자 모두 게시판

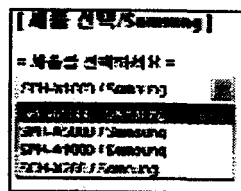
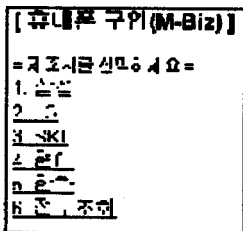


<그림 10> WML, XML/EDI를 이용한 주문관리 프로토타입 시스템 구성도

접근이 용이하며 고객에 대한 반응속도가 빨라지게 된다. 판매자의 빠른 답변은 고객의 신뢰성 향상에 도움을 준다. 고객과 판매자 사이의 거래 데이터는 유선망에서는 XML/EDI에 의해 DB에 저장되며 무선망의 경우에는 WML↔DB로 연동되어 저장된다. 게시판 DB는 판매자의 서버에 위치해 있으며 WML과 ASP 코드 및 MS Access DB를 사용하여 연결된다.

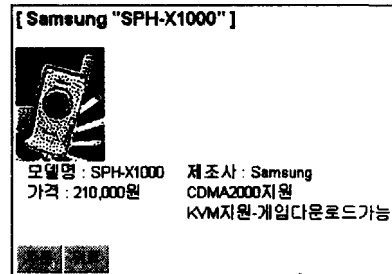
① WML, XML/EDI를 이용한 주문 및 주문처리현황 조회

고객이 휴대폰(WML) 혹은 기존 인터넷망(XML/EDI)을 통해 판매자의 서버에 접속하면 휴대폰을 주문할 수 있는 메뉴가 나타난다. 휴대폰 주문메뉴에서 제조회사를 선택하면 제품을 선택하는 메뉴가 나온다(<그림 11> 참조).



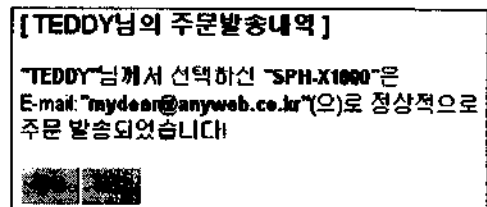
<그림 11> 제조회사 선택/제품선택 메뉴

제품을 선택하면 제품의 제조회사, 제품모델명, 가격과 특징, 주문가능한 재고수량, 제품사진이 나타나게 되는데 재고가 부족한 경우에는 “주문이 불가능 합니다”라는 메시지가 나오게 된다. 이 때의 데이터는 전부 판매자 DB에서 읽어오게 되며 사용자가 선택한 제품명을 키워드로 DB를 조회한다.



<그림 12> WML 기반 제품조회/주문메뉴

하단의 “주문”버튼을 누르면 고객의 휴대폰 화면에 주문 내역서가 나타나게 되며 판매자의 DB에 고객의 주문 기록이 저장된다.



<그림 13> 고객 휴대폰의 거래 내역

유선망의 (인터넷)경우에는 XML/EDI로 판매자의 DB에 주문내역이 저장된다.

② 판매자의 주문확인 및 Legacy 시스템 연계

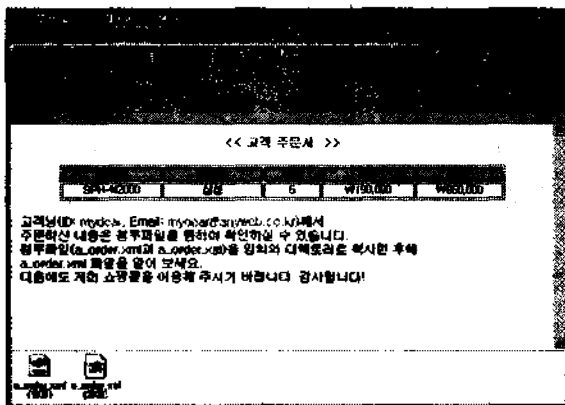
고객의 제품주문 후 판매자는 DB에 저장된 데이터를 참조하여 고객이 정상적인 과정을 통해 주문을 하였는가를 확인한다. 비정상적인 과정을 통해 주문하였을 경우 고객에게 주문취소 메일을 보내며 정상적인 주문임을 확인하였을 경우, 주문처리과정(발송·기각·대기 등)을 DB에 입력한다.

고객의 주문처리 과정이 완료되면 판매자는 Email을 통해 주문내역서를 발송한다. 이

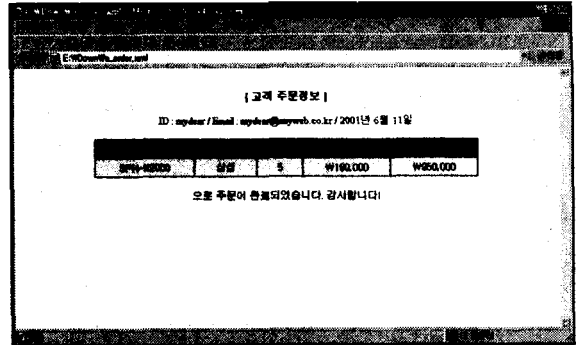
때에는 무선망으로 주문을 하였을 때도 XML/EDI를 사용하여 유선망으로 전송한다. 고객 입장에서 주문내역서를 보관해야 하는데 판매자로부터 WML 형식으로 주문내역서를 받을 경우 고객 DB에 저장하기 어렵기 때문이다. 이 때 사용자의 주문내역서는 XML과 XSL로 구성되어 있다. 판매자가 기존 시스템(Legacy 시스템)을 가지고 있다면, 같은 DB를 사용함으로 기존 시스템과의 연동이 가능하다.

③ 고객이 주문내역서 확인 후 고객 DB 저장 (XML/EDI)

판매자의 주문과정 처리가 완료되면 고객은 Email을 통해 XML과 XSL 파일을 받을 수 있다. 전송된 파일을 조회하는 코드는 HTML이며 첨부파일을 열면 XML 파일을 고객의 DB에 업데이트가 되며 열람도 가능하다. <그림 14>는 마이크로소프트사의 메일계정 프로그램인 아웃룩(Outlook Express)을 통해 주문서를 확인한 것이고, <그림 15>는 고객 DB에 업데이트된 XSL 스타일시트가 적용된 XML파일의 화면이다.



<그림 14> HTML 폼과 첨부 파일



<그림 15> XML에 XSL을 적용한 화면

<그림 16>은 XML DTD와 바디를 판매자의 서버에서 생성하는 코드를 보여준다. DTD의 속성은 고객 이름, Email, 주문일시, ID, 주문제품과 가격 등의 데이터 집합으로 구성되어 있다.

```

sql="select * from mobile where p_num=&request('p_num')
set rs=Server.CreateObject("ADODB.Recordset")
rs.Open sql.conn,1

id="customer"
//request('id')
name="상표"
request('name')
email="mydear@anyweb.co.kr"
//request('email')
model=rs("model")
company=rs("vender")
price=rs("price")
spec=rs("spec")
stock=rs("stock")
sell_check=rs("sell_check")

sql2="update mobile set stock=&stock&,"sell_check="&sell_check&
" where p_num=&request('p_num')
conn.execute(sql2)
%"

<%
Sxml = "xml version = &amp;quot;1.0&amp;quot; encoding=&amp;quot;euc-kr&amp;quot;?" &amp;
chr(13)
Sxml = Sxml +
"&lt;?xml-stylesheet type='text/xsl&amp;quot;"
href=&amp;quot;http://202.30.101.65/web/wml/m_order.xsl&amp;quot;?&amp;quot;" &amp;chr(13)
Sxml = Sxml + "&lt;!DOCTYPE DOCUMENT [" &amp;chr(13)
Sxml = Sxml + "(ELEMENT CUSTOMER (CUSTOMER)-&gt;" &amp;chr(13)
Sxml = Sxml + "(ELEMENT CUSTOMER (NAME,EMAIL,DATE,ORDERS))-&gt;" &amp;chr(
13)
Sxml = Sxml + "(ELEMENT NAME (IDNAME))-&amp;quot;" &amp;chr(13)
Sxml = Sxml + "(ELEMENT EMAIL (ENAME))-&amp;quot;" &amp;chr(13)
Sxml = Sxml + "(ELEMENT IDNAME (#PCDATA))-&amp;quot;" &amp;chr(13)
Sxml = Sxml + "(ELEMENT ENAME (#PCDATA))-&amp;quot;" &amp;chr(13) //이항출력
</pre

```

<그림 16> XML DTD 생성

<그림 17> Sxml 라인은 고객에게 전송하는 XML의 DTD와 바디 코드 파일을 생성하

는 부분으로 고객의 이름, Email, 구매제품의 모델, 제조사, 가격, 구입일자 등의 정보를 포함하고 있다. XML 파일과 동시에 전송되는 HTML 파일도 Content 라인에서 생성되어 전송이 된다.

```

$xml=Replace($xml,"&quot;","&#34;")
Content = "<DOCUMENT>" &chr(13)
Content = content * "<CUSTOMER>" &chr(13)
Content = content * " <NAME>" &chr(13)
Content = content * " <IDNAME>"&name&"</IDNAME>" &chr(13)
Content = content * " <NAME>" &chr(13)
Content = content * " <EMAIL>" &chr(13)
Content = content * " <ENAME>"&email&"</ENAME>" &chr(13)
Content = content * " <EMAIL>" &chr(13)
Content = content * " <DATE>"&year(now())&"년 "&month(now())&"월 "&day(now())&"일"</DATE>" &chr(13)
Content = content * " <ORDERS>" &chr(13)
Content = content * " <ITEM>" &chr(13)
Content = content * " <MODEL>"&model&"</MODEL>" &chr(13)
Content = content * " <COMPANY>"&company&"</COMPANY>" &chr(13)
Content = content * " <PRICE>"&price&"</PRICE>" &chr(13)
Content = content * " <SPEC>"&spec&"</SPEC>" &chr(13)
Content = content * " </ITEM>" &chr(13)
Content = content * " </ORDERS>" &chr(13)
Content = content * " </CUSTOMER>" &chr(13)
Content = content * " </DOCUMENT>" &chr(13)
  
```

<그림 17> XML 바디 생성코드

```

<?xml version="1.0" encoding="euc-kr" ?>
<?stylesheet xmlns:xsl="http://www.w3.org/TR/W3-XSL" ?>
<xsl:template match="/">
<html>
<head>
</head><br/>
<body>
<center>
<font size = "4" color = "BLUE"><b>고객 주문정보</b></font>
<br/>
<xsl:for-each select="DOCUMENT/CUSTOMER">
ID : <xsl:value-of select="NAME/IDNAME"/><br/>
Email : <xsl:value-of select="EMAIL"/><br/>
<xsl:value-of select="DATE"/>
<br/>
<table width="600" border="1" cellpadding="2" cellspacing="0"
bordercolorlight="#000000" bordercolordark="#FFFFFF" align="center"
style="font-family: 굴림; font-size: 9pt">
<tr>
<td bgcolor="#669900" align="center">제품명</td>
<td bgcolor="#669900" align="center">모델</td>
<td bgcolor="#669900" align="center">제조사</td>
<td bgcolor="#669900" align="center">가격</td>
<td bgcolor="#669900" align="center">구입일자</td>
<td bgcolor="#669900" align="center">합계</td>
</tr>
<xsl:for-each select="ORDERS/ITEM">
<tr bgcolor="#FFFFFF">
<td align="right"><xsl:value-of select="PRODUCT"/></td>
<td align="right"><xsl:value-of select="MODEL"/></td>
<td align="right"><xsl:value-of select="COMPANY"/></td>
<td align="right"><xsl:value-of select="NUMBER"/></td>
<td align="right"><xsl:value-of select="PRICE"/></td>
<td align="right"><xsl:value-of select="TOTAL"/></td>
</tr>
</xsl:for-each>
</table>
<xsl:for-each><p>
<font size="3" color="black">으로 주문이 완료되었습니다. 감사합니다
</font>
</xsl:for-each>
</center>
</body>
</html>
</xsl:template>
</xsl:script>
  
```

<그림 18> XSL 코드

<그림 17>은 고객의 주문 데이터를 DB에서 참조하여 주문내역서의 XML 바디를 생성하는 코드로 위의 DTD와 함께 하나의 파일로 판매자의 서버에 생성되고 고객 DB로 발송된다. XML 바디는 XML 트리를 이루는 메인 코드이다.

첨부된 XSL은 XML에 스타일시트를 적용하여 테이블과 색상 등의 표현을 가능하게 해준다. 이러한 과정을 통해 XML/EDI로 고객에서 발송된 파일은 차후 고객이 자신의 주문내역을 확인할 때 고객 DB를 통해 언제든지 열람이 가능하다는 장점이 있다(<그림 18> 참조).

4. 결론 및 향후 연구방향

본 논문에서 프로토타입으로 제시한 무선 인터넷을 통한 제품구매는 먼 얘기가 아니다. 현재도 비슷한 서비스가 실시되고 있고 결제 방법의 개선이나 보안문제가 해결되면 손쉽게 휴대폰으로 제품을 검색하고 구매하는 시대가 곧 오리라 생각된다.

휴대폰을 통하여 제품을 구매하고 그 결과를 Email을 사용한 XML/EDI로 확인할 수 있는 절차는 휴대폰과 유선망 뿐만 아니라 모든 제품에 적용될 수 있다. XML/EDI의 적용으로 판매자와 고객의 DB에 바로 수정이 가능하고, 고객은 주문내역과 주문 진행상황 등을 바로 확인할 수 있으므로 제품구매의 신용도에 더욱 무게를 줄 수 있지 않을까 생각된다. 또한 휴대폰을 통한 전자상거래 구현은 시간, 장소에 구애받지 않으므로 발전의 가능성이 더 높다고 할 수 있다. 판매자는 유선 인터넷망에서 XML/EDI를 기반으로 데이터를 교환하고 무

선망의 경우에는 WML을 통해 데이터를 교환한다. 본 논문에서는 XML 데이터를 데이터베이스와 연동하는 방법을 유선망에서 구현하였으며, 무선망에서는 WML을 통해 데이터베이스와 연동하는 것을 구현해 보았다. 차후 추가하고자 하는 부분은 휴대폰 결제시 SSL(Secure Socket Layer) 보안 프로토콜의 적용 등을 통한 신뢰성 높은 결제 시스템을 구축할 것이다. 또한 SCM의 영역을 다른 Value Chain으로 확장하여 WML과 XML/EDI를 공급자-하청업체 등에 적용하려 한다.

현재 Mobile의 문제점은 사용상의 불편함

과 전송속도, 요금문제 등을 생각할 수 있는데, 차후 기술의 발달로 좀 더 다양한 콘텐츠를 서비스 할 수 있을 것이고, 시간, 장소에 구애받지 않기 때문에 실시간으로 요구되는 예약, 증권, 실시간 정보 등을 무선으로 서비스할 수 있을 것이다. 얼마 전 CDMA2000/1x 서비스가 실시되어 144Kbps의 속도로 데이터 전송이 가능해졌으며 패킷 요금제 도입으로 상거래의 기반이 점점 갖추어지고 있다. 요금이 조금 더 합리적으로 결정된다면 앞으로 휴대폰을 통한 전자상거래 시장은 무궁무진하게 발전될 것이라 믿는다.

참고문헌

- [1] [김대송, 1999] 김대송, 김화진, 무선인터넷 어플리케이션 프로그래밍, 삼양출판사, 1999.
- [2] [홍준호, 2000] 홍준호, 송건철, 김정석, about WAP, 영진.com, 2000.
- [3] [송호성, 2000] 송호성, 안우길, 홍순성, ASP3.0 Bible, 영진.com, 2000.
- [4] [마틴 프로스트, 2001] 마틴 프로스트 저, WML과 WML스크립트 시작하기, 한빛미디어, 2001.
- [5] [강은선, 1999] 강은선, XML 프라이어, 파워북, 1999.
- [6] [박종훈, 1998] 박종훈, XML 응용과 제품개발 현황, 경영과 컴퓨터, 1998.
- [7] [이태억, 1999] 이태억, "Supply Chain Management 개념과 전망", 대한산업공학과/한국경영과학회 '98춘계공동학술대회, 1999.
- [8] [정경희, 1998] 정경희, 문서기술언어-SGML, HTML, XML, 1998.
- [9] [이원석, 1998] 이원석, XML 문서제작, KRIC, 1998.
- [10] [Frank Boumphrey, 1999] Frank Boumphrey 외 11인, "XMI APPLICATIONS", 정보문화사, 1999.
- [11] [David RR webber, 1998] David RR webber, Introduction XML/EDI Frameworks, Electronic Markers, Vol. 8~No.1, 1998.
- [12] [James Clark, 2000] James Clark, "Comparison of SGML and XML", <http://www.w3.org/TR/NOTE-SGML-XML.htm>.

- [13] [Microsoft홈페이지, 2001] "XSL Tutorial",
<http://www.microsoft.com/XML/xml/tutorial>
- [14] [Natanya Pitts, 2001] Natanya Pitts, XML Black Book 2nd Edition, 영진.com, 2001.
- [15] [Simon St, 1998] Simon St. Laurent, XML : A PRIMER, IDG Books worldwide, Inc, 1998.
- [16] [Alan Kotok, 2000] Alan Kotok, "Introduction to XML and EDI",
<http://xml.com/pub/1999/08/edi/index.html>

저자소개

안동률(e-mail: mydear@anyweb.co.kr)

명지대학교 산업공학과 학사

현재 명지대학교 산업시스템공학부 석사과정

관심 분야 : 전자상거래, 정보통신(무선인터넷)

박정선(e-mail: jspark@mju.ac.kr)

서울대학교 산업공학과 학사

KAIST 경영과학 석사

텍사스 주립대학(오스틴) MIS 박사

현재 명지대학교 산업시스템공학부 부교수

관심 분야 : 전자상거래 응용/보안, 에이전트 개발, DB응용