

PC 및 PDA 브라우저 지원을 위한 XML 기반의 웹 콘텐츠 개발 사례 연구

김경아* · 옹환승**

*이화여자대학교 정보과학대학원 정보과학과 · **이화여자대학교 컴퓨터학과

요 약

PDA, 휴대폰 등 무선 기기의 발달과 대중화와 더불어 무선 인터넷의 사용이 활성화되고 있다. 이에 따라 기존 PC 브라우저뿐만 아니라 PDA 등에서 검색할 수 있는 웹 콘텐츠의 수요가 증가하고 있으며, 풍부한 양질의 콘텐츠 제공을 위해 이미 제작된 유선 인터넷용 콘텐츠를 무선 인터넷에도 제공하기 위해 콘텐츠 변환 솔루션과 마크업 언어 통합 등의 많은 연구가 이루어지고 있다. 본 논문에서는 이기종 단말기를 지원하는 웹 콘텐츠 제작을 위해서 XSLT 엔진으로서 공개 소프트웨어를 사용하기 때문에 고비용의 부담이 없고, 사용하기에도 간편하여 기존 HTML을 사용하는 웹 프로그래머들도 DTD와 XSL에 대한 기본적인 사항만 학습하면 쉽게 사용할 수 있는 XML 기반의 웹 콘텐츠 제작 기법을 사례 개발을 통해 제안한다.

A Case Study of Developing XML-based Web Contents Supporting PC and PDA Browser

Kyung-A Kim* · Hwan-Seung Yong**

ABSTRACT

Due to rapid advance of wireless communication technology and popularization of wireless devices, demand on wireless internet contents is gradually increasing. Therefore, there are many researches and solution developments to provide good qualified contents quickly for wireless internet. For example, researches into converting wired web contents into wireless web contents or using integrated markup language like XHTML basic to make contents. In this paper, I propose a method to develop XML-based web contents which uses PHP language for data fetch from MySQL database. This method use open source software for a cost saving. Due to use of PHP extension as a XSLT engine, this method is very easy to apply. For a example of this method, a web content of movie information is implemented for PC and PDA browser. Developing XML-based web contents is useful not only for supporting devices of multiple type, but also for rapid changes of user interface design and for exchange of contents between web sites.

1. 서론

PDA(Personal Digital Assistant), 휴대폰 등의 무선 기기의 발달과 대중화에 따라 무선 인터넷의 사용이 활성화되고 있다. 가트너그룹에 의하면 2000년에서 2010년까지의 주도 기술로 XML, WAP, Webtop, Bluetooth, E-Cash 등 무선 인터넷 관련 요소 기술들을 그 핵심으로 선정하고 있으며, 2002년에는 6억 개의 단말기에서 무선 인터넷을 이용할 것이라고 예상하고 있다 [1]. 이는 기존 PC 브라우저뿐만 아니라 PDA 등에서 검색할 수 있는 웹 콘텐츠 수요의 증가를 가져올 것으로 판단된다.

양질의 무선 인터넷 콘텐츠를 신속하고 효율적으로 개발하기 위해 기존의 수많은 인터넷 콘텐츠들 중 무선인터넷에 적합한 콘텐츠 선택에 대한 연구와 이러한 콘텐츠들을 무선 단말기 상에서 접근하고자 하는 요구는 증대되고 있고, 이러한 요구에 부응하기 위해 유선 인터넷의 콘텐츠를 무선 인터넷에서 사용하기 위한 다양한 연구와 노력들이 수행되고 있다. 즉, 유선 인터넷 콘텐츠를 자동으로 무선 인터넷 콘텐츠로 변환해 주는 콘텐츠 변환 솔루션[2,3,4,5]의 개발이라든지, XML 기반의 유무선 단말기용 웹 문서 생성기[6] 개발에 관한 연구, 무선 인터넷 서비스를 위한 통합 마크업 언어의 설계[7]에 관한 연구, WAP(Wireless Application Protocol) forum이 XHTML(eXtensible Hyper Text Markup Language) basic을 WAP2.0의 기반으로 하는 것[8]들이 그것이라 하겠다.

그러나, 현실적으로 XML(eXtensible Markup language)로 신규 사이트를 구축하는 것에 대한 부담, 콘텐츠 변환 솔루션의 고가의 비용부담 등

으로 인해 유선 인터넷을 위한 웹 콘텐츠와 무선 인터넷을 위한 콘텐츠를 이중으로 제작하고 있다.

XML기반의 웹 콘텐츠를 제작하는 것이 유무선 통합 콘텐츠를 제작하는 가장 이상적인 방법임을 주장하거나[2], XML의 장점이나 도입가치를 설명하는 연구 논문이나 서적들은 XML을 이용해 다양한 단말기를 지원하는 콘텐츠를 제작할 수 있다고 설명하고 있다[2,9,10,11].

그러나, 막상 XML 기반의 웹 콘텐츠 제작을 위해 기존에 제작된 HTML기반의 웹 페이지들을 어떻게 쉽게 옮겨 가게 할 것인가, 현재 사용하고 있는 데이터베이스와의 연동은 어떻게 할 것인가 하는 실무적인 입장에서 보면 명쾌한 해답을 얻기 힘들다.

본 논문에서는 웹 콘텐츠 제작에 널리 쓰이고 있는 MySQL, PHP를 이용한 웹 콘텐츠를 XML 기반의 웹 콘텐츠로 제작하기 위한 방법을 소개한다.

본 논문의 목표는 첫째, 데이터의 구조와 표현을 분리하는 XML의 특성을 이용하여 이기종 단말기를 지원하는 웹 콘텐츠의 제작이 가능함을 보여 주는 것이다. 둘째, XML 파서와 XSLT 엔진을 공개 소프트웨어로 사용하여 금전적 비용이 없도록 한다. 셋째, XML을 사용하더라도 데이터베이스를 변경하거나 데이터의 구조를 변경하지 않고 기존 데이터를 그대로 이용하도록 한다. 넷째, XSLT를 사용함에 있어 PHP 확장 함수(extension)에서 지원하는 wrapper 함수를 사용하여 기존 웹 프로그래머의 XML 기반 기술로의 진입이 용이한 방법을 제안한다.

2장에서는 관련 연구로서 유무선 콘텐츠 변환기와 마크업 언어 통합을 위한 XHTML basic에 대해 알아보고, XML 기반 웹 콘텐츠 제작에 필요한 기반 기술들을 살펴본다. 3장은 본 논문이

제안하는 PC 및 PDA 브라우저를 지원하기 위한 XML 기반의 웹 콘텐츠 제작 방법을 소개하고, 4장과 5장에서는 영화정보 콘텐츠를 설계, 구현함으로써 본 논문의 제안을 설명할 것이다. 6장은 결론과 향후 연구 방향에 대해 언급하고자 한다.

II. 관련 연구

유선 인터넷과 무선 인터넷뿐만 아니라 무선 인터넷 안에서도 각 기종간 또는 업자간 사용하는 표준이 다르고 콘텐츠 개발에 사용하는 언어도 WML(Wireless Markup Language), HDML(Handheld Device Markup Language), mHTML(mobile HTML), sHTML(small HTML), cHTML(compact HTML) 등의 다양한 언어가 존재하고 있다. 하나의 콘텐츠로 여러 단말기에 제공하기 위한 연구는 크게 콘텐츠 변환기에 관한 연구[2,3,4,5], 통합 마크업 언어에 관한 연구[7,8], XML 기반 콘텐츠 제작에 관한 연구 등으로 나누어 볼 수 있다.

2.1 유무선 콘텐츠 변환기 (Contents Converter)

기존의 유선 사이트의 콘텐츠를 개별 클라이언트 단말기에 적합한 포맷과 환경으로 적절히 변환할 수 있는 방안에 대한 필요가 증대되고 있고, 여러 벤더들이 상용 변환 솔루션을 이미 시장에 공급하고 있다. 일반적으로 유선 사이트의 콘텐츠를 단말기에 따른 적절한 포맷과 환경으로 변환 가능하게 하는 툴을 Converter 또는 Transcoder라고 한다[2].

콘텐츠 변환기 (Contents Converter)에 관한 국내 연구로는 HTML 태그를 WML태그로 변환하는 변환표를 만들고 이 변환표에 따라 HTML을 파싱(Parsing)하여 실시간으로 WML로 변경

하는 콘텐츠 변환기를 구현한 연구가 있다[4]. 그러나 HTML을 WML로 변환하는 것에 그쳐 WAP 프로토콜을 사용하지 않는 다른 표준을 사용하는 경우는 사용할 수 없는 한계가 있다.

콘텐츠 변환기는 기본적으로 변환의 대상이 되는 사이트의 HTML 소스로부터 텍스트를 추출해서 클라이언트 단말기에 적합한 마크업 언어로 변환하는 과정을 따른다.

콘텐츠 변환기는 벤더가 미리 설정한 표준변환 규칙에 따라서 소스 HTML페이지를 자동적으로 변환하는 자동형 변환기(Automated Converter)와 변환 규칙의 설정에 개발자 또는 사이트 운영자의 판단이 요구되는 설정형 변환기(Configurable Converter)로 분류된다.

자동형 변환기는 WAP Gateway에 변환기(converter)가 포함되는 경우로 임의의 HTML 페이지에 대해 변환을 시도하게 된다. 따라서, 복잡한 멀티미디어 정보를 포함한 페이지는 변환에 실패할 수 있고, 디스플레이 관점에서 QoS(Quality of Service)를 보장하기 어렵고 예러가 발생할 소지가 크다. 시중에 나와 있는 제품으로는 Phone.com의 WAP Gateway, Argo의 Acti-gate 등이 있다.

설정형 변환기는 개발자나 사이트 운영자가 유선 사이트에 대한 변환 규칙을 설정할 수 있다. 텍스트 정보 중 무선 사이트로 변환할 내용을 지정할 수 있고, 이미지가 포함된 경우는 이미지의 변환과 관련된 설정을 지정할 수 있다. Oracle 9i Application server Wireless Edition, IBM의 Websphere Everyplace Suite, OpenTV사의 Spyglass Prism 등의 제품이 나와 있다.

콘텐츠 변환기를 이용하면 다양한 단말기, 다양한 표준에 대한 개별 개발 비용이 절감되므로 전체 개발비용이 적어지고, 뿐만 아니라, 단말기 각

각에 대한 개발이 필요 없으므로 빠른 시장 진입이 가능하다는 장점이 있다. 그러나, 현재 시장에서 구할 수 있는 콘텐츠 변환 솔루션의 비용이 수천만원 대 이상의 비교적 고가의 비용이 들고, 아무리 훌륭한 콘텐츠 변환 솔루션이라 해도 신규 개발에 비해 개발의 유연성과 디스플레이 품질이 떨어질 수 밖에 없다는 단점이 있다.

2.2 마크업 언어 통합 - XHTML Basic

XHTML(The Extensible Hyper Text Markup Language)은 1998년 5월, 미국 산호세에서 열린 HTML 워크샵("Shaping the Future of HTML")에서 당시 W3C(World Wide Web Consortium)에서 주력하고 있는 XML의 형태를 가진 새로운 HTML로 제안되었다. XHTML은 HTML 4.0의 기능을 수용하며, 기존의 브라우저에서도 사용할 수 있으며, XML의 응용(application)으로 개발되었다[12].

HTML이 SGML의 한 응용이라면, XHTML은 XML의 한 응용으로서 HTML의 기능을 가지는 마크업 언어로 "XML+HTML"이라고 볼 수 있다. 기존의 HTML의 명령어들을 그대로 유지하면서 XML포맷에 맞추도록 필요한 요소들을 가미한 언어 체계이다.

XHTML을 사용하면 Mobile phone, PALM, Set-top box 등을 지원할 수 있으며, 전자상거래 등에서 사용하는 특정양식을 XHTML의 새로운 양식 옵션을 통하여 지원할 수 있다는 데에 커다란 장점이 있다[13].

WAP Forum에서는 XHTMLMP(XHTML Mobile Profile)를 WAP 2.0을 위한 마크업 언어로 채택하였는데, XHTMLMP는 W3C의 XHTML Basic을 기본으로 하여 확장된 것이다[16]. XHTML basic은 XHTML 1.0의 무선 인터넷 버전이라 할 수 있다. XHTML에서 요구

하는 규칙에 대한 엄수는 어느 정도 균일한 사이트의 렌더링을 보장한다. 기존의 WML로 구현된 무선 사이트의 경우, 단말기마다 균일한 렌더링을 보장하지 못했다. 이로 인해 개발자 입장에서는 사이트가 지원할 다양한 단말기를 직접 이용해 테스트를 거쳐야 하는 부담이 존재했다. XHTML은 균일한 렌더링을 보장하므로 개발자에게는 과도한 테스트에 대한 부담을 줄여주고 다양한 단말기에 대한 지원을 용이하게 하는 잇점이 있다[8].

XHTML basic은 WML, HDML, cHTML 등의 다양한 무선 인터넷을 위한 마크업 언어의 통합이라는 측면에서 큰 의의와 효과를 발휘할 것이다. 그러나, XHTML을 사용한다 하더라도, 증권 정보나 뉴스처럼 비교적 표현 형식이 간단한 정보가 아닌 복잡한 유선 인터넷의 웹 콘텐츠를 단말기의 화면 크기가 작은 무선 단말기를 위해, 표현되는 정보의 양을 제한하거나 정보의 화면 배치를 달리하는 것과 같은 처리는 불가능하다.

2.3 XML 기반 웹 콘텐츠 제작 요소 기술

2.3.1 XML 기반 웹 콘텐츠오 HTML 기반 웹 콘텐츠의 비교

앞서 서론에서 언급했던 바와 같이, 콘텐츠 제 공업계에서는 이미 제작된 유선 웹 콘텐츠가 있음에도 불구하고 여러 가지 현실적인 이유(콘텐츠 변환기의 고비용, XML 전문가 부족, 난해한 기존 DB 공유 방법 등)로 PDA와 같은 무선 단말기를 지원하기 위해 새로 HTML 웹 페이지를 제작하는 경우가 많이 있다. 본 절에서는 이기종 단말기를 지원하기 위해 HTML 기반의 웹 콘텐츠를 제작하는 것과 XML 기반의 웹 콘텐츠를 제작하는 것이 어떤 장단점을 갖는지 알아본다.

XML과 HTML은 둘 다 마크업 언어(Markup Language)이긴 하지만, HTML은 SGML이나 XML처럼 마크업 언어로 정의된 하나의 문서구조

이며, XML은 SGML처럼 사용자가 원하는 문서의 구조를 정의할 수 있는 상위 개념의 마크업 언어이다[9]. 따라서, XML과 HTML을 평면적으로 비교하기 보다 쓰임새(응용분야)에 따라 비교해야 할 것이다.

XML과 HTML을 표현양식의 관점에서 비교하면 HTML은 태그 자체가 표현양식을 가지고 있는 반면, XML의 태그는 표현양식과 상관없는 자료의 구조를 나타낸다. XML 문서를 브라우저를 통해 HTML 문서처럼 보려면 표현양식을 가지고 있는 XSL과 결합해야 한다.

〈표 1〉 XML과 HTML의 비교

비교항목	XML	HTML	
태그	데이터의 구조	태그에 들어싸인 데이터의 표현양식	
종속성	스타일, 표현매체 비종속적	스타일, 표현매체 종속적	
스타일 언어	XSL	CSS	
다매체 지원	웹 디자인 추가	복수개의 XSL 제작	복수개의 CSS, HTML 제작
	프로그램 추가	데이터 추출용 Script 프로그램 1번만 개발	지원 단말기가 추가 될 때마다 Script 프로그램 추가 개발
	개발 소요 시간	XSL 추가제작 시간	HTML, CSS 디자인 추가시간 + Script 등 프로그램 추가시간

HTML은 데이터와 표현양식이 결합되어 있기 때문에 스타일을 변화시키려고 하거나 매체를 추가로 지원하려고 할 때 HTML자체를 다시 작성해야 한다. 반면, XML은 데이터와 표현양식이 분리되어 있기 때문에 XML 기반의 웹 사이트는 XSL만 변경해서 적용하면 되므로 스타일을 변경하거나 다른 종류의 단말기(예를 들면, PDA나 휴대폰 등)를 추가적으로 지원하려고 할 때 XML을 다시 작성할 필요는 없다.

HTML로 제작된 웹 콘텐츠를 PDA에서 서비스하도록 지원하려면 PDA의 화면 크기를 고려하여 HTML과 CSS를 별도로 디자인하여야 하고, 데이터 추출을 위한 PHP 등의 코드 추가 작업이 필요할 경우 새로 디자인된 html 파일에도 똑같은 PHP 코드 추가 작업이 이루어져야 한다. 지원 매체를 추가하기 위해 웹 디자이너와 프로그래머 모두 별도의 작업시간이 필요하다.

반면, XML 기반의 웹 콘텐츠의 경우는 지원 단말기를 추가해도 데이터가 변하는 것이 아니기 때문에, 이미 만들어진 xml 파일이 있는 경우 새로 추가된 단말기를 위한 xsl 파일 작업만 추가하면 된다. 동적으로 데이터를 추출해야 하는 경우라 하더라도, 이미 데이터를 추출하여 XML화 하는 프로그램이 작성되어 있을 것이므로 xml 파일과 관련된 추가 작업은 필요하지 않다. 당연히 추가 작업 시간도 새로 추가되는 단말기용 xsl 파일 작업에만 필요하게 된다.

XML 기반의 웹 콘텐츠 제작의 또 다른 장점은 웹 콘텐츠 제작 작업에 있어 디자이너와 프로그래머의 작업 영역 구분이 보다 뚜렷해진다는 점이다. HTML로 작업할 경우, 디자인이 끝난 html 파일에 프로그래머가 자바 스크립트나 PHP 등의 프로그램 요소를 추가하는 순서로 작업이 이루어진다. 디자인이 변경될 경우, 디자이너가 만든 html 파일에 다시 프로그램을 입히는 작업을 해야만 했다. XML을 사용할 경우는 디자이너와 프로그래머 사이에 DTD에 대한 정보만 공유하게 되면 각자 따로따로 디자이너는 xsl 파일을 작업하고, 프로그래머는 xml 파일 생성에 관한 작업을 하면 된다. 굳이 디자이너의 xsl 디자인 작업이 끝날 때까지 기다리지 않아도 되고, DTD가 변경되지 않는다면, 디자인 변경 작업은 프로그래머의 관여 없이 이루어질 수 있다.

2.3.2 XSLT 기본 개념

XSL(eXtensible Stylesheet Language)[15]은 XML의 스타일시트를 표현하는 언어이다. XSL은 세 부분으로 되어 있는데, XML문서의 변환을 위한 언어인 XSLT (XSL Transformations)[16], XSLT에서 XML문서의 부분을 접근하고 참조하는데 사용되는 언어인 XPath[17], XML을 출력 미디어(인쇄 매체)에 따라 보다 정교하게 출력 스타일을 적용할 수 있도록 하는 문서 기술 언어(Document description language)인 XSL Formatting Object[18]가 그것이다.

XSLT Process는 XML 문서와 XSLT Stylesheet를 읽어 들여서 새로운 XML문서나 XML문서의 일부분을 만들어낸다.

XSLT를 변환 기술로 활용할 수 있는 측면에서 보자면, XSLT의 주요 핵심 기능은 XML에서 원하는 정보를 추출하여 사용자가 원하는 형태로 변환해 주는 것이다. 크게 다음과 같은 세 가지 측면에서 이러한 기능의 중요성을 부각시킬 수 있다[10].

첫째, 맞춤 서비스와 퍼스널라이제이션[19] 부분에서 사용자의 다양한 요구 사항을 만족시킬 수 있는 핵심 기술이 될 수 있다. 둘째, POST-PC 시대를 맞이하여 다양한 기능을 가진 다양한 형태의 단말기들에 일관된 공통 정보를 단절성이 없는 서비스로 제공하기 위하여 단말기의 특성을 고려한 최적화된 콘텐츠를 제공할 수 있는 핵심 기술이 될 수 있다. 셋째, 내용이 자주 변경되는 웹사이트를 효과적으로 운영할 수 있는 기술을 제공할 수 있다.

컨텐츠 측면에서 보자면, XML과 이를 단말기에 적합하게 변화하여 주는 XSLT 기술은 기존에 유선 인터넷으로 제공하던 정보들을 단절 없이 무선 인터넷 분야에서도 제공해주기 위하여 도입할 수 있는 핵심 기술로 볼 수 있다.

2.3.3 expat과 sablotron 소개

expat은 James Clark이 이끄는 Thai Open Source Software Center Ltd.에서 1998년 처음 만든 C언어로 개발된 XML 1.0 파서(parser)이다[20]. Expat 2.0에 대한 작업이 <http://sourceforge.net/projects/expat/>에서 이루어지고 있으며, 현재 구할 수 있는 최종 버전은 2001년 7월에 만들어진 expat 1.95.2이며, Windows용과 Unix(Linux)용이 있다.

Python, Perl, Php와 같은 script 언어에서 쉽게 접근할 수 있는 API 함수들이 이미 개발되어 있다.

Apache를 사용할 경우 apache-1.3.7 이후 버전은 expat-lite 버전이 포함되어져 있고, apache-1.3.9 이후 버전은 php를 컴파일할 때 expat이 자동으로 포함된다[21].

Sablotron은 체코 공화국의 Ginger Alliance에서 만든 오픈 프로젝트이고, XSLT, DOM, XPath를 구현한 빠르고, 간결하고, 이식성 있는 XML toolkit이다. Sablotron 프로젝트의 목적은 W3C 스펙에 맞는 가볍고, 신뢰할 수 있고, 빠른 XML 라이브러리 프로세서를 만드는 것이다. Sablotron은 C++로 개발되었고, Solaris, FreeBSD, OpenBSD, HP-UX, 그리고 모든 Win32 Microsoft 플랫폼에서 동작한다. Sablotron은 James Clark의 expat을 XML 파서로 사용한다[22].

Sablotron은 커맨드 라인 인터페이스인 `sabcmd`의 제공과 더불어 C API와 Python, Perl, Php를 위한 확장 함수(extension or wrapper)의 사용이 가능하다.

Php 4.0.3 부터 sablotron extension이 추가되었다. Php 4.1 이후 버전에서는 `configure --enable-xslt --with-xslt=sablot` 옵션을 사용하면 sablotron extension을 사용할 수 있다.

III. PC 및 PDA 지원을 위한 XML 기반 웹 콘텐츠 개발 기법 제안

이 장에서는 본 논문에서 제안하는 XML 기반 웹 콘텐츠 개발 기법이 어떤 가정 하에 제안된 것이며, 개발 환경과 개발 도구로써 어떤 프로그램을 이용하는지 등의 제안 개요를 설명한다. 그리고 본 논문에서 제안하는 방법을 실증하기 위해 제작할 PC와 PDA에서 서비스 할 수 있는 영화 정보 웹 콘텐츠의 내용을 간략히 설명한다.

3.1 제안을 위한 전제

본 논문에서 제안하는 XML 기반 웹 콘텐츠 제작 방법은 기존 유선 인터넷을 통해 웹 서비스를 하고 있는 콘텐츠를 PDA용 무선 인터넷 서비스에 추가로 제공하는 것을 전제로 한다. 또한, 기존 웹 서비스 환경은 UNIX OS를 사용하고 데이터는 관계형 데이터베이스인 MySQL에 저장되어 있으며, 데이터 추출을 위해 서버사이드 스크립트 언어인 PHP를 사용하는 것을 전제로 한다.

XML 기반으로 전환할 때 PHP와 함께 가장 쉽게 활용할 수 있는 sablotron을 XSLT 엔진으로 사용한다.

3.2 제안 개요

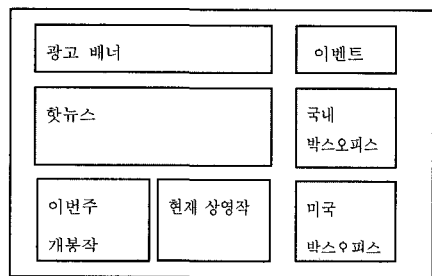
본 논문에서 제안하는 개발 환경은 PHP에서 API 함수를 제공하고 있는 expat과 sablotron 엔진을 각각 XML 파서와 XSLT 엔진으로 사용한다.

본 논문에서 제안하는 방법을 증명하기 위해 PC와 PDA에서 서비스할 수 있는 영화정보 웹 콘텐츠를 구현해 보도록 하겠다.

영화정보는 게시판 형식의 핫뉴스와 영화 소개 정보인 이번주 개봉영화, 현재 상영영화, 순위 리

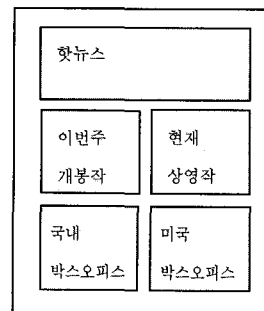
스트를 보여주는 국내 박스오피스, 미국 박스오피스 정보로 구성된다.

PC와 PDA의 화면 크기가 다른 것을 고려하여, PC용 서비스 화면과 PDA용 서비스 화면의 레이아웃을 다르게 설계하였다. 핫뉴스 등 각각의 정보도 <table>, 등의 크기를 단말기에 맞도록 디자인하였다.



(그림 1) 영화정보 PC용 화면 레이아웃

(그림 1)은 PC 브라우저용 화면 레이아웃이다. 화면 상단에 광고 배너와 이벤트 이미지가 들어가는 것을 볼 수 있고, 전반적으로 세로 보다 가로가 긴 화면 배치를 가지고 있다.



(그림 2) 영화정보 PDA용 화면 레이아웃

(그림 2)는 PDA에서 볼 수 있는 화면 레이아웃이다. 화면이 PC에서 볼 수 있는 것 보다 작아

서 광고배너와 이벤트 이미지를 생략하고, 전반적인 화면 배치도 세로로 좁은 PDA 화면에 따라 세로로 길게 배치하였다.

3.3 개발 환경

시스템 환경은 UNIX OS에 데이터베이스는 MySQL을 사용하고 웹서버는 apache를 사용하였다. 환경설정은 아래 표와 같다.

<표 2> XML 기반 웹 콘텐츠 개발 환경

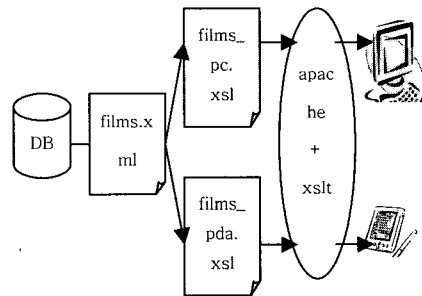
운영체제	UNIX
데이터베이스	MySQL
웹서버	apache 1.3.19
Script 언어	php 4.1.2
XML 파서	expat 1.95.2
XSLT 엔진	sablotron 0.95

본 시스템에서는 XML 파서로 expat을, XSLT 엔진으로 sablotron을 사용하는데, expat은 ISO-8859-1, US-ASCII, UTF-8의 세가지 encoding만을 지원하므로 한글 데이터를 처리하려면 libiconv 라이브러리를 먼저 설치해야 한다. libiconv는 Unicode가 아닌 텍스트를 교환하는 어플리케이션이 서로 다른 encoding을 사용할 때 code conversion을 제공하는 GNU(GNU's not UNIX) 라이브러리[23]이다.

expat과 sablotron을 설치한 다음 php를 다시 컴파일한다.

N. 시스템 설계

4.1 시스템 구성도



(그림 3) 시스템 구성도

본 논문에서 구현하는 시스템의 가장 중요한 요소는 XSLT를 이용해서 두 가지 다른 단말기에서 같은 정보를 다른 형태로 구현한다는 것이다.

영화정보 DB에서 데이터를 추출하여 films.xml을 생성하고 PC 브라우저용 xsl인 films_pc.xsl, PDA 용 xsl인 films_pda.xsl을 xslt processing을 통해 생성된 결과 화면으로 각각 PC 또는 PDA에서 서비스되도록 한다.

앞서 제안을 위한 전제에서 기술한 바와 같이 XSLT 엔진은 sablotron을 웹서버는 apache를 사용하고, 영화정보는 MySQL에 저장되어 있다.

4.2 영화정보 DB와 DTD 설계

본 논문에서 구현하는 영화정보 콘텐츠는 MySQL을 사용하며, '핫뉴스', '이번주 개봉작', '현재 상영작', '국내 박스오피스', '미국 박스오피스'의 다섯 가지 카테고리의 정보를 제공한다.

'핫뉴스' 정보는 hotnews 테이블에, '이번주 개봉작'은 openmovie 테이블에, '현재 상영작'은 playmovie 테이블에, '국내 박스오피스' 정보는

korbox 테이블에, '미국 박스오피스'는 usabox 테이블에 각각 저장되며, 각 테이블의 구조는 <표 3>과 같다.

<표 3> 영화정보 DB 스키마

테이블명	내 용	구 조
hotnews	핫뉴스	(일련번호, 제목, 내용)
openmovie	이번주 개봉작	(영화코드, 제목, 개봉일)
playmovie	현재 상영작	(영화코드, 제목, 개봉일)
korbox	국내박스오피스	(일련번호, 순위, 일자, 영화코드, 영화제목)
usabox	국내박스오피스	(일련번호, 순위, 일자, 영화코드, 영화제목)

XML에서는 문서의 구조를 DTD(Document Type Definition)를 통해서 정의한다. 문서의 구조를 나타내는 DTD를 먼저 작성한 다음, DTD에 맞춰서 유효한 XML문서를 작성해야 XML 파서에서 그 문서를 유효한 문서로 처리하게 되고, XSL과 연결할 때도 원하는 결과의 출력을 얻게 된다.

<표 3>에서 보는 바와 같이 영화정보 중 '이번주 개봉작'과 '현재 상영작', '국내 박스오피스'와 '미국 박스오피스' 정보는 각각 같은 데이터 구조를 가지고 있음을 알 수 있다. 따라서, 같은 데이터 구조를 가진 정보들은 같은 DTD를 공유하도록 한다.

<표 4> 영화정보 DTD 파일 목록

DTD File 명	내 용
films.dtd	영화정보 DTD
hotnews.dtd	핫뉴스
openfilm.dtd	이번주 개봉작, 현재상영작
boxoffice.dtd	국내 박스오피스, 미국 박스오피스
link.dtd	연결(하이퍼링크) 기능

(그림 4)는 모듈화된 films.dtd를 보여주고 있다. films.dtd는 위에서 보는 바와 같이 hot-news.dtd, openfilms.dtd, boxoffice.dtd를 외부 엔티타로 가지고 있다.

```
<!ENTITY %hotnews SYSTEM "hotnews.dtd">
<!ENTITY %openfilm SYSTEM "openfilm.dtd">
<!ENTITY %boxoffice SYSTEM "boxoffice.dtd">
<!ELEMENT films (hotnews?,openfilm*,boxoffice*)>
```

(그림 4) films.dtd

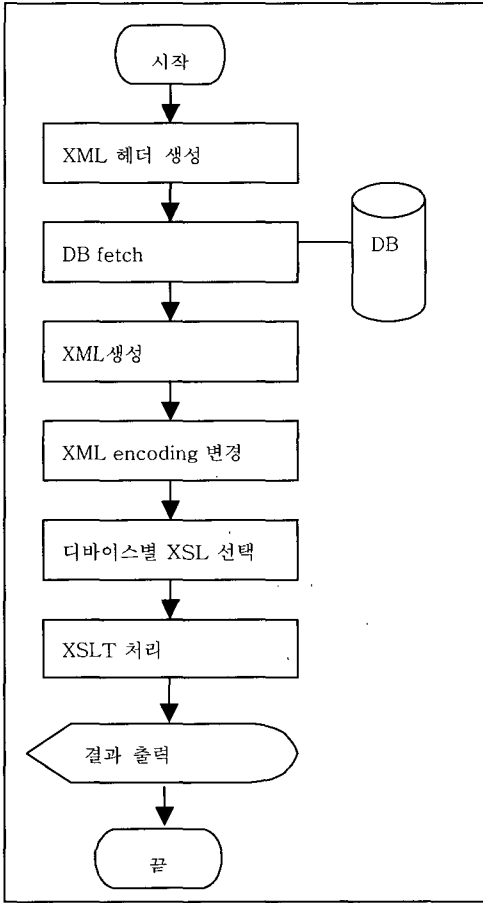
아래 그림은 openfilm.dtd를 보여준다.

```
<!ENTITY % link SYSTEM "link.dtd">
%link;
<!ELEMENT openfilm firstfilm,(filmlist)* >
<!ATTLIST openfilm category (#PCDATA)>
<!ELEMENT firstfilm (#PCDATA)>
<!ATTLIST firstfilm %link;>
<!ELEMENT filmlist (#PCDATA)>
<!ATTLIST filmlist %link;>
```

(그림 5) openfilm.dtd

'이번주 개봉작'과 '현재 상영작', '국내 박스오피스'와 '미국 박스오피스' 정보는 각각 같은 DTD를 공유한다고 하였다. 이 경우 XML에 표현된 정보가 '이번주 개봉작'인지 '현재 상영작'인지를 구분하는 속성이 필요하다. 여기서는 open-film.dtd와 boxoffice.dtd에 'category'라는 속성을 두어 구분자 역할을 하게 하였다.

4.3 영화정보 웹 콘텐츠 생성 절차



(그림 6) 웹 콘텐츠 생성절차

영화정보 웹 콘텐츠 생성은 영화 정보를 DB에서 추출하여 XML로 생성하고 디바이스에 따른 XSL를 선택하여 XSLT Process를 거쳐 각각에 해당하는 HTML을 생성하여 출력하는 것을 기본적인 절차로 한다.

(그림 6)은 웹 콘텐츠 생성절차를 순서도의 형식으로 도식화하여 보인 것이다.

맨 먼저 ‘<?xml version="1.0"...?’으로 시작하는 XML 헤더를 생성한다. 영화정보 DB에 연결

하여 해당 정보를 추출(fetch)하고 films.dtd에 정의한 각 요소의 규칙을 토대로 films 정보의 xml 태그를 생성한다. 생성된 xml은 한글정보를 포함하기 때문에 XSLT를 수행할 때 문제가 없도록 encoding을 변경하고 PC를 위한 xsl을 선택할 것인지, PDA를 위한 xsl을 선택할 것인지를 결정한 다음, xslt 프로세스를 생성하여 XSLT를 수행하도록 하면 결과값이 xsl에서 지정한 대로 html이 생성되고, 생성된 html을 화면에 출력하도록 한다.

V. 시스템 구현

5.1 영화정보 데이터의 XML 생성

(그림 7)은 영화정보 DB에서 연결해서 데이터를 추출하고 XML 태그를 붙여 XML파일을 생성하는 PHP 프로그램의 일부이다. 굵은 글씨로 표현된 것은 xml 정보를 나타낸다. 먼저, XML을 저장할 변수를 하나 선언해서 ‘<?xml version="1.0"?’의 XML 헤더를 저장한 다음, films의 최상위 요소(root element)인 ‘<films>’의 네임 스페이스를 정의하고 영화정보 DB에 연결한다.

‘핫뉴스’ 정보를 위한 ‘<hotnews>’ 태그를 생성하고, 최신 4개의 목록을 select하여 ‘<newslist>’를 4개 만든다. XML에서는 반드시 끝태그가 있어야 하므로 ‘</newslist>’와 ‘</hotnews>’도 반드시 추가하도록 한다.

```

<?php
// XML Header 만들기
$xml = "";
$xml .= "<?xml version='1.0'
        encoding='UTF-8'?'>\n";
$xml .= "<films xmlns:films='films.dtd'>\n";
// 영화 DB 연결
include "/dbClass.php";
$db = new MySQLdb;
$db->connect( );
// 핫뉴스 가져오기
$sql = "select id, title from hotnews order by
id desc limit 4";
$result = $db->query($sql);
$num_rows = MySQL_num_rows($result);
$num = 0;
if ($num_rows) {
    // <hotnews> XML 시작 태그 만들기
    $xml .= "<hotnews xmlns:hotnews=
            'films.dtd'>\n";
    while ($row = MySQL_fetch_array($result)) {
        $num++;
        $id = $row["id"];
        $title = htmlspecialchars($row["title"]);
        // <newslist> 추가하기
        $xml .= "<newslist href=
                '/Board/view.html?id=".$id."'>
                $title.</newslist>\n";
    }
    // </hotnews> 끝태그 만들기
    $xml .= "</hotnews>\n";
}
(중략)
$xml .= "</films>\n";
$db->close();
// 한글 데이터를 'UTF-8'로 인코딩 변환하기
$xml_src = iconv("EUC-KR", "UTF-8", $xml);
$fp = fopen("films.xml", "w");
fwrite($fp, $xml_src);
fclose($fp);
?>

```

(그림 7) 영화 정보 XML 생성 (films.php)

‘핫뉴스’와 마찬가지로 방법으로 ‘이번주 개봉작’, ‘현재 상영작’, ‘국내 박스오피스’, ‘미국 박스오피스’의 xml 태그를 생성하고 끝은 ‘</films>’로 마무리한다.

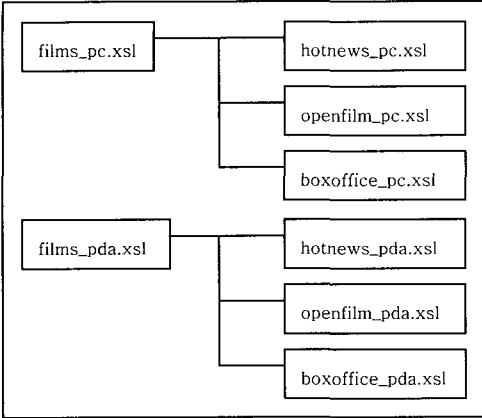
이렇게 하여 \$xml 변수에 films 의 xml 정보가 다 담기면 \$xml의 encoding을 ‘UTF-8’로 변경한 후 films.xml 파일을 생성한다.

5.2 XSL의 모듈화

XSL(extendible Stylesheet Language)의 기능은 단순히 표현양식으로서의 변형에 그치지 않는다. XSL은 프로그래밍 언어로서의 기능을 가지고 있다. XSLT의 엘리먼트는 XSLT 프로세스에서 특정 기능을 수행하도록 지시하는 명령어의 기능을 수행한다. 예를 들면, <xsl:import>, <xsl:include>는 XSL의 모듈화를 지원하고, <xsl:variable>, <xsl:param> 등은 변수의 사용을 가능하게 하고, <xsl:choose>, <xsl:if>, <xsl:for-each> 등은 변환의 흐름 제어를 가능하게 한다.

본 논문의 예제에서는 XSL의 모듈화를 위해서 <xsl:import href="..."> 엘리먼트를 이용했다. 영화 정보의 모든 내용을 하나의 xsl에서 모두 표현하는 것이 아니라, 정보의 종류에 따라 각각의 xsl을 만들고 상위의 xsl에서는 import만 하면 된다.

(그림 8)은 films 정보의 xsl 파일 계층도를 보여준다. PC 브라우저를 위한 films_pc.xsl과 PDA 브라우저를 위한 films_pda.xsl로 크게 구분되고, 정보의 내용별로 모듈화된 각각의 xsl 파일들도 각각 _pc.xsl과 _pda.xsl로 파일명을 구분하였다.



(그림 8) 영화정보 XSL 계층도

```

<?xml version="1.0"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
>
<xsl:import href="/hotnews_pc.xml"/>
<xsl:import href="/openfilm_pc.xml"/>
<xsl:import href="/boxoffice_pc.xml"/>
<xsl:output method="html"
indent="yes" encoding="EUC-KR"/>

<xsl:template match="films">
<html>
(중략)
<!-- Hot News -->
<xsl:apply-templates select="hotnews"/>
(중략)
</html>
</xsl:template>
<xsl:template match="hotnews">
<xsl:apply-imports/>

```

(그림 9) <xsl:import>를 사용한 xsl의 모듈화 (films_pc.xml)

(그림 9)는 <xsl:import>, <xsl:apply-imports>를 이용하여 xsl의 모듈화를 보여주는 films_pc.xml의 일부이다. hotnews_pc.xml, openfilm_pc.xml, boxoffice_pc.xml을 각각 import하고 apply-templates를 전개할 때는 apply-imports를 사용해서 import된 xsl 파일이 실행될 수 있게 하였다.

그리고, 다른 정보이지만 구조가 같다는 이유로 DTD와 XSL을 함께 사용하는 '이번주 개봉영화'와 '현재 상영영화', '국내 박스오피스'와 '미국 박스오피스'의 제목 이미지 출력을 위해 <xsl:if> 엘리먼트를 사용했다.

```

<xsl:template match="boxoffice">
<table width="169" cellpadding="0"
cellspacing="0" border="0">
<tr>
<xsl:if test="(@category = 'koreabox')">
<td></td>
</xsl:if>
<xsl:if test="(@category = 'usabox')">
<td></td>
</xsl:if>
</tr>
(중략)
</xsl:template>

```

(그림 10) <xsl:if>의 사용(boxoffice_pc.xml)

(그림 10)은 <xsl:if> 엘리먼트를 사용하여 국내 박스오피스 (title_r_boxoffice.gif)와 미국 박스오피스 (title_a_boxoffice.gif)를 구분하여 출력하는 boxoffice_pc.xml의 일부이다.

5.3 php, sablotron XSLT Processing

php에서는 sablotron XSLT 엔진을 쉽게 사

용할 수 있는 wrapper 함수들을 제공하고 있다. xslt_create()는 xslt_process()를 실행할 수 있는 XSLT 프로세스를 생성한다.

xslt_process()는 xslt_create()에서 생성된 프로세스와 xml 파일, xsl 파일을 인자로 하여 XSLT를 수행하고 그 결과를 리턴한다.

xslt_errno()와 xslt_error()은 각각 에러 번호와 에러 메시지를 출력하는 함수이다.

xslt_free()는 할당받은 xslt 프로세스 자원을 해제한다.

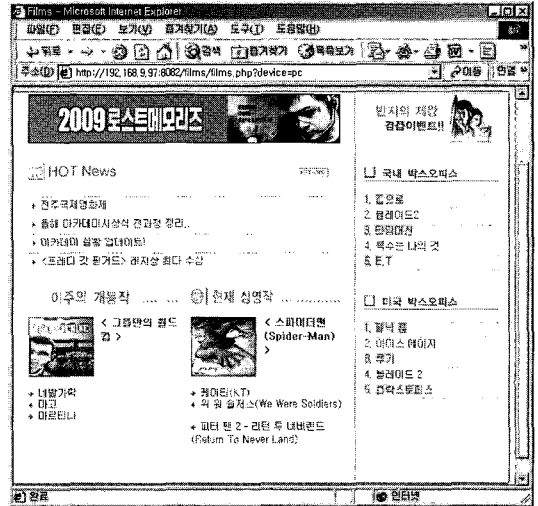
```
<?php
// XSLT Process 생성하기
$xmlproc = xslt_create();
// 단말기에 따른 xslt_process 작업하기
if ($device == "pda")
    $xmlresult = xslt_process($xmlproc,
        "films.xml", "films_pda.xsl");
else
    $xmlresult = xslt_process($xmlproc,
        "films.xml", "films_pc.xsl");
if (!$xmlresult) {
    echo "error is ". xslt_error($xmlproc);
    echo " error code is ". xslt_errno($xmlproc);
}
// XSLT Process 자원 할당 해제
xslt_free($xmlproc);

// 결과화면 보여주기
echo "<pre>\n";
echo $xmlresult;
echo "</pre>\n";
?>
```

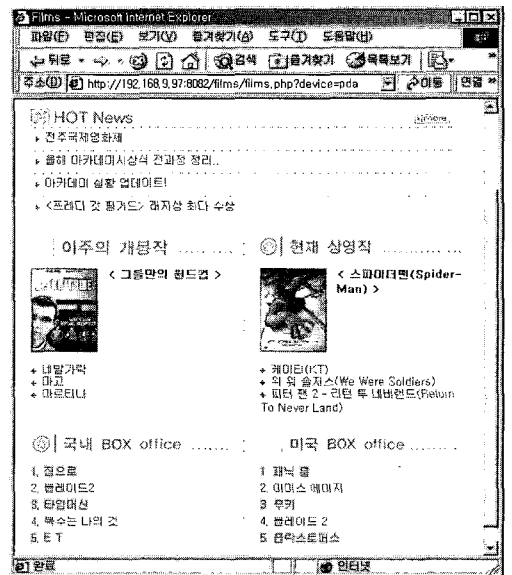
(그림 11) XSLT 수행 (films.php)

(그림 11)의 php 프로그램은 XSLT 프로세스를 생성하고 단말기에 따라 xsl 파일을 선택하여 XSLT를 수행하고 결과를 출력하는 films.php 프로그램의 일부분이다.

5.4 PC와 PDA 브라우저를 위한 웹 콘텐츠 실행 화면



(그림 12) PC 브라우저를 위한 영화정보 실행화면



(그림 13) PDA 브라우저를 위한 영화정보 실행화면

(그림 12)에서 (그림 1)과 같은 화면 레이아웃을 가진 PC 브라우저를 위한 웹 콘텐츠 화면을 볼 수 있다. (그림 13)는 (그림 2)의 PDA 브라

우저를 위한 화면 레이아웃과 같은 영화정보 웹 콘텐츠를 보여주고 있다.

(그림 12)와 (그림 13)에서 보는 바와 같이 하나의 xml 파일을 두 개의 각기 다른 단말기를 위해 디자인된 xsl 파일에 연결하여, 크기가 서로 다른 PC와 PDA 브라우저를 위해 영화정보의 화면 구성이 다른 것을 확인할 수 있다.

Ⅴ. 결 론

무선 인터넷을 위한 콘텐츠의 수요와 질을 충족시키기 위해 유선 인터넷의 콘텐츠를 무선 인터넷에 제공하기 위한 연구가 이어지고 있고, 솔루션들이 속속 개발되고 있다.

본 논문에서는 이와 관련된 콘텐츠 변환 기술, 무선 인터넷 통합 마크업 언어인 XHTML basic, XML 기반의 콘텐츠 제작에 대해 알아보고, 데이터의 구조와 표현을 분리하는 XML의 특성을 이용하여 XSLT를 사용하여 하나의 XML 데이터에 단말기에 따라 다른 형식의 디스플레이를 위한 복수 XSL의 적용을 영화정보 콘텐츠를 구현함으로써 보여주었다.

본 연구와 기존 연구들의 차이점이라면, 기존 연구들은 HTTP와 WAP간의 콘텐츠 변환이나, m-HTML, s-HTML, c-HTML 등 무선 인터넷용 마크업 언어의 통합 콘텐츠 제작에 관한 연구에 치중하여 있었고, 본 연구에서는 같은 HTTP를 사용하는 PC와 PDA 간의 콘텐츠 공유를 위한 방법을 제안하였다.

본 논문에 제안하는 방법의 장점은 첫째, XML 파서와 XSLT 엔진을 Open Source로 사용하기 때문에 다른 솔루션을 도입할 때 필요한 금전적 비용이 들지 않는다. 둘째, XML을 사용하더라도 데이터베이스를 변경하거나 데이터의 구조를 변경하지 않을 뿐 아니라, 데이터베이스에서 데이터를

추출하는 프로그램을 재사용할 수 있다. 셋째, XSLT를 사용함에 있어 php extension에서 지원하는 wrapper 함수를 사용하기 때문에 몇몇 함수의 사용법만 익히면 쉽게 사용할 수 있어, 기존 웹 프로그래머의 XML 기반 기술로의 진입을 용이하게 한다.

본 논문의 구현에서 사용한 expat과 sablotron 라이브러리는 한글 사용의 불편과 Xlink의 사용이 제한되어 있지만, 유선 인터넷 콘텐츠를 무리 없이 무선 인터넷에서 공유할 수 있는 방법을 제공하고 있다.

굳이 무선 인터넷을 위한 콘텐츠 공유가 목적이 아니라 할지라도 XML 기반의 콘텐츠 제작은 잦은 디자인 변경이 있는 정보를 XSL의 변경을 통해 개발 기간을 단축할 수 있다. 또한 웹 디자이너와 프로그래머의 작업 영역이 뚜렷이 구분되기 때문에 서로간의 역할 분담도 확실해지고, 콘텐츠 제작 초기의 DTD 공유를 제외하고는 작업의 독립성도 보장되어 효율성을 높일 수 있다.

뿐만 아니라, 정보 교환의 표준인 XML을 기반으로 콘텐츠를 제작하는 것은 단순히 보기 위한 콘텐츠만이 아니라, 정보 교류를 위한 콘텐츠로서의 기회가 부여되는 것이다. 이런 의미에서도 콘텐츠를 XML 기반으로 변경하기 위한 효율적인 방법들에 관한 지속적인 연구가 필요하다 하겠다.

참 고 문 헌

- [1] Mosca, "무선인터넷 가이드", <http://www.i-biznet.com/mobg/default.asp>, 2001.5
- [2] 권오성, "WAP Converter" <http://www.mosca.co.kr>, 2000.11
- [3] H. Ouahid, A. Karmouch "Converting Web Pages into Well-formed XML Documents",

- Proceedings of IEEE International Conference on Communication, pp. 676-680, 1999
- [4] 안원섭, 나연목, "HTML 필터링을 통한 실시간 WML 콘텐츠 변환기의 설계 및 구현", 한국정보과학회 학술발표논문집 Vol.27 No.2 pp.263-265, 2000년 가을
- [5] 박기현, 강동우, 권정선, "HTML 필터링 기능을 갖춘 WAP 게이트웨이 시스템 구축", 정보과학회 논문지:컴퓨팅의 실제 제 7권 제 4호, pp.350-358, 2001.8
- [6] 박경은, "XML의 구조 정보를 이용한 유.무선 단말기용 웹 문서 생성기", 석사학위논문, 이화여자대학교 과학기술대학원, 2002
- [7] 정병수, 이민석, 김강희, 안광선, "통합 무선인터넷 서비스를 위한 시스템의 설계 및 구현", 한국정보과학회 학술발표논문집 Vol.28 No.1 pp.484-486, 2001년 봄
- [8] 권오성, "Xhtml based Integration between Wired Internet and Wireless Internet", 모스카 Weekly Newsletter : 時空초월 제29호 (2001년 5월 2일)
- [9] Richard Vidgen, Steve Goodwin, "Special Feature E-Commerce XML: what is it good for?", computing & control engineering journal, pp.119-124, June 2000
- [10] 김채미, 최학열, 김심석, "전문가와 함께 가는 XML Camp", 마이트Press, 2001
- [11] 이종호, "XML과 전자상거래", 정보문화사, pp.107, 2001
- [12] W3C, "Shaping the Future of HTML", W3C work-shop 4th/5th May 1998, <http://www.w3.org/MarkUp/future/>
- [13] 김민식, "XHTML을 중심으로 한 무선 서비스 표준", KISDI IT FOCUS, 정보통신정책연구원, pp.68-71, 2001년 4월호
- [14] WAP Forum, "WAP 2.0 Technical White Paper", January 2002, http://www.wapforum.org/what/WAPWhite_Paper1.pdf
- [15] W3C, "The Extensible Stylesheet Language (XSL)", <http://www.w3.org/Style/XSL/>
- [16] W3C, James Clark, "XSL Transformations (XSLT)", <http://www.w3.org/TR/xslt>
- [17] W3C, James Clark, Steve DeRose, "XML Path Language (XPath)", <http://www.w3.org/TR/xpath>
- [18] 전윤경, "xsl-fo document 개념을 중심으로 한 따라하기 예제", 2001년 11월, [http://www.devworld.co.kr/doc/xml/xmlloan/content/xsl-fo document.html](http://www.devworld.co.kr/doc/xml/xmlloan/content/xsl-fo%20document.html)
- [19] 송민영, 이기호, "무선 단말기용 XML기반 맞춤형 서비스 시스템 설계", 한국정보과학회 학술발표논문집 Vol.28 No.2, pp.142-144, 2001년 가을
- [20] James Clark, "expat - XML Parser Toolkit", <http://www.jclark.com/xml/expat.html>
- [21] The PHP Group, "PHP:Manual:XML parse functions", <http://kr2.php.net/manual/kr/ref.xml.php>
- [22] Ginger Alliance Open Resource Center, "sablotron", http://www.gingerall.com/charlie/ga/xml/p_sab.xml
- [23] GNU, "libiconv", <http://www.gnu.org/software/libiconv/>

김 경 아



1994년 이화여자대학교 전
자계산학과 (이학사)

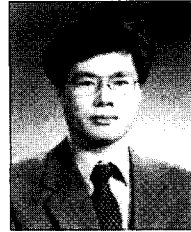
2002년 이화여자대학교 정
보과학대학원 정보과학과
(정보과학 석사)

1994년 - 현재 한국통신하

이텔 기술본부 응용개발팀

관심분야 : XML, 웹 콘텐츠 개발, 유무선 콘텐츠
통합 등

용 환 승



1983년 서울대학교 컴퓨터
공학과 졸업 (학사)

1985년 서울대 대학원 컴퓨
터공학과 졸업 (공학석사)

1994년 서울대 대학원 컴퓨
터공학과 졸업 (공학박사)

1995년 - 현재 이화여자대학교 컴퓨터학과 부교수,
2000년 8월 - 2002년 7월 정보과학대학원 교학
부장

관심분야 : 데이터베이스, XML, 데이터마이닝 등