
XML을 이용한 웹 메일 통합 시스템의 설계 및 구현

Design and Implementation of Web Mail Integration System Using XML

권정식*, 조대제**

안동대학교 컴퓨터공학과*, 안동대학교 멀티미디어공학과**

Jung-Sik Kwon(jskwon@andong.ac.kr)*, Dae-Jea Cho(djcho@andong.ac.kr)**

요약

대부분의 웹 메일 서비스는 SMTP와 POP3을 이용하여 구현된다. 이런 웹 메일 서비스는 기본적인 송수신 기능 뿐 만 아니라, 여러 가지 부가적인 기능을 사용자들에게 제공한다. 하지만 메일 송수신 데이터는 그것을 관리하는 서버에 저장되기 때문에, 이 서버에 대한 접속 권한이 없는 일반 웹 메일 사용자는 자신의 메일 데이터를 가공할 수가 없다.

본 논문에서는 XML을 이용한 웹 메일 통합 시스템을 구현하였다. 제안된 시스템에서는 기존의 웹 메일 시스템에 저장된 메일 데이터와 사용자 환경을 XML문서로 변환하고 데이터베이스에 삽입한다. XML 데이터를 데이터베이스에 저장하기 위해 XML을 이용한 동적인 문서처리 방법을 사용하고 변환된 XML 파일을 레코드 단위로 분석한다. 제안된 시스템은 사용자가 자신의 메일 계정을 변경할 경우에, 구축된 데이터베이스를 이용하여 자신의 메일 데이터와 환경 설정을 쉽게 할 수 있도록 한다. 변환된 XML 문서는 무선 인터넷 마크업 언어인 WML(Wireless Markup Language)로도 쉽게 변환 가능하여 모바일(Mobile)서비스를 지원하는 경우, 재구축으로 인한 시간과 경비의 낭비와 개발자의 수고를 덜 수 있다.

■ 중심어 : | XML | 메일시스템 | 마크업언어 | 스팸메일 |

Abstract

Web-Mail Services are implemented with SMTP and POP3 in these days. These web mail services provide not only the basic sending and receiving functions but also additional functions for user. However, the sent and received data are stored in the web mail server. So general web mail users can't process their mail data if they have not access ID for the web mail server.

In this paper, we design and implement WMIS(Web Mail Integration System) using XML. In the proposed system, mail data and user's environments in the old web mail system are transformed into XML documents and inserted into database. In order to store XML data into database, we use dynamic document processing method and analyze XML file by record unit. When the users change their mail accounts, they easily can establish the web environments and their mail data. Because the transformed XML document is converted into WML easily, the WMIS can reduce development cost, labors and time.

■ keyword : | XML | Mail system | Markup language | Spam mail |

* 이 논문은 2004학년도 안동대학교 학술연구 조성비에 의하여 연구되었음.

접수번호 : #040909-002

접수일자 : 2004년 9월 9일

심사완료일 : 2004년 10월 28일

교신저자 : 조대제, e-mail : djcho@andong.ac.kr

I. 서론

전자우편은 인터넷시대의 중요한 통신 수단으로 자리 잡았다. 전자우편을 이용하는 방법으로는 클라이언트/서버 시스템과 웹 메일 시스템을 이용하는 방법이 있는데, 클라이언트/서버 시스템은 개인의 로컬 시스템에 메일 전용 어플리케이션을 설치하고 메일 계정을 가진 서버에 접속하여 메일 메시지를 받아 볼 수 있다. 그리고 웹 메일 시스템은 웹 메일 서비스를 제공하는 업체에 회원 가입을 통해 사용자 계정을 만들어 사용하는 방법으로 온라인 상태에서 어디서든지 접속이 가능하며, 웹 브라우저만 있으면 별다른 소프트웨어를 설치하지 않아도 전자우편을 송수신을 할 수 있어 쉽게 이용할 수 있다.

하지만 기존의 전자우편 송수신 시스템은 여러 가지 문제점을 가지고 있다. 클라이언트/서버 시스템은 특정 파일에 메일 메시지가 저장되고 웹 메일 시스템은 데이터베이스에 고정적인 형식으로 메일 메시지를 저장하기 때문에 다른 어플리케이션과 호환하여 사용할 수가 없다. 메일 정보를 이용하기 위해서는 매 번 데이터베이스에 접근하여 해당 메일 데이터를 가지고 온다. 그리고 고정적인 레이아웃만을 제공하기 때문에 사용자가 원하는 레이아웃을 제공하지 못하고 있다[1,2]. 또한 사용자 컴퓨터가 변경되면 이전 컴퓨터에 저장된 메일을 가져오기 위해서 메일 박스를 복사해야하는 번거로움이 있다. 웹 메일 시스템은 사용자가 메일 계정을 변경하였을 때 기존 사용자의 주소록, 받은 메일, 지운 메일, 스팸 메일 차단 정보 등을 가져오지 못하기 때문에 사용자정보를 재구축해야 하는 불편한 점이 있다. 또한 서버가 정지되어 사용자가 메일 서비스를 받지 못하거나 데이터의 예러가 발생하여 메일을 유실하였을 경우도 문제점으로 지적되고 있다. 따라서 이와 같이 기존의 메일 시스템들은 메일 데이터의 확장성, 사용자 환경정보, 사용자 레이아웃, 데이터의 안전성 등에 문제점이 있다.

본 논문에서는 XML을 이용한 웹 메일 통합 시스템을 구현하였다. 제안한 시스템에서는 기존의 MIME타입의 메시지와 웹 메일 시스템의 메일 데이터와 사용자 환경을 XML문서로 변환하고 데이터베이스에 삽입하였다. 사용자는 메일 계정을 변경 할 경우에 자신의 메

일 데이터와 환경설정을 재구축 할 수 있도록 하는 웹 메일 시스템의 통합 방법을 제시하였다.

제안한 시스템은 기존의 웹 메일 시스템의 메일 송수신 기능과 메일 관리 기능에 바탕을 두고 SMTP/MIME과 XML의 문서구조를 비교 분석하고 DOM(Document Object Model)을 이용하여 XML로 변환하였다. XML 데이터를 데이터베이스에 저장하기 위해 XML을 이용한 동적인 문서처리 기술과 변환된 XML 파일을 레코드 단위로 분석하였다. 또한 데이터베이스를 XSLT(eXensible Stylesheet Language Transformation)를 이용해 XML로 변환함으로써 데이터의 재 사용성을 높이고 이동성을 용이하게 하였다. 기존의 메일 데이터를 XML로 변환함에 따라 무선 인터넷 마크업 언어인 WML(Wireless Markup Language)로도 쉽게 변환 가능하여 모바일(Mobile) 서비스를 지원하는 경우 재구축으로 인한 시간과 경비의 낭비와 개발자의 수고를 덜 수 있다[3].

II. 관련연구

1. 전자메일 프로토콜

현재 전자 메일에서 사용되는 주요 프로토콜 및 기술에는 SMTP, POP, IMAP, MIME등이 있다. SMTP(Simple Mail Transfer Protocol)는 서버간에 메일을 전송하는데 사용하는 프로토콜로서, 사용자가 보낸 메일을 서버끼리 주고받는데 사용한다.

POP(Post Office Protocol)은 메일 서버에 저장된 메일을 사용자의 PC로 전송하는데 사용하는 프로토콜이다. 현재 많이 사용되는 메일 송·수신 방법인 웹 메일은 일반적인 메일 시스템과 달리 특별한 사용자 인터페이스나 메일 서버에서 메일을 전송 받아야 하는 프로토콜 없이 웹브라우저에서 메일을 읽을 수 있도록 만든 시스템이다.

IMAP(Internet Message Access Protocol)는 POP3보다도 유연하고 뛰어나다고 할 수 있다. 서버 측에 메일 박스를 둘 수 있다는 것과 메일의 헤드만을 읽을 수 있는 것이 특징이다. 클라이언트는 서버의 메일

박스에서 메일을 삭제하지 않고 필요한 메일만 복사할 수도 있고 헤드만 읽고 판단하여 크기가 큰 메일이나 우선도가 낮은 메일을 나중에 읽거나 삭제할 수 있다. 이러한 특징 때문에 이동 기기 액세스에 적합한 프로토콜이라고 할 수 있다.

MIME은 인터넷상에서의 메시지전송 표준 규약인 SMTP의 텍스트 메시지를 전송하기 위해 고안된 프로토콜이다. MIME은 어떻게 바이너리 파일을 텍스트로 표현하여 그것이 텍스트 기반의 전자메일을 통해 보내질 수 있는가를 기술하고 있는 일련의 명세들로서 SMTP와 더불어 인터넷 메일 시스템의 기초를 이루고 있는 핵심이다. 아스키 데이터만을 처리할 수 있는 원래의 인터넷 전자우편 프로토콜, 즉 SMTP를 확장하여 오디오, 비디오, 이미지, 응용프로그램, 기타 여러 종류의 데이터 파일들을 주고받을 수 있도록 기능이 확장된 프로토콜이다. 이것은 메시지 유형의 결정이나, 미래를 위한 성장 지원 등을 포함하고 있으며, 비 구조화된 데이터이지만 완전한 텍스트로 되어있으므로 쉽게 XML 데이터로 변환 가능할 수 있다[2].

2. XML을 이용한 웹 메일 통합 시스템의 필요성

SMTP/MIME타입으로 된 메일 정보는 재사용성과 확장성이 부족하다. 또한 무선 인터넷에서의 접근이 어렵고, 새로운 기술과 개념이 많이 등장하는 시대에 맞추어 다양한 접근경로를 지원하지 못한다는 것을 알 수 있다.

본 논문에서는 웹 메일 시스템의 장점인 쉬운 인터페이스, 온라인상태에서의 간단한 접근 방법과 클라이언트/서버 메일 시스템의 장점인 오프라인에서도 검색 가능하다는 것과 XML기술의 장점인 확장성, 데이터 표현의 다양화 등의 장점들을 가지고 통합 시스템을 구현하였다. 기존의 메일 시스템의 단점이었던 고정적인 사용자 인터페이스, 메일 계정 변경할 경우 개인의 메일과 사용자 환경 설정 정보의 재구축이 불가능하였던 것들을 XML을 이용하여 단점들을 보완하였다.

III. WMIS(WebMail Integration System)의 설계

본 논문에서는 웹 메일 서버들의 여러 가지 기능에 사용자의 주소록, 받은 메일, 지운 메일 등을 XML파일로 변환, XML파일을 다시 데이터베이스에 삽입하여 서로 다른 기반의 웹 메일 서비스를 이용할 때도 저장해놓은 XML파일을 이용해 기존의 환경을 재구축하고 오프라인 상에서도 검색 할 수 있는 기능을 구현하였다.

1. 시스템의 구성 및 처리 절차

본 시스템은 메일 송수신기, ToXML 변환기, XML 파일 생성기, XSL 생성기, XMLtoDB 변환기, DBtoXML 변환기 등 6개의 모듈로 구성할 수 있다. 그림 1은 본 시스템을 구성한 것으로써 메일 송수신 모듈은 실제로 메일을 보내고 받는 기능을 하며, ToXML 변환기는 MIME으로 구성되어진 메일을 XML로 변환하는 기능을 행한다. 그리고 XML생성기와 XSL생성기는 XML문서와 XSL을 자동으로 생성하며, XMLtoDB 변환기는 변환된 XML문서를 데이터베이스에 삽입시키는 역할을 한다. DBtoXML 변환기는 데이터베이스에 저장된 데이터를 이용해 XML문서로 다시 만드는 모듈이다. 각 각의 모듈들은 데이터베이스에 저장되어 있는 메일, 주소록, 기타 환경설정 정보들을 XML 파일로 변환하고 다시 데이터베이스에 갱신 및 등록을 하는 역할 등을 가지고 있으며, 사용자 개성에 맞는 XSL을 통해 변환된 XML 데이터를 사용자가 원하는 대로 출력 할 수 있다[4]. 이렇게 데이터를 분리함으로써 XML 파일 자체가 중요한 정보가 되게 되어 사용자는 웹 메일 서비스 벤더들의 데이터 분실, 서비스 에러에 대해서 대비 할 수 있다.

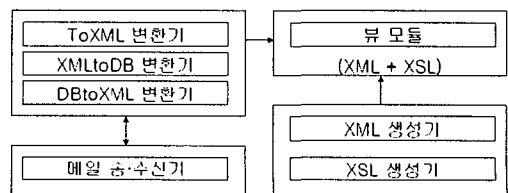


그림 1. 시스템 구성도

본 논문에서 제시한 시스템인 WMIS의 처리 절차는 그림 2와 같다. 사용자가 웹 메일 서비스를 이용하여 메일을 송수신하고 받은 메일을 읽게 되면 데이터베이스에 저장되고 이 메일은 사용자의 ID와 동일한 폴더 안에 XML파일로 변환되어 저장 된다. 또한 데이터베이스에 저장되어진 메일은 DBtoXML 변환기를 통해 XML파일로 저장되며, 사용자 개인 PC에 다운로드 할 수 있다. 사용자가 저장해 놓은 XML파일은 XMLtoDB 변환기의 파일 삽입과정과 파일 해석 과정을 통해 데이터베이스에 삽입 가능하다.

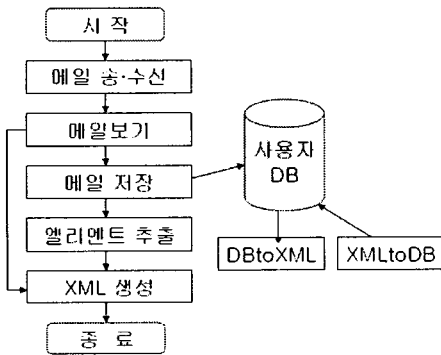


그림 2. 시스템 처리절차

2. ToXML 변환기

받은 메일들은 SMTP/MIME 문서로 Windows 2000서버의 메일 수신 폴더(inetpub/mail_root/drop)에 *.eml이란 확장자를 가진 파일로 저장된다. 이 문서는 사용자의 요구에 따라서 메일 읽기 과정에서 XML 파일로 변환할 수 있다. 그리고 이 파일은 저장할 수 있도록 창이 표시가 되고 사용자는 저장할 수 있다. eml 문서의 메시지 형식 중 헤더 부분은 식별자와 값으로 구별이 된다. 식별자를 XML문서의 엘리먼트로 1:1 대응시켜서 변환하고 SMTP메시지의 콘텐츠를 엘리먼트의 값으로 대응시켜 엘리먼트 사이에 삽입한다. 그리고 XML이 잘 구성된 문서(well formed)가 되도록 XML 문서의 정보와 루트 엘리먼트를 삽입시킨다. 여기서 DBtoXML 변환기와 구분되는 것은 ToXML 변환기에서는 SMTP/MIME문서를 변환을 하기 때문에 변환되

어진 문서에는 변환을 행하고 DBtoXML 변환기는 데이터베이스에 저장된 내용을 XML파일로 변환한다. 그리고 SMTP/MIME 메시지의 콘텐츠 부분의 마크업에 쓰이는 문자, "<", ">" 등은 XML 파서가 처리 할 수 있도록 ASCII코드에 지정되어진 문자 "<" , ">" 로 모두 치환시킨다. 그리고 본문 부분의 내용 또한 모두 XML파서가 처리할 수 있도록 치환시킨다.

본 논문에서는 비 정형화 된 데이터를 XML문서로 변환하는 방법들 중에서 전부를 변환하지 않고 필요한 부분들만을 XML로 변환하는 방식을 이용하였다. 이렇게 함으로써, 사용자가 원하는 데이터만을 변환하기 때문에 변환된 데이터의 가독성이나 변환속도 등이 향상된다. 표 1과 그림 3은 XML로 변환된 엘리먼트 구조와 정의를 나타내고 있다.

표 1. XML로 변환된 엘리먼트 정의

엘리먼트	설명	엘리먼트	설명
<subject>	메일의 제목	<from_user>	메일을 보낸 계정
<s_date>	메일보낸 날짜	<content>	메일의 본문
<to_user>	메일받은 계정		

메일들은 모두 그림 3과 같은 구조로 변환이 되어 진다. 각각의 메일데이터들은 MAIL이라는 Root 엘리먼트와 그 하위에 <r_mail>이란 엘리먼트로 각각의 메일 데이터들을 분리할 하며, <r_mail> 엘리먼트 하위의 지식 엘리먼트에는 <subject>, <to_user>, <from_user>, <content>, <s_date>라는 엘리먼트가 존재한다. 표 2와 같이 각각의 하위 엘리먼트들은 SMTP/MIME의 데이터와 매핑되어 5가지의 정보들을 가져오게 된다.

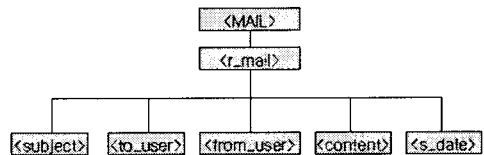


그림 3. XML로 변환된 엘리먼트 구조

표 2. MIME → XML변환 매칭표

SMTP/MIME	XML변환 후
From	from_user
To	to_user
Subject	subject
date	s_date
body	content

3. XMLtoDB 및 DBtoXML 변환기

사용자가 웹 메일 서비스를 이용하면서 타 메일 서비스를 이용하게 될 경우, 기존의 메일이나 사용자 환경정보를 다른 웹 메일 서버에 재구축을 할 때 사용한다. 기존에 만들어진 XML데이터를 사용자는 게시판에 자료를 등록하는 것과 같은 절차로 업로드하게 되며 새로운 서버는 XML파일을 해석하는 과정을 거쳐 데이터베이스에 등록, 갱신하게 된다. 그리고 XML문서가 유효하지 않은 경우 파서는 상황에 따라서 자바스크립트가 제공하는 표준적인 Error 오브젝트와 DOM이 제공하는 ParseError 오브젝트를 반환한다[5]. 사용자가 유효하지 않은 XML 문서를 업로드할 경우에 XMLDOMParseError 오브젝트를 반환하고 그 메시지를 사용자에게 보여준다.

DBtoXML 변환기는 데이터베이스에 등록되어 있는 메일이나 주소록, 기타 사용자 환경설정 등을 XML 파일로 변환하여 메일 서비스의 에러에 대비할 수 있다. 메일 계정을 변경 할 경우 사용자 환경 재구축을 위해서 필요하기 때문에 XML로 변환을 해야 한다. 사용자는 필요한 정보만 변환하면 되므로 변환된 파일의 용량이나 처리 속도 면에서 우수하다[6]. 데이터베이스에 있는 데이터들을 XML 데이터로 변환하는 방법으로는 태그 매칭을 이용하는 방법과 데이터베이스에서 XML을 지원하는 여러 가지 기능을 이용하는 방법이 있다[5,7].

본 시스템에서는 XSLT 스타일 시트를 이용한 방법으로 구현하였다. XSLT를 이용함으로써, 테이블 구조에 맞추어 자동적으로 변환 할 수 있어 테이블 구조에 상관없이 XML로 변환 가능하다.

4. XSL 생성기

생성되고 조합된 메일 문서를 보기 위해 사용자 정의

XSL문서를 만드는 것으로써, 사용자는 XSL 만들기 메뉴에서 폼에 제공되는 옵션들을 이용해 자기 고유의 XSL문서를 만들 수 있다. 이렇게 다양한 XSL문서를 이용하여 사용자의 XML문서에 적용함으로써 레이아웃, 출력 목록, 정렬, 색상, 폰트, 테이블 크기 등을 임의로 정할 수 있다. 이렇게 함으로써 사용자는 직접 생성한 스타일 시트 문서로 동일한 하나의 데이터를 다양한 형식으로 출력시킬 수 있다. 작성한 XSL을 기본 XSL로 지정 할 수 있어 사용자가 정의한 화면으로 출력 할 수 있다.

IV. WMIS의 구현

본 논문에서 제안한 시스템의 개발 환경은 마이크로소프트사의 Windows 2000 Advanced Server에서 제공하는 IIS5.0의 SMTP를 사용하였으며, ASP언어로 구현하였다. XML 데이터들은 XML V4.1.2의 스펙을 따르고 문서의 유효성 검사와 XML 문서처리를 위해 MS XML3.0을 파서로 사용하였다. 데이터베이스는 MS-SQL 2000을 사용하였다.

1. 메일 송수신기

메일 송수신기는 윈도우즈2000 서버 환경에서 EMWAC(European Microsoft Windows NT Academic Center)[8]이라는 무료 메일 서버를 이용하여 구현하였다. IIS 5.0에는 SMTP를 지원을 하고 있어 기본적인 송수신 기능을 그대로 사용을 하였으며, IIS 5.0에서 지원하지 않는 POP3기능만을 EMWAC를 통해 사용하였다. 그림 4에서는 EMWAC 메일 서버의 기본적인 기능을 보여 주고 있다.

2. ToXML 변환기

ASP3.0의 FSO(File System Object) 객체와 DOM API를 사용하여 구현하였다. 사용자가 새로운 메일을 XML파일로 변환하고 싶을 경우에 변환을 원하는 메일의 상세보기 페이지에서 변환 버튼을 클릭하면 변환과정이 시작되며 서버에서 자동으로 메일 데이터가 XML

데이터로 변환된다. 선택한 XML데이터는 서버에서 XML 파서와 DOM API에 의해 트리 구조로 메모리에 저장된다. DOM은 문서의 구조를 메모리에 저장시켜 놓고 검색하여 노드의 삽입이 가능하도록 한다. DOM API는 MAIL 루트 노드를 검색하고 r_mail이라는 자식 노드를 생성시키고 하위 엘리먼트를 생성시킨다.

이 하위 엘리먼트의 값들은 SMTP/MIME에서 필요한 프러퍼티만을 가져와 시작태그와 끝태그 사이에 삽입한다. FSO의 CreateTextFile, OpenTextFile, WriteLine 메소드를 통해 XML 파일을 생성, 입력할 수 있게 된다.

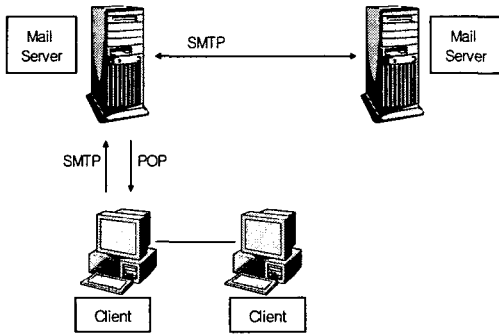


그림 4. EMWAC 의 기본 동작

3. XMLtoDB 변환기

XML로 변환한 데이터를 데이터베이스에 등록, 갱신하여야 할 때 필요한 모듈로써, 변환한 메일데이터나 사용자 환경설정정보 등을 새로운 메일 계정의 재구축에 필요하다. 그림 5는 XMLtoDB변환기의 과정을 보여주고 있다.

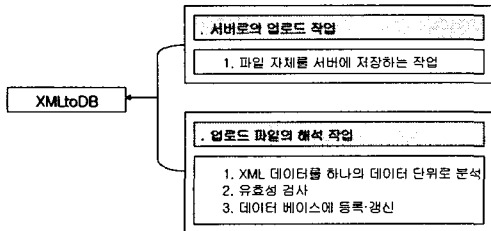


그림 5. XMLtoDB 과정

XMLtoDB 변환기는 서버로의 파일 업로드 작업과 업로드 파일의 해석 작업으로 크게 두 부분으로 나눌 수 있다. XMLtoDB 변환기에서의 과정은 아래 그림과 같이 간단히 나누어 볼 수 있다. 서버의 업로드 과정은 컴포넌트 Basp21이란 외부 컴포넌트를 사용하여 XML 문서를 업로드한다. 그리고 XML 데이터를 하나의 레코드 단위로 분석을 하게 된다. 루트 MAIL 엘리먼트 아래의 모든 요소들을 가져오면 데이터들은 유효성 검사를 한다.

이 과정은 XML 데이터가 잘 구성된 것(Well-Fromed) 인지를 검사를 한다. 다음은 XML 데이터들은 데이터베이스에 등록 또는 갱신을 한다. 업로드된 XML파일은 루트엘리먼트 <MAIL>아래의 모든 요소들을 가져오고 <r_mail>엘리먼트로 각각의 메일 데이터를 구분하여 하나의 레코드단위로 데이터베이스에 입력을 한다. 그림 6과 그림 7은 XMLtoDB 과정의 서버로 업로드 작업과 파일 해석 작업의 코드이다.

```

(<%
var objBsp=new ActiveXObject("basp21");
var objFs=new ActiveXObject("Scripting.FileSystemObject");
lngTB=Request.TotalBytes;
bin=Request.BinaryRead(lngTB);
' 읽어낸 데이터를 이진 데이터로 읽어 낸다. 이 내용을 기반으로
basp21 컴포넌트가 업로드 처리를 한다.
strOrgPath=objBsp.FormFileName(bin,"strFile");
strName=objFs.GetFileName(strOrgPath);
' 업로드 경로 strOrgOath에서 파일명만을 얻어 변수 strName에 저장한다.
strPath=objFs.BuildPath(Server.MapPath("/tmp"),strName);
' 파일을 저장할 곳은 현재 폴더 아래의 tmp 폴더로 정하고 이곳에
변수strName을 연결한다.
lngLen=objBsp.FormSaveAs(bin,"strFile",strPath);
If(lngLen=0){
' FormSaveAS메소드는 업로드에 성공한 경우에는 파일의 바이트수
' 실패한 경우에는 음수를 반환하기 때문에 성공을 하게 되면 아래
의 코드를 실행한다.
%>
(form action="process.asp? url=(%=Server.URLEncode(strName)%)
method="POST")
    다음 내용으로 처리를 시작해도 되겠습니까?
    (input type="submit" name="start" value="처리시작")
    (input type="submit" name="cancel" value="처리중지")
    .....
    .....
(<%
)elseif(
    Response.Write("업로드 실패");
%>
%>

```

그림6. XMLtoDB 파일 업로드 코드

```
(%
var objFs=new ActiveXObject("Scripting.FileSystemObject");
var objCon=new ActiveXObject("ADODB.Connection");
objCon.Open("xml");
strPath=Request.QueryString("url").Item();
if(Request.Form("Start").Item()!=null){
var objDoc=new ActiveXObject("Microsoft.XMLDOM");
objDoc.async=false;
objDoc.load(Server.MapPath("./tmp/" + strPath));
' 업로드한 xml 파일을 로드 한다.
...
for(i=0;i<nodMAIL.length-1;i++){
' 루트엘리먼트(<MAIL>) 아래의 모든 엘리먼트를 가져 온다.
var nodInfos=nodMAIL.item(i).childNodes;
' <r_mail> 요소 아래의 자식 요소들을 가져온다. 여기서 가져온
자식요소가 필드 값이 된다.
for(j=0;j<nodInfos.length-1;j++){
' 각각의 노드를 비교하여 지정한 엘리먼트의 값을 가지고 온다.
switch(nodInfos.item(j).nodeName){
case "subject" :
strSubject=nodInfos.item(j).text;
break;
case "to_user"
...
var objRs=objCon.Execute("SELECT * FROM mail WHERE user_id=" +
if(objRs.EOF){ '만약 사용자의 메일박스에 메일이 없다면 insert문을 수행
SQL="INSERT INTO mail(subject,to_user,from_user,content,s_date
...
}else{ ' 해당 사용자의 메일박스에 메일이 있다면 update문을 수행
SQL="UPDATE mail SET subject=" + strSubject + ",to_user"+
}
...
objFs.DeleteFile(Server.MapPath("./tmp/" + strPath),true);
```

그림 7. XMLtoDB 파일 해석 코드

4. DBtoXML 변환기

데이터베이스로는 SQL Server 2000을 사용하며 ADO 2.5(ActiveX Data Object)의 Write 메소드를 이용하여 구현하였다. 즉, 가져온 레코드셋을 바탕으로 XML문서를 작성할 수 있다[6]. 현재 소개된 DBtoXML 변환에 관한 방법들은 데이터 내용을 매핑하여 XML 트리 구조로 출력하는 것과 데이터베이스의 기본적인 변환기능을 이용하는 것으로 나눌 수 있다[7,8].

본 논문에서 제시한 시스템에서는 데이터베이스에서 가져온 데이터를 XSLT 스타일시트를 이용하여 출력하는 방법을 이용하였다. 이런 방법은 데이터베이스의 테이블의 구조와 상관없이 XML로 변환이 가능하며 XSLT도 고정적인 트리 형태가 아닌 최대한의 보편성을 가질 수 있도록 구성 가능하다. mail이라는 테이블에 저장된 레코드를 가져오게 되고, 레코드셋이 EOF에 도착 할 때까지 루프를 반복하면서 각각의 엘리먼트들을

가져오게 되고 write 메소드를 이용해서 정리할 수 있다. 또한 ContentType, AddHeader를 이용해서 파일로 다운로드받을 수도 있다. 그림 8에서의와 같은 구조로 하나의 레코드는 하나의 FieldName N을 구성하게 되며 표 3은 각 노드 명을 설명하고 있다.

표 3. DBtoXML 변환 구조

노드명	노드 개요
<MAIL>	데이터베이스의 추출 결과 / XML 루트 요소
<r_mail>	하나의 레코드 정보를 정의
<fieldName N>	각 필드의 정보를 정의 (요소명은 필드명에 대응)

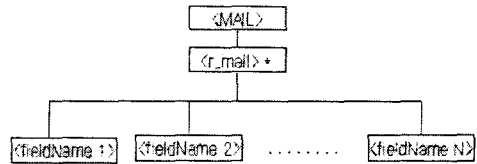


그림 8. DBtoXML 변환 구조

V. 구현 결과 및 검토

먼저 사용자는 XML파일로 변환된 메일을 서버에 등록 할 수 있으며, 사용자는 데이터베이스에 등록시킬 파일을 선택하고, 서버는 이 파일을 유효성검사를 거쳐 데이터베이스에 등록한다. 그림 9는 등록할 XML파일이 데이터베이스에 등록된 모습이다. 사용자는 데이터베이스에 등록되어 있는 자신의 보관 편지함에 있는 메일들을 XML파일로 변환할 수 있다.

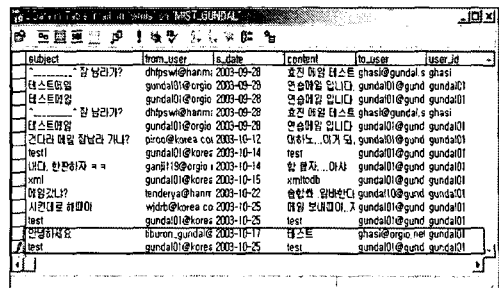


그림 9. 메일이 RDB에 등록된 화면

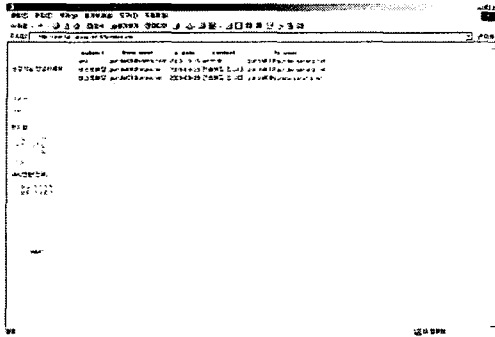


그림 10. XSLT를 이용해서 브라우저에 출력

변환되어진 사용자의 XML 파일들은 XSLT를 이용하여 웹 브라우저에 출력되며 사용자는 임의로 저장할 수 있다. 그림 10에서는 XML로 변환되어진 메일 데이터와 XSL을 이용해 브라우저에 출력된 것을 확인할 수 있으며, 그림 11은 변환된 데이터를 보여주고 있다.



그림 11. XML로 변환된 데이터

사용자가 전자우편을 송수신할 때 사용하는 웹 메일 시스템과 클라이언트/서버 시스템은 고정적인 출력 형식과 메일 데이터의 이용과 사용자들의 개인 정보의 활용에 있어 많은 제약이 있었다. 표 4는 기존의 웹 메일 시스템과 클라이언트/서버 시스템과 본 논문에서 제시한 시스템을 비교 분석하였다.

먼저 접근성 측면에서 비교하면 웹 메일 시스템과 WMIS는 인터넷이 가능한 어느 지역에서도 메일을 사

용할 수 있어 접근성이 좋다. 하지만 클라이언트/서버 메일 시스템은 아웃룩 익스프레스와 같은 메일 전용 어플리케이션을 설치하고, 메일 계정을 획득해야 하므로 접근성이 좋지 않다. 또한 WMIS는 메일 데이터를 데이터베이스에 저장하므로 데이터의 독립성을 유지할 수 있으며 개인화가 가능하다. WMIS에서는 메일 데이터를 구조화된 문서인 XML로 저장하기 때문에 가독성이 좋다. 그리고 무선 인터넷 언어인 WML은 XML의 서브셋(subset)이므로 XML로 저장된 데이터를 바로 무선 인터넷 응용에 사용할 수 있다.

표 4. 메일 시스템의 특징 비교

	웹 메일 시스템	C/S 시스템	WMIS
접근성	쉽다	어렵다	쉽다
데이터의 개인화	불가능	가능	가능
데이터의 독립성	없다	있다	있다
구조화된 문서	없다	있다	있다
재사용성	없다	없다	있다
확장성	없다	없다	있다
보관성	없다	있다	있다
이동성	없다	있다	있다
무선인터넷 지원	없다	없다	가능
가독성	없다	없다	있다

VI. 결론

인터넷을 이용한 정보의 양이 증가하면서 개인이 필요로 하는 정보의 양도 증가하고 있다. 그 단적인 예로, 자신에게 온 중요한 메일과 웹 메일의 개인 환경설정 정보 또한 필요하다.

본 논문에서 제시한 XML을 이용한 웹 메일 통합 시스템은 기존의 웹 메일 시스템과 클라이언트/서버 메일 시스템의 편리성과 XML기술의 장점을 하나의 시스템에서 통합하였다. 웹에서 메일 데이터와 사용자 환경을 효율적으로 잘 활용할 수 있도록 XML로 변환하고, 개인정보를 잘 이용하여 웹 상태에서 메일 데이터와 사용자 정보를 통합 할 수 있는 방법을 제시하였다. XML은 문서 자체가 데이터를 구조화하여 표현하고 있으므로

작은 데이터베이스라 이해할 수 있다. 그래서 자신의 중요 메일이나 사용자환경 등을 XML파일로 정리해서 사용자의 PC에 다운받을 수 있다. 개인의 정보 보호나 네트워크 트래픽을 줄일 수 있고, 사용자의 개성에 맞게 XSL문서를 생성할 수 있어 다양한 형식의 사용자 인터페이스를 제공할 수 있다. 또한 XML로 메일 데이터를 변환하여 무선 인터넷 시스템을 개발 할 경우 쉽게 무선 인터넷 언어인 WML을 지원 할 수 있다. 즉 서로 다른 데이터, 어플리케이션, 디바이스 사이의 통합이 가능해 질 수 있다. 이로써 시스템의 개발에 따른 인력, 시간, 개발비 등을 줄일 수 있어 효율적인 개발 방법을 제시 할 수 있다. 그리고 사용자가 메일계정을 XML변환을 지원하는 웹 메일 계정으로 변경하면 사용자는 손쉽게 자신의 정보를 재구축 할 수 있다. 그리고 메일 계정을 변경하여도 사용자는 불편을 느끼지 않게 되므로 벤더들에게는 새로운 비즈니스 모델을 제시할 수 있다.

향후 연구해야 할 과제는 XML 문서에 대한 보안이 필요하고, 개인이 설정한 스팸메일 차단 정보의 변환에 대해서도 연구가 필요하다. 로컬 시스템에서 데이터 검색, 수정을 할 수 있는 전용 어플리케이션을 개발하고, 무선 인터넷 상에서의 접근하여 이용할 수 있도록 메일 데이터를 WML로 변환하는 방법과 XML을 데이터베이스로 변환하는 과정에서 복잡한 계층 구조를 표현하는 방법에 대한 연구가 필요하다.

참 고 문 헌

- [1] 박순영, "XML기반의 웹메일 시스템의 설계 및 구현", 한국정보처리학회 2001년도 추계학술발표논문집, pp. 233-236, 2001.
- [2] Jonathan Borden, "XML MIME Transformation Protocol(XMTP)," The Open Healthcare Group, 1998.
- [3] <http://www.w3.org/TR/FR C-XML>
- [4] 이정수, "XML 데이터베이스 시스템을 기반으로 한 B2B 통합 시스템 개발", 한국정보처리학회 2002년도 춘계학술발표논문집, pp. 112-116, 2002.

- [5] http://matilda.snu.ac.kr/doc/analysis/wap_mail.htm
- [6] Joo Kyung Soo, "A Design of Middleware Components for the Connection between XML and RDB," 2001 IEEE International Symposium on Industrial Electronics Proceedings, Pusan, Korea, June, 2002.
- [7] 김영선, 장덕철, "XML Parser 추출에 의한 모바일 콘텐츠 변환 설계", 멀티미디어학회 논문지, 제6권, 제2호, pp. 267-275, 2003.
- [8] <http://www.emwac.ed.ac.uk>

저 자 소 개

권 정 식(Jung-Sik Kwon)

준회원



- 2000년 2월 : 안동대학교 컴퓨터 공학과(공학사)
- 2004년 2월 : 안동대학교 대학원 컴퓨터공학과(공학석사)
- <관심분야> : XML, 메일 시스템, 서버기술

조 대 제(Dae-Jea Cho)

종신회원



- 1986년 2월 : 경북대학교 대학원 전자공학과(공학석사)
- 2000년 8월 : 경북대학교 대학원 컴퓨터공학과(공학박사)
- 2002년 9월~현재 : 안동대학교 멀티미디어공학과 교수

<관심분야> : IT, 문화 콘텐츠, 콘텐츠 보안