

인터넷 디스크 부가 서비스 Webmailer 구현

한진우, 민경왕, 정태일, 장현희, 박성순
안양대학교 컴퓨터공학과

Webmailer Implementation for Internet Disk

Jinwoo Han, Kyungwang Min, Teail Jung, Hyunhee Jang and Sung-Soon Park
Dept. of Computer Engineering, Anyang Univ.

요 약

현재 인터넷 디스크는 앞으로의 발전을 위하여 Webmailer의 기능을 부여하는 등 여러 기능을 추가하는 것이 필요하다. 그리고 두 Application 사이에 장점을 더욱 큰 장점으로 발전 시킬 필요성이 있다. 본 논문에서는 Linux 기반의 스토리지에 메일 발송 기능을 추가하여 메일을 이용한 스토리지상의 DATA 전송 방법을 제시한다. 그리고 메일의 첨부파일에 대한 관리 방법에 대하여 제안하여 실제 메일 시스템을 구성하였다.

1. 서론

네트워크 속도의 향상과 하드웨어의 발전으로 인터넷 디스크를 사용하는 사용자들의 수는 점차 늘어나고 있다. 인터넷 디스크는 자신의 하드웨어 웹상에 두고 장소에 구애 받지 않고 자료를 사용할 수 있는 점과 안전하게 자료를 저장시켜 놓을 수 있는 커다란 장점을 가진다. 하지만 다양한 사용자의 증가를 위해서는 이러한 장점 위에 좀더 부가된 기능의 추가가 필요하다. 그리하여 자료의 보관이라는 특징을 가진 인터넷 디스크와 인터넷 사용자들이 가장 많이 사용하는 Application 중에 하나인 메일과의 연계는 위에서 제기 되고 있는 부가기능 들을 해결할 수 있는 하나의 방안이 된다. 인터넷이 접속된 어느 장소에서든 메일 사용과 저장 자료의 사용이 가능하다. 이는 인터넷 디스크 사용 빈도수의 증가와 더불어 많은 사용자들이 메일을 인터넷 디스크 안에서 사용할 수 있도록

유도한다. 여기서 메일이란 단순한 메일의 전송을 의미 하는 것이 아니다. 일반 포탈 사이트에서 제공하고 있는 Webmailer 의 기능을 가진다. 인터넷 디스크 장점을 이용하여 좀더 확장된 의미의 Webmailer 구현에 목적을 두고 있다. 이 Webmailer 는 모든 형식의 메일을 송수신하여 정확히 메일 헤더를 분석한다. 그리하여 안정적인 Webmailer 의 기능을 수행한다.

확장된 Webmailer 의 가장 큰 기능으로는 메일의 용량 제한으로 인해 메일 수신에 있어서 문제가 되었던 부분의 해결이다. 그리고 인터넷 디스크 안의 파일에 대한 타 사용자들과의 공유에 대한 확장방법이다. 그리하여 메일로 파일 전송시 첨부 파일의 용량의 한계로 인해 대용량의 파일을 전송하지 못하게 되는 단점을 보완하였다. 실제적인 첨부파일의 전송이 아닌 파일위치에 대한 정보만을 보내주어 링크되어 있는 파일에 대한 전송이 가능해졌다. 그리고 그 동안 받아온 메일들에

대해서 따로 보관 할 수 있는 장소를 제공함으로써 메일의 내용과 첨부파일들에 대하여 자신의 인터넷 디스크에 안전하게 보관하는 것이 가능하다.

본 논문의 구성은 다음과 같다. 먼저 2 장에서는 POP3, IMAP, SMTP/MIME 등의 메일 프로토콜에 대한 관련연구와 3 장에서는 Webmailer 의 구성에 대하여 기술하고 4 장에서는 첨부 파일 처리와 대용량 파일의 전송 방법에 대하여 기술한다. 그리고 5 장에서는 향후 발전 방안, 결론에 대해서 제시한다.

2. 관련 연구

SMTP Protocol은 Email을 보낼 때 사용되는 TCP/IP Protocol로서 Sendmail을 사용하여 인터넷 디스크 웹 페이지에서 메일 발송을 처리한다. 그러나, 수신 측에서의 큐 메시지능력의 제한으로 인해, 대개 수신을 위해서는 POP3나 IMAP 중의 하나의 프로토콜을 쓰는 것이 보통이며, 이러한 프로토콜은 서버에 각자의 우편함을 두고 사용자 메시지를 저장한 다음, 주기적으로 서버에 접속하여 편지를 다운로드 하는 식으로 운영된다[4].

POP3 전자우편을 수신하기 위한 표준 프로토콜로서 가장 최신 버전이다. POP3는 인터넷 서버가 사용자를 위해 전자우편을 수신하고 그 내용을 보관하기 위해 사용되는 클라이언트/서버 프로토콜이다. 사용자는 주기적으로 서버에 있는 자신의 메일 수신함을 점검하고, 만약 수신된 메일이 있으면 클라이언트 쪽으로 다운로드 한다[3].

IMAP 은 로컬서버에서 전자우편을 액세스하기 위한 표준 프로토콜이다. IMAP은 인터넷 서버를 이용하여 전자우편을 수신하고 보관하는 클라이언트/서버형 프로토콜이다. 사용자 (또는 사용자의 전자우편 클라이언트 프로그램)는 편지의 제목과 송신자를 보고, 메일을 실제로 다운로드할 것인지를 결정할 수 있다. 사용자는 서버에 폴더나 우편함을 만들거나 관리할 수 있으며, 메시지를 지우거나 메시지의 일부 또는 전체의 내용에 대해 검색을 수행할 수 있다[5].

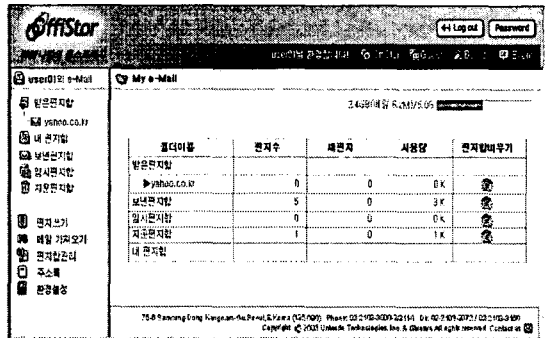
MIME은 아스키 데이터만을 처리할 수 있는 원래의

인터넷 전자우편 프로토콜, 즉 SMTP를 확장하여 오디오, 비디오, 이미지, 응용프로그램, 기타 여러가지 종류의 멀티미디어 데이터 파일들을 주고받을 수 있도록 기능이 확장된 프로토콜이다. 서버들은 어떤 웹 전송에라도 시작부분에 MIME 헤더를 집어넣으며, 클라이언트들은 헤더가 나타내는 데이터 형식에 따라 이를 재생시키기 위한 적절한 응용프로그램을 선택하여 실행 시킨다[2].

3. Webmailer 구성 및 세부 기능 설명

Webmailer 부분의 전체적인 구성은 그림 [1]과 같이 구성된다. 각각의 편지함과 세부적인 기능을 사용할 수 있는 메뉴를 메인 창에 모두 보여 주어 사용자가 보다 편리하게 사용할 수 있다. 받은 편지함은 POP3 과정을 통하여 가지고 온 메일들이 각각의 편지함 별로 구분하여 놓았다.

그리고 임시 편지함을 두어 메일 발송 시 서버의 문제로 정상적으로 발송되지 않은 메일들에 대하여 이후 다시 발송 할 수 있도록 따로 저장 시킬 수 있는 기능을 가진다.

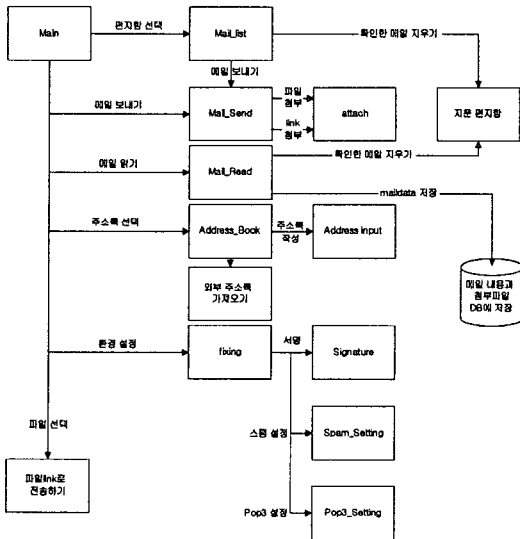


[그림 1] 메일 메인 화면 구현도

편지 쓰기는 POP3 설정을 해놓은 주소 중에서 보내는 주소를 선택하여 상대방에게 메일을 전송할 수 있도록 하여 준다. 편지 쓰기에서의 가장 큰 특징은 메일 첨부에 있다. 첨부 파일을 자신의 local computer에 있는 파일에 대한 첨부과 인터넷 디스크 상에 있는 파일에 대한 첨부를 모두 가능 하도록 구성했다. 그리고 인터넷 디스크의 파일을 첨부하는

경우에는 실제로 파일을 메일에 첨부하는 형태와 대용량 파일에 대하여 파일의 위치를 링크시켜 메일내용에 첨부하여 보낼 수 있는 2가지의 형태로 구현했다.

메일을 가져오는 POP3 부분의 경우는 각각의 서버에 접속하여 새로 받은 메일을 가져 오도록 된다. 메일을 가져오는 과정에서 각 메일에 첨부 되어 있는 첨부 파일은 maildata라는 인터넷 디스크의 폴더에 메일의 헤더를 분석하는 과정에서 자동으로 저장된다. 그리고 메일내용을 가지고 있는 .eml 파일의 경우에는 maildata 폴더 속에 보이지 않는 형태로 보관하여 사용자가 임의로 메일 내용을 가진 파일을 삭제하는 것을 방지시켰다. 이러한 과정으로 인해 메일의 내용을 자신의 인터넷 디스크가 허용하는 범위 내에서 얼마든지 보관 하는 것이 가능하다.



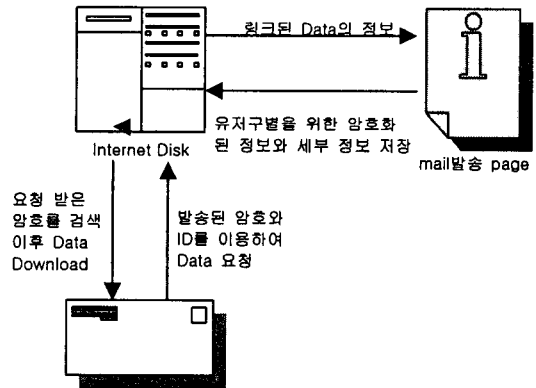
[그림 2] 메일 서버 구성도

그 밖에 세부 적인 기능들의 구성은 그림 [2]에 자세히 보여진다. 위의 기능들 중 메일 주소록에서 외부 주소록 가져오기는 다른 Webmailer 들의 주소록 정보를 가져올 수 있도록 작성되었다. MS Outlook 과 Outlook Express 의 주소록의 많은 정보를 가져와 본 Webmailer의 주소록에 필요 정보만을 저장시킨다. 이는 모든 정보를 비교하여 두 Webmailer의 많은 정보들 중 주소록에 필요한 정보만을 걸러 가

져오는 필터링 과정을 거치게 된다. 이외의 기능들은 여타 Webmailer에서 볼 수 있는 모든 기능을 제공하고 있다. 그리하여 사용자들이 인터넷 디스크를 Webmailer와 연동 하여 사용하는데 불편함이 없도록 하였다.

4. 파일 첨부 기능과 DATA 처리 과정

인터넷 디스크의 대용량 파일에 대한 전송 방법으로는 2가지 방식이 있다. 우선 웹 스토리지 상의 파일을 선택하여 메일 본문 속에 링크로 타 사용자가 받아 갈수 있도록 하여 전송하는 것이다. 이것은 여러 상용 서비스에서 지원하는 것으로서 파일정보에 링크를 걸어 파일을 download 한다. 하지만 본 Webmailer에서는 다른 메일의 첨부기능과 같이 local에 있는 파일을 첨부하는 것 이외에 스토리지 상의 파일에 대해서도 첨부가 가능하게 구성 되었다. 만일 용량이 큰 파일을 메일로 전송하길 원할 경우에는 웹 스토리지 상의 파일에 한하여 링크로 전송하기라는 기능을 제공한다. 이로서 대용량 파일의 전송이 가능해졌다



[그림 3] 파일 링크 연동 방법

파일을 링크로 전송 하여 받는 방법에는 다음의 과정들이 이루어진다. DB에 링크로 메일을 전송한 파일에 대한 정보를 저장 하고 링크된 파일에는 2가

지 정보를 유지 한다. 스토리지상의 DB에서 정보를 비교하여 요청한 파일에 대한 download를 허용한다. 하나의 정보는 파일 소유주의 ID 이고 다른 하나의 정보는 각각의 파일에 주어진 password이다. 위의 두 가지 정보는 인코딩 되어 전송된다. 인터넷 디스크에서는 요청된 정보에 대해서 디코딩 하여 값을 비교한 이후에 파일의 download 허가 여부를 결정한다.

이러한 기능은 인터넷 디스크의 사용자들 사이의 자료 공유를 도와줄 뿐 만 아니라, 인터넷 디스크의 사용자가 아닌 경우에도 자신의 data를 전송 받을 수 있도록 하는데 사용된다. 그림 [3]에서는 메일의 발송과 파일 download 과정에서 이루어지는 일을 도식화하여 보여준다.

또 하나의 인터넷 디스크와의 연동 방법으로는 첨부파일을 저장하는 방법이다. 보통 메일을 받아 읽어들이면 첨부 파일을 메일 내용 안에 보관하기 때문에 메일 사용자들만 첨부파일에만 접근하는 것이 불가능했다. 하지만 인터넷 디스크의 저장공간을 이용하면 첨부 파일에 대한 관리를 따로 할 수 있다. 메일의 헤더를 분석하여 첨부 파일을 걸러내는 과정 중에 첨부 파일을 자신의 저장공간의 일부분에 따로 분류 한다. 그래서 사용자가 따로 저장 하지 않더라도 알아서 파일을 저장한다. 자료저장의 기능을 이용하여 첨부파일들을 안전하게 보관 한다. 뿐만 아니라 첨부 파일들에 대하여 웹 상에서 내용을 볼 수 있다. 이는 첨부파일들의 관리뿐만 아니라 받은 메일을 관리하는데 있어서도 많은 이득을 제공한다.

인터넷 디스크에 연동된 Webmailer는 제한된 저장공간을 사용하여야 하는 다른 메일 사이트들의 단점과 자신의 컴퓨터에서만 메일을 확인할 수 있는 Webmailer들의 단점을 보완 하였다. 이는 웹 상이던 어디에서라고 접근이 가능하고 자신의 스토리지 할당량 만큼은 얼마든지 메일을 저장 시킬 수 있기 때문에 가능해졌다. 충분히 다른 Webmailer들을 대처할 수 있는 기능을 가지고 있으며 현재의 인터넷 디스크의 활용성을 좀더 확장 시킬 수가 있다.

5. 고찰 및 결론

앞으로 POP3 과정을 구현함에 있어서 Linux C를 이용한 소켓 통신을 통한 방법으로 웹 상에서 독립적으로 작동되는 부분이 구현 중에 있다. 그리고 여타 Webmailer들의 주소록과의 연동을 위해서 ActiveX를 이용한 주소록 정보의 상호 교환에 대한 부분도 관련 API를 분석하는 작업을 진행하여 구현하는 것을 목표로 한다.

메일은 인터넷 사용자들이 검색과 더불어 가장 많이 사용되어지는 Application이다. 그리고 이는 앞으로도 오랫동안 지속될 것이다. 이에 더욱 다양하고 편리한 기능이 추가되어야 한다. 본 논문에서는 인터넷 디스크와의 연동을 통해 하나의 방안을 제시하고 있다. 이에 현재의 기능뿐만 아니라 더욱 발전된 기능으로 상호 발전 가능한 Application으로의 개발이 필요 하다.

[참고문헌]

- [1] David Wood, "인터넷 이 메일 프로그래밍", O'Reilly 1999.
- [2] RFC 1341, MIME (Multipurpose Internet Mail Extentions), <http://www.faqs.org/rfcs/rfc1341.html>, June 1992.
- [3] RFC 1939, POP3(Post Office Protocol – Version 3), <http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1939.html> May 1996.
- [4] RFC 821, SMTP(Simple Mail Transfer Protocol), <http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc0821.html> August 1982.
- [5] RFC2060, IMAP (Internet Message Access Protocol Version 4rev1.) <http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc2060.html> , August 1996.
- [6] 박성수, "리눅스 서버관리 실무 바이블", 대한민국서버관리자그룹, 2002
- [7] OWL 연구소, "PHP 웹 솔루션", PCBOOK, 2000
- [8] Richard Stone & Neil Matthew, "Linux Programming", 정보 문화사, 2000
- [9] Nigel McFarlane 외 7인 공저, "PROFESSIONAL JavaScript", 정보 문화사, 2000