

인터넷 우체국을 위한 Front Agent 의 설계 및 구현[†]

권오형*, 정준영*, 이준석**, 윤기송**, 정민수*, 조유섭*

*경남대학교 컴퓨터공학과

**전자통신 연구원

Design and Implementation of Front Agent for Electronic Post-Office

Oh-Hyung Kweon*, Joon-Young Jung*, Joon-Suk Lee*

Gi-Song Youn*, Min-Soo Jung*, You-Sub Cho*

*Dept. of Computer Engineering, Kyungnam University

**Electronics and Telecommunications Research institute

요 약

인터넷 우편 시스템은 인터넷에서 고객이 작성한 우편을 메일 뿐만 아니라 이를 직접 인쇄하여 다수의 수신자에게 보통우편으로 제작하여 직접 배달하는 우편서비스로 인쇄부터 봉합까지 자동으로 처리하는 시스템이다. 따라서 직접적인 우편을 전달함으로써 기존의 디지털 문명의 의한 각박해진 현 사회에 보다 인간미가 넘치는 매개체로서의 역할과 획기적인 시간 단축과 비용절감의 효과를 누리하고자 한다. 현 인터넷 우편 시스템은 크게 3 가지로 Front Agent, End Agent, Management Agent 나뉘어 지는데 본 논문에서는 Front Agent 를 중심으로 서술하였다.

1. 서 론

인터넷 우편 시스템은 네트워크망을 이용하여 기존에 널리 퍼져 있는 전자메일을 통한 메일 서비스가 아닌 인터넷을 통해 기존의 전자 메일 우편 기술을 우편 배달 시스템과 연동하여 사용자가 직접 웹 상에서 작성한 텍스트, 기존의 아래아 한글 파일과 워드 파일에서 작성한 문서를 저렴한 비용으로 실제로 신속, 정확하게 다수의 수신자에게 배달 하기 위한 기술이다.

본 논문의 구성은 다음과 같다. 2 장에서는 인터넷 우편 시스템의 Front Agent 부분에 대해 기술하고, 3 장에서는 Front Agent 구조와 구현, 웹 서버와의 통신을 제시하고, 4 장에서는 결론 및 향후연구에 관하여 기술하였다.

2. 전처리국(Front Agent)

2.1 인터넷 우편 시스템의 분석

인터넷 우편 시스템은 세부적으로 Front Agent, CGI, End Agent, Management Agent, Data Base, Web Server, Mail Server 등으로 구성된다.

이 논문에서는 Front Agent를 중심으로 서술하였는데 이

는 사용자에게 우편을 작성할 수 있는 환경을 제공하고 작성된 문서를 MIME 메시지로 만들어 CGI프로그램을 통하여 웹서버로 전송하는 역할을 하고 사용자의 로컬 시스템에 접근하기 위한 애플릿 보안과 인증, 사용자가 작성한 문서의 페이지 수를 구한다.

	기능
Front Agent	<ul style="list-style-type: none"> • 사용자의 인터페이스 작성 • MIME 작성 • 애플릿 보안 • 한글과 워드 파일의 페이지수 구하기 • 사용자 정보 임시저장(Applet)
End Agent	<ul style="list-style-type: none"> • POP3 기술 • 한글/워드 첨부 문서 프린트 • Bar Code 명세
Management Agent	<ul style="list-style-type: none"> • 관리자 모듈 프로그램
CGI	<ul style="list-style-type: none"> • SMTP 기술 • Front Agent 에서 받은 데이터를 검사, 처리하여 메일 서버로 전송 • 메일관련 데이터를 DB 에 저장

[표 1] 인터넷 우편 시스템의 기본 구조

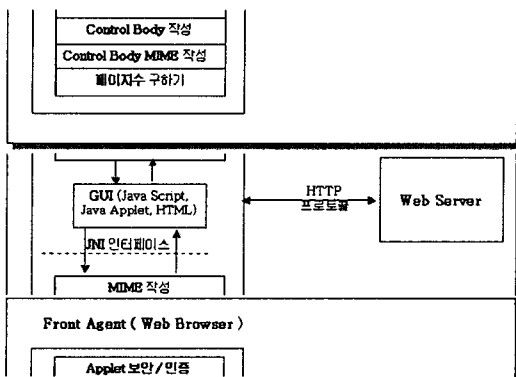
표 1은 인터넷 우편 시스템에서 필요한 기능에 대해 간략하게 소개하고 있다.

2.2 Front Agent 설계

[†] 본 연구는 2000년도 ETRI 용역 과제의 지원에 의해 수행됨

Front Agent 는 사용자에게 직접 보여주는 모듈로서 사용자의 편의성에 맞춰야 한다. Front Agent는 사용자의 인터페이스를 작성하기 위해 Java Script와 Java Applet을 MIME작성을 위해 C++, Applet와 C++의 상호 통신을 위해 JNI를 사용하였다. 사용자가 웹 브라우저를 이용하여 메일을 작성하여 전송하면 이를 MIME 메시지로 만들고, 수신자와 송신자의 정보를 포함한 통제(Control) 정보를 한 개의 body로 만들고 이것을 HTTP 프로토콜을 이용하여 웹 서버로 전송한다.

Front-Agent 모듈은 그림 1과 같이 크게 GUI, MIME작성, 통제 바디(Control Body) 작성, 페이지수 구하기, 그리고 HTTP 프로토콜로 구성된다.



[그림 1] Front Agent 구조

2.2.1 GUI

사용자 인터페이스는 Java Script, Java Applet, JNI, C++를 이용하여 개발하는데 Applet은 사용자가 입력한 데이터를 임시 저장하거나 정보를 보여주기 위해서 사용되었다. 현 시스템을 사용하기 위해 필요한 DLL을 사용자의 로컬 시스템에 설치하고 현 시스템에서 지원하는 이미 사용자 시스템에 저장된 파일을 read, write 하기 위해서 인증을 해야 한다. 여기서 현 시스템을 사용하기 위한 DLL이 하는 역할은 한글문서나 워드 문서의 페이지 수를 구하거나 현 시스템 사용을 위한 버전 확인(Version Check)등을 수행한다.

이러한 현 시스템을 위해 사용하기 위해 필요한 DLL을 사용자의 로컬 시스템(Local System)에 설치하는 것은 애플릿 보안에 위반 되는 행위이다. 따라서 현 시스템에서는 애플릿 전자서명을 위해 인증기관(CA)으로부터

소프트웨어 서명용 전자 인증서를 발급 받아서 사용하여야 하는 대신 테스트용 서명도구 중 하나인 SDK for Java 4.0을 이용하여 테스트용 인증서를 만들어 사용하였다.

JNI는 Applet을 통하여 특수한 해결 방안을 마련하기 위하여 네이티브(Native) 코드를 사용하는 경우가 존재하는데 이는 클래스 라이브러리만으로는 작성하지 못하는 프로그램을 위해서 네이티브 코드 라이브러리를 사용하여 해결한다. 현 시스템에서 JNI는 C++로 작성된 모듈을 이용하고 첨부 파일(한글파일, MS word 파일) 등의 파일 열기의 대화상자를 위한 사용자의 하드디스크 접근을 위해 사용되었다.

2.2.2 MIME(Multi-purpose Internet Mail Extension)

기존의 E-mail은 송수신 할 수 있는 정보는 7bit로 된 ASCII로 구성되어 단지 텍스트만을 전송함으로써 멀티미디어 데이터를 전달할 수 없는데 이러한 한계를 극복하고자 MIME 작성하는데 다시 말하면 인터넷 전자 메일을 통하여 여러 다른 종류의 파일들을 전송가능 하게 하기 위해 C++를 이용하여 작성하였다.

2.2.3 Control Body MIME

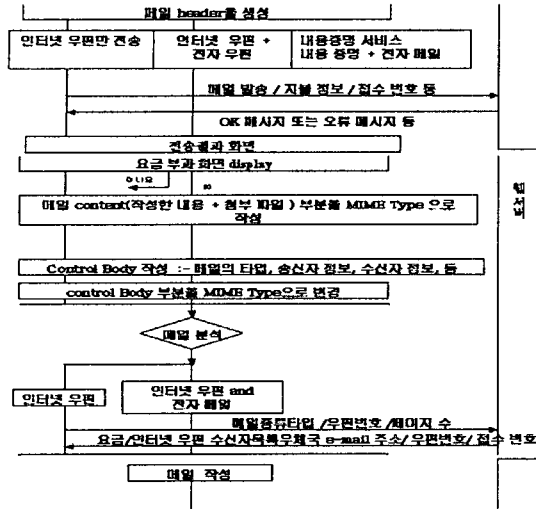
인터넷 우편을 보내기 위해 필요한 정보를 MIME에 추가 시켜 보내기 위함으로 C++를 이용하여 작성하였다.

2.2.4 페이지수 구하기

페이지수 구하기 모듈은 사용자가 작성한 문서(첨부 파일)의 페이지수를 구하기 위함으로써 현 시스템에서는 아래아 한글과 MS Word 그리고 사용자 글쓰기 입력창에서 직접 입력한 글에 대한 페이지 수를 구하는데 한글 파일은 헤드 파일을 검색하여 페이지수를 구하며 MS Word는 MS Word Automation을 이용하여 페이지수를 구하며, 사용자가 시스템에서 제공하는 글쓰기 입력창을 통하여 입력한 텍스트 파일에 대한 글쓰기 페이지수의 계산은 C++에서 작성한 모듈에 접근하기 위해 JNI를 이용한 DLL의 함수를 호출하여 해결하였다.

3. Front Agent 구조와 구현

웹 브라우저에서 사용자가 필요한 정보를 입력, 메일을 작성한 후 웹 서버로 데이터 전송하고, 받는 구조의 흐름도는 그림 2와 같다.



[그림 2] Front Agent 전체 흐름도

웹 브라우저 초기화면에서 사용자가 메일 작성과 전송에 필요한 데이터를 입력한 후 전송 하면 CGI를 통해 데이터가 처리되면서 Data base에 필요한 정보를 등록하고 CGI에서 처리된 데이터는 다시 브라우저로 전송하게 되어 이 데이터를 이용하여 요금부과 화면을 생성한다. 요금부과 화면에서는 사용자가 보낸 인터넷 우편에 대한 결제를 담당하는 부분으로 사용자가 결제 수단을 선택한 뒤 CGI로 데이터를 전송하게 되면 이때 JNI를 통해 MIME작성 모듈을 호출하여 메일에 대한 MIME를 작성한다. CGI에서 데이터를 처리하고 다시 처리된 데이터를 웹 브라우저로 전송하여 전송결과 화면을 생성한다.

3.1 애플릿 보안과 인증

처음 초기화면에서 현 시스템을 Load 하였을 경우 필요한 DLL의 설치 여부의 확인과 DLL 설치, 버전을 확인 하고 사용자가 미리 정의한 E-Mail, 수신자 정보 파일을 접근하기 위해서는 로컬 시스템의 접근이 불가피하다. 애플릿을 통한 사용자의 로컬시스템의 접근은 애플릿 보안에 위배되는 행위로서 이러한 보안제한을 완화하기 위하여 사용자에게 로컬에 접근할 수 있는 권한을 얻어야한다. 이를 위해서는 인증기관으로부터 소프트웨어 서명용 전자 인증서를 발급 받아야 하지만 현 시스템에서는 테스트용 서명 도구를 사용하였다. 소프트웨어 서명용 전자인증서로는 Netscape Object Signing Certificate, Microsoft Authenticode Certificate 이 있으며 테스트용 서명도구로는 Netscape (SignTool), IE (SDK for Java), JDK1.1 (javakey) 등이 있다. 각 브라우저마다 (SUN, Netscape, IE)는 전자 서명 방식이 다르며, 서로 호환되지 않는데 현 시스템에서는 익스플로러에서 전자 서명을 위한 SDK for Java 4.0 툴을 이용하였다. 테스트용 인증서를 구현하는 방법은 다음과 같다.

```

① makecert -sv Test.pvk -n "CN=인증서" Test.cer
② cert2spc Test.cer Test.spc
③ cabarc n mycab.cab *.class
④ signcode -spc Test.spc -v Test.pvk mycab.cab
    
```

[그림 3] 테스트용 인증서 생성

① 공개키, 비밀키 쌍이 자동적으로 생성되어 비밀키는 레지스트리에 저장되며 테스트용 인증서가 생성된다.
 ② SDK 는 .spc 형식의 인증서를 읽을 수 있으므로 반드시 인증서 파일명을 .spc로 변환해 주어야 한다.
 ③ 파일들(.class)을 .cab 파일로 만든다.

④ 인증서를 .cab 파일에 포함시키고 비밀키로 전자 서명한다.

3.2 애플릿 간의 통신

초기화면과 수신자정보를 입력하는 화면 간의 데이터의 상호 전달을 위해서 static class 를 사용하였다. Class 생성시 static 를 부여하면 class 는 메모리 하나만 생성하게 된다. 따라서 서로 다른 웹페이지간의 통신을 위해서 static class 를 생성하면 결국 하나만 생성되는 것인데 결국은 같은 메모리의 객체를 얻게 됨으로서 static class 의 변수나 함수를 통해 서로의 통신이 가능하게 된다.

3.2 MIME 작성

인터넷 전자 메일을 통하여 여러 다른 종류의 파일들을 전송 가능하게 하기 위해 MIME 작성을 하는데 우선 사용자가 작성한 메일의 데이터에 대한 encoding 작업과 line delimiter 변경 작업을 수행한다.

MIME 의 구성은 MIME 의 주 Header 부분, Content-Type(내용 형식)부분, Content - Transfer - Encodiong header(내용 전송부호화)부분으로 나뉘어져 있다.

MIME 버전 Header 부분은 송수신된 메시지를 적절히 해석할 수 있도록 MIME 버전을 기술하는 부분이고, Content Type 은 MIME 에서 정의하고 있는 데이터 형식에는 7 가지로서 text/plain, Video, Application, Image, Audio, Mutipart, Message 가 있다. 현 시스템에서는 Plain text, Application 을 지원한다. Charset 은 영문으로 작성되었을 경우 charset="us-ascii" 이고 한글이 포함 되었을 경우 charset="ISO-2022-KR"이며 다른 문자는 지원하지 않는다. 파일 이름을 가질 경우 헤더 필드의 확장이 필요로 하는데 name 에서 정의를 한다. 헤더에 ASCII 문자 이외의 것이 들어가기 위한 방법이다.

encoded-word := "=?" charse "?" encoding "?" encoded-text "=?"

=? 문자집합 ?encoding 방식 ?=
 문자집합 : US-ASCII, EUC-KR 등을 뜻한다.
 Encoding 방식은 B,Q 가 있는데 본 시스템은 B 만 지원한다.

-“B” Encoding : base64 Encoding 을 이용한다.
 -“Q” Encoding : Quoted-Printable 와 유사한 Encoding 방식을 이용한다.

Content-Transfer-Encoding 는 base64 를 사용하는데 이는 본문이 인코딩 되는 방식을 표시하기 위함이다. 따라서 마지막에 세미콜론 뒤에 오는 base64 라든가 Quoted Printable 라고 적혀 있으면 그러한 방식을 인코딩 하였다는 뜻이다. 인코딩 방식에는 7bit, 8bit, binary, Quoted-Printable, x-token 등이 있다. 이중 7bit, 8bit, base64, Quoted-Printable 등이 많이 사용하는데 요즘에는 한글은 base64 로 인코딩 하는 걸 원칙으로 하고 있다. 현 시스템에서는 base64 를 사용한다. base64 방식은 24 bit, 즉 3byte 를 입력 받아서, 이를 6bit 씩 잘라 4byte 씩 묶어 아스키 형태로 출력하는 인코딩 하는 방식이다. 이는 각 바이트의 상위 두 비트를 0 으로 함으로서 ASCII 문자라는 것이 보장된다. 따라서 데이터의 크기가 원래 보다 33%정도 커지게 되는데 원래의 데이터가 인코딩 한 후에 크기가 얼마정도 일지를 예측할 수 있으며 다른 인코딩 방식에 비해 공간을 덜 차지 하는 장점을 가지고 있다.

```
Content-Type:Application/X_ControlInfo
Content-Transfer-Encoding:base64

TWV0aG9kPTIwMTEKUGFnZU51bT0yCnNlbnRlcl0
zJSVvcHVzNTAyQG9yZ2l1Lm5l
dCUIMTAwMjAwJSWxx7/Ax/wlJbDms7IguLa76r3DI
MfVxvexuCDIq7mutb8gNC0x
JSUKcmVjZW12ZXI9aGF3azcxM0BoYXdrLmNvbS5r
eXVuZ25hbS5hYy5rciUIMSU!
MjM0OTY0JSW067G4saS/qr3DILTevK2xuCDB+MO1
MrW/IDlwMy0xMiUlsejB37+1
JSUwMDA5MDUwMTMxMzUjTAWMDElJXh4eHh4
eHgtMDAxNm0JSU7Cg==

-----_NextPart_000_01BEE14.DAB71220--
```

[그림 5] Base64 인코딩된 Control Body

3.4 통제 바디(Control Body) 작성

Control Body 는 실제적인 인터넷 우편을 전송하는데 필요한 사용자가 작성한 송신자, 수신자등의 정보가 포함되어 있다. Control Body 에 들어가는 내용과 양식은 그림 4 과 같다.

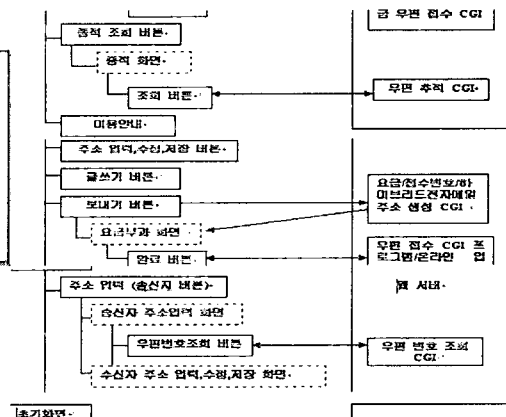
```
①page=2
②Method= 2011
③sender=2%opus502@orgio.net%100-100%권오형%경
남 마산시 합포구 홍문동 4-1%
④receiver=hawk713@hawk.com.kyungnam.ac.kr%1%23
4-964%대구광역시 달서구 진천 2 동 203-12 %김중
영%000905013135%0001%xxxxxxx-001734%;
```

[그림 4] Control Body

- ① 수신자가 작성한 메일의 총페이지 수의 정보가 들어가는 부분이다. (첨부파일+작성한 txt)
 - ② 인터넷 우편의 전송방법과 봉투양식에 대한 정보가 들어간다. (보통우편, 등기우편, 내용증명, 칼라인쇄, 배달증명, 봉합식)
2011 → 등기우편, 배달증명, 봉합식
 - ③ 송신자의 정보가 들어간다.
Type1=우편번호/이름/주소
Type2=E-mail 주소/우편번호/이름/주소
 - ④ 수신자에 대한 정보가 들어가는 부분이다.
receiver= 배달 우체국 전자 메일 주소% 우편을 처리하는 배달 우체국이 같은 수신자 수 %우편번호%주소%수신자이름%접수번호%일련번호%(등기번호)%; 다른 수신자...
- 이러한 양식으로 작성한 내용을 MIME type 로 변경한 결과의 마지막 부분에 아래와 같이 생성된 통제바디의 MIME body 를 추가하면된다.

3.5 웹 서버와의 통신

Front Agent에서 생성되는 화면과 버튼을 기반으로 하여 CGI 프로그램과 통신하는 방법은 아래 그림과 같다.



[그림 6] 웹 서버와의 통신

3.6 페이지수 구하기

현 시스템에서는 페이지수를 구하기 위한 pageNum.dll 파일을 제공하는데 이는 사용자가 입력한 txt 형식의 문서와 이미 작성된 아래아 한글이나 MS Word 파일에 대한 페이지수를 계산하는 역할을 한다. 사용자가 글쓰기 화면에서 입력한 내용에 대한 페이지 수와 라인수를 구하여 보여주는데 이는 애플릿에서 DLL 의 함수를 호출하기 위하여 JNI 를 이용하여 구현하였고 아래아 한글파일은 헤드 파일을 검색, MS Word 는 MS Word Automation 을 이용한 MS Word 파일의 페이지 수를 구하였다.

4. 결론 및 향후 연구 과제

본 논문에서 제시한 인터넷 우편시스템을 이용함으로써 기존의 E-mail 의 전송뿐만 아니라 사용자가 작성한 문서를 직접 여러 사람에게 보다 저렴한 비용으로 신속, 정확하게 전송할 수 있다. 향후 연구 과제로는 현재 코드 서명을 하기 위하여 테스트용 인증서를 사용하였으나 이를 인증기관에서 제공하는 인증서를 사용하여 코드 서명을 해야 한다. 또한 현재 내용증명 서비스가 지원 되지 않고 있는데 이를 위해서는 인증서와 서명키를 얻어 Control Body 에 인증서와 서명결과를 포함하여 MIME 를 작성 해야 하는데 이때 MIME 을 서명 하기 위한 알고리즘을 선택, 연구가 필요하다.

참고 문헌

- [1] <http://www.microsoft.com/java/>
- [2] <http://www.microsoft.com/java/sdk/>
- [3] <http://developer.netscape.com/docs/manuals/signedobj/>
- [4] <http://developer.netscape.com/software/signedobj/>
- [5] <http://www.javaland.co.kr>
- [6] <http://www.javapeople.net>
- [7] <http://java.frehosting.co.kr>
- [8] <http://www.javastudy.co.kr>
- [9] 오세만, 이양선, 김상훈, 고팡만 저, '자바입문', '생능출판사'
- [10] 윤경구 저, '지나와 함께하는 자바 2', '대림출판사'
- [11] Michael Morrison 외 19인 저, '자바 언리쉬드', '대림출판사'
- [12] 황부현 저, 'JAVA 프로그래밍', '정익사'
- [13] 이재광 저, '자바보안과 암호화', '한빛미디어'