

〈主 題〉

MHS 구현

공 상 환
(한국전자통신연구소 컴퓨터통신연구실)

■ 차

례 ■

I. 서 론	나. 사용자 처리기(UA) 블록
II. MHS 기능모델	다. 메세지 전송 처리기(MTA) 블록
III. EX400 시스템	라. 시스템 동작
가. 시스템 구조	IV. 결 론

I. 서 론

이제 컴퓨터를 활용하는 이용자들은 그들이 원하는 정보가 어디있는가에 관계없이, 아주 다양한 범주의 데이터들을 요구하고 또 다루고 싶어하며, 또한 상대방이 어디에 있던간에 다양한 통신 매체를 이용하여 정보를 교환하기를 원한다.

오랜동안 많은 이들의 사연을 담거나, 또는 문서형태로 여러가지 수송수단을 이용하여 전달되어 오던 편지라는 수단은 이제 컴퓨터와 통신망, 소프트웨어와 같은 정보통신 기술이 발달하게 됨에 따라 컴퓨터 통신망이라는 전자매체를 이용하므로써 간편하고, 신속한 정보의 전달수단이 되었다.

본 고에서는 컴퓨터 통신을 이용하여 사용자간의 정보를 교환해 주는 시스템인 메시지처리 시스템(MHS : Message Handling System)에 대한 기본개념과 CCITT X.400('88년)과 국내 기능표준인 MHS-88 (Version 2.0)을 위주로 개발한 EX400 시스템이라는 구현사례를 살펴보고자 한다.

MHS에 대한 연구는 국제적으로 CCITT / SG VII / WP4과 ISO / IEC JTC1 SC18 / WG4에서 표준화가 다루어 지고 있으며, 국내에서는 OSIA / TG-MHS와 TTA / MHS실무위원회 등을 통해 MHS 기능표준('88년도 Version 2.1)까지 완료한 상태에 있다.

MHS의 구현사례를 보면, 이미 외국에서는 미국, 일본, 프랑스, 영국 등 많은 국가에서 시스템을 개발하여 활용 중에 있으며, 국내에서도 데이콤에서 개발한 메일 400이 있고, KT 및 Kcom 등에서 외국 제품을 도입하여 서비스를 제공하거나 예정 중에 있다.

II. MHS 기능모델

MHS란 사용자가 컴퓨터를 이용하여 보내고 싶은 정보를 축적 전송 방식으로 원거리에 있는 수신자에게 전달해 주는 통신 서비스 기능이다.

MHS에서 다루는 메시지는 크게 세 가지 유형으로 구분할 수 있는데, 먼저 사용자의 통상적인 서신으로 봉투와 내용으로 구성되는 메시지(Message)와 특정 메시지의 전송가능여부 판단을 위해 이용되고 봉투만으로 구성되는 프로브(Probe) 그리고 메시지의 배달확인이나 수신확인에 이용되는 리포트(Report)를 들 수 있다.

중요한 MHS의 구성요소로는 그림 1의 '88년도 MHS 기능모델에서 보는 바와 같이 사용자 처리기(UA : User Agent)와 메시지 전송 처리(MTA : Message Transfer Agent), 메시지 저장기(MS : Message Store), 접근 장치(AU : Access Unit) 그리고 물리배달 접근 장치(PDAU : Physical Delivery Access Unit)들이

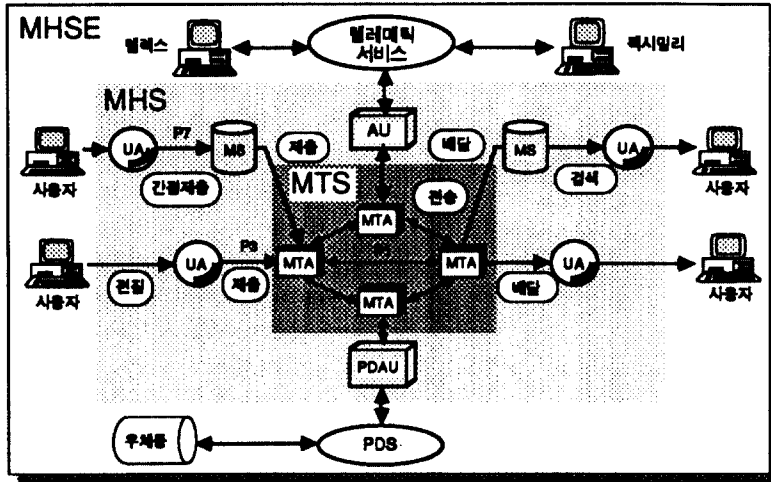


그림 1. MHS 기능모델

포함된다.

UA는 사용자의 메시지를 편집하는 기능과 MTA로 메시지를 제출하는 기능을 담당하며, MTA는 UA로부터 받은 메시지를 해당 수신자가 있는 시스템으로 메시지를 전송하는 역할을 수행한다. 특정 시스템에서 UA가 MTA 없이 단독적으로 존재하게 되는 경우 원격 UA라고 부르며, 이 경우 UA는 송신할 메시지를 인접한 시스템의 MS로 제출하여 메시지를 저장하게 되고, 이 MS로 하여금 MTA로 메시지를 전달하게 한다. AU는 팩시밀리나 텔렉스 등과의 연동을 위한 추가적인 인터페이스 장치이며, PDAU를 이용할 경우 실제 편지와 같이 인쇄, 포장된 우편물을 전달할 수 있다.

III. EX400 시스템

EX400 시스템의 1차 프로토타입을 위한 구현범위는 MHS의 기능모델에서 소개된 여러가지 기능계층 중 UA와 MTA를 대상으로 하였으며, 특히 EX400의 개발에 참여한 팀은 국내 기능규격의 작성에도 참여하여 국내 기능표준에 적합한 시스템 구현이 비교적 용이하게 이루어 졌다고 볼 수 있다.

구현환경으로는 AT&T의 System V Release 3.2를 운영체제로 하여 범용성을 높이고자 하였고, Motorola Delta-series 시스템 3대를 testbed로 활용하였다. MHS 하부의 OSI 프로토콜을 위하여는 ISODE(ISO Devel-

opment Environment 6.0)을 이용하였다.

가. 시스템 구조

EX400 시스템의 소프트웨어 구조는 그림 2에서 보는 바와 같이 UA 블록과 MTA 블록으로 구성되며, MTA 이하에는 ISODE가 활용되므로 이에 대한 인터페이스가 있다.

EX400에서 사용되는 프로토콜 구조를 살펴보면, UA간에는 개인간 메시지 통신 프로토콜인 P1, MTA간에는 메시지 전송 프로토콜인 P2, 그리고 UA와 MTA, MS와 MTA사이에는 MTS(Message Transfer System) 접근 프로토콜인 P3, 그리고 UA와 MS간의 프로토콜로 P7이 이용된다.

나. 사용자 처리기(UA) 블록

UA 블록의 구성은 앞서 살펴 본 그림 2의 소프트웨어의 구조에서 보는 바와 같이 사용자 인터페이스, 명령어 해석기, 메시지 데이터베이스, UAL(User Application Layer) / P3 인터페이스로 구분된다.

각 모듈에 대한 설명에 앞서, UA 블록이 EX400 시스템 사용자를 위해 제공하는 명령어의 유형들을 살펴보면 다음과 같다.

- 메시지 작성 및 수정 명령어
- 메시지 확인 및 화면 출력, 프린터 출력 명령어
- 메시지 처리 및 화일관리 명령어

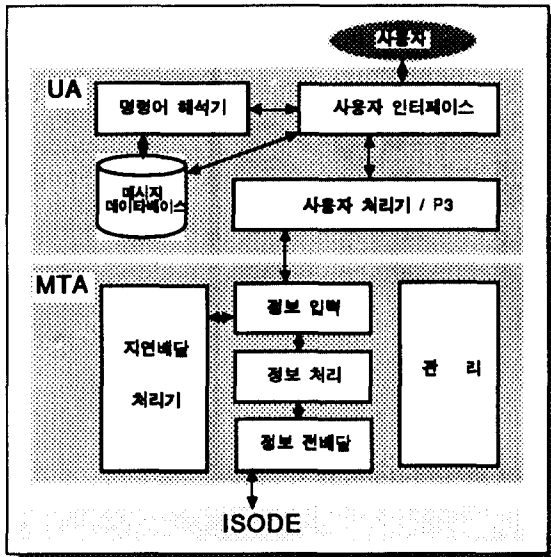


그림 2. EX400 소프트웨어 구조

- 메시지 송수신 처리 및 전송 명령어
- 시스템 기동 및 종료 명령어
- 기타 HELP 명령어 등

(1)사용자 인터페이스

이 모듈은 사용자가 입력하는 명령어를 읽어 들이고, 또한 시스템에서 출력되는 데이터를 사용자에게 표시해 주는 역할을 담당한다.

명령어 입력방식은 명령어에 필요한 데이터의 양이 적을 경우 command 모드로 입력을 처리하나, 데이터의 양이 많을 경우는 메뉴 방식을 이용하게 된다. 메뉴방식의 경우는 UNIX에서 제공되는 curses 라이브러리를 이용하였다.

(2)명령어 해석기

명령어 해석기는 사용자 인터페이스 모듈에서 넘어온 입력 명령어, 파라미터 들을 번역하여 해당되는 처리 루틴이 호출될 수 있도록 한다.

입력 명령어를 해석하는 방식을 보면, 명령어의 약어 부분에 대하여는 컴파일 형태로 해석하고 명령어의 나머지 부분은 인터프리터 형태로 해석하는 두 가지 방식을 갖는 데, 이는 명령어의 약어 처리를 가능하게 함으로써 다소간의 처리 속도와 메모리 사용면에서 효율을 얻게 한다.

(3)메시지 데이터베이스

UA에서 MS의 기능인 메시지 저장 및 검색 기능을 수행하도록 하여 UA와 MS를 동일 시스템내에 구현하였다. 메시지 데이터베이스는 UNIX 파일 시스템을 이용하여 room안에 있는 메시지 수와 상태, 소유자 및 생성 날짜에 대한 정보를 갖는 map 파일과, 메시지 헤더를 갖고 있는 헤더 파일과 메시지 본체 부분을 갖는 본체 파일로 구성하였다.

map 파일은 메시지의 검색 및 관리 기능을 수행하는데 필요하며, profile 파일은 사용자 개인의 주소 등에 관한 사항들과 메시지 수신시 자동 응답, 자동 폐기, 자동 재전송 및 IPN(Interpersonal Notification) 메시지 작성에 필요한 사항들을 기록하는 파일이다.

log 파일은 모든 메시지의 송/수신 사항들을 기록하는데 사용하는 파일이고, inroom 디렉토리는 작성한 기안 메시지와 송신한 메시지 및 수신한 메시지를 저장하는 기본 room이다.

메시지 데이터베이스의 메시지에 대한 접근은 우선 map 파일에서 메시지에 대한 기초적인 사항들에 접근하고 map 파일 내용 속에 있는 헤더 파일의 이름에 접근하여 헤더파일을 분석하고 헤더 파일의 한 내용인 본체 부분 파일 이름을 얻어 본체 부분에 접근함으로써 이루어 진다.

(4)UAL/ P3 인터페이스

ti +2 UA와 MTA는 UNIX 시스템 V에서 제공하는 IPC를 이용하여 통신한다. UAL에서 작성한 P2 메시지와 send 명령 입력시 사용자로부터 원하는 선택사항들을 받아 P3 제출봉투를 작성하고 P3 제출 루틴을 불러 MTA로 메시지를 제출한다. 선택사항이 없이 즉시 전송하길 원하는 경우에는 P3 제출 인수의 기본값을 할당하여 MTA에 제출한다.

다. 메시지 전송 처리기(MTA) 블록

MTA는 MHS의 기능 객체중의 하나로서 UA로 부터 P3 제출 형태로, 또는 이웃 MTA에서 P1 전달 형태로 데이터를 받으면 이 데이터를 해석하여 해당 정보를 P3 배달 형태로 수신자에게 배달하거나, 아니면 P1 전달 형태로 이웃 MTA에게 전달한다.

MTA 블록에서 제공하는 서비스들은 다음과 같다.

- 내용 유형 표시(Content Type Indication)
- 배달 시각 소인 표시(Delivery Time Stamp Indication)

- 메시지 식별(Message Identification)
- 비 배달 통지(Non-delivery Notification)
- 제출 시각 소인 표시(Submission Time Stamp Indication)
- 사용자 배달 정보 등록(User / UA Capabilities Registration)
- 대리 수신자 허용(Alternate Recipient Allowed)
- 대리 수신자 지정(Alternate Recipient Assignment)
- 지연 배달(Deferred Delivery)
- 지연 배달 취소(Deferred Delivery Cancellation)
- 배달 통지(Delivery Notification)
- 타 수신자 공개(Disclosure of Other Recipients)
- 배포 목록 확장 내력 표시(DL Expansion History Indication)
- 동보 배달(Multi Destination Delivery)
- 송신자 요구 대리 수신자(originator Requested Alternate Recipient)
- 비 배달 통지 금지(Prevention of Non-delivery Notification)
- 검증(Probe)
- 송신자에 의한 수신 전환 불허용(Redirection Disallowed by Originator)
- 도착 메시지의 수신 전환(Redirection of Incoming Messages)
- 내용 반송(Return of Content)

(1)정보 입력 모듈

입력정보에는 메시지, 프로브, 리포트가 있다. 정보를 입력받으면 각각의 정보의 형태에 따라 P3 메시지 / 프로브 디코딩, P1 정보 BER(Basic Encoding Rules) 디코딩을 수행한다. UA로부터 받은 메시지에 대해 지연 배달 요청이 있으면 지연 배달 처리기를 수행하고 MTA는 종료된다.

P3 형태의 제출 정보로는 메시지와 프로브가 있다. UA에서는 메시지와 프로브의 제출시 내부 인코딩을 하여 화일에 기록한 형태로 MTA에 제출하므로 MTA의 정보 입력 모듈에서는 해당 정보에 대한 내부 디코딩을 수행하여 PDU의 각 요소들을 내부 구별 가능 형태로 작성한다. 메시지가 제출될때에는 송신자가 지연 배달을 요청할때가 있다. 정보입력 모듈에서는 P3 제출 메시지의 지연배달 요소를 조사하여 지연배달 요청이 있는지를 검사한다. 지연배달 요청이 있는 경우 지연 배달 처리기라는 디몬 프로세스가 관리하

는 지연 배달 목록에 등록시킨후 MTA를 종료한다. 이때 지연 배달 처리 디몬이 존재하지 않으면 이를 실행시킨다. 활동 중이면 새로운 지연 배달 요청이 있음을 알려준다. UA로부터 메시지 / 프로브가 제출된 후 MTA는 메시지 큐 IPC를 이용하여 제출 식별자, 제출 시간, 내용 식별자등을 UA에 보내준다.

이웃 MTA에서 전달되어온 P1 정보의 입력인 경우, 전달되어온 정보는 BER에 따라 인코딩되어 있으므로 이를 디코딩 해야 한다. P1 정보의 전달 메시지, 전달 프로브, 전달 리포트에 대해 해당 디코딩을 수행한다.

송신자에 의해 지연 배달 요청이 있었던 메시지에 대해서 배달 시간이 되면, 지연 배달 처리기는 MTA를 수행시켜 UA에서 제출된 메시지처럼 처리한다.

(2)정보 처리 모듈

정보 입력 모듈로 부터 정보를 받아 처리하는 데, 이때 들어 온 메시지 / 프로브에 수신자가 없는 경우 수신자가 없음을 알리는 리포트를 생성한다. 정보 입력 모듈에서 오류가 있었으면 전 수신자에 대해 오류가 있음을 지적한다. 정보 입력 모듈로 부터 받은 정보가 들어오면 그것이 메시지 / 프로브인지 리포트인지에 따라 정보를 처리한다.

메시지 / 프로브의 경우, 들어 온 메시지 / 프로브에 대해 추적 정보를 얻고 각 수신자에 대해 수신자 처리 유형을 결정하며 수신 전환, 분배 리스트의 확장을 수행한다. 다른 MTA로 전달되는 수신자에 대해서는 라우팅 테이블을 조사하여, 어느 MTA인지를 결정하여 같은 MTA끼리 수신자를 묶어 정보 전배달 모듈에 보내준다.

리포트 처리의 경우는 정보 제출에 따라 처음 리포트를 생성하거나 전달된 리포트를 처리한다. 제출된 정보에 대해 세가지 유형의 리포트를 요구할 수 있는데, 전혀 리포트를 생성하지 않는 경우, 오류가 있는 경우에만 리포트를 생성하는 경우, 어떤 경우라도 리포트를 생성하는 경우이다. 생성된 리포트를 받는 수신자는 원래 메시지 / 프로브의 송신자가 된다. 생성된 리포트나 전달되어 온 리포트에 대해 다른 MTA로 전달할 것인지, 혹은 그 MTA의 수신자에게 배달할 것인지를 결정하여 정보 전배달 모듈로 보낸다.

(3)정보 전배달 모듈

정보 처리 모듈로 부터 받은 정보에 대해 배달, 또

는 다른 MTA로 전달을 수행한다. 정보 처리 모듈로부터 수신자 처리 유형이 배달로 확인된 수신자에 대해 메시지 / 리포트를 그 수신자에게로 배달한다. 이때 각 정보에 대해 P3 배달 봉투를 만들어 내부 인코딩에 의해 만든 헤더 화일과 본체 화일을 각 수신자의 디렉토리 밑에 놓고 수신자의 터미널에 정보가 왔음을 알려 주기 위해 경보를 울린다. 프로브의 경우는 배달로 확인된 수신자에 대해 실제 배달을 수행하는 게 아니라 배달 가능 여부를 타진하여 그 결과를 리포트로 알려준다.

또한 정보전달의 경우는 정보 처리 모듈로부터 수신자 처리 유형이 전달로 확인된 수신자에 대해 정보를 그 MTA로 전달한다. 이때 각 정보에 대해 P1 전달 봉투를 작성한후 BER 인코딩하여 ISODE의 RTSE, ACSE등의 서비스를 이용해서 다른 MTA로 정보를 전달한다.

(4)관리 모듈

관리 모듈은 정보전달상의 각종 제약 사항을 데이터베이스에 수록하는 등록 서비스를 중심으로 구현하였다. 사용자는 배달에 대한 제반 사항들 즉, 배달 메시지의 길이, 배달 부호화 정보 유형, 배달 내용 유형등에 대해서나 또는 수신자 지정 대리 수신자등의 항목에 대해 변경 요구를 행할 수 있다.

(5)지연배달 처리기(DDP : Deferred Delivery Processor)

EX400에서는 지연배달 요청을 송신 MTA에서만 처리한다. DDP는 송신 MTA에서의 신호 도착시 이

를 취급하는 기능, 지연배달 시간 도래시의 처리기능, 지연배달 요청 화일을 갱신하는 기능, 그리고 지연배달 취소요청을 취급하는 기능등을 수행한다.

(6)MTA와 ISODE와의 인터페이스

ISODE는 OSI 계층의 상위 계층의 서비스를 제공한다. 즉, 응용 계층의 ROSE, RTSE, ACSE 기능과 ASN.1 컴파일러, 표현 계층, 세션 계층의 기능, 그리고 트랜스포트 계층의 기능을 제공한다.

라. 시스템 동작

그림 3에서 보는 바와 같이, EX400 시스템의 동작은 사용자가 명령어를 이용하여 사용자 인터페이스 기능 및 UA 기능을 동시에 수행하는 EX400이라는 프로세스를 기동시키므로써 수행된다. EX400은 사용자가 메시지를 편집하는 과정을 지원하고, 최종 편집된 메시지의 송신요청을 받게 되면, 전송을 위하여 MTA 프로세스를 동작시키게 된다. 즉, EX400 프로세스와 MTA 프로세스 사이에는 1 : 1 대응관계를 형성하게 되는데, 이 방식은 하나의 MTA 프로세스가 여러 사용자의 메시지를 동시에 처리하는 방식보다 성능면에서 양호하기 때문이다.

EX400 시스템이 사용자 명령어를 받기 위해서는 사전에 시스템 기동과정이 요구되는데, 이 과정에서 기동되어 계속 상주하는 프로세스로는 원격지의 다른 시스템으로부터 오는 메시지를 기다리고, 메시지가 들어 오면 이 메시지를 사용자에게 전달하도록 하기위한 TSAPD(Transport Service Access Point Daemon)이 있다.

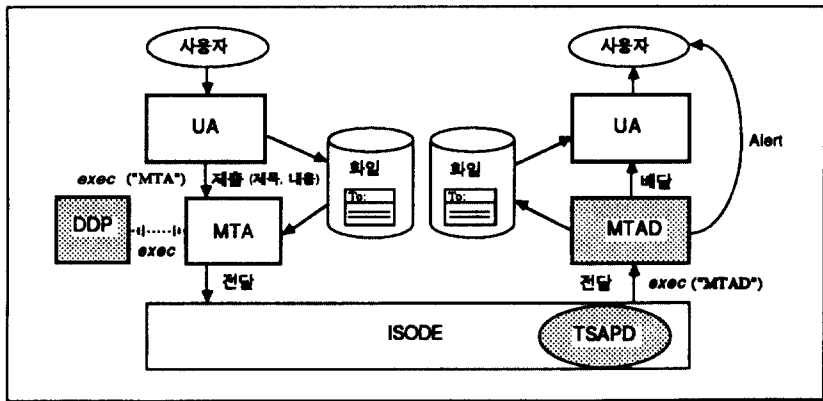


그림 3. EX400 시스템 동작구조

지연배달 처리를 담당하는 DDP 프로세스는 사용자의 지연배달 요청이 지연배달 테이블에 등록되면, 등록된 요청을 처리하기 위해 동작되며, 한 순간에 오직 하나의 DDP 프로세스가 처리하게 되므로 새로운 지연배달 요청을 추가할 때 DDP 프로세스가 이미 동작 중이면 그 요청은 테이블에만 추가된다.

IV. 결 론

EX400 시스템의 기능확장을 위하여 그림 4에서 보는 바와 같이 MS 및 RUA기능이 추가로 구현 중에 있으며, CCITT X.500인 디렉토리 시스템(DIR)의 개발이 동시에 추진중에 있고, 금년말에 가서는 MHS와 DIR간의 통합이 가능하리라고 본다.

MHS를 위한 MS 기능의 추가적인 고려는 PC 또는 workstation 규모의 시스템에서 UA 기능만을 보유하게 되는 경우에 이를 수용하기 위해 RUA 기능과 함께 구현이 추진되고 있다.

또는 금년도의 주 연구내용인 DIR은 비단 MHS만을 위한 시스템이라기 보다는 FTAM이나 network management 등의 분야에서 많은 활용이 예상되고 있다.

EX400의 활용은 단순히 전자 메일 시스템의 기능으로 뿐만 아니라 현재 국내에서 활발히 연구 중인 EDI의 엔진으로 또는 현재 ETRI의 정보통신 표준센터에서 추진 중인 MHS 프로토콜 적합성 시험을 위한 참조 시스템으로도 이용될 수 있다.

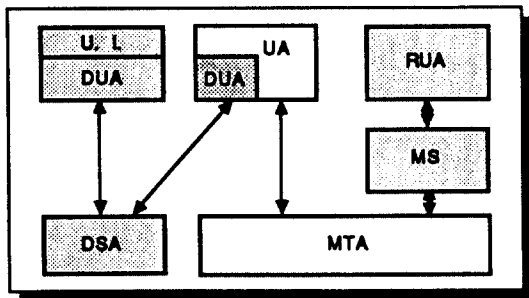


그림 4. EX400 시스템 확장

참 고 문 헌

1. "CCITT Recommendation X.400-X.430," 1988.
2. Marshall T. Rose, "The ISO Development Environment : User' Manual," 1990.5.
3. 박승민 외 2, "메시지 처리 시스템," 정보과학회 정보통신기술 8호, 1990.6.
4. 한우용 외 4, "EX400에서의 IPMS 구현," 동계 컴퓨터통신 Workshop, 1992.1.
5. 성홍석 외 4, "EX400에서의 MTA 구현," 동계 컴퓨터통신 Workshop, 1992.1.
6. 한기천 외 6, "CCITT X.400(MHS)의 구현에 관한 연구," 한국정보과학회 봄 학술발표논문집 Vol.19, No.1, 1992.4.



궁 상 환

- 1977년 숭실대학교 전자계산학과 학사
- 1983년 고려대학교 전자정보처리학과 석사
- 1977년~1981년 제2군수지원사령부 군수관리 자동화시스템 개발
- 1981년 4월~현재 한국전자통신 연구소 컴퓨터통신 연구실장