

センタ型電子メールシステムELMSと MHSとの相互接続

加藤聡彦 鈴木健二

国際電信電話株式会社 研究所

近年CCITTでは、公衆網を介したメッセージ通信処理システム(MHS)をX.400シリーズとして勧告化しており、各国でその構築が進められている。このような、公衆網におけるメッセージ通信サービスの進展によって、これまでに個別に作成されてきたセンタ型のメールシステムに公衆網のMHSインタフェースをもたせることにより、MHSを介した相互接続が可能となる。本稿では、筆者等が開発したVAX11/780上で動作するMHS用ソフトウェア・パッケージの構成と、先に作成済みのセンタ型電子メールシステムELMSからMHSへのアクセスを実現した結果について報告する。

Interconnection of Center Type Electronic Mail System ELMS and MHS

Toshihiko KATO Kenji SUZUKI

KDD R & D Labs. 2-1-23, Nakameguro, Meguro-ku, Tokyo, 153

CCITT has standardized Message Handling Systems (MHS) in association with public data networks as Recommendations X.400 to X.430 and some messaging systems have been implemented based on MHS. Consequently it becomes possible that the center type mail systems which are developed independently are connected with each other by implementing MHS interface in each mail system. This paper describes the configuration of the MHS software packages which the authors have developed using VAX 11 / 780 , and the results of the interconnection of the center type electronic mail system ELMS and MHS.

1. はじめに

近年、CCITTでは、公衆網を介したメッセージ通信処理システム(MHS)をX.400シリーズとして勧告化しており^[1]、各国でその構築が進められている。このような、公衆網におけるメッセージ通信サービスの進展によって、これまでに個別に作成されてきたメッセージ・センタに公衆網のMHSインタフェースをもたせることにより、MHSを介して相互接続することが可能となる。個別のメッセージ・センタから公衆網のMHSへのアクセスを実現するためにはゲートウェイを作成するのが一般的であるが、ゲートウェイの汎用化を図るためには、そこで実現するMHS用のソフトウェアをできるだけパッケージ化することが望ましい。筆者等はこれまで、VAX11/780上でOSIトランスポート、セッション・プロトコル用ソフトウェアを開発し、実用化を図ってきた^[2,3]が、この度、これらOSIソフトウェアとインタフェースして動作するMHS用ソフトウェア・パッケージ^[4]を作成し、先に作成済みのセンタ型の電子メールシステム(ELMS)^[5]からMHSへのアクセスを可能としたので、以下にその概要を報告する。

2 ELMSとMHSの相互接続の基本設計方針

個別のセンタ型のメールシステムでは、図1に示すように、様々な種類の加入者端末が、公衆網を介

して無手順やX.25等の手順でアクセスし、センタに加入しているユーザの間でメッセージをやりとりできる。さらにMHSインタフェースを実現することにより、X.400シリーズに従った手順で他のメッセージ・センタの加入者ともメッセージをやりとりできるようになる。

そこで個別のメールシステムとMHSの相互接続方式の検討を目的として、既存のELMSとMHSの相互接続を行うために、以下の方針を立てた。

(1) MHS用ソフトウェア・パッケージの実装

作成するMHSインタフェースプログラムは、メッセージセンタ及びメッセージ通信ゲートウェイの構築等に汎用的に使用できるよう、以下の方針でパッケージ化する。

① MHSの提供するメッセージ転送(MT)サービス/個人間メッセージング(IPM)サービスのサービス要素をできるだけ多く実装する。

② 他の多くのMHSシステムと接続が可能のようにO/Rネームは、X.400の定めるフォーム1の3つのバリエーションとフォーム2の全てをサポート可能とし、O/Rネームの各要素の長さも大きくとる。

③ MTサービスを提供するMTA(Message Transfer Agent)は1つのプロセス(以下P1プロセスと呼ぶ)で実現し、IPMサービスを提供するUA(User Agent)については、UAの内で共通的な機能を提供するプロセス(以下P2プロセスと呼ぶ)とユーザ毎の処理を

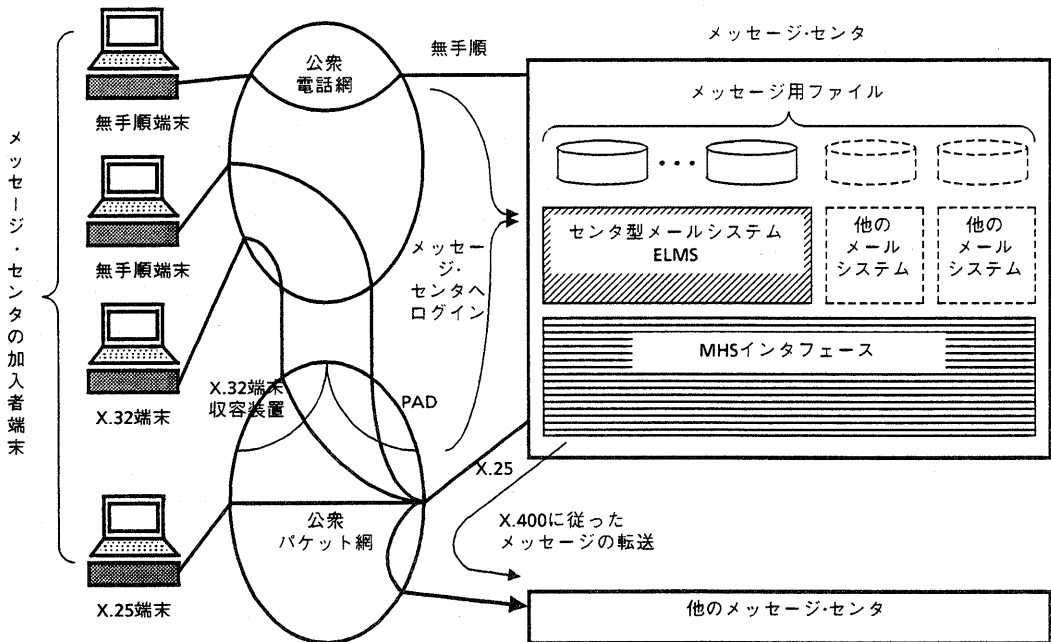


図1 メッセージ・センタの接続形態

行うユーザ対応のプロセスで実現する。P1プロセスとP2プロセス間のインタフェースは勧告X.411の定めるサービスプリミティブに対応させる。またP2プロセスとユーザプロセスの間には新たにインタフェースを定義する。

④ MPDUとUAPDUの処理はそれぞれMTAとUAに対応するプロセスで行う。また各プロセスではPDUに対応して構造化されたデータ構造を用い、X.409に従ったPDUの作成/解析の処理の負荷を軽くする。

⑤ 同時に複数の異なるユーザプロセスがアクセスできることを前提として設計し、当面はELMSとMHSとの相互接続のみを実現する。

⑥ 移植性を考慮してC言語により実装する。

(2) 汎用的な相互接続方式の検討

ELMSとMHSの相互接続では、ELMS以外の個別メールシステムに一般化できるように、次の方針を用いる。

① MHSインタフェースプログラムは、上述の汎用パッケージを用い、ELMSとの変換機能は基本的にはELMS側に持たせる。

② ELMSの機能変更は最小限とする。

以下、作成したMHSパッケージの機能と、ELMSとMHSの相互接続の実現法について述べる。

3. MHSパッケージの機能と構成

本MHSパッケージは上述の方針に基づき、図2に示すようなプロセス構成で実現される。以下にP1及

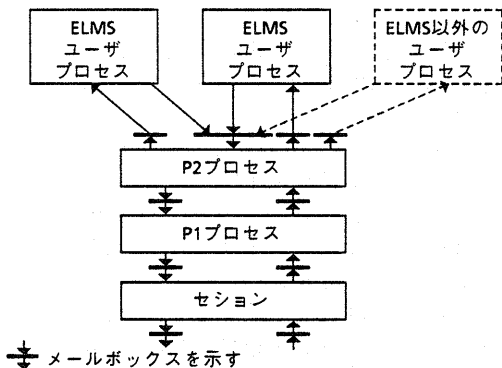


図2 MHSパッケージのプロセス構成

びP2プロセスを実現する各パッケージの機能について述べる。

3.1 P1パッケージの機能

本パッケージはMTAの機能のうち、変換以外の全てのMTサービスをサポートしており、図3に示すように5つの機能モジュールと3つの管理ファイルから構成される。

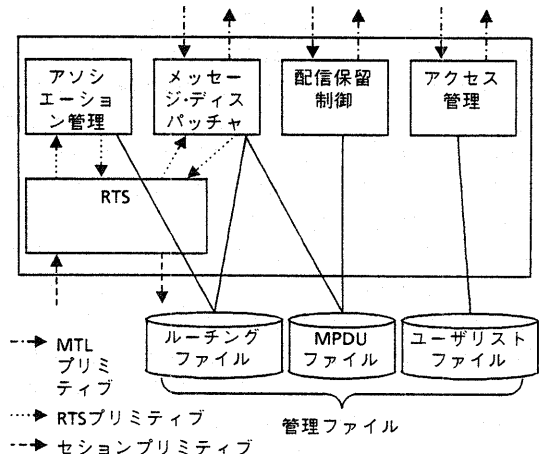


図3 P1パッケージの構成

(1) 機能モジュールの動作の概要

① アソシエーション管理モジュール

ここでは、通信を行うMTAとの間のアソシエーションの確立、解放、TURNの管理を行う。アソシエーションの確立では、システムが立ち上がっている間は定期的な張るか、転送すべきMPDUが存在する場合にのみ動的に張るかを相手MTA毎に指定できる。また動的なアソシエーションの解放は、転送すべきMPDUがなくなった後一定時間後に行われ、この時間は相手MTA毎に指定できる。アソシエーションの確立の相手のADMD名、定期的か動的かの指定等はルーチングファイルを参照する。

② アクセス管理モジュール

本モジュールでは、UAとMTA間のアクセスの確立と解放、及びパスワードの変更やユーザの管理情報の登録等を行う。

P2プロセスからのLOGON要求に対して、ユーザリストファイルから加入者ユーザかどうかをチェックし、正当なユーザの場合はLOGON中のユーザとして登録する。また加入者ユーザに対するMPDUを受信した場合、ユーザがLOGON中でない時は、LOGON指示を発行してLOGON応答を待つ。その後LOGON応答を受信した段階でLOGON中のユーザとして登録する。

③ メッセージ・ディスパッチャ

ここでは、メッセージの発信、配信、中継、及び配信(不能)通知の送出等を行う。

P2プロセスからのSUBMIT要求に対して、まず受取人の数やO/Rネーム等のそのメッセージに関する情報をユーザリストファイルに登録する。次に受取人のO/Rネームとルーチングファイルからルートを決め、ルート毎にMPDUを作成して、MPDUファ

イルに一旦書き込みSUBMIT確認を通知する。P1プログラムは、定期的にMPDUファイルをサーチし、蓄積されているMPDUをRTSの機能を用いて転送する。このように、一旦ファイルを介してMPDUを送信するのは、遅延配信や、プライオリティの高いMPDUの優先的な転送に対処するためである。遅延配信に対してはそのキャンセルも可能とした。また、転送するためのアソシエーションは、使用可能なものから、会話モードがモノログのもの、TWAで送信権を所有するもの、TWAで送信権を所有しないものの順で選択する。送信権を所有しないアソシエーションを選択した場合は、TURN-PLEASE要求を送信する。

また、RTSにより受信したMPDUに対して、まず構文チェックを行い、それが自分宛でない場合は中継処理を行い、自分宛の場合はP2プロセスへの配信を行う。配信処理ではまずMPDUファイルに書き込み、P1プログラムがMPDUファイルをサーチした場合にユーザとのアクセスが確立されているならば、DELIVER指示でメッセージを配信する。さらに本システムではDELIVER応答を新たに導入しており、この情報により配信通知MPDUを作成して送信する。受信したMPDUを一旦MPDUファイルに蓄積するのは、配信保留を行うためと、LOGON応答を待つためである。

④ 配信保留制御モジュール

本モジュールでは、CONTROL要求によるユーザの制限の一時的な変更を行う。P2プロセスからのCONTROL要求に対して、ユーザリストファイルに保持された、メッセージが配信可能かどうか/配信可能なメッセージの最大長等の制限を変更する。一時的な変更により制限が解除された場合は、メッセージ・ディスパッチャがメッセージを配信する。

⑤ RTS(Reliable Transfer Server)

RTSは、アソシエーション管理モジュールやメッセージ・ディスパッチャの要求により、セッション・サービスを利用して、実際のアソシエーションの確立、解放やメッセージの転送を行う。また、セッション・コネクションの異常時には、コネクションの再確立やアクティビティの再開等のリカバリ機能も有する。具体的には、転送中に相手から例外報告に対してはアクティビティを中断し、アボートが通知された場合はコネクションを再確立し、受信が確認されていないデータからアクティビティを再開する。TRANSFER要求に指定された転送時間内にメッセージが送れない場合は、そのアクティビ

ティを廃棄し、メッセージ全体の転送が失敗したことを通知する。

(2) 管理ファイル

次にP1パッケージが使用する3つの管理ファイルについて述べる。これらのファイルは、相対編成ファイルとして実現し、レコード番号をキーとしてアクセスを行う。これらのアクセスにはVMSのRMS (Record Management Service)を使用する。

① ユーザリストファイル

このファイルは、MTAに加入しているUAの情報を管理しており、ユーザ情報部とメッセージ情報部から構成される。

ユーザ情報部には次のようなUAを特徴付ける情報が含まれる。

- ・UAのO/Rネームとパスワード
 - ・LOGON中であるか否か等のUAの状態
 - ・UAが受信できるメッセージの最大長及び符号化情報タイプ
 - ・UAに許されたMTサービスエレメントのリスト
- メッセージ情報部にはUAが発信/配信するメッセージに関する情報が含まれる。

発信したメッセージに関して

- ・SUBMIT イベントID、UA Contents ID、MPDU ID
- ・受取人の数とそのO/Rネーム
- ・MPDUファイルの中のメッセージの格納場所

配信できるメッセージに関して

- ・配信できるメッセージの数と長さ、及びMPDUファイルの中の格納場所
- ・報告できる配信または配信不能通知の数と長さ、及びMPDUファイルの中の格納場所

② MPDUファイル

このファイルは、発信/配信/中継メッセージ等を一時的に保管するために使用される。ファイルは、他のMTAに転送すべきメッセージを保管する転送メッセージ領域、自システムのUAに配信するメッセージを保管する配信メッセージ領域、ユーザの制限ですぐに配信できないメッセージを保管する配信保留メッセージ領域、遅延配信日時が経過していないメッセージを保管する遅延配信メッセージ領域に分けられる。

③ ルーティングファイル

このファイルは、アソシエーション管理モジュールが、アソシエーション確立時に相手のSSAPアドレスを決定する場合や、メッセージディスパッチャがメッセージの転送ルート決定する場合に使用される。ファイルは相手MTAの国名、

ADMD名、PRMD名からSSAPアドレスへマッピングし、対応するアソシエーションの属性も参照することができる構成になっている。

(3) MPDUの取扱い

P1パッケージでは、MPDUの内部処理を容易にするために、MPDUの抽象構文に対応する構造体を使用して送信MPDUのフォーマット、受信MPDUの解析を行っている。User MPDUに対応する構造体の一部を図4に示す。

```

UMPDUEnvelop ::= SET { MPDUId,
  [2] IMPLICIT SEQUENCE OF RecipientInfo }
MPDUId ::= [APPLICATION 4] IMPLICIT SEQUENCE {
  GlobalDomainId, IA5String }
RecipientInfo ::= SET { recipient ORName,
  [0] IMPLICIT ExtensionId,
  [1] IMPLICIT PerRecipientFlag,
  [2] IMPLICIT ExplicitConv OPTIONAL }
  
```

項目名	データ型
MPDU Id	struct MPDUId
RecipientInfo数	int
RecipientInfo先頭ポインタ	struct RecipientInfoへのポインタ
RecipientInfo最終ポインタ	struct RecipientInfoへのポインタ

項目名	データ型
GlobalDomainId	struct GlobalDomainId
IA5String長	short int
IA5String	charの配列(64バイト)

項目名	データ型
次RecipientInfoへのポインタ	struct RecipientInfoへのポインタ
Recipient O/R Name	struct ORName
ExtensionId	int
PerRecipientFlag	フィールド(1バイト)
ExplicitConv長	char
ExplicitConv	char

図4 User MPDUの処理用の構造体(一部)

3.2 P2パッケージ

本パッケージはその上位のユーザプロセスと共同で、UAの機能を実現する。ここではP2パッケージが提供する機能の概要を述べる。

(1) P2パッケージの機能

① ユーザプロセスとのインタフェース

ユーザプロセスは動的に生成/消滅するため、ユーザプロセスからP2プロセスへのコマンドとそれに対する応答はメールボックスを介してやり取りさせ、またP2プロセスからユーザプロセスへ通知される受信メッセージ等は、ユーザ毎に設けた受信メッセージファイルを介してやり取りさせる。

ユーザプロセスからP2プロセスへのコマンド/応答としては、MTLプリミティブのLOGON要求/確認、SUBMIT要求/確認、LOGOFF要求/確認に対応するインタフェースを定義した。またユーザプロセスとP2プロセスの間のメールボックスは、固定的に割り当てることはできないため、図5に示すように、P2プロセスはユーザプロセスからのプリミティブを受信するメールボックスを1つ用意し、またP2プロセスからユーザプロセスへプリミティブを送信するためのメールボックスはユーザプロセスに合わせて動的に生成/消滅させた。この時、メールボックス名は一意である必要があるため、ユーザプロセスがVMSにより付与されたプロセスIDを用いて決定するようにした。

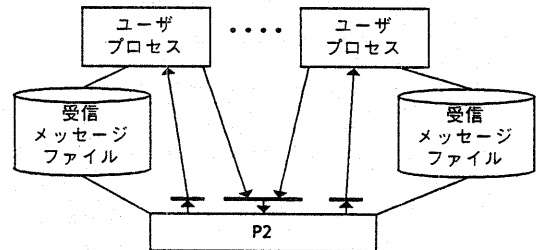


図5 P2プロセスとユーザプロセスの接続

② メッセージ送信処理

本システムではメッセージの送信において、UAPDUのフォーマットはユーザプロセスで行わせている。従って、P2プロセスはユーザプロセスからのメッセージ送信要求に対して、単にSUBMIT要求をP1プロセスに発行する処理を行うのみである。UAPDUのフォーマットをユーザプロセスに任せた理由は、本ソフトウェアをゲートウェイに使用した場合に変換処理を少なくするためである。

③ メッセージ受信処理

P2プロセスでは、P1プロセスからDELIVER指示で渡される受信UAPDUの解析を行い、ユーザプロセスが個々に所有する受信メッセージファイルに書き込む。また、P1プロセスからの配信(不能)通知もユーザプロセスに渡す必要があるため、受信メッセージファイルに書き込む。受信メッセージファイルの情報は、受信UAPDUのフォーマットのままでなく、ユーザプロセスの処理が容易なように構造化されている。このように、ユーザプロセスに渡すメッセージの形式が送信時と異なるのは、P2プロセスが受信UAPDUの構文チェックを必ず行う必要があるためである。

④ 処理の対象となるメッセージのボディ

本パッケージで扱うことのできるメッセージのボディはIA5テキストとそれがフォワードされたもののみである。

(2) 受信メッセージファイルの構成

受信メッセージファイルもP1パッケージが使用する管理ファイルと同様に相対編成ファイルとして実現される。このファイルはIM UAPDU用、SR UAPDU用、配信(不能)通知用の3種類の領域に分けられ、レコードを動的に確保/解放することにより、UAPDUに対応するデータ構造を実現している。IM UAPDUのヘッディングに関する情報を保持するための領域の一部を図5に示す。

```

IM-UAPDU := SEQUENCE { Heading, Body }
Heading := SET { IPMessageId,
  primaryRecipients [2] IMPLICIT SEQUENCE OF
  Recipient OPTIONAL }
IPMessageId ::= [APPLICATION 11] IMPLICIT SET {
  ORName OPTIONAL, PrintableString }
Recipient ::= SET { [0] IMPLICIT ORDescriptor,
  receiptRequest [1] IMPLICIT BIT STRING {
  receiptNotification (0) } DEFAULT {},
  replyRequest [2] IMPLICIT BOOLEAN DEFAULT FALSE }
  
```

IM-UAPDU	項目名	データ型
	レコードNo	unsigned
	IPMessageIdポインタ	unsigned
	PrimaryRecipient 数	short int
	PrimaryRecipient 先頭ポインタ	unsigned

IPMessage Id	項目名	データ型
	レコードNo	unsigned
	O/R Name長	short int
	O/R Name	struct ORName
	PrintableString長	short int
	PrintableString	charの配列(64バイト)

Recipient	項目名	データ型
	レコードNo	unsigned
	次レコードへのポインタ	unsigned
	O/R Name長	short int
	O/R Name	struct ORName
	receiptNotification	short int
	ReplyRequest	short int

図6 IM UAPDU用の領域(一部)

4. ELMSとMHSとの相互接続方式

4.1 ELMSの概要^[5]

ELMS (Electric Mail System)はセンタ型のメールシステムであり、表1に示すようなメールサービスを提供している。

ELMSでは、ユーザがメールの編集/送信等を行う場合に、各ユーザ端末からELMSプロセスが起動される。各ユーザには、受信メールと配達日指定までのメールを蓄えるELMSメールファイルが用意され

表1 ELMSのメールサービス

サービス	概要
基本サービス	メールの送信要求があれば、直ちに相手に送信する
配達日指定サービス	配達指定日まで、送信側に格納しておき、該当日に相手へ送信する
同報サービス	同時に複数の相手にメールを送信する
速達サービス	受信側での読み出しが最優先となるように指定する
書留サービス	受信側でメールを読み出したことを送信側に通知する
私信サービス	送信したメールが、相手のIDを用いない限り読み出せないようにする
コピー配布サービス	メール本体が送信完了後、メール本体のコピーを指定された相手に基本サービスで送信する
自動消去サービス	送受信処理終了後、不要となったメール本体を自動的に消去する
告知板サービス	Bulletin Board上に記入されたメッセージをユーザが読み出せる

る。メール送信時にはELMSシステムで一元的に管理されるELMSユーザリストファイルを参照して、送信相手の正当性等のチェックする。

ELMSではユーザを識別するために、ユーザ自身により登録されたユーザ名と、ELMSの管理者が一意に付加するユーザコードを用いている。またELMSメールファイル内のメールを一意に識別するためにメール番号を付与している。

4.2 MHSにおけるELMSの管理方式

ELMSをMHSで管理するために、次の方法を取った。

① ELMSは独自のユーザを持つメールシステムであり、さらに同一計算機内に他のメールシステムが起動されることも考えられる。そこでELMSをMHSにおいて位置付けるために、ELMSは1つのプライベート管理ドメイン(PRMD)を形成するものとする。

② ELMSのユーザのO/Rネームは国名、ADMD名、PRMD名、ドメイン定義属性リストを用いる。ドメイン定義属性リストには、ELMSが管理するユーザ名とユーザコードを指定する。

③ ELMSでは独自にユーザ管理/メールの保持を行っているため、MHSでは基本的には個々のELMSユーザの管理は行わない。すなわち、P1プロセスのユーザリストファイルでは個々のELMSユーザの管理は行わず、さらにP2プロセスとユーザプロセスの間の受信メッセージファイルも全てのELMSユーザに対して1つだけ用意する。但し、P1プロセスはELMS宛のメッセージを受信した場合、そのユーザが登録されているかを確認する必要がある

が、この確認にはELMSが管理している情報を使用する。

4.3 ELMSとMHSの相互変換の実現

ELMSとMHSの間の相互変換は、ELMSにサブルーチンとして組み込まれた変換プログラムによって行う。以下にその主な機能を示す(図7参照)。

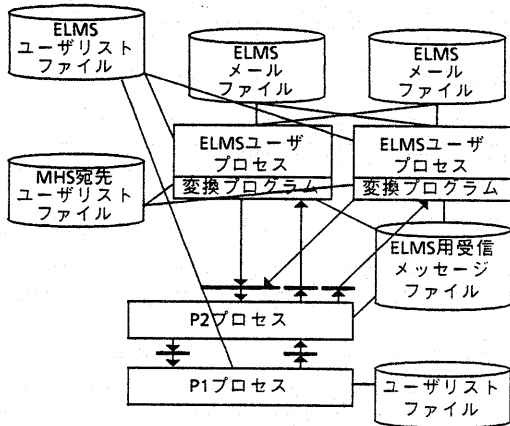


図7 ELMSとMHSの相互接続方式

(1) MHSに対するアクセス管理

ELMSをMHSで1ユーザとして管理しているため、複数のELMSユーザプロセスが起動された場合最初のプロセスからのみMHSに対するアクセス確立要求が発行され、また最後に消滅するプロセスからのみアクセス解放要求が出される必要がある。変換プログラムではこの機能をVAX/VMSの共用イメージによるデータの共用化を用いて実現した。

(2) 外部ユーザの名前の対応付け

ELMSユーザは独自の名前(以下ELMSネームと呼ぶ)を有しており、ELMSのユーザインタフェースによりメールを編集する場合は、外部ユーザに対してもELMSネームの形式に従って入力する必要がある。そこで外部ユーザのELMSネームとO/Rネームとの対応付けを、図7に示すMHS宛先ユーザリストファイルを用いて管理する。メッセージを送信する場合は変換プログラムが相手のELMSネームをキーとしてMHS宛先ユーザリストファイルからO/Rネームを検索する。また、受信時には逆に相手のO/RネームからELMSネームを選択し、ELMSユーザに表示している。

(3) メールの発信

ELMSユーザから自システム内の他のELMSユーザ宛のメッセージは、直接相手のELMSメールファイルに書き込まれる。また外部ユーザ宛のメッセージである場合は、変換プログラムがUAPDUに変換されP2プロセスに渡される。発信メッセージが自

システム宛か外部ユーザ宛かは、ELMSユーザリストファイルとMHS宛先ユーザリストファイルを順にサーチすることにより判別する。

また1つのメッセージの宛先にELMSユーザと外部ユーザが指定された場合は、ELMSユーザへの配信はELMSシステム内で行われ、外部ユーザ宛の発信処理のみがP1/P2プロセスで行われる。

(4) メールの配信

P1プロセスはPRMD名がELMSであるメッセージを受信した場合は、正当なユーザであるかを確認するため、図7に示すように直接ELMSユーザリストファイルを参照する。

ELMSユーザ宛のメッセージは全てELMS用受信メッセージファイルに蓄積され、変換プログラムは、ELMSユーザプロセス起動時やユーザから要求された時点等にこのファイルをサーチし、自分宛のメールを取り込む。

4.4 MHSサービスとELMSサービスの対応

MHSサービスとELMSサービスを対応させるために、ELMSにないMHSサービスについては、必要なもの以外は、メール発信時には使用せず受信時には基本的には無視するようにし、またELMSとMHSで重複したサービスはELMSの機能を用いるようにした。MHSとELMSのサービスの対応について代表的なものを表2に示す。

5. 結果と考察

MHSパッケージ及びELMSをVAX 11/780とpVAX上で動作させX.25網経由の通信実験を行い、正常に動作することを確認した。本プログラムの開発を通して以下の考察を得た。

(1) MHSパッケージの開発

① 本パッケージでは、P1/P2プログラムのにおいて、RTS等の機能毎のモジュール化、MPDU/UAPDUの作成/チェック等の関数化に留意して作成した。従って、他のMHSゲートウェイシステムの構築等に本パッケージを有効に使用できる。

② 本パッケージを用いて、KDDの提供するメールボックスサービスのMessaviaとの相互接続を行っており、本パッケージの動作の正当性は確認されていると考えられる。

③ 本パッケージが使用するPDUの各パラメータ長の制限は、実用上充分大きい。MPDU用の作業領域や受信メッセージファイルで示したように、SEQUENCE OFの要素は動的に処理領域を確保するため、メモリまたはファイルの容量の許す限り指定できる。またユーザリストファイル等で指定する

表2 MHSとELMSのサービス対応

MHS	ELMS	対応付けの方法
Access management	サービスなし	MHSとのアクセスの確立や解放のためのルーチンを組み込む
Delivery time stamp indication	基本サービス	ELMSメールの受信日時に対応づける。ただしフォーマットはYYMMDDHHMMとする
Message identification	サービスなし	イベントIDはP1プロセスが付加し変換プログラムで管理する。またUA Contents IDはELMSのメール番号を使用する
Non-delivery notification	基本サービス	DELIVER指示は変換プログラムが処理しその後はELMSの手順に従う
Original encoded information types	サービスなし	発信時にはIA5Textを指定し、DELIVER指示で通知される情報は無視する
Registered encoded information types	サービスなし	P1プロセスの管理するユーザリストファイルに登録する時、IA5TextとUndefinedを登録しておく
IP-message identification	基本サービス	ELMSのメール番号と対応付ける
Deferred delivery	配達日指定サービス	ELMSのサービスを利用する。そのためMHSのサービスは利用しない
Deferred delivery cancellation	自動消去サービス	同上
Delivery notification	サービスなし	ELMSが外部ユーザに発信したメールは、配信通知がくるまで保持する必要がある
Grade of delivery selection	速達サービス	PriorityとしてはURGENTとNORMALのみを使用する
Multi-destination delivery	同報サービス/ コピー配布サービス	外部ユーザ宛の場合は、メールのTOとCCフィールドの区別なくP1プロセスがコピーを作成する
Primary and copy recipients Indication	基本サービス	Primary RecipientはメールのTOフィールドに、Copy RecipientはCCフィールドにそれぞれ対応付ける
Importance indication	サービスなし	送信時は指定しない。受信時は無視する
Sensitivity indication	私信サービス	Sensitivityの内PRIVATEに対応させる
Blind copy recipient indication	サービスなし	送信時は指定しない。受信時には無視するためBlind copy recipientも互いに受取人であることを知らされない
Receipt notification	書留サービス	受取人がメールを読み出した時に、変換プログラムがSR-UAPDUを送信する。またSR-UAPDU受信処理を付加する必要がある

関係上O/R ネームの要素は最大長に制限を設けているが、ADMD名/PRMD名は64文字、パーソナルネームの名字は48文字等のように大きめにとっている。

(2) 個別メールシステムとMHSの相互接続方法

ELMSとMHSの相互接続を通して、センタ型個別メールシステムをMHSに接続する方法について、次のような一般の方針をたてることができる。

- ① 個別メールシステムの個々のユーザの管理については、MHSで行わず個別メールシステムにまかせるのが現実的である。但し、P1プロセスでメッセージを受信した場合には、受取人の正当性を個別メールシステムに尋ねる機能が必要である。
- ② ①の結果MHSへのアクセス要求/解放をユーザプロセスの間で共通化する必要がある。
- ③ 個別メールシステム内のメールの配信は独自に行わせる。従ってP1プロセスでは個別メールシステムのユーザ宛のメッセージのコピーは作成しないようにする必要がある。
- ④ 個別メールシステム内の名前の体系の制限により外部ユーザのネームの対応をとる必要がある場合がある。
- ⑤ 配信の処理の対応をとる必要がある。すなわち、センタ型メールシステムでは、相手メールファイルに書き込めばメールの配信が終了するが、外部

ユーザ宛のメッセージに対しては、配信通知待ちの状態を管理しNOTIFY指示で配信の完了とする機能を追加する必要がある場合がある。

- ⑥ アクセス管理や符号化情報タイプ登録などMHSの接続に不可欠なものを除いて、個別メールシステムが提供しない機能は送信時には使用せず、受信した場合は無視するのが現実的であると考えられる。

6. おわりに

本稿ではMHSパッケージの実装とセンタ型メールシステムELMSとMHSの相互接続について述べた。今後はACSE/プレゼンテーション層にインタフェースする1988年版MHSの実装、X.32パソコンからの簡易なMHSアクセス方式の検討等を行う予定である。最後に、日頃御指導頂くKDD研究所村谷所長、小野次長、浦野情報処理研究室長に感謝します。

- 参考文献 [1]: CCITT, "Rec. X.400 ~ X.420", Oct. 1984.
 [2]: 鈴木, 加藤, "OSIトランスポート・プロトコルのインプリメントと製品検証", 情処学会分散処理システム研究会, 22-9, May 1984.
 [3]: 鈴木, 加藤, "OSIセッションレーヤ標準のインプリメント", 情処学会分散処理システム研究会, 24-4, Nov. 1984.
 [4]: 鈴木, 加藤, 浦野, "MHS用ソフトウェア・パッケージの作成", 第34回情処全大, 7Z-6, March 1987.
 [5]: 鈴木, 浦野, 小野, 渡辺, "インハウス・ネットワークにおけるエレクトロニック・メール・システム(ELMS)について", 情処学会分散処理システム研究会, 11-5, Nov. 1981.