

言語としてみたメールアドレス： 宛先アドレスへの集合演算の導入

舛本 克典 村山 優子 天野 橋太郎

{katsy,murayama,amano}@nets.ce.hiroshima-cu.ac.jp

広島市立大学情報科学部

電子メールはコンピュータネットワーク上で最も活用されている応用である。その利点は非同期通信に加え、手軽なマルチキャスト機能が挙げられる。後者を實現するメーリングリストと一般に呼ばれるグループは、個人のメールボックス同様に名前を持ち、そのメンバー構成を管理するメールリレーのドメインアドレスをアドレスとする。このメーリングリストは非常に便利である反面、一時的にあるメンバーだけにメールを送りたくないということができない。本研究では、メールの宛先アドレス指定に集合演算を用いることにより、より柔軟な宛先指定を行なうことを試みた。これはアドレス指定も一つの言語として扱える可能性を意味する。今回の実装では扱う全てのアドレスは同じドメイン内にあると仮定する。

Email addresses as a language: Incorporating set operations to an email destination address

Katsunori Masumoto Yuko Murayama Kitsutarō Amano

{katsy,murayama,amano}@nets.ce.hiroshima-cu.ac.jp

Faculty of Information Sciences, Hiroshima City University

Various applications have been running on the Internet. Email is the most essential one among them. It provides the users with the multicast function as well as asynchronous communication. Multicast is used for a mail list which is a group for discussion or information distribution and has a group name; its address is the domain address of mail relay which manages the composition of group members. The current problem with this multicast function is that one cannot exclude some members out of a group. We propose incorporating set operations such as “-” into the destination address of an email message, so that the group addresses could be used more effectively. A prototype has been implemented to exercise such an addressing scheme in a small departmental domain.

1 はじめに

今日電子メールは従来の個人間通信のような利用のみならず、TV番組において電子メールアドレスを公表し、視聴者からの回答を求めるなどその利用は幅広い。近年のインターネットの爆発的な普及によるユーザー数の増加がその背景にある。電子メールは、非同期ではあるが即時性やグローバル性に富む。しかし、一通のメッセージを他の多くの人に同時に送信するマルチキャスト機能が最大の利点であろう。このマルチキャスト機能は、同じテーマに興味を持った人たちが集まり電子メールを介して意見を述べ合うメーリングリストにおいて特に活用されている。しかしながら現在のマルチキャスト機能では、一時的にあるアドレスにメールを送りたくない場合、対処できない。そこで本研究ではメールの宛先アドレスに集合演算を組み込むことのできるような問題を回避し、さらにメールの利便性を高めることを提案する。

以下に、電子メールの概要を説明し、本研究を報告する。

2 現在の電子メールシステム

本節では、電子メールシステムの概要を述べる。メールの使用環境は様々であるが、典型的と思われるものを以下に説明する。

2.1 メールシステムの概要

ITU-Tで標準化されているモデルによると、ホスト間で通信手順に基づき、メールのやりとりを行なうのがMTA (Mail Transport Agent) であり、MTAにより配送されたメールをユーザーが読み書きするためにMUA (Mail User Agent) がある。(図1) 電子メールを使用するためには配送システムであるMTAを構築しなければならない。多くの場合はメールサーバを置き、それが電子メールの管理を一括して行う。メールサーバに登録されているユーザーはそれぞれメールボックス(郵便箱)を持ち、ここから必要な時に読み出す形になる。小さな組織ではメールサーバは一台という場合もあるが組織が大きくなるにつれてメールサーバを複数台設置する場合が多い。

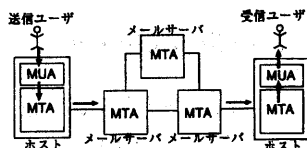


図1: MUAとMTA

メールの送信の際にはUNIX系のワークステーションでは、MUAであるmail,mhなどがMTAのsendmailに各メッセージを引き渡し、sendmailは主に以下のような処理を行なう。

- メール宛先アドレスを解釈
- メール配送経路の決定
- エイリアス(別名)の検査

メールサーバを設定している場合は、クライアント側のMTAはエイリアス(別名)の検査だけを行ない、後はサーバのMTAに渡す様に設定を行なう場合が多い。通常、MTA間のメールのデータ転送方式はTCP/IP上ではSMTP(Simple Mail Transfer Protocol)[1]¹を使う。

メールサーバは宛先アドレスからDNS(Domain Name System)を使用してMX(Mail eXchange)という経路配送情報に従いメールを配送する。目的地が自分であれば、宛先のユーザーのメールの蓄積場所にメールの内容を書き込む。宛先がこのマシンでなければ、SMTPを使い目的のメールサーバにメールを転送する。

受信時には、メッセージがドメイン宛に送られた場合は、そのドメインのメールサーバ内のMTAがエイリアスやユーザーIDの有無などの検査を行いメールの蓄積場所にメールの内容を書き込む。電子メールがホスト宛に送られた場合、ホスト内のMTAがエイリアスの検査を行いそのドメインのサーバ内のMTAに転送する。ユーザーがメールを読む場合は、ユーザー側のホスト内のMTAがサーバ内のMTAからSMTPを介してメールデータを受信する。ユーザーがMUAであるmail,mhを使用し、そのメールを読む。図2は、メールサーバを介してメールを送受信した場合の例を示す。

2.2 マルチキャスト機能

メールのマルチキャストを実現する代表的な機能を以下に示す。

- CC,BCCの指定
カーボンコピー(Carbon Copy:CC)は電子メールの写しを受取手(To:で指定した宛先)以外に送信する機能である。受取手は自分以外に同じ内容のメールが誰に送信されたことがわかる。隠しカーボンコピー(Blind Carbon Copy:BCC)は、カーボンコピーと同じメールの写しを受取人以外に送信する機能であるが、受取人には写しに誰に送られたのかかわからない。
- エイリアス(別名)
エイリアスは電子メールのアドレスを別の名前登録する機能で、sendmailでは別名(/etc/aliasesなど)ファイルに書き込むことにより設定される。例えば

saru: katsy

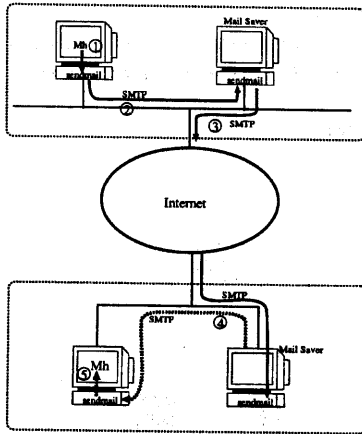
と書かれていれば、saru宛ならsendmailがkatsyに宛先を書き換える。又、エイリアスは複数の電子メールア

¹最近ではESMTP[2]も多く使われる。

は既存のシステムを利用することを考える。

3.1 システム構成案

以下のようなシステムの構成案が考えられる。



- ① ユーザがメールを書く
- ② sendmailによりサーバのsendmailに転送
- ③ DNSから宛先を調べ送信
- ④ 着積場所からメールを転送する
- ⑤ ユーザがメールを読む

図 2: メールの送受信

ドレスに対しても別名が可能である。

```
animal: saru, inu, kiji
```

が最も単純な書き方で、animal 宛のメールを saru と inu と kiji に送るものである。又、宛先をファイルに書き込んでおくことも可能で

```
animal::include:/var/mail-lists/animal-list
```

のように animal という宛先アドレスに対して、その構成員を animal-list というファイルに指定できる。

3 アドレス指定の問題とその解決

電子メールのマルチキャスト機能において特にエイリアス機能は、メーリングリストによく使用される。しかしエイリアスのアドレスの一部にメールを送りたくないという指定はできない。本研究では、宛先アドレスに集合演算を組み込むことにより宛先アドレス同士の引き算やAND演算を可能にし、そのような不都合を解決することを提案する。ただしメールサーバがメールを一括管理する UNIX 系の環境とする。

このような研究で新しいシステムを構築するものとしてメール配送システムの MILD[3] などが研究されているが本研究で

1. メールの集合演算のための独自の中央集中処理型のサーバを用意し、そのサーバ宛に集合演算を含む電子メールを送り、そこでサーバに登録されたエイリアスだけを解決し、メールを出す。
2. MUA-MTA モデルにおいて MTA に集合演算を含む宛先アドレスを解析する機能を持たせる。この場合 MTA の全てを作り直すことはせずに 2 節の sendmail のような既存の MTA を利用するプログラムを作成する。このプログラムを電子メールの送信者のドメインのメールサーバに載せて、もし他のドメインのユーザ情報やエイリアス情報が必要な時は自分で直接調べる。
3. 2 の場合において各ドメインのメールサーバにプログラムを載せ、そのプログラムが自分のドメイン内のアドレスやエイリアスの解決を担当し、他のドメインのメールサーバとその情報をやりとりし、集合演算を含む宛先アドレスを解決する。

1 の手法はサーバに多大な負荷がかかったり、非常に手間がかかるなどの短所が多い。2 の手法は調べるドメインのメールサーバのマシンをプログラムが DNS で調べなければならないので非常に複雑になりやすい。以下では 3 番目の方法について考察する。

3.2 各ドメインのメールサーバに集合演算解析プログラムを載せるシステム

以下ではまず集合演算を含む宛先アドレスのモデルと、MTA と集合演算解析プログラムを関係に述べた後、このシステムにおいてローカルなドメイン内と異なるドメインのメールサーバ間のそれぞれにおいて集合演算解析プログラムが必要とする機能について述べる (図 3 参照)。

3.2.1 集合演算を含む宛先アドレスのモデル

A, B を集合とする集合演算を含む宛先アドレスのモデルを以下に示す。

- 和 (+) $A + B = \{x | x \in A \vee x \in B\}$
- 差 (-) $A - B = \{x | x \in A \wedge x \notin B, B \subset A\}$
- AND (&) $A \& B = \{x | x \in A \wedge x \in B\}$
- 括弧 (()) 演算の優先

上記のような演算を宛先アドレスに組み込むことを目的とする。

3.2.2 集合演算解析プログラム

集合演算解析プログラムは集合演算を含む宛先アドレスを読み込みエイリアスなどの解析と演算を行ないMTAに返すプログラムである。サーバのMTAと集合演算解析プログラムとの間でのやりとりを行なうには以下の方法を使う。

MTAは通常、宛先を解析してメールを目的のサーバのMTAに送信する。この時にMTAは、エイリアスの検査も行なうがそのエイリアス機能は一括送信や転送だけでなくあるアドレスにおいて、そのメール全体を標準入力としてプログラムに渡すことも可能である。この機能を利用すれば、あるアドレスならばメール全体を集合演算解析プログラムに渡すことができる。又、通常の宛先には集合演算解析プログラムに渡すアドレスが書かれる必要があるため通常の宛先とは他に、集合演算を含む独自の宛先情報が必要とする。

以降本稿において集合演算解析プログラムがエイリアスを個々のアドレスに分解し演算ができる状態にすることをアドレスを解決すると言う表現を使う。

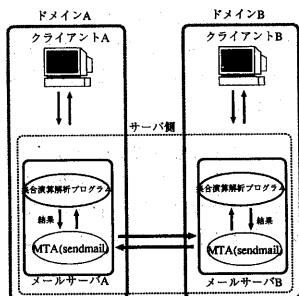


図3: 集合演算を含む宛先解析のシステム

3.2.3 同一ドメイン内での拡張案

エイリアスは一つのシステムだけでなく他のシステムに設定することが出来る。例えば図4の様にメールサーバAのエイリアスファイルに animal という宛先アドレスなら、Aの末端にあるシステムBのホストのファイルを見るという設定が出来る。

このようなエイリアスを解決するためにローカルなメールサーバにある集合演算解析プログラムと末端のクライアントのホスト間で何らかのエイリアスの情報をやり取りする必要がある。その方法としてSMTPを使うことが考えられる。

メールの転送を行うためのプロトコルであるSMTPは送信側SMTPと受信側SMTPがTCP接続を確立し、送信側のSMTPはコマンドを発信し、受信側のSMTPはそれに答える。この送信側のSMTPのコマンドの中にエイリアスを解決するためにEXPN (expand: 展開) コマンドがある。集合演算解析プログラムがこのSMTPを話し、EXPNを使うことが出来

れば、図4のようなエイリアスも解決できる。図4で使用する

```
A: ESPN animal
B: 250-<saru@cha.nets.ce.hiroshima-cu.ac.jp>
B: 250 <kiji@cha.nets.ce.hiroshima-cu.ac.jp>
```

のようになる。

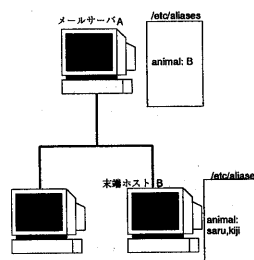


図4: ローカルなドメイン内でのエイリアス

ローカルなドメイン内での集合演算解析プログラムの必要な機能をまとめると以下のようなになる。

- エイリアスを解決し、もしそのローカル内の他のホストへの指定があればそのホストへSMTPを使い必要な情報を取得したのちに演算を行ないその結果をMTAに渡し、メールを送信してもらう。

以上のことが実現できれば、同一ドメイン内でそのドメインのメールサーバに集合演算解析プログラムを載せることでそのローカルなドメイン内のアドレスの解決ができる。

3.2.4 異なるドメインのメールサーバ間での拡張案

図3において異なるドメイン間において集合演算を含むアドレスの解決をするには以下のような方法が考えられる。

各ドメインのメールサーバが集合演算解析プログラムを持ち、宛先の中に自分のドメイン以外のアドレスがあれば自分の解決できるところだけを解決し、他のドメインのアドレス部分はそのドメインの集合演算解析プログラムにメール全体を渡す。全部解決できた時点で、初めて演算を行ないその結果のアドレスにメールを送信する。この場合のプログラムに必要な機能を以下に示す。

- メールサーバは自分のドメイン内で解決できるアドレスやエイリアスを解決したら [] の中にそのアドレスを書き込む。 [] 内のアドレスで @ 以下が省略されている場合はそのドメイン名を書き加える。

- 解決できないアドレスがあればその宛先の集合演算解析プログラムにメール全体を送る。その場合、宛先が末端のホスト宛ならばそのホストがそのドメインのメールサーバの集合演算解析プログラムに転送する。
- 宛先に書かれているアドレスやエイリアスを全て解決して ([] で全て囲まれる) 初めて演算を行い、その結果のアドレスにメールを送る。

上記のことが実現できれば組織外にあるアドレスやエイリアスも集合演算を含むメールアドレスに含むことが可能になる。

このような過程の例を図5に示す。この図では「まさる」、「さとる」、「かずのり」は広島市立大学に属しており、メーリングリスト A,B 内のリストに入っている。広島市立大学の送信者「さとる」は、集合演算を含む独自の宛先に

A-B@okayama-u.ac.jp

と指定したとする。手順は以下のようになる。

1. 広島市立大学のメールサーバにある集合演算解析プログラムはAは@以下が無いことから自分のドメインと判断しアドレスの解決を行い、[]内にアドレスを書き込む。この結果のように[]とドメインをつける。

A-B@岡山大
↓
[[まさる@広島大+さとる@広島大+かずのり@広島大]-B@岡山大

2. 「B@岡山大」は広島市立大学のドメイン外であり、エイリアスか個人名であるのかわからないので岡山大のメールサーバにある集合演算解析プログラムに手順1の結果の[]を含むアドレスを集合演算を含む独自の宛先に書き、メール全体を渡す。
3. 岡山大の集合演算解析プログラムは集合演算を含む独自の宛先内の自分の担当するアドレスを探し出しそれを解決した後に[]をつける。

[[まさる@広島大+さとる@広島大+かずのり@広島大]-B@岡山大
↓
[[まさる@広島大+さとる@広島大+かずのり@広島大]-
-[まさる@広島大+さとる@広島大]

4. もう解決すべきアドレスがないのでアドレスの演算を行なう。

5. 演算の結果のアドレス「かずのり@広島大」にメールを送る。

送信者 : さとる
 独自宛先: A-B@okayama-u.ac.jp
 人名 : 広島市立大の人物
 英字 : メーリングリスト

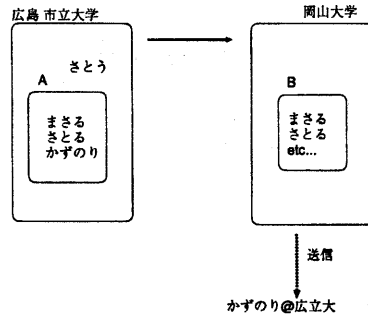


図5: メールサーバ間の集合演算を含むアドレス解決

4 プロトタイプ製作

4.1 概要

本研究では、集合演算を含む宛先アドレスを読み込み解析するプログラムのプロトタイプ作成した。手法と仕様を以下に示す。ただし扱われる全ての電子メールアドレスは一つのドメインにあると仮定した。

4.2 手法

本稿の3.2.2の手法を用い、To: set-mail 宛ならばメール全体を集合演算解析プログラムsmailに渡す。MTAはsendmailを使用した。又、電子メールの宛先や日付や配送に関する情報を含むヘッダにおいてXで始まるフィールドは、電子メールの配送には関係なく、利用者が自分で定義するフィールドでさまざまな情報を残すために使われる。そこでヘッダの宛先To:の他に集合演算を含む独自の宛先情報XTo:を使用する。XTo:には、送信者が集合演算を含む宛先アドレスを書き込む。使用できる演算子は和(+), 差(-), AND(&), 括弧(())である。差演算において"\"は"hiroshima-cu"のようなアドレス中の"-"と区別するために入れる。演算の優先順位は算術演算の和差積と同じとする。使用時には利用者がヘッダ内にXTo:を書き加え、その後1000文字以内で上記の演算子を使いアドレスを書き込み、後は通常のメールと同じ方法の場合XTo:を書くと同様状態で送信されBXTo:を使用するとBccで送ったのと同じように、メール受取人は自分以外誰に同じ内容のメールが送られたか知ることができない。通常のBccと

cc は使えない。メールを受けとった sendmail は、エイリアスを探し、宛先が set-mail 宛ならアドレス解析プログラム smail にメールを渡す。(図 6)

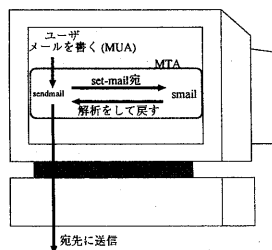


図 6: smail

5 むすび

本稿では宛先アドレスへの集合演算の導入について述べた。その実現案として本研究では各ドメインサーバごとに集合演算解析プログラムを載せる方法を述べたがまだ問題点も多く、上記の拡張を行なうには他の集合演算解析プログラムにメールの中身を全体を送るのでセキュリティ的な問題点生まれたり、エイリアスの中にさらに他のドメインのエイリアスが書かれているといった入れ子構造のエイリアスへの対応など問題点が多い。又、プロトタイプは実質一つのホストにあるエイリアスしか参照することができない。今後はプロトタイプの拡張と上記のような問題点の研究を進めていく予定である。

謝辞

本研究を進めるにあたり、助言を頂いた日立システム開発研究所の寺田真敏氏に感謝致します。又、広島市立大学情報科学部情報工学科の王仁峰君、中本泰然君とのディスカッションは研究を進める上で大変参考になりました。

参考文献

- [1] J.Postel, "SIMPLE MAIL TRANSFER PROTOCOL," RFC821, August 1982.
- [2] J.Klensin, N.Freed, M.Rose, E.Stefferud and D.Crocker, "SMTP Service Extensions," RFC1869, November 1995.
- [3] 萩野 浩明, 門林 理恵子, 清 一陸, 塚本 昌彦, 西尾 章治郎 "推論機能を用いたメールの分配システム MILD におけるメール検索機構," マルチメディア通信と分散処理ワークショップ論文集, 1995.
- [4] SunSoft, "メールシステムの管理," October 1995.
- [5] 下山 智明, 城谷 洋司, "SUN システム管理," October 1991.