

メール宛先アドレスの言語化における ネットワークアクセス制御

村山 優子
岩手県立大学

電子メールにおけるグループ通信はメーリングリストによるマルチキャストにより実現されているが、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ような指定ができない。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。実装はドメイン内環境で行った。しかし、通常メーリングリストは地球規模のネットワーク上の様々なドメインで管理されている。従って、本手法の実用化のためには、異なるドメイン間のメーリングリストの管理についてのアクセス制御が必要となる。本稿ではこの問題提起を行う。

Network Access Control for Implementing Flexible Email Addressing

Yuko Murayama
Iwate Prefectural University

Group communication in email is performed by multicast making use of mailing lists. However, it is not quite possible to exclude any member out of the list temporarily. We try and apply set operations to the destination address field of an email message to solve the problem. The implementation is done within a domain, however, in reality mailing lists are managed in a distributed way over the Internet. We need some control over access to those mail lists. This paper presents this problem.

1 はじめに

コンピュータネットワークの応用として最も普及している電子メールでは、そのマルチキャスト機能によりグループ通信を実現している。

単純なマルチキャストは、宛先 (To:) や CC や BCC などのコピー先指定欄などにグループのメンバーのアドレスを羅列することにより可能となる。メンバーの数が多い場合、あるいは複数

の宛先が恒常的に使用される場合は、グループ名とその構成員をあらかじめ登録しておき、個人アドレス同様にグループ名を宛先に指定することでマルチキャストを行う。

このグループ名宛先指定における問題点として、一時的なメンバー削除の機能がないことが挙げられる。例えば友人の誕生日のプレゼント相談などを行うとき、その友人を除いた宛先指定が望まれる。そこで、本研究ではこのような問題の対策として、宛先アドレス指定に集合演算を組み込むことによりメールの利便性を高めることを提案した [2]。インターネット上で、この案を実現するためには、グループのメンバー管理を行うメールサーバ間の自律と協調の必要性が生じる。本稿では、メールサーバに集合演算処理機能を組み込む実験の報告に加え、グループのメンバー情報の分散処理をインターネット上で行なうためのアクセス制御についての問題提起を行う。

2 メールアドレスへの集合演算の導入

グループは名前とメンバーをメールサーバに登録することにより作成され、グループ宛のメッセージには、その宛先欄にグループ名とそのアドレスとして以下のようにグループが登録されているメールサーバのドメインアドレスを指定する。

To: skiers@soft.iwate-pu.ac.jp

上記の例は、岩手県立大学ソフトウェア情報学部のメールサーバに登録されているスキー仲間のグループに宛てる場合である。特にアクセス制御などが施行されていない限り、グループ宛のメッセージはグループ外からも送付可能である。

本研究ではユーザはメッセージの宛先欄に以下のような演算子を指定できるようにした。

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

例えば以下のような宛先指定が可能となる。

To: skiers@soft.iwate-pu.ac.jp
- santa-claus@christmas.org

上記の例ではスキー仲間のメールリストから santa-claus@christmas.org というメンバーを除いた宛先を指定している。

このようなアドレスへの集合演算処理実現のためには、以下の手続きが必要となる。ただし、ここでは演算において、例えば $A - B$ の場合、 A を対象アドレス群、 B を引数アドレス群と呼ぶ。

1. 演算の対象アドレス群を個別アドレスに分解
2. 演算の引数アドレス群を個別アドレスに分解
3. 上記 1 と 2 の個別アドレス群同士について演算処理
4. 3 の演算結果に基づくメッセージ転送

本稿ではこれらはすべて一ヶ所で処理されるものと仮定し、モデル化、プロトタイプの報告を以下に行い、その後、これらの処理を一ヶ所では行えない場合について考察する。

3 システム・モデル

ITU-T により標準化されたメールシステムのモデルでは、MTA (Mail Transport Agent) がメッセージの転送および配送を行い、MUA (Mail User Agent) を通してユーザが読み書きする。実際のメールシステムはクライアント・サーバモデルの形で実装されており、図 1 に示すように、メールサーバが MTA の機能を有し、クライアントシステムが MUA と終端 MTA などの機能を有す。

メールサーバにはユーザのメールボックス (郵便箱) が置かれ、メッセージが配送されて来る。メールサーバの 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、MTA は別名 (エイリアス) の検査を行う。別名は通常ファイルなどに固定的に定義され、あるユーザについての別名を登録するためにある。メーリング・リストはこの別名機能を使い作

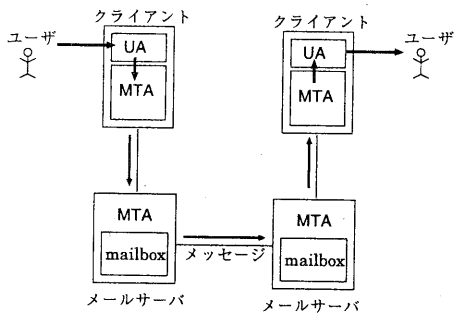


図 1: メールシステムのモデル

成されている。例えば、前出の **skiers** の場合、**soft.iwate-pu.ac.jp** というドメインを担当するメールサーバで別名指定ファイルにグループ名 (**skiers**) とその構成員を定義している。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。

本研究では通常のメールシステムのモデルにおいて、メールサーバの MTA に集合演算処理機能を付加することにした (図 2 参照)。これは、サーバがグループの構成員を定義する機能を持つからである。

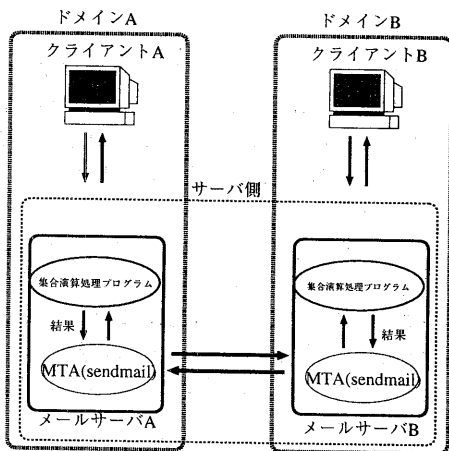


図 2: 集合演算処理機能をもつメールシステム

4 プロトタイプ

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システムのプロトタイプを製作した。ただし演算を施す対象のメールアドレスはすべて一つのドメインだけとした。

ユーザはメッセージ発信の際、集合演算を含むアドレスを XTo: の欄に、To: 部分に別名ファイルに登録したプロトタイプシステムの別名を指定し、通常のメールと同じ方法で送信する。メッセージはプロトタイプメールサーバにおいて図 3 に示すように MTA (sendmail) から集合演算処理プログラム **smail** に渡される。XTo: の欄の処理結果が今度は To: 部分に書かれ、再度 **sendmail** に渡され、通常のメッセージとして配送される。

XTo は 2 種類あり、

- XTo: cc と同じ状態で送信され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。
- BXTTo: 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。

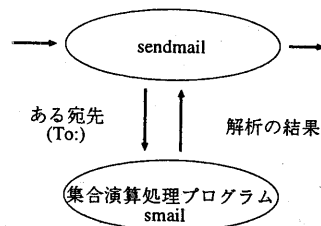


図 3: プロトタイプシステム

システムは SUN Sparcstation 20, OS は SUN OS 4.1.4, MTA には **sendmail R8** を使用した。集合演算処理のプログラムは **yacc/lex** で作成し、解析の結果を **sendmail** に渡す作業に当たるプログラムを **c** 言語で記述した。

5 メンバー情報に対するアクセス制御

現在のプロトタイプでは集合演算の対象となるアドレスがすべてプロトタイプシステム上に定義されなければならない、インターネット上での実用化に向けて拡張が必要である。

同一ドメイン内の異なるシステム上に定義されたアドレスについては、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MTA 同士の通信プロトコルであるSimple Mail Transfer Protocol[1]により構成員情報などの交換が可能であろう。すなわち、宛先アドレスの中にそのドメイン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うことが可能となる。

しかし、演算対象アドレスは本来メール可能なすべてのネットワーク上のアドレスでなければならない。メールシステムはインターネット上で分散処理を行なっているため、グループの登録と管理が様々なサイトで自律的に行なわれている。2節で必要とした対象アドレスや引数アドレスの分解のためには、図4に示すように、ひとつのドメイン内の情報だけでは対処できない場合が多いと考えられる。そこでドメイン同士の協調が必要となるが、以下の問題が生じる。

1. メンバー情報の秘匿性
2. 個人のメールアドレスの秘匿性
3. 対象アドレスと引数アドレスの分解度

上記1では、個々のグループのメンバー情報は公開の場合は良いが、非公開の場合そこでアドレス分解が不可能となり問題となる。

また2では、個人のメールアドレスの秘匿性を重視するユーザが存在する場合、別名でグループに登録している可能性もあるので演算処理が正しく行えない可能性がある。

3の問題は、対象アドレスと引数アドレスの分解度を同じレベルにすることである。対象アドレスと引数は同じ分解度でなければ、演算できない。すなわち、どちらかが個別アドレスで、ど

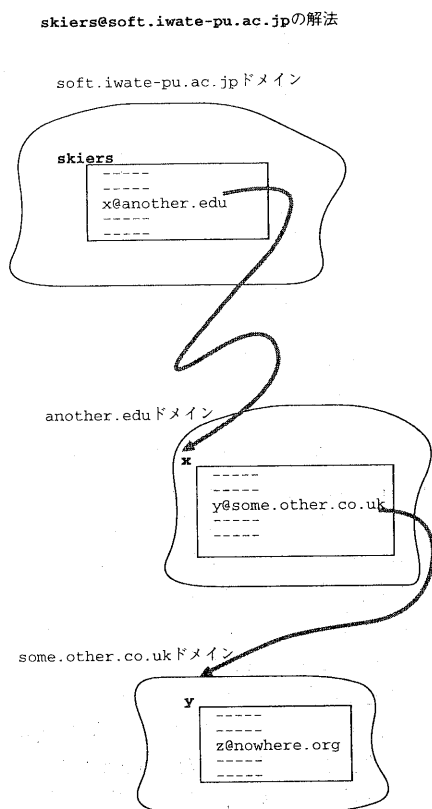


図4: メンバーリストのリンク

ちらかがグループアドレスでは演算ができない。また、複数のグループが同じメンバーを持つ場合、グループ名レベルでは演算はできない。従って、アドレスの分解度は個人メールアドレスが妥当であろう。

1と2の問題を解決するには、アドレス分解作業を各ドメインが独立に行い、メンバー情報の秘匿性を保持しながら、集合演算を施さなければならない。対象アドレスの秘匿性と引数アドレスの秘匿性を同時に保つためには、演算処理を部分的に行い、さらにその処理を信頼できる第三者の演算ブローカのようなエージェントの存在が必要となろう。このエージェントは、対象アドレス群を引数アドレス群とともに演算処

理する。対象アドレス群については、そのアドレス群を管理するドメインのサーバが情報提供を行い、引数アドレス群についてはその引数グループアドレスを管理するサーバが情報を提供するなどの工夫が必要となる。エージェントは、どのサーバがどのアドレス処理を依頼したかなどの情報についての秘匿性は保持すると仮定する。しかし、サーバ側はグループ名など余分な情報はエージェントには与えないとする。従って、エージェント側では、与えられたアドレス群と別に与えられた引数アドレス群に対して演算を施すという処理だけを行うこととする。エージェントの処理結果は対象アドレス群の管理サーバに返され、そこからメールが配布される。

6 むすび

本研究では宛先アドレスへの集合演算の導入を提案し、その際に発生するメールサーバの自律と協調の問題について考えた。この集合演算の延長上にはメールアドレスが実は言語として機能できるという考えがある。MILD[3]はまさにそれを実現しているが、インターネット上の既存のメールシステムと全く独立なものを構築している。本研究では、既存のメールシステムへの変更を最小にしたシステムを目指し、可能な限りの言語化を試みたいと考える。現在のプロトタイプは実質上一つのマシン上でしか使用することができないが、その拡張について考察した。そこで浮上する問題にドメインのサーバ間の自律性と協調の度合いである。アドレス解決におけるドメイン間のアクセス制御問題を挙げ、演算エージェントを用いることを提案した。今後、さらにこの処理系についてエージェントなどの定義、プロトコル構築を検討して行く計画である。

参考文献

- [1] J.Postel, "SIMPLE MAIL TRANSFER PROTOCOL," RFC821, August 1982.
- [2] 舛本 克典 "メールアドレスへの集合演算の導入," 広島市立大学 情報科学部 情報工学科 4 年生卒業論文, 1998 年 3 月.
- [3] 萩野 浩明, 門林 理恵子, 清 一隆, 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.