

RMXにおける受信者別メール本文生成機能および本文参照型アドレスの実装

松澤 慧[†] 北園 達也[†] 小船井 寛[†] 遠山元道^{††}

[†] 慶応義塾大学理工学部情報工学科 〒223-8522 神奈川県横浜市港北区日吉 3-14-1

E-mail: [†]{matt,zonop,obunai}@db.ics.keio.ac.jp, ^{††}toyama@ics.keio.ac.jp

あらまし ルールベースメール配送システム RMX において、先行研究段階では、受信者ごとに本文を編集する機構は送信者が作成した文章のみを扱うものであった。そこで本論文ではデータベースの値から、受信者全員に対して個別の語句を本文に差し込み、かつデータベースへの問い合わせ回数を最小限にしながら、多様な文章を生成する手法を提案した。また、先行研究段階では、RMX はデータベースに問い合わせを行うにも際に漢字やひらがな等、メールアドレスに使用できない文字列をパラメータとする検索ができなかった。本論文では、メールアドレスに”\$”を記入し、本文に記述された文字列を参照することによって、メールアドレスに書けない文字をパラメータとしてデータベースに問い合わせることが可能になった。

キーワード 電子メール, RMX

Implementation of the E-mail Text Generating Function and Referring to Body Form Address in RMX

Satoru MATSUZAWA[†], Tatsuya KITAZONO[†], Kan OBUNAI[†], and Motomichi TOYAMA^{††}

^{††}Department of Information and Computer Science,
Keio University

Hiyoshi 3-14-1, Kouhoku-ku, Yokohama-shi, Kanagawa, 223-8522 Japan

E-mail: [†]{matt,zonop,obunai}@db.ics.keio.ac.jp, ^{††}toyama@ics.keio.ac.jp

1. はじめに

RMX(Rule-based e-Mail eXchange System) はユーザが設定したルールとメールアドレスを基に、データベースへアクセスを行い、得られたユーザ集合に対してメールを配信するメール転送エージェントである。RMX の機能のうちの一つにメール本文において配送ルールとパラメータの組み合わせを条件として、ユーザに適した文章をサーバ上で作成する本文編集機能がある。この本文編集機能だが、先行研究段階では、メールを送信するたびにデータベースに問い合わせをする処理を行なっているため、大量のメールを送信するとき、送信者の意図しない時間を必要とする可能性があった。ユーザがこの機能を使用するにあたって、この処理でのデータベースへの問い合わせ回数は極力抑えるべきである。そこで本論文ではデータベースに問い合わせ回数をメールの送信件数からメール本文での条件の回数に依存させることで、処理時間を削減する手法を提案する。更に本文編集機能に加えて、データベース上のアドレス以

外の他のユーザ情報を取得する機構を実装した。ここで取得したデータベースの値を本文に挿入することで、それぞれの受信者に対して個別のメール本文の生成が可能になった。

また、データベースに問い合わせを行う RMX は先行研究段階ではメールアドレスからクエリを生成するため、一般にメールアドレスに使用できないひらがな、カタカナ及び漢字等を使用することができなかった。そこで本論文では、メールアドレスに”\$”と表記した上で、本文に”\$パラメータ”とすることによって、メールアドレスでは使用することのできない文字をパラメータとしてデータベースに問い合わせることを可能にした。以下、本稿の構成を示す。まず、2 章で RMX の概要について述べる。3 章で本文編集機能について、4 章で本文生成機能について、5 章で本文参照型アドレスについて、6 章で評価について述べ、7 章で結論を述べる。

2. RMX

Rule-based e-Mail eXchange(RMX) は遠山研究室が提案し

ている電子メール配信方式である。RMX では下記のような記述により複数の送信先を指定する。RMX ではアドレスの記述方法は基本形式と関数形式の2つがあり、これはそのうち基本形式の記述方式を示している。以下、基本形式アドレス、関数形式アドレスの順に説明する。

< RMX のメール配信先指定 >:=< パラメータ >
< パラメータ > @ < 配送ルール名 > < 配送ルール >
. < サブドメイン > . < ドメイン >

RMX のメールアドレスは以上のようにサブドメイン以前を表す配送範囲記述部分とドメイン部分が“.”の記号で区別されている。配送範囲記述部は一つ以上のパラメータの組み合わせで構成され、@記号によってパラメータと配送ルール名に分けられる。サブドメインは後述する設定ファイルの名前に相当する。RMX はこのような配送範囲記述を受け取る。そして、指定された配送ルールとそのパラメータに基づきデータベースに問い合わせを行い実際の送信先アドレスを得る。最終的に得られたメールアドレスに基づき配送が行われる。図 1

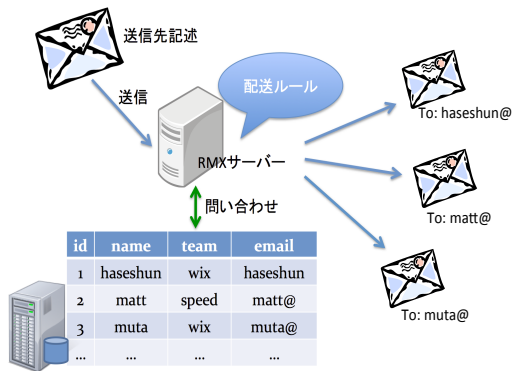


図 1 RMX におけるメール配信の流れ

2.1 配送ルール

配送ルールとは配送範囲記述とそれに基づき送信先メールアドレスを得るクエリを関連付けるルールである。配送ルールは以下のように定義する。

配送ルール名
Type:パラメータの型
query: 送信先メールアドレスを得るためのクエリ

query 部分は SQL によって記述される。RMX は記述された配送ルール名に対応するクエリにパラメータを挿入し、問い合わせを行うことで送信先メールアドレスの集合を得る。このような配送ルールを用いることにより、利用者は簡潔な記述で配送範囲を指定することができる。以下に配送ルール定義の例を示す。

上記の例では学年を integer 型で受け取り、それに基づいて

```
grade
gradeType= integer
grade[1]= select s.address from student s
where s.grade = $1
```

メール配信を行う grade ルールを定義している。メールの宛先が 4@grade.example.edu の場合、図 2 のように query 部分で利用者のメールアドレスと学年が格納されている表 student から、学年が 4 年の学生のメールアドレスを得るクエリを記述している。



id	name	grade	email
1	haseshun	3	haseshun@
2	matt	4	matt@
3	muta	4	muta@
...	...	...	...

図 2 grade ルールと配送例

2.2 複数の配送ルールの組み合わせ (基本形式)

ここでは、複数の配送ルールを組み合わせでメールを送る際に、RMX で使用可能な演算子について説明する。演算子 Envelope-To フィールドの配送ルールに使用することにより、ユーザに対してより詳細な配送範囲の指定を可能とする。

2.2.1 積 集 合

Syntax: < par₁ > < par_n > @ < name₁ >
. < name_n > . < subdomain > . < domain >
Semantics: name₁ (par₁) ∩ ... ∩ name_n (par_n)

“.” は、Envelope-To フィールドに複数の配送ルールを使用したい時に用いられる演算子である。指定された複数の配送ルールの結果の積集合を取り、最終的に得られた結果に対して配送を行う。この演算子を使用するときは、“.” で区分されたパラメータと配送ルール名の数を一致させ、@を挟んでパラメータとそれに対応する配送ルール名を順番に並べる。

例：

toyama.1@lab.grade.lab.example.edu

2.2.2 和 集 合

Syntax: < par₁ > + ... + < par_n > @ < name₁ >
. < subdomain > . < domain >
Semantics: name₁ (par₁) ∪ ... ∪ name₁ (par_n)

“+” は、ある配送ルールに対して論理和を得たい複数のパラメータを指定するときに用いられる。与えられた複数のパラ

メータをそれぞれ配送ルールのクエリに代入し、得られた結果の和集合を取る。そして、最終的に得られた結果に対して配送を行う。

例：

3+4@grade.example.edu

2.2.3 多相 (ポリモルフィック)

$ListOpe := -$

$UnionOpe := +$

$InterOpe := .$

$Arg := string \mid integer$

$ListPara := Arg \mid Arg ListOpe ListPara$

$UnionPara := ListPara \mid ListPara UnionOpe UnionPara$

$InterPara := UnionPara \mid UnionPara InterOpe InterPara$

$RuleList := subdomain \mid rule InterOpe RuleList$

$Address := InterPara @ RuleList . domain$

基本形式は以上の形式を取る。

2.3 関数形式アドレス

関数形式アドレスは以下の形式を取る。

$ListOpe := -$

$UnionOpe := +$

$RuleOpe := . \mid + \mid -$

$Arg := string \mid integer$

$Para := Arg \mid Arg ListOpe Para$

$ParaList := Para \mid Para UnionOpe ParaList$

$Exp := rule \{ ParaList \}$

$ExpList := Exp \mid Exp RuleOpe ExpList$

$Address := ExpList @ subdomain . domain$

学年が3年, または4年で, 名前がmattである学生へメールを送る場合の例は以下のようになる。

例：

grade{3+4}.name{matt}@lab.example.edu

関数形式では配送ルール間のユニオンが可能である。例えば学年が4年である, または名前がmattである学生に対してメールを送ろうとした場合以下のように記述すればよい。

例:

grade{4}+name{matt}@lab.example.edu

また, ルール間に演算子“-”記述することで配送ルール間の差集合が利用可能である。例えば学年が4年であるが, 名前がmattでない学生に対してメールを送ろうとした場合以下のように記述すればよい。

例:

grade{4}-name{matt}@lab.example.edu

3. 本文編集機能

本文編集機能とは, メール本文に記載されたタグ情報をもとに, サーバ上で自動的に受け取り手のユーザに適した文章を作成する機能である。2010年に青山ら[4]が提案したものである。本論文では, その中の Variable Matching について取り上げる。

3.1 Variable Matching

Variable Matching は, if タグを用いて文章ごとに配送ルールを適用する。

$\langle if \text{ 配送ルール名} = \text{パラメータ} \rangle contents \langle /if \rangle$

図3で示すように, 文章を配送ルール, パラメータを明記した if タグで囲うことにより, その条件に適したユーザにのみ囲われた文章を抽出する。この作業はサーバ上で行われる。したがって, 送り手のユーザがタグを用いたメール本文を作成し, 送信すると, 自動で受け取り手のユーザに適した文章だけが抽出されたメールが作成され, 送信される。例えば, 架空のオンラインショップにおいて, 地域ごとに異なるジャンルの販売物で地域限定のタイムセールを行う告知のメールを, 受信者の住む地域に適したメールを送りたい場合, 図3の左側の文面を送信する。このとき地域が東京の顧客は右側上部の文面が, 地域が大阪の顧客には右側下部の文面が送信される。

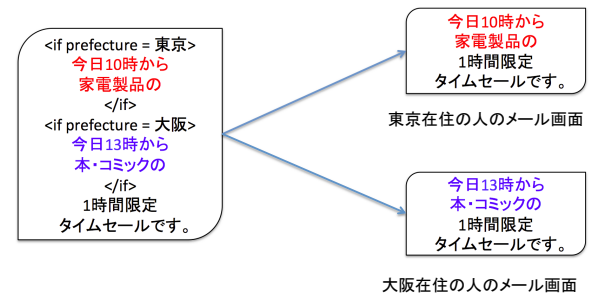


図3 Variable Matching を利用した時の例

3.2 問題点と提案手法

先行研究段階ではメール本文を編集する際に, メール1通ごとにデータベースに問い合わせを行っていたため, 処理時間は転送数に依存していた。そのため転送数が多い場合には送信者が予期しない処理時間を必要とする可能性があった。そこで本論文では送信者が作成したメールの本文を走査し, そこ で得られたタグの条件をもとにデータベース問い合わせを1回のみ行う手法を提案した。これによって処理時間の転送数への依存を解消し, 転送数が多い場合の本文編集機能の処理時間の削減を実現した。

4. 本文生成機能

RMX は、ユーザが指定したルールに基づき、データベースへアクセスを行い、対象となるユーザ集合を抽出する作業をサーバ上で自動で行う。この特徴を生かして、本論文ではメール本文に記載されたタグ情報をもとに、サーバ上で自動で受け取り手のユーザに対応したデータベースの値を挿入し、受け取り手に対して個別の文章を作成する機能を提案する。

4.1 機能概要

本文生成機能では、value タグを用いてデータベースの値を挿入箇所に後述する挿入ルールを適用する。

< value 挿入ルール内のルール >

データベースの値を差し込みたい部分に value タグを使用することにより、RMX サーバは、受信者のメールアドレスに対応したデータベースの値をメール本文に挿入する。この作業はサーバ上で行われるため、送り手のユーザがタグを用いたメール本文を作成し、送信すると、受け取り手のユーザに対応したメール本文がサーバ上で作成され受け取り手に送信される。例えば、value タグに受信者の名前を差し込むルールである”name”が定義されていたとき、図4のように用いる。

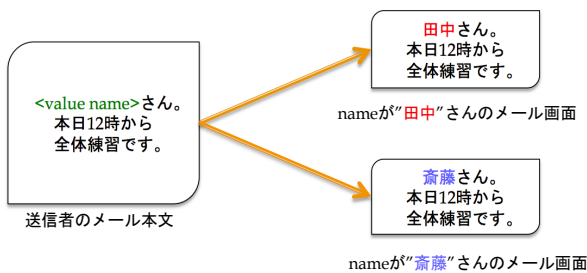


図4 本文生成機能を利用した時の例

4.2 挿入ルール

value タグで、データベースの値を挿入できる一方で、データベースのすべての値を挿入できるように実装しては、住所、電話番号等の個人情報の流出を許してしまう形となり、セキュリティ上の欠陥となりかねない。そのため、本文編集機能では value タグ内で使用することのできる値は、RMX の設定ファイル内で定義された挿入ルール内のルールとして管理者によって制限される。挿入ルールは、以下のように定義する。例えば、メール本文に、student テーブル内の、アドレスに対応した学科をメール本文中に差し込みたいときは、設定ファイルに以下のようにルールを定義しておく。

```
dept
deptType = String
dept[0] = select s.address, s.dept from student s;
```

このとき、本文生成機能は、配送ルールと併用が可能である。もし、配送ルールとして dept を利用したい場合は、設定ファイルに以下のように記述する。このとき、dept[0] で定義されたものが挿入ルール、dept[1] で定義されたものが配送ルールとして処理される。

また、設定ファイルに定義されていない挿入ルールのルールが

```
dept
deptType = String
dept[0] = select s.address, s.dept from student s;
dept[1] = select s.address from student s
where s.dept = $1
```

来た場合、エラーとして、送信者に定義されていない挿入ルールのルールであることを記したメールを送信する。これによって、送信者がエラーに気づき、正しいメールを送るように促す。

5. 本文参照型アドレス

従来のアドレス表記について考える。

例:

name{matt}@lab.example.edu

このアドレスは、name が配送ルールであり、matt がパラメータである。このとき、RMX 上では配送ルール name で定義されている SQL を呼び出し、matt をパラメータとして配送ルールの SQL の where 句に代入し、データベースに問い合わせを行うが、メールアドレスは一般に使用出来る文字が限定されているため、従来の RMX では、パラメータにメールアドレスで使用できない文字を使用できなかった。データベースに問い合わせを行う RMX の性質上限られた文字しか使用することができない要素は、システムを導入する上で大きな妨げとなる可能性がある。この問題を解決するため、本論文では本文参照型アドレスを提案する。

5.1 機能概要

本文参照型アドレスは、配送ルールのパラメータとして本文から代入したい部分に”\$”を記述し、本文の冒頭で、”\$パラメータ”と記述することで、パラメータを本文から参照して、データベースに問い合わせを行うことができる。例えば、学科が情報工学科である学生へ本文参照型アドレスを用いて、メールを送る場合、メールアドレスは次の表記となる。サブドメインは university とする。

dept{\$}@university.example.edu

そして、メール本文には、図5のようにパラメータを記述する。

また、メールアドレスには、本文を参照するパラメータは複数使用可能である。例えば、学科が情報工学科で研究室が遠山である学生に本文参照型アドレスを用いて、メールを送る場合、メールアドレスは以下ようになる。そして、メール本文には図6のように記入する。

\$情報工学科

情報工学科の皆さん、

本日は10時から説明会です、

遅刻する事のないように集合してください、

図5 本文参照型アドレス (パラメータが一つ) の時の例

以下、配送ルールのパラメータに"\$"を用いた数だけメール本文冒頭でパラメータを順に代入していく、

dept{\$}.lab{\$}@university.example.edu

\$情報工学科

\$遠山|

遠山研の皆さん、

本日は10時から発表会です、

遅刻する事のないように集合してください、

図6 本文参照型アドレス (パラメータが複数) の時の例

RMX は、本文参照型アドレスで書かれたメールを受け取ると、RMX は、本文を取得し、本文冒頭にパラメータがあるかどうかをチェックする。このとき、本来のメールの本文をパラメータとして取得する事のないように、パラメータが書かれている行は最初の文字を"\$"として、パラメータであることを明示しておく。そして、RMX はパラメータを上から順に取り出す。このとき RMX では、取り出したパラメータを UTF-8 で URL エンコードを行う、その後、エンコードしたパラメータをメールアドレスの左側から\$をパラメータを代入していく。最後に、メール本文のパラメータで利用した部分の行を削除する。これは受信者にどのようなパラメータを用いたかをそのまま表記するのは、セキュリティ上問題があるためである。

6. 評価

6.1 本文編集機能の処理評価

if タグを一つ本文に用いて、RMX サーバがメールサーバに送信要求する件数を変えたとき、従来の本文編集機能と本論文で提案する本文編集機能の処理時間の評価を行う。RMX サーバのスペックは、CPU に Intel(R) Xeon(R) X5355 2.66GHz × 2、メモリは 16G、OS は OpenSUSE 11.4、MTA として、qmail1.03 を使用している。実験の結果を図7、図8、図9に示す。ただし、表の単位はミリ秒である。結果を見ると、10 通までは、従来手法、提案手法と処理時間に差は見られないが、100 通と 1000 通での処理時間は、提案手法の方が処理時間を大幅

に抑えている。この要因としては、従来の手法が、RMX サーバからメールサーバへ送信要求するたびに、データベースへ問い合わせを行っていたため、データベースへの問い合わせ時間が問い合わせ回数が増えるに連れて処理時間を必要したことに對して、提案手法は、データベースへの問い合わせ回数を if タグの回数にのみに抑えているので、提案手法の処理時間は本文を編集する部分のみのサーバでの処理を行った時間に抑えられることができたからだと考える。

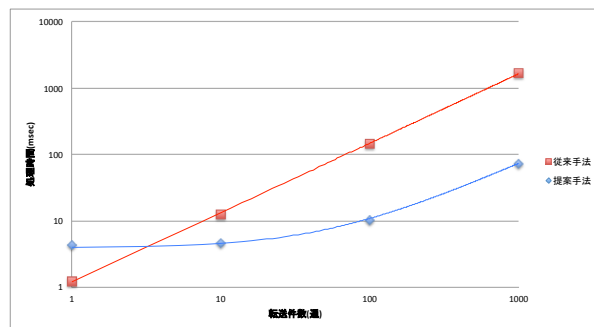


図7 if タグを 1 つ使用した時の処理時間の比較

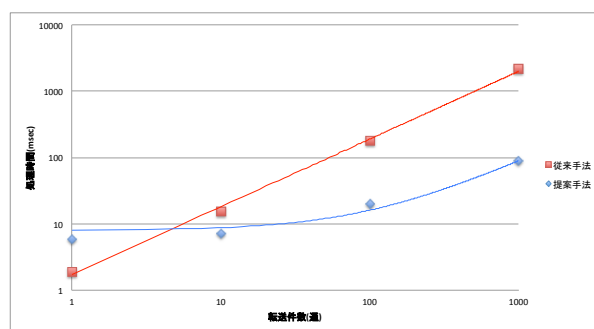


図8 if タグを 2 つ使用した時の処理時間の比較

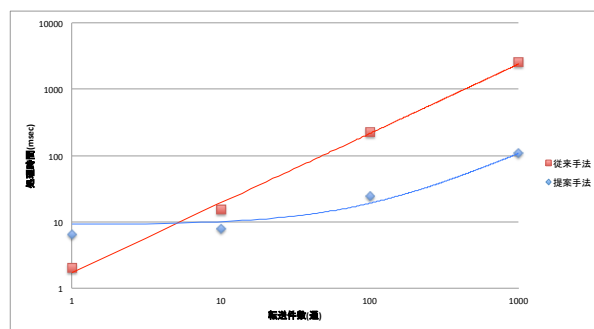


図9 if タグを 3 つ使用した時の処理時間の比較

6.2 本文生成機能での評価実験

被験者 5 人に架空のオンラインショップを想定し筆者が用意したデータベースに登録されている架空の顧客 5 人に向けて商品の発注完了のメールを作成してもらい、通常のメールを用いる方法と本文生成機能を利用した RMX を用いる方法の 2 通りで、メール作成時間を比較し、評価する。ただし、このような業務的なメールはテンプレートがあると仮定し、予め用意した

テンプレートを用いてメールを作成してもらった。この実験において使用出来る本文生成機能はデータベースの属性を value タグとともに記述すると、その値を入れることができると仮定した。図 10 に用意したテンプレートを、図 11 に本文生成機能とテンプレートを利用して作成した文面を、表 1 に今回実験で用いたデータベースを示す。評価した結果を図 12 に示す。作成時間の平均は、通常のメールを用いた場合は 214.7 秒、本文生成機能を用いた場合 74.3 秒であった。これは、本文生成機能は value タグを用いてメールを 1 通作成すれば受け取り手に対応したメールを作成することができることに對して、通常のメールでは、受け取り手の情報をキーボードで入力しなければならないため、メールを受信者に向けて計 5 通作成する必要があったためであると考えられる。この結果を考慮すると、テンプレートで予め準備された業務的なメールほど、送信者の入力の際の負担が減少するため、本文生成機能は有用であるといえる。

表 1 実験で用いた顧客データベース

customer				
name	e-mail	item	price	address
佐藤	sato@	PSP	15000	神奈川県川崎市中原区
鈴木	suzuki@	WiiU	30000	東京都港区
田中	tanaka@	PS3	25000	神奈川県横浜市港北区
伊藤	ito@	X-Box360	20000	大阪府大阪市西成区
本田	honda@	PS-Vita	20000	埼玉県熊谷市

様

Onlineshop.co.jp をご利用いただき、ありがとうございます。
お客様からご注文いただいた商品を本日発送いたしました。
発送いたしました商品は以下のとおりです。
商品:
金額:
お届け先:
(以降の住所はプライバシーのため表示しません。)
Onlineshop.co.jp のまたのご利用をお待ちしております。

図 10 実験で用意したテンプレート

<value name> 様

Onlineshop.co.jp をご利用いただき、ありがとうございます。
お客様からご注文いただいた商品を本日発送いたしました。
発送いたしました商品は以下のとおりです。
商品: <value item>
金額: <value price>
お届け先: <value address>
(以降の住所はプライバシーのため表示しません。)
Onlineshop.co.jp のまたのご利用をお待ちしております。

図 11 本文生成機能を利用した場合の実験の解答

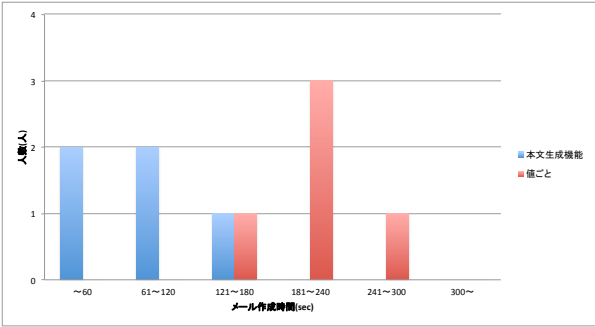


図 12 メール作成時間の比較

7. おわりに

本研究では、RMX における本文に受信者個別のメール本文を作成する本文生成機能、既存の本文を編集する機能の処理を高速にする手法の提案を行った。また、一般のメールアドレスに使用できない文字をパラメータを可能にする本文参照型アドレスを導入した。

本文生成機能では、データベースから挿入ルールを用いて受け取り手個別の文章を生成し、多様な文章を作成する手法を提案した。これによって多様な本文を送信するときに、送信者の負担を大きく減らすことが可能にした。

また、本文参照型アドレスでは、メールアドレスに\$を、本文冒頭にパラメータを書き、RMX で使用するパラメータでの制限をなくした。これによって、データベースに問い合わせするパラメータを一般のメールアドレスで使用出来る文字以外のひらがな、カタカナ、漢字等を利用しているデータベースでも RMX を利用することが可能となった。

文 献

[1] 高畑 理, 藤沼 健太郎, 石橋 玲, 遠山 元道. "Magic Mirror Mailing: 個人情報データベースを利用する柔軟なメール配送システム", 情報処理学会データベースシステム研究報告 Pages:123-128 July 2001

[2] Kim Hanki, Sang-Gyu Shin, Motomichi Toyama. "A Rule-Based Mailing System for an Organization", International Workshop on INformation Processing over Evolving Networks, June 2006

[3] 原田 哲志, 慎 祥揆, 遠山 元道."RMX における電子メール送受信範囲管理方式の提案", DBWS2007

[4] 青山 陽亮, 遠山 元道."RMX におけるポリモルフィックルールとメール本文編集機能の導入", DEIM2010

[5] 北園 達也, 青山 陽亮, 遠山 元道." RMX における関数形式アドレスおよびデバッグ支援機能の実装", DEIM2011

[6] 小船井 寛, 青山 陽亮, 北園 達也, 遠山 元道. " RMX におけるルール軍管理機構・エイリアス機構等の実装", DEIM2012