

## 名宛入りメール送信プログラムの一例

鹿子木 幹 雄

(受付 2002年10月3日)

本学情報センターの OS は UNIX である。UNIX は本来、LINUX にも展開していく自由な OS であり、システム内部のディレクトリ構造、インターネット接続、Eメール機能の公開性は高い。本学の“ns1”ホストでも、ある時期までは“slog”，“tlog”などの利用によって、他者の Eメール送受信の状況をトレースすることさえ可能だった。この自由度は、その後諸般の事情でかなり制限され、失われてしまったが、自由な OS つまり、可塑性 (flexibility) の高いシステム環境は変わっていない。内部コマンドの使用法説明 (“man”), OS への接続 (“who”), OS への接続ログの照会 (“last”), ログイン名の照会 (“finger”), システム/他ユーザーのディレクトリへの移動 (“cd”) などは、制限されていない。内部エディタ “vi”, 定型編集言語 “awk” の標準装備にも、変更はない。私は、担当しているコンピュータ利用授業で、この “vi”, “awk” の 2 つのツールの利用によって、各自の Eメール環境を “らく” にし、キーボードという必要悪との “つきあい” を最小限にするよう言いつづけているのだが、履修学生の理解はなかなか得られない<sup>1)</sup>。

### I. IT 時代のプログラミング

この表現自体が時代錯誤と言われかねないぐらい、プログラミングについての関心は低くなっている。学生は “WORD” でレポート・論文などを書き、携帯電話でメールを送受し、できれば自前のホームページを持ちたいというような段階にとどまっている。Eメールについても、情報センターの UNIX 自体のメール環境を利用せず、専ら “Al-Mail” ソフトを使っている。彼らが携帯電話で交換しているのは、短文のチャットメールで、ゼミ・授業・サークルなどの相互連絡には、“Al-Mail” や “Outlook Express” などの汎用ソフトを利用しているのである。「今さら プログラミングなんて」というのが学生一般の意見だろう。しかし、周知のように、汎用性が高い “Outlook Express” のメールリングリストが、“コンピュータヴィール

---

1) Compaq Tru64 UNIX V5.1A (Rev. 1885). “vi” については、L. Lamb, 福崎俊博訳『vi 入門』アスキー出版局, 1992年; “awk” については、D. Dougherty, 福崎俊博訳『sed & awk プログラミング』オライリー・ジャパン, 1997年を参照。

ス”の標的になってきた。高度 IT 時代の現在こそ、私たちは、どんなメールソフトを使うかを意識する必要がある。情報センターは“コンピュータウイルス”の防護に最大限の努力をはらってくれているのだから、情報センターの各自のディレクトリーの上で、私たちが自前のメールソフトを作った場合、安全性はかなり高くなるのではないか。実際のプログラム開発は、このことを目標にしたのではなく、文末に記すように、長期間の種々の偶然的な事情によるものであるが\*)、結果的に、“ウイルス防護法”のための 1 提案にもなったのではないか。これが、この研究ノートを発表する理由である。諸賢のご批判、ご助言をえたい。

---

\*) 筆者が学内で複数の人にメール送信を始めたのは1997年10月だが、通信数がふえるに従って、送信方法も手動から自動化の模索、そしてプログラム化へと変わってきた。

- [1] 最初は JUNIX のメール用の標準ドットファイルである“mailrc”を利用した。このファイルの 1 行の長さには制限があるらしい (511バイト?) が、ゼミナール学生、1 学部の教員程度の受信者をグルーピングするのに不便はなかったので、“mail group\_A< message”のコマンドで、グループ数だけしばらく手動送信をした。その後“awk”を利用して、この送信をプログラム化し、ずっとその送信をしてきた。
- [2] 2000年だったのだろうか。この方式の場合、いわゆる「カーボンコピー」(Cc:)と同じに、同一グループの他の人たちのログイン名が並列表示され、受信者各人のプライバシーがなくなっている、との注意を受けた。全員の通信にこの「並列表示」の名宛てがつくので、プロバイダーへの負荷も大きくなる。事務局局メールの中には 現在もこの方式のものがある (問題部局の研修なども考えなくてはならないだろう)。
- [3] 注意してくれた人は「ブラインドカーボンコピー」(Bcc:)を提案した。この送信は、通信文の中に“~b group\_A”の 1 行を書いておき“mail kanokogi < message”のコマンドを実行することで実現できたが、その“awk”プログラム化ができず、ふたたび手動送信に戻らざるをえなくなってしまった。受信側のプライバシー保全と手動作業との“はざま”に立ちつくしていたのである。
- [4] そのとき、筆者の通信を 無用な“流布物”ないし“配信物”と見る口頭注意 (2001.4.6, 2001.4.18), 文書注意 (2001.8.9, 2001.12.25), 関連申立 (平成13年(モ)第714号訴訟記録閲覧制限事件)があった。そこで、市販のワープロ (ソフト) の「差しこみ印刷」類似のプログラムを作り、受信者側に「私信」の意識を喚起し、「自分は不要」などの意思表示などを待てるようにすれば、複数であっても、「私信」にできると考えた。
- [5] 結果的に メーリングリストを手作りし、その要素の氏名 (等!) などで、送信文を編集し、リストのアドレスに、1 通ずつを“Mail logname < message”の内部命令で送信するようにした。これが本プログラムの基本構成である。ここでのメーリングリストの要素数、配列はまったく自由なので、筆者は“virus free”なメーリングプログラムの新提案)になったのではないかと考えている。
- [6] もちろん 個別性の外見をもつ印刷プログラムは「差しこみ印刷」として汎用されている。しかし“量産型”メールで筆者が今までに受信したメールの中に、この方式のものはなかった。たしかに、普通は“量産型”の通信に“個別性”を与える必要などはない。ともあれこの「差しこみ」型、つまり「定型編集」型のメーリングプログラムは筆者の特殊な状況の副産物なのである。
- [7] なお 筆者は1997年11月以降、英文メールも送信しているが「不要」とする連絡はなかった。筆者に正確な英語が書けているはずがないので、今でも「分かる?」と聞いている。すると「分かる。ありがとう」という美辞だけが帰ってくる。本学公文書の言語環境の方に問題があるのだろう。教務部以外の部局も、最小限すべての文書の見出しに英語を併記し 学内をさらに「国際化」する必要があるのではないかと思う。

## II. 発信用メールソフト “email”

UNIX 上の定型編集言語 “awk” は「同時通訳型」が基本で、60年代の “BASIC” に似ている<sup>2)</sup>。この型のコンピュータ言語の特徴は、「読める」という点にあり、当然、自由に修正・改良してゆけるといいう長所がある。このプログラムは、発信専用であり、送受双方向の “完全なソフト” にはなっていないが、それも可能であろう。「可視型」プログラムは、いつでも改良可能 (flexible) なのである。

プログラムの基本構造は、つぎのように単純な 3 つのループからなっている<sup>3)</sup>。

- [1] メーリングリストを 1 行ずつ読み、そのデータを読みこむ外枠のループ  
(ここに、ループが正常に作動し、だれのテキスト文を処理しているかを確認するための “モニター” をつけてある),
- [2] つぎに、送信テキストのメール表題 (Subject) の部分で新しい “dummy” ファイルの編集処理に入り (“print > file”), つづいて個人別の項目を処理した上で (“print name >> file”), 送信テキスト本文を原文どおり出力していく (“print \$0 >>file”) 中枠のループ,
- [3] 編集ずみの個別用のテキスト “dummy” を送信する一番内枠のループ  
(送信行の文頭を “#” で注釈行にしてしまい、不作動にすれば、プログラムをテストできる。その場合、メーリングリストの最後の人へのテキストが “dummy” に入っているのので、その内容を “vi” エディタで確認できる。この暫定ファイルは、つぎのメールの送信時に上書きされるから、無駄ではない)。

```
awk '          # filename: "email" (2002.09.24)
BEGIN {
list = "semi"; text = "letter"; file = "dummy"
system("mv /usr/spool/mail/kanokogi Kanokogi")      # 受信リストの待避
#1 リストの読み込み
  while ( (getline < list) > 0 && $0 != "" )
    { FS = "¥t"
      split($1, l, " "); logname = l[1]              # 末尾空白の除去
      z = split($2, n, " "); name = n[1] " " n[z]    # "姓_名"への書換え
```

- 2) Interpreter 言語。compile して高速化もできるようだが、私のように 2 ~ 3 桁程度の人たちへの通信に使う場合、数十秒で実行が終わるので、その必要を感じていない。
- 3) このプログラムを “FTP” 等のソフトで UNIX の各自のディレクトリにコピーした後、UNIX のプロンプトに “chmod 700 email” と入力するだけで実行ファイルにできる。

```

    printf logname " = " name " " " # 送信モニター
#2 メール文の仮ファイル(dummy)への書換え
    while ( (getline < text) > 0 )
    { FS = " "
      if ( $1 == "~s" ) print $0 > file # Subject行の処理
      else if ( $0 == "~くん" )
        print name " <くん" >> file # 名宛の書込み
      else if ( $1 != "~s" && $0 != "~くん" )
        print $0 >> file # 通信本文のコピー
      close(file) }
    close(text)
#3 メールの送信
    system("Mail " logname " < " file ) # 仮ファイルの送信
    close(file) }
    close(list) # リスト読込みの終了
    print ""
    system("mv Kanokogi /usr/spool/mail/kanokogi") #受信リストの復帰
} '

```

### III. メーリングリストの一例 “semi”

連動させるメーリングリストは、まったく自由に設計できる。もっとも単純なものは、つぎのような UNIX の登録ユーザー名 (logname) と氏名だけの 2 項目 (field) タイプであろうが、所属・住所・コードなどの多項目であっても構わない。項目の数も順序も自由に設定できるので、既成のソフトウェアを標的にするコンピュータウイルスの侵入を結果的にむずかしくできる。

ただ、メーリングリストを見やすくし、各項目の内部に空白 (space), “.-/#\*” などの記号も自由に使いたいというような場合、項目 (field) 間の区分記号 (separator) にタブ (Tab = tabulator) を使うのがよいと思う。また、メールアドレス (logname) のように長短差がいちじるしい場合などには、末尾に空白 (space) を加えておき、プログラムの方で split 関数によりそれを除去するというような “小わざ” も必要になる<sup>4)</sup>。

4) Tab は標準の 8 バイト。リストを見やすくするため、短い項目には Tab または空白を加えるなどの処理をする。その項目名は、\$1, \$2, …, \$(NF-1), \$NF 等となる。

ログイン名(Tab)氏 名

ykamed03→ 亀田 行男<sup>5)</sup>

yturu04 → 鶴 由紀子

tkan05 → 環 子寛

(7バイト以下のログイン名末尾には空白を加え、視認性を改善している例)

#### IV. 送信メール文の例——“letter”

~s コンパのお知らせ

2002年10月16日

~くん

コンパ係 波奈ちゃん

( 通 信 文 )

[受信例] Subject: “コンパのお知らせ”

2002年10月16日

亀田 行男 くん

コンパ係 波奈ちゃん

ゼミ・コンパの日取りが決まった。11月09日(土)18時からです。場所はアストラム本通り駅ちかくの南北飯店。会費は2000円だけど、チコク15分ごとにバッキン500円取るから、だれかさんなんか、とくに注意しろよ。バッキンは2次会で使います。

(部分が定型編集されていて、受信者に親密な印象を与えられる例)

[補] 既存のプログラムに名宛の編集機能を加えたただけだが、“Mail logname < file”の“Mail”を受信リストへの単独表示命令として誤処理し、その後の“logname < file”部分を受信リストの注目メールへの入力エラーとして処理してしまい、最後にはその注目メールを削除する——という思わぬトラブルに悩まされた。筆者の能力では、受信リストの一時待避という対症療法に訴えるほかなかった。末筆ながら、故田辺教授ならびに関係各位のご好意に感謝する。(2002.10.31)

- 5) 私は、このような6文字方式の氏名表示を常用しているが、今までに不便を感じるような氏名に出会ったことはない。区分文字には2バイトの空白文字を使っている。送信時の名宛用に“亀田 行男”のように変換するには split 関数が返す要素の数(“z”)を利用している(z = split (name, n, " ")。この場合, “z”には3が返され, “n[1]”が“亀田”, “n[z]”が“行男”になる。

## Research Note

### Program Sample: Mailing with Addresses, etc. by using a pattern editing language “awk”

Mikio Kanokogi

(Received on October 3, 2002)

The operating system of the Information Center HSU is Compaq Tru64 UNIX V5.1A (Rev. 1885). UNIX is thoroughly opened system, from which developed one of the most remarkable OSs is LINUX. The directory structure, connection function to the Internet and accordingly e-mail function in UNIX are opened. Formerly, the users in “ns1” host could trace the sending results by using “slog” or “tlog” easily. Though this free nature was restricted in some degree mainly for protection of the system, its essential flexible character as “free OS” remains unchanged. For example, replies for manual asking (“man”), users check at the time (“who”), connection log in the system (“last”), login-name ascertainment (“finger”) and directory changing function to anywhere (“cd”) are very useful. Naturally equipment situation with “vi”-editor and “awk”, pattern matching programable editor is not altered. I recommend the students in my computer classes always to make use of these languages in order to build up their own more comfortable environment, but in vain.

#### **I. Programming in Information Technology Age**

This expression for itself might sound anachronistically. The students around me seem to remain to use the “WORD” for reporting papers and their mobile phones for e-mails. Few of them can wish to obtain (perhaps by “Java”) programming skill for their own homepages. They are almost poor utilizer of the mailing environment in UNIX itself. The mobile phones can convey short chatterings merely, not some long informations within the classes, seminars or clubs. They manipulate “Al-Mail” or “Outlook Express” softwares easily for their purpose. Their opinion might be mostly negative against programming. Though they do not believe the necessity of hand-made programs, such popularized softwares can be one of the most accessible targets of the “virus”. In the IT age, we should have self-made mailing programs. This is the reason why I contribute this report.

## II. Mailing Program for Mail-Sending, “Email”

The “pattern matching” language “awk” is classified to “interpreting” one, similar to the “BASIC”, popular in 60s, also with its lower speed. In reality, such an old-fashioned computer language has merit, easy to be read. It can be also freely reedited or improved. The program consists of three recursive loops:

- [1] Outer loop reads your mailing list line by line  
(In order to monitor this action, names or other necessary items, under editing are shown in the display),
- [2] Middle loop treats “~s” (subject) line, at first, in order to produce each new temporal file, “dummy” after that it edits personal line(s) and the text in attributing way, and
- [3] Inner loop sends each edited “dummy” file to the receiver  
(For testing, you can write “#” — remark line symbol — at the head of line “system(“Mail “ logname “ < “ file )”)<sup>1)</sup>.

```

awk '          # filename: "Email" (24.09.2002)
BEGIN {
list = "group"; text = "Letter"; file = "Dummy"
system("mv /usr/spool/mail/kanokogi Kanokogi")      # mail table escape
#1 read up the list
  while ( (getline < list) > 0 && $0 != "" )
    { FS = "¥t"
      split($1, l, " "); logname = l[1]              # trim spaces at tail
      name = $(NF-1); belonging = $NF               # field no. from right
      printf belonging " / " name " "              # monitor
#2 text editing to temporal file
  while ( (getline < text) > 0 )
    { FS = " "
      if ( $1 == "~s" ) print $0 > file             # "Subject" line
      else if ( $0 == "organ" )
        print belonging >> file                    # institute, etc.
      else if ( $0 == "Dear" )
        print "Dear " name ", " >> file           # addressing

```

1) In order to have a copied — perhaps by using “FTP”, etc. — program, you need to order at the UNIX prompt “chmod 700 Email”.



#### IV. Sample of e-Mail Letter: "Letter"

~s New Commission?<sup>3)</sup>

16. 10. 2002

organ

Dear

Secretariat of Japan Peace Association

[ Sample: Letter to a Receiver ] Subject: "New Commission?"

Hiroshima City University

Dear Prof. Saburo Okamura,

Secretariat of Japan Peace Association

Re: Additional Commission of "Deport Issue"

In the coming meeting on Sat 09.11.2002 at HSU, a new commission of "Deport Issue" seems necessary for the circumstances. Please, talk together about it urgently. We are waiting for your reply or suggestion. Thank you.

( The meshed lines are the result of the pattern matching editing to give a kind of friendly or private impression of this e-mail)

- \*) It was on 28.11.1997 that the author had begun to send "Letter"-series in English by his poor ability. Mailing way has been changed from manual to automatic method according to the number of letters.
- [1] At first, the author had used dot-file for mailing, ".mailrc" file in UNIX, that seems to have some limitation of line-length, 512 bytes or so. Nevertheless this limitation did not mean difficulties to manipulate such scale of number as seminar-students or faculty-staffs. He worked as many times as the number of the groups and designed "awk" program of the same action afterwards.
- [2] It was perhaps in 2000 that the author was warned by a colleague that the program showed the names of the same group inevitably, just like as "Cc:", carbon-copy method, to infringe privacies in the group. — This method is still now applied in some e-mails from the offices HSU, that may mean comparably heavy load upon the whole system at the same time.

3) Escape subcommand for Subject.

Program Sample: Mailing with Addresses, etc. by using a pattern editing language “awk”

- [3] The kind staff gave hint at that time to turn to “Bcc:”, blind carbon-copy way. The author realized that in his “awk” program, by adding a new line “b group\_A” and send mails by command-line “Mail author <message” for only one group. But he could not write a right program to send letters for plural “Bcc:”-groups and return again in manual works.
- [4] Then, an urgent necessity to get rid of this tedious work unexpectedly: the author was advised restrain himself from “spreading” mails for many unlimited persons verbally (on 6.4.2001, 18.4.2001) or by notes (9.8.2001, 25.12.2001), or by a related issue (Heisei13-Mo-714 Regulation-Appeal of a Official Document for the Hiroshima Regional Court). The author made new model of program with private addresses, to emphasize impression between “personal” connections, in order to react easily for “unnecessary”, etc.
- [5] As a result, the author could open gate for new free mailing-lists with unlimited number of items and/or their order. In the “awk” program the text will be reedited one by one in its specified line(s) and the new text, thus reedited, will be sent by the program line “Mail logname < message”. These are the main logic of this program. As the designing of mailinglist is quite arbitrary, this program can afford “virus-free” environment for the users.
- [6] Though “Individual Data Inserting”-typed printing softwares are quite popular, the author has never received such “personal” informations from the “mass-information” senders. There might be no needs to give “personal” outlook in the “mass-informations”. Anyway, trial of introducing of “Individual Data Inserting”-typed concept into the mailing program by the language “awk” is a “byproduct” of his peculiar situation in HSU.
- [7] By the way, the authour considers that the reason why he has never accepted “unnecessary” replies from non-Japanese colleagues might not be the high quality of his informations in “Letter”-series, but merely the lack of English official informations in HSU. He would like to recommend the sections of the Office to offer bilingual informations at least in the titles of their official mailings.

[Supplement] This program is a revised one from a already running one, by additioning addressing function with section or belonging. But the author was annoied with a trouble, where the system interpreted “Mail” command in “Mail logname < file” as an independent one, that lead the following texts “logname < file” as illegal orders and as a result, killed the conferred mails automatically. He asks for your kind advices for such strange action. He himself could only let the mail table to escape during the procedure. He might perhaps be allowed to express his deep gratitudes to late Prof. Taku Tanabe and to you all. (2002.10.31)