

TR-I-0084

電子メール及びニュースに関する管理者用マニュアル

*The E-Mail & News Administrator's Manual*

野垣内出\*      丸山活輝\*      北研二\*  
Izuru NOGAI TO\*    Katsuteru MARUYAMA\*    Kenji KITA\*  
青井周一\*\*  
Shuichi AOI\*\*

1989.5.15

ATR 自動翻訳電話研究所

version 1.0

### 概要

ATR自動翻訳電話研究所では、研究環境の一環として、電子メール及びニュースのサービスを行っている。既にユーザのための一般的なマニュアルは報告した。ここでは特にシステムの概要及び各プログラムの機能説明と、インストールのために必要な環境設定やインストール手順についても記載している。これらはシステム管理者及びメール管理者だけでなく、一般ユーザが個人環境を整備する上での資料となるであろう。

\*ATR 自動翻訳電話研究所  
ATR Interpreting Telephony Research Laboratories

\*\* 日進ソフトウェア(株)  
NISSIN Software Company

©ATR 自動翻訳電話研究所 1989  
ATR Interpreting Telephony Research Laboratories

# 目次

1. ネットワークの管理	1
2. ニュースシステム	
2.1 システムの概要	9
2.2 各プログラムの説明	12
2.3 データ構造	30
2.4 インストレーションガイド	38
3. メールシステム	
3.1 システムの概要	52
3.2 各プログラムの説明	54
3.3 データ構造	61
3.4 インストレーションガイド	67
4. uucpシステム	
4.1 システムの概要	74
4.2 各プログラムの説明	76
4.3 データ構造	84
4.4 インストレーションガイド	91
5. ATR内のネットワーク環境	103
5.1 ゲートウェイマシンの移行	103
5.2 ネットワークアドレスの変更	105
6. 参考文献	109

## 付録

- A ニュース・インストラクション・ガイド(USENET版)
- B ニュース・インストラクション・ガイド(日本語対応版)
- C JUNETのアドレス表現について
- D sendmail.cf作成システム(mailconf6.2J)
- E sendmail.cf作成システム(mailconf6.3J)
- F uucp4.3インストラクション・ガイド(USENET版)
- G JUNET内のサイトマップ

## はじめに

コンピュータ・ネットワークの発展とともに、各ホスト間を結ぶためのシステムが数多く開発されてきた。特に一般的なものとしては、電子メール・電子掲示板(ニュース)システムなどが有名である。ATRは日本の学術ネットワークであるJUNETにも参加しており、JUNET内で提供されているサービスを受けるためには、自分たちでマシン間ファイル転送(UUCP)・電子メール・電子掲示板(ニュース)の各システムを導入して、これらの管理・運用を行わなければならない。システムのインストールの手助けとなるインストール・マニュアルが余り整備されていないため、初めからシステムの構築を行うためには、unixに精通しているだけでなく、これらのシステムにも精通していなければ、システムの導入に多大な時間が必要となろう。ATR自動翻訳電話研究所及びATR全体としても、安定したシステム運用が行われるようになり、今後のシステムの拡張やメール管理者の育成・教育の手助けとなるように、今まで蓄積したノウハウをまとめ上げることとした。ここでは、インストール方法の他に、電子メール・ニュースの計算機内部データについてもできる限り詳しく説明した。このため、個人環境の整備、例えば、Emacsからmailを読むために必要な個人のライブラリーの拡張などの参考資料ともなるように心掛けた。

## 当りポート作成にあたって

JUNETで流れている多くのニュースの中から、パブリックドメインすなわち、JUNET全体の財産であると考えられるもの、特に重要なものを本文43ページから45ページにかけて引用した。また、文章化していなかったもので、パブリックドメインのソフトウェア・キットに付属のインストール・マニュアルを付録として収録した。

## 1. ネットワークの管理

自分の管理しているマシンに1日1回はログインして、次の作業を行う。1.1は外部接続を行っているゲートウェイ・マシンのみに必要で、他のマシンでは1.2以降からの作業が必要である。

### 1.1 外部との接続(uusnapの起動)の確認。 ゲートウェイ・マシンのみ必要。

```
%uusnap
Kddlab 12 Cmd_ _ _ _20 Data Login Failed:
Nttlab   Cmd_ _ _ _
```

あまりにも接続が悪い場合、login failedなどで送りのCmdが10通以上溜っている場合、強制的に掃き出す。

```
%/usr/lib/uucp/uucico -x7 -r1 -skddlab
                               (サイト名)
```

エラーが発生した場合、そのエラーメッセージによって以下の処理をとる。

Read,Open Error---->送るファイルのモードを変える。

### 1.2 内部のサブドメインとの接続の確認。

内部のデータ転送にはメールシステムが使われるため、メールが正常に動いているか調べる。

```
%ps -ax|grep sendmail
```

メールキューの中を見て、掃き出されないまま長時間経っているものを調べる。

```
%mailq
```

a) 明らかに他のサブドメインがシステムダウンしている場合  
(今日はatr-spがダウンしている)

```
Mail Queue(1 request)
--QID----Size-----Q-Time-----Sender/Recipient-
AA20670    547   Mon Mar 7 11:56   abc@speech2.cs.cmu.edu
                                           (Deferred:Connection timed out)
                                           post@atr-sp.atr.junet
```

uusnapでKddlabかNttlabからデータを受けていないことを確認してから、mailqでダウンしていないサイトへのmailの発送が終了していることを確認し、mqueueにあるすべての転送すべきファイルをomqueueに移す。

```
%mkdir /usr/spool/omqueue
%mv /usr/spool/mqueue/*AA* /usr/spool/omqueue/.
```

その後、atr-spが立ち上がったら、強制的にメールをすべて掃き出す。この時、メールの日数が3日(72時間)以上前に送られてきたものであっても、ファイル更新を行えば送り主に返送されることはない。

```
%/usr/lib/sendmail -oQ/usr/spool/omqueue -q -v
```

b) メールが相手側に受入れられない場合

```
Mail Queue(1 request)
---QID---Size-----Q-Time-----Sender/Recipient-
AA20670      547   Mon Mar 7 11:56   abc@speech2.cs.cmu.edu
                                     (Deferred: Refuse to....)
                                     post@atr-sp.atr.junet
```

\*回線のチェック      --->      ケーブルの線がぬけている、etc.....

c) sendmailがダウンした場合

sendmailを再度立ちあげ、メールが送られたことを確認

```
%/usr/lib/sendmail -bd -q10m -om
```

この段階でメールがmqueueに溜ったまま送れない場合は、強制的に掃き出す。

```
%/usr/lib/sendmail -q -v
```

d) mailqで見たときに、フラグ(QID番号の後に\*)が立っている場合

```
%mailq
```

```
---QID---Size-----Q-Time-----Sender/Recipient-
AA20671*      1023   Mon Mar 7 12:56   aoi@atr-la
                                     dump@atr-1n.atr.junet
```

この場合、上のsendmail-q-vを起動しても、強制的に掃き出すことができない。なぜなら、sendmailが起動されている途中でマシンがダウンするなどして、sendmailのプロセスが異常終了した場合にロック・ファイルが残ることがある。このロック・ファイルが残っていると、mqueueに溜まっているメールを強制的に掃き出そうとしても、ロックが掛かっている状態と認識してメールを掃き出すことはできない。

(処置)      ロック・ファイルを消す。(システムが立ち上がる時は、/etc/rc.localで設定されている。)

```
%rm /usr/spool/mqueue/1fAA20671
%/usr/lib/sendmail -q -v
```

### 1.3 メールの管理

#### 1) ユーザの管理

ちょっとしたアドレスの誤りでも、メールは届かない。特に、外部に出したメールのアドレスの指定がおかしい場合、他のサイトを渡り歩いてエラーメールとして戻ってくる。その間の通信料が多大にかかっていることを認識させる。そのために、ユーザに対しての教育を充分に行わなければいけない。そのエラーメールはポスト宛に届くので、送り主にその旨伝える。人によってはメールを読まずに溜りっぱなしになっている場合があり、スプール領域は通常バックアップされないので、システムに異常が起こった時にこのエリアを壊してメールが紛失する可能性がある。また、相当量のメールが溜まるとスプールがパンクする恐れもあるので、常時監視を行い、余りに多くのメールを溜めているユーザには注意を促す必要がある。

```
%ls -l /usr/spool/mail
```

```
total902
-rw----- 1 aki          0 Mar 10 09:05 aki
-rw----- 1 ikue       183857 Mar 9 13:08 ikue
      :
      :
```

lsコマンドでスプールに溜まっているメールのサイズを調べた結果、ikueというユーザが183Kbyteものメールを読まずに放置していることを示している。

### 1.4 ニュースの管理

#### 1) newsのディスク領域の管理

1日に届くnewsの量がだいたい2MBぐらいあるために、/usr/spool/newsの領域のとれる大きさによってある程度まで溜ったら、消去しなければならない。そのために、expireコマンドがある。

```
%/usr/lib/news/expire -e90
```

これは、90日経った古いメールを消去するという意味である。expireコマンドを手でそのつど起動するのは面倒なので、/usr/lib/crontabに普通指定する。atr-la内では、以下の様に記述されている。

```
45 1,11 * * * /usr/lib/news/expire -e90 -a
```

1:45と11:45の1日2回expireコマンドを起動して、3ヶ月経ったものは/usr/spool/oldnewsに移しかえている。ちなみに/usr/spool/oldnewsにリンクがはられており、/postoffice/oldnewsの領域を使用している。

しかし、この/postoffice/oldnewsもディスク容量に制限があるために、ディスクがフル状態になる前にバックアップを取り、すべてクリアしてしまう。最近この作業は3-4週間に1回行っている。

2) ニュースを定起動で送信する。

```
11,11 13 15 17 19 21 * * * /usr/lib/news/sendbatch -c kddlalab
```

これも crontab 内に設定しておく。kddlalab に 11 時から 2 時間おきにニュースを送る。sendbatch のコマンドの中から compress を呼んで、ニュースをブロック単位に分けてからスプールに溜った順番に掃き出している。(この設定は、ゲートウェイ・マシンだけ必要)

3) ニュースグループの追加、削除

新しいニュースグループが出来た場合、自動的に更新される。ニュースグループが削除された場合には、ポスト宛にどのグループが削除され、またどのグループが追加になったかをメールで送られてくる。その中にニュースグループの追加及び削除するスクリプトも入っている。以下のファイルのコマンドの部分だけを取り出して実行する。

[例]

```
The following newsgroups are not valid and should be removed.
```

```
fj.video
fj.works
fj.works.apollo
```

```
You can do this by executing the command:
```

```
/usr/lib/news/rmggroup fj.video fj.works fj.works.apollo
```

```
The following newsgroups were missing and should be added.
```

```
fj.kanakan.misc
fj.kanakan.wnn
fj.rec.av
```

```
You can do this by executing the command(s):
```

```
/usr/lib/news/inews -C fj.kanakan.misc < /dev/null
/usr/lib/news/inews -C fj.kanakan.wnn < /dev/null
/usr/lib/news/inews -C fj.rec.av < /dev/null
```

```
The following newsgroups are moderated and not marked so.
```

```
fj.announce
fj.announce.misc
```

```
You can correct this by executing the command(s):
```

```
/usr/lib/news/inews -C fj.announce moderated < /dev/null
```

ニュースグループが追加後の確認事項

- a) active ファイルにグループ名が記述されること。
- b) 記事を格納するディレクトリを作成すること。



\*) 留意点

この動作がうまくいかないまま確認しないで放っておくと、各決められたディレクトリに入るべきものが、/usr/spool/news/junkにsaveされて、そのグループに関する記事を読むことはできない。junkの中にファイルがいっぱい溜っていた場合、うまくニュースグループの追加の処理が行われていない可能性が高いので、どのグループがおかしいか調べる。

4) ニュースの定期講読

JUNETで送られてくるニュースの中には、ポストマスターが特に知っておくべき情報があるので、定期的に講読する必要がある。

[読むべきニュースグループ]

fj.junet	JUNETに関する全般的な情報
fj.guide.admin	JUNETに関するガイド (管理者用)
fj.guide.general	(一般向け)
fj.guide.newusers	(初心者用)
fj.general	JUNETに関する一般的な議論
fj.mail	メールに関する議論
fj.news.newsite	新しい参加サイトについての情報
fj.news.config	JUNETサイトのダウン、ルート変更などについての連絡
fj.news.sa	システム管理者に役立つ情報
comp.mail.uucp	UUCPによるメールリンクに関する情報
news.sysadmin	USENETの管理者にむけた情報ときどき重要な情報 がのる
fj.news.adm	ニュース管理者にむけた情報
fj.news.b	ニュースのソフトウェアについての情報(パッチなど) がながれる

5) ニュースやメールなどのインストール

JUNETの各サイトとの互換性を保つために、最新バージョンのものをインストールする。常にニュースで送られてくるパッチ情報などに注意を払って、最新のパッチを当てるようにする。

6) ユーザーへのインフォメーション

新しいソフトウェアをインストールしたり、ネットワークに関するトラブルなどについての報告を、ニュースを使って一般ユーザに知らせる。

## 1.5 セキュリティー・チェック

### 1) パスワードの設定

通常パスワードを設定する場合、なるべく他人に悟られないようなものを設定する必要がある。実際にパスワードを入力する時に、どうしても頭の中に浮かんでくるキーワード(例えば、奥さんや婚約者の名前、生年月日等)を入力しがちであるが、これは最も見破られやすいパスワードと言われている。特に、特権ユーザがこのような設定を行うのは具の骨頂である。通常このようなパスワードは見破られやすいが、少し工夫するだけで盗まれにくいパスワードの設定ができる。例えば、奥さんの名前がrikakoなら、そのまま入力するのではなく、“I love rikako”もしくは、“mywife rikako”などブランクをパスワード中に設定すれば(ブランクの使用が認められていることが余り知られていないため)、そう簡単に盗まれる心配はない。しかし、万全には万全を期すために、定期的にパスワードの変更を行ったほうがよい。

### 2) セット・ユーザIDビットのチェック

セット・ユーザIDビットの考え方は、他のOSにはないunix独特の思想に基づいている。アメリカではこのアイデアが特許にもなっている。

メール転送の制御をしているsendmailのファイルの形態に注目してほしい。先頭から3ビット目にSというビットが立っているのがわかるだろう。ユーザ側から電子メールを送る場合、sendmailがシステム領域にデータを格納して、指定されたサイトのスプール内にメールを書き込むので、一般ユーザが行った要求を特権ユーザとして処理されなければならない。そのため、sendmailには、セット・ユーザIDビットが立てられている。

```
-rwsr-xr-x  2  root 70656 Nov  5 15:53 sendmail*
```

ここで注意してもらいたいのは、通常システムで用意されているコマンド等でセット・ユーザIDビットがたっているものは別に問題はないのだが、一般ユーザがリモートで接続されているサイトからオーナーが特権ユーザでセット・ユーザIDビットを立てたプログラム(プログラム中でsystem関数を動かす簡単なものでも)をコピーするなどして、特権ユーザとして一般ユーザがシステムを破壊することが容易に出来てしまう。そのため、1日に1回位深夜にでもセット・ユーザIDビットをチェックするシェルを動かす必要がある。

[例]

```
find / -user root -perm -4000 -exec ls -l {} \;      ¥
    | mail root
exit
```

### 3) ネットワーク環境のための設定ファイルのチェック

#### a) /.rhostsの設定

リモート間コピーの制約やリモート・ログイン時のパスワード入力の省略を行える。

[例] atr-sp の/.rhosts

```
atr-la
atr-ln
```

このような記述を行えば、atr-spからatr-la,atr-lnにどんなファイルでも転送可能になる(ただし、atr-la,atr-lnにもatr-spの記述が必要)。このファイルに余りむやみにホスト名を増やさないこと。

#### b) /etc/hosts.equivの設定

リモート・ログイン時に同じユーザであれば、パスワードの入力の省略ができる。

#### c) /etc/passwdの設定

このファイルは、各ユーザのパスワード等を記述してあり(パスワードは暗号化)、一般ユーザが直接書き込みできないようにしておく。特権ユーザのパスワードの変更や新たな特権ユーザの追加等を防ぐため。

#### d) /etc/ttysの設定

ttysファイルは、ターミナルからの制約条件を付けることができる。例えば、コンソール以外から直接特権ユーザではログインできないようにして、必ずsuコマンドで特権ユーザになるようにしておく。そうすると、システムが/usr/adm/sulogというファイルに誰がいつ特権ユーザになったかなどの情報を残しているの、もしシステム管理者でない人が特権ユーザになっていないか等のチェックができる。下記にultrix2.0のttysの記述の例を示した。ステータスのsecureは直接特権ユーザになれる端末を示し、nomodemは直接特権ユーザになれない。

[例]

```
# name getty          type          status        comments
#
console "/etc/getty e"   dw3           on secure     # console terminal
tty00   "/usr/jsy/etc/getty K" vt282        on nomodem    # direct connect tty
```

#### 4) コンピュータ・ウイルスについて

コンピュータ・ウイルスとは、ネットワーク等を通じてプログラムを流し、相手側のシステムを破壊していく。今まで述べたセキュリティのチェックが余り行われていないところなどが狙われやすい。しかし、上記のセキュリティのチェックを行っていても完全とは言いきれない。

例えば、1988年11月ぐらいにアメリカのARPANETを中心にネットワークが麻痺したのは、unixのOS側で用意されているsendmailのデバッグ・モード(bsd4.2に現存する初期のテスト用オプションで現在はセキュリティの面からバグとされている。)から各システムに潜入して混乱を巻き起こした。この他アメリカでは、マキントッシュを中心としたパソコンなどでも、数年前から発生していて社会問題にもなっている。パブリック・ドメイン(著作権は放棄しないが自由に使えるソフトウェア)と言われるプログラムがネットワークを通じて送られるようになり、巧妙にそのプログラム中にシステムを破壊するものを仕組んである。最近日本のパソコン通信などでも問題になり始めている。

元来、コンピュータ・ネットワークとは、コンピュータ間の計算機資源を効率よく共同利用することを目的として始まった。その為、多様なネットワーク機能を備えているunixが、採用されるようになった。unix上では、より円滑に効率よく通信するために、いろいろなソフトウェアが開発された。オープンなシステムである反面、セキュリティは完全とは言えず、外部から潜入されやすいと言われている。インフルエンザにも香港型・ソ連型と毎年のように違ったウイルスが発生してそれに対するワクチンの開発が追いつかないのと同じで、コンピュータ・ウイルスに対するワクチン(ウイルスを撃退するようなプログラム)の開発は立ち遅れているのが現状である。もし、ワクチンが開発されても、新たなウイルスが次々と発生するならば、それも無意味なことかもしれない。

そうすれば、もっとセキュリティを強化すればいいじゃないかと言われそうだが、そうするとより円滑に効率よく通信するという目的が損なわれないとも限らない。今の現状では、管理面の強化を計ることしか考えられない。本来のコンピュータ・ネットワークの意義を弁え、各自の良識を信じるより仕方ないのかもしれない。

## 2. ニュースシステム(news2.11version)

### 2.1 システム概要

ニュースとは、電子掲示板といわれるものである。メールは1対1のやり取りに対して、ニュースは1対複数のやり取りができる。テーマや議論によっていろいろなグループを形成し、自分の好きなものを講読でき、その記事に関する投稿などができる。

#### 2.1.1 各機能説明

ニュースシステムには、次の5つの機能がある。

##### 1) 記事受信機能

他のサイトから送られてきたニュースを、mail経由で自動的に受けとる。受け取ったニュースをスプール内に格納する。

##### 2) 記事送信機能

ホストマシンであれば、他のサイトから受け取ったニュースを下位のドメインに配布する。記事を投函した場合、ローカルなものであればローカル内だけ、それ以外のものに関してはJUNETを経由して他のサイトにも送信する。

##### 3) 記事講読機能

ニュースを読むための機能。テーマによりグループが分かれており、好きなグループの選択ができる。まず、選択したグループの古いニュースからそのヘッダー部(タイトル、投函者名など)の情報を表示する。読みとばしや、他のニュースグループのサーチや、投函者へのリプライメールなど各種の機能を持つ。

##### 4) 記事投函機能

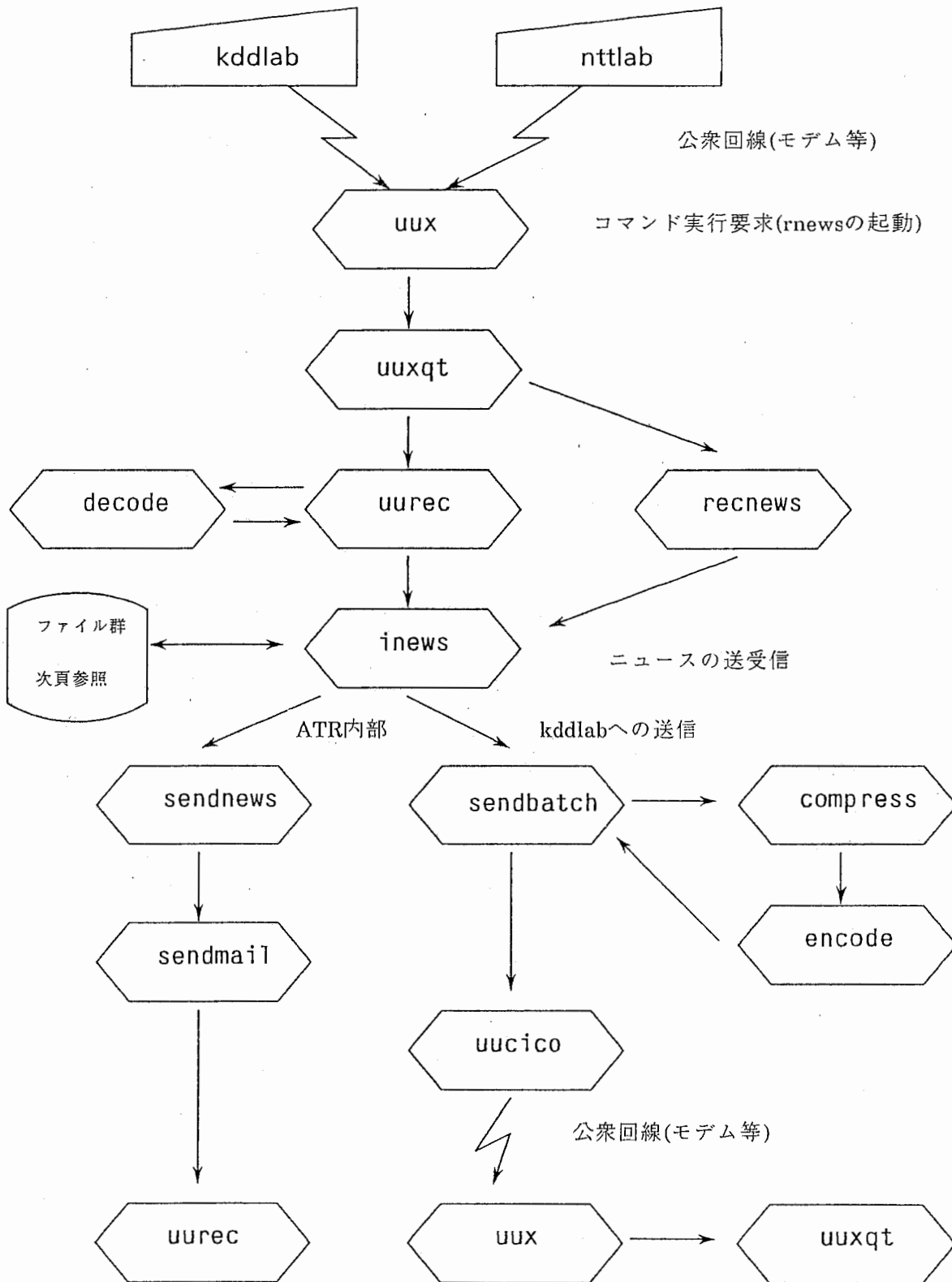
あるグループに記事を投稿するための機能。グループの指定を行い、タイトルを表示すれば、各サイトに記事が送られる。実際には、スプールに格納するための前処理として、ヘッダー部の形成(メール形式で送られるので)を自動的に行っているだけである。

##### 5) 記事削除機能

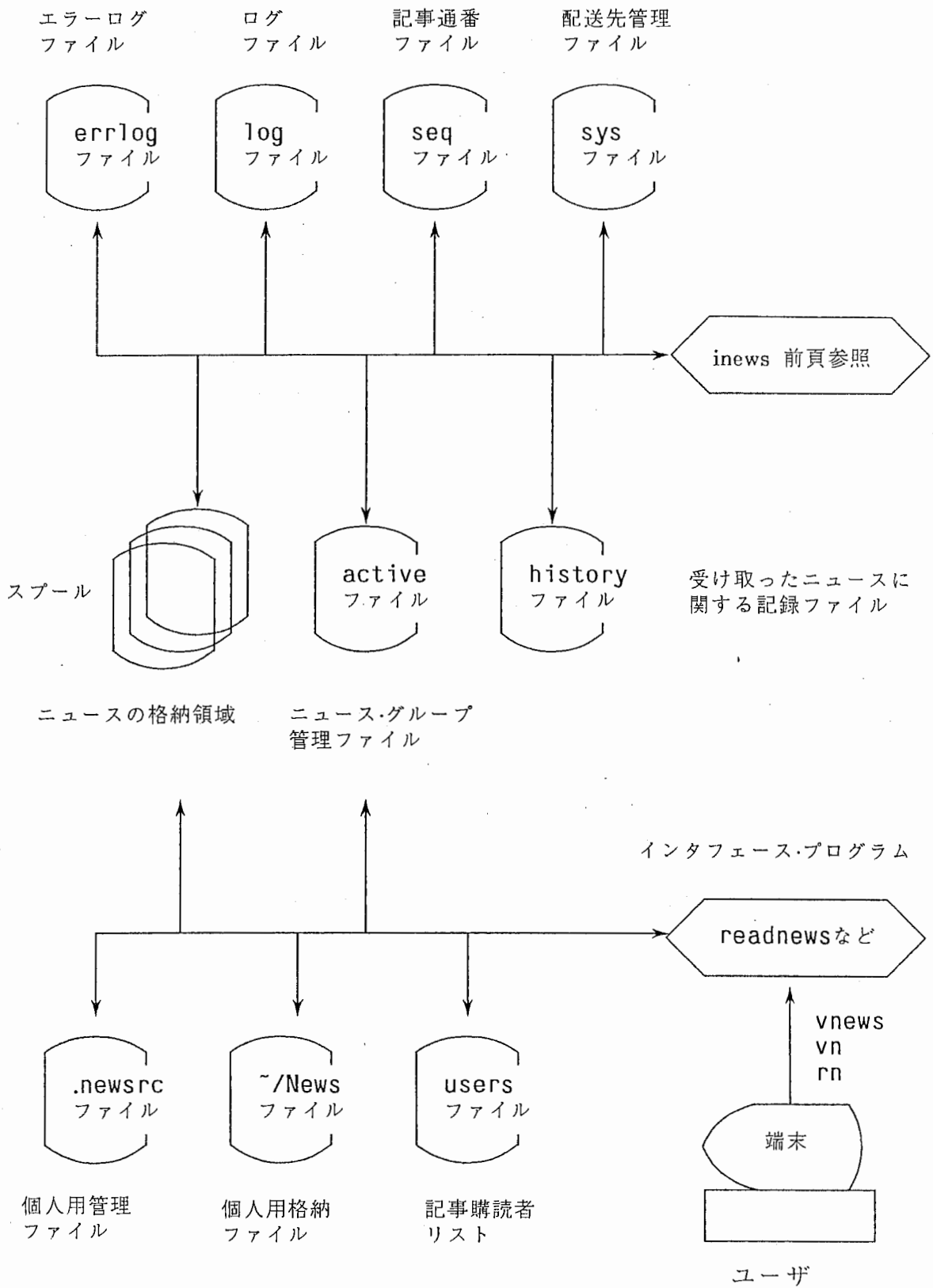
ディスクの容量には限りがあるので、記事を全てセーブすることは不可能なため、古い記事から指定された日付以降の記事を削除する。毎日最低1回以上、通常夜中に定起動で記事の削除を行う。

2.1.2 システム構成図

1) ニュース・システム側



2) ユーザ・インタフェース



## 2.2 各プログラム説明

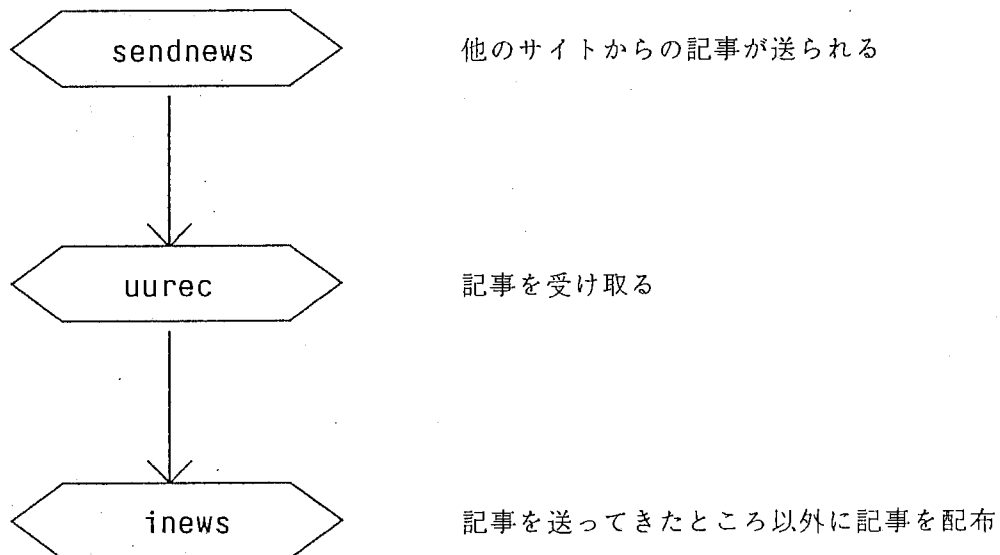
### 2.2.1 記事受信機能

#### ★処理済ニュース受信プログラム(uurec)

##### 1. 機能

ホストマシンもしくは、ローカル内で送られてきたヘッダー生成済のニュースを自動的に受け取る。

##### 2. 他のプログラムとの関連



##### 3. 入出力情報

メール形式で送られてきたニュース

##### 4. 処理内容

- 1) sendnewsにより起動される。
- 2) メール経由で送信された記事を読み込み。
- 3) デコードを行う。(暗号解読)
- 4) スプールに記事の格納と他のサイトへの記事の配布のためにinewsを起動する。

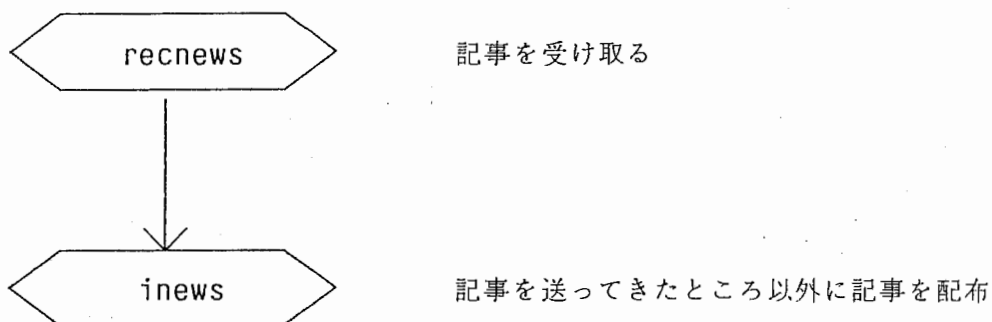


★未処理ニュース受信プログラム(recnews)

1. 機能

標準入力から記事を読み込み、ヘッダー部分の生成が不完全な部分を補いinewsを起動する。

2. 他のプログラムとの関連



3. 入出力情報

標準入力

4. 処理内容

- 1) メール経由で送信された記事を読み込み。
- 2) ヘッダー部分の欠落している部分を補う。
- 3) スプールに記事の格納と他のサイトへの記事の配布のためにinewsを起動する。

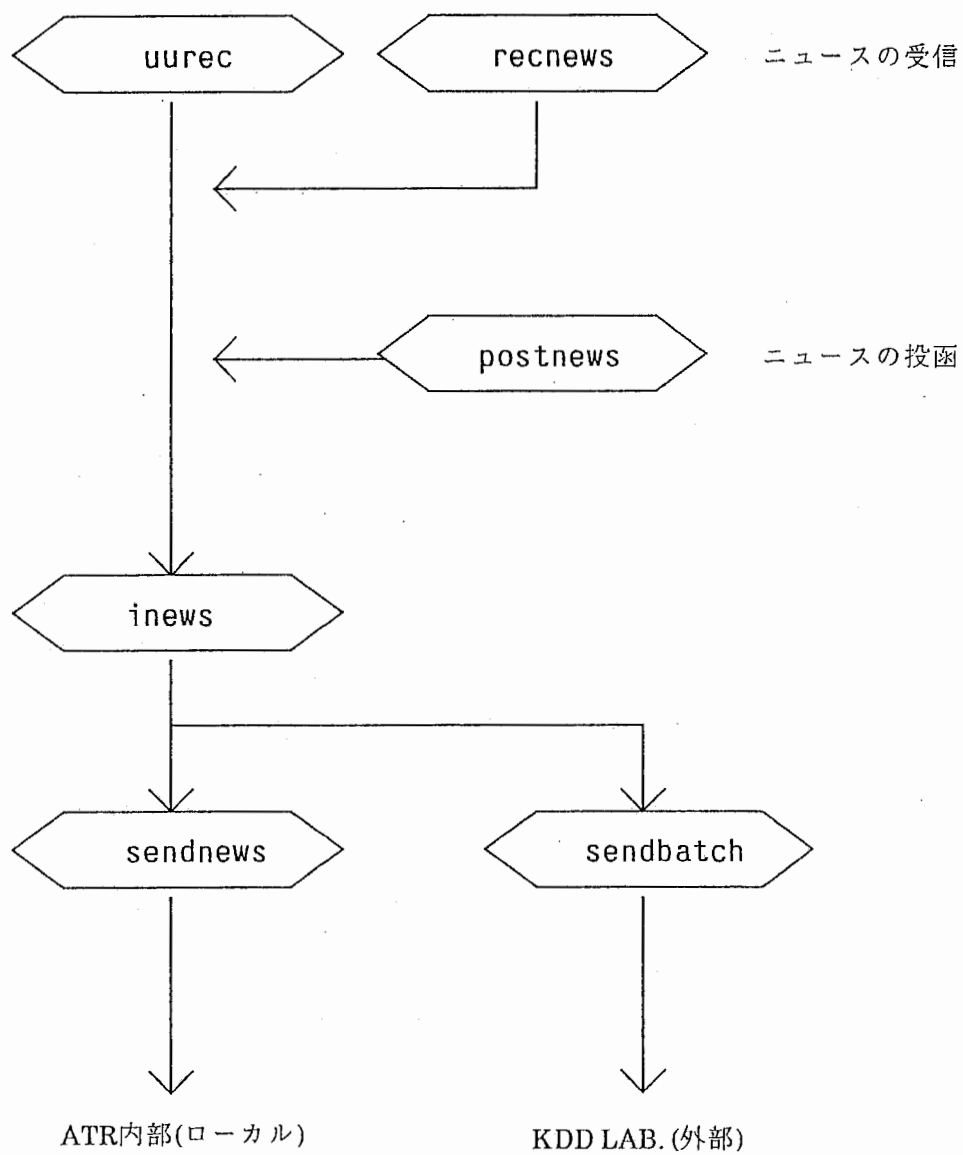
## 2.2.2 記事送信機能

### ★ニュース送受信プログラム(inews)

#### 1. 機能

新しいニュースが到着すれば、プールに格納する。そして、配送すべきサイトにニュースを順次転送する。

#### 2. 他のプログラムとの関連



### 3. 入出力情報

1) /usr/spool/news/.sys.nnn	一時的な記事
2) /usr/spool/news/.rnews	rnews -Uによる処理がまだ行われていない スパールされた記事
3) /usr/spool/news/*	記事
4) /usr/spool/news/batched/*	外部に配送すべきファイル一覧(ファイル名は uucpでニュースを送るサイト名)
5) /usr/lib/news/active	既知のニュースグループのリストおよびそれ らの最大ローカル記事番号
6) /usr/lib/news/seq	最後の記事の通番
7) /usr/lib/news/history	当マシンに現在格納されている記事の一覧
8) /usr/lib/news/sys	システムサブスクリプションリスト (配送すべきサイトの一覧)
9) /usr/lib/news/distributions	提案された配布範囲のコード名
10) /usr/lib/news/log	記事の送信記録
11) /usr/lib/news/errlog	送信失敗のエラーログ

### 4. 処理内容

- 1) activeファイルを読み込む。
- 2) expire(記事削除プログラム)が起動されている場合、一時的に送信するニュースを/usr/spool/news/.rnewsに格納する。expireが終了しだい、expireプログラム内からrnews(inewsと実体は同じ)を起動して次の3項以降の処理を行う。
- 3) 2)の場合以外、/usr/spool/news/[newsgroup]/\*にニュースを格納する。格納されるファイルはそのニュースグループが発生してからシーケンスに番号が付けられ、その番号がファイル名となる。このとき、エラーが発生した場合errlogファイルに発生原因を記述する。送られてきたニュース・グループが設定されていない場合(activeファイルに登録されていない)、/usr/spool/news/junkにそのニュースを格納する。
- 4) activeファイルの更新ニュースの到着により、最新のファイルno.の部分を書き換える。
- 5) historyファイルに送られてきたニュースの情報を書き込む。
- 6) sysファイルを読み込み、受け取ったニュースをどこに送るか、どのニュースグループを送るかsysファイルに書かれてある内容をもとにニュースを配布する。外部に配送すべきニュースは、/usr/spool/news/batched/サイト名ファイルにニュースのファイルNo.を書き込む。この時、logファイルに記事をどこに送ったかを記録しておく。
- 7) 新しいニュースグループが作成された場合、そのグループの記事が入るエリアの確保と、activeファイルにそのニュースグループを追加する。

5. 特記事項

1) オプション一覧

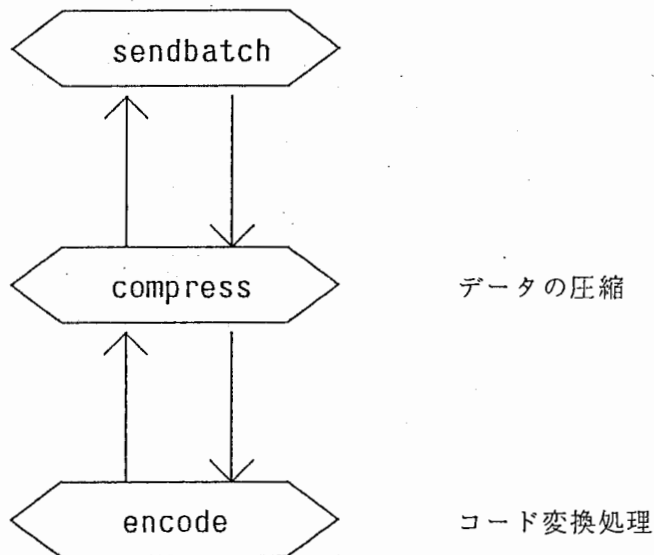
オプション	引数	内容
-c	なし	制御メッセージ送信
-C	グループ名	新しいニュースグループの作成
-d	地域名	記事の配布範囲の設定
-e	日付	任意の日付を指定
-f	なし	senderの指定、送り主の所を他人の名前に書き換えられた場合、自動的にsenderが書き込まれる(偽造記事防止)
-F	なし	References:フィールドの作成
-h	なし	ファイルが既にヘッダーと本文からなっているので、ヘッダー作成の処理を省略する
-M	なし	制約つきニュースグループの設定、ニュースを出せる人間の限定
-o	なし	デフォルトのorganization名の無視
-r	なし	デフォルトのorganization名の無視
-S	なし	rnews-Uと同じ、.rnewsに一時的に溜っているニュースを送信
-x	なし	sysファイルの指定にかかわらず、指定したサイトに記事を送らない

## ★ニュースバッチ送信プログラム(sendbatch)

### 1. 機能

記事をバッチにして送信する。通常、外部に記事を送信する場合に用いられる。

### 2.他のプログラムとの関連



### 3. 入出力情報

- 1) `/usr/spool/news/*` 記事
- 2) `/usr/spool/news/batched/サイト名` uucpを通じて外部に配送すべきファイル一覧

### 4. 処理内容

- 1) `inews`により、外部に配送すべきニュースの一覧が`/usr/spool/news/batched/[sitename]`に書き込まれているので、それらのファイルを全て一括で読み込む。
- 2) `compress`を起動して、配送すべき全てのデータをブロック単位に切り分けデータの圧縮を行う。
- 3) JUNETで標準の漢字コードはJISの7ビットキャラクタなので、`encode`を起動して8ビットキャラクタを7ビットキャラクタに変換する。
- 4) `uux`を起動して、リモート・コマンド実行要求を相手側ホストに出す。

## 5. 特記事項

### 1) オプション一覧

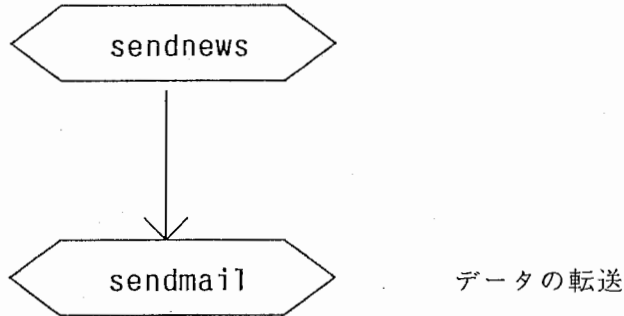
オプション	引数	内容
-bBC	なし	直接compressを起動。圧縮アルゴリズムの変更も可能
-c	なし	compressの起動
-c7	なし	記事のバッチを圧縮して、encodeの起動により7ビット キャラクタに変換する
-i	なし	Message-IDのバッチをihaveプロトコル
-o	なし	ニュースの旧バージョンのための古い形式
-s	バイト数	バッチのサイズの変更 デフォルトのバッチサイズ50,000バイト

★メール経由ニュース送信プログラム(sendnews)

1. 機能

記事をメール経由で送信する。通常、内部に記事を送信する場合に用いられる。

2. 他のプログラムとの関連



3. 入出力情報

1) /usr/spool/news/\*          記事

4. 処理内容

1) 標準入力から記事を読み込む。

2) メール形式で記事を送るために、ヘッダ部の書き換えを行う。

5. 特記事項

1) オプション一覧

オプション	引数	内容
-a	なし	ARPANET経由で記事の送信
-b	なし	Berknet経由で記事の送信
-n	なし	記事内のニュースグループを指定されたニュースグループに変更する
-N	なし	uurecによるデコーディングのための各行にプリペンドさせる
-o	なし	古い形式の記事を扱う

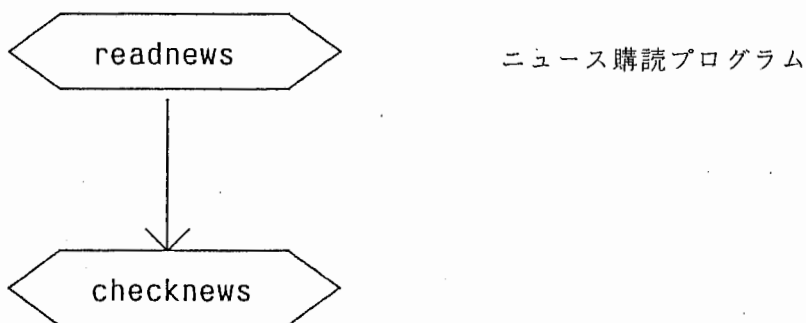
### 2.2.3 記事購読機能

#### ★ニュース更新チェックプログラム(checknews)

##### 1. 機能

新しいニュースが到着しているか調べる。そして、ユーザが未読の記事について報告する。

##### 2. 他のプログラムとの関連



##### 3. 入出力情報

- 1) activeファイル
- 2) ホームディレクトリ/.newsrcファイル

##### 4. 処理内容

readnews(ニュース購読プログラム)起動後に、以下の処理を行う。

- 1) ホームディレクトリ下の.newsrcファイルと/usr/lib/news/activeファイルを読み込む。
- 2) .newsrcファイルとactiveファイルを比較して、未読の記事を調べる。
- 3) 未読の記事があれば、ファイルno.の何番から何番までが未購読なのか、その情報をreadnews(ニュース購読プログラム)にわたす。

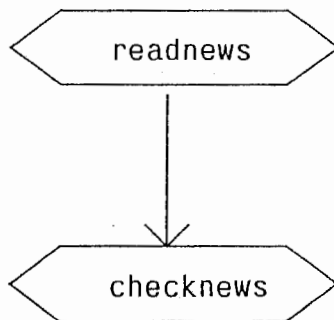


## ★ニュース講読プログラム(readnews)

### 1. 機能

ニュースを講読するためのプログラム

### 2. 他のプログラムとの関連



ニュース更新チェック

### 3. 入出力情報

- |                       |                 |
|-----------------------|-----------------|
| 1) activeファイル         | ニュース・グループ管理ファイル |
| 2) ~/.newsrcファイル      | 個人用管理ファイル       |
| 3) /usr/spool/news/*  | スプール内の全ての記事     |
| 4) /usr/lib/news/help | オプション一覧ファイル     |

### 4. 処理内容

- 1) 入力チェック
- 2) .newsrcとactiveファイルの読み込み
- 3) checknewsの起動(未読記事の報告)
- 4) ヘッダーの表示

[例]

-----  
Newsgroup:atr.general  
-----

Article 130 of 142, Mar 7 10:03.  
Subject: hotel info in kyoto  
From: saito@atr-la.atr.junet(HiroakiSaito@atr)  
(26lines)More?[ynq]

5) 選択モード処理

入力されたキーにより、次の各処理を行う。

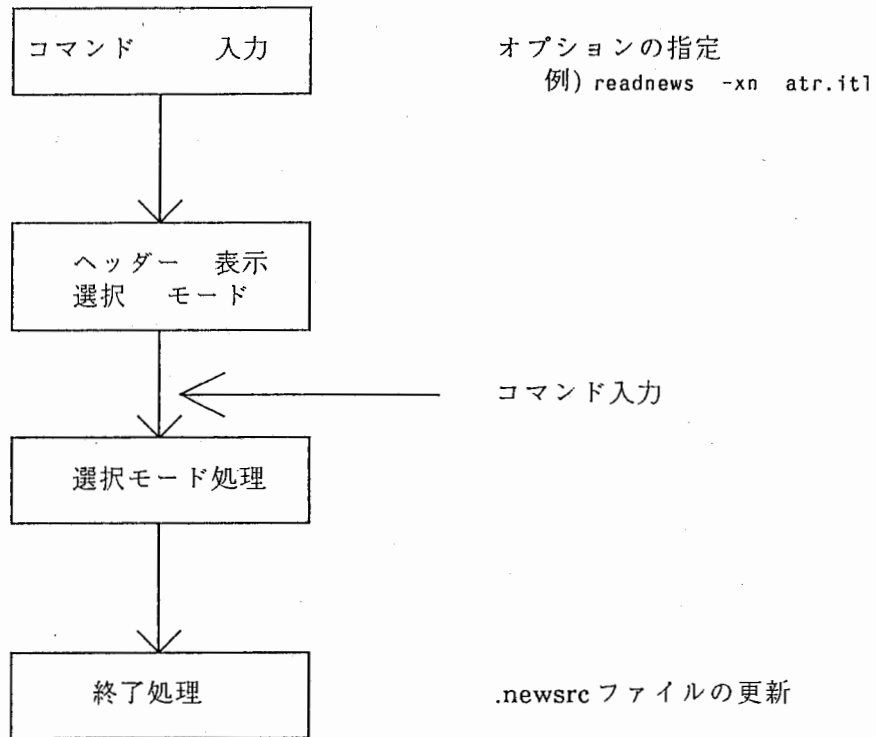
- a) ニュースの表示
- b) 次(前)の記事のヘッダー表示
- c) さらに詳しいヘッダーの表示
- d) 記事の読みとばし
- f) 他のニュースグループのサーチ
- g) 読んでいる記事のセーブ
- i) 画面表示されている記事に対してニュースを送る。  
(ヘッダーの自動生成)
- j) 画面表示されている記事の相手にリプライメールを送る。  
(ヘッダーの自動生成)
- k) help機能(入力コマンド一覧の表示)

6) 終了処理

選択モードから、xかqの入力により処理を終了する。終了する際に、.newsrcファイルの更新を行う。更新のときに、activeファイルに存在するニュースグループが.newsrcファイルにない場合、それらのグループを全て.newsrcファイルに追加する。ただし、xの入力の時は、.newsrcの更新を行わない。

## 5.特記事項

### 1) 動作説明



2) 入力コマンドのオプション一覧

オプション	引数	内容
-a	日付	指定された日付以降の記事を検索する
-c	メーラ	講読した記事を全てメーラにセーブする
-e	なし	ニュースの内容を表示しないで、タイトルだけの表示。 <code>.newsrc</code> の更新は行わない
-l	なし	<code>-e</code> と同じ動作をするが、 <code>.newsrc</code> の更新を行う
-n	グループ名	ニュースグループの指定
-p	なし	選択モードに移らず、全ての選択された記事を画面に出力する
-r	なし	新しいニュースから講読する
-t	タイトル名	指定されたタイトルの記事の検索
-x	なし	処理終了後、 <code>.newsrc</code> ファイルの更新を行わない

## 3) 選択モードからのコマンド一覧

その1

コマンド	引数	内容
—	なし	前の記事に戻る。2回タイプするともとの記事に戻る事が可能
#	なし	ニュースグループの名前とサイズを調べる
!	コマンド名	rednewsからunixのコマンドを起動
b	なし	記事をバックアップする
c	なし	記事をキャンセルする。ただし、作成者かスーパーユーザのみ可能
d	なし	記事をいくつかのダイジェストにし、それぞれに対して読んだり返答したりできる。
D	なし	暗号化した記事を解読する
e	なし	読んだ記事を未読状態のままにする
f	タイトル名 (通常なし)	フォローアップ記事を送信
fd	なし	fとほとんど変わらない
h	なし	通常表示されるヘッダーのさらに詳しい情報の出力
H	なし	ある記事に対して知られている全ての情報を出力する
K	なし	現在のニュースグループ内に残っている記事を全て読まれたものとして、次のニュースグループに移る
n	なし	記事の読みとばし
N	グループ名	指定されたグループに移る
p	なし	前の記事に戻る
P	なし	1つ前のニュースグループに移る
q	なし	処理終了-xか-lの指定がない場合は、.newsrcファイルを更新

コマンド	引数	内容
r	なし	リプライメールを出す
rd	なし	rとほとんど変わらない
s	ファイル名	読んでいる記事をサブする。通常ホームディレクトリ下のNewsというディレクトリ内にセーブ
U	なし	次のニュースグループに移る
v	なし	ニュースシステムの現在のバージョンを知る
w	なし	sコマンドと同じ
x	なし	処理の終了.newsrcファイルは更新されない
X	システム名	記事を指定されたシステムに転送する
y	なし	現在の記事を出力して、次の記事に移る
番号	なし	その記事番号の記事に移る
+ [n]	なし	n個の記事をスキップする

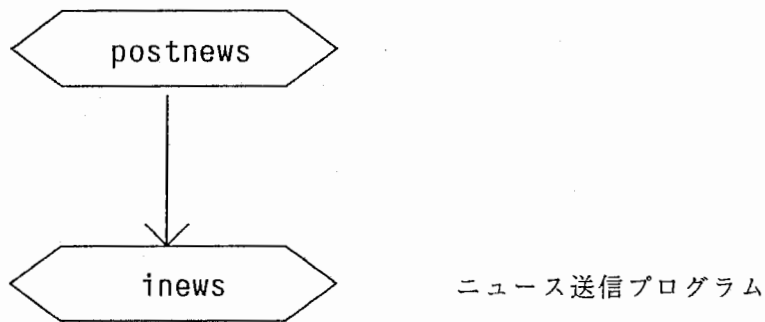
## 2.2.4 記事投函機能

### ★ニュース投函プログラム(postnews)

#### 1. 機能

指定したグループに記事を出す。

#### 2. 他のプログラムとの関連



#### 3. 入出力情報

- |                  |                 |
|------------------|-----------------|
| 1) activeファイル    | ニュース・グループ管理ファイル |
| 2) distributions | 配布範囲とその説明       |
| 3) newsgroups    | ニュースグループの簡単な説明  |

#### 4. 処理内容

- 1) ヘッダーの自動生成(postnews起動後、画面に表示されている各項目の入力[ニュースグループ等])
- 2) viエディタの起動(ニュースの内容の入力)
- 3) inewsを起動して入力された記事を送信する。

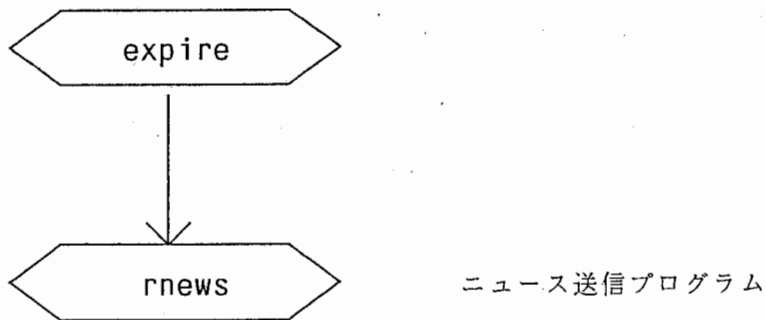
## 2.2.5記事削除機能

### ★ニュース削除プログラム(expire)

#### 1. 機能

古くなったニュースの削除を行う。

#### 2. 他のプログラムとの関連



#### 3. 入出力情報

active	ニュース・グループ管理ファイル
history	現在格納されている記事の一覧
history.dir	
history.pag	historyファイルのバイナリデータ
ohistory	historyファイルのバックアップファイル

#### 4.処理内容

- 1) historyファイルを読み込む。
- 2) 指定された日付以降の記事をhistoryファイルから探し出す。もし、historyファイルがクラッシュしている場合には、historyファイルのバイナリデータからhistoryファイルの再編集を行う。
- 3) スプールの中に格納されている古い記事を削除する。
- 4) historyファイル内の削除された記事の情報が記述されている部分を削除する。



- 5) 記事送信プログラムの起動 (rnews) expire中に一時的に.rnewsというディレクトリに溜められた記事を送信する。rnewsはinewsと実体は同じで、expire中に記事を出した時にデータ損失を回避するためのコマンドである。

## 5. 特記事項

### 1) オプション一覧

オプション	引数	内容
-a	なし	記事の削除をおこなわずに、その他の領域に移しかえる
-e	日付	指定された日付以前の記事を全て削除
-E	日付	指定された日付以前のhistoryファイルの中の情報を削除
-f	ユーザ名	特定のユーザからのニュースを全て削除に
-h	なし	historyファイルを無視して記事ファイルの日付により記事の削除を行う
-i	なし	記事内のexpiration dateと指定された日付のどちらも経過している記事の削除
-l	なし	記事内のexpiration dateの無視
-n	グループ名	指定されたニュースグループの削除視
-p	なし	記事の日付の記事がポストされた日付に変更
-r	なし	historyファイルの再構築を行ってから記事の削除を行う
-v	なし	verboseモードテストするときに使用
-u	なし	activeファイル中の最小の記事番号を更新

## 2.3 データ構造

ニュースシステムに必要なデータについて記述する。

### ★目次

1. /usr/lib/news/active
2. /usr/lib/news/aliases
3. /usr/lib/news/distributions
4. /usr/lib/news/errlog
5. /usr/lib/news/help
6. /usr/lib/news/history
7. /usr/lib/news/log
8. /usr/lib/news/mailpaths
9. /usr/lib/news/newsgroups
10. /usr/lib/news/seq
11. /usr/lib/news/sys
12. /usr/lib/news/users
13. /usr/lib/news/vnews.help
14. /usr/spool/news/[newsgroup]/\*
15. /usr/spool/news/batched/[sitename]

## 1. activeファイル

ニュースグループの管理ファイル

[例]

```
atr.cog 00005 00005 n
atr.general 00131 00123 y
atr.itl 00762 00659 y
atr.singles 00049 00039 y
atr.test 00216 00163 y
fj.announce 00216 00163 m
```

```
|
|
|
|
```

1ワード目: ニュースグループ名

2ワード目: 最新ファイルno.

3ワード目: 最古ファイルno.

4ワード目: y: 誰でもニュースを出せるグループ

n: ニュースの投函を禁止しているグループ

m: 特定の人しか出せないグループ  
(JUNETの管理者が出す記事を厳選)

\* ニュースグループの追加、削除の制御

\* expireが起動されるごとに最新のファイル情報を提供する。

\* expireが起動される前にoactiveというバックアップファイルにactiveファイルがコピーされる。

## 2. aliasesファイル

ニュースグループの書き換え参照ファイル

[例]

```
net.trivia          net.games.trivia
net.stat            net.math.stat
net.unix.wizards    net.unix-wizards
net.apollo          net.works.apollo
net.puzzles         net.puzzle
```

\* 最初のnet.triviaというグループに関するニュースが送られてきたら、今度他のサイトに送る時はnet.games.traviaというニュースグループで配布される。

## 3. distributionsファイル

ニュースを配布する範囲のコード化したものを記述

[例]

```
local              Local to this site
usa                Everywhere in the USA
na                 Everywhere in North America
world              Everywhere on Usenet in the world (sameasnet)
```

## 4. errlogファイル

ニュース・システム中で出しているメッセージの中でエラーメッセージにあたるものをsaveしている。

[例]

```
No v  9  12:50      <1091@icsts1.osaka-u.junet>      inews:Unknown
                               newsgroup fj.sys.news not localized
Nov   10  10:26     <2636@ih1pf.ATT.COM>           inews:Unknown
                               newsgroup comp.software-eng not localized
Nov   10  10:30     <3831@elecvox.eecs.unsw.oz>     inews:Unknown
                               newsgroup unsw.text not localized
```

## 5. helpファイル

readnewsでニュースを講読していて選択モードのオプションを調べたときに表示されるファイル

[内容]

Command	Meaning
y	Yes.(Or just hit return.) Prints this article and goes on.
n	No.Goes on to next article with out printing current one.
d	Digest.Breaks a digest article up into seperate articles.
q	Quit.Update.newsrc if -l or -x not used.
U	Unsubscribe.You won't be shown this newsgroup any more.
C	Cancel an article you posted.
r	Reply.Reply to article's author via mail.
f[title]	Submit a follow up article.
N[newsgroup]	Go to next newsgroup or named newsgroup.
s[file]	Save.Article is appended to file (default is "Articles").
s program	Run program with article as standard input.
e	Erase.Forget that an article was read.
h	Print verbose header.Use H for extremely verbose header.
!	Shell escape.
<number>	Go to message #<number> in this newsgroup.
-	Go back to last article.
b	Back up one article in the current group.
K	Mark the rest of the articles incurrent group as read.
X	Exit.Don't update .newsrc.
V	Version.Print current news version number.
l	List unread articles in newsgroup.
L	List all articles in newsgroup.
c,f,r,e,h,and s	can be followed by -'s to refer to the previous article

## 6. historyファイル

受け取ったニュースに関する記録。どこから、どういうグループについて、何時に着いたかなど。ohistoryというバックアップファイルを更新されるごとに生成する。

[例]

```
<1858@hplabsc.HP.COM>      06/01/87   21:10  news.announce.conferences/7
<2260@hplabsc.HP.COM>      07/30/87   11:04  news.announce.conferences/39
<855@hplabsz.HPL.HP.COM>   10/15/87   10:25  news.announce.conferences/63
<875@hplabsz.HPL.HP.COM>   10/16/87   13:54  news.announce.conferences/68
```

\* historyファイルが壊れた場合、history.pagというファイルを参照して復旧することができる。

`%/usr/lib/news/expire -r` (historyファイルの復旧)

## 7. logファイル

どこからどのようなニュースが届いたか、またそのニュースを隣接するどのサイトに送ったかなどのシステムを運用していく上で重要な情報が記述されている。

[例]

```
Mar 16 22:07 kdd1ab          <1573@pixar.UUCP> sent to atr-ln,atr-hr,
                               atr-sw,rcn01
Mar 16 22:07 kdd1ab          received <12424@brl-adm.ARPA> ng comp.lang.
                               pascal subj 'Re: WriteIn troubles'
                               from jc5z+@andrew.cmu.EDU (JohnL.Cusack)
Mar 16 22:07 kdd1ab          <12424@brl-adm.ARPA> sent to atr-ln,atr-hr,
                               atr-sw,rcn01
```

## 8. mailpathsファイル

usenetもしくはinternet経由でニュースを送信するときのメールアドレスの表記について記述されている。

```
backbone      %s@junet
internet      %s@junet
```

## 9. newsgroupsファイル

登録されているニュースグループとその内容について記述されている。新しいグループが作られた場合、自動的にこのファイルにグループ名とその内容が追加される。

[例]

general Articles that should be read by every one on your local system

mod.amiga.binaries	Encoded public domain programs in binary form.
mod.amiga.sources	Public domain software in source code format.
mod.announce.newusers	Explanatory postings for newusers.
mod.computers.68k	68000-based systems.

## 10. seqファイル

最後の記事の通番ファイル

[例]

1801

## 11. sysファイル

隣接するサイトへどのようにニュースを配布するか、どのグループについて送るかを記述している。

[例]

ホスト名:配送するグループの指定::配送のために使用するプログラム

```
atr-la:net,comp,sci,rec,news,soc,talk,misc,mod,fj,atr,ts1,world,to.atr-la::
atr-ln:net,comp,sci,rec,news,soc,talk,misc,mod,fj,atr,ts1,world,to.atr-ln::/usr/lib/news/
sendnews -a rnews@atr-ln
atr-hr:net,comp,sci,rec,news,soc,talk,misc,mod,fj,atr,world,to.atr-hr::/usr/lib/news/sendnews
-a rnews@atr-hr
atr-sw:net,comp,sci,rec,news,soc,talk,misc,mod,fj,atr,world,to.atr-sw::/usr/lib/news/sendnews
-a rnews@atr-sw
rcn01:net,comp,sci,rec,news,soc,talk,misc,mod,fj,atr,world,to.rcn01::/usr/lib/news/sendnews
-a rnews@rcn01
kddlab:net,comp,sci,rec,news,soc,talk,misc,mod,fj,world,to.kddlab:F:/usr/spool/news/batched/k
ddlab
```

## 12. usersファイル

ニュースを読んだユーザ名を記録する。(readnewsを初めて使った時と、.newsrcを新たに構築した時のみ)

```
root
aoi
isoda
post
dump
 |
 |
```

## 13. vnews.helpファイル

vnewsでニュースを講読していて選択モードのオプションを調べたときに表示されるファイル

[内容]

Vnews commands: (each may be preceded by a non-negative count)

CR	Next page or article	D	Decryptarot13joke
n	Go to next article	A	Go to article numbered count
e	Mark current article as unread	<	Go to article with given ID
+	Go forwards count articles	p	Go to parent article
-	Go to previous article	ug	Unsubscribe to this group
^B	Go backwards count pages	^L	Redraw screen
^N	Go forward count lines	v	Print netnews version
^P	Go backwards count lines	q	Quit
^D	Go forward half a page	x	Quit without updating .newsrc
^U	Go backwards half a page	c	Cancel the current article
h	Display article header	H	Display all article headers
!	Escape to shell	?	Display this message
r	Reply to article using editor	K	Mark rest of newsgroup read
R	Reply--put current article in reply	b	Go back 1 article in same group
ESC-r	Reply directly using mailer	m	Move on to next item in a digest
f	Post a follow up article	s	Save article in file
N	Go to newsgroup (next is default)	w	Save with out header
l	List unread articles in group	L	List all articles in group

[Press ^L to see article again]



#### 14. /usr/spool/news/[newsgroup]\*

実際にニュースが格納されているエリアは、/usr/spool/news/の下にニュース・グループ名のディレクトリ構造になっている。例えば、atr.testというニュース・グループを新たに作成した場合、/usr/spool/news/atr/test/というディレクトリを作り、その中にニュースを格納していく。新たに作成されたatr.testに初めてニュースを投函した場合、そのニュースのファイル名を1とし、新たに出されたニュースに対して1番から順番にそれをファイル名として格納される。

#### 15. /usr/spool/news/batched/[sitename]

日本国内など外部にニュースを送る場合の管理ファイルである。

[内容]            /usr/spool//news/batched/kddlab

```
/usr/spool/news/fj/question/unix/380
/usr/spool/news/fj/question/junet/1204
/usr/spool/news/fj/general/120
/usr/spool/news/comp/ai/24301
```

ATRではニュースはkddlabに転送しているので、/usr/spool/news/batched/kddlabというファイルに転送すべきニュースのファイルNo.が書き込まれている。この場合、ユーザがfj.question.unix,fj.question.junet,fj.general,comp.ai宛にニュースを送ろうとしている。これを見て、sendbatchがuuxを起動してkddlabにこの4つのファイルを送信するのである。

## 2.4 インストレーションガイド

### 2.4.1 概要

ニュース・システムは、アメリカのUSENETで開発され、その後、日本のJUNET内でも利用されるようになった。日本国内では日本語を扱うケースが多く、そのためニュース・システムの一部を修正して、漢字対応版が作られた。最新のバージョンは、news2.11 patch#14である。

### 2.4.2 特徴

- 1) 漢字の記事が読めるようになった。
- 2) 漢字コードの統一(新旧JISコード以外はだめ)

### 2.4.3 漢字の取扱いについて

JUNETでの漢字コードはJISコードで統一されているので、JISコードを使っているサイトについては、漢字コードを特に意識しないで使えるが、その他のコード例えばDECコードなどを使用している場合は、漢字の取扱いに注意を払う必要がある。ATR自動翻訳電話研究所内では、DECコードを中心としたファイルなどを整備しており、端末等もDECコードの設定をしているため、JISコード漢字を読むためにいくつかのツールを用意している。ニュースを読むときは、漢字変換するフィルターなどを使えば、簡単に読むことができる。ニュースを出すときは、記事を一旦JISコードに変換してポストニュースしないといけない。もしうまく変換されずにニュースを送ってしまったら、JUNETにつながっている他のサイトに多大の迷惑をかけるので、くれぐれも注意してポストしましょう。初めて出すときなどは、atr.testにあらかじめ出しておいて、うまく送られているか確認してから、投函するようにしましょう。

## 2.4.4 インストールの方法

### 1) 作業エリア

/usr/local1/src/news2.11/bnews/doc	ドキュメント類
/man	新バージョンマニュアル
/pat	パッチファイル
/src	ソースファイル

### 2) 最新のパッチをあてる。

既に、ソースにパッチがあてられている。

`%patch` < パッチファイル名

### 3) ヘッダーファイルの設定(ここでは、bsd版unix上での設定方法について述べているが、もしbsdのunixでない場合は、付録A及び付録Bを参照して下さい。)

bsd版のためのヘッダーファイルを使用

`%cp defs.h.BSD defs.h`

defs.hの以下の部分を変更する。

(ドメイン名の変更)

59c59

```
< #define MYDOMAIN ".atr.junet"          /*Localdomain*/
---
> #define MYDOMAIN ".cc.titech.JUNET    /*Localdomain*/
```

(sendmailのオプション設定)

75c75

```
< #define SENDMAIL "/usr/lib/sendmail-oi-oem"
---
> #define SENDMAIL "/usr/lib/sendmail"
```

(機関の名前)

77c77

```
< #define MYORG "ATRInternational,Japan"
```

---

```
> #define MYORG "/usr/lib/news/organization"
```

(JISコードから他のコード変換をするための関数の指定)

131,132c131,132

```
< #define NTOHK "/usr/jsy/bin/kcode +j"
```

```
< #define HTONK "/usr/jsy/bin/kcode -j"
```

---

```
> /* #define NTOHK "/usr/local/jtosj"
```

```
>/ * #define HTONK "/usr/local/sjtoj -j"
```

\*) 通常ここはコメントのままがいい。

#### 4) makeファイルの書き換え

bsd版用のmakefileを使用

```
%cp Makefile.BSD makefile
```

(ユーザー名及びグループ名の変更)

14,15c14,15

```
< NEWSUSR = daemon
```

```
< NEWSGRP = uucp
```

---

```
> NEWSUSR = news
```

```
> NEWSGRP = news
```

(既存のディレクトリがあるため)

81,82c81,82

```
< # -mkdir $(DESTDIR)$(BINDIR)
```

```
< # -mkdir $(DESTDIR)$(LIBDIR)
```

---

```
> -mkdir $(DESTDIR)$(BINDIR)
```

```
> -mkdir $(DESTDIR)$(LIBDIR)
```

5) データベースの書き換え(sys,activeファイル)

配送先の指定を行う。(sysファイルの変更)

[例]

```
atr-la:net,comp,sci,rec,news,soc,talk,misc,mod,fj,atr,ts1,world,to.atr-la::
atr-ln:net,comp,sci,rec,news,soc,talk,misc,mod,fj,atr,ts1,world,to.atr-ln::
/usr/lib/news/sendnews -a rnews@atr-ln
atr-hr:net,comp,sci,rec,news,soc,talk,misc,mod,fj,atr,world,to.atr-hr::
/usr/lib/news/sendnews -a rnews@atr-hr
atr-sw:net,comp,sci,rec,news,soc,talk,misc,mod,fj,atr,world,to.atr-sw::
/usr/lib/news/sendnews -a rnews@atr-sw
rcn01:net,comp,sci,rec,news,soc,talk,misc,mod,fj,atr,world,to.rcn01::/usr/lib/news/se
ndnews -a rnews@rcn01
kddlab:net,comp,sci,rec,news,soc,talk,misc,mod,fj,world,to.kddlab:F:/usr/spool/news/b
atched/kddlab
```

最新のニュース・グループの編成を行う(activeファイルの作成)。以下のnewsgroups.shを起動して最新のニュース・グループの登録を行う。通常 'make update'でactiveファイルの更新を行ってくれるのだが、データが古いため自分で最新のニュース・グループをactiveファイルに登録しないとイケない。新たにニュース・システムを導入する場合のみ必要で、バージョンアップなどでインストールする場合はこの5)項のデータの修正は必要ない。

[例]

```
/usr/lib/news/newsgroups.sh -n atr.itl
/usr/lib/news/newsgroups.sh -n atr.general
/usr/lib/news/newsgroups.sh -n atr.testexit
|
|
|
|
```

6) makeの起動

```
%cd /usr/local1/src/news2.11/bnews/src
```

```
%su
```

```
%make
```

```
%make install
```

```
%make update (データの作成)
```

7) inewsのモードを確認 (モードが6755ならOK)

```
%ls -l /usr/lib/news/inews
```

```
-rwsr-sr-x 2 daemon 70656 Nov 5 15:53 inews*
```

8) マニュアルのインストール

ニューバージョン対応のマニュアルをインストールする。

```
%su
```

```
%cd /usr/local1/src/news2.11/bnews/doc
```

```
%make
```

```
%make install
```

```
%man checknews expire getdate inews news newsrc postnews  
readnews recnews sendbatch sendnews uurec vnews
```

## 2.4.5 ニュースシステムのインストールチェック手順

自分のマシンのノード名を `mysys`、ニュースをフィードしてくれる上流ノード名を `oopsvax` とする。ディストリビューションキットからインストールして出来た `LIBDIR/active` は古いので、上流サイトに最新のニュースグループ構成を教えてください。そしてそれに合致するように `LIBDIR/active`, `LIBDIR/newsgroups` を書き替える。

・`LIBDIR/sys` に、`mysys` の記述だけを書きます。  
`mysys:fj,japan,world,comp,news,to::`

・ニュースグループ `to.mysys` と `local.test` を作成する。

1) `to.myname` にポストしてみる。

- a) `LIBDIR/log` にログが残っていること、
- b) `LINDIR/errlog` にエラーが記録されていないこと、
- c) `SPOOLDIR/to/myname/` に投稿した記事が入っていることまたそのヘッダの内容も正当であること

`Organization:`, `From:`, `Date:`, `Distribution:` に注目。

- d) ヒストリ(注1)に今投稿した記事に関する情報が追加されていること
- e) ニュースリーダーでそれが読めること
- f) `SPOOLDIR/.ar*`, `SPOOLDIR/.in*`, `/tmp/<*` などといった一時ファイルが残っていないこと。を確認する。

注1)

BSD では `LIBDIR/histiry`, SystemV では、`LIBDIR/history.d/[0-9]` がヒストリファイルである。これらはアスキーテキストファイルです。BSD での `history.dir`, `history.pag` は非アスキーですので `cat` で見ないように。

2) `local.test` でも同様にテストします。

`to.` で始まるニュースグループは特別扱いをされているので、`local` でもテストをする必要がある。

3) `expire` がうまく動くことを確認する。 `-v` フラグ(デバッグ用)を活用すれば便利。

- 4) 記事のキャンセルがうまく出来ることを確認する。

ここまでで、ローカルな機能の動作確認は完了したので、次は他のノードとの通信の確認を行う。

・ニュースグループ to.oopsvax を作成し、LIBDIR/sys に oopsvax を追加します。

```
oopsvax:fj,japan,world,comp,news,to.oopsvax:F:
```

・それから oopsvax の管理者にお願いして、ニュースグループ to.mysys を oopsvax で作成してもらう。

- 5) to.oopsvax にテストの記事を投稿する。

LIBDIR/log に oopsvax に送ったむね記録が残り、BATCHDIR/oopsvax に記事のファイル名(多分SPOOLDIR/to/oopsvax/1)が書かれていることを確認する。

- 7) sendbatch で記事を oopsvax あてに送り出す。

oopsvax の知り合いに頼んで、その記事がうまく oopsvax で受信できていること(oopsvax でニュースリーダで読めること)を確認する。

- 8) oopsvax からいくつか記事を送ってもらう。

to.mysys に投稿してもらってもいいし、普通の記事を送ってもらってもかまわない。そしてうまく受信できていることを確認。

◎これで大体うまく動いていることが確認できているはずだ。

それでも、どうしても心配なら、local よりは大きい、一番ちいさな配布範囲のニュースグループと distribution: で投稿してみる。

たとえば 大阪大学情報工学科なら、  
Newsgroups: osk.test  
Distribution: ics

となる。ローカルニュースグループを持たないところは、fj.test を使って確認。



記事の投稿がうまくいったら、BATCHDIR/ooopsvoxに記事のファイル名が入っていることを確認する。確認したら、ついでにその記事をキャンセルしてみる。キャンセルするコントロールメッセージのファイル名(SPOOLDIR/control/ なんか)がBATCHDIR/ooopsvoxに追加されるはずなので、うまくいってもいかなくてもBATCHDIR/ooopsvoxを消す。これでゴミアーティクルを流さずにローカル内だけでテストができる。

◎こんどこそうまく動いていることが確認。

mypassからooopsvoxへ行く記事の数がたいしてないと予想されるなら、LIBDIR/sysのooopsvoxのエントリの第3フィールドの`F'を消してみる。

こうした場合は、もういちどto.ooopsvoxへ記事を投稿してみて、こんどはいきなりuuxが起動されてuucpジョブが登録されたことを確認する。/usr/lib/uucp/LOGFILEの末尾に、

```
news ooopsvox (7/24-0:40-25236) XQT QUE'D (rnews)
```

とかいう記録が出来ていることで確認する。

まっ、このようにしてfj.testを使わずに、また使ったにしても、全国に記事を流すことなしに、ニュースシステムのインストールのチェックが出来るのである。もちろん、100%完全に確認できたわけではないので、あとは実際に記事を受信し、またfjその他に記事を投稿してゆくことで(もし問題点があれば)明らかになるであろう。

## 2.4.6 rnのインストール

### [修正箇所]

- 1) バージョン2.11のreadnewsでは、各自が作ったnewsrcファイルに登録されていないnewsgroupを書きたしてしまう。readnewsを起動するときには、newsgroupを指定するので別段問題はないのだが、rnでnewsを読む場合には、書きたされたnewsrcファイルが、500ラインを超えてしまうので"too many lines newsrc"というメッセージがでてnewsが読めなくなってしまう。(readnewsとrnの併用禁止)

対策として、rnの読み込み可能ライン数を1024に変え、common.hとinit.cを修正した。詳しいことは、下記を参照。

```
/*
*/usr/local/src/news/rn/common.h
/*

27,31c27
< *
< *
< * changed DEFINE for readnews. 87/11/26 NOGA & AOI
< * MAXRCLINE 500 --> 1024
< * HASHSIZ 547 --> 2048
---

212c208
< #define MAXRCLINE 1024 /* number of lines allowed in .newsrc*/
---
> #define MAXRCLINE 500 /* number of lines allowed in .newsrc*/

216c212
< #define HASHSIZ 2048 /*should be prime,and at least MAXRCLINE+10%*/
---
> #define HASHSIZ 547 /*should be prime,and at least MAXRCLINE+10%*/
```

```

/*
/* /usr/local/src/news/rn/init.c
/*

298c298
<      set_nickname("atr.itl");
---
>      set_nickname("net.announce");

302c302
<      fputs("Can't find atr.itl Wrong news version?%n",stdout);
---
>      fputs("Can't find net.announce Wrong news version?%n",stdout);

```

- 2) rnでnewsを読んでいる最中にメールを送ろうとfを入力した時に出てくるヘッダ-の Reply-to:の所がloginname@atr-la.UUCPとなっている。UUCPの所をatr.junetと変更する。

```

*
* /usr/local/src/news/rn/Pnews.SH
*

288c288
< Reply-To: $logname@$sitename.atr.junet ($fullname)
---
> Reply-To: $logname@$sitename.UUCP ($fullname)

*
* /usr/local/src/news/rn/intrp.c
*

783c783
<      sprintf(tmpbuf,"%s@s:atr.junet",dot,beg);
---
>      sprintf(tmpbuf,"%s@s.UUCP",dot,beg);

```

```
*
* /usr/local/src/news/rn/common.h
*
```

507c503

```
< # define NEWSHEADER "Newsgroups: %F%nSubject: Re: %S%nSummary: %nExpires: %nReferences:
%R%nSender: %nReply-To: %L%H.atr.junet (%N)%nFollowup-To: %nDistribution: %D%nOrganization:
%o%nKeywords: %n%n"
---
```

```
> # define NEWSHEADER "Newsgroups: %F%nSubject: Re: %S%nSummary: %nExpires: %nReferences:
%R%nSender: %nReply-To: %L%H.UUCP(%N)%n Followup-To: %nDistribution: %D%nOrganization:
%o%nKeywords: %n%n"
```

574c570

```
< # define CANCELHEADER "Newsgroups: %n%nSubject: msg cancel %i%nReferences: %R%nReply-To:
%L%H.atr.junet (%N)%n Distribution : %D%nOrganization: %o%n%n%i cancelled from rn.%n"
---
```

```
> # define CANCELHEADER "Newsgroups: %n%nSubject: msg cancel %i%nReferences: %R%nReply-To:
%L%H.UUCP (%N)%n Distribution: %D%nOrganization: %o%n%n%i cancelled from rn.%n"
```

### 3) Pnews.SHの修正

日本国内のニュースグループfj\*にニュースを送れるようにする。Pnews.SHの次の2ヶ所を変更する。

185c182

```
< Everywhere: net,mod,fa
---
```

```
> Everywhere: fj,net,mod,fa
```

209a207,209

```
> fj*)
```

```
> dist='fj'
```

```
> ;;
```

## 2.4.7 rnインストール手順

### 1) Configureの起動

%su

%Configure

Beginning of configuration questions for rnkit.

(Makeファイル及びPnewsの書換え開始)

Checking echo to see how to suppress newlines...

...using -n.

Type carriage return to continue. Your cursor should be here-->

(リターン入力)

[Type carriage return to continue]

(リターン入力)

Is your "test" built into sh? [n] (OK to guess)

(リターン入力)

binaries on multiple machines? [n]

(リターン入力)

Your sitename appears to be "atr-ln".

Is this correct? [y] (このサイトが、lnならリターン入力)

For example: "University of Southern North Dakota,Hoople"

Organization: 機関名(atrを入力)

is used for news administration? [news] (リターンを入力)

Does your /etc/passwd file keep full names in Berkeley format

(name first thing after ':')? [y] (リターン入力)

Your system appears to use index() and rindex() rather than strchr() and strrchr(). Is

this correct? [y] (リターン入力)

that file? [F] (リターン入力)

Give the full path name of the shell most people like to use on your system: [/bin/csh]

ニューcシェルなら/usr/local/bin/newcshを入力

What pager is used on your system? [/usr/ucb/more]

ページャーの設定

(リターン入力)

What is the default editor on your system? [/usr/ucb/vi]

エディタの設定

(リターン入力)

Where do you want to put the public executables? [/usr/local/bin]

(リターン入力)

Where do you want to put these private executables?

(リターン入力)

[/usr/lib/news/rn] (~nameok) RNLIB = /usr/lib/news/rn

(リターン入力)

Where do the manual pages go?[/usr/man/man1]

(リターン入力)

The categories you will be asked are:

local organization? atrを入力

organization? atrを入力

city? osakaを入力

state/province? osakaを入力

country? japanを入力

continent? asiaを入力

End of configuration questions.

この後、Pnewsとmakeファイルを自動的に生成してmakeを起動する。

2) makeの起動

```
%make
```

```
%make install
```

3) コマンドの動作の確認

a) rnの起動

```
%rn
```

\*各オプションの動作確認

b) Pnewsの起動

```
%Pnews
```

- \* 各オプションの動作確認
- \* 何々というグループにニュースが投函できるか確認
- \* ヘッダーのサイト名の語尾がUUCPでなく、junetになっているか確認

### 3. メールシステム

#### 3.1 メールシステム概要

マシン間の情報伝達の方法として、メールシステムがあげられる。ある端末にログインしている相手に、何か知らせたい事やファイルを転送する場合に、相手のログイン名を指定すれば、その宛宛にファイルが転送される。しばらくして、ログインしている相手側の端末に'You have a new mail'というメッセージが表示されて、メールが届いた事を知らせる。例えば工場内などでちょっとした用件のために席を立つのは非効率なことであり、その場合メールが威力を発揮する。JUNETとの接続により、国内はもちろん海外にもメールが送れるようになっている。

##### 3.1.1 各機能説明

メールシステムには、次のような機能がある。

- 1) メール送信機能
- 2) メール受信機能

##### 1) メール送信機能

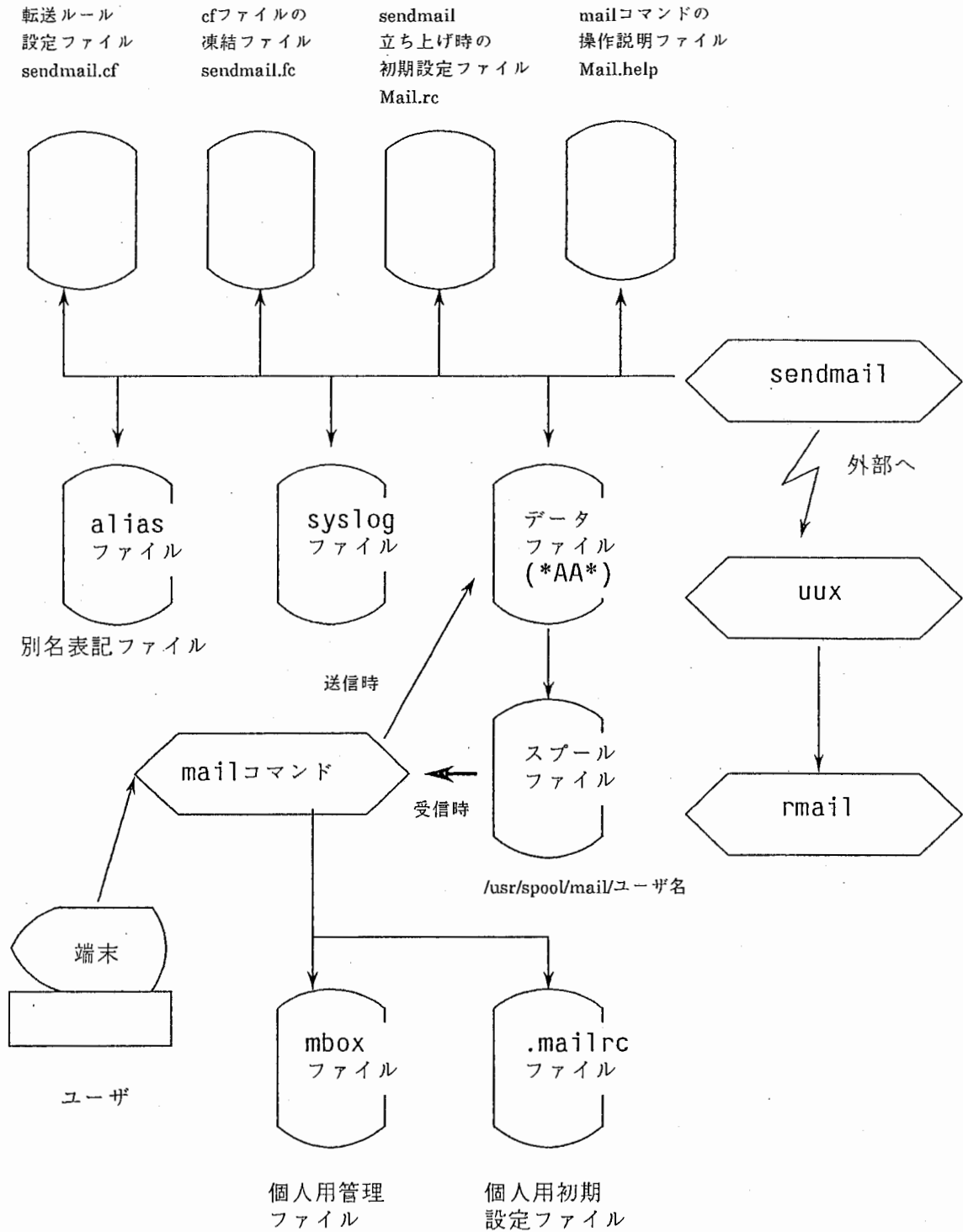
宛先や本文を入力後、ヘッダーの自動生成処理が行われたファイルを転送先のスプールに送信する。

##### 2) メール受信機能

自分宛のメールが届いた時、標準出力にその旨メッセージが表示される。メールを読んだ後、読んだファイルはホームディレクトリ下のmboxというファイルにセーブされる。



### 3.1.2 システム構成



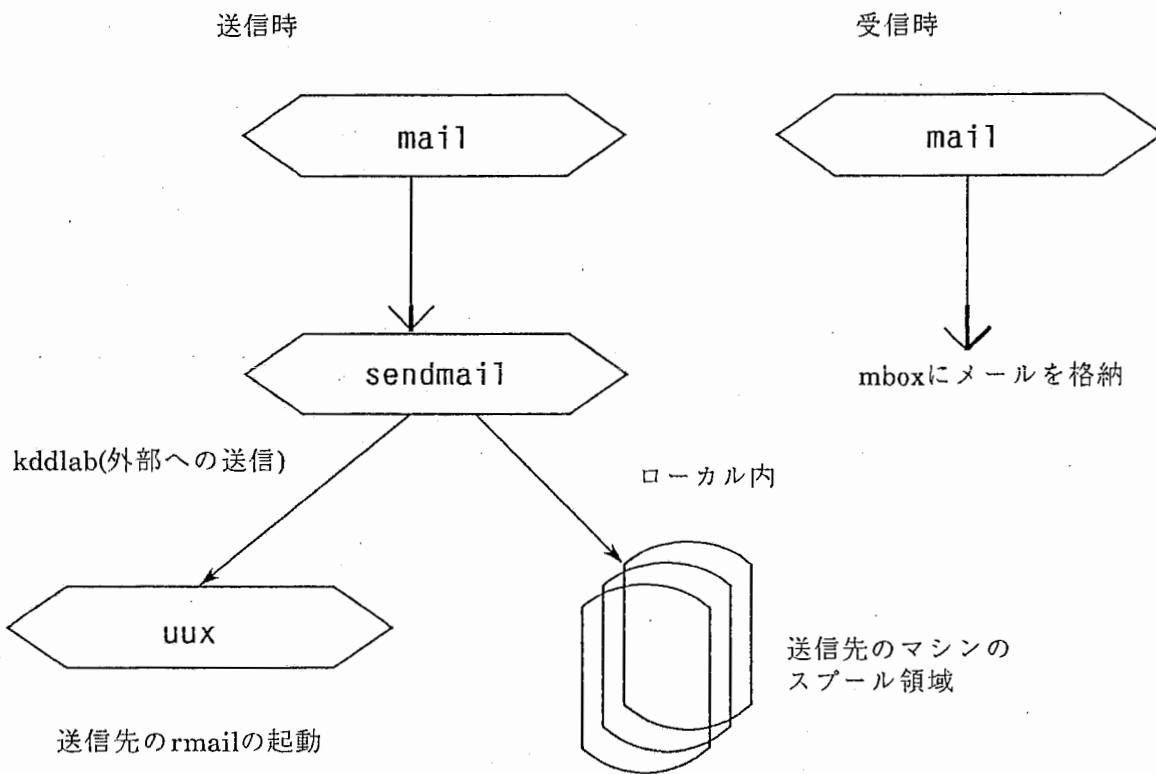
## 3.2 各プログラム説明

### 3.2.1 mailプログラム

#### 1. 機能

他のユーザにメールの送信、他のユーザからのメールの受信を行う。

#### 2. 他のプログラムとの関連



### 3. 入出力情報

- |                       |                  |
|-----------------------|------------------|
| 1) /usr/spool/mail/*  | 受信したメールが格納されるエリア |
| 2) ~/mbox             | 読んだ記事がセーブされるファイル |
| 3) ~/.mailrc          | メール初期設定ファイル      |
| 4) /tmp/R#            | 本文記述時のテンポラリファイル  |
| 5) /usr/lib/Mail.help | ヘルプファイル          |
| 6) /usr/lib/Mail.rc   | システム初期設定ファイル     |

### 4. 処理内容

#### [送信時]

- 1) 入力チェック
- 2) 各項目の入力(subject等)
- 3) 本文の入力(edの起動)、テンポラリファイルにセーブ
- 4) sendmailの起動(メール送信の処理を行う)

#### [受信時]

- 1) 入力チェック
- 2) 簡単なヘッダの表示
- 3) 選択モード処理 (コマンド入力により以下の処理を行う。)
  - a) メッセージ出力
  - b) メッセージ編集
  - c) メッセージ削除
  - d) リプライメールの送信
  - e) unixのコマンドの起動
  - f) ヘッダの出力
  - g) ヘルプ機能

#### 4) 終了処理

メールを /usr/spool/mail/[username] ファイルから、mail コマンドで読んだメールだけを、そのユーザのホームディレクトリの mbox に移しかえて処理を終了する。

### 5. 特記事項

#### 1) ~/.mailrc(初期設定ファイル)のコマンドについて

set .....各項目の設定

set ask.....mail コマンドでメールを送るときに subject を聞いてくる。

set askcc.....指定した送信先以外の送信先を聞いてくる。

set EDITOR.....エディタの設定

set SHELL.....シェルスクリプトの設定

alias.....送り先名の簡略化

#### 2) オプション一覧

オプション	引数	意味
-f	ファイル名	既に読んで mbox か他のファイルにセーブしたものを読む。
-i	なし	回線の割込み信号を無視してメールを送信
-n	なし	Mail.rc ファイルを参照しない
-s	subject名	サブジェクトの指定
-u	ユーザ名	ユーザ名の指定
-v	なし	verbose mode

3)選択モードからのコマンド一覧

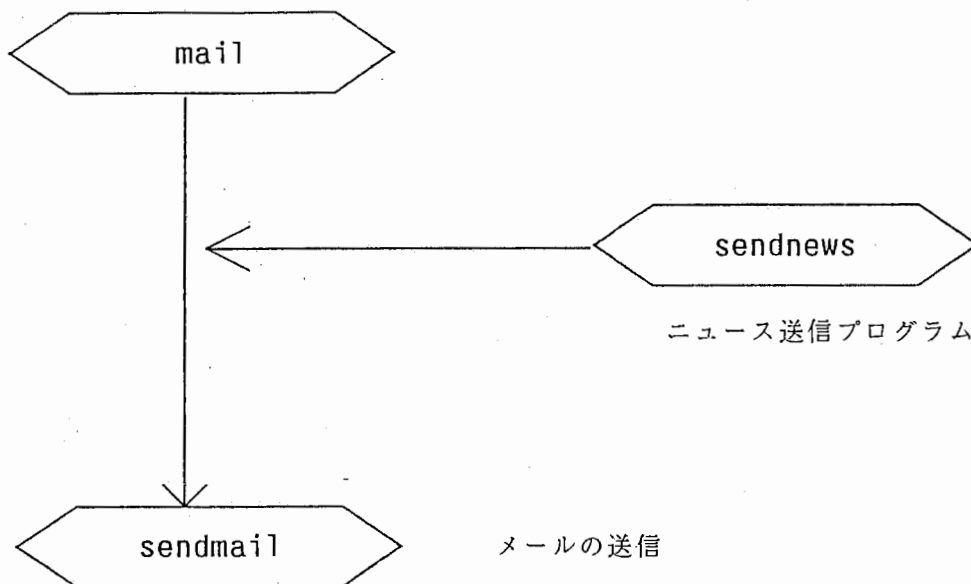
コマンド名	意味
t	メッセージ出力
n	次のメッセージの出力
e	メッセージの編集
f	ヘッダのみの表示
d	メッセージ削除
s	ファイル名の指定によりメールの格納
u	メッセージ削除の取消
r	リプライメール(送信者及び受信者)
R	リプライメール(送信者のみ)
pre	メッセージを/usr/spool/mailへ戻す。
m	特定の人にメールを送信
q	mboxにメールをセーブして処理を終了
x	メールのセーブを行わずに処理を終了
h	現在のmailのメッセージのヘッダを表示
!	シェルエスケープ
ch	ディレクトリの変更

### 3.2.2 sendmailプログラム

#### 1. 機能

メールの送信を行う。メール形式のもの(ニュースなど)もこのsendmailを使ってファイル転送を行っている。

#### 2. 他のプログラムとの関連



#### 3. 入出力情報

- |                               |                              |
|-------------------------------|------------------------------|
| 1) /usr/lib/aliases           | ログイン名の簡略ファイル                 |
| 2) /usr/lib/aliases.pag       |                              |
| 3) /usr/lib/aliases.dir       | aliasファイルのバイナリデータ            |
| 4) /usr/lib/sendmail.cf       | コンフィギュレーションファイル(転送ルールの設定)    |
| 5) /usr/lib/sendmail.fc       | sendmail.cfのイメージファイル(凍結ファイル) |
| 6) /usr/lib/sendmail.hf       | SMTPのヘルプファイル                 |
| 7) /usr/lib/sendmail.st       |                              |
| 8) /usr/bin/uux               | リモートマシンのコマンド実行               |
| 9) /usr/spool/mqueue/*AA*     | 転送前に作られるメールファイル              |
| 10) /usr/spool/mqueue/syslog* | 転送記録ファイル                     |
| 11) /usr/spool/mail/[usrname] | 転送後セーブされるファイル                |

#### 4. 処理内容

- 1) mailコマンドより起動
- 2) mailプログラムで作成されたテンポラリファイルをヘッダ部と本文に分けて/usr/spool/mqueueにセーブする。
- 3) メールキュー(mqueue)に溜ったメールをヘッダ部の指定されたユーザのプール(/usr/spool/mail/[username])に順次送信する。

送信の手順を以下に示す。

- a) sendmail.cfファイルの読み込み
  - b) 送信先のユーザ名及びホスト名が存在するかどうか確認する。もしなければ、エラーメールを送信元とメールの管理者に送る。
  - c) メールを送信してから3日以上経過しているか確認する。経過しておれば、メールを送信しない。
  - d) ローカル内のメールは送り先のマシンの/usr/spool/mail/[username]にデータを書き込み、外部へはuuxを起動して送り先の/usr/spool/mail/[username]にメールを転送する。その時の転送経路は、sendmail.cfを参照する。
- 4) メールキューに溜っているメールの表示を行う。
  - 5) コマンドの設定により、各処理のトレースを行う。

## 5. 特記事項

### 1)コマンド一覧

コマンド名	意味
-bd	デーモンで起動
-bi	アライアスデータの初期設定
-bm	通常のメール起動オプション(デフォルト)
-bp	メールキューの中の表示(mailqコマンド)
-bs	プロトコルの指定(SMTP)
-bt	テストモード(転送アドレスの表示)
-bz	sendmail.fcファイルの作成
-n	alias設定の禁止
-q	強制的にキューイングされているメールを掃き出す。このコマンドの後に時間が設定されている場合は、その時間内送り先のマシンにメールが送信されるまで処理を継続
-v	verbose mode
-m	aliasの設定を参照
-o	古いスタイルのヘッダを使用
-Q	メールキューの指定
-T(時間)	指定された時間を超えて溜っているメールを削除 デフォルトでは、3日になっている



### 3.3データ構造

メールシステムに必要なデータについて説明する。

#### ★目次

1. /usr/lib/sendmail.cf
2. /usr/lib/sendmail.fc
3. /usr/lib/sendmail.hf
4. /usr/lib/sendmail.st
5. /usr/lib/aliases
6. /usr/lib/Mail.help
7. /usr/lib/Mail.rc
8. /usr/spool/mqueue/\*AA\*
9. /usr/spool/mqueue/syslog\*
10. /usr/spool/mail/[username]
11. ~/mbox
12. ~/.mailrc

## 1. sendmail.cfファイル

メール転送のためのルール設定ファイル各サイトによっていろいろ設定を変える。ファイルの解析は、相当の熟練を必要とし、ましてファイルの修正となると手作業ではほとんど出来ないだろう。そこで、sendmail.cf作成のためのツールが用意されているので(JUNET対応版)、それを使ってこのファイルを作成する。具体的には、メールの送り先により転送経路・メーラーの指定を行う。例えば、社外にはuucpメーラを使い、社内のマシン間の転送はsmtpメーラを使用する。

## 2. sendmail.fcファイル

sendmail.cfファイルのイメージファイル

/usr/lib/sendmail -bz'このコマンドを起動したときに作成する。作成された後は、sendmail.cfファイルを直接参照しないでこのファイルを参照する。sendmail.cfファイルを修正した場合、sendmail.fcファイルを削除してから上記のコマンドを起動してsendmail.fcファイルを作成しなおさないといけない。

## 3. sendmail.hfファイル

smtpのヘルプコマンドで表示される内容を記述

## 4. sendmail.stファイル

メールの統計を取るために必要なファイル

## 5. aliasesファイル

### 名前の簡略ファイル

フォーマットはA:Bで、Aには正式名・Bには別名の記述を行う。このファイルは、ただ単にユーザ名が長いから短くする以外に、例えばroot宛にくるシステムからのメッセージなどを管理者へ転送するようにしたり、同じグループの人には共通の名前を設けてそこにメールを送れば、同じグループの人全員にメールが転送されるようになる。

[例]

```
#Basic system aliases -- these MUST be present
MAILER-DAEMON:post
postmaster:post
daemon:post
uucp:post
root:abe

#Aliases to handle mail to msgs and news
msgs:"|/usr/ucb/msgs-s"
rnews:"|/usr/lib/news/uurec"
usenet:post

#people. Add your people here.
admin:noga
kawabata:kaw

#People in ATR International
members:kurematu,yodogawa,noga,iida,umeda,katagiri,sato,aizawa,
tohkura,sagisaka,shikano
```

aliasesファイルを書き換えた場合、newaliasesコマンドを起動してaliasesファイルの再定義を行う必要がある。その時に、aliases.pag、aliases.dirというファイルが生成される。

## 6. Mail.helpファイル

mailコマンド起動後の選択モード処理中にヘルプコマンドを入力した時に、表示されるヘルプファイル

[例]

```
Mail Commands
t <messagelist>          typemessages
n                        goto and type next message
e <messagelist>          edit messages
f <messagelist>          give head lines of messages
d <messagelist>          delete messages
s <messagelist> file     append messages to file
u <messagelist>          undelete messages
r <messagelist>          reply to messages
R <messagelist>          reply to messages (to sender only)
pre <messagelist>        make messages go back to /usr/mail
m <userlist>             mail to specific users
q                        quit, saving unresolved messages in mbox
x                        quit,do not remove system mailbox
h                        print out active message headers
!                        shelle scape
ch [directory]          chdir to directory or home if none given
```

A <messagelist> consists of integers,ranges of same,or user names separated by spaces.If omitted,Mail uses the last message typed.

A <userlist> consists of user names or distribution names separated by spaces.Distribution names are defined in .sendrc in your home directory.

## 7. Mail.rcファイル

メールシステムの初期設定ファイル

[例]

```
set append dot save
```

## 8. queueファイル

送信要求されたメールを一時的に格納するエリア\*AA\*という形でファイルが格納され、4つ野データファイルを作成する。\*AAの後にくる5桁の数字は、送信要求したときのsendmailのプロセスidが入る。

### ファイルの説明

qfAA01485...コントロール・ファイル

メール送信のためのメールのヘッダ部分に相当する。

dfAA01485...データ・ファイル

メールの本文に当たる。

lfAA01485...ロック・ファイル

このファイルが存在する間、メールの送信は不可能

xfAA01485...バックアップ・ファイル

コントロールファイルのコピーファイルで、sendmailの処理中メールのヘッダのバックアップを取っておいて、ヘッダファイルが欠落した時に備えている。

## 9. syslogファイル

送信されたメールの記録ファイル

フォーマットは、日付・送信時のpid-no.・キューイングされているファイル名・送信または受信先・サイズ(送信時はデレイタイム)・優先順位(送信時はプロセスの動作状態)となっている。

[例]

```
May 31 13:22:46 localhost: 1489 sendmail: AA01485: to=<m.mcd@lear.stanford.edu>,
delay=00:00:03, stat=Sent
May 31 13:38:16 localhost: 1827sendmail: AA01827: message-id =
<8805310437.AA00889@atrv07.local>
May 31 14:17:47 localhost: 12080 sendmail: AA12064: to=irie@atr-hr.atr.JUNET,
delay=00:00:06,stat=Sent
May 31 14:31:23 localhost: 16282 sendmail: AA16282: from=aoi,size=361,class=0
May 31 14:31:25 localhost: 16307 sendmail: AA16282: to=aoi,delay=00:00:06,stat=Sent
```

#### 10. スプーリングファイル(/usr/spool/mail/\*)

送信されたメールの格納ファイル。/usr/spool/mail下に各ユーザ名で取り込まれ、ユーザがmailコマンドで手紙を読む時はここを参照する。

#### 11. メールボックス(mbox)

個人の郵便箱。手紙を読んだ後に、各ユーザ毎のホームディレクトリに格納するようになっている。

#### 12. メール環境設定ファイル(.mailrc)

個人用の環境設定ファイル詳細は、3.2.2の5項を参照して下さい。

### 3.4 インストレーションガイド

メールの転送経路を指定するためには、sendmail.cfの書き換えが必要である。しかし、sendmail.cfを直接書き換えるのは技術的難易度が高いため、sendmail.cfを構築するためのツールとしてmailconfが用意されている。JUNETに加入しているサイトでは、このツールを利用してsendmail.cfの構築を行わなければならない。最近のバージョンはmailconf6.3Jと言われるもので、以下にそのインストール方法について説明する。

#### 3.4.1 [特徴]

- 1) 自分の配下にあるドメインの管理関係、ネットワークの結合関係を明確にしている。
- 2) ドメインの別名が登録できる。
- 3) 特殊なメイラの登録ができる。
- 4) さらに、バージョン6.3では、東大経由でcsnetとリレーが可能となった。(ただし、東大計算機センターへの申請が必要)

#### 3.4.2 [データフォーマット]

##### 1) alias.dat

最初に実アドレス書いて、次に別名を記述する。別名は複数の別名を持ってもいい。

##### [例]

```
# titech and titan
# format:
# real-name : alias-name(s)
titech.junet:                titan.junet
                             tokodai.junet
cs.titech.junet:            koudai.titech.junet

# Osaka Univ.
handai.junet:                osaka-u.junet

# Tokyo Univ.
u-tokyo.junet:               utyo.junet
                             todai.junet
```

## 2) mailer.dat

メイラの指定を行う。各種のアドレス表記に対応するために必要なデータである。

```
[例] # mailer name #address domain address
M junet-n notouch nodelete #permit any addressing scheme
M junet-o domain nondelete #old junet? user%dom1@dom2
M junet domain nondelete #old junet? user%dom1@dom2
M smtp-n domain nondelete #
M smtp-o domain nondelete #
M smtp domain nondelete #
M pmdf route nondelete #force '@dom1,@dom2:user@dom3'
#M mmdf route nondelete #force '@dom1,@dom2:user@dom3'
M uucp bang delete #force 'sys!sysB!user'
M uucp-o bang delete #force 'sys!sysB!user'
M uucp-n bang nondelets #force 'sys!sysB!dom1!user'
#U
U uucp
S smtp
```

### [記述方法]

#### a) 1ワード目

M: メイラの指定

S: ローカルのアドレス解析のためのメイラ指定

U: 普通のメイラでは認識できないアドレスを解析するためのメイラ指定

#### b) 2ワード目

具体的メイラ種別

junet,smtp,uucp,pmdf

#### c) 3ワード目

アドレスタイプの変更を行う。どこに送るかによってメイラが決まり、そこで有効なアドレスタイプに変更する。

domain ---> user%domain%domain@domain

route ---> domain,domain:user@domain

bang ---> host!host(domain)!user

notouch ---> 変更なし



d) 4ワード目

delete: uucpで送るときにアドレス内にドメイン名を記述できない。  
3word目がbangのときだけ指定できる。

nodelete: 普通はこの指定。アドレス内にドメイン名を記述できる。

3) desc.dat

ドメインの管理関係について記述するファイルである。atr-laの記述フォーマットを示す。

- a) まず、laはメインドメインで他に5つのサブドメインが存在する。この場合、atr.junetが外部との受け口で、その下にla(laとatr.junetは同じ意味)、ln、sp、sw、hr、rcn01がある。la以外のサブドメインについては、この部分の記述は必要ない。

```
#
# name information
#
# description for atr-la.atr.junet
name:
    atr.junet
lower_level_name:
    atr-la atr-ln atr-sp atr-sw atr-hr rcn01
```

- b) 次に、ここはla内なので、laに関する詳しい説明を行なう。laの下にはサブドメインがないので、lower\_level\_name:にNONEをする。

```
# description for atr-la.atr.junet
name:
    atr-la.atr.junet
lower_level_name:
    NONE
#
# rout information
#
```

- c) atr内のマシンの結合関係を示す。今現在いるのはlaなので、メイラーの指定をlocalとして、他のサイトをsmtpの指定にする。

ドメイン名メイラ実アドレス

```
atr-la:      local   atr-la.atr.junet
              atr.junet

atr-ln:      smtp    atr-ln.atr.junet

atr-sp:      smtp    atr-sp.atr.junet

atr-sw:      smtp    atr-sw.atr.junet

atr-hr:      smtp    atr-hr.atr.junet

rcn01:      smtp    rcn01.atr.junet
```

- d) サブドメインの記述方法

上記と同じ箇所のサブドメインでの記述方法の違いは、ローカルでないアドレスはすべてatr-laに送る。これは、atr-ln内の記述です。

```
# Local Area Network
atr-la:      smtp    atr-la.atr.junet
              atr.junet
              junet

atr-ln:      local   atr-ln.atr.junet

atr-sp:      smtp    atr-sp.atr.junet

atr-sw:      smtp    atr-sw.atr.junet

atr-hr:      smtp    atr-hr.atr.junet

rcn01:      smtp    rcn01.atr.junet
```

e) この記述は、メインドメインから外部に送る場合の指定です。

Nttlab、京都大学、大阪大学、九州大学、神戸大学へはNttlabを経由で、それ以外はKddlab経由でメールが送られる。サブドメイン内では必要ない。

```
# Other organizations
# sitename :Mail-system-name sub-domainname
#
kddlab:      junet      kddlabs.junet      #KDDLlab.
              junet
nttlab:      junet      ntt.junet           #NTTLab.
              kyoto-u.junet      #KyotoUniv.
              osaka-u.junet      #OsakaUniv.
              kyushu-u.junet     #KyushuUniv.
              kobe-u.junet       #KobeUniv.
```

### 3.4.3 インストールの手順

上の3つのファイルの記述さえ正確に書けておれば、問題はない。

#### 1) 各マシンでの作業領域(キットのあるエリア)

```
atr-la: /usr/local/src/UUCP/mailconf_6.3J
atr-ln: /usr/local/src/UUCP4.3/mailconf_6.3J
atr-sp: /usr/local/src/UUCP/mailconf_6.3J
```

#### 2) mailconf\_6.3J内のエリア説明

```
DESC: alias.datなどデータエリア
MST: sendmail.cfを作成するためのもとのなるデータエリア
doc: インストールに関する資料
src: mailconfのソースがある。
```

#### 3) インストールの手順

```
%cd ../mailconf_6.3J/src      (作業エリアに移る)

%su                            (スーパーユーザで作業)

%make                          (mailconfの作成)

%mailconf                      (mailconfの起動)

%ls -l se*
-rw-r--r--  1 root   17769 Mar  9 12:01 sendmail.cf

                                (sendmail.cfファイルが作成されたことを確認)

%ps -ax | grep se              (sendmailのPIDを調べる)

18539 ?  I    1:16 /usr/lib/sendmail -bd -q10m -om
9644  S7 S    0:00 grepse

%kill -9 18539
```

\*\*誰もメールを送っていないことを確認後  
(古いsendmailのサーバの終了)

```
%ls -l /usr/lib/se*
```

```
-rwsr-xr-x 1 root 82944 Jun 23 1987 /usr/lib/sendmail*  
-rwsr-xr-x 1 root 82944 Jun 11 1987 /usr/lib/sendmail.bak*  
-rw-r--r-- 1 root 17773 Dec 18 10:44 /usr/lib/sendmail.cf  
-rw-r--r-- 1 root 12621 Jun 11 1987 /usr/lib/sendmail.cf.bak  
-rw-r--r-- 1 root 12441 Jun 26 1987 /usr/lib/sendmail.cf.old  
-rw----- 1 root 98408 Mar 1 13:33 /usr/lib/sendmail.fc  
-rw-r--r-- 1 root 1520 Jun 11 1987 /usr/lib/sendmail.hf
```

```
%rm /usr/lib/sendmail.fc (古いsendmail.cfのメモリイメージを消去)
```

```
%mv /usr/lib/sendmail.cf /usr/lib/sendmail.cf.org  
(古いsendmail.cfをセーブ)
```

```
%mv sendmail.cf /usr/lib/. (新しいsendmail.cfの移行)
```

```
%/usr/lib/sendmail -bz (新しいsendmail.cfのメモリイメージを作成)
```

```
%/usr/lib/sendmail -bt (テストモードで試験を試みる。)
```

```
%/usr/lib/sendmail -bd -q10m -om (新しいsendmailサーバの起動)
```

## 4. uucpシステム (version 4.3)

### 4.1 システム概要

unixシステムのマシン間のファイル転送及びコマンド実行を可能にしたシステム

#### 4.1.1 各機能説明

当システムには、次の2つの機能がある。

- 1) システム間ファイル送受信機能
- 2) システム間コマンド実行機能

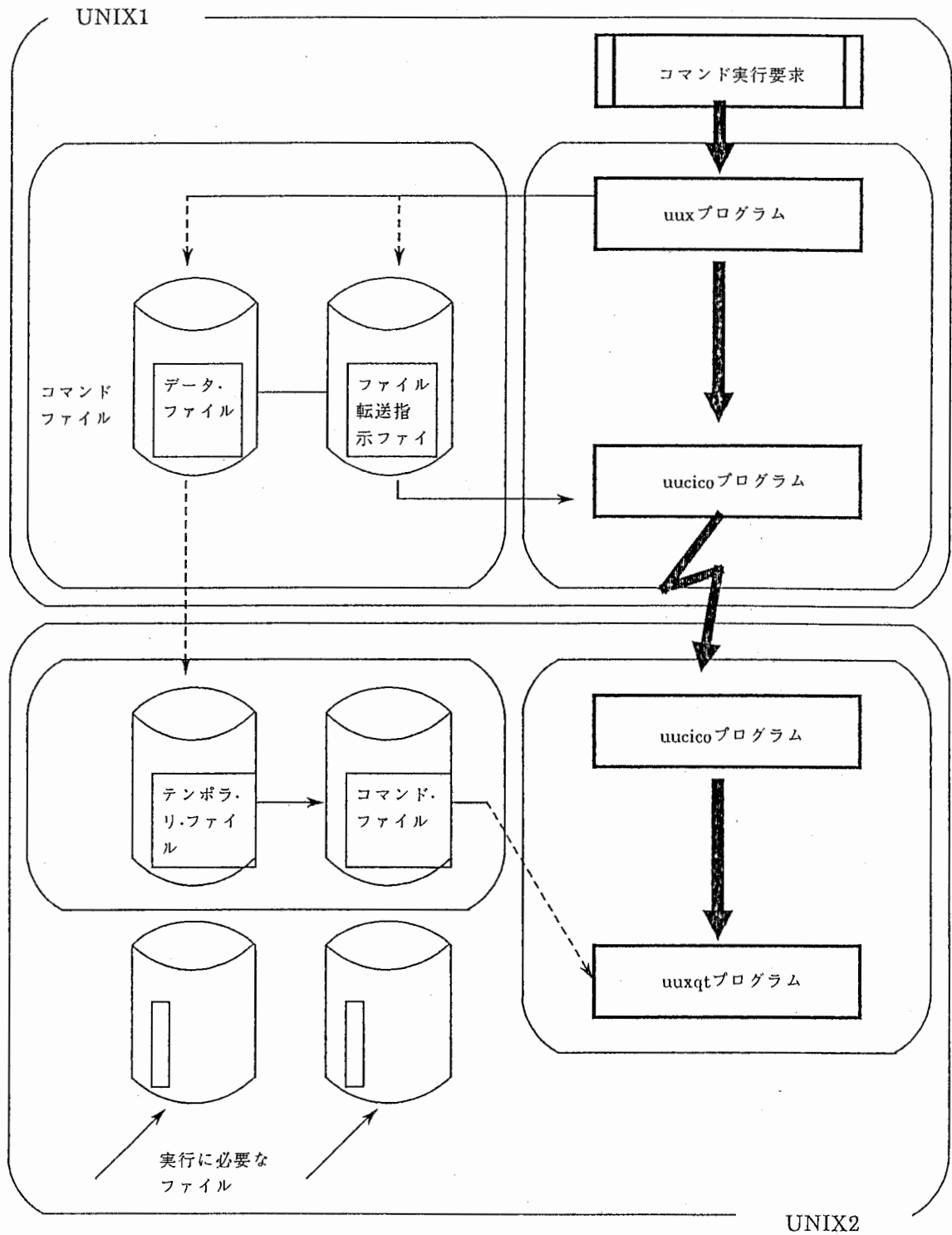
##### 1) システム間ファイル送受信機能

指定したファイルを他のunix稼働システムにファイルを転送する。また、転送されてきたデータを指定されたディレクトリに格納する。

##### 2) システム間コマンド実行機能

指定されたコマンドをファイルにセーブして他のシステムに送信する。受信側のサイトでは、受信したファイルを読み込みファイル内に記述されたコマンドを実行する。また、ファイルの種類により例えばメールファイルであるなら、rmailを起動してメール受信の処理を行う。

4.1.2 システム構成



## 4.2 各プログラム説明

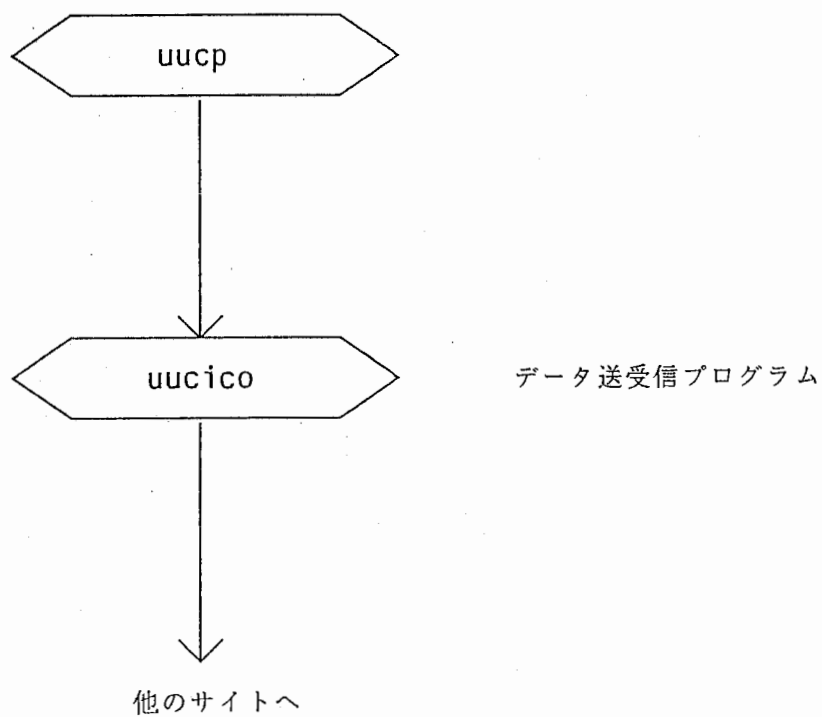
### 4.2.1 システム間ファイル送受信機能

#### ★ファイル転送プログラム(uucp)

##### 1. 機能

他のサイトへのデータ転送要求を行う。

##### 2. 他のプログラムとの関連



##### 3. 入出力情報

1) /usr/spool/uucp/\*

転送されるデータを一時的に溜めるエリア



#### 4. 処理内容

- 1) 引数チェック
- 2) 転送データのチェック(一度に転送できるデータ量は、50000バイト15ファイルまでとする。)
- 3) 転送されるデータに一時的に名前を付けて、スプーリングする。
- 4) データ送信プログラム(uucico)の起動

#### 5. 特記事項

- 1) オプションの説明

オプション	意味
-d	ファイルの複写に必要な全てのディレクトリを作成
-c	/usr/spool/uucpの下にスプール・ファイルを作成せずに、もとのファイルをそのまま送信
-m	複写が終了した時点でuucpを要求した側にメールを送信

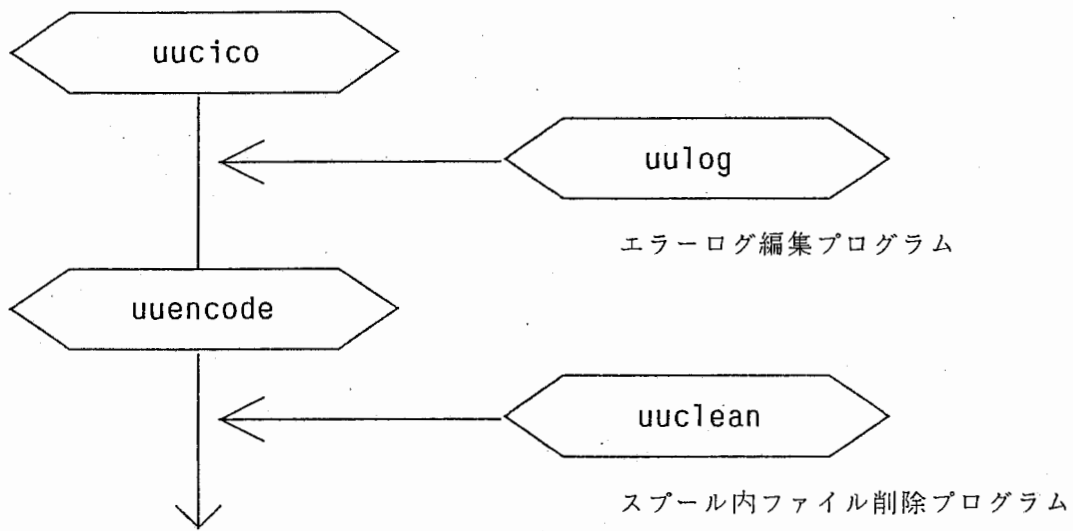
★ファイル送受信プログラム(uucico)

1. 機能

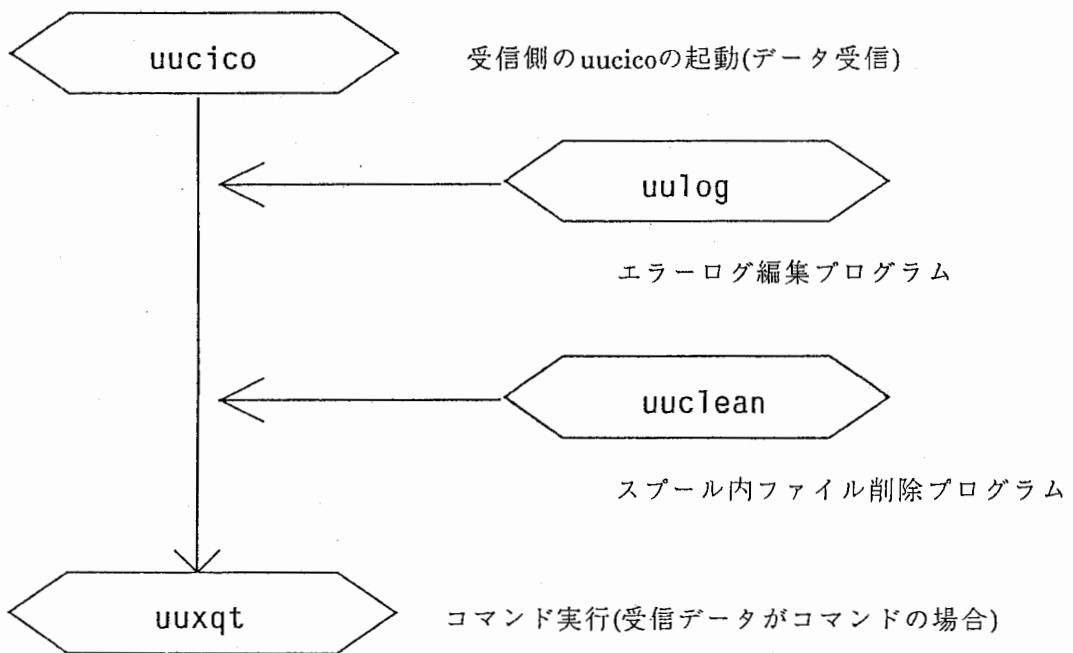
uucp及びuuxから当プログラムが起動される。ファイル転送側のシステムからuucicoを起動して、相手側のuucicoに起動をかけてファイルの受信を行う。

2. 他のプログラムとの関連

送信側



受信側



### 3. 入出力情報

- |                              |               |
|------------------------------|---------------|
| 1) /usr/spool/uucp/ERRLOG    | システム・エラー・ファイル |
| 2) /usr/spool/uucp/LOGFILE   | 転送動作記録ファイル    |
| 3) /usr/spool/uucp/SYSLOG    | 転送データ記録ファイル   |
| 4) /usr/lib/uucp/L-devices   | 接続回線管理ファイル    |
| 5) /usr/lib/uucp/L-dialcodes | 電話番号略称ファイル    |
| 6) /usr/lib/uucp/L.sys       | リモートホスト情報ファイル |

### 4. 処理内容

- 1) 引数チェック
- 2) 電話回線から送信できるようにセットアップする。
- 3) 回線状態をonにする。
- 4) L.sysファイルを読み込む。
- 5) L.sysに記述されているモデムを使用して、相手側を呼び出しログインする。
- 6) ファイルを転送する。転送後、転送されたデータの記録をLOGFILE及びSYSLOGに書き込む。
- 7) スプール内の転送されたファイルを削除する。
- 8) エラー処理  
当処理中に起こったエラーをERRLOGファイルに書き込む。

### 5. 特記事項

- 1) オプション説明

オプション	意味
-r1	マスターモードでプログラムを起動
-s	データを指定されたシステムに送信
-d	スプーリングするディレクトリの指定
-x	デバッグモードのレベルを指定

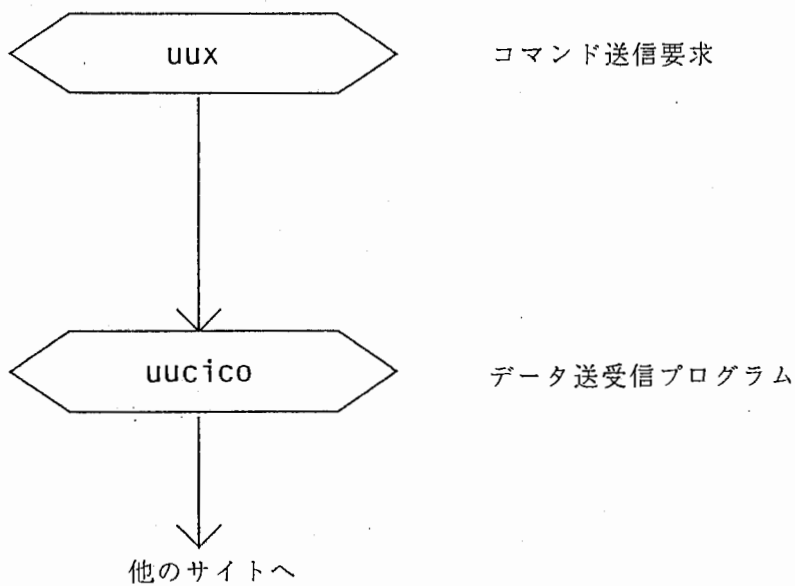
## 4.2.2 システム間コマンド実行機能

### ★コマンド実行要求プログラム(uux)

#### 1. 機能

他のunixシステムに対して、コマンド実行要求を出す。

#### 2. 他のプログラムとの関連



#### 3. 入出力情報

- 1) /usr/spool/uucp/\*                      転送するデータを一時的に溜めるエリア

#### 4. 処理内容

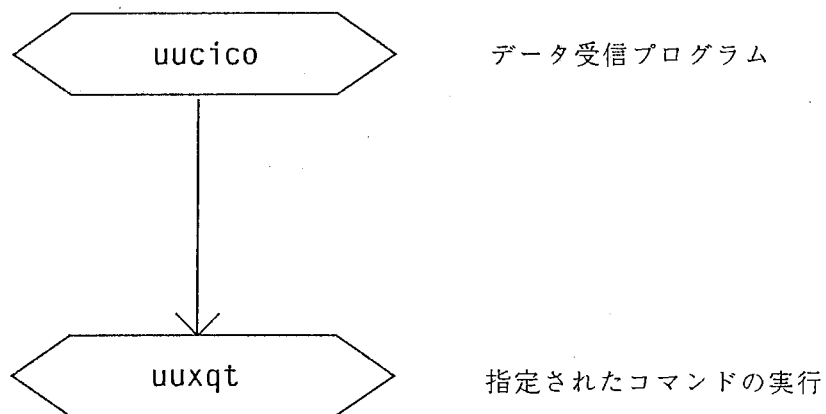
- 1) 引数チェック
- 2) 指定されたコマンドをファイルに置き換える。
- 3) 置き換えられたファイルを一時的に、スプールに溜める。
- 4) uucicoを起動して、ファイルを転送する。

## ★コマンド実行プログラム(uuxqt)

### 1. 機能

uucicoにより受信したファイルをもとに、そのファイルがコマンド実行命令のものであればコマンドを実行する。また、メールやニュース形式のファイルに関しては、それらのファイルを認識した時点で各受信プログラムに起動をかけるようにする。

### 2. 他のプログラムとの関連



### 3. 入出力情報

- 1) /usr/spool/uucp/\*                      転送されたファイルが格納されているエリア
- 2) /usr/lib/uucp/L.cmds                  リモートコマンド登録ファイル

### 4. 処理内容

- 1) uucicoでスプーリングされたファイルをオープンする。
- 2) ファイルの種類によって起動をかけるプログラムを決める。例えば、メール形式のファイルならrmail、ニュース形式のファイルならrnews、その他のファイルはruusendを指定する。
- 3) L.cmdsファイルを読み込む。
- 4) 起動をかけるプログラムが、L.cmdsに記述されている事を確認。記述されていなければ、異常終了させる。

★通信回線接続状況表示プログラム(uusnap)

1. 機能

今現在の接続されているサイトとの通信状態を標準出力に表示する。

2. 他のプログラムとの関連

なし

3. 入出力情報

1) /usr/spool/uucp/STST/[sitename]           回線状態ファイル

4. 処理内容

- 1) /usr/spool/uucp/STST内に存在するファイルを読み込む。
- 2) 読み込んだファイルをもとに、各接続サイトとの通信状況を標示する。

5 .特記事項

1) 操作例

```
%uusnap
```

```
Kddlab 12 Cmd _____ 20 Data Login Failed:
```

```
Nttlab   Cmd _____
```

```
%
```

★転送速度表示プログラム(uurate)

1. 機能

1日のデータ量・転送速度を表示する。

2. 他のプログラムとの関連

なし

3. 入出力情報

1) /usr/spool/uucp/SYSLOG            転送データ記録ファイル

4. 処理内容

- 1) SYSLOGファイルを読み込む。
- 2) 各送信先ごとに転送データと送信データ分けて、データ量と転送時間をカウントアップする。
- 3) 上の処理を終えたら、平均のデータ転送時間とトータルのデータ数と転送時間を計算する。
- 4) 上記のデータを標準出力に表示する。

5. 特記事項

1) 操作例

```
%/usr/lib/uucp/uurate
```

```
got from kddlab 128 files 872814 bytes 8330 secs 104.78 bytes/sec
got from nttlab 54 files 34520 bytes 99 secs 347.49 bytes/sec
sent to kddlab 152 files 202431 bytes 1821 secs 111.14 bytes/sec
sent to nttlab 28 files 16006 bytes 149 secs 107.09 bytes/sec
kddlab live10151 secs dead 997 secs
nttlab live 248 secs dead 100 secs
```

```
%
```

### 4.3 データ構造

uucpに必要なデータについて記述する。

#### ★目次

1. /usr/spool/uucp/ERRLOG
2. /usr/spool/uucp/LOGFILE
3. /usr/spool/uucp/SYSLOG
4. /usr/spool/uucp/STST/\*
5. /usr/lib/uucp/L-devices
6. /usr/lib/uucp/L-dialcodes
7. /usr/lib/uucp/L.cmds
8. /usr/lib/uucp/L.sys
9. /usr/spool/uucp/C.
10. /usr/spool/uucp/D.
11. /usr/spool/uucp/D.[hostname]
12. /usr/spool/uucp/D.[hostname]X
13. /usr/spool/uucp/TM.
14. /usr/spool/uucp/X.



## 1. エラーログ・ファイル(ERRLOG)

システム内で発生したエラーを全て記録する。

フォーマットは、ASSERT ERRORの後にエラーの起こったプログラム名・プロセスid・発生時間・エラーの内容となっている。

[例]

```
ASSERT ERROR (uuxqt) pid: 9856 (6/2-20:52) CAN'T OPEN
                               /usr/spool/uucp/X.kdd1abd2hr3 (0)
ASSERT ERROR (uuxqt) pid: 15331 (6/2-21:22) CAN'T OPEN
                               /usr/spool/uucp/X.kdd1abd2h13 (0)
ASSERT ERROR (uux) pid: 676 (6/11-21:03) CAN NOT GET SEQLOCK (0)
ASSERT ERROR (uux) pid: 817 (6/11-21:09) CAN NOT GET SEQLOCK (0)
ASSERT ERROR (uucico) pid: 17396 (6/12-14:15) SYSTAT OPEN FAIL (0)
ASSERT ERROR (uucico) pid: 17401 (6/12-14:15) SYSTAT OPEN FAIL (0)
ASSERT ERROR (uucico) pid: 10044 (8/17-11:30) CAN NOT OPEN
                               /usr/spool/uucp/sys(0)
ASSERT ERROR (uucico) pid: 10044 (8/17-11:30) SYSTAT OPEN FAIL
                               /usr/spool/uucp/STST./STST.(2)
ASSERT ERROR (uuxqt) pid: 9279 (10/31-12:02) DEAD LOCK ./LCK..XQT (2)
ASSERT ERROR (uucico) pid: 28578 (11/3-19:33) DEAD LOCK ./LCK..ttyAa (3)
```

## 2. ログ・ファイル(LOGFILE)

各プログラムの動作(どういうデータを受信または、送信したか)の記録ファイル

フォーマットは、送り主・送信先・(送信日-送信時間-pid.no)・ログ(動作の詳細)となっている。

[例]

```
root kddlab (5/23-12:45-19430) FAILED (call to kddlab)
root nttlalab (5/23-12:46-19447) SUCCEEDED (call to nttlalab)
root nttlalab (5/23-12:46-19447) OK (conversation complete)
daemon kddlab (5/23-12:48-19465) XQT QUE'D (rmail nbs@cunyvms.cuny.edu)
daemon kddlab (5/23-13:16-20006) REQUEST (S D.atr-1aB3Kd2 D.atr-1aS3Kd2 daemon)
shikano kddlab (5/23-14:17-20996) REQUEST (S D.atr-1aB3Kj2 D.atr-1aS3Kj2 shikano)
uucp kddlab (5/23-14:18-21040) uucp XQT (rmail aoi@atr-1a.atr.junet)
```

### 3. システムログ・ファイル(SYSLOG)

全ての転送・送信データのサイズ、転送または送信時間の記録ファイル

フォーマットは、送り主・送信先・(送信日・送信時間)(メッセージid)・データの種別・データのサイズ・転送にかかった時間となっている。

[例]

```
daemon kddlab (5/23-14:17) (580367831.67) sent data 1699 bytes 15.33 secs
daemon kddlab (5/23-14:17) (580367835.82) sent data 109 bytes 0.04secs
shikano kddlab (5/23-14:17) (580367846.35) sent data 766 bytes 6.27secs
shikano kddlab (5/23-14:17) (580367850.52) sent data 96 bytes 0.04secs
uucp kddlab (5/23-14:17) (580367871.33) received data 1510 bytes 15.60 secs
uucp kddlab (5/23-14:17) (580367876.10) received data 87 bytes 2.56 secs
daemon kddlab (5/23-14:46) (580369608.40) sent data 661 bytes 5.66 secs
daemon kddlab (5/23-14:46) (580369612.85) sent data 92 bytes 0.06 secs
uucp nttl (5/23-14:47) (580369669.07) received data 1089 bytes 2.66 secs
uucp nttl (5/23-14:47) (580369670.70) received data 86 bytes 0.59 secs
uucp nttl (5/23-14:47) (580369673.92) received data 816 bytes 2.10 secs
uucp nttl (5/23-14:47) (580369676.49) received data 83 bytes 0.80 secs
daemon kddlab (5/23-15:17) (580371472.49) sent data 90 bytes 0.03 secs
daemon kddlab (5/23-15:18) (580371497.55) sent data 2244 bytes 20.79 secs
daemon kddlab (5/23-15:18) (580371501.71) sent data 90 bytes 0.04 secs
root kddlab (5/23-15:18) (580371513.45) sent data 833 bytes 7.49 secs
root kddlab (5/23-15:18) (580371517.50) sent data 63 bytes 0.02 secs
```

### 4. 回線状態表示ファイル(STST/[接続サイト名])

uusnapで接続されているサイトへの接続状況を調べるために必要なファイル。uucicoが起動されている間に常に接続状態が変わるため、その都度このファイルを更新している。正常にuucicoの処理が終了したら、このファイルは削除される。

フォーマットは、データ数・uucicoの起動回数・メッセージid・転送速度・メッセージ・通信先の順になっている。

[例]                   ファイル名    /usr/spool/STST/kddlab

```
4 1 580382154 600 LOGIN FAILED kddlab
```

## 5. 接続回線管理ファイル(L-devices)

どの回線から転送するか設定する。

フォーマットは、接続方法・接続回線名・呼出し装置名・転送速度となっている。

[例]

```
PAD ttyA0 unused 9600 direct
PAD ttyA1 unused 9600 direct
PAD ttyA2 ttyA2 9600 direct f
PAD ttyA3 unused 9600 direct
DIR ttyA9 unused 1200 direct
DIR ttyAa unused 1200 direct
DIR ttyAb unused 1200 direct
PAD ttyA0 unused 9600 direct
PAD ttyA1 unused 9600 direct
PAD ttyA2 ttyA2 9600 direct
PAD ttyA3 unused 9600 direct
ACU ttyA9 unused 1200 direct
ACU ttyAa unused 1200 direct
ACU ttyAb unused 1200 direct
```

## 6. L-dialcodesファイル

電話番号の略称を記述するファイル

[例]

```
# ExampleL-dialcodes
# I suppose there are sites that actually use this sort of thing.
boston 149
cb 8-353
chicago 127-
dr 8-374
hog 133-
hoy 135-
ih 8-367
in 8-356
ix 8-372
mh
mv 8-245
```

## 7. リモートコマンドファイル (L.cmds)

このファイルに記述されたコマンドのみuuxのリモートコマンド実行要求を実行する。

[例]

```
PATH=/bin:/usr/bin:/usr/ucb
rmail
rnews
ruusend
bnproc
unbatchnews
uux
# These next programs are security holes ,but noone seems to care
lpr
who
uusend
finger
cunbatch
```

## 8. L.sysファイル

uucpネットワークで接続するリモートホストについての記述

フォーマットは、リモートホスト名・時間(Anyは常時接続可)・接続装置・転送速度・接続ポート・電話番号・ログイン手順となっている。

#DDX

```
nttlab Any PAD 9600 ttyA2 ""¥r *~1-¥r-*~1-¥r-*~1-¥r-* ¥r ready~1-¥r- ready ls¥r error ¥r
ready ¥d¥d¥d¥c ""SET¥0401:0,2:0,3:2,4:10,5:1,6:0,7:0,9:0,10:0,12:1,13:0,14:0,15:0,
16:0,17:0,18:0¥r ""¥d¥d¥d¥c ""SET¥0401:0,2:0,3:2,4:10,5:1,6:0,7:0,9:0,10:0,12:1,13:0,
14:0,15:0,16:0,17:0,18:0¥r ""C¥s0000001"" ¥d¥d¥d¥d¥d¥d¥d¥d¥d¥d¥d¥d¥d¥d¥d¥d¥d¥d¥d¥d
ogin:-¥r¥r-ogin:-¥b¥r-
ogin: nttlab ssword: himitu
```

#MODEM

```
#nttlab Any DIR 1200 ttyAb ""¥r¥d¥r¥d¥r¥d¥r"" ATDP0400586175 CONNECT ¥r¥r¥ ogin:-¥r¥r-
ogin:-¥r¥r-ogin:
#nttlab Any DIR 1200 ttyAa ""¥r¥d¥r¥d¥r¥d¥r > DP0400586175¥r INITIATING ¥r¥r
""¥r¥rlogin:-r-login: kddl lab ssword: himitu
```

9. /usr/spool/uucp/C./\*

ファイル転送コマンドの命令ファイル

[例]

```
S D.atrpostB5cK2 D.atrpostS5cK2 daemon - D.atrpostB5cK2 0666
S D.atrpostX5cK0 X.atrpostA5cK3 daemon - D.atrpostX5cK0 0666
```

10. /usr/spool/uucp/D./\*

他のサイトから受信したデータの格納される領域

11. /usr/spool/uucp/D.[hostname]/\*

送信するデータ(メール・ニュースなど)を格納される領域

12 /usr/spool/uucp/D.[hostname]X/\*

実行すべきコマンドとそれに与える関数、入力ファイル等の記述

[例] /usr/spool/uucp/D.atrpostX/D.atrpostX5cJ0

```
U root atrpost
N
Z
R root
F D.atrpostS5cJ2
I D.atrpostS5cJ2
C rnews
```

この場合、オーナーがrootでこちらに溜まっているニュースを転送先のシステムのrnewsを起動することによって、相手システムにニュースを受け取らせることを意味している。

13. /usr/spool/uucp/TM./\*

ファイルを受信するときの一時ファイル。ファイル全体を正しく受信したあと、目的とする名前につけかえる。TM.pid.seqというふうにuucicoのプロセスIDが真中にきて、末尾に各uucicoプロセスごとのシリアル番号がつく。例えば、TM.08244.001というファイルに一時的に格納される。

14. /usr/spool/uucp/X./\*

リモート側からのX.(コマンド実行要求に関する情報)ファイルを格納される領域

## 4.4 インストレーション・ガイド

### 4.4.1 外部との接続テスト

#### 1. 接続ポートの設定

unixのtipコマンドを使って、モデム等からリモート・ログインできるかどうか確認する。そのために、`/etc/ttys`と`/etc/remole`というファイルの設定を変更しなければならない。

##### 1) ttysファイルの設定

実際の運用例をもとに設定方法について述べる。1つのサイトとuucpによるデータ転送をすることとなり、モデムを4台(2台は受信専用、2台は送信専用)設置することとなった。各モデムが接続されているシリアル・ポートのデバイス名は、受信専用がttyh0,ttyh1で、送信専用がttyh2,ttyh3に割当てることとする。以下にデフォルトのttysファイルがある。

```
/etc/ttys
```

```
12console
02ttya
02ttyb
02ttyh0
02ttyh1
02ttyh2
02ttyh3
```

ttyh0とttyh1は受信専用なので、こちらからはアクセスできないようにDTRフラグを立てておく必要がある。そこで、ttyh0とttyh1の最初のフラグに1のビットを立てる。

```
/etc/ttys
```

```
12console
02ttya
02ttyb
12ttyh0
12ttyh1
02ttyh2
02ttyh3
```

ttysファイルの修正が終われば、ターミナルの設定を変えたことをinitプロセスに伝えるために、"kill-1 1"を起動する。

```
%kill -1 1
```

このttysファイルはbsd4.2の記述方法をもとにしているが、ultrix等拡張unixでは多少記述方法が異なる。その点は、管理者用マニュアルで再確認する必要がある。

## 2) remoteファイルの設定

tipコマンドで接続確認をする場合、最低限/etc/remoteファイルに接続するデバイスとインターフェース速度を記述しなければならない。

```
/etc/remote
```

```
test1:dv=/dev/ttyh0:br#9600
test2:dv=/dev/ttyh1:br#9600
test3:dv=/dev/ttyh2:br#9600
test4:dv=/dev/ttyh3:br#9600
```

dvはデバイスの指定、brはインターフェース速度を示す。

## 3) tipコマンドによる接続確認

```
%tip test3
CONNECTED
ATDP0694941830      (モデム用のコマンドを入力して相手側のモデム
                     に発信)

RING

CONNECT    FAST      (相手側モデムが信号を受信)

リターンを入力

kddlab      (ultrix2.2)

login:        (ログインプロンプトが返ってくることを確認)
```



## 2. モデムの設定

tipコマンドでモデムから発信する場合、各モデムによってコマンドが用意されており、多少コマンドの使い方が違う。ここでは、高速インテリジェンモデムといわれているトライル・プレーザ(以下TB)について詳しく説明する。TBは、18,000bpsまで転送速度を上げることができる。TBのコマンドはヘイズ社のATコマンド言語に準拠しているので、ヘイズ社のモデムを使ったことがあれば、ログイン手順等に困る心配はない。ただ、モデム自体にいろいろな機能があるため、最適な使い方をしようと思えば、各レジスタの設定を変更しなければならない。その設定方法について、順を追って説明する。

### 1) TBの自己診断の実行

工場から出荷されたモデムが正常に動作すかどうかの確認するためのコマンドが用意されているので、それを使って確認をする。

AT&T (自己診断コマンドの入力)

”Diagnostics Complete”というメッセージが表示されれば、OK

### 2) レジスタ値の変更

ATN& (レジスタ値の表示)

E1 F1 M1 Q0 P V1 X1 Version BD4.00

S00=001 S01=000 S02=043 S03=013 S04=010 S05=008 S06=003 S07=059 S08=002 S09=006

S10=007 S11=070 S12=050

S45=000 S47=004 S48=000 S49=000

S50=255 S51=004 S52=002 S53=000 S54=003 S55=000 S56=017 S57=019 S58=003 S59=000

S60=000 S61=045 S62=003 S63=001 S64=000 S65=000 S66=001 S67=000 S68=255

S90=000 S91=000 S92=000 S95=000

S100=000 S101=000 S102=000 S104=000

S110=001 S111=255 S112=001

S121=000

S150=000 S151=004 S152=001 S153=001 S154=000 S155=000

S160=010 S161=020 S162=002 S163=003 S164=007

このコマンドを入力すると、全てのレジスタ値を表示してくれる。レジスタ値は通常工場で設定されたままになっているので、用途に応じて設定の変更を行わなければならない。

ATS1=5 (レジスタ1を5にセット)

これは、レジスタ値を変更するためのコマンドである。ATSと入力してから変更したいレジスタNo.を入れ、その後に変更する値を入力する。ここでは、レジスタ1に5を入力していることを意味する。

### 3) レジスタの説明

ここでは特に、uucpでTBを使うために必要なレジスタについて説明する。

- S7     キャリア・トーンを待つ時間(単位sec)  
        リモート側から送られてくるキャリア・トーンが遅い場合、この値の設定を大きくする。     デフォルト40秒、最大59秒
- S50    伝送速度の設定  
        相手側のモデムに設定を合わせる。もしわからなければ、0を設定して自動選択モードとする。相手側のモデムもTBなら、255のFAST操作(PEPモード)にすると、高速伝送が可能となる。
- 0: 自動スピード選択(=自動互換)  
        1: 300bps     操作     (bell 103)  
        2: 1200bps  操作     (bell 212AまたはV.22)  
        3: 2400bps  操作     (V.22 bis)  
        255:FAST操作     (PEPモード)
- S51    インタフェース速度の設定  
        通常、9600bpsに固定しなければならない。そのため、このレジスタを4に設定すると同時に、S66(インタフェース速度の固定を指定するレジスタ)にもビット立てないといけない。     (S51=4,S66=1)
- S52    DTRの制御  
        uucp等でログイン失敗した場合に、DTRフラグを落として電話を切ってモデムの初期化設定を行う。     (S52=2)
- S60    データ・フォーマットの指定  
        通常はデフォルトの0(8bit,none parity, one stop bit)でよい。 (S60=0)
- S110   データの圧縮  
        S50の設定をFAST操作にしている、なおかつ相手側のモデムもデータ圧縮を可能にしている場合のみ有効である。 (S110=1)
- S111   通信プロトコルの設定  
        uucpでしか使用しない場合は、30(gプロトコル)の設定を行う。リモート側のモデムも考慮して、デフォルトの255(リモート側モデムで設定された通信プロトコルに自動的に合わせる)のままにしておいた方が好ましい。

\*) もし、このレジスタの設定方法がよくわからない場合は、1)項のATN?で表示されているレジスタの値に全て変えれば、相手側がTBでuucpにのみ使う場合には別に何も考えることはない。

#### 4) 接続テスト

レジスタの設定が全て終われば、実際にテストに入るのだが、その前に今まで設定したレジスタをメモリーに記憶させる。これをしないでモデムの発信を行ってテストをしてモデムの接続をやめた時に、レジスタの値が工場から出荷された時の値にもどってしまう。そうすると、もう一度最初から設定を変えていかななくてはならない。

AT&W

(メモリーに変更したレジスタ値を記憶)

こうしておく、モデムのテスト中に失敗した場合や、電源を落としてイニシャライズをかけても、変更した箇所がもとのデフォルト値に戻ることはない。ここまで、終わって初めてtipコマンドによる接続確認のテストを行う。

電源を入れる。

① MR,CTSのインジケータが点灯していることを確認

%tip test  
CONNECTED

② DTRインジケータが点灯していることを確認

ATDP0694941830

(モデム用のコマンドを入力して相手側のモデムに発信)

RING

③ OHインジケータが点灯していることを確認

CONNECT FAST

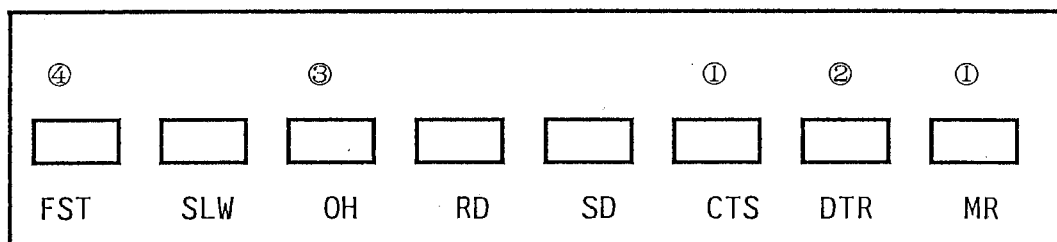
④ FSTインジケータが点灯していることを確認  
(相手側モデムが信号をFASTモードで受信)

リターンを入力

kddlalab (ultrix2.2)

login:

(ログイン・プロンプトが返ってくることを確認)



## 4.4.2 uucpに必要なデータベースの作成

### 1) パスワード・ファイルの設定

ファイル転送を行う場合、uucicoを起動して所定のログイン手順に基づき、モデム等で相手ホストを呼び出す。loginプロンプトがかえってきてuucp用のアカウントでログインすると、相手側のuucicoが自動的に動作して、リモート間の2つのマシンのリンクが電話回線を使ってはられることになる。そのため、相手ホストにuucp用アカウントを設ける必要がある。

```
uucp:*:4:1:UNIX-to-UNIX Copy:/usr/spool/uucppublic:/usr/lib/uucp/uucico
atrlab:*:4:1:UNIX-to-UNIX Copy:/usr/spool/uucppublic:/usr/lib/uucp/uucico
```

通常、unixのシステム側で"uucp"というアカウントが用意されているが、単一のマシンに接続されている場合はいいが、複数のホストと接続されている場合は各ホスト用のアカウントを設ける必要がある。できれば、単一ホストと接続されている場合も、保安面を考えると、外部から認識できないようなユニークなログイン名にすべきだ。

### 2) L.sysファイルの作成

L.sysは、uucicoで相手のホストと通信を始めるために必要な動作について記述する。すなわち、モデムの接続とログイン手順の指定を行う。

[例]

```
kddlal Any DIR 1200 ttya "" ATDP0422586175 CONNECT "" ogin:-EOT-ogin: kddlal ssword:
himitu
```

第1フィールド      通信するホスト名

第2フィールド      接続時間、通常はAny(常時接続可)を指定。もし時間の設定をしたい場合は、以下のような設定をする。

Any900-1700                      毎日午前9時から午後5時まで接続可能

また、1週間に1回とか曜日の指定をする場合は、日曜日から"Su,Mo,Tu,We,Th,Fr,Sa"、ウィクデイはWkと指定できる。このような条件が複数にある場合、"|"でつなぐことができる。例えば、水曜日の午前9時から午後5時までと木、金曜日の午後1時から午後5時までという場合には、次のようにすればいい。

We900-1700|ThFr1300-1700

第3フィールド 接続装置

DIR 直接接続(通常モデムを接続している場合)  
ACU ACUに接続している場合  
PAD Multi-PAD(DDX回線を使用)に接続している場合

第4フィールド 転送速度(接続されているモデムの転送速度)。ここでは、1200bpsになっている。

第5フィールド 接続ポートの指定。ここでは、/dev/ttyaに接続されている。

第6フィールド ログイン手順の記述

先程の例をもとに解説する。

1. ""はヌルストリング(「\r」を送ることを意味する。)
2. 「ATDP0422586175」(モデムの発信コマンド)を送る。
3. 「CONNECT」送られてくるのを待つ。
4. ""はヌルストリング(「\r」を送ることを意味する。)
5. 「ogin:」が送られてくるのを待つ。もしも「ogin:」が送られてこない場合はEOT、すなわちControl-Dを送り、また「ogin:」待つ。
6. 「ogin:」がきたら「kddlal\r」uucp用のログイン名を送る。
7. 「ssword:」が送られてくるのを待つ。
8. 「ssword:」がきたら「himitu\r」と送る。
9. 相手ホストのuucicoと通信をはじめる。

この指定は、接続されているモデムやログインすべき計算機のログイン・プロンプトなどによって変わる。基本的には、これこれを送ってこれこれを待つ、というシーケンスを記述するというのを頭に入れておけばいい。

### 3) L-devicesファイルの作成

tipの場合、/etc/remoteに接続するモデムの転送速度などを記述するが、cuではこのL-devicesがそれにあたる。記述フォーマットは、接続装置の種類、tty番号、転送速度となっている。

```
PAD ttyh0 9600
PAD ttyh1 9600
PAD ttyh2 9600
PAD ttyh3 9600
DIR ttya 9600
DIR ttyb 1200
```

### 4) L.cmdsファイルの作成

リモート・コマンド要求を相手側ホストから受けた場合、セキュリティの面を考えて、実行できるコマンドを制約する必要がある。以下のL.cmdsに記述されているrmailからuuxまでは、メールとニュースのシステムがあれば必然的なものでり、lpr以下のコマンドについては、特に取扱いに憂慮すべきだ。(一般ユーザには隠しておく。)

```
PATH=/bin:/usr/bin:/usr/ucb
rmail
rnews
ruusend
bnproc
unbatchnews
uux
# These next programs are security holes ,but noone seems to care
lpr
who
uusend
finger
cunbatch
```

## 5) L-dailcodesファイルの作成

このファイルはuucicoで通信する時に電話番号を略称するためのファイルである。以下のサンプルは標準に設定されているものである。特に頻繁に電話をかけるエリアには、ここに記述していると便利である。

```
#      ExampleL-dialcodes
# I suppose there are sites that actually use this sort of thing.
boston 149
cb 8-353
chicago 127-
dr 8-374
hog 133-
hoy 135-
ih 8-367
in 8-356
ix 8-372
mh
mv 8-245
```

#### 4.4.3 インストール手順

##### 1) makeの起動

```
%cd /usr/local/src/uucp      (uucpのキットの入っているエリアに移動)

%make                          (エラーが出ずに、正常終了することを確認)

%su                             (できれば、シングル・ユーザ・モードで作業)

%make install                  (実体の更新)
```

##### 2) 機能確認

uucicoのデバッグ・モードでテスト

```
%uucico -x7 -r1 -skddlab

net kddlab (4/13-10:11-4293) DEBUG (Local Enabled)
net kddlab (4/13-10:11-4293) NO CALL (RETRY TIME NOT REACHED)
RETRY TIME (12000) NOT REACHED
net kddlab (4/13-10:11-4293) continuing anyway (debugging)
finds (kddlab) called
getto: call no. ttyAa for sys kddlab
Using DIR to call
Opening /dev/ttyAa      .....①
login called
wanted """"
got: that
send "ATDP034990000"
ATDP034990000
wanted "CONNECT"
RING

CONNECT

got: that

ogin:

ssword:
```



kddlab (ultrix2.2)

ogin: kddlab .....②

ssword: himitu

uucp kddlab (2/10-13:16-5724) SUCCEDED (call to kddlab).....③

|  
|  
|  
|

uucp kddlab (2/10-13:32-6814) OK (conversation complete).....④

### 3) トラブル・シュート

上記のログは、uucicoが正常に起動されて終了するまでの過程をデバッグ・モードで確かめている。うまくインストールが行われていない場合、①②③④の箇所がおかしくなることが多い。その原因と問題解決の方法について述べる。

#### ① ファイル・リード・エラー

L-devicesの記述をもう一度確認。モデムの設定速度とL-devicesの記述が  
あるかどうか?

#### ② ログインの失敗

tipコマンドでログインしてモデムの設定がおかしくないか確認する。もし  
おかしくなければ、tipコマンドでログインした時の操作手順とL.sysの記述  
があるかどうか確認する。

#### ③ ログイン後、uucicoのチェックに引っ掛かる。

通常ログインに成功すれば、uucicoのセキュリティ・チェック等に引っ掛かる  
ことは少ない。この時に起こるエラーとして、以下のものが考えられる。

**HANDSHAKE FAILED(Your host unknown to me.)**

相手のホストがこちらと通信できるようにデータ・ベースの書換えをまだ行  
っていないため、リモート側のuucicoがこちらのホスト名を認識できない。

**BAD READ(expected 'H' got FAIL)**

回線にノイズが多いか、モデムの相性が悪い。モデムのボーレートを下げ  
てもうまくいかない場合は、違う電話線を使うなどの対策が必要。

#### ④ リモート側の uucico 起動後の異常終了

##### Conversation Failed

実際に uucico が起動されてデータの転送を始めだすのだが、その後回線の状態が悪くなり、ノイズが多く発生して uucico 自体異常終了してしまう。接続されているサイトと相談して、他の電話回線に切り替える。

##### OK(conversation complete)後のTIMEOUT

これは uucico が終了(データの送信・受信が全て完了)してからのエラーなので、別に気にする必要はない。これはどういう原因で起こるかという点、System V と BSD 間の uucp による通信だけで、セッションの最後の接続を切るところが違うため、BSD のほうが 1 回余分に回線切断の確認応答を行う。そのため、System V 側の uucico は終了しているのに、BSD が応答をまっているので、毎回 conversation complete のあとにタイムアウトのログが残ってしまう。

## 5. ATR内のネットワーク環境

学研都市への研究所移転にともない、ネットワーク環境の改善及び今後のネットワークの拡張に備えるため、ゲートウェイ・マシンの移行、ネットワーク・アドレスの変更を行った。移転前の環境と移転後の環境を比較しながら、現在のATR内のネットワーク環境について説明する。

### 5.1 ゲートウェイ・マシンの移行

fig.1は、移転前のメール・ニュースの配送経路を示したものである。自動翻訳電話研究所のatr-la(vax8600/ultrix2.0)をゲートウェイとして、メール・ニュースの配送及び外部への送信を行っていた。しかし、一研究所が代表して行ってきたゲートウェイ・サービスを4研究所を統括しているATR-Iが本来行うべきであるという意見が強かったため、ゲートウェイ・サービスをATR-I(企画部)のatrpost(sun3/260)に移行することとなった。(次頁fig.2 参照)

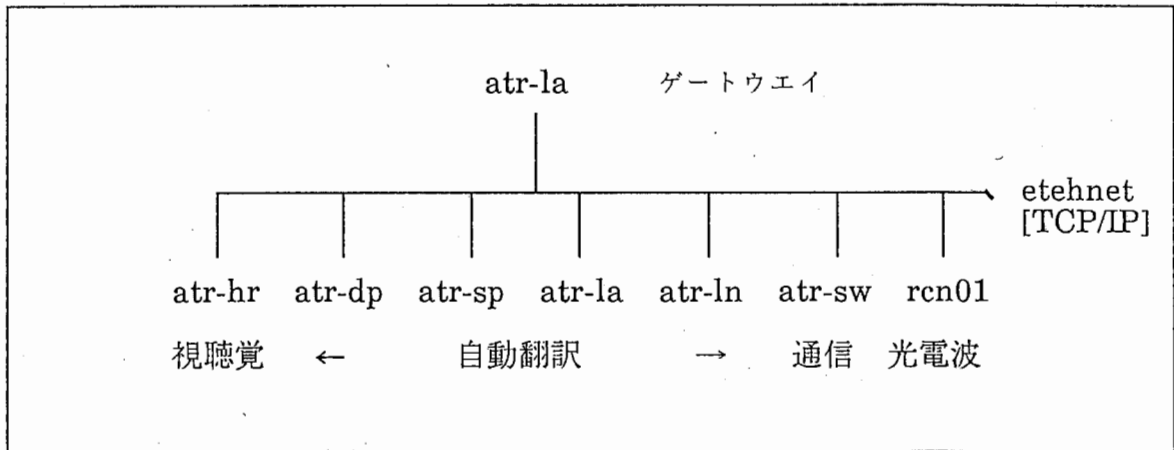


fig.1 ATR内でのメール及びニュースの配送構成図 (移転前)

fig.2は、移転後のメール・ニュースの配送経路を示したものである。完全にゲートウェイ・サービスは、ATR-I(企画部)が行うこととなり、atrpost(sun3/260)がその受け口となっている。そして、各R&Dとも同等のサービスを行うために、1研究所に1メール・ホストを原則として、メール・ニュースの配送先を一カ所に限定している。ただし、内部的にそのメール・ホストの下に階層的にメールなどのサービスは、自由に行うことができる。移転前まで当研究所では、ゲートウェイの下にメール・システムをインストールしているサイトが3つぶら下がっていた。移転後もこれらのサイトにメールのサービスを行うために、下記のようにatr-laを自動翻訳のゲートウェイとして、さらに3つのマシンにメールやニュースの配送するようにしている。

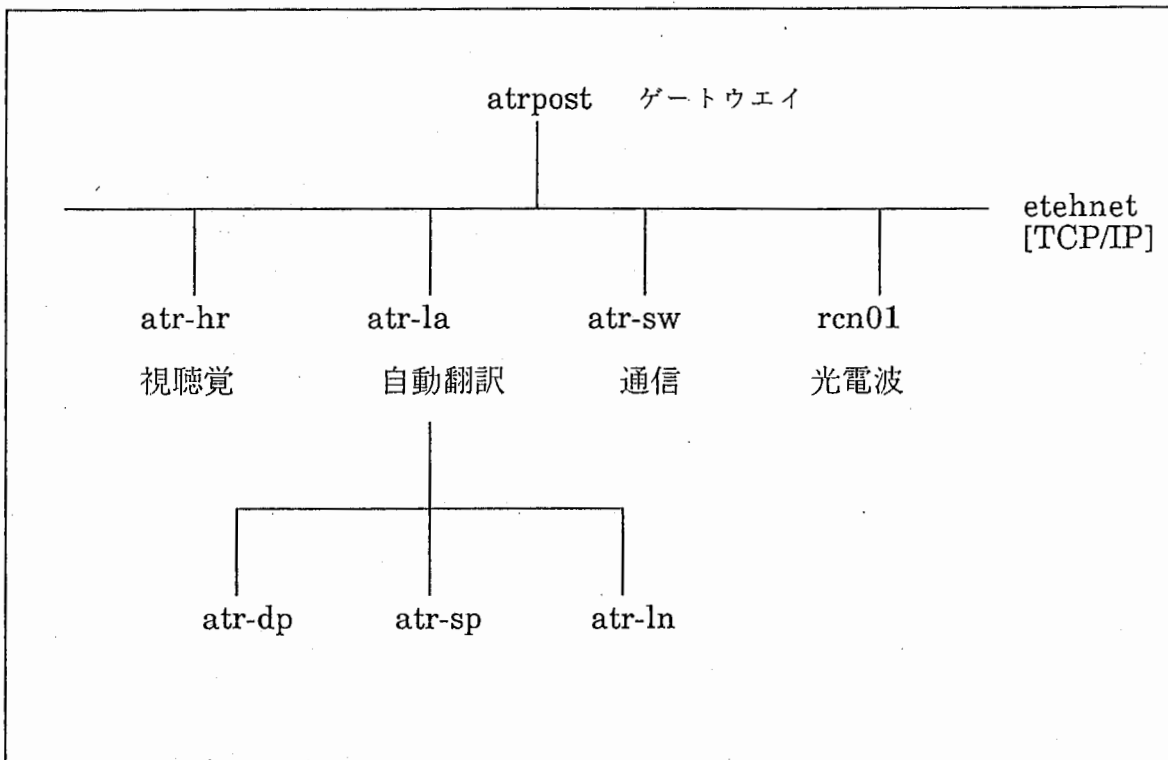


fig.2 ATR内でのメール及びニュースの配送構成図 (移転後)

## 5.2 ネットワーク・アドレスの変更

移転前と同じく1本のイーサネットにすべてのマシンが接続されている現状では、物理的に254のマシンしか接続できず、現在既に200台前後のマシンが接続されていることを考えると、将来的にネットワークに新たにマシンが接続できない可能性もある。そのため、ルーティング・テーブルの設定を行い、いくつかのアドレス体系を作ることにより、254を越えるマシンの接続が可能となる。以下にルーティング・テーブルの設定方法について述べる。

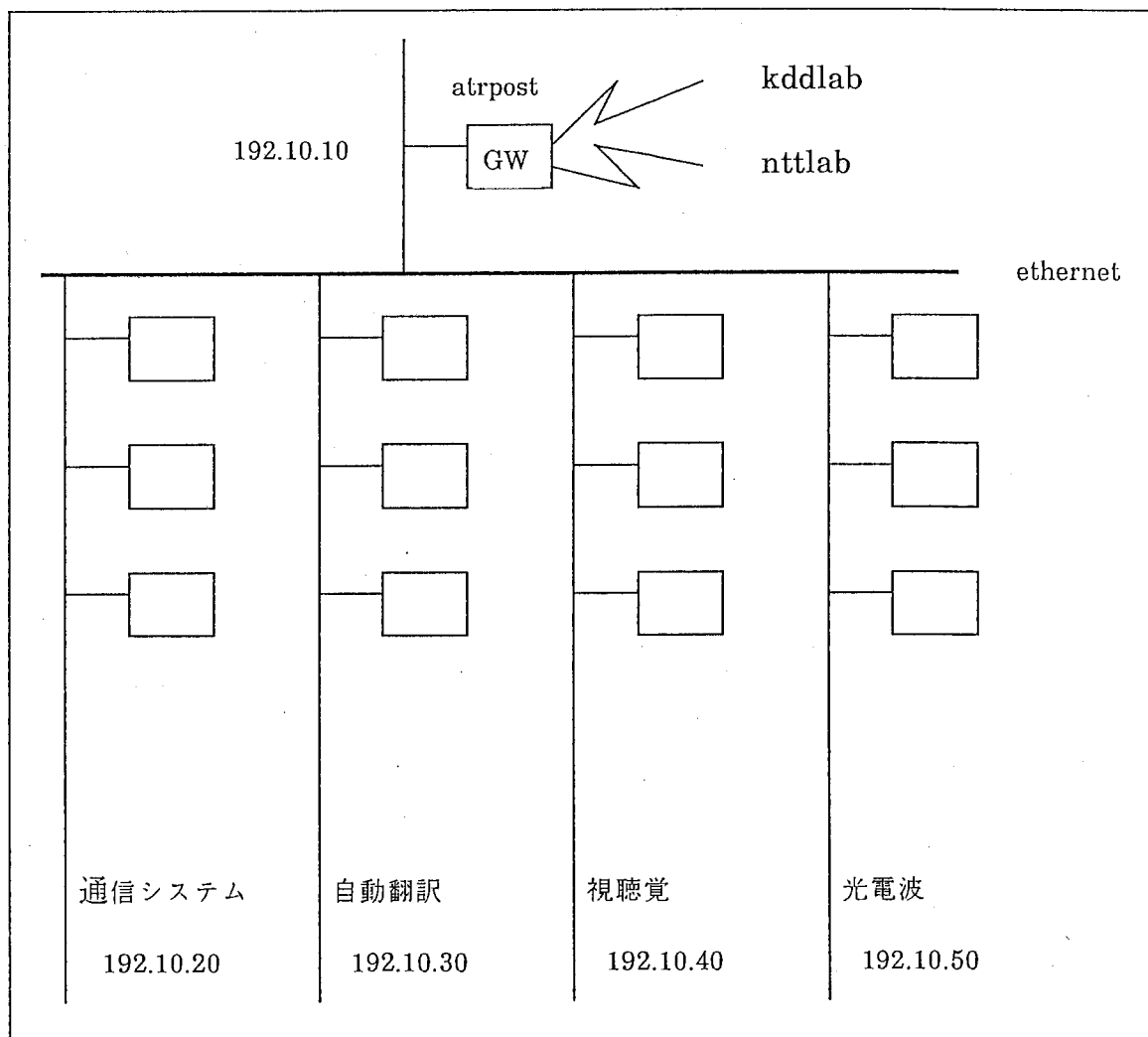


fig.3 新アドレス体系下のネットワーク概要図

- 1) 各R&Dごとにアドレス・テーブル(/etc/hostsファイル)の整理を行う。自動翻訳に割り当てられているアドレスは、192.10.30.1からなので、自動翻訳内のすべてのマシンに新しいアドレスを割り付ける。

```
% cat /etc/hosts
#
# Gateway to kddl,lab,nttlab
#
192.10.10.1      atrpost                #SUN-3 (ATR-I)
#
# Sub gateway (R&D)
#
192.10.20.1     atr-sw                #通信メール・ホスト
192.10.40.1     atr-hr                #視聴覚メール・ホスト
192.10.50.1     atr-rd                #光電波メール・ホスト
#
# Local machine (ATR-ITL)
#
192.10.30.1     atr-la  la           mailhost #
192.10.30.2     atr-sp  sp           ATR-SP  #
192.10.30.3     atr-ln  ln           ATR-LN  #
192.10.30.4     atr-dp  dp           ATR-DP  #
                :
                :
```

- 2) 各マシンのアドレス・テーブル変更

新しいアドレスが割り付けられたら、各マシンの/etc/hostsにコピーする。ローカル内でしか通信しないマシンはその後リブートすれば作業は完了する。もし、メール・ホスト等、他の研究所と通信を行う必要があるマシンでは、次の3)以降の作業を行ってからリブートしなければならない。

- 3) 各R&Dごとのネットワーク・アドレス体系を確立する。  
(データ・ベースにアドレス体系を書き込む。)

```
% cat /etc/networks
#
# Internet networks
#
loop      127                loopback
INTER     192.10.10           ATR Inter
TUUSIN    192.10.20           通信システム
HONYAKU   192.10.30           自動翻訳
SIKAKU    192.10.40           視聴覚
HIKARI    192.10.50           光電波
```

- 4) ルーティング・テーブルの設定

routeコマンドを使ってルーティング・テーブルの設定を行い、実際に他の研究所のマシンと通信ができるかどうか確認する。

```
% /etc/route add INTER atr-la 0
% /etc/route add TUUSIN atr-la 0
% /etc/route add HONYAKU atr-la 0
% /etc/route add SIKAKU atr-la 0
% /etc/route add HIKARI atr-la 0
```

```
% /etc/netstat -r
```

(ルーティング・テーブルが設定されているかどうか確認)

```
Routing tables
Destination      Gateway          Flags    Refcnt  Use
Interface
SIKAKU           atr-la          U        0        84120   de0
INTER            atr-la          U        0        266870  de0
HIKARI           atr-la          U        0        1140    de0
TUUSIN           atr-la          U        0        11441   de0
HONYAKU          atr-la          U        20       5095916 de0
loop             localhost       U        2        126343  lo0
```

```
% telnet 192.10.20.1
```

(他の研究所のマシンとコネクションが確立されることを確認)

```
Trying...
Connected to 192.10.20.1.
Escape character is '^]'.
atr-sw login:
```

## 5) /etc/rcファイルまたは/etc/rc.localの修正

unixのバージョンによって多少ファイルの内容が違うため、rc.localかrcのどちらかでinetdが定義されている箇所より後に、先程のrouteコマンドの設定を書き込む。このファイルにルーティング・テーブルの設定がされていないと、マシンがダウンして立ち上がる時にテーブルをイニシャライズしてしまうため、その度に手でコマンドを入力しなければならない。このファイルの修正が終われば、リブートしてから立ち上がった後、ルーティング・テーブルが設定されているかどうか確認する。

[例]

```
% vi /etc/rc.local
```

```
if [ -f /etc/route ]; then
```

```
    (echo -n 'set-up routing table start ... ') >/dev/console
    /etc/route add INTER atr-1a 0 ; (echo -n 'INTER ') >/dev/console
    /etc/route add TUUSIN atr-1a 0 ; (echo -n 'TUUSIN ') >/dev/console
    /etc/route add HONYAKU atr-1a 0 ; (echo -n 'HONYAKU ') >/dev/console
    /etc/route add SIKAKU atr-1a 0 ; (echo -n 'SIKAKU ') >/dev/console
    /etc/route add HIKARI atr-1a 0 ; (echo -n 'HIKARI ') >/dev/console
    (echo '...done.') >/dev/console
    netstat -r >/dev/console
```

```
fi
```

```
% /etc/shutdown -r now (マシンをリブートする。)
```

```
% /etc/netstat -r (ルーティング・テーブルが設定されている  
かどうか確認)
```

Routing tables

Destination	Gateway	Flags	Refcnt	Use
Interface				
SIKAKU	atr-1a	U	0	84120 de0
INTER	atr-1a	U	0	266870 de0
HIKARI	atr-1a	U	0	1140 de0
TUUSIN	atr-1a	U	0	11441 de0
HONYAKU	atr-1a	U	20	5095916 de0
loop	localhost	U	2	126343 lo0



6. 参考文献

1. プロフェッショナル UNIX 村井 純・井上尚司・砂原秀樹共著(アスキー出版)
2. コンピュータ・ネットワーク bit臨時増刊 石田晴久・徳田雄洋・徳田英幸(共立出版)
3. UNIX マガジン Unix Communication notes(1988年7月から11月連載)  
斎藤明紀・山口 英 (アスキー出版)
4. JUNETの手引 JUNETの手引作成委員会 編

# 索引

[ア]		ニュース投函プログラム	27
アドレス・テーブル	106	ニュースの管理	3
エイリアス	32,56,58,59,62	ニュースの削除	4
エラーメール	3	ニュースの追加	4
エラーログ・ファイル	32	ニュースの定期購読	5
		ニュースの配布範囲指定	32
[カ]		ニュースバッチ送信プログラム	17
外部との接続	3	ネットワーク・アドレスの変更	105
漢字の取扱いについて	38	ネットワーク環境	103
記事のキャンセル	25,44	ネットワーク環境設定ファイル	7
記事の購読	21		
キュー・ファイル	65	[ハ]	
ゲートウェイ・マシンの移行	103	パスワードの設定	6,96
コントロール・ファイル	65	バックアップ・ファイル	65
コンピュータ・ウイルス	8	フォローアップ・ニュース	25
コンフィギュレーション・ファイル	58	プロトコル	60
		ヘッダの自動生成	27
[サ]			
シェル・エスケープ	57	[マ]	
システム間ファイル送受信プログラム	76	未処理ニュース受信プログラム	13
		メイラ	68
システム間コマンド実行プログラム	58,80	メール環境設定ファイル	66
処理済ニュース受信プログラム	12	メール・キュー	60
スプーリング・ファイル	66	メール・システム	52
セキュリティ・チェック	6	メール・システム初期設定ファイル	55,64
セット・ユーザID・ビット	6	メール・システム構成	53
		メール受信機能	52,54
[タ]		メール送信機能	52,54
タイトル	24	メール配送プログラム	58
データの圧縮	4,17	メール・ボックス	66
データ・ファイル	65	モデムの設定	93
デコード	12		
テスト・モード	60	[ヤ]	
トライル・ブレーザ	93	ユーザの管理	3
[ナ]		[ラ]	
ニュースグループ	5,19,24,25,31,35	リプライ・メール	26,57
ニュースグループ管理ファイル	32,56,58	ルーティング・テーブルの設定	107
		ロック・ファイル	2,65
ニュース購読プログラム	21		
ニュース削除プログラム	15,28		
ニュース・システム	9		
ニュース送受信プログラム	14		

[A]		[J]	
active	15,20,21,22,28,29,31,41,43	JIS code	28
alias	32,56,58,59,62	junk	4,17
alias.dat	6,7		
alias.dir	58,62	[L]	
alias.pag	58,62	L.cmds	81,88,98
		L-devices	79,87,98
[B]		L-dailcodes	79,87,99
batched	15,17,36	log	15,33,43
		LOGFILE	45,79,85
[C]		L.sys	79,88,96
C.	89		
checknews	20,21	[M]	
compress	4,17	mail	54,55,56,58,59
Configure	49	mailconf	72
control	89	mailer.dat	68
crontab	6,7	Mail.help	55,64
		mailpath	34
[D]		mailq	1,60
D.	89	Mail.rc	55,56,64
D. [hostname]	89	mbox	55,56
D. [hostname]X	89	mqueue	1,58,59
dec code	38		
desc.dat	69	[N]	
D/TR	91	netstat	107,108
		networks	107
[E]		newaliases	63
encode	17	news	3,15,17,21,36
ERRLOG	79,85	newsgroups	27,34,43
errlog	15,32,43	newsgroups.sh	41
expiration date	29		
expire	3,15,28,29,31,33,43	[O]	
		oactive	31
[H]		ohistory	28
help	21,33	oldnews	3
history	15,28,29,33		
history.dir	28	[P]	
history.pag	28,33	passwd	7
hosts	106	patch	39
hosts.equiv	7	Pnews	50,51
		Pnews.SH	48
[I]		postnews	27
inews	12,13,14,27,29,41		

[R]			[V]	
rc		107	verbose mode	29,56,60
rc.local		107	vnews.help	36
readnews	20,21,33,36,46			
recnews		13		
remote		92		
rmail		54,81	[X]	
rn		46,51	X.	90
rnews	28,29,81			
route		107		
ruusend		81		
[S]			.mailrc	55,56,66
sendbatch		4,17,44	.newsrc	20,21,22,24,25,26,36,46
sendmail	1,2,6,8,19,54,55,58		.rhosts	7
sendmail.cf		58,59,62	.rnews	15,29
sendmail.fc		58,59,62	.sys.nnn	15
sendmail.hf		58,59,62		
sendmail.st		58,59,62		
sendnews		12,19,58		
seq		15		
set		56		
shutdown		107		
smtp		60,62		
STST		82,86		
sulog		7		
sys	15,35,41,43,44			
SYSLOG		79,83,85		
syslog		58,65		
[T]				
TB(trailBlazer)		93		
tip		92		
TM.		89		
ttys		7,91		
[U]				
users		36		
uucico		1,78		
uucp		74,76		
uurate		83		
uurec		12		
uusnap		1,82		
uux	17,54,58,80			

付録 A

ニュースインストレーションガイド(USENET版)

## USENET Version B Installation

Matt Glickman  
Computer Science Division  
Department of Electrical Engineering and Computer Sciences  
University of California  
Berkeley, California 94720

Revised by Mark Horton for version 2.10

Revised by Rick Adams for version 2.11

### 1. Introduction.

This document is intended to help a USENET site install and maintain the network news software. Please ask questions of Rick Adams\*; such questions will help to point out areas that need to be addressed here.

The overall order of things to do is:

- (a) Find somebody to link up with. You need a network connection of some kind, for example, ARPANET or UUCP. If you must use UUCP and have no connections, you must have at least a dialup and preferably a dialer, and find someone willing to call your machine. The USENET directory may be helpful in finding some other site geographically near yours to hook up to.
- (b) Create a `localize.sh` script to make local changes to the `makefile` and `defs.h` files. (Section 2 gives more details about creating `localize.sh`.) Once you're finished editing `localize.sh`, create a `defs.h` and `Makefile` tailored for your site with the command

```
sh localize.sh
```

Inspect `defs.h` and `Makefile` to ensure that all your local customizations got into your final versions. If you saw a "?" when you ran `localize.sh`, one or both of the files is certainly wrong. It's a good idea to anchor the patterns in `localize.sh`'s `ed(1)` scripts, especially in its `Makefile` editing lines. For instance, use `/^UUXFLAGS/` instead of `/UUXFLAGS/`.

---

\* ARPANET: `rick@seismo.CSS.GOV`, UUCP: `seismo!rick`

- (c) Compile the software using the make(1) command.
- (d) Su(1) and type "make install". This will copy the files out to the right place and make directories containing most of the important files. It will configure you in with a connection to oopsvax via UUCP links. This is undoubtedly wrong, so you will have to configure links as needed. If this is your initial installation or you are upgrading from a version older than 2.11, do "make update". This will cause various checks to be performed on important files in LIBDIR. The results will be reported to you. If you are not sure if you should do "make update", do it. It will not hurt anything if you have already done it.
- (e) After editing the configuration table, get your contact at the other end of the link to add you to their netnews sys file.
- (f) Post a message to the to.sysname newsgroup which should be set up to go only to the site you are linked to, as a test. Have the other person send a message to your system using the same mechanism. If this doesn't work, find the problem and fix it. (Please don't use misc.test unless there is no alternative. It is almost always possible to use test, or to.sysname or some local.test group, instead of misc.test.)
- (g) Fill out a USENET directory form (the file dirform in the misc directory). Post a copy to the USENET newsgroup news.newsite and mail a copy to cbsgd!uucpmap.
- (h) Format the document "How to Read the Network News" (the file howto.mn in the doc directory), the document "How to Use USENET Effectively" (the file manner.mn in the doc directory) and the document "Copyright Law" (the file copyright.mn in the doc directory) and post them to your general newsgroup with a long expiration date. You can use inews(1) or postnews(1) to do this.
- (i) It will probably be necessary to fix your uucp commands to allow rnews and to support the -z and -n options (if you are lucky enough to have the source).

## 2. Installation

### 2.1. Configuration

Local configuration of the USENET version B software requires you to edit a few files. Most importantly, the defs.h and Makefile files must be created from their templates defs.dist and Makefile.dst. You should create a shell script called localize.sh which copies the files and makes local changes to the copies. Even for a completely vanilla site, some changes will be necessary. For example, your script should start with localize.v7 or localize.usg. You should include the name of the local organization (MYORG) and the uid of the local news superuser (ROOTID). You should also choose how your hostname will be determined. If you are a USG site, define UNAME in defs.h. If you are running 4.[23] BSD, define GHNAME in defs.h. If you have your UUCP name in a file, define UUNAME in defs.h to the name of that file. Otherwise, news will look in the file /usr/include/whoami.h for a line of the form

```
#define sysname your-sysname
```

If you are running System 3 or System 5, you are a USG site. Otherwise, unless you are in AT&T, you are probably a V7 site. The previously mentioned defines are the only modifications that are necessary to install news at your site. However, you will probably want to change some of the ones listed below. If your compiler does not accept "(void)", the simplest thing to do is add "Dvoid=int" to the CFLAGS line in the Makefile.

A sample localize shell script can be found in localize.sample. The most important parameters are:

#### 2.1.1. ROOTID

The numerical uid of the person who is the news super user. This should not be set to 0. Normally it is set to the uid of the news contact person for the site. If it is not defined, the uid of NOTIFY will be looked up in /etc/passwd and used instead.

#### 2.1.2. NUMASK

Mask for umask(2) system call. Set it to something like 022 for a secure system. Unsecure systems might want 002 or 000. This mask controls the mode of news files created by the software. Insecure modes would allow people to edit the files directly.

#### 2.1.3. DFLTEXP

The default number of seconds after which an article will expire. Two weeks (1,209,600 seconds) is the default choice. If you wish to expire articles faster than two weeks, it is recommended that you use the -e flag to expire instead of decreasing DFLTEXP.

#### 2.1.4. HISTEXP

Articles which were posted more than HISTEXP ago are considered too old and are moved into the junk directory. This is because they are too old to be in the history file, so it is impossible to tell if they really should be accepted or are endlessly looping around the network. (This was theoretically possible before this feature was added.) The articles are removed after DFLTEXP seconds, but a copy of their "Message-ID" is kept in the history file for HISTEXP seconds (the default is 4 weeks). If you are a leaf node, you do not need to have HISTEXP longer than DFLTEXP. However, major nodes with more than one news feed should have HISTEXP set to the default 4 weeks if possible.

#### 2.1.5. DFLTSUB

The default subscription list. If a user does not specify any list of newsgroups, this will be used. Popular choices are all and general,all.general.

#### 2.1.6. TMAIL

This is the version of the Berkeley Mail(1) program that has the -T option. If left undefined, the -M option to readnews(1) will be disabled.

#### 2.1.7. ADMSUB



This newsgroup (or newsgroup list) will always be selected unless the user specifies a newsgroup list that doesn't include ADMSUB on the command line. That is, as long as the user doesn't use the -n flag to readnews on the command line, ADMSUB will always be selected. This is usually set to general,all.announce. (The intent of this parameter is to have certain newsgroups which users are required to subscribe to. A typical site might require general.)

#### 2.1.8. PAGE

The default program to which articles should be piped for paging. This can be disabled or changed by the environment variable PAGER. If you have it, the Berkeley more(1) command should be used, since the + option allows the headers to be skipped.

#### 2.1.9. NOTIFY

If defined, this character string will be used as a username to send mail to in the event of certain control messages of interest. (Currently these are newgroup, rmgroup, sendsys, checkgroups, and senduuname.) As distributed, mail will be sent to user usenet. It is recommended you create such a mailbox (have it forwarded to yourself) if possible, since this makes it easier for another site to contact the site administrator for your site. If you are unable to do this (e.g., you are not the super user) you should change this name to yourself. Also, messages about missing or extra newsgroups are mailed to this user by the checkgroups control message.

#### 2.1.10. DFTXMIT

This is the default command to use to transmit news if no explicit command is given in the fourth field of the sys file. It normally includes uux(1) with the z option. You should install this modification to UUCP at once; otherwise your users will be bombarded with annoying uux completion messages. However, you can turn this off to get news installed.

#### 2.1.11. UXMIT

This is the default command used if the U flag is present in the flags portion of a sys file line. In this case, the second "%s" refers to the name of a file in the news spool area, not a temporary file. It can usually only be used when local modifications are made to the uucp system, such as the c option to uux.

#### 2.1.12. DFTEDITOR

This is the full path name of the default editor to use during followups and replies. It should be set to the most popular text editor on your system. As distributed, vi(1) is used.

#### 2.1.13. UUPROG

If this is defined, it will be used as a command to run when the senduuname control message is sent around. Otherwise the command uuname(1) will be run. Normally, this program should be placed in LIBDIR.

#### 2.1.14. MANUALLY

If this is defined, incoming rmgrou messages will not automatically remove the group. News will instead mail a message to NOTIFY advising that the group should be removed. If you define MANUALLY, you should have NOTIFY defined. MANUALLY is defined by default to protect you against accidental or malicious removal of an important newsgroup.

#### 2.1.15. NONEWGROUPS

If this is defined, incoming newgroup messages will not automatically create the group. News will instead mail a message to NOTIFY advising that the group should be created. If you define NONEWGROUPS, you should have NOTIFY defined. NONEWGROUPS is undefined by default to make it easier to automatically maintain the news system.

#### 2.1.16. BATCH

If set, this is the name of a program that will be used to unpack batched articles (those beginning with the character "#"). Batched articles normally are files reading

```
#! rnews 1234
article containing 1234 characters
#! rnews 4321
article containing 4321 characters
...
```

Batching is strongly recommended for increased efficiency on both sides.

#### 2.1.17. SPOOLNEWS

If this is defined, all incoming news (inews or rnews) will be placed in the directory SPOOLDIR/.rnews for later processing by rnews -U. This is useful when you want to appear to accept incoming news as soon as it arrives (to get it out of the uuxqt queue for example), but want to defer the actual processing until later. If you define this option, make sure you invoke rnews -U from crontab regularly or you may wonder where your news is going!

#### 2.1.18. LOCALNAME

Most systems have a full name database on line somewhere, showing for each user what their full name is. Most often this is in the gecost field of /etc/passwd. If your system has such a database, LOCALNAME should be left undefined. If not, define LOCALNAME, and articles posted will only receive full names from local user information specified in NAME or \$HOME\$HOME/.name by the user. If you have a nonstandard gcos format (not finger(1) or RJE) it will be necessary to make local changes to fullname.c as appropriate on your system.

#### 2.1.19. INTERNET

If this is defined, replies will use the "From" or "Reply To" headers of the news article. Otherwise, leave it disabled and replies will use the "Path" header.

You can define this option even if your site does not understand internet style headers. The

file LIBDIR/mailpaths contains a line beginning with "internet". The rest of this line is a string for printf to use to route your mail to a site that does understand internet style headers. If your site understands internet style headers, this string would just be "%s". If not, it might look like "seismo!%s" if seismo was your nearest site that understands that format.

#### 2.1.20. MYDOMAIN

When generating internet addresses, this domain will be appended to the local site name to form mailing address domains. For example, on system ucbox with user root, if MYDOMAIN is set to ".UUCP", addresses generated will read "root@ucbox.UUCP". If MYDOMAIN is ".Berkeley.EDU", the address would be "root@ucbox.Berkeley.EDU". If your site is in more than one domain, use your primary domain. The domain always begins with a period. If your site name contains the domain, everything after the first "." will be stripped off to get the news host name, so you still must define your domain name with MYDOMAIN.

#### 2.1.21. CHEAP

Do not chown(1) spool files to news. This will cause the owner of the file to be the person that started the inews process. This is used for obscure accounting reasons on some systems.

#### 2.1.22. OLD

Define this if any of your USENET neighbors run 2.9 or earlier versions of B news. It will cause all headers written to contain two extra lines, "Article-I.D." and "Posted", for downward compatibility. Once all your neighbors have converted, you can save disk space and transmission costs by turning this off. It is strongly encouraged that they convert. 2.11 is much faster than 2.9. The performance difference is dramatic.

#### 2.1.23. UNAME

Define this if the uname(2) system call is available locally, even though you are not a USG system. USG systems always have uname(2) available and ignore this setting.

#### 2.1.24. GHNAME

Define this if the 4.[23] BSD gethostname(2) system call is available. If neither UNAME or GHNAME is defined, inews will determine the name of the local system by reading /usr/include/whoami.h.

#### 2.1.25. UUNAME

Define this to be the name of the file in which you store your UUCP name. (If you store your UUCP name in a file.)

#### 2.1.26. V7MAIL

Define this if your system uses V7 mail conventions. The V7 mail convention is that a mailbox contains several messages concatenated, each message beginning with a line reading "From user date" and ending in a blank line. If this is defined, articles saved will have these lines added so

that mail can be used to look at saved news.

#### 2.1.27. SORTACTIVE

Define this if you want the news groups presented in the order of each person's .newsr(5) instead of the active file.

#### 2.1.28. ZAPNOTES

Define this if you want old style notesfile id's in the body of the article to be convert into "Ni-ID" fields in the header and "Orphaned Responses" to be rejected.

#### 2.1.29. DIGPAGE

If this is defined, vnews(1) will attempt to process the subarticles of a digest instead of treating the article as one big files.

#### 2.1.30. DOXREF

Define this if you are using rn(1). Rn uses this option to keep from showing the same article twice.

#### 2.1.31. MULTICAST

If your transport mechanism supports multi-casting of messages defune this. Currently ACSNET is the only network that can handle this.

#### 2.1.32. BSD4\_\_2

Define this if you are running 4.2 or 4.3 BSD UNIX.  
of the article to be converted into "NfID" fields in the header

#### 2.1.33. BSD4\_\_1C

Define this if you are running 4.1C BSD UNIX.

#### 2.1.34. LOCKF

Define this if your system supports the lockf() system call as defined in the System V Interface Definition. Be warned, many System V UNIX systems do not comply with the SVID.

#### 2.1.35. ALWAYSALIAS

Normally, the LIBDIR/aliases file is processed only for newsgroups that are not found in the LIBDIR/active file. If this is defined, the LIBDIR/aliases will be checked for ALL newsgroups. This is useful for a transition period when massive newsgroup renaming is occurring, but should normally be undefined as it is computationally expensive.

#### 2.1.36. SENDMAIL

Use this program instead of recmail(8) for sending mail.

#### 2.1.37. MMDF

Use MMDF instead of recmail for sending mail.

---

\*UNIX is a registered trademark of AT&T.

#### 2.1.38. MYORG

This should be set to the name of your organization. Please keep the name short, because it will be printed, along with the electronic address and full name of the author of each message. Forty characters is probably a good upper bound on the length. If the city and state or country of your organization are not obvious, please try to include them. If the organization name begins with a "/", it will be taken as the name of a file. The first line in that file will be used as the organization. This permits the same binary to be used on many different machines. A good file name would be LIBDIR/organization. For example, an organization might read "AT&T Bell Labs, Murray Hill", "U.C. Berkeley", "MIT", or "Computer Corp. of America, Cambridge, Mass".

#### 2.1.39. HIDDENNET

If you want all your news to look like it came from a single machine instead of from every machine on your local network, define HIDDENNET to be the name of the machine you wish to pretend to be. Make sure that you have your own machine defined as ME in the sysfile or you may get some unnecessary article retransmission.

#### 2.1.40. NICENESS

If NICENESS is defined, rnews does a nice(2) to priority NICENESS before processing news.

#### 2.1.41. FASCIST

If this is defined, inews checks to see if the posting user is allowed to post to the given newsgroup. If the username is not in the file LIBDIR/authorized then the default newsgroup pattern in the symbol FASCIST is used.

The format of the file authorized is:

```
user:allowed groups
```

For example:

```
root:all.all,all
naughtyperson:junk,talk.politics.misc
```

operator:!all.all,general,test,comp.unix.questions

An open environment could have FASCIST set to all and then individual entries could be made in the authorized file to prevent certain individuals from posting to such a wide area.

Note that a distribution of all does not mean to allow posting only to local groups - all includes all.all. Use all,!all.all to get that behavior

#### 2.1.42. SMALL\_\_ADDRESS\_\_SPACE

Define this if your machine has 16 bit (or smaller) pointers. If you are on a PDP11\*, this is automatically defined.

#### 2.1.43. ORGDISTRIB

This can be defined to be the name of the distribution that is "local" to your organization. If a control message arrives with ORGDISTRIB as the distribution, it is believed and acted upon. This is useful for organizations with one "main" news machine and several subordinate ones. All machines in that organization could define NONEWGROUPS, MANUALLY and also ORGDISTRIB. Then, when a control message arrives at the main machine, it would be mailed to NOTIFY. If that person determined that it was a legitimate message, they could resubmit the message to their local machine with a distribution of ORGDISTRIB. They would not have to submit this message to the other machines, as they would believe it because of the distribution. They would reject the control message if it had any other distribution.

### 2.2. Makefile

There are also a few parameters in the Makefile as well. These are:

#### 2.2.1. OSTYPE

This is the type of UNIX system you are using. It should be either V7 or USG. Any BSD system is V7. Any System 3 or System 5 system is USG. This is normally set by localize.sh.

#### 2.2.2. NEWSUSR

This is the owner (user name) of inews. If you are a superuser, you should probably create a new user id (traditionally news) and use this id. If you are not a superuser, you can use your own user id. If you are able to, you should create a mail alias usenet and have mail to this alias forwarded to you. This will make it easier for other sites to find the right person in the presence of changing jobs and out of date or nonexistent directory pages. NEWSUSR and ROOTID do not need to represent the same user.

#### 2.2.3. NEWSGRP

This is the group (name) to which inews belongs. The same considerations as NEWSUSR apply.

---

\*PDP11 is a trademark of Digital Equipment Corporation.

#### 2.2.4. SPOOLDIR

This directory contains subdirectories in which news articles will be stored. It is normally `/usr/spool/news`.

For each newsgroup (e.g. `comp.unix`) there will be a subdirectory `/usr/spool/news/comp/unix` containing articles, whose file names are sequential numbers, e.g., `/usr/spool/news/comp/unix/1`, etc.

Each article file is in a mail-compatible format. It begins with a number of header lines, followed by a blank line, followed by the body of the article. The format has deliberately been chosen to be compatible with the ARPANET standard for mail documented in RFC 822.

You should place news in an area of the disk with enough free space to hold the news you intend to keep on line. The total volume of news in the network wide groups currently runs about 1 Mbyte per day. If you expire news after the default 2 weeks, you will need about 14 Mbytes of disk space (plus some extra as a safety margin and to allow for increased traffic in the future). If you only receive some of the newsgroups, or expire news after a different interval, these figures can be adjusted accordingly.

#### 2.2.5. BATCHDIR

This directory will contain the list of articles to send to each system. It is normally `/usr/spool/batch`.

#### 2.2.6. LIBDIR

This directory will contain various system files. It is normally `/usr/lib/news`.

#### 2.2.7. BINDIR

This is the directory in which `readnews`, `postnews`, `vnews`, and `checknews(1)` are to be installed. This is normally `/usr/bin`. If you decide to set `BINDIR` to a local binary directory, you should consider that the `rnews` command must be in a directory that can be found by `uuxqt`, which normally only searches `/bin` and `/usr/bin`.

#### 2.2.8. UUXFLAGS

These are the flags with which `uux` will be called.

#### 2.2.9. LNRNEWS

This is the program used to link `rnews` and `inews`. If you have symbolic links, you can replace the "ln" with "ln s".

### 2.2.10. SCCSID

If this is defined, sccs ids will be included in each file. If you are short on address space, don't define this.

## 3. FILES

This section lists the files in LIBDIR and comments briefly what they do.

### 3.1. active

This is a list of active newsgroups. It is automatically updated as new newsgroups come in. The order here is the order news is initially presented by readnews, so you can edit this file to put important newsgroups first. If you have SORTACTIVE defined, after the first time the user invokes readnews, it will be presented in the order of the user's .newsrc. Each line of the active file contains four fields, separated by a space: the newsgroup name, the highest local article number (for the most recently received article), the lowest local article number that has not yet expired, and a single character used to determine if the user can post to that newsgroup. If the character is "y" the user is permitted to post articles to that group. If the character is "n" the user is not permitted to post articles to that group. If the character is "m" the newsgroup is moderated and may not be posted to directly. Instead, articles posted to that newsgroup are automatically mailed to the moderator of that group. (This field takes the place of the ngfile in earlier versions of news.) Local article numbers begin at 1 and count sequentially within the newsgroup as articles are received. They do not usually correspond to local article numbers on other sites. The article numbers are always stored as a five digit number (with leading zeros) to allow updating of the file in place.

The active file should contain all active world wide newsgroups. You should use the sys file to keep out unwanted newsgroups.

### 3.2. aliases

This file is used to map bad newsgroup names to the correct ones. (For example, rec.jokes is mapped into rec.humor). Each line consists of two fields separated by white space. If the first field is found in the newsgroup list of the incoming article, it is changed to the second field. This change takes place in the article before it is passed on to other systems, not just locally.

### 3.3. batch

This program reads a list of filenames of articles and outputs the articles themselves. It is typically used by the shell script sendbatch.

### 3.4. caesar

This is a program to do Caesar decoding of rotated text, on a line by line basis. The standard input is copied to the standard output, rotating each line according to a static single letter frequency table. If an integer argument is given (e.g., 13), every line is rotated by that argument,



without regard to letter frequencies. This program is invoked by the D readnews command. It is also used by postnews with the "13" argument to encode selected material for posting.

### 3.5. checkgroups

Checkgroups is a shell file to aid in automatically checking the accuracy of your active file. It is executed by the checkgroups control message and mails a list of out of date newsgroups to the person defined by NOTIFY. It also updates the newsgroups file that is used by postnews as a helpfile for newsgroup selection.

### 3.6. compress

This program does a modified Lempel-Ziv data compression. It is used by the compressed batching scheme. It averages 50% compression on a typical batch of news.

### 3.7. distributions

This is a list of distributions that are valid for your site. Each line has two fields separated by the first space on the line. The first field is the name of the distribution (e.g., usa, na, etc.). The second field is text describing the distribution. As distributed, this file is only correct for sites in the USA. You should examine this file and add or delete the appropriate distributions.

### 3.8. encode

This program transforms an 8bit binary file into a file suitable for sending over a link that only allows 7bit characters. It is used by sendbatch c7.

### 3.9. errlog

This file contains the "important" error messages found in the log file. These errors usually indicate that something was wrong with an article. This file should be watched closely. The log file contains much more verbose information and it is often difficult to detect errors in it.

### 3.10. expire

This program expires old articles and archives them if archiving is selected. It is typically run once a day from cron(8).

### 3.11. help

This contains a list of commands printed when an illegal command is typed to readnews.

### 3.12. history

A list of every article that has come in to your system. It is used to reject articles that come in for the second time (presumably via a different path). This file will grow but is cleaned out by the expire(8) command.

### 3.13. history.d

On USG systems, this directory contains 10 files (history.[0-9]) which are used as part of a simple hashing algorithm to speed up history searches. Since V7 systems have DBM, this is not used on V7 systems.

### 3.14. history.dir, history.pag

These two files are used on V7 systems as a hashed version of history, containing the message id's of all articles in history. They are only used if DDBM and ldbm appear in Makefile.

### 3.15. inews

This is the program that actually sends and receives news. All other news submitting programs interface eventually with it. It is not intended to be used directly by a human, so it is no longer in BINDIR.

### 3.16. log

If present, a log of articles processed and error conditions is kept here. This file grows without limit unless cleaned out periodically. The trimlib script in misc can be invoked by cron(8) daily or weekly to keep the log short.

### 3.17. mailpaths

This file contains the mail paths to a Usenet backbone site and a site that can handle internet style mail addresses. Each line consists of two fields. The first field is either the keyword backbone or the keyword internet. The second field is a string to be used by printf to create a mail path. As distributed, they are almost certainly wrong. You will need to modify the paths so they work from your site. As an example, suppose that your site's closest backbone site was seismo and the closest site that understood the internet mail syntax was decuac. Your mailpaths file would look like:

```
backbone seismo!%s
internet decuac!%s
```

The backbone path is used when posting articles to moderated groups. A mail alias exists on almost all backbone sites to forward mail to the proper moderator for the group. The mail aliases should always be up to date on these sites, so when a group changes moderators, only the backbone sites need to update any files. The mail alias is the newsgroups name with "." changed to "". Using the above example mailpaths file, a submission to comp.sources.unix would be mailed to seismo!comp-sources-unix, where it would be forwarded to the current moderator for that group.

The internet keyword is used when the INTERNET option is defined for replying to news articles. A site whose mailer does not understand internet syntax can (and should) define INTERNET so that mailed replies use the From: line of the article instead of the unreliable Path: line.

### 3.18. newsgroups

This file is displayed by postnews when a user hits ?in response to its request for newsgroups. It is also used by vnews when it displays the newsgroup name. It is updated automatically by the checkgroups control message.

### 3.19. notify

This file allows a site administrator to declare which types of control messages they wish to notification on and which they don't. The format is two words separated by white space. The first word is the name of the control message type (like checkgroups). The second word (if it exists) is the address to mail the notification to. If the second word does not exist then it is assumed that no mail is desired from this type of control message. If the LIBDIR/notify file doesn't contain any recognizable control message types or is empty, then it is assumed that no mail is desired from any type of control message.

The key word "all" may be used to select what address all known control message types should be sent to. Since parsing of the file is sequential, the key word "all" could be first in the file and specific declarations specifying differing addresses could follow which would override any previous listing. This example of LIBDIR/notify declares that mail will be sent to "usenet" for all control message types except the sendsys type. Sendsys messages will be sent to "security".

```
# Comments are ok since unrecognized message types are ignored.  
all      usenet  
sendsys  security
```

This example will only send mail to "usenet" for the checkgroups type of control message. All others will not have any mail associated with them.

```
# Comments are ok since unrecognized message types are ignored. checkgroups usenet
```

(This overrides the NOTIFY option in defs.h). Having a null file is useful if one person administers several systems and does not want multiple copies of control message notifications.

### 3.20. oactive, ohistory, ohistory.dir, ohistory.pag

These are copies of the corresponding active, history, history.dir, and history.pag files before expire ran. They are kept in case something happens to the originals.

### 3.21. recmail

This program can serve as a link between news and your local mailer. If you have sendmail(8), don't use recmail. Sendmail is much more useful.

### 3.22. recnews

A program which allows you to send mail to get news posted. You usually need to run sendmail or delivermail(8) to be able to use this.

### 3.23. recording

A list of newsgroup classes and filenames to display recordings for. The recording feature is analogous to the recordings played in some areas when you dial directory assistance, trying to be annoying and make you think twice. Recordings on certain newsgroups are intended to remind the user of the rules for the newsgroup, or, in the case of a company worried about letting proprietary information out, reminding authors that anything they say is seen outside the company and so proprietary information should not be included.

The file contains one line per recording. The line contains two fields, separated by a space. The first field is the news group class (e.g., talk.all), the second field is the name of the file containing the recorded message. If the file name does not begin with a slash, it will be searched for in LIBDIR. Sample recording files can be found in the misc directory.

### 3.24. rmgroup

This shell file should be used to remove any groups that are no longer used.

### 3.25. sendbatch

This shell file is used to send batched articles to other systems. It is typically run from cron(8). See the manual page for more details.

### 3.26. sendnews

A program to send news internally from one computer to another. It is useful if you must use mail links to transmit articles.

### 3.27. seq

This file contains the current sequence number for your system. It is used to generate unique article id's.

### 3.28. sys

This file contains a list of all your neighbors, which news groups they get, and how to send news to them. The format is documented below.

### 3.29 unbatch

This program is used to unbatch the incoming batched news and feed each article to rnews. It's horrible and will go away in the future.

### 3.30. users

A list of users that have read news on your system.

### 3.31. uurec

A program to receive news sent by sendnews(8).

### 3.32. vnews.help

This is the helpfile used by vnews.

## 4. Setting Up Links

There are two basic types of links for exchanging news: those that use mail and those that don't. The ones that use mail are more indirect, yet more versatile, while the ones that don't are simpler. The default method does not use mail, so that is discussed first.

### 4.1. Non-mail Links

The basic theory behind a nonmail link is that the rnews program is invoked on the remote system with the article being transmitted as the standard input. This is possible on several networks, but the most common implementation is via the UUCP network. Using the uux command, the command which is forked to the shell looks like:

```
uux -r -z remotesys!rnews < article
```

This is the default transmission method. In order to set up such a link, obviously a UUCP link with the remote system must be in effect. In addition, rnews must be available and executable by uuxqt on the remote machine. In most cases, this means that rnews must be in /usr/bin so uux can find it. Also, the list of allowed UUCP commands (in /usr/src/usr.bin/uucp/uuxqt.c or /usr/lib/uucp/L.cmds, depending on the version of UUCP) should be checked to make sure that rnews is an allowed command.

Other networks that allow remote execution include the BERKNET, BLICN (usend(1)), many Ethernets, and the NSC hyperchannel (nusend(1)). It is important, however, that a spooling mechanism be available. Otherwise, if system A tries to send an article to system B via a remote execution command, and B is down, the article could be lost. Spooling arranges that the system will try again when B comes back up.

### 4.2. Mail Links

When using mail to transmit articles, two intermediary programs are necessary. These are sendnews and uurec(8). The idea is that when system A wants to send an article to system B, the sys file on system A has an entry for system B such as:

```
LIBDIR/sendnews -a rnews@B
```

which runs sendnews on the article. The -a option specifies that the mail should be formatted for the ARPANET. Sendnews packages the article and mails it to "rnews@B". Somehow, the B system is expected to make sure that all mail to user "rnews" is fed as in put to the program uurec. This program unpackages it and invokes rnews.

The best way to get mail to "rnews" fed into uurec is to use sendmail or delivermail, if you are on a

system running them.

Create an alias in /usr/lib/aliases as follows:

```
rnews: "|LIBDIR/uurec"
```

and sendmail will handle it. If you do not have a facility for forwarding mail to a program, you can gimmick your mailer to watch for it (using popen(3S), this is easy) or, if you don't want to do any programming, you can have cron(8) invoke uurec every hour with /usr/spool/mail/rnews as standard input. This solution is messier because uurec must potentially deal with multiple messages, something that has never been tested.

#### 5. Format of the syssys file

To set up a link to another site, edit the sys file in LIBDIR. This file is similar to the L.sys file of UUCP. Each line contains four fields, separated by colons: The system name of a site to which you forward news. Normally all systems you have links to will be included. You should also have a line for your own system. If this field is ME, it will be used as if it were your local system name. If the system name is followed by a "/", the article will not be forwarded to this system if it has already passed through any of the (comma separated) list of sites immediately following the "/". For example, if the sysline was:

```
yoursite/sitea,siteb,sitec:world,comp,sci,news,rec,misc,na,usa,to.your site::
```

the incoming article would only be forwarded to yoursite if it had not already been to any of sitea, siteb, or sitec. This is normally used to reduce the number of duplicate articles received at a site that has multiple main newsfeeds.

- (2) The newsgroups to be forwarded to that site. This is a pattern of the same kind as a subscription list. Generally, you will list classes of newsgroups, that is, using all for everything. A typical forwarding list for a new site would be

```
world,comp,sci,news,rec,misc,soc,talk,na,usa,to.sysname
```

where sysname is the name of the remote system. (Of course, if you are not in the USA or North America, you would remove those distributions and replace them with the ones appropriate for you.) In particular, you don't want to forward all since local newsgroups (those without dots) should not be sent. For the line describing your own system, this field describes the newsgroups your site will accept from remote sites. Thus, if another site insists on sending you a news group you don't want, for example rec.humor, include !rec.humor here.

- (3) This field contains flags describing the connection.

A indicates that the other site is running an A version of netnews.

B indicates a B version. Leaving it empty defaults to B. If you are reading this document, you have a B version. Some existing sites run A versions. If you aren't sure, ask your contact

at the other site, with whom you should be talking to set this up anyway.

- F indicates that the fourth field is the name of a file. The full path name of a file containing the article in SPOOL will be appended to this file. If the fourth field is blank, the filename BATCHDIR/sysname, (where sysname is specified in the first field) will be used.
- L prevents transmission unless the article was created on this site. If a number follows the L (e.g., L3), sites less than that number of hops away will be considered local. (It is recommended that you feed an L link to a backbone site, to ensure that your submissions will be more likely to get to the entire network, even in the event of a local problem. Please make sure that a mail link exists too, so you can get replies.)
- H can be used to interpolate the history file into the command.
- S says to execute the transmission command directly instead of forking a shell.
- U arranges that the parameter to the optional "%s" in the command field to be filled in with a permanent file name from SPOOL instead of a temporary customized file name.
- M says to use multicasting. Multicasting is described in an appendix.
- I automatically defines the F flag. However, it puts the article's Message-Id into the file instead of the filename containing the article. This is used with the ihave/sendme control messages.
- m (lower case m not upper case M) modifies the newsgroup pattern match to only return true if it matches the pattern in the second field AND the group is moderated. For example, if your sys file looked like:
- ```
avax:world,comp,sci:mF:/usr/spool/batch/avax.sendimmediately
avax:world,comp,sci,rec,talk,misc,soc,to.avax:F
```
- any moderated articles in the newsgroups comp or sci would be placed in the file avax.sendimmediately in the SPOOL directory. All other news for that machine would be placed in the file avax. You could then transmit the articles in avax.sendimmediately more often than the rest of the news.
- u works like m except that it only matches unmoderated news groups.
- N indicates that news should be sent using the ihave/sendme protocol. It should not really be used anymore as I is much more efficient.
- (4) This field is the command to be run to send news to the remote site. The article will be on the standard input. Leaving this field blank means an ordinary UUCP link is being used, that is, the command defaults to

```
uux -r -z sysname!rnews
```

The - option tells uux to expect input from the standard input. The -z option is nonstandard you should add it (see the minus.z files in the uucp source directory). It shuts off the annoying message you would otherwise get mailed to you telling you that your article was broadcast successfully. To avoid using the -z option, change the source or put the uux command in the fourth field. The -r option tells uux not to call the other system once the job is queued. This turns out to ease the load on the system, at the expense of making news be transmitted a bit slower. The news will be sent when the next call is made; usually this means the next time mail is sent to or from your system. If this turns out to be unreasonably long, put a line in crontab to run

```
/usr/lib/uucp/uucico r1 ssysname
```

every hour or so.

Here is a sample sys file for a site myvax with connections to yourvax where myvax also passes news on to downstream. We assume that myvax and downstream exchange a local newsgroup classng.all as well as the network wide newsgroups. News to down stream is batched. We also assume that myvax and yourvax are in the USA, while downstream is in Canada.

```
myvax:world,comp,sci,rec,news,misc,soc,talk,na,usa,lng,to
yourvax:world,comp,sci,rec,news,misc,soc,talk,na,usa,to.yourvax
downstream:world,comp,sci,rec,news,misc,soc,talk,na,lng,to.downstream:F
```

## 6. Posting Methods

The basic method is postnews. This program will prompt you for the title, newsgroups, and distribution, then place you in the editor. (The system default EDITOR is used unless the environment variable EDITOR is set, overriding the system default.) The text should be typed after the blank line. The title and newsgroups are available for editing at the top of the buffer. Other header lines can be added, such as an expiration date or a distribution. When you write out the file and exit from the editor, you will be prompted for what to do next. Your choices are: write the message to a file, send the message, list the message or edit it again.

Another method is to use mail. This can only be done on systems that allow mail to a given name to be fed into an arbitrary program as input. This is easily done with the Berkeley delivermail or sendmail program, and not with any other mailer the author is familiar with. (It may be possible to painfully set this up with MMDF, provided the newsgroup name is no more than 8 characters long.) To use mail, set up an alias such as the following:

```
sci.physics: "[LIBDIR/recnews sci.physics"
```

Whenever a user sends mail to sci.physics, this starts up the given shell command which calls recnews with one argument, the name of the newsgroup. You need to create one alias for each newsgroup, and to keep the list up to date as new newsgroups are created. Recnews(8) will in turn invoke inews.



Note that there are problems with `recnews`. There is no way to use it to post to multiple newsgroups without creating separate articles (something frowned upon because it forces people to read the same thing more than once). Also, there is no way to make the recording feature (to remind people to not accidentally divulge proprietary information) work when `recnews` is used.

## 7. Various considerations

### 7.1. Spooling

If `expire` is already running and `inews` or `rnews` is invoked, the incoming article will be saved in `SPOOLDIR/.rnews` for later processing by `rnews -U`. This is to prevent the incoming article from getting "lost" and not being in the history file. `Expire` automatically runs `rnews -U` when it has finished. If you have defined `SPOOLNEWS`, you should also invoke `rnews -U` from `crontab` on a regular basis.

If you are not using some type of "system" locking (e.g. `flock()` or `lockf()`) you must make sure that the various lock files are removed when the system is rebooted after a crash. You can put a line in `/etc/rc` to remove these lock files. The most important one is `LIBDIR/active.lock`.

### 7.2. Setuid bits

The current intended state of affairs is that `inews` runs `setuid` to `NEWSUSR`. The `readnews` and `vnews` programs do not need to be `setuid`. This makes it possible to write your own interface to read news instead of using `readnews`. (As distributed, `inews` is also `setgid`. I know of no good reason for this.)

### 7.3. Modes of Spool Directories

All the files should be writable by `NEWSUSR`. However, due to a glitch, you will probably have to make the `SPOOLDIR` and its subdirectories mode `777`. It could be `755` except for one problem. When a new newsgroup comes in, `inews` will attempt to `mkdir(1)` a new subdirectory of `SPOOLDIR` for the newsgroup. Since both `inews` and `mkdir` are `setuid`, `mkdir` will use the `uid` of the person who ran `inews` instead of `NEWSUSR` when checking for permissions. If the directory mode isn't `777` the check will fail. 4.[23] BSD sites do not have this problem as `mkdir()` is a system call and will create the directory with the "correct" ownership. Here are several alternatives if you don't want a `777` directory around:

#### 7.3.1. Fix Real Uid

If `inews` is always run by `cron` or as `root`, the real `uid` can be arranged to be `root` or `NEWSUSR`. This is a poor solution since it makes the local creation of new newsgroups require super user permissions, and is a potential security hole. If this approach is taken, care must be taken to ensure that the owner of the created directory is `NEWSUSR`.

### 7.3.2. Change the Kernel

Inews will do: `setuid(geteuid())(seesetuid(2)and geteuid(2))` before it forks the `mkdir`. If your system permits this call, there will be no problem. In particular, Berkeley 4.0 UNIX and later systems allow this. An alternative change to the kernel is to automatically stack uids: when a `setuid` program is run, set the new real uid to the old effective uid.

### 7.3.3. Groups

You could have inews be `setgid` to `NEWSGRP` and all files writable by the group. This approach has been tested and the problem turns out to be that the `mkdir` command uses the `access(2)` system call to check permissions. Since `access` uses the real gid, you run into the same problem.

### 7.3.4. Another Mkdir

You could create a version of `mkdir` that does less checking and put it in a directory that can only be accessed by `NEWSUSR` (mode 700, owned by `NEWSUSR`). Have inews fork this `mkdir`.

## 7.4. Expiration dates

To get articles to expire automatically, put a line in `crontab` to run

```
LIBDIR/expire
```

every night. This command deletes all expired news. The `-a` `newsgroups` option causes all expired news to be archived under `/usr/spool/oldnews` depending on which newsgroups are selected. (See `expire(8)` for details.)

Sometimes news is not expired when it should be. Be sure to check that `expire` has permissions to unlink files, and that it is properly `setuid` to `NEWSUSR`. You can manually invoke `expire` with the `-v` (verbose) option to find out what it's doing. Adding levels of verbosity (e.g., `-v6`) will get more and more output.

## 7.5. Version to Version

Version B will understand incoming news in either version A or B format, automatically (presuming `OLD` is defined in `defs.h`). Version B will generate either format, depending on the flag in the third field of the `sys` line. Version A will not understand version B format. Thus, it is possible for two version B sites to communicate using version A format. This will work but is not a good idea, since the translation from B to A loses information (such as the expiration date) which will not be there when translated back to version B.

News from versions A and 2.9 B do not conform to the USENET interchange standard. 2.10 B (and later) supports the standard and will communicate with either A or 2.9 B news. A news is

written (losing other header information) if A is in the flags for the system. If OLD is defined, 2.10 will write out headers with both standard ("Date" "Message-ID") and 2.9 ("Posted""Article-I.D.") lines so that either B system will properly handle the article. Incoming news is recognized by the first letter (A for A news), or the lack of an "@" in the "From" line (2.9).Missing fields are constructed as well as possible from the available information.

## 7.6. Using the news reading program rn

You must have installed all posted patches to rn up to and including #30. Otherwise, rn will not work properly with news 2.11.

## 7.7. Distributions

News 2.11 is much more particular about handling distributions than previous versions. In particular, make sure that you have the distribution world in every line of your sys file unless you are specifically limiting the distribution and are sure you know what you are doing.

## 8. Control Messages

Some news systems will send you articles that are not for human consumption. They are messages to your news system called control messages. Such messages contain the "Control" header. Older systems use newsgroups matching all.all.ctl, and this will still work, although the "Control" header is preferred. Since the newsgroup name is used for distribution only, and is not checked to ensure it's in the active file, such newsgroup names can still be used. This makes it possible to post network wide control messages with world.msg.ctl (or restricted broadcast such asbt1.msg.ctl) or messages for a particular system: to.ucbvax.ctl. Messages are canceled, however, with a "Control" line in a message to the same newsgroup(s) as the original message.

A control message contains a command and zero or more arguments (much like a UNIX program). The subject of the article contains the command and arguments. The body of the article is usually ignored, although some messages can use it for additional text information.

### 8.1. ihave/sendme

Two control messages are ihave and sendme. These messages allow two participating sites to set up a link so that one site will tell the other site it has a given article and wait for a request before it actually sends it. The normal case is to send an entire article to a system, which consults the history file to see if the article has already been seen, and then throws it away if it has been seen before.

Use of these control messages can cut down on this wasted transmission, but if you have a polled UUCP connection, they can slow down receipt of news due to polling delays. It is up to each connected pair of sites whether they want to use this protocol. The choice is controlled by the I flag in the sys file. In the case of a leaf node (one with only one neighbor) there is no advantage to this protocol. Even if both sites are able to initiate a connection (have dialers or the link is hard wired)

the -r option on the uux can cause 2 hour or more delays in propagating news. If transmission time and phone bills dominate your costs, and you are sending news to several sites, and large article bodies dominate the costs (rather than the headers and the time spent by UUCP negotiating transmission) it is probably worthwhile to use ihave/sendme. If your costs are dominated by CPU load from UUCP, or if you send news to a site that cannot get it from anywhere else, you probably do not want to use this protocol. The decision can be made independently for each site in your sys file.

To use the ihave/sendme protocol, you should have the I flag defined in the sys file for the remote site. For example:

```
elsie:world,na,usa,comp,sci,news,rec,soc,misc,talk,to.elsie:IF
```

Note that the fourth field must be blank.

Normally, a message arrives at this site, is found to be ok (i.e. it hasn't already been received), and is considered for retransmission to other sites. If the system you are sending to has the flag defined, the Message-ID is saved in the file specified in the sys entry.

After a period of time,

```
sendbatch -i elsie
```

is run by cron(8). It takes this file, makes it into an ihave control message, and sends it to the other site. When the message is received on the other site, the Message-IDs are looked up, and if any of the messages have not been received, they are put into a sendme control message. When the ihave message has been fully processed, the sendme message is sent to the site that sent the ihave message.

When the original site receives the sendme message, it sends the requested articles in the format designated by the sys file (the I flag is ignored in this case). In this example, it would write into the file /usr/spool/batch/elsie for later processing by sendbatch.

## 8.2. newgroup

This message has one or two arguments, the name of a newsgroup to be created and the second optional argument moderated. This allows special action to be taken locally when a new newsgroup is created. It is generated by the C option to inews. By default, the newsgroup is added to the active file, and mail is sent to the local contact advising that this has happened. The directory will be created when a message for that newsgroup arrives. If NONEWGROUPS is defined, the newsgroup will not be created and NOTIFY will be sent mail explaining how to do it manually. See the routine "c\_\_newgroup" in control.c if you want something different to happen. (Note that, although the body of the message contains a brief description of the purpose of the group, this body is usually thrown away by existing software.) If the second argument is the keyword moderated, the newsgroup is created moderated, else it is unmoderated.

### 8.3. rmgroup

This message has one argument, the name of a newsgroup to be removed. It is used for network-wide cancellation of a newsgroup. If MANUALLY is not defined, it will remove the articles, directory, and active file line for the group. There is a shell script rmgroup that does essentially the same thing as this message, but the shell script only removes the group locally. We recommend that you leave MANUALLY defined, and when you receive mail advising you of the demise of the newsgroup, you run rmgroup by hand. This will prevent accidental or malicious removal of a good newsgroup.

### 8.4. cancel

This message cancels a given article. It takes one argument, the message id of the article to cancel. It should be broadcast to the same newsgroup as the original article. If the article to be canceled is not present, the control message will not be propagated to downstream sites.

### 8.5. sendsys

The sys file is mailed to the originator of the message. There are no arguments. This is used for making maps. Since your sys file is public information, you should not remove or change this control message.

### 8.6. senduname

The uname program is run and the output is mailed to the originator of the message. There are no arguments. This was used for making UUCP maps. It is of questionable use any more. If you do not run UUCP or have sites in your L.sys which are a secret, you may wish to edit this. Note that only the output of uname is mailed, not the contents of L.sys (which news does not have access to anyway). If you do make a change, you should arrange that some mail still is sent out to the originator of the message, so the originator will know your site received it. See the code in routine "c\_\_senduname" in control.c.

### 8.7. version

The local version name/number of the netnews software is mailed back to the originator of the control message.

### 8.8. checkgroups

This control message is an attempt at semi-automatic maintenance of the list of active news groups. This control messages takes the body of the article and pipes it into LIBDIR/checkgroups. As mentioned previously, LIBDIR/checkgroups will update the newsgroups file, add any missing newsgroups, and mail a message to NOTIFY about any old newsgroups that should be removed. It is expected that the person who maintains the list of active newsgroups will

broadcast this control message on a regular basis.

## 8.9. Other Messages

Any unrecognized message will cause an error message to be mailed to the local news administrator. Additional messages may be defined as time goes on, such as messages to automatically update directories or maps. You should be willing to go into the code (control.c) and add messages as they become standardized.

## 9. Maintenance

There are some things you should do periodically to keep your news system running smoothly. We hope to eventually automate all or most of this, but right now some of it must be done by hand.

The history and log files in your LIBDIR directory will grow. You should make sure that they are cleaned up periodically. The LIBDIR/expire program will remove lines from history corresponding to deleted articles, but it is a good idea to check the file every few weeks to make sure it is not going wild. Be sure not to completely lose your history file when you clean it up, in case another neighbor tries to send you an article you recently got. (If you only get news from one site it is safe to clean it out completely.)

The log file is not automatically cleaned out by any netnews software, and will grow quickly. The misc/trimlib script can be installed in LIBDIR/LIBDIR/trimlib, and invoked regularly by cron(8).

You should also clean out old newsgroups that are no longer active. To remove a newsgroup misc.foo, you should run the shell script rmgroup with misc.foo as the argument. That is,

```
LIBDIR/rmgroup misc.foo
```

Note that clearing up UUCP constipation is another thing you'll have to do if you have flaky hardware or phone lines. If you have more than one connection, chances are that UUCP will get clogged up when one of your neighbors goes down for more than a few hours. Various spooling schemes are being worked on to help make the news/uucp system more robust, but one thing you can and should do, if you find your /usr/spool/uucp directory getting too big, is to install a subdirectory fix to UUCP. A quick and dirty version of this is available from Duke, which traps the file oriented system calls at the assembly language level and maps, for example, D.fooA1234 into D.foo/D.fooA1234. Since the C. and D.local directories still get big, in practice this can still create some big directories, but the directories tend to be a factor of 5 smaller, resulting in a factor of 25 improvement to speed (since a directory traversal for all files is quadratic on UNIX). Right now, unless you have the uucp distributed with 4.3 BSD or the so called HoneyDanBer uucp, UUCP is the weak link in netnews distribution, and you should certainly keep an eye on it.

## 10. Creating New Newsgroups

As system news administrator, you are able to create newsgroups. To create a newsgroup, first make sure this is the right thing to do. Normally a suggestion is first posted to news.groups, whatever.relatedgroup for a world wide newsgroup(whatever.relatedgroup should be the group which you are proposing to sub-divide. For instance, to propose creating rec.arts.tv.soaps, post the original article to rec.arts.tv, news.groups.) Followups are made to news.groups only.(You can force this by putting the line:

Followup-To: news.groups

in the headers of your original posting.) If it is established that there is general interest in such a group, and a name is agreed on, then someone creates it by typing the command

```
inews -d local -C newsgroup
```

This will create the active entry locally. The directory will be created automatically when the first article for that newsgroup is received. It will also prompt you for a paragraph describing the group and start up an inews to post a newsgroup control message announcing the group.

You must be the super user to use the -C option to inews.(That is, your uid must match ROOTID. It is recommended that you change ROOTID to your own uid so you don't have to su to create newsgroups.) If you change the distribution to be something other than local, the newsgroups message will be sent to the specified distribution.

## 11. Conversion from A to B

If you are currently running version A on your system, note that B is incompatible with A. The files are stored in a different format (headers have mail like field names now).The directory organization is different (each newsgroup has a sub directory of its own, and the file names are numbers rather than site.id pairs). There are no bitmap, uindex, or nindex files to be trashed (which articles have been read is stored in each users .newsrfile).The user interface is slightly different (news/netnews(1) is now called readnews, news is posted using inews, subscription is done by editing .newsr, the sense of the -c option is reversed, news is presented in newsgroup order, the -a and -t options now probably need -x as well, and there are many minor changes).

We decided not to provide a program to convert from version A to version B. Rather, the following strategy was adopted for conversion:

- (1) Install the new news in a different spool directory from the old one. For example, you can use /usr/spool/newnews. You can change to the standard name later if you want.Get it to work for local messages.
- (2) Post an article to newsgroup general with the old news announcing the change. Make available documentation such as the accompanying paper How to Read the Network News to the users. This article will be the last one in the old news.
- (3) Chmod the old news directory to 555 to prevent any more news from being posted. (Actually, this will prevent the bitfile from being updated, so it may not be a good idea.)

- (4) Replace the old rnews program with the new rnews program.
- (5) Test it by having your neighbor send you a message.
- (6) Wait a reasonable period for everyone to have read the final article with the old news. Perhaps a few weeks is right.
- (7) Uninstall the old news.

Users will have to invoke readnews instead of netnews to read news. Depending on your old method of posting, this could be changed too. (If you were using mail, it does not need to be changed.) They will also have to fix their subscriptions. In general, they can type

```
netnews -s
```

to see what they subscribe to on the old system, and then create a file in their home directory called .newsrc containing

```
options -n their subscription
```

The format of the subscription pattern matching is the same as in A except that ALL is replaced by all (change to lower case). Something along the lines of this could be used to automate this:

```
(echo -n "options -s" ; netnews -s | sed s/ALL/all/) > .newsrc
```

## 12. Conversion from 2.9 to 2.11

Conversion from 2.9 to 2.11 is not nearly as involved as an A to B conversion. The user interface does not change much, and the user .newsrc files are not affected. However, it is recommended that you do the conversion during a time when no news is received, so that incoming news will not get lost. One way to ensure this is to make /usr/bin/rnews be a shell script which saves the article in /usr/spool/innews/#### (#### is the process id of the particular shell and will be unique for each article).

The first step to conversion is to customize the sources. In the past, you had to take a fresh distribution and edit the defs.h file and Makefile to suit local preferences. If you had many local changes, or didn't record the local changes, upgrading could be annoying. 2.11 provides a mechanism to automate these changes. Create a shell script in the src directory called localize.sh. (You can use localize.sample as a template.) This shell script should copy defs.dist to defs.h, and copy Makefile.dst to Makefile. It should chmod any files that need to be changed (often Makefile and defs.h) to a writable mode. Then it should invoke ed(1) on the files, making any necessary local changes.

The next step is to compile the software, with make(1). It may be necessary to update the localize.sh file until you are satisfied with the compilation. Note that after any change to the Makefile in localize.sh, you should run localize.sh by hand. Otherwise, although make will run it



for you, it will then continue to do the make with the old Makefile.

When the software is compiled, you should run the `cvt.active.sh` shell script, with the `lib` and `spool` directories as parameters. This will create a new active file in `LIBDIR/active`. Then run `cvt.links.sh` with the `lib` and `spool` directories as parameters. Then run `cvt.names.sh` with the `lib` and `spool` directories as parameters. Old news will be linked into the new hierarchy while leaving links in the old hierarchy. If you were using the default library and spool directories, you would do the following:

```
sh cvt.active.sh LIBDIR /usr/spool/news
sh cvt.links.sh LIBDIR /usr/spool/news
sh cvt.names.sh LIBDIR /usr/spool/news
```

The next step is to back up the old binaries:

```
mv BINDIR/rnews BINDIR/ornews
...
```

and to install 2.11 with

```
make install
```

Once it is installed, any incoming news will be placed into the new hierarchy but not the old one. The critical time window is between running the three shell files and installing the new software. Any incoming news between these two points will appear in only the old hierarchy and be lost to the new software. If any significant time elapses here, you should divert `rnews` into a separate spool directory as described above.

It is crucial that you run `expire` before any new news arrives. `Expire` will update several key files automatically.

Finally, test things by posting articles to `to.neighbor` newsgroups and watching some incoming news, and announce the change to your users.

When you are satisfied that the conversion was successful, run the shell file `cvt.clean.sh` which will remove the old 2.9 news hierarchy.

### 13. Converting from 2.10.x to 2.11

Converting from version 2.10 or later to 2.11 requires no special intervention. All files are updated automatically when you do "make update".

## Appendix A: Setting up a Compressed, Batched Newsfeed

First, BATCH must have been #define'd when you built the news system. To check, look in the file defs.h in the news source directory. BATCH should be defined as a program name (by default, unbatch). If it's undefined or commented out, define it, remake the news system, and install the new software.

You'll also need a working compress program. Use the one shipped with this news distribution, which is based on version 4.0. Your news neighbors should be running a compatible version of compress. Versions 3.0 and 4.0 are compatible with each other, but both are incompatible with versions 2.0 and earlier.

Update your sys file. First, add the F flag to the other news system's line. For instance, if your compressed and batched news feed is named frobozz, and its sys file entry looks like:

```
frobozz:world,comp,sci,rec,misc,soc,talk,news,na,usa,  
ca,to.frobozz
```

then add the F flag as the third (colon-separated) field:

```
frobozz:world,comp,sci,rec,misc,soc,talk,news,na,usa,  
ca,to.frobozz:F
```

Now the pathnames of articles to be sent will be stashed in a file. This file can be specified in the fourth field of the sys entry. Normally, you can just use the default of BATCHDIR/system, (where BATCHDIR is usually /usr/spool/batch and system is the name of the remote system, in this example frobozz). A name of that form is necessary: the sendbatch script, which sends the batched news, looks for a file name of this form to decide if there's news for the remote system.

Your completed sys file line should look something like:

```
frobozz:world,comp,sci,rec,misc,soc,talk,news,na,usa,to.frobozz:F
```

In your crontab file, find or create at least two news lines: one that runs nightly, and one that runs every hour or so. The nightlyrun script should run expire, trim log files, and perhaps compile weekly statistics that you post to a local area newsgroup one day a week. The hourlyrun script should complete the transmitting task with a line like:

```
sendbatch c frobozz
```

Make sure the script knows how to get to the directory in which sendbatch lives. You can either mention the directory in the script's PATH-setting line, or replace sendbatch with its full pathname. Sendbatch reads the files mentioned in /usr/spool/batch/frobozz, batches them, optionally compresses them, sends them to the remote system, and arranges for remote processing.

This remote processing can be directed by another file in BATCHDIR. Make an executable file with a name of the form BATCHDIR/system.cmd (for this example, /usr/spool/batch/frobozz.cmd). Put a line in it specifying the command that the remote system should execute to unpack the news batches that your system will send. An example frobozz.cmd would be:

```
uux -r -z -n -gd frobozz!rnews
```

Now your system will transmit compressed batches. The receiving side of the business is handled largely by a program called rnews, which will call other programs in LIBDIR to do additional processing on the incoming batches.

Make sure there is an executable file called rnews in the BINDIR directory (check the Makefile for its actual location). It must be reachable by UUCP or by whatever transport you'll use to transfer the netnews. If you defined BINDIR as /usr/bin, you should have no problems because uuxqt can already get there. If you defined it as a different directory, you may have to teach uuxqt to look in that directory; accomplishing this varies from system to system. On 4.[23] BSD, add the directory to the PATH= line of your UUCP L.cmds file. On System V, on the rnews line of your L.cmds file, add a comma followed by the remote system's name on that line. If yours is in /usr/bin/news/rnews, your L.cmds file will look like:

```
[For 4.[23]BSD]
PATH=/bin:/usr/bin:/usr/bin/news
rnews
```

```
[For System V]
/usr/bin/news/rnews,frobozz
```

Other systems have a similar file in the /usr/lib/uucp directory by which you can specify added programs and paths different from the defaults. HP-UX, for example, has a /usr/lib/uucp/COMMANDS file which expands uuxqt's horizons. HoneyDanBer uucp (a.k.a AT&T Basic Networking Utilities) has a Permissions file that controls what is executed. In more restrictive cases, paths are compiled into uuxqt. If you can't modify any UUCP files, just put rnews in /usr/bin.

Tell the person at the other end of your newsfeed to use sendbatch -c to send you news. Once that's in place, watch your UUCP LOGFILE and your news log and errlog files to ensure that news is being correctly received and unpacked on your system.

Older compressed batching systems will try to exec cunbatch instead of rnews. If you are still communicating with these, leave cunbatch in BINDIR until they have upgraded their software.

## Appendix B: MULTICAST

If this is defined (in defs.h) then two new flag characters become defined in the sys file. The first, and most important, of these is the M flag.

If the M flag is set on some line in the sys file, then the fourth field (transfer command) is redefined to become a multicast name. That is simply another system name, expected to be found in the first field of some line in the sys file (textually following the line containing the M flag).

When a news item is being retransmitted, if it should (according to the subscription list) be sent to a system that has the M flag set, then instead of a command being run immediately to transmit the news, the news system remembers the system name, along with the multicast name (fourth field).

Eventually the multicast system name is found in the first field of a sys file line. If its subscription list allows transmission of this news item, then its command will be executed. This command may have up to two "%s" substitutions in it. The second of those is replaced by the name of a file containing the news item (used with the U flag). The first is subjected to rather special treatment. The whole "word" (delimited by white space) containing that "%s" is duplicated as many times as there were systems with the M flag set that referenced this multicast name (which might be 0 times, causing that "word" to be omitted). In each of these duplicates, the "%s" is replaced by the name of a system. Note the multicast system name itself is not included in this process. Then the command is executed as usual.

The second flag available if the news system is built with MULTICAST defined is O. If this flag is set, then the sys file line will be ignored unless the system name is a multicast name from some earlier line with the M flag, and the news item is to be sent to that (earlier) system. This allows the subscription list for the multicast system name (which is likely to be a fake system name, invented just for this purpose) to be given a very wide subscription list (like all) without any unusual effects.

Here is an example. Assume that you wish to forward comp.unix to four people by mail. You could do this as ...

```
fred:comp.unix::mail fred
harry:comp.unix::mail harry
jane:comp.unix::mail jane
tony:comp.unix::mail tony
```

however this causes the mail program to be started 4 times, once for each recipient. On some systems starting the mail program is a very expensive operation. If MULTICAST is defined, an alternative method is

```
fred:comp.unix:M:tony
harry:comp.unix:M:tony
jane:comp.unix:M:tony
tony:comp.unix::mail tony %s
```

This would cause just one command to be run: "mail tony fred harry jane". Note that "tony" must still be explicitly included in the argument list to the mail command; the "%s" does not expand to include the multicast "system name" itself.

A more useful way of doing this, which does not assume that all the mail readers will want to read the same newsgroups is as follows.

```
fred:comp.unix:M:Mail
harry:sci.physics,sci.astro:M:Mail
jane:comp.unix.wizards,soc.women:M:Mail
tony:comp.unix,comp.unix.wizards,rec.humor:M:Mail
Mail:all:O:mail %s
```

Now, if a news item in group comp.unix was received, the command

```
mail fred tony
```

would be executed. If the news were in both comp.unix and comp.unix.wizards then the command would be

```
mail fred jane tony
```

If a news item in sci.med (which noone gets by mail) arrives, then the "Mail" line will be ignored, because of the O flag. "Mail" is a fake system invented just so its "transfer command" can be used to send news to the other recipients.

The same kind of technique can be used for normal transfer of news to other systems if your transport network supports a facility to send to many other systems in one command. (That is, if it has a multicast facility.) SunIII (the network used in Australia) has this ability, so a typical Australian sys file looks like

```
emuvax:world,aus,comp.sci:M:FakeName
kremlin:world,aus,comp.sci:M:FakeName
kanga:world,aus,comp.unix:M:FakeName
FakeName:all:OUS:/bin/sendfile NRSareporter -d%s -x%s
```

A news item in aus.general causes the following command

```
/bin/sendfile -NRSareporter -demuvax -dkremlin -dkanga -x/usr/spool/...
```

to be executed. Just one command is run to send the news to three remote systems.

If a multicast system has the F flag set, then the name of a file containing the news is appended to the file whose name is in the fourth field, as usual. But on the same line, separated by spaces, will be appended the names of all the systems that referenced this multicast system.

For example, if the Australian site wanted to batch news, instead of sending it directly, it would simply change the last line of its sys file to

```
FakeName:all:F:/usr/spool/batched/allsites
```

Then a news item in misc.jobs would cause the following line to be appended to /usr/spool/batched/allsites

```
/usr/spool/news/misc/jobs/5542 emuvax kremlin
```

This can then be processed later, in something like the normal manner. (Unfortunately no command to do this processing are yet available.)

Caution: when MULTICAST is defined, the first "%s" in all transfer commands is used for multicast, regardless of whether or not the system name is ever used as the last field of some line with the M flag set. To use the U flag in such a case, a dummy "%s" should be used, it will simply be omitted from the command that is executed.

As an example, if a sys file line without MULTICAST defined was:

```
foovax:comp,world,na,usa:U:uux foovax!foonews <%s
```

with MULTICAST defined, it would need to be changed to:

```
foovax:comp,world,na,usa:U:uux foovax!foonews %s <%s
```

Additional caution: The numbers of system names that may be used in this way are quite severely restricted. Typically there may only be about 10 multicast system names, and each of those is restricted to sending to no more than about 20 systems. These limits are dynamic (that is, the numbers counted are the number of multicast systems receiving any single news item, and the number of systems that each of those will actually cause this particular news item to be sent to). These limits should easily suffice for real news sending to remote systems; however they are not likely to suffice if you want to mail news to everyone on your host.

## Appendix C: Installing news on a Eunice system

Eunice does not have links. To get around this problem without wasting megabytes of disk space, links are simulated. Each article is stored under the directory corresponding to the first group on the Newsgroups line. The remaining directories contain a short file giving the name of the real file. Readnews and vnews know how to follow these links; so does rn if it is compiled with LINKART defined (rn is not included in this distribution).

If you expire different groups at different times, an article disappears when the first group in the Newsgroups line expires, even if it's cross posted to a group you retain longer. Also, the archival option for expire (-a flag) does not work.

Eunice does not have the set-user-id bit. Instead, the VMS feature allowing installation of privileged images is used. inews and rnews must be installed with SYSPRV privilege. To allow VMS to install inews (rnews is a copy of it), it must be linked with the /notrace flag. Despite the documentation, this only seems to work if the VMS linker is used on VMS - format object files, as of Eunice version 4.2. So it is VERY IMPORTANT to type vmsobj before building news.

The DCL command file euninstal.com installs inews and rnews with the proper privileges. It is run by "make install" as the last step. It is possible that it will fail because your system does not have sufficient global page table space or known file list entries. See the Eunice installation manual to learn how to increase the values of these parameters (you will have to reboot the system).

The euninstal.com script must be run every time the system is booted. I suggest invoking it from /etc/suchmod.com.

It is crucial that LIBDIR/active and LIBDIR/seq be in Unix format, as opposed to VMS format. Use etc/vmstounix to assure this.

Not all options have been tested; if you select options other than those set by the localize.vms script (except for obvious things like etc) I can't guarantee that it will work. In particular, I haven't attempted to define SPOOLNEWS or MULTICAST. That doesn't mean they won't work, just that I haven't tried them.

If you choose to run compressed batching, I recommend that you put cunbatch in your L.cmds file and have your news feed send compressed batches the "old" way, because this saves an exec, and forks and execs are so expensive on Eunice (the new way is for the first line to be #cunbatch, and have rnews exec cunbatch).

If you have questions on Eunice and news contact:

Joe Buck, Entropic Processing, Inc.

seismo!epiwr!lepimass!jbuck  
hplabs!oliveblepimass!jbuck

付録 B

ニュースインストラクションガイド(日本語対応版)



## Note to implement Bnews 2.11 (J)

Akira Kato-

Tokyo Institute of Technology

### ABSTRACT

This memo describe the overview of installation of Bnews 2.11 (J). (J) shows the patch for JUNET (This tag is not exist officially), which allows us to use Kanji in the articles.

#### 1 . Treatment of Kanji

We, JUNET, use not only English articles but also Romaji and Kanji articles. There are several ways to express Kanji, i.e., JIS-6226/6228 code, Shift-JIS code, EUC code, DEC Kanji Code or many other minor codes. When we use Kanji locally, we can select one of the ways indepently to other people. However, when we use Kanji in the network wide apprication such as electronic mail/news system, we should use standard expression to communicate each other. In JUNET, we have decide to use KANJI code based on JIS-6226/6228 with a local restriction:

Newline character is  
not allowed in the Kanji mode.

When we lseek()s a file to arbitrate position, we can hardly determine the character mode at the point. To find out the character mode, we should traverse the file backward until we find an escape sequence. If we do not apply this res-triction, we should back to the top of the file in worst case. In order to limit the mount of backward search, there striction is neccessary and the maximum amount of backward search is bounded to the length of the line.

To handle Kanji articles, two major implementation can be considered. One is that the Kanji article is spooled in JIS code regardness of the local code handled by the local Unix. In this implementation, code conversion is neccessary when we post an article and read articles. Another imple-mentation is that the Kanji article is spooled in the localcode. In this implementation, code conversion is done when a article arrives at the system and is distributed from the system.

I use the former implementation in this distribution  
because:

- + The overhead of gateway can be reduced.
- + It is hard to recover the original escape sequence, i.e., ESC- $\$$ -@ may be mapped to ESC- $\$$ -B during the twice code conversion.
- + When the spool area of a file server is shared among the client machine by NFS/RFS, it

is better to keep the articles in the common code. If the server is Shift-JIS and clients are EUC, JIS, Shift-JIS, .....,you should make a lot of filters.

+ Same problem occurs when news server distributes news articles via nntp protocol.

This distribution do not contains these code conversion routine in postnews / readnews / vnews, and assumes that there are Kanji filters (JIS <-> local code) in each system if it supports Kanji.

## 2. Installation of Bnews 2.11 (J)

Original procedure of installation begins to edit localize.sh,which dist.hand Makefile using I think it is more difficult to make a correct sed program than edit dist.h and Makefile directly. I prepared two set of those files for 4.2BSD and SystemV. If your system is 4.[23]BSD, execute as follows.

```
% cp defs.h.BSD defs.h
% cp Makefile.BSD Makefile
```

If your system is SystemV,

```
% cp defs.h.USG defs.h
% cp Makefile.USG Makefile
```

Next, you SHOULD edit these files to match your environment. The important macros defined in defs.h are following:

**BATCH** Don't undefine this even you don't receive batched newsnow.

**PAGER** This PAGER is used in readnews / vnews. It is assumed that it can handle your local Kanji code. If your

system is not Kanji Unix, use less posted by keisuke@cs.titech.junet. If you have to convert Kanji code and the filter is named ntohk, define PAGER as

**INTERNET**

Define this if you can write mails with JUNET standard addressing, user@domain notation.

**MYDOMAIN**

Change this to your domain name. This must start with '.' like '.cs.titech.JUNET'

**UNAME** If your system is SystemV and you can change your uname easily, define this.

**UUNAME** If your system is SystemV and you can hardly change your uname, use this instead of UNAME. Don't forget to write your system name to /etc/uucpname.

MYORG It is recommended to use path name like /usr/lib/news/organization rather than write your organization name directly in defs.h. Don't forget to write your organization name to/usr/lib/news/organization. Only the first line is used.

JUNETDefine this. If you undefine this, escape character(¥ 033) is vanished.

NTOHK-Specify the filter to convert JUNET standard Kanji code to your local code.

HTONKSpecify the filter to convert your local code to JUNET standard Kanji code.

Important macros defined in Makefile are following.

#### NEWSUSR

Define the user id of your news system.

#### NEWSGRP

Define the group id of your news system. In typical system, user id of news is 6 and group id is 6 like:news:\*:6:6:News Administrator:/usr/spool/news:/bin/sh

#### SPOOLDIR

Spool directory where articles are installed.

#### BATCHDIR

When you use batch scheme, path of installed articles are recorded instead of passing articles to some command like uux. This defines default directory in which the record file is created.

#### INSCOMPRESS

If your system is 4.3BSD, define this like one shown in the comment.

After editing those two files, you can make the news system:

```
% make
% su
# make install
# make update
```

Now, please check the files such as

/usr/lib/news/organization and /etc/uucpname (if necessary).

### 3. News Distribution

In 2.11, batch mechanism has been changed and no longer use /usr/bin/cunbatch. If a site A distributes news to siteB with batch and compress, uux B!rnews is invoked instead of uux B!cunbatch. The first input line of uux is When the uucp carries the file, rnews is invoked at

site B. rnews read invoke /usr/lib/news/batchdir/cunbatch which decompresses the input data and pipes to rnews again.

The compress is now version 4.0 while most sites are using 2.0. Then you should edit the /usr/lib/crontab to feed news in the old fashioned way to the neighbors which do not install the 2.11 news system yet. If a line of crontab is

```
5 0,4,8,12,16,20 * * * /usr/lib/news/sendbatch siteA -c siteB siteC
```

and the rnews at siteC is still 2.10.2, it should be

```
5 0,4,8,12,16,20 * * * /usr/lib/news/sendbatch siteA -c siteB -o -C siteC
```

where -o specifies to use 'uux siteC!cunbatch ', and -C forces the compress to produce the compatible output. After you confirm that all of your neighbors has been installed 2.11, you can remove -o and -C. And you had better keep /usr/bin/cunbatch for a while.

The format of /usr/lib/news/sys is not changed, however, you should supply full path if you specify to execute a command which is not located in the standard path. Therefore, you should write

```
oopsvax:fj,world,...,misc::/usr/lib/news/sendnews -a rnews@oopsvax  
rather than oopsvax:fj,world,...,misc::sendnews -a rnews@oopsvax
```

Each entry of sys file can be splitted into several lines, by terminating a physical line with '¥' character. But there would be a bug as reported in the net news. Please take a care if you use this function.

## 4. Kanji Issue

### 4.1. inews/rnews

In this distribution, Kanji articles are expressed in JIS-code even in the spool directory. Therefore, no conversion is done when receiving and distributing the news articles, and inews/rnews have no conversion capabilities. The difference from the original is that this version passes '¥ 033' characters.

### 4.2. readnews

When we read articles by readnews, it is necessary to convert the Kanji code from JIS code to arbitrary code. This can be simply done to define the PAGER macro in defs.h to include the real pager and filter connected via the pipe.

### 4.3. postnews

When we post an article by postnews, code conversion is necessary, because we usually use editor to create the article and this is in local code for many cases. postnews uses HTONK filter defined in defs.h just before invoking the inews. When we post a followup, we sometimes

include the original article. This inclusion mechanism requires to convert the JIS code for the spooled article to local code. This conversion is done by a filter NTOHK defined in defs.h.

#### 4.4. rn/Pnews

Many netland people use rn instead of readnews. rn should convert the Kanji code and this can be done by modifying the art.c file. Now only one version (output code is old-fashioned JIS for the many silly terminals) is available, however, incomplete.

Pnews is postnews. This program must convert Kanji code as the postnews do. You can patch it by change the line include

#### 4.5. vnews/vn

I don't use both programs, I don't know how to patch them. If you hack up with them, post the result, please.

付録 C

JUNETのアドレス表現について

# JUNETのアドレス表現について

Tue Nov 4 12:43:26 JST 1986

村井純、田中啓介  
加藤朗、篠田陽一

Junet運営委員会

## 概要

JUNETは階層的なドメインの概念に基づいた名前指定法を採用しており、すでに4.9, 6.1と二つのバージョンのsendmail用のルール生成系を用いてきた。

JUNETの国際的な位置づけを考え、特にCSNETなどの海外ネットワークとのメッセージ交換の便を考慮し、アドレス解析ルールを改良し、バージョン6.2のルール生成系をリリースすることになった。ここでは、この新しい解析ルールにおけるアドレスの指定方法、及びその解析の概要を解説する。なお、本文は純粋に技術的な側面について述べたものであり、JUNET以外のネットワークへのメッセージについては、課金などの問題があり、だれでもそのリンクを用いてメッセージを送信してよいということではないことに注意されたい。

### 1. 基本形

JUNETの名前指定において、'@'を基本演算子として用いる。  
その基本形は以下の通りである。

`<local_name>@<domain_name>`

ここで、`<domain_name>`の部分は、階層的なドメイン構造による名前であり、JUNET内での宛先となるシステムを一意に決定する。`<local_name>`の部分は、目的のドメインに到達するまではなんらの解釈も行われず、`<domain_name>`で指定されるドメインの名前解決担当システムが、その解釈を行う。

この基本形におけるもっとも簡単な名前の例は以下のものである。

`USER@DOM-`

### 2. 基本形の拡張

JUNETとゲートウェイによって接続された他のネットワーク、

---

-以下、USERはユーザ名、DOMはドメイン名、SYSはシステム名、NETはネットワーク名をそれぞれ表現する一般的なシンボルとする。

あるいはsendmailを持たないシステム上のユーザを宛先とする場合、先に示した

<local\_name> の部分が拡張される。即ち、バージョン6.2のルールにおいて <local\_name> の部分の表記に使用可能な文字は以下に示すものである。

- ・ ユーザ名、システム名、ドメイン名、ネットワーク名等の名前識別用の文字列(英数字及び '-', '+', '\_' からなる)
- ・ UUCPベースのアドレス指定用の特殊文字'!
- ・ ドメインベースのアドレス指定用の特殊文字'%'および'.'
- ・ その他、対象ネットワークでのアドレス指定用の特殊文字

以上の文字を使用した場合のアドレスの例を以下に示す。

1) SYS!USER@DOM.junet

ドメイン "DOM.junet" における担当システムまでメッセージが転送された後、そのシステムによって "SYS!USER" というアドレスとして解釈される。この場合は UUCP 形式であるため、"SYS" というシステムの "USER" に対してメッセージが転送される。

2) USER%DOM\_A.NET@DOM\_B.junet

"DOM\_B.junet" において "USER@DOM\_A.NET" へのメッセージであると解釈される。この際、'%' 記号は '@' 記号に変換され、"USER@DOM\_A.NET" として解析を行う。この表現における "DOM\_B.junet" の部分は即ち宛先ネットワークに対する JUNET 内のゲートウェイを指定している。(但し従来通り、著名なネットワークに対してはゲートウェイシステムの指定を必要とせず直接 "USER@DOM\_A.NET" の表現を用いてのメッセージの転送が可能である。)

3) USER%DOM\_A.NET\_A%DOM\_B.NET\_B@DOM\_C.junet

JUNET 内のゲートウェイ "DOM\_C.junet" にて

"USER%DOM\_A.NET\_A%DOM\_B.NET\_B" が解釈されるが、この際、もっとも右側にある '%' 記号が '@' 記号に変換される。即ち、"USER%DOM\_A.NET\_A@DOM\_B.NET\_B" として解析が行われる。この表現は、JUNET と直接リンクのある "NET\_B" を通じて "NET\_A" にメッセージを転送する場合のものである。また、一部のネットワークに対しては、JUNET 内のゲートウェイを指定せず、USER%DOM\_A.NET@DOM\_B.NET\_B と記述することも可能である。

4) SYS!USER%DOM\_A.NET\_A@DOM\_B.junet "DOM\_B.junet" にて

"SYS!USER%DOM\_A.NET\_A" を "SYS!USER@DOM\_A.NET\_A" に変換して、"DOM\_A.NET\_A" を評価してメッセージの転送を行う。但し、メッセージを転送されたネットワーク側で "SYS!USER@DOM\_A.NET\_A" の解釈をどの様に行っているかについて十分な注意が必要である。

### 3. UUCP 形式のアドレス表現

バージョン6.2のルールでは、UUCP 形式のアドレス表現内にドメイン表現を折り込む



ことを可能にしている。つまり、"USER@DOM"を"DOM!USER"として表現することができる。ここで、"SYS!USER"の表現と区別するために、DOMには少なくとも1つの'!'を含まなければならない。そこで、"USER@junet"という記述は、"junet.!USER"のように、ドメイン名の後に'!'を付けることで区別する。

例として、

SYS\_\_A!DOM.junet!SYS\_\_B!USER

という表現が用いられた場合、メッセージは先ず"SYS\_\_A"に転送される。"SYS\_\_A"では宛先アドレス"DOM.junet!SYS\_\_B!USER"に対して、この表現を標準形式に変換して解釈を行う。即ちこの例では、宛先アドレスは"SYS\_\_B!USER@DOM.junet"として解析される。

この機能により今までドメインベースのアドレスが使用できなかったsendmailを持たないシステムからでもドメイン名を折り込んだアドレスを用いてメッセージを転送することが可能となった。この場合の一般的なアドレス形式は以下の通りとなる。 <もっとも近いsendmailを有するシステムまでのpath>!<ドメイン名>!USER

#### 4. 最後に

今回リリースされるバージョン6.2ルールにおいては、これまで不明瞭な部分の多かった'!'と'@'の取り扱いやJUNET以外のネットワークとの関係に対しそのアドレス体系を明瞭にした。ここで取り上げた規則は国際的な取り決めに従ったものであり、今後大きく変化することはないと考えられる。今回のリリースでは、'@','%','!'以外の演算子を用いるネットワーク(DECNET, EAN, COSACなど)へのメッセージ交換については対処していない。これらについては、次のリリースに取り込まれると考えられる。

#### 付録

JUNETから直接アドレスを指定できるネットワークは次の通りである。

CSNETuser@site.CSNET

ARPA-NET

user@domain.EDU

user@domain.COM

user@domain.GOV

user@domain.NET

user@domain.{EDU,COM,GOV,NET}.arpa -

ACSNETuser@domain.OZ

---

-{x,y,z}はx、y、zのいずれかを示す。

付録 D

sendmail.cf作成システム(mailconf6.2J)

JUNETバージョン6.2ルール作成システム-mailconf-  
TueNov412:55:22JST1986

田中啓介、村井純  
篠田陽一、加藤朗

Junet運営委員会

## 概要

JUNETでは今回バージョン6.2のメールアドレス解析ルールが採用された。ここでは、新バージョンのルールに基づき、各システムにおけるメールアドレス解析ルールであるsendmailの構成ファイルsendmail.cfを作成するためのツールであるmailconfについての解説を行う。

### 1.システム使用方法の概要

本システムを使用して新しいsendmailのルールファイル"sendmail.cf"を作成する手順は以下の通りである。

- 1) システムのドメイン構造内での管理関係、およびネットワークの結合状況を反映させたデータベースを作成する。(ファイル名"desc.dat"-その書式は後述)
- 2) ドメインの別名の登録(ファイル名"alias.dat")
- 3) 使用mailer(メッセージ転送を担当するシステム)の登録(ファイル名"mailer")
- 4) 特殊なmailerを使用する場合、その登録。(ファイル名"mailer名.mst"但し、標準となっているjunet,uucp,smtpの各mailerのみを使用する場合、登録の必要はない。)
- 5) mailconfを起動して新しい"sendmail.cf"を作成。
- 6) 新しい"sendmail.cf"を使ってsendmailを再起動させる。

### 2.mailconfのインストール

ルールジェネレータ"mailconf"のインストール手順は以下に示す通りである。

- 1) 以下のディレクトリを設定し、"Makefile"にその情報を反映させる。

TMPD:テンポラリファイル作成用ディレクトリ  
(default:/tmp)

DATAD:データファイル用ディレクトリ  
(default:/usr/local/junet/data)

MSTD:マスタファイル用ディレクトリ  
(default:/usr/local/junet/master)

DESTDIR:"mailconf"インストール用ディレクトリ  
(default:/usr/local/bin)

これらのディレクトリが存在しない場合にはこの時点で速やかに作成しなければならない。

- 2) インストールを行うため、  
    #make install  
    を実行する。
- 3) データファイルを作成する。  
    "alias.dat"及び"mailer"が存在しなくてもルールの作成は可能であるが、"desc.dat"は必ず存在しなくてはならない。これらの3つのファイルは、同一ディレクトリになければならない。
- 4) マスタファイルの作成・変更。  
    標準的なアドレス解析ルールを変更する場合にのみ"sendmail.mst"を修正する。

### 3.ルール作成手順の詳細。

新ルール作成の詳細な手順は以下に示す通りである。

```
%cd /usr/local/junet/data
{データベースファイルの存在するディレクトリへ移動}
%vi desc.dat
{基本データベースの変更}
%vi alias.dat
{ドメインの別名の登録}
%vi mailer
{特殊な mailer の登録-必要な場合のみ-}
%cd /usr/local/junet/master
{sendmail.cfの元データ用のディレクトリへ移動}
%vi sendmail.mst
{特殊な mailer のための変更-必要な場合のみ-}
%vi <mailer__name>.mst
{特殊な mailer のための登録-必要な場合のみ-}
%cd /tmp
{新しい"sendmail.cf"を作成するための作業用ディレクトリへ移動}
%mailconf
{新しい"sendmail.cf"の作成}
%su
#cd /usr/lib
#mv sendmail.cf sendmail.cf.old
```

```

{古い"sendmail.cf"の保存}

#cp /tmp/sendmail.cf
  {新しい"sendmail"の登録}
#rm sendmail.cf
  {古い"sendmail.cf"のメモリイメージの消去}
#kill-KILL<process_id_of_sendmail_daemon>
  {古い"sendmail.cf"を利用しているsendmailサーバの終了}
#sendmail -bd -t1h
  {新しいsendamilサーバの起動}
#exit
%rm /tmp/sendmail.cf

```

#### 4. mailconfの起動

mailconfの呼び出し形式は以下の通りである。

```
mailconf[-ddatadir][-mmasterdir][-ooutput]
```

この実行によりそのシステムのネットワーク環境を反映したデータベースを元にsendmailのルールファイル"sendmail.cf"をカレントディレクトリに作成する。

-dオプションは、データベース用ディレクトリ(MakefileではDATADで定義)に対して、設定値以外のディレクトリを使用する場合にそのディレクトリを指定する。-mオプションは、マスタファイル用ディレクトリ(MakefileではMSTDで定義)に対して、設定値以外のディレクトリを使用する場合にそのディレクトリを指定する。-oオプションは出力ファイルの名前に対して設定値の"sendmail.cf"以外の名前を使用する場合にその名前を指定する。

#### 5. 各データベースの記述

##### 5.1. mailer

バージョン6.2ルールにて標準で利用可能なmailerとしては、

・JUNET mailer

(UUCPリンクにおいてドメインベースのアドレス指定を行うためのmailer)

・SMTP mailer

(TCP/IPリンクにおいてドメインベースのアドレス指定を行うためのmailer)

・UUCP mailer

(UUCPリンクにおいて!を用いたアドレス指定を行うためのmailer)

・LOCAL mailer(同一システム内のユーザにメッセージを転送するためのmailer)が設定されており、それぞれjunet,smtp,uucp,localというシンボルを与えられている。また、特殊なmailerとして、JUNET以外のゲートウェイ用のmailerとして働くGJUNETmailer(シンボルはgjUNET)がある。

これらの標準mailer以外のmailerを利用する場合以下の登録が必要である。

## 5.2. ドメイン管理・ネットワーク環境用データベース

desc.dat

"desc.dat"は2つの部分から成り立っている。1つはドメインの管理関係を記述する部分であり、もう1つはネットワークの接続環境とメッセージ転送のための経路情報を記述する部分である。

この2つは"desc.dat"内でどのような順番で記述されてもよく、またコメント("#"記号から行末まで)もどこに位置しても構わない。

### 5.2.1. ドメインの管理関係

ドメインの管理関係の記述は以下の書式に従って行われる。

```
name:<full_domain_name>
```

```
lower_level_name:<list_of_names_in_this_domain>
```

<full\_domain\_name>の項には、このシステムが管理を担当するドメイン名の正式名称を記述し、の項には、そのドメインに所属するドメイン名を総て記述する。通常のメールアドレスのように、'junet'の省略は出来ないことに注意されたい。

この記述は管理を担当するドメインの数だけ存在しなければならない。例えば、"sysA.dom1.dom2.junet"なる正式名称を持つシステムが"dom1.dom2.junet"ドメインの管理をも担当する場合の記述は、

```
#for'dom1.dom2.junet'  
name:dom1.dom2.junet  
lower_level_name:sysAsysB  
#for'sysA.dom1.dom2.junet'  
name:sysA.dom1.dom2.junet  
lower_level_name:NONE
```

となる。

### 5.2.2. ネットワーク接続環境

この記述は以下の書式に従って行われる。

```
<destination>:<mailer>
```

```
<list_of_domains_to_be_fowarded>
```

ここで<destination>には直接通信の可能なシステム名を、<mailer>にはそのシステムにメッセージを転送する際に用いるべきmailer名を、指定する。<list\_of\_domains\_to\_be\_fowarded>にはそのシステムに解決を以てするドメイン名のリストをそれぞれ記述する。ここで、mailer名として標準で使用できるのは前述のjunet,smtp,uucp,local,gjunetの5つである。

例えば、ドメイン名"sysB.dom1.dom2.junet"を宛先とするメッセージは、"smtp"mailerで"sysB"に転送し、"dom2.junet"や"junet"を宛先とするメッセージを"junet"mailerで"sysC"に転送するための記述は以下の通りである。

```
sysB:smtpsysB.dom1.dom2.junet
```

sysC:junet  
dom2:junet  
junet

### 5.3. ドメイン別名データベース-alias.dat-

ドメイン名に別名を定義するための書式は以下の通りである。

<real\_name>:<list\_of\_alias\_names>

双方の表現ともfull\_domain\_nameでなくてはならない。(最後の"junet"の省略は出来ない)

### 5.4. 特殊mailer宣言データベース-mailer-

このファイルは、標準で設定されているもの以外のmailerを利用する場合、及び"!形式で表現された宛先をもつメッセージに対しての特殊な処理を定義する際に使用される。

#### 5.4.1. 標準以外のmailerの使用宣言

宣言は以下の形式で行われる。

M<mailer\_name>

<mailer\_name>として、既に定義されているmailer名は使用できない。

この宣言を行った場合には、マスタファイル用ディレクトリにそのmailerの起動方法の記述及び"From:","To:"フィールドにて用いられるアドレスへの変換の記述を"sendmail.cf"の記述法に従って行ったものをファイル名"<mailer\_name>.mst"として登録しておかなければならない。

また、そのmailerのために特殊なアドレス解析が必要となる場合には"sendmail.mst"ファイルにその処理の内容を記述しなければならない。

mailerを定義した場合、そのmailerを使用すべき隣接システム名のリストがsendmailルールのクラスとして定義されるが、クラス名が何になるかは、mailer数により異なる。mailer名.mstの中で、そのmailerに割り当てられるクラス名は、'&&'で他のmailerに割り当てられているクラス名の参照は、'&mailer名&'として行う。また、'&'自身は'¥&'で、'¥'は'¥¥'として記述する。

また、本生成系は、mailer間におけるルールセットの衝突に対しては、何の検査も行っていないので注意されたい。

#### 5.4.2. "!形式アドレスに対しての特殊処理

"mailer"ファイル内に

U<system\_name><mailer\_name>

の記述を行うと、"!を用いて経路指定を行ったアドレスを持つメッセージを無条件に<mailer\_name>で指定されたmailerを使用して<system\_name>に転送する。この記述は、ローカルエリアネットワーク内でUUCPアドレスの解決システムが決定されている場合にそのシステムまでの経路を記述する必要をなくすものである。

-クラスの概念に関しては4.2BSD/4.3BSDのsendmail関係の文献参照のこと。

但し、

sysX!sysY!sysZ!user

形式のアドレスに対して、"sysX"がそのシステムから直接通信可能であるシステムである場合にはこのルールは適用されない。

#### 5.5. sendmail.cfベースファイル-sendmail.mst-

このファイルは、特殊なアドレス解析ルールを折り込む場合以外には、変更の必要はない。

変更の必要がある場合には、"sendmail.cf"の書式に従って必要部分を変更する。但し、行頭が"\*\*\*\*\*<"で始まっている行を変更した場合、"mailconf"で正しいルールが形成されなくなる可能性がある。

#### 謝辞とお詫び

新ルール及び本システム作成にあたり、貴重な御意見をいただいたJUNET利用者の皆さんに感謝する。また、ルールテストのため一部のメール転送に不都合が生じたことを深くお詫びする。

#### バグ情報

現在のバージョンでは、'!'や'::'を演算子とするネットワーク(COSAC,EAN,DECNETなど)へのアドレス変換はサポートされていない。新ルール及び本システムのバグについての報告、その他のコメントを次のアドレスに電子メールによって送付していただければ幸いである。

junet-admin@junet



付録 E

sendmail.cf作成システム(mailconf6.3J)

# JUNETバージョン6.3ルール作成システム-mailconf-

田中啓介、村井純  
篠田陽一、加藤朗  
Junet運営委員会

## 概要

基本的な考え方はmailconf6.2Jと同様であり、特別な処理を必要としない限り、6.2J用のdesc.datが使用出来ますが、その他のファイルに若干の書式変更があります。注意してください。

-----  
目次  
-----

- 1)mailconfのデータファイルの記述
- 2)sendmail用ルール作成の手順
- 3)mailconfの応用事例
- 4)mailconf6.3Jでの変更点

ドキュメントとしては以下の文書が存在します。

|           |                                                    |
|-----------|----------------------------------------------------|
| whatis6.3 | :mailconfによって作成されるsendmail用のルールに関して6.3Jでの変更点を示します。 |
| howtouse  | :mailconfを使用したsendmail用のルール作成手順を示します。              |
| datafile  | :データ、実行等に関して6.2Jからの変更点のみを示します。                     |
| tech      | :mailconfの応用について若干の例を示します。                         |

また、6.2Jルールのドキュメント付録Eを参照して下さい。アドレスの構造等についてはこちらに詳しいものがあります。

## 1)mailconfのデータファイルの記述

Keisuke'Kieko'Tanaka

### 1.はじめに

ここでは、mailconfによって作成されるsendmail用の規則を制御するためのデータファイルの書式について示す。

### 2.データファイル

データファイルは

1)名前管理関係、経路制御、メッセージ転送機構の制御を行なうためのデータ

- ・主descriptionfile:desc.dat
- ・別名定義ファイル:alias.dat
- ・mailer定義ファイル:mailer.dat

2)sendmail.cfの元データ

- ・sendmail.cfの骨格:sendmail.mst
- ・ホストに依存した規則:host.mst
- ・mailerに依存した規則:<mailer>.mst

から成立しているが、一般的に各サイトで記述しなければならないのは1)に含まれるdesc.datとalias.datのみである。

これ以外のファイルは、

・特殊なメッセージ転送機構を使用する場合

=>mailer.dat,<mailer>.mst

・特殊な環境にあるホストである場合

(概念の異なるアドレスを使用しているネットワークとのゲートウェイであるなどの理由による)

=>host.mst,sendmail.mst,..

・JUNETと異なる管理体系のネットワークにも属している

=>sendmail.mst,...

という状況でしか修正の必要はない。

## 3.各データファイルの説明

### 3.1desc.dat

基本的な記述様式は6.2J版と同様であり、6.2Jで使用していたデータがそのまま使用可能である。

#### 3.1.1.基本的な記述

"desc.dat"は2つの部分から成り立っている。1つはドメインの管理関係を記述する部分であり、もう1つはネットワークの接続環境とメッセージ転送のための経路情報を記述する部分である。

この2つは"desc.dat"内でどの様な順番で記述されてもよく、またコメント("#"記号から行末まで)もどこに位置しても構わない。

### 3.1.1.1.ドメインの管理関係

ドメインの管理関係の記述は以下の書式に従って行われる。

```
name:
    <full_domain_name>
lower_level_name:
    <list_of_names_in_this_domain>
```

<full\_domain\_name>の項には、このシステムが管理を担当するドメイン名の正式名称を記述し、の項には、そのドメインに所属するドメイン名を総て記述する。通常のメールアドレスのように、'junet'の省略は出来ないことに注意されたい。

この記述は管理を担当するドメインの数だけ存在しなければならない。例えば、"sysA.dom1.dom2.junet"なる正式名称を持つシステムが"dom1.dom2.junet"ドメインの管理をも担当する場合の記述は、

```
#for'dom1.dom2.junet'
name:
    dom1.dom2.junet
lower_level_name:
    sysAsysB
#for'sysA.dom1.dom2.junet'
name:
    sysA.dom1.dom2.junet
lower_level_name:
    NONE
```

となる。

### 3.1.1.2.ネットワーク接続環境

この記述は以下の書式に従って行われる。

```
<destination>:<mailer><list_of_domains_to_be_fowarded>
```

ここで<destination>には直接通信の可能なシステム名を、<mailer>にはそのシステムにメッセージを転送する際に用いるべきmailer名を、指定する。<list\_of\_domains\_to\_be\_fowarded>にはそのシステムに解決を以てするドメイン名のリストをそれぞれ記述する。

例えば、ドメイン名"sysB.dom1.dom2.junet"を宛先とするメッセージは、"smtp"mailerで"sysB"に転送し、"dom2.junet"や"junet"を宛先とするメッセージを"junet"mailerで"sysC"に転送するための記述は以下の通りである。

```
sysB: smtp sysB.dom1.dom2.junet
sysC: junet dom2.junet
      junet
```

mailerに関して、今回配布されたmailconfにおいてデフォルトで使用可能なのは

```
junet, junet-o, junet-n
uucp, uucp-o, uucp-n
smtp, smtp-o, smtp-n
pmdf, tops,d echnet
```

である。(なお、junetはjunet-oと、smtpはsmtp-oと、uucpはuucp-oと同じものである。)一般的なサイトにとってはjunet,smtp,uucpのみの使用で十分であろう。また、

pmdf,tops,deconetについては初回の配布分には含まれていない、最終テスト後2回目の配布分とする。

それぞれのmailerでの特徴は後述のmailer.dat及び<mailer>.mstによって規定される。

### 3.1.2.desc.datの共有

名前指定でシステムを表現する記述として

```
name:
    *.cc.u-tokyo.junet
lower__level__domain:
    NONE
```

という\*を使った記述が可能となった。\*部分は実行時に-hで指定されたホスト名、あるいは-hでの指定がない場合には実行ホスト名が割り当てられる。もちろん、システム名を明記する従来の記述も可能である。

これによって例えば、slave.datに

```
name:
    *.cc.u-tokyo.junet
lower__level__domain:
    NONE

ccut:  smtp  ccut.cc.u-tokyo.junet
        cc.u-tokyo.junet
        u-tokyo.junet
        junet
```

の記述を行ない

```
mailconf-s-h<machine__name>
```

をcc.u-tokyo.junetドメインに属する全てのマシンに対しての<machine\_\_name>で行なうと一つのデータファイルで簡単にルールファイルが作成できる。

## 3.2.alias.dat

### 3.2.1.基本的な記述

記述様式は6.2J版と同じ。

正式名: 別名

です。ここでの正式名とは全く省略のないアドレスを指す。

例えば、

```
junet:      jp,jpn
u-tokyo.junet:  utyo.junet
handai.junet:  handai.junet
```

によって、junetの別名としてjpとjpnが、u-tokyo.junetの別名としてutyo.junetなどが使用できる。

### 3.2.2.階層の異なる別名

例えば、組織内のネットワークの拡大に伴って、新たに中間的なドメインを設立する場合など、例えばccut.u-tokyo.junetであったものを中間ドメインccの新設に伴ってccut.cc.u-

tokyo.junetとする場合、

```
ccut.cc.u-tokyo.junet:ccut.u-tokyo.junet
```

の設定を行なえば、ccut.u-tokyo.junetは全てccut.cc.u-tokyo.junetに変換されます。

### 3.3mailer.dat

書式が変更になったので6.2J版のものは使えない。以下に例を示す。

```
#mailer name    address    domainaddress
M junet-n       notouch    nondelete   #permitanyaddressingscheme
M junet-o       domain     nondelete   #oldjunet?user%dom1@dom2
M junet         domain     nondelete   #oldjunet?user%dom1@dom2
M smtp-n       notouch    nondelete   #
M smtp-o       domain     nondelete   #
M smtp         domain     nondelete   #
M pmdf         route     nondelete   #force'@dom1,@dom2:user@dom3'
M uucp-o       bang       delete      #force'sysA!sysB!user'
M uucp-n       bang       nondelete   #force'sysA!sysB!dom1!user'
U uucp-o       #UUCP     mailer
S smtp-o       #SMTP     mailer
```

#から行末まではコメントとして扱われる。

行頭の文字(M,S,U)はコマンド種別を示し、

Mはmailer設定

Uは解決不可能なhost!user形式を解決するためのmailer

(例えば、"desc.dat"には明記していないUUCPlink等を用いた、host!user形式のmailを  
処理するためのmailer)

Sはuser@[210.10.10.1]形式のアドレスに対して選択されるべきmailerをそれぞれ指定する。

Mコマンドの場合:

第3フィールド(notouch,domain,route,bang)はアドレスタイプを示し、recipientのアドレス(rmailやsendmail起動時の引数となるアドレス)に対しての変更すべき形式を示す。

domainは全てuser%domain%domain@domainタイプに変更する。

routeは全てdomain,domain:user@domainタイプに変更する。

bangは全てhost!host(domain)!userタイプに変更する。

notouchは変更を行なわない。

従って、先の例ではjunetmailerは@d,@d:u@dタイプのアドレスはu%d%d@dに変換してからメッセージの転送を行なう。

第4フィールド(nondelete,delete)はBANGタイプのmailerに対してのみ有効であり、

deleteはアドレス内にドメインの記述('を含む記述)を許さない

nondeleteはアドレス内にドメインの記述を許す

ことを指定する。

また、Mで指定されたmailerに対しては転送機構の呼び出し形式、及びヘッダ情報が記述された<mailer>.mstが存在しなければならない。

### 3.4\*.mst

master-dirに存在するファイルの中では以下のコマンドが使用可能である。ここでコマンドの開始・終了示す文字としては'&'を使用している。記述の中で'&'をそのまま使用する際には'¥¥'を先行させなければならない。'¥¥'そのものを示すためには"¥¥¥¥"となる。

|            |                                                                                                     |
|------------|-----------------------------------------------------------------------------------------------------|
| &Clabel&   | あるマクロ文字を割り当てる。                                                                                      |
| &Daddress& | addressをsendmail.cfの内部表現に置換する。(例えば、"&Dboko.cc.u-tokyo.junet&"は"boko.UPLVLA"に置換されてsendmail.cf上に現れる。) |
| &Ldomain&  | domainをマクロ文字を使用した表現に置換する。(例えば、"&Lboko.cc.u-tokyo.junet&"は"boko.\$X"に置換されてsendmail.cf上に現れる。)         |
| &Mmailer&  | mailerに割り当てられたマクロ文字に置換する。(例えば、"&Msmtpl&"は"S"に置換されてsendmail.cf上に現れる。)                                |

また、特殊なものとして

|                |                                     |
|----------------|-------------------------------------|
| "<mailer>.mst" | 内でしか使用できないものとして                     |
| &Smailer&      | mailerのFrom:フィールドを変換するためのルールセットの番号。 |
| &Rmailer&      | mailerのTo:フィールドを変換するためのルールセットの番号。   |
| "sendmail.mst" | 内で使用している                            |
| &Ifile&        | ファイルの挿入                             |
| "host.mst"     | 内で使用する                              |
| &Hlabel&       | hostdependent情報の定義                  |

がある。

それぞれの例についてはexampleのファイルを参照のこと。

注意:

&Clabael&コマンドについては以下のラベルがデフォルトで定義されている。

|                 |                                                  |
|-----------------|--------------------------------------------------|
| &Cmyhost&       | ホスト名(例えば、"clapton")を示すマクロ文字。                     |
| &Cmydomain&     | 正式ドメイン名(例えば、"clapton.cc.u-tokyo.junet")を示すマクロ文字。 |
| &Cmyparentnode& | 上位ドメイン名(例えば、"cc.u-tokyo.junet")を示すマクロ文字。         |

### 3.4.1host.mst

以下に"host.mst"の例を示す。

```
#hostdependrewritingrule
#for'ccut.cc.u-tokyo.junet'whichisCSNET-JUNETgateway
```

```

&Hmacro&
#hostdependentmacros

#####
# CSNET-JUNET gateway specifics #
#####
#our domain/host name in CSNET
D&Cour-domain-name-in-csnet&utokyo-relay.csnet
#D&Cour-domain-name-in-csnet&japan

#These are CSNET domains known in JUNET.
C&Ccsnet-domain&arpa bitnet csnet
C&Ccsnet-domain&edu com gov mil net org
C&Ccsnet-domain&au de fr il kr nz uk us
#C&Ccsnet-domain&jUNET

&Hcleanup&
# host dependent cleanup(S8)
# none

&Halloc-mailer&
# host dependent mailer allocation
R<@relay.cs.net>,$*    @$>9$1    strip'relay.cs.net',retry
R<@relay.cs.net>:$*    @$>9$1    strip'relay.cs.net',retry
R$*<@relay.cs.net>    @$>9$1    strip'relay.cs.net',retry

&Hrewrite&
# mailer dependent rewriting
# none

```

&Hmacro&以下の記述はsendmail.cfの先頭のマクロ定義部に挿入される。ここでは通常sendmail.cf内部で使用するマクロの定義を行なう。この例では&Cour-domain-name-in-csnet&、&Ccsnet-domain&に対するマクロを定義している。現在&C&コマンドによるマクロの定義は&Hmacro&部でしか行なえず、他の場所では参照しかできない。

&Hcleanup&以下の記述はS8としてsendmail.cf内に挿入される。これは全ての解析に先だって参照されるので解析の前処理を行なう場合に記述を行なう。

&Halloc-mailer&以下の記述はmailerの選択を行なうS7の呼び出しを行なう前に実行される。mailerに依存しない書き換え等を行なう場合に記述する。この例では"relay.cs.net"を削除するために使用している。(S9は解析を最初からやり直す。)

&Hrewrite&以下の記述はmailerの選択後、実際の送信の前に実行される。mailerに依存した書き換えが必要な場合に記述する。mailer選択後のアドレスは<@(target\_\_host).(mailer).mailer>:(address)の形式で与えられる。

### 3.4.2<mailer>.mst



以下にjunet.mstの例を示す。

```
#####  
#####JUNETMailerspecification  
#####  
# Old JUNET UUCP Mailer specification (single recipient)  
Mjunet,      P=/usr/bin/uux,F=sDFMhuUP,S=&S&,R=&R&,  
              A=uux--r$h!rmail($u)
```

```
#####OldJUNETS/Rrewritingrules(norouteaddress)  
# From: field  
S&S&  
# handle old uucp style  
R$*<@$.OSTYLE>    @$w!$2!$1          thishost!h!u  
# non-route address  
R$*<@$+>          @$1<@$2>          alreadyok  
# route address  
R<@$+>$+          $:$>27<@>:<@$1>$2      elim.route-addr  
R<@>:$+          @$1          remove<@>:  
# name  
R$+              @$1<@$&Cmydomain&>  tackonourhostname  
# To: field  
S&R&  
# none  
R$*<@$+>$*        @$>19$1<@$2>$3        tackdomainname  
R$*@$*          @$1@$2          alreadyok  
R$+              @$1<@$&Cmydomain&>  tackonourhostname
```

記述中、&S&と&R&はそれぞれ自分自身に割り当てられたルールセット番号に置換される。  
次はsmtp.mstの例である。

```
#####  
#####SMTP Mailer specification  
#####  
# Old JUNET SMTP Mailer specification  
Msmtp,      P=[IPC],F=msDFMuXP,S=&Sjunet&,R=&Rjunet&,  
              A=IPC$h,E=¥ ¥ r ¥ ¥ n  
##### Old JUNET S/R rewriting rules (no route address)  
# From: field-defined in 'junet.mst'  
  
# To: field-defined in 'junet.mst'
```

ここで、&Sjunet&と&Rjunet&はそれぞれ junet mailer に割り当てられたルールセット番号に置換される。

### 3.4.3sendmail.mst

書式はsendmail.cfと同じであるが、先に示した&.&のコマンドが使用できる。また、&Ifile&はファイルの挿入位置を示している。

#### 4.おわりに

書式、内容等には改善の予知があると思われます。ご意見等ございましたらお知らせ下さい。また、説明が不十分な点があるかもしれません、mailでお問い合わせ下さい。

## 2)JUNET用sendmailルール6.3J対応mailconfの使い方

### 1.はじめに

6.3J対応のmailconfでは、6.2J版よりも柔軟性を高めたつもりです。そのためかえっていろいろな指定を行わなくてはならなくなりました。不便な点が残っているとは思いますが、改善案等ありましたらお知らせ下さい。

### 2.ディレクトリ

src/ :mailconfのソースプログラムが存在するディレクトリ。  
DESC/ :descriptionfile用のディレクトリ  
(以下、desc-dirとする)  
MST/ :masterfile用のディレクトリ  
(以下、master-dirとする)

### 3.コンパイル

srcディレクトリに移動した後、各サイトの状況に応じたセットアップを行なう。

#### 3.1.Makefile

コンパイル時のオプションとしては以下の2つが指定できる。

-DBSD : 4BSD系のOSである場合。  
-DDEBUG : デバッグ情報を得る場合。

#### 3.2.defs.h

desc-dirとmaster-dirに関する設定はこのファイル内で行なわれる。その他にも、以下の情報が設定される。

DESCDIR : desc-dirの設定(デフォルト値は"./DESC")  
MASTERDIR : master-dirの設定(デフォルト値は"./MST")  
IGNRCASE : ドメイン名に関しての大文字・小文字の区別をしない場合に設定する  
GETHOSTNAME : ライブラリ関数gethostname();を持っている場合に設定する  
UNAME : ライブラリ関数uname();を持っている場合に設定する  
SORTCOM : sortコマンドのパス名を設定する

#### 3.3.mailconfの作成

srcディレクトリでmakeを実行する。

## 4.データファイルの設定

以下のデータファイルを各システムに応じて設定する。

desc-dir:  
・名前管理関係・経路記述ファイル desc.dat  
・別名設定ファイル alias.dat  
・mailer設定ファイル mailer.dat

master-dir:

|                             |              |
|-----------------------------|--------------|
| ·host固有のrewritingrule設定ファイル | host.mst     |
| ·sendmail.cfのもととなるデータ       | sendmail.mst |
| ·各mailer固有のrewritingrule    | <mailer>.mst |

以上のファイルがそれぞれのディレクトリに存在していなければならない。なお、desc-dirはコンパイル時のDESCDIRによって設定され、あるいはmailconf実行時の-dオプションによって指定される。また、master-dirはコンパイル時MASTERDIRによって設定、実行時-mオプションによって指定が可能である。

各データファイルの書式に関しては別の項で述べる。

## 5.mailconfの起動

```
mailconf[-ddesc-dir][--mmaster-dir][-ooutput][-hhostname][-s][-xd]
```

各オプションの説明

- d: desc-dirをコンパイル時の設定とは異なるものとする。
- m: master-dirをコンパイル時の設定とは異なるものとする。
- o: 出力されるファイル名
- h: ホスト名の指定  
(-hと指定するとホスト名として、mailconfが実行されたホストの名前が採用される。)
- s: master/slaveの選択  
(“desc.dat”のファイル名についての項を参照のこと)
- x: デバッグレベルの設定

ここで、出力ファイル名のdefault値は-hの指定がない場合は“./sendmail.cf”であるが-hによるホスト名の指定があると“./<host\_name>.cf”となる。

また、主データファイルdesc.datはそのファイル名に関して以下の規則に基づいてサーチされます。

- h 指定が行なわれない場合は“desc.dat”を使用する。
- h でホスト名が指定された場合は先ず“<host>.dat”を探し、  
“<host>.dat”が存在しない場合には“master.dat”を探し(但し、-sオプションによりスレーブ指定が行なわれている場合には代わりに“slave.dat”を探す。)  
“master.dat”/“slave.dat”が存在しない場合には“desc.dat”を使用する。

いずれのファイルも記述の様式は同じです。

## 6.snedmail.cfのinstall

作成されたsendmail.cfを目的マシンの/usr/lib/sendmail.cfとしてinstallします。

freezeされたルールファイル(/usr/lib/sendmail.fc)を使用する場合には

```
/usr/lib/sendmail-bz
```

の実行が、sendmailがSMTPサーバとして動作している場合には

```
sendmail -bd -q??
```

の再起動が必要です。

### 3)mailconf活用法

#### 1.はじめに

ここでは、mailconfを使って作成するsendmail.cfに対して付加的な機能を加えるための手法を示します。

#### 2.ドメインマスタ機能の分散

例えば、junetドメインのドメインマスタ機能を複数のマシンで実現する場合、真のドメインマスタ(user@junet形式のアドレスを処理するマシン)以外のマシンではdesc.datの記述を次のようにします。

```
name:
    junet
option:
    not__local
lower__level__name:
    u-tokyo
    titech
..
```

こうすると、そのマシンではXXX.junetというアドレスに対してXXXが存在するかしないかのチェックを行ない、存在していない場合はそのmailの発信者に対してエラーメッセージを返します。但し、user@junet形式のアドレスを指定されたメッセージのみは真のドメインマスタに転送します。

### 3.From:フィールドの変更

例えば、junetmailerやsmtpmailerでのFrom:アドレスのコントロールはjunet.mst内で行なわれています。ここでは、通常

```
From:user@system.domA.domB.junet
```

というアドレスを生成しますが、これをシステム名を除いた代表的なアドレス

```
From:user@domA.domB.junet
```

に変更したい場合は以下のような変更を必要とします。

```
From: を生成するための規則(junet.mst内)
# From: field
S&S&
# handle old uucp style
R$*<@$.OSTYLE>          @$w!$2!$1          thishost!h!u
```

```

# non-route address
R$* <@$+>          $$1 <@$2>          already ok
# route address
R <@$+>$+          $:>27 <@>:<@$1>$2      elim.route-addr
R <@>:$+          $$1remove <@>:
# name
R$+                $$1 <@$&Cmydomain&>    tack on our hostname
    の最後の行
R$+                $$1 <@$&Cmydomain&>    tack on our hostname
    を
R$+                $$1 <@$&Cmyparentnode&> tack on our hostname
    に変更します。

```

&Cmydomain&は対象システムの正式なドメイン名(例えばclapton.cc.u-tokyo.junet)、&Cmyparentnodeはそのドメイン名からシステム名のみを削除した名前(cc.u-tokyo.junet)を表すマクロにそれぞれ置換される。

さらに異なる名前、例えば"u-tokyo.junet"に変更したい場合はホストに依存する規則を記述するためのファイルhost.mstに

```

&Hmacro&
# host dependent macros
D&Cgeneric_from&u-tokyo.junet

```

の記述を加え、先の行を

```

R$+                $$1 <@$&Cgeneric_from&>

```

に変更するとよいでしょう。

但し、このような変更を行なう際に、あまり複雑な変更を行なうと代表ドメインマスタとaliasなどの関係によってはループしてしまうことがあります。注意して下さい。

#### 4.他のネットワークでの使用

このmailconfは、JUNET用の規則を生成しますが、決してJUNET専用ではありません。ソース及びマスタファイル(\*.mst、但しhost.mstは除く)内のどこにも、コメントを除いて再上位のドメインをjunetに特定するようなことは行なっていません。JUNETと同じ解析のアルゴリズムを採用するのであれば、どこのネットワークでも使用できるように設計しています。

#### 4)sendmail規則バージョン6.3Jの特徴

Keisuke'Keiko'Tanaka

##### 1.はじめに

以下の記述はバージョン6.3JのJUNET用sendmailルールの旧バージョン(6.2J)からの変更点のみを扱っています。従って、6.2Jに基づくアドレス表記に関しての理解を前提としています。

## 2. 取り扱うアドレスについて

6.3Jは基本的に6.2Jのbug-fixであり、特に日本国内のサイトとしか接続のないサイトに関してはそれほど大きな変更はありません。

### 2.1. ドメインによる経路指定について

大きな追加機能としては、RFC822の規定に基づくドメインを使った経路指定

```
@domain1,@domain2,@domain3:user@domain4
```

形式のアドレスの取り扱いを可能にしています。これは、JUNETでよく使用される形式

```
user%domain4%domain3%domain2@domain1
```

と同様の意味を持ちます。これによって、直接CSNETとのリンクを持っているサイトが

```
@dom,@dom:user@dom
```

形式に対して解析エラーを生じることがなくなりました。但し、日本国内の使用に関しては従来通り

```
user%dom%dom@dom
```

の形式を使用した方がよいでしょう。

### 2.2. アドレスの形式変換のためのルールセットの内蔵

以下の形式のアドレス

```
@dom1,@dom2:user@dom3
```

```
user%dom3%dom2@dom1
```

```
dom1!dom2!dom3!user
```

の間の相互変換を可能とするルールセットを内部に持っています。

しかし、これらの相互変換を(意識して)利用するのは基本的に海外とのゲートウェイなど特殊な環境にあるシステムだけです。

例えば、東京大学のCSNET-JUNET間ゲートウェイではこの機能を使ってCSNETからの@d:u@d形式のアドレスを全てu%d@d形式に変換して国内に流しています。

### 2.3. source-routingは基本的に使用しない

また、6.3Jではドメイン形式を使用したアドレスを全ての基本としています。従って、

```
sysA!sysB!sysC!user
```

形式のアドレスは、解析は可能であっても出来る限り使用しない方向に持っています。

例えば、6.2Jではsource-routingを使用した場合のFrom:フィールドはやはりsource-routingの形式で形成されていました。6.3Jではこれを改め、ドメイン形式のアドレスが解析出来る相手である限りFrom:フィールドはドメイン形式で生成されます。

#### 2.4.somewhere!の排除

以前はrmailの変更で対処していたsomewhere!を解析の最初に排除するようにしました。

但し、CSNETからのメッセージに対してのrelay.cs.netの排除は国内の他のサイトには基本的に必要ないので、一般的な規則には含めていません。

このような特殊な文字列のアドレス内からの排除に関してはmailconfの使用法の項をご覧ください。

### 2.解析の手順について

#### 2.1.mailer選択と送信の分離

以前の規則でルールセットS0で同時に行なっていたmailerの選択と送信を分離しています。ルールセットS7(S0から呼ばれる)でmailerと送信すべきホストを決定した後、S0で実際の送信を行ないます。

これによって、あるmailer、あるいはhostを利用する場合に共通した書き換えを行なうことが可能となりました。

### 3.さいごに

以上の特徴は全て6.3J用のmailconfにおいて、データの記述、修正によって制御が可能となっています。

付録 F

UUCPインストレーションガイド(USENET版)



## INSTALLATION GUIDE README 5.6 85/06/24

This version is based on the "rti uucp" that was on the 4.2BSD tape. It contains many additional features and fixes from Usenet, Tom Truscott, Guy Harris, Lou Salkind, and many others.

A few new subdirectories have been added. Do a "make makedirs" to make sure that you have all of them.

I recommend that you change the uucp mail handler in your sendmail.cf to something like:

```
# Muucp, P=/usr/bin/uux, F=sDFMhuU, S=13, R=23, M=100000,  
# A=uux -r $h!rmail ($u)  
Muucp, P=/usr/bin/uux, F=sDFMmhuU, S=13, R=23, M=100000,  
    A=uux -r -z -a$g -gC $h!rmail ($u)
```

The -a\$g provides a return address in case mail fails (So it won't go to your machine! uucp anymore.) The -gC specifies a grade for mail. C is a good choice for mail. News should run at 'd'. This way, mail gets sent before news. The 'm' flag specifies that this mail can send to multiple addresses, which uux can.

The man pages now describe all the options for the various commands. Make sure you read them. A list of the functional differences is in "Changes".

You should also look through the UUAIDS directory. There are some useful programs and hints there in.

The maximum length of a site name has been changed from the old 7 to the 14. This is to be compatible with the HoneyDanBer uucp (aka BNU 1) which is as close to a standard uucp as there is. Sites which have sitenames longer than 7 characters that only send you the first 7 characters are broken and should be fixed. However, there is a way of compensating for this until they fix their problem. For each site you talk to that has a name longer than 7 characters, put a line in /usr/lib/uucp/L.aliases of

the form:

```
fullname name-truncated-to-7-characters.
```

E.g:

```
tektronix tektron  
lbl-csam lbl-csa  
rochester rochest  
ut-sally ut-sall
```

(See UUAIDS/L.aliases for more details on aliasing uucp names.) If the sitename is  $\leq 7$  characters, you don't have to do anything.

Rick Adams  
rick@seismo.ARPA  
June 19, 1985

Tom Truscott, rti!trt,decvax!duke!trt (919)541-6488 Research Triangle Institute Bob Gray,  
gray@berkeley, duke!adiron!bob (315) 336-4989

This is a variant of the uucp used at decvax, ittvax, rti, mcnc, adiron,duke and others. There have been tons of bug fixes and enhancements from people on the usenet (thank you). Speed is substantially improved. This version fixes essentially all of the McGeady's bug list.

Enhancements:

Dialers                Lots of dialers included.

Subdirectories        /usr/spool/uucp is now split into 7 subdirectories. This is a huge help on busy systems.

/usr/lib/uucp/L.cmds List of commands permitted for remote execution. A line of form 'PATH = ...' sets the search path.

expect-send sequence Escape characters now permitted: %r, %n.  
%r, not %n, is default char sent at end of string.  
%c        (put at end of string). Dont send ending %r.  
%d        pause 1 second (%d%d pauses 2 seconds)  
"" P\_ZERO`expect nothing, start sending zero parity.'  
P\_EVEN (default), P\_ODD, P\_ONE other parity modes.  
%05      Send a control-E  
"" ""    `expect nothing, send a %r'.

uupoll [sysname] Polls named system.

uusnap               Displays spooled files, and pending uuxqts.

This version runs on all VAXen and PDPs under UNIX-V7 and 4.1bsd, 4.2BSD. It also runs on Gould/SEL Concept series machines (e.g. 32/8750), DUAL 68000 unisoft. It also runs on BTL system III, IV, and V.

UUCP installers should read the two papers (by Dave Nowitz) in Vol 2B of version 7 manuals and UUAIDS/setup.tblms. Understand each step below before executing. Some steps will vary slightly from system to system.

\*\*\*\* INSTALLATION \*\*\*\*

1. If you are currently running uucp, save the old programs!:

```
su root
cd /usr/bin
for i in uucp uux uulog uuname      (csh: foreach i (uucp ...))
do
    cp $i $i.old
done                                (csh: end)
cd /usr/lib/uucp
for i in uucico uuxqt uuclean
do
    cp $i $i.old
done
```

2. Editing Makefile and uucp.h

4.2 sites using the supported dialers are all set with defaults.

non-4.2

- a) sites need to install the Berkeley directory reading library.

Try (cd libndir; make install).

Edit Makefile to have LIBNDIR= -lndir

define NDIR in uucp.h.

- b) Check LDFLAGS, OWNER, GROUP, and LIBUUCICO.

- c) pick a method to allow uucp to know its system:

Check out GETHOSTNAME/UNAME/WHOAMI/CCWHOAMI in uucp.h

- d) define SYSIII if appropriate in uucp.h.

- e) Your "make" may fail because the Makefile is so large.

If so, in /usr/src/cmds/make/defs, change

'#define NFLEFTS 60' to 512, and re-make make.

3. Make the new commands.

make

4. WAIT UNTIL THE UUCP SYSTEM IS IDLE!! Single-user is best.

su root(it is important that chmod, chown and chgrp work below)

5. Install the new commands:

make install

(If you are chicken, type 'make -n install' first).

6. Edit and install the control files:

Look in UUAIDS. Edit and install into /usr/lib/uucp if necessary USERFILE, L.cmds, L.sys, L.devices, L.dialcodes. THESE FILES MUST BE OWNED BY THE SAME OWNER AND GROUP AS THE UUCP COMMANDS AND UUCP SPOOL FILES!!! (probably uucp, daemon). The format for dialers is slightly different so that any dialer can be handled.

### 7. Make new subdirectories:

For safety: `cd /usr/spool/uucp; tar c . (save Qed files on tape)`

The following assumes your site name is produced by ``uname -l``.

```
make mkdirs
```

If your system is duke, then the subdirectories created are named C., D., D.duke, D.dukeX, TM., XTMP and X..

```
rmdir /usr/lib/uucp/.XQTDIR
```

which is obsolete (XTMP replaces it).

### 8. Move old Qed files:

If you have spooled files, they must be moved into the subdirectories. Assuming all spool files are in `/usr/spool/uucp` (i.e. you did not have subdirs before) the following command will move the spool files to the right subdirectories:

```
make mvspoolfiles; ??? does this work now??????????
```

Files beginning C. are put in the C. subdirectory, and so on. Files beginning D.dukeX are put in that directory, \*not\* D.. (Note: if you already had a subdirectory version of uucp, you need only create the new subdirectories mentioned above and move the relevant files there.

Delete other old directories if you had any (e.g. "LOG.")

### 9. Compact `/usr/spool/uucp`:

```
cd /usr/spool
```

```
mkdir nuucp
```

```
chown uucp nuucp
```

```
chgrp daemon nuucp
```

```
for i in uucp/*          (csh: foreach i (uucp/*))
```

```
do
```

```
    mv $i nuucp
```

```
done(csh: end)
```

```
rmdir uucp
```

```
mv nuucp uucp
```

(Note: this does \*not\* work if a filesystem is mounted on `/usr/spool/uucp`! If that is the case, you should `'tar /usr/spool/uucp` somewhere, unmount, `re-mkfs`, and `re-mount` the filesystem, and `tar` the uucp files back.)

### 10. Test the new system

Test by mailing a letter somewhere and back. If it works, the new system is probably fine. Otherwise, figure out what is wrong. Start by examining LOGFILE. Try `/usr/lib/uucp/uucico -r1 -sname -x7` If things are no-go, you can back out the changes by restoring the old uu programs and the spooled files. NOTE: The subdirectories foul up a non-subdir version, so be sure that subdirectories exist/do not exist as appropriate.

### 11. Install handy UUAIDS

Look in uuaids for handy other stuff. In particular, uu.\* are shell scripts that can be run via cron hourly, daily, and weekly to keep uucp trim. L-devices, L-dialcodes, L.cmds, L.sys, USERFILE are sample files. THEY MOST LIKELY NEED TO BE MODIFIED TO USE THEM ON YOUR OWN SYSTEM!

uu.hourly            Makes sure sites are polled.  
uu.daily             Runs uuclean to clean up /usr/spool/uucp.  
                      \*\*IT HANDLES SUBDIRECTORIES!!  
                      Renames LOGFILE to LOGFILE.old.  
uu.weekly            Renames SYSLOG to SYSLOG.old.

The above scripts should be run via cron.

uucpsummary

Summarizes LOGFILE and SYSLOG. It's very helpful in telling what is going on and who is calling who.

uucp.daily

Daily shell script run at ittvax. Manages SYSLOG files in a convenient form for uusage. Not integrated in time for this distribution.

Comments, bug reports, \*and improved code\* are welcome.

Tom Truscott

Bob Gray

付録 G

JUNET内のサイトマップ

1. JUNET接続サイトマップ
2. JUNETに接続されているドメイン・リスト

## 1. JUNET接続サイトマップ

### 表記法

1. (組織名)

2. サイト名の右から出ている接続  
同一ドメイン(組織)内の接続を表す  
site1--+-site2  
+-site3

3. サイト名の中程から出ている接続  
異なるドメイン(組織)間の接続を表す  
+---site2  
|  
site1  
|  
+---site3

4. #####での接続

```
site1
#
#---site2
#
#---site3
#
#---site4
|
+---site5
```

というのは、

```
site1
|||||
|||+---site2
|||
||+---site3
||
|+---site4
|
+-----site5
```

という意味。

5. 大文字のサイト名。

--SITE1

以前の調査では反応があったのに、今回は、返答が無かったサイト。

全133サイト

```
#--dclsic
#
#--secisl
#--seikei
#
#--jnglab
#--astd
# |
# +-kiwi(juice)--seaside--kwmux----jmis
#
#--assun      (SIGMA)
#
#--dlabvax    (上智大)
#
#--astecv     (アステック)
#
#      +-sravos
#      +-sravb
#      +-sravd
#      +-SRAVC
#--srava--SRAGW (SRA)
# #
# #---nccva--CEC8000
# #      +-nccs1
# #      +-NCCS2
# #--sonyvd--sonyve---isrcne--isrcng
# #      +iwai1
# #      +kos2
# #--hstvax
# #--jsdvax---NTSVAX
#
#      +--fxvax  (FUJIXEROX)
#      |
#--kossvax--k3sun3 (慶応大)
#      |      +--koccvax
#      |      +--koscvox
#      |      +--KOEAVAX
#      |
#      +-PATTY--charlie (リコー)
#      +-ELTPDP--ELTFUJI--HODAKA
#      +-RSKVAX
#
#
```





```

#---icot
# #
# #----casun1---cau1          (キャノン)
# #----jepri(DENKEN)
# #----okis1(OKI)
# |
# |
# +-ETLVAX
# +-etlcom          (電総研)
# | +-----+
#-gama              |          (筑波大学)
# +-ulisvax-ulisdpd |          (図書館情報大)
# |
#---tutics
#---hubs            |          (北大)
# | +-----+
# |
nttlab+---nttsp1+---nttspa      (NTT)
# |          +---nttspb
# |
# +-nttdpe+---nttdpf---NTTOSE
# |          +-ntttsd
# |          +-nttbcs+---nttvdt--nttcoin
# |          |          +-nttkb
# |          +---NTTCVG
#
#--yufuji          (山梨大)
#
# +-hagiwara----yajima
# +-doshita
# +-kuee---kueif1
# +-kurims----->[iegva1]
#-kuis+---kutsuda      (京大)
# | +-octpat
# | +-sakains
# |
# +---heway(kaba)
#
#
#--ouicsu+---ics750      (阪大)
# # +-icssm----icsts1--creamy--mercury--venus
# # +-fuji3
# #
# #-iegva1+---wrlvx      (松下)
# # +--->[kurims]
# #
# #--shpnar+---SHPVAX      (シャープ)
# # +---shpcs1
# #--abc          (朝日放送)

```

```
#    +--kusys4--kbrs4b (神戸大)
#
#--suvax1---SUVAX0      (静岡大)
#    |
#    +-rdvsx1-rdnew2    (ローランド)
#
#--kyu-cs----+--kyu-cc  (九州大)
#                +--csce
#                +--SOFTW
#                +--qecuts
#
#-tsuda
#
#--flats(riken)
#    |
#    +decjrd
```

2. JUNETに接続されているドメイン・リスト(アルファベット順)

|    |           |                                                      |
|----|-----------|------------------------------------------------------|
| 1  | air       | AI Language Research Institute, Ltd.                 |
| 2  | anritsu   | Anritsu Corp.                                        |
| 3  | aoyama    | Aoyama Gakuin University                             |
| 4  | asahi     | Asahi Broadcasting                                   |
| 5  | asahi-np  | The Asahi Shimbun Publishing Company                 |
| 6  | ascii     | ASCII Corporation                                    |
| 7  | asp       | ASP Corp.                                            |
| 8  | asr       | ASR Corporation                                      |
| 9  | astd      | AST Souken                                           |
| 10 | astec     | ASTEC, Inc.                                          |
| 11 | astem     | Astem Research Institute                             |
| 12 | atr       | Advanced Telecomm. Research Inst. International      |
| 13 | att-j     | AT&T Unix Pacific Co., Ltd.                          |
| 14 | brother   | Brother Insudtry Co. Ltd.                            |
| 15 | cac       | Computer Applications Co. Ltd.                       |
| 16 | canon     | Canon Inc.                                           |
| 17 | canopus   | Canopus Ele. Co. Ltd.                                |
| 18 | casio     | Casio                                                |
| 19 | cec       | Chuo Elec. Co.,Ltd.                                  |
| 20 | chukyo-u  | Chukhyo University                                   |
| 21 | chuo-u    | Chuo University                                      |
| 22 | ciea      | Central Inst. of Exp. Animal Pre-Clinical Lab.       |
| 23 | citoh     | C.Itoh & Co., Ltd.                                   |
| 24 | crl       | Communications Research Laboratory                   |
| 25 | csk       | CSK Corporation                                      |
| 26 | dcl       | Digital Computer Limited                             |
| 27 | dec-j     | Nihon Digital Equipment Corporation                  |
| 28 | denken    | Central Research Inst. of Electric Power Industry    |
| 29 | dit       | dit Co., Ltd.                                        |
| 30 | dnp       | Dai Nippon Printing Co., Ltd.                        |
| 31 | edr       | Japan Electronic Dictionary Research Institute, Ltd. |
| 32 | eric      | Eric                                                 |
| 33 | etl       | Electrotechnical Laboratory                          |
| 34 | fit       | Fukuoka Institute of Technology                      |
| 35 | foretune  | Foretune Co., Ltd                                    |
| 36 | fujitsu   | Fujitsu Ltd.                                         |
| 37 | fujixerox | Fuji Xerox CO., Ltd.                                 |
| 38 | fukui-u   | Fukui University                                     |
| 39 | fukuoka-u | Fukuoka University                                   |
| 40 | gctech    | Graphics Communication Technologies R&D Dept.        |
| 41 | global    | Global Softwares                                     |
| 42 | gsj       | Geological Survey of Japan                           |
| 43 | gunma-u   | Gunma University                                     |
| 44 | hankyu    | Hankyu Dentetsu Corp.                                |
| 45 | hitachi   | Advanced Research Laboratory, Hitachi, Ltd.          |

|    |           |                                                  |
|----|-----------|--------------------------------------------------|
| 46 | hokudai   | Hokkaido University                              |
| 47 | horiba    | Horiba Seisaku                                   |
| 48 | hst       | Hirata Software Technology Co., Ltd.             |
| 49 | ibmtrl    | Tokyo Research Laboratory, IBM Japan, Ltd.       |
| 50 | icm       | ICM Co. Ltd.                                     |
| 51 | icot      | Institute for New Generation Computer Technology |
| 52 | ihi       | Ishikawajima-Harima Heavy Industries Co.,Ltd.    |
| 53 | ijaic     | ICOT-JIPDEC AI. Center                           |
| 54 | intec     | Intec Corp.                                      |
| 55 | ipa       | Info. Proc. Assoc.                               |
| 56 | ipri      | Industrial Products Research Institute           |
| 57 | isac      | International Sekisu AI Corporation              |
| 58 | ism       | Toukei Suuri Kenkyuujo                           |
| 59 | jip       | Japan Information Processing Service Co.,Ltd.    |
| 60 | jsd       | Joint System Development Corporation             |
| 61 | jus       | Japan Unix Society                               |
| 62 | jusoft    | Nippon Unisoft Corporation                       |
| 63 | k3        | Kanri Kougaku Kenkyuujo                          |
| 64 | kaba      | Kyoto Artificial Brain Associates                |
| 65 | kajima    | Kajima Corp.                                     |
| 66 | kansai-u  | Kansai University                                |
| 67 | kao       | Kao                                              |
| 68 | kcs       | Koube Computer Service Co., Ltd.                 |
| 69 | kddlabs   | KDD Lab.                                         |
| 70 | keio      | Keio University                                  |
| 71 | kiic      | Kyoto Industry Information Center                |
| 72 | kit       | Kyoto Institute of Technology                    |
| 73 | kk        | Kozo Keikaku Engineering Inc.                    |
| 74 | kobe-u    | Kobe University                                  |
| 75 | konan-u   | Konan Univ.                                      |
| 76 | kpress    | Kabushiki Shinbun                                |
| 77 | kubota    | KUBOTA Ltd.                                      |
| 78 | kurume-it | Kurume Institute of Technology                   |
| 79 | kyocera   | Kyocera Tokyo Central Research Laboratory        |
| 80 | kyoto-su  | Kyoto Sangyou University                         |
| 81 | kyoto-u   | Kyoto University                                 |
| 82 | kyushu-id | Kyushu Institute of Design                       |
| 83 | kyushu-u  | Kyushu University                                |
| 84 | kyutech   | Kyushu Institute of Technology                   |
| 85 | m-kasei   | Mitsubishi Kasei                                 |
| 86 | mei       | Matsushita Electric Industrial Co., Ltd.         |
| 87 | meiji-u   | Meiji University                                 |
| 88 | mel       | Mechanical Engineering Laboratory                |
| 89 | melco     | Mitsubishi Electric Corporation                  |
| 90 | minpaku   | National Museum of Ethnology                     |
| 91 | mita      | Mita Industrial Co., Ltd.                        |
| 92 | mot-j     | Motorola Japan                                   |
| 93 | mri       | Mitsubishi Research Institute, Inc.              |
| 94 | msr       | Mitsuizosen Systems Research                     |

|     |                 |                                                |
|-----|-----------------|------------------------------------------------|
| 95  | n-denso         | Nihon Denso Corp.                              |
| 96  | nacsis          | National Center for Science Information System |
| 97  | nagano          | Nagano University                              |
| 98  | nagaoka         | Nagaoka University of Technology               |
| 99  | nagaoka-coltech | Nagaoka College of Technology                  |
| 100 | nagoya-u        | Nagoya University                              |
| 101 | nara-k          | Nara National College of Technology            |
| 102 | ncc             | NRI & NCC Co. Ltd.                             |
| 103 | ndg             | Nihon Data General                             |
| 104 | nec             | NEC Corporation                                |
| 105 | nig             | National Institute of Genetics                 |
| 106 | nikon           | Nikon Corp.                                    |
| 107 | nitech          | Nagoya Institute of Technology                 |
| 108 | nsc             | Nippon Steel Corporation                       |
| 109 | nts             | Nippon Timeshare Co., Ltd.                     |
| 110 | ntt             | Nippon Telegraph and Telephone Corporation     |
| 111 | oita-u          | Oita University                                |
| 112 | oki             | Ok Electric Industry Co.,Ltd.                  |
| 113 | omron           | OMRON Tateisi Electronics Co.                  |
| 114 | osaka-cu        | Osaka City University                          |
| 115 | osaka-u         | Osaka University                               |
| 116 | osakac          | Osaka Elec. & Comm. Univ.                      |
| 117 | peccom          | Sekiyu Sangyou Kasseika Center                 |
| 118 | pentel          | Pentel                                         |
| 119 | qmflp           | GOTO Quantum Magneto Flux Logic Proj.          |
| 120 | recruit         | Institute for Supercomputing Research, Recruit |
| 121 | ricoh           | RICOH Company, Ltd.                            |
| 122 | rikei           | Rikei Corp.                                    |
| 123 | riken           | Institute of Physical and Chemical Research    |
| 124 | roland          | Roland Corporation                             |
| 125 | rtri            | Railway Tech. Research Inst.                   |
| 126 | sanno           | SANNO College                                  |
| 127 | sanwa-ss        | Sanwa Seisakusho                               |
| 128 | sanyo           | SANYO Electric Co., Ltd.                       |
| 129 | sasebo          | Sasebo College of Technology                   |
| 130 | seclab          | SECOM Intelligent Systems Laboratory           |
| 131 | seikei          | Seikei University                              |
| 132 | sharp           | Sharp Corporation                              |
| 133 | sheart          | University of the Sacred Heart                 |
| 134 | shinshu-u       | Shinshu University                             |
| 135 | shizuoka        | Shizuoka University                            |
| 136 | sibasoku        | shibasoku Corp.                                |
| 137 | sigma           | Information-technology Promotion Agency, Japan |
| 138 | softbank        | Softbank Research Institute                    |
| 139 | sony            | Sony Corporation                               |
| 140 | sonytek         | Sony Tektro.                                   |
| 141 | sophia          | Sophia University                              |
| 142 | soum            | SOUN Corporation                               |
| 143 | sra             | Software Research Associates, Inc.             |

|     |            |                                                     |
|-----|------------|-----------------------------------------------------|
| 144 | stanley    | STANLEY ELECTGRIC Co., LTD.                         |
| 145 | sumiden    | Sumitomo Electric Industries, LTD.                  |
| 146 | sumikin    | Sumitomo Metal Industries                           |
| 147 | sun-j      | Sun Micro Inc. Japan                                |
| 148 | symbolics  | Nihon Symbolics Corporation                         |
| 149 | t-gitan    | Toyama Prefectural College of Technology            |
| 150 | tcomp      | Tokyo College of Medico Pharmaco                    |
| 151 | telema     | Telematique International Research Laboratories     |
| 152 | tis        | Toyo Information System                             |
| 153 | titech     | Tokyo Institute of Technology                       |
| 154 | tohoku     | Tohoku University                                   |
| 155 | tohoku-u   | Tohoku Univ. (Computer Center)                      |
| 156 | tokuyama   | Tokuyama National Technical College                 |
| 157 | toshiba    | Toshiba Corp.                                       |
| 158 | toyo       | Toyo University                                     |
| 159 | toyota-ti  | Toyota Technological Institute                      |
| 160 | tsuda      | Tsuda College                                       |
| 161 | tsukuba    | Tsukuba Univ.                                       |
| 162 | tuat       | Tokyo University of Agriculture and Technology      |
| 163 | tut        | Toyohashi University of Technology                  |
| 164 | tytlabs    | Toyota Central Research & Development Lab., Inc.    |
| 165 | u-tokyo    | Univ. of Tokyo                                      |
| 166 | uclosk     | Universal Computer Laboratory Corp.                 |
| 167 | uec        | University of Electro-Communications                |
| 168 | ulis       | University of Library and Information Science       |
| 169 | unisys     | Nihon UNISYS, LTD.                                  |
| 170 | waseda     | Waseda University                                   |
| 171 | yamagata-u | Yamagata University                                 |
| 172 | yamaha     | Yamaha Corporation                                  |
| 173 | yamanashi  | Yamanashi University                                |
| 174 | yaskawa    | Yaskawa Electric Mfg. Co., Ltd.                     |
| 175 | yhp        | Yokogawa Hewlett Packard Computer Operation         |
| 176 | ysc        | Yokohama Science Popularizing Association for Youth |