

**Smail**

**COLLABORATORS**

	<i>TITLE :</i> Smail		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		January 7, 2023	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1 Smail</b>	<b>1</b>
1.1 Smail	1
1.2 Features	1
1.3 Introduction	2
1.4 Installation	3
1.5 Different Smail programs	4
1.6 Setup a path file	4
1.7 Don't use paths-files	5
1.8 BSMTP setup	5
1.9 Usage	6
1.10 Address	8
1.11 Addresses accepted	9
1.12 Routing	9
1.13 Re-routing	9
1.14 Fromming	10
1.15 Miscellaneous	11
1.16 Headers	11
1.17 Undeliverable mail	12
1.18 Return-Receipt-To headers	12
1.19 From and Realname	12
1.20 Config options	13
1.21 Command line parsing	14
1.22 Smarthost	14
1.23 Logfiles	15
1.24 RMail mode	15
1.25 Check user	15
1.26 Local Mailer	16
1.27 BSMTP	16
1.28 Aliases	17
1.29 Alias format	17

---

---

1.30 Mailing list . . . . .	19
1.31 Include other alias files . . . . .	19
1.32 Mail forwarding . . . . .	19
1.33 Command piping . . . . .	19
1.34 Thanks . . . . .	20
1.35 Version and Author . . . . .	20
1.36 Wish list . . . . .	20
1.37 Copyright . . . . .	20
1.38 Source files . . . . .	21
1.39 History . . . . .	21

---

# Chapter 1

## Smail

### 1.1 Smail

Smail, rmail - a UUCP mailer

Features

Introduction

Installation

Usage

Address

Miscellaneous

Aliases

Thanks

Version and Author

Wish list

Copyright

Source files

History

### 1.2 Features

- o complete substitution for Dillons sendmail
  - o using paths file for UUCP routing
-

- o re-routing of addresses according to the paths file
- o mail forwarding for users (~/.forward and uumail:<user>)
- o compatible parameters with Dilllon's sendmail (except -raw)
- o Return-To-Receipt recognition
- o returning undeliverable mails to sender and postmaster of your host
- o support of the standard Getty, MultiUserFileSystem and AXSh passwd files
- o smarthost option if your map files are not perfect :-)
- o easy setup of a mailing list via alias include files
- o extended log files
- o accepts the following addresses
  - bang paths (e.g host1!host2!user)
  - @ addresses (e.g user@host1)
  - user%site%site.domain@domain
  - domain!site.domain%site%user
- o BSMTMP (Batch Simple Mail Transport Protocol)
- o call uux (instead of writing directly in uuspool)
- o Checking local users permission to send mail via UUCP/BSMTMP
- o source files available
- o docs in AmigaGuide format
- o Installer script
- o only runs with OS2.0 or higher

## 1.3 Introduction

The Smail/rmail program replaces Dilllons sendmail to become the UUCP mail transport mechanism. rmail receives mail from UUCP, Smail introduces mail into UUCP.

Smail/rmail can work with or without sendmail, or another intelligent mail system. For hosts with just mail, Smail/rmail subsumes some of the functions of sendmail, and hands only local mail to the Local Mailer

.

To varying degrees, Smail/rmail automatically routes the addresses it processes. Smail/rmail most often routes domain style addresses (i.e. user@domain), producing a UUCP path (i.e. host!address) or a local address (i.e. user) but it can also re-route explicit UUCP paths.

## 1.4 Installation

Please use the Installer script !!!

If this doesn't work do the following things:

1.) Copy

```
Smail      -> uucp:c/sendmail
Smail      -> uucp:c/rmail
mail       -> uucp:c/mail
```

or for the AXSh Version

```
Smail_axsh -> uucp:c/sendmail
Smail_axsh -> uucp:c/rmail
mail_axsh  -> uucp:c/mail
```

or for the MUFs Version

```
Smail_mufs -> uucp:c/sendmail
Smail_mufs -> uucp:c/rmail
mail_mufs  -> uucp:c/mail
```

2) Edit the

Config options  
of Smail in "uulib:config."

3) Make a directory "Mail" in "uulib:"

4) If you want to use

Aliases

edit "uulib:mail/aliases". If you have already an alias file from Dillons sendmail, just copy it to "uulib:mail/aliases" or edit the SMALias in "uulib:config".

5) If you are using a different NAME for sendmail/rmail than "sendmail"/"rmail" (e.g Smail or Smail -r) you have to insert a Sendmail/Rmail Entry in "uulib:config". This is necessary if Smail cannot deliver a mail or Smail found a "Return-To-Receipt Header". In either case Smail calls the sendmail program to deliver the mails.

6) Setup a path file (if you want to use this option)

---

- ```
(see
  Setup a path file
)
```
- 7) Setup BSMTMP stuff (if you want to use this option)  
 (see  
   BSMTMP setup  
 )

Different Smail programs

Setup a path file

Don't use paths-files

BSMTMP setup

## 1.5 Different Smail programs

- o Getty
 

List all your site's users in `getty:passwd`. The real name must be entered in the comment field. You only need this if you don't use the `ENV:` variable called "REALNAME". Besides Smail takes the home-directory for Mail forwarding from the `getty's passwd`.
- o AXSh
 

These programs use the AXSh `passwd` instead of Getty's `passwd`.
- o MUFSS
 

These programs use the MUFSS `passwd` instead of Getty's `passwd`.

## 1.6 Setup a path file

Convert your paths file (generated by `pathalias -f`) to ↔  
 Smail  
 format using "convert" and "(q)sort". For the CShell the  
 command line is:

```
pathalias -f <maps |convert|qsort >uulib:mail/paths
```

For receivers that are not in your maps data (nobody is perfect) setup SM  
 Smarthost  
 in "uulib:config". This host  
 receives all mails for receivers that are not listed in  
 "uulib:mail/paths".

Attention!!!

Use the following syntax for your maps:

```
----- snip -----
host.with.domain=host
yourhost host (DAILY+FAST)
----- snap -----
```

NOT

```
----- snip -----
host=host.with.domain
yourhost host.with.domain (DAILY+FAST)
----- snap -----
```

If you don't have real maps use demo\_maps/demo\_paths.  
You must change

```
mavhh           -> your_site_without_domain
mavhh.hanse.de -> your_site_with_domain
torfhh          -> 1.site_you_call_without_domain
torfhh.hanse.de -> 1.site_you_call_with_domain
wolfhh          -> 2.site_you_call_without_domain
wolfhh.hanse.de -> 2.site_you_call_with_domain
```

Attention!!!

Don't forget to sort the path file.

## 1.7 Don't use paths-files

```
                Insert SM
Smarthost
    your_smart_host in your "uulib:config"
file. The smarthost is the host receiving all your mail.
```

## 1.8 BSMTTP setup

```
                Copy
Smail           -> uucp:c/rsmtmp
Smail           -> uucp:c/rcsmtp
Smail           -> uucp:c/rfsmtmp
batch_smtp      -> uucp:c/batch_smtp
```

Setup the SMCompress, SMBsmtpCmd, SMBsmtpSites, Compress and Freeze Keywords in "uulib:config".

Add every host that should receive mails via BSMTTP to "uulib:mail/bsmtpp\_sites".

Read the  
BSMTP  
section.

## 1.9 Usage

```
Smail [ Options ] address ...  
rmail [ Options ] address ...
```

-A Print the resolved addresses. DON'T collect a message or  
invoke a mailer (e.g. for address resolving in scripts)

-d Be verbose and DON'T invoke other mailers.

-v Be verbose but still invoke other mailers.

-h hostname

Set hostname. You may alternatively set this via the

```
Config options  
in "uulib:config", too.
```

-H hostdomain

Set hostdomain. You may alternatively set this via the

```
Config options  
in "uulib:config", too.
```

-f address

Use address on the "From:" line in locally generated  
mail.

-p pathfile

Set path database file name if not "uulib:mail/paths"

You may even set this via the

```
Config options  
in  
"uulib:config".
```

-a aliasfile

Set alias file name if not "uulib:mail/aliases"

You may even set this via the

```
Config options  
in  
"uulib:config".
```

-n namelist

Smail supports another type of aliasing intended for  
full name resolution using a sorted file, namelist or  
name/address pairs. This allows mail to  
George.P.Burdell@gatech.edu to be delivered  
appropriately. These aliases are by nature very simple  
since they are not composed of long lists of recipients

for each alias. They are also numerous since mail to George.P.Burdell may be addressed to Burdell, G.Burdell, George.Burdell, P.Burdell, G.P.Burdell, or George.P.Burdell. This simpler form of aliasing uses the same fast searching algorithm that is used for the paths file, so it keeps resolution time manageable. You may even set this via the Config options in "uulib:config", too.

#### Attention!!!

On AmigaUUCP you normally don't find a uux with queueing. So the next three options (-q -m -u) are mostly useless.

#### -q number

Take number as the queueing threshold. When routing a mail ( -o, -O, or domain addressed mail ) to a given host and the costs listed in the paths file are less than the queueing threshold the mail will be sent immediately. This overrides the default threshold of 300.

#### -m number

Most number jobs will be handed to uux for immediate delivery by a single invocation of Smail (default 2)

#### -u uuxflags

Use uuxflags as the flags passed to uux for remote mail. This overrides any of the default values and other queueing strategies.

-M If Smail fails to deliver the mail don't send this back to the sender.

-C Consult the paths file for the cost of the path even when not routing the mail. This makes it possible to use the cost information when sending pure UUCP path mail without re-routing it.

-o Route the first component of a UUCP path (host!address) in addition to routing domain addresses (user@domain) (see

Routing  
) Deactivate re-routing !

-O Re-route UUCP paths, trying successively larger righthand substrings of a path until a component is recognized (see

Routing  
) . Activate re-routing !

-r Running Smail in rmail mode. In this mode Smail/rmail does not search for for addresses (to send the mail to) in the mail file, but ONLY takes the addresses from command line (see

RMail mode  
) .

---

- l Instead of routing a domain address send it to the local mailer for processing. Normally, only local addresses go to the local mailer.
  
- L Send all addresses to the local mailer for processing, including UUCP paths. This options is used, if you want Smail ONLY to resolve the address and NOT to deliver the mail.
  
- B Send the mail to EVERY receiver via BSMTTP (Batch Simple Mail Transport Protocol) ! If you want to send only to some receivers/hosts via BSMTTP use the "uulib:mail/bsmtp\_sites" file (see BSMTTP ).
  
- c address  
Use address on the "Cc:" line in locally generated mail.
  
- b address  
Use address on the "Bcc:" line in locally generated mail.
  
- t address  
Use address on the "To:" line in locally generated mail.
  
- F file  
Use file as input for the locally mail.  
Attention!!!  
Your filename has to start with t:. Otherwise use <file.
  
- s text  
Use text on the "Subject:" line in locally generated mail.
  
- R text  
Use text as real name in locally generated mail.

The -l flag causes rmail to pass all domain addresses through to the local mailer to process addresses for non UUCP domains. The -L flag causes rmail to pass even explicit UUCP paths through to the local mailer, presumably to make use of other transport mechanisms. In both cases rmail defers any routing until Smail gets hold of it.

## 1.10 Address

Addresses accepted

Routing

Re-routing

Fromming

## 1.11 Addresses accepted

Smail/rmail understands "user%site%site.domain@domain" and "user@domain" to be a domain address, "host!address" and "domain!site.domain%site%user" to be a UUCP path and anything else to be a local address.

Because hostile rmail's unpredictably interpret mixed UUCP/domain addresses, Smail/rmail understands "domain!user" to be a domain address and generates "path!domain!user" when mailing to a cognate Smail/rmail host. To distinguish domain "domain!user" from UUCP "host!address" "domain" contains at least one (1) period. Smail/rmail gives precedence to @ over ! when parsing mixed addresses, thus a!b@c is parsed as (a!b)@c, rather than a!(b@c).

## 1.12 Routing

Because Smail/rmail is the UUCP transport mechanism it can only effect delivery on UUCP paths and local addresses; domain addresses require resolution into UUCP paths or local addresses. To resolve a domain address Smail/rmail finds a route to the most specific part of the domain specification listed in the routing table. Two degrees of resolution can occur:

Full resolution: Smail/rmail finds a route for the entire domain specification, and tacks the user specification onto the end of the UUCP path. The address can also fully resolved into a local address (the UUCP path is null).

Partial resolution: Smail/rmail finds a route for only righthand part of the domain specification so it tacks the complete address (in the form domain!user) onto the end of the UUCP path. Since this syntax is not widely understood UUCP gateways listed in the path database must install new UUCP software either Smail/rmail or new sendmail configuration files (or both).

If partially resolved address routes to the local host (means having a null UUCP path) it's treated as an error. According to the routing table the local host takes the responsibility for parsing the address correctly.

## 1.13 Re-routing

The -o flag makes Smail/rmail attempt to route the first component of a UUCP path, probably to impress people with how many UUCP hosts it knows. If this fails it sends the unrouted address to the

Smarthost

The `-O` flag makes Smail/rmail take a UUCP path and route the rightmost component of it (saving the user name). This is implemented for hosts having very up-to-date routing tables.

If a route cannot be discerned from the available routing database an additional attempt to route the mail is made by searching for an entry in the database containing the route to a

Smarthost

. If this entry exists the mail will be forwarded using that route. This allows a host to depend on another (presumably better informed) host to deliver its mail. This kind of arrangement should be worked out - in advance - with the smart host's administrator.

After Smail/rmail resolves an address it reparses it to see if it is now a UUCP path or local address. If the new address turns out to be another domain address Smail complains as it doesn't like to resolve more than once. This error occurs when an address partially resolves the local host.

By default Smail won't alter the explicit bang path routing of any mail message. If the stated path is unuseable, (i.e. the next hop host is unknown) then Smail will ALWAYS apply routing and attempt to deliver the mail to the potential new address. If either this fails REROUTE routing will be applied to the address and another attempt to deliver is made. Finally an attempt to find a path to a better informed host (see

Smarthost

) will be made and the mail is passed to that host. If there is no path file Smail will call uux directly with the smarthost as host. So you must have an entry for your smarthost in "uulib:l.sys".

## 1.14 Fromming

Smail/rmail improves "From\_" and ">From\_" lines to a simple from argument, which it can pass to sendmail or use to create its own "From" line. The rule for fromming is: concatenate each "remote from" host (separating them by "!'s), and tack on the address on the last "From\_" line; if that address is in user@domain format rewrite it as domain!user; ignore host or domain if either is simply the local hostname. It also removes redundant information from the "From\_" line. For instance:

```
...!myhost!myhost.mydomain!...
```

becomes

...!myhost!...

Leading occurrences of the local host name are elided as well.

Smail/rmail generates its own From\_ line, unless it is feeding sendmail, which is happy with the -ffrom argument. For UUCP bound mail, Smail/rmail generates a "remote from hostname", where hostname is the UUCP hostname (not the domain name), so that From\_ can indicate a valid UUCP path, leaving the sender's domain address in "From:".

## 1.15 Miscellaneous

Headers

Undeliverable mail

Return-Receipt-To headers

From and Realname

Config options

Command line parsing

Smarthost

Logfiles

RMail mode

Check user

Local Mailer

BSMTP

## 1.16 Headers

Certain headers, "To:", "From:", "Date:", etc., are ←  
required

by RFC822. If these headers are absent in locally generated mail, they will be inserted by Smail. Also, a line of trace information, called a "Received:" line, will be inserted at the top of each message. In addition Smail supports "Cc:", "Bcc:" and

Return-Receipt-To headers

..

---

## 1.17 Undeliverable mail

Surely nobody likes his mail not reach its intended destination but it sometimes happens that way. Mail that is found to be undeliverable (i.e., unknown user or unknown host) will be returned to the sender and to postmaster of the local host.

## 1.18 Return-Receipt-To headers

Smail/rmail will send a mail with a copy of the header ↔ of the original mail to the address after the "Return-To-Receipt:" Header if the mail is for a user on your site. You may turn this off via the SMReceipt keyword in the "uulib:config" file. See Local Mailer for the recognition whether a user exists on your site.

Attention!!!

Only one Return-Receipt-To: header line and address per mails is allowed ! This will be fixed soon.

Attention!!!

If the Smail/rmail delivers a mail to two or more recipients with one call of your local mailer and one fails, no return mail is send. e.g:

```
mail testmail user0 user1 user2
```

Only the delivery to user1 fails => No receipt mail is generated, because the "mail" command fails.

## 1.19 From and Realname

If you send a mail, Smail/rmail looks whether there is

- a) a -f argument and a -R argument  
=> take the fromname from -f and the realname from -R
- b) -f argument and no -R argument.  
=> take the fromname from -f, try to find the fromname in getty:passwd/axsh:etc/passwd and take the comment as realname. If not found look for Realname in "uulib:config".
- c) no -f argument and no -R argument  
=> o Getty/AXSh version  
look at the local variable USER and REALNAME. If no REALNAME environment variable exists, try to find USER in getty:passwd or axsh:etc/passwd and take the

- comment as realname. If not found look for RealName in "uulib:config". If no USER is found look for UserName in "uulib:config".
- o MUFS version
  - Determinate via MUFS who is the owner of this task. If this is not possible, do the same as the Getty/AXSh version.

## 1.20 Config options

You can configure the following options in "uulib:config ←  
"

```
# react to Return-To-Receipt
SMReceipt          Yes
# DUUCP-styled logfile
SMLogfile          uuspool:Smail.log
# wCNews styled logfile
SMMaillogfile     uulib:Mail/logfile
# evaluate the cost of a mail
SMGetcost          Yes
# Record-file (could be nil: # for nologging)
SMSentlog          t:Smail.sent
# ascii passwd
SMPasswd           getty:passwd
# ascii paths file (converted and sorted!)
SMPath            uulib:Mail/paths
# ascii aliases file
SMAlias           uulib:Mail/aliases
# ascii fullnames file
SMFullname        uulib:Mail/fullname
# name of the smarthost
SMSmarthost       wolfhh
SMRerouting        1
# 0 - route C, B, or A (whichever works); set by -O
#   this means Smail do re-routing
# 1 - route A always; set by -o
#   this means Smail do no re-routing
# 2 - route A if A is a domain

SMHandle           0
# 0 - handle UUCP and DOMAIN addresses
#   this means Smail does everything for you
# 1 - handle UUCP only; set by -l
#   this means Smail work only for UUCP addresses
# 2 - all mail is LOCAL; set by -L
#   this means Smail gives all work to the local mailer
#   this is useful for using sendmail for the real mail
#   transport

SMCompress         compress
# the command to compress your BSMTP mails
# the command is execute (see
    BSMTP
    )
```

```
# cmd <infile >outfile

SMBsmtpCmd          rcsntp
# the command on the host, that receive your BSMTTP
# mails, to deliver the BSMTTP mails (see
    BSMTTP
    )

SMBsmtpSites        uulib:mail/bsmtp_sites
# file in which all BSMTTP sites are written down
# (see
    BSMTTP
    )

Compress             sys:c/compress
Freeze               sys:c/freeze
# complete path for compress and freeze

SMCheckuser         No
# check sender permission to send mails via UUCP/BSMTTP

SMCheckFile         uulib:mail/checkuser
# file in which all users are written down
```

## 1.21 Command line parsing

If you submit a string with spaces for an option (e.g for a subject), you have to enclose them into double quotes because Smail/rmail use the normal argv[] parsing routines.

Example:

Wrong:

```
Smail <mail -s this is a nice subject ...
```

Right:

```
Smail <mail -s "this is a nice subject" ...
```

If you want to enclose double quotes themselves in your subject you have to stuff the double quotes.

in the CShell:

```
Smail <mail -s "this is a nice *\"subject*\"" ...
```

in C= shell:

```
Smail <mail -s "this is a nice *\"subject*\"" ...
```

## 1.22 Smarthost

If Smail/rmail can't resolve a path to the receiver of a mail, Smail/rmail will try to sent this mail to a smarthost. You must label the name of the smarthost in "uulib:config" with the keyword SMSmarthost. Normally Smail/rmail tries to resolve the path to the smarthost. If this is not

---

possible (e.g you don't have a paths file) Smail/rmail will call uux directly.

## 1.23 Logfiles

SMSentlog will backup the complete mail and a little header with the sender and the maillength

SMLogfile will write logmessages in the DUUCP logfile format  
e.g:  
(09/07-18:30:10) Smail,-,- cmd: "mail T:03 aussem" from wolfhh!chris

SMMaillogfile writes a logfile in wCNews style  
e.g:  
Sep 10 20:05:37: + 'aussem' 'wolfhh!tester' 1354 bytes

If you want to disable a log file set the entry in "uulib:config" to nil:.

## 1.24 RMail mode

If Smail is called with the program name (argv[0] for C programmers) rmail, Smail/rmail switch to the rmail mode (the same as if you start Smail -r ...). In this mode Smail/rmail does not parse the mail body (e.g. for receiver or sender of the mail).

If Smail is called with the program name (argv[0] for C programmers) rsmtp or rcsmtmp or rfsmtmp, Smail/rmail switch to the rmail and the BSMTMP mode (the same as if you start Smail -Br ...). Smail/rmail will try to extract the BSMTMP message and deliver this with rmail command.

## 1.25 Check user

Smail/rmail can check the permission, whether a local user is really allowed to send a mail via UUCP/BSMTMP. To activate this option just turn the SMCheckuser Option to "Yes" and setup who is allowed to mail via UUCP/BSMTMP.

The setup depends on the Smail/rmail version:

- o AXSh/Getty version  
All users that are written down line by line in "uulib:mail/checkuser" (see SMCheckuser and SMCheckFile) are allowed to send mails via UUCP/BSMTMP.
  - o MUFs version  
All users that are in group "mail", are allowed to send mails via UUCP/BSMTMP. Additionally all users that are
-

written down line by line in "uulib:mail/checkuser" (see SMCheckuser and SMCheckFile) are allowed to send mails via UUCP/BSMTP.

Attention!!!

Only the mails from local users are checked and the usernames in the checkuser file are case insensitive.

## 1.26 Local Mailer

If Smail/rmail delivers a local mail it calls the local mail program with the following syntax:

```
mail mail-file address0 .. addressN-1
```

Smail comes with three versions of mail:

- o mail\_axsh uses the AXSh passwd file to determinate whether a user exists
- o mail\_mufs uses the MUFs passwd file to determinate whether a user exists
- o mail uses uumail: mail folder to determinate whether a user exists. If no mail-folder exists the user does not exist.

If you want to deliver a mail to a non existing user set the environment variable MAILTONOEXIST.

## 1.27 BSMTP

BSMTP is a option for Smail/rmail to save transmission time, because your mails are compressed and batched (like you do with news).

You can activate the BSMTP protocol directly via the -B option. The better way is to write down all hosts that receives mails via BSMTP, in the "uulib:mail/bsmtp\_sites" file. Simply one site by every line (case insensitive!).

BSMTP collects all mail in uuspool:batch/<hostname>. Before you call your host you must execute Smail with -D option

Attention!!!

You have to batch your mails EACH TIME before calling your host ! If you forget to batch your mails they'll never be delivered !

Here comes an example:

---

Your node for BSMTMP is testhh.

```
put in "uulib:mail/bsmtp_sites"
----- snip -----
testhh
----- snip -----
```

All mails Smail/rmail sends to testhh will be stored in "uuspool:batch/testhh".

Before you call testhh execute (don't forget it!)

Smail -Dh testhh

The command for compressing your mails can be changed via SMCompress config option (default compress). The command for your host receiving your BSMTMP mails can be changed via SMBsmtpCmd config option (default rcsmtp). Normally you should use the following pairs:

| SMCompress  |     | SMBsmtpCmd |
|-------------|-----|------------|
| compress    | <-> | rcsmtp     |
| freeze      | <-> | rfsmtp     |
| no compress | <-> | rsmtmp     |

If you want to receive your mails with BSMTMP you must copy (or make a link) from Smail/rmail to rcsmtp, rfsmtp and rsmtmp. Then talk to your host's system operator from which you want to receive BSMTMP mails.

## 1.28 Aliases

Alias format

Mailing list

Include other alias files

Mail forwarding

Command piping

## 1.29 Alias format

An attempt has been made to remain compatible with sendmail alias file format, though the syntax is much more format free than sendmail. As distributed, case differences are ignored when comparing names to aliases. Only alias names which resolve to the local host are recognized, and are stored in their local form. Lines which start with a white space are continuation lines. Parenthesized strings are taken as

comments (no nesting), as is anything after a '#'. Here are some examples:

```
# this whole line is a comment
#
# These are equivalent definitions

alias_name      recip1 recip2 recip3
alias_name:     recip1, recip2 , recip3

alias_name      recip1 recip2
                recip3

alias_name      recip1 # Recip1's name
                recip2 # Recip2's name
                recip3 # Recip3's name

alias_name      recip1 (Recp1's name) recip2 (Recp2's name)
                recip3 (Recp3's name)

alias_name@thishost      recip1 recip2 recip3

alias_name@thisdomain    recip1 recip2 recip3

thisdomain!alias_name    recip1 recip2 recip3
```

All aliases are recursive, so care must be taken in their definition. Smail aliasing attempts to prevent infinite loops, and to do what was intended by the user. For example, the alias:

```
mylogin mypc!mylogin mylogin
```

Expands to

```
mypc!mylogin mylogin
```

even though the second occurrence of mylogin matches the alias name. If you want to prevent Smail from alias the second mylogin, use a \ before mylogin.

Both forms of file inclusion are recursive, too, so watch out for nesting include files. They may lead to infinite loops.

While the cost of parsing an alias file is usually negligible, it's wise to take savings anywhere savings can be found. Therefore, it's worth mentioning Smail's parsing strategy. Smail will try to get by with doing as little work as possible when aliasing. If on a particular invocation of Smail, none of the recipient addresses are local, (i.e., not potential aliases) then the aliases file won't even be read. Similarly, when an aliases file is read, it does not expand any of the :include: files until they are referenced. Thus, in the alias (above) for mylist, the file :include:uulib:mylist would not be opened and read (parsed) unless mail was sent to mylist. Wise use of :include: files can greatly increase the efficiency of the alias utility.

It's not clear exactly where the break-even point is when deciding to use an `:include:` file in an alias, versus having all of the recipients listed on the line; but if a mailing list is large (whatever that means) it is wise to use the `:include:` feature to save on parsing costs. Note that this discussion only applies to the first form of file inclusion, since reading an aliases file constitutes a reference to `:include:` files of the second form.

### 1.30 Mailing list

Mailing lists are easily handled by two forms of file inclusion. The first form is the same as is supported by `sendmail`

```
mylist      :include:uulib:mylist
```

In this example, each entry in `uulib:mylist` would be added to the alias for `mylist`. The second form is unique to Smail. It allows the aliases file to include other aliases files.

### 1.31 Include other alias files

```
:include:uulib:more-aliases
```

This would include the file `uulib:more-aliases` as a regular alias file. This makes it easier to maintain groups of aliases that change frequently, such as the list of netnews moderators.

### 1.32 Mail forwarding

There is another form of aliasing which works with the alias capability. This is called user forwarding. For a given user name, if there is no alias for the user and the file `users-home/.forward` exists its contents will be treated as an alias for the user. The syntax is the same as that of the recipient lists in the alias file described above. If you want to use forwarding in `uumail:` put a file with the name of the user you want to forward in `uumail:` and let the first line have starts with `"Forward:"`. The rest of the file will be treated as an alias described above.

### 1.33 Command piping

If the alias starts with an `|` Smail will call the string after the `|` with the mail file as input.

---

## 1.34 Thanks

for Betatesting...

Gerhard Schneider  
Thorsten Gau  
Christoph Haas

## 1.35 Version and Author

This Smail based on Smail V2.5 from Christopher Seiwald in 1987.  
Enhancements, enhancement requests, trouble reports, etc.,  
should be sent to

aussem@mavhh.hanse.de

\$Id: smail.txt,v 1.10 1993/11/16 23:52:00 Aussem Exp \$

## 1.36 Wish list

- o SMTP via AmiTCP (I need a beta tester for this option !)
- o gdbm support for the alias/path/bsmtp and user files

## 1.37 Copyright

This program is free software; you can redistribute it and/or  
modify it under the terms of the GNU General Public License as  
published by the Free Software Foundation; either version 2 of  
the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,  
but WITHOUT ANY WARRANTY; without even the implied warranty of  
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU  
General Public License for more details.

You should have received a copy of the GNU General Public License  
along with this program; if not, write to the Free Software  
Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

---

## 1.38 Source files

You need to recompile them SAS C V6.3 and (HW)RCS. If you want to recompile the AXSh version, you need the source files of AXSh.lib to compile a resident and register parameter version of the axsh.lib. If you want to recompile the MultiUserFileSystem you the include files from MUFS distribution.

Attention!!!

If you change something,

- a) please inform me, so that I can submit your changes in the "official" distribution.
- b) check your changes in as branches ! It is easier for me to merge them.

## 1.39 History

1.0 - initial release