The Elm Alias System Users Guide

What aliases are and how to use them
in the **Elm** *mail system*

The Elm Mail System
(Version 2.4)

Syd Weinstein, Elm Coordinator
Datacomp Systems, Inc.
3837 Byron Road
Huntingdon Valley, PA 19006-2320

email: elm@DSI.COM or dsinc!elm

The Elm Alias System Users Guide

(The Elm Mail System, Version 2.4)

October 1, 1992

Syd Weinstein
Datacomp Systems, Inc.
3837 Byron Road
Huntingdon Valley, PA 19006-2320

email: elm@DSI.COM or dsinc!elm

Derived from
‘‘The Elm Mail System, Version 2.0’’
by
Dave Taylor
Intuitive Systems
Mountain View, California
email: taylor@intuitive.com or limbo!taylor

This document is intended as a supplement to the *Elm Users Guide* and is only of interest to those users desiring more knowledge about how aliases work and how to create strange and exciting aliases for their systems (okay, so maybe it’s not *that* exciting!)

This document is broken up into the following sections; user aliases, group aliases, system aliases, editing and installing new aliases, general warnings and other chit-chat.

1. User Aliases

The simplest type of aliases in the Elm system are individual user aliases. These are made up of three parts;

aliasname list = *username* = *address*

Where the *aliasname list* is either a single aliasname[1] or a list of aliasnames separated by commas. Aliasnames are case insensitive and will be converted to all lower case by the alias system.

1. Aliasnames can be any combination of letters, numbers, hyphens (‘–’), periods (‘.’), or underscores (‘_’). Letter case is not significant, that is, ‘‘FRED’’, ‘‘Fred’’, and ‘‘fred’’ are identical.

Username is used to indicate the full ''real name'' of the user. For example, if you had an alias for ''taylor'' to get to me, the *username* field would contain ''Dave Taylor'' or perhaps ''Dave Taylor at HP'' or some other permutation. The name can optionally be entered in the format of ''Taylor; Dave''. When this is done, the alias system can sort aliases on a last name basis for presentation on the alias menu within Elm. Elm uses this information to add the user's full name to the addresses of outbound mail to get more readable addresses. No matter which format a name is entered, Elm displays it and puts it in mail headers the way you would expect (i.e., ''Dave Taylor'').

Address is either the user's full electronic mail address or, if your system uses a smart mailer, the minimum address needed to specify the destination. For example, say my smart mailer contained information on how to get to machine ''hp-sdd'' and I wanted to have an address for my friend Ken there — I could have his address specified as simply ''ken@hp-sdd'' (or alternatively ''hp-sdd!ken'' since the two are functionally equivalent).

Let's get on to some examples, shall we?

Consider this excerpt from my own *.elm/aliases.text* file;

```
# Mail aliases for Dave Taylor
# Friends from UCSD
addie          = Addie Silva    = addie@hp-sdd.SDD.HP.COM
frank,minjarez = Frank Minjarez = Minjarez.Verify@dockmaster.ddn.mil
pizzini        = Ken Pizzini    = hplabs!ucbvax!ucdavis!pai!ken
george         = Burdell; George P., Famous GT Guy= gburdell@matd.gatech.edu
```

Note that the alias for Frank Minjarez has two *aliasnames* associated with it, *frank* and *minjarez*. Also notice that the first and second aliases use the Internet style naming convention of *user@machine* whereas the third uses the *uucp* notation of *machine!user*. The fourth alias illustrates two features for the *username* field. An individual's name can be entered in the form of *lastname; firstname, comment* which allows the alias system to be able to sort alias names on a last name basis. Additionally, any text after the comma is treated as a comment. It is stored in the alias file and displayed on the alias menu but will not appear in the comment field (between the parentheses) in the header of a message. This allows you to enter useful notes about an individual that you would not otherwise want to include in the **To:** header of a mail message.

For the most part, the notational format is transparent within the alias file — the only time it *does* make a difference is if you have to specify more than the machine that the user is receiving mail on (as in the third example above).

Suffice to say that if you must specify any sort of uucp route that you should use the uucp notation as much as possible to ensure that the system expands the correct machine name. Similarly, if you're bouncing mail through different Internet sites (to get to a foreign system, as in the example below) you should use the notation that system expects, e.g.:

```
listserv%byuadmin.BITNET@rugters.edu
```

2. Group Aliases

After the confusion of user aliases, group aliases are even more fun! For the most part the notation is very similar:

aliasname list = *groupname* = *address list*

Where *aliasname list* and *groupname* are exactly equivalent to the corresponding fields in user aliases.

The interesting part is the *address list* field; this field is actually in the same notation as the aliasname list (i.e., a list of addresses separated by commas), so it's really quite easy to create. It's best to illustrate by example:

```
friends, mypals = The Gang of Six = joe, larry, mary, joanna,
                                    nancy, michael
```

Notice that you can continue onto as many lines as you'd like so long as each additional line starts with either a *<space>* or a *<tab>* character.

At times it may be desirable to send a message to everyone in the group except one or two people. This can be accomplished by adding the individuals' actual IDs to the **To:** list, preceded by a minus sign ('−'). For example, if you were having a surprise party for *nancy* from the above group and you wanted to discuss the details with the others, you could mail the message to:

```
friends −nancy
```

Unlike in prior versions of Elm, group alias addresses need not be *previously defined aliases* or valid mail addresses on the current machine. Any valid delivery address, local user, or prior user or group alias may now be part of a group alias *address list*.

3. System Aliases

System aliases are functionally equivalent to the individual Elm alias lists each Elm user has (both user aliases and group aliases) but are *read only* for everyone but the Elm administrator. The format of the file is identical to the users' files, and the only difference is that this file is expected to be located in the directory that contains the **system_hash_file** and **system_data_file** files (see the *Elm Configuration Guide* for more details on these variables).

Simply create the system alias file in the specified directory as you would a normal alias file, and install it using the newalias command with the -g option (see the following section for more details on that). Voila!!

4. Editing and Installing New Aliases

To install new aliases, you need merely to create, or modify, the file *$HOME/.elm/aliases.text* until you're satisfied with it and it meets the requirements discussed above. You can then try to install it with the command:

$ **newalias**

which either reports the number of aliases installed or the errors encountered trying to parse and store the given alias list.

Note that blank lines are no problem and that comments are not only allowed but actually encouraged, and must have '#' as the first character of each comment line.

5. Other Stuff not Covered Yet

Probably the biggest question you have in your mind right now is ''But how the heck does this relate to my existing *Berkeley Mail* aliases and the lower-level *sendmail* alias system?'' Well, rest assured that if you *really* want to have your aliases down in the transport you can. No problem.

And for those ex-*Berkeley Mail* fans, you can translate your aliases into the format that Elm wants by running them through the *awk* script listed in the appendix.

Finally, if you have any problems or questions, try looking in the *newalias* manual entry.

Appendix One
An *awk* Script for Translating Aliases
from a Berkeley Mail *.mailrc* File
to an Elm *.elm/aliases.text* File

```
BEGIN { print "# Elm .elm/aliases.text file, from a .mailrc file..."
      print ""
      }
next_line == 1 {
      next_line = 0;
      group = ""
      for (i = 1; i <= NF; i++) {
        if (i == NF && $i == "\\") sep = ""
        else                       sep = ", "
        if ($i == "\\") {
          group = sprintf("%s,", group)
          next_line = 1;
        }
        else if (length(group) > 0)
          group = sprintf("%s%s%s", group, sep, $i);
        else
          group = $i;
        }
        print "\t" group
      }
$1 ~ /[Aa]lias | [Gg]roup/ {
      if ( NF == 3)
        print $2 " = user alias = " $3;
      else {
        group = ""
        for (i = 3; i <= NF; i++) {
          if (i == NF && $i == "\\") sep = ""
          else        sep = ", "
          if ($i == "\\") {
            group = sprintf("%s,", group)
            next_line = 1;
          }
          else if (length(group) > 0)
            group = sprintf("%s%s%s", group, sep, $i);
          else
            group = $i;
          }
          print $2 " = group alias = " group;
        }
      }
```

Note: this script is contained in the release as *utils/mailrc.awk*.