

Elm

Aliases, Folders und Filter

Autor: Hubert Feyrer
(feyrer@rrzcl.rz.uni-regensburg.de)

Version 1.0
vom
23. Juli 1993

Diese Dokumentation zum Mail-Client „Elm“ beschreibt zuerst das Anlegen von Aliases für einzelne Personen und Gruppen von Personen, dann das Arbeiten mit mehreren Foldern. Anschließend wird ein Programm aus dem Elm-Archiv erläutert, das es ermöglicht, Briefe automatisch bei ihrem Eintreffen in Folder anzulegen. Der Anhang gibt ein Beispiel zur Elm-Konfiguration mit Hilfe der Datei `~/elm/elmrc`.

Hubert Feyrer:

Elm — Aliases, Folders und Filter.

Diesen Text finden Sie als PostScript-Datei `elm.ps` auf dem FTP-Server `ftp.uni-regensburg.de` im Verzeichnis `/pub/docs/rz`.

Diese Einführung wurde unter \LaTeX mit deutscher Anpassung gesetzt und mit `dvips` nach PostScript konvertiert.

Der vorliegende Text wurde mit größter Sorgfalt geschrieben. Der Autor und die Universität Regensburg können für Fehler und daraus resultierende Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen.

Inhaltsverzeichnis

1	What is it?	5
2	Aliases	7
2.1	Manuelles Ändern und Ergänzen von Aliases	7
2.2	Aliases ändern mit Elm	8
2.3	Beispiel: Verteiler-Liste	8
3	Folder	11
3.1	Ablegen von Nachrichten	11
3.2	Lesen von Foldern	11
4	Filter	13
4.1	Regeln	13
4.2	Überprüfen der Regeln	14
4.3	Aufruf	14
4.4	Beispiel: Aussortieren der Linux-Mailing-List	14
A	Mein elmrc :-)	17
B	Literaturverzeichnis	19

1 What is it?

Elm ist ein Shell-basierender Mail-Client, nutzt jedoch — im Gegensatz zu vielen seiner Vorgänger, siehe [19] — den gesamten Bildschirm aus.

Außerdem bietet es ein Reihe von Leistungsmerkmalen, die man anderswo vergeblich sucht. Dazu zählen:

- Das Anlegen eigener Aliases
- Das Arbeiten mit mehreren Ablagen (Foldern)
- Das automatische Einsortieren (filtern) der ankommenden Post in die verschiedenen Ablagen nach frei wählbaren Kriterien
- Die Möglichkeit, Eingabemasken (Forms) zu definieren, auszufüllen und zu versenden
- Das Verschlüsseln von Teilen des Briefes
- Das Versenden von beliebigen Daten (z. B. Bilder, Texte, ...) als Anlagen

Der vorliegende Text geht auf die ersten drei Punkte ein, die Elm-Dokumentation liefert weitere Details zu sämtlichen oben aufgeführten Features.

2 Aliases

Mit Hilfe von Aliases ist es möglich, Personen unter einem anderen Namen per EMail zu erreichen (Personen-Alias) bzw. mehrere Personen zu einer EMail-Adresse zusammenzufassen (Gruppen-Alias).

2.1 Manuelles Ändern und Ergänzen von Aliases

Die Aliases, die jeder Benutzer selbst verwaltet, liegen in den Datei `~/elm/aliases.text` im Home-Directory. Sie liegen außerdem noch in binärer Form vor, dazu gleich mehr.

Die Datei `aliases.text` besteht aus einzelnen Zeilen, von denen jede ein Alias definiert. Eine solche Zeile besteht dabei aus drei Feldern, durch „=" getrennt:

Alias-Namensliste = voller Name = EMail-Adresse

Aliasnamens-Liste ist dabei der oder die Namen, unter dem man die entsprechende Person erreichen will; Definiert man mehrere Aliases, so sind diese mit „," zu trennen. *Voller Name* sollte genau diesen enthalten, er wird dann bei Post von dieser Person beim Absender anstatt der EMail-Adresse angegeben. Mit der *EMail-Adresse* wird dann schließlich die volle EMail-Adresse der entsprechenden Person angegeben.

Beispiel:

```
hubert,feyrer = Hubert Feyrer = feyrer@rrzc1.rz.uni-regensburg.de
```

Damit kann man den Autor dieses Textes sowohl als „hubert“ als auch als „feyrer“ anschreiben, beides kommt korrekt an. Wer andersherum von mir Post bekommt, bekommt als Absender „Hubert Feyrer“ von Elm angezeigt, und nicht mehr `feyrer@rrzc1.rz.uni-regensburg.de` o. ä..

Gruppenaliases gehen analog dazu:

Alias-Namensliste = voller Name = Aliases

Alias-Namensliste gibt wiederum einen oder mehrere Namen an, unter denen man die Leute ansprechen möchte. Mit dem *vollen Name* kann man die Gruppe im Klartext beschreiben. *Aliases* ist schließlich eine Liste mit den Aliases der Leute, die der Gruppe angehören sollen. Es macht Sinn, hier nur Aliases und keine EMail-Adressen zuzulassen, da bei Antworten die Absender identifiziert werden müssen.

Auch hierzu ein Beispiel:

```
# Personenaliases
hubert,feyrer = Hubert Feyrer = feyrer@rrzc1.rz.uni-regensburg.de
wuerfl       = Karl Wuerfl   = wuerfl@rrzs1.rz.uni-regensburg.de

# Gruppe mit zwei Mitgliedern
news-admin   = News-Admin    = feyrer,wuerfl
```

Hier werden zuerst zwei Personen-Aliases definiert. Diese werden dann anschließend zur Gruppe „news-admin“ zusammengefaßt, ein Brief an „news-admin“ erreicht beide Mitglieder.

Das Beispiel zeigt außerdem, daß Leerzeilen und Zeilen, die mit einem „#“ beginnen, ignoriert werden. Letztere können für Kommentare verwendet werden.

Nachdem die Aliases in `aliases.text` definiert wurden, müssen sie noch in eine binäre Form gebracht werden, die einen schnelleren Zugriff ermöglicht. Dies geschieht mit dem Befehl „`newalias`“, er muß nach jeder Veränderung der Alias-Datei ausgeführt werden; Erst dann kann Elm auf die Aliases zugreifen.

2.2 Aliases ändern mit Elm

Es ist auch möglich, Aliases vom Elm aus zu verwalten. Dazu wählt man im Index mit „a“ den Alias-Modus an.

Dort kann man mit „l“ alle definierten Aliases auflisten lassen, mit „m“ kann man neue Aliases anlegen. Dabei wird wiederum nach dem zu definierenden Alias und dem vollen Namen gefragt. Anschließend ist entweder die EMail-Adresse oder eine Liste von Aliases anzugeben, entsprechen, ob es sich um ein Personen- oder ein Gruppen-Alias handelt.

Mit „q“ verläßt man den Alias-Modus wieder, Elm startet automatisch „`newalias`“ und die Aliases können dann verwendet werden.

Es ist darauf zu achten, daß man ein Alias nicht definieren und gleich anschließend wieder löschen kann, ohne den Alias-Modus zu verlassen, da das Alias noch nicht offiziell bekannt ist!

2.3 Beispiel: Verteiler-Liste

Der Alias-Mechanismus kann auch dazu eingesetzt werden, private Verteilerlisten aufzusetzen. Im Folgenden soll als Beispiel eine Verteilerliste von Unix-Systemadministratoren auf dem Campus zusammengestellt werden¹.

Folgende Personen sollen auf der Verteiler-Liste stehen:

- H. Feyrer (`root@news.uni-regensburg.de`)
- M. Kussinger (`root@rphc1.rz.uni-regensburg.de`)
- W. Pulina (`pulina@rrzc1.rz.uni-regensburg.de`)
- F. Wunsch (`wunsch@rphs1.rz.uni-regensburg.de`)
- K. Würfl (`wuerfl@rrzs1.rz.uni-regensburg.de`)

Wie üblich muß zuerst wieder eine Liste mit Personen-Aliases ausgestellt werden:

```
# Personenaliases
feyrer    = H. Feyrer          = root@news.uni-regensburg.de
kussinger = M. Kussinger       = root@rphc1.rz.uni-regensburg.de
```

¹Die Liste erhebt bei weitem keinen Anspruch auf Vollständigkeit!


```
pulina    = W. Pulina          = pulina@rrzc1.rz.uni-regensburg.de
wuensch   = F. Wuensch        = wuensch@rphs1.rz.uni-regensburg.de
wuerfl    = K. Wuerfl         = wuerfl@rrzs1.rz.uni-regensburg.de
```

Diese Personen können nun zu einer Gruppe von System-Administratoren „SysAdm“ zusammengefasst werden:

```
# lauter SysAdmins
SysAdm = Unix-Systemadministratoren = feyrer,kussinger,
                                           pulina,wuensch,wuerfl
```

Nachdem dies in `~/elm/aliases.text` abgestellt wurde und „newalias“ gelaufen ist kann mit „elm SysAdm“ Post an alle System-Administratoren verschickt werden.

Mit Hilfe von Verteilerlisten können beliebig eigene Aliases für Personengruppen erstellt werden, um den Nachrichtenaustausch mit mehreren Leuten — etwa zur reinen Informationsübermittlung — zu erleichtern.

3 Folder

Folder sind Ordnern, in die Briefe einsortiert werden können. Dies kann wahlfrei nach ihrem Inhalt, Absender, Datum, ...geschehen.

3.1 Ablegen von Nachrichten

Mit „s“ kann im Elm ein Brief in eine Datei abgespeichert werden. Enthält der Dateiname als erstes Zeichen ein „=“, so wird der Name als Folder-Name interpretiert, in den der Brief dann gespeichert wird. „=“ wird zu ~/News expandiert (falls in der Datei ~/elm/elmrc nicht anders eingestellt), die Folder werden dort angelegt.

3.2 Lesen von Foldern

Beim Start von Elm bekommt man als Default die neu angekommenen Briefe angezeigt. Möchte man jedoch einen anderen Folder anwählen, so kann man dies mit „c“. Es wird daraufhin der Name des Folders abgefragt, ein „=“ davor ist nicht nötig; Mit „?“ kann man sich alle Folder anzeigen lassen.

Nachdem man einen neuen Folder angewählt hat, kann man die darin enthaltenen Briefe wie gewohnt lesen, abspeichern, löschen,

4 Filter

Wohl jeder, der schon einmal mehr als eine Mailing-Liste bestellt hat, kennt das Problem: Man bekommt soviel Post herein, daß das Aussortieren zum Fulltime-Job entartet. Selbst wenn das Archivieren der einzelnen Listen in Foldern von manchen Mailern unterstützt wird, so mußte man bisher immer noch selbst auswählen, welche Mail zu welcher Liste gehört.

Das muß nicht sein! Das zu Elm gehörende Programm „**filter**“ kann dazu verwendet werden, neu ankommende Post entsprechend definierbarer Regeln zu speichern, löschen, weiterzuleiten, etc.

4.1 Regeln

Die Regeln, gemäß denen die Post behandelt wird, stehen in der Datei `~/.elm/filter-rules`. Die Grundstruktur der Regeln ist immer gleich:

```
if (Bedingung) then Aktion
```

Die *Bedingung* besteht dabei aus beliebig vielen Feldern der Vergleichen „*Feld Relation Wert*“, mehrere solcher Vergleiche können mit „and“ verknüpft werden; mit „not“ ist eine Negation möglich.

Feld kann eine der folgenden Header-Zeilen sein:

- `subject`
- `sender`
- `from`
- `to`
- `lines`

Wird als *Feld lines* verwendet, dann sind die Relationen „<“, „>“, „>=“, „<=“, „!=“ und „=“ möglich. Die restlichen Felder können mit „=“ auf Gleichheit abgefragt werden, mit „matches“ oder „~“ kann überprüft werden, ob das Feld einem regulären Ausdruck entspricht.

An *Aktionen* steht Folgendes zur Auswahl:

- `delete`: Löscht entsprechenden Brief
- `save Foldername`: Speichert Brief in entsprechenden Folder
- `savecopy Foldername`: Speichert Brief in entsprechenden und gleichzeitig in Mailbox für hereinkommende Post
- `forward Adresse`: Leitet Brief an *Adresse* weiter
- `forwardc Adresse`: Leitet Brief an *Adresse* weiter und läßt eine Kopie in der Mailbox
- `execute Befehl`: Filtert den Brief als Standard-Eingabe in den angegebenen Befehl
- `executec Befehl`: Filtert den Brief als Standard-Eingabe in den angegebenen Befehl und läßt eine Kopie in der System-Mailbox

- **leave:** Läßt Brief in Mailbox

Beim Abspeichern des Briefes in einen Folder kann auf die folgenden Macros zugegriffen werden:

Macro	Expandiert
%d	Tag im Monat (1-31)
%D	Tag der Woche (0-6)
%h	Stunde (0-23)
%m	Monat (1-12)
%r	Rückadresse der Nachricht
%s	Betreff
%S	‘‘Re: Betreff’’
%t	Zeit (Stunden, Minuten), Format: SS:MM
%y	Jahre seit 1900
%&	Der letzte regulären Ausdrucks
%1-%9	Der i-te Teil des letzten regulären Ausdrucks

4.2 Überprüfen der Regeln

Nachdem die Regeln in `filter-rules` eingetragen wurden, ist es ratsame, sie auf Syntax-Fehler überprüfen zu lassen. Dazu `filter` mit der Option `„-r“` (Regel) aufrufen: `„filter -r“`. Ist alles korrekt, so werden die einzelnen Regeln nochmals im Klartext ausgegeben. Fehler werden entsprechend angezeigt.

4.3 Aufruf

Nachdem die Regeln passen, kann man sich daranmachen, `„filter“` zu benutzen. Dazu braucht man `„sendmail“` oder einen anderen Mail Transport Agent, der es erlaubt, Nachrichten weiterzuleiten. Bei Verwendung von `„sendmail“` ist folgender Befehl in die Datei `~/forward` einzutragen:

```
"| /usr/local/bin/filter"
```

Dabei werden eventuelle Fehlermeldungen allerdings unterschlagen. Um diese Meldungen zu erhalten bietet es sich an, sie mittels der `„-o“`-Option in eine Datei umzuleiten, die Anführungszeichen müssen übrigens mit angegeben werden:

```
"| /usr/local/bin/filter -o /tmp/c9020.filter-errors"
```

Die Elm-Dokumentation enthält detaillierte Informationen, siehe [17].

4.4 Beispiel: Aussortieren der Linux-Mailing-List

Das folgende Beispiel soll zeigen, wie mit Hilfe des Elm-Filter-Systems Post von Mailing-Lists aussortiert werden soll. Dabei sollen die Kanäle `NORMAL`, `DOC`, `LINUXNEWS`, `SCSI`, `ANNOUNCE`, `TAPE` und `680X0` der Linux-Mailing-List dienen. Die Gruppen wurden folgendermaßen bestellt:

```
echo 'X-Mn-Admin: join ANNOUNCE' | mail linux-activists-request@niksula.hut.fi
echo 'X-Mn-Admin: join DOC' | mail linux-activists-request@niksula.hut.fi
echo 'X-Mn-Admin: join LINUXNEWS' | mail linux-activists-request@niksula.hut.fi
echo 'X-Mn-Admin: join SCSI' | mail linux-activists-request@niksula.hut.fi
echo 'X-Mn-Admin: join TAPE' | mail linux-activists-request@niksula.hut.fi
echo 'X-Mn-Admin: join 680X0' | mail linux-activists-request@niksula.hut.fi
echo 'X-Mn-Admin: join NORMAL' | mail linux-activists-request@niksula.hut.fi
```

Um diese für jeden Monat in entsprechende Folder abzulegen muß in ~/elm/filter-rules folgendes stehen:

```
if(to="Linux-Activists" and subject="ANNOUNCE")      savecopy "Linux-ANNOUNCE-%m"
if(to="Linux-Activists" and subject="DOC Channel")   savecopy "Linux-DOC-%m"
if(to="Linux-Activists" and subject="Linux News")   savecopy "Linux-LINUXNEWS-%m"
if(to="Linux-Activists" and subject="SCSI Channel") savecopy "Linux-SCSI-%m"
if(to="Linux-Activists" and subject="TAPE Channel") savecopy "Linux-TAPE-%m"
if(to="Linux-Activists" and subject="680X0 Channel") savecopy "Linux-680x0-%m"
if(to="Linux-Activists") savecopy "Linux-NORMAL-%m"
```


A Mein elmrc :-)

Hier soll an einem Beispiel aufgezeigt werden, welche Möglichkeiten Elm zur Konfiguration bietet. [18] enthält eine komplette Liste aller Optionen inklusive Beschreibung.

```
#####
#
# ~/.elm/elmrc                               Hubert Feyrer, 19930605-01
#
#####

# Zu verwendender Editor
editor          = emacs -nw %s -f text-mode -f turn-on-auto-fill

# Mein voller Name
fullname        = Hubert Feyrer

# Wer zeigt mir Nachrichten an?
pager           = /usr/ucb/more

# Hier den Drucker richtig angeben! Gilt fuer RZ-SUN-Pool
print           = lpr -Plaserlp

# Wo steht meine Signature?
signature       = ~/.signature

# Reihenfolge, in der die Nachrichten angezeigt werden
sortby          = reverse-received

# Welche Header sollen entfernt werden?
weedout         = "*clear-weed-list*"
                "Path:" "Via:" "Sent:" "Date:" "Status:" "Original"
                "Phase" "Return" "Posted" "X-" "To:" "Cc:" "cc:"
                "Lines:" "Xref:" "Received"
                "*end-of-user-headers*"

# Bei Loeschen nicht nachfragen
alwaysdelete    = on

# Ungelesene Mails immer im System-Folder halten
alwayskeep      = on

# Gelesene Mail nicht in "received"-Folder ablegen
alwaysstore     = off

# Werte aus "alwaysdelete", "alwayskeep" und "alwaysstore"
# verwenden, keine Rueckfragen!
```

```
ask                                = off

# Beim senden keine "Carbon Copies", bitte!
askcc                              = off

# Bei Antworten immer Originaltext einfüegen
autocopy                            = on

# Forms erlauben
forms                                = on

# Leere Folder nicht loeschen
keepempty                            = on

# Bei seitenweisem blaettern Cursor verschieben
movepage                             = on

# Nach Anzeigen eines Briefes warten
promptafter                          = on

# Keine Dashes vor Signature (was soll das ueberhaupt?)!
sigdashes                            = off

# Ungewollte Header entfernen
weed                                 = on
```

B Literaturverzeichnis

Literatur

- [1] Syd Weomstein (elm@DSI.COM). *answer(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [2] Syd Weomstein (elm@DSI.COM). *autoreply(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [3] Syd Weomstein (elm@DSI.COM). *checkalias(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [4] Syd Weomstein (elm@DSI.COM). *Elm Configuration Guide*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [5] Syd Weomstein (elm@DSI.COM). *Elm Forms Mode Guide*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [6] Syd Weomstein (elm@DSI.COM). *elm(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [7] Syd Weomstein (elm@DSI.COM). *fastmail(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [8] Syd Weomstein (elm@DSI.COM). *filter(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [9] Syd Weomstein (elm@DSI.COM). *frm(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [10] Syd Weomstein (elm@DSI.COM). *listalias(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [11] Syd Weomstein (elm@DSI.COM). *messages(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [12] Syd Weomstein (elm@DSI.COM). *newalias(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [13] Syd Weomstein (elm@DSI.COM). *newmail(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [14] Syd Weomstein (elm@DSI.COM). *printmail(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [15] Syd Weomstein (elm@DSI.COM). *readmsg(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [16] Syd Weomstein (elm@DSI.COM). *The Elm Alias System Users Guide*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [17] Syd Weomstein (elm@DSI.COM). *The Elm Filter System Guide*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [18] Syd Weomstein (elm@DSI.COM). *The Elm Reference Guide*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [19] Syd Weomstein (elm@DSI.COM). *The Elm Users Guide*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.

- [20] Syd Weomstein (elm@DSI.COM). *wnewmail(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.