

SS 1993

## Bericht im zweiten praktischen Studiensemester

### Hauptthemen:

Installation und Betreuung eines NetNews-Servers,  
Installation und Dokumentation von IRC- und Mail-Clients.

Name, Vorname: Feyrer, Hubert

Studiengruppe: I6T

### Ausbildungsbetrieb:

Rechenzentrum der  
Universität Regensburg  
Universitätsstr. 31  
8400 Regensburg

Abgabetermin: 25. Juni 1993

### Kenntnis genommen:

.....

(Hr. K. Würfl, Praktikumsbetreuer)



## Vorwort

Dieser Praktikumsbericht gliedert sich in drei Teile auf, entsprechend den drei im Praktikum behandelten Themengebieten.

Zuerst wird ausführlich das Thema NetNews diskutiert, darauf folgt eine Einführung in das Internet Relay Chat (IRC) und die Beschreibung einiger Features des Mail-Clients „Elm“.

Die Einführung zum Thema NetNews erklärt, was NetNews sind, und welche Konventionen im zugehörigen Netzwerk — dem Usenet — gängig sind. Als nächstes folgt dann eine Beschreibung der Server- und Client-Installation, gefolgt von einer allgemeinen Aufzählung der Leistungsmerkmale von Newsreadern. Am Beispiel von TIN und XRn wird gezeigt, welche Funktionen benötigt werden, um das Medium News auszuschöpfen. Der weitere Text enthält eine Kurzbeschreibung der zur Zeit an der Uni Regensburg existierenden Gruppen, sowie eine Auflistung interessanter Gruppen zu den einzelnen Betriebssystemen. Der letzte Teil gibt Hinweise für die Installation der bereits übersetzten Newsreader auf eigenen Workstations.

Die Einführung ins IRC im zweiten Teil besteht aus einer allgemeinen Erläuterung, was IRC ist und welche Konventionen gelten. Nachdem die einzelnen Befehle beschrieben werden, folgt ein Abschnitt mit Informationen zur Vertiefung. Den Abschluß bilden Antworten auf häufig gestellte Fragen.

Die Dokumentation zum Mail-Client „Elm“ beschreibt zuerst das Anlegen von Aliases für einzelne Personen und Gruppen von Personen, dann das Arbeiten mit mehreren Foldern. Im Anschluß daran 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

Mallersdorf, 15. Juni 1993



# Inhaltsverzeichnis

<b>I</b>	<b>NetNews</b>	<b>5</b>
<b>1</b>	<b>Was sind NetNews?</b>	<b>5</b>
<b>2</b>	<b>Konventionen</b>	<b>6</b>
<b>3</b>	<b>Funktionsumfang der NewsReader</b>	<b>6</b>
3.1	TIN (Unix) . . . . .	11
3.2	XRn (Unix) . . . . .	15
3.3	Andere Betriebssysteme . . . . .	17
<b>4</b>	<b>Server-Installation</b>	<b>19</b>
4.1	Komponenten des News-Servers . . . . .	19
4.1.1	Die Hardware: Sun SPARCstation 10 . . . . .	19
4.1.2	Das Betriebssystem: SunOS 4.1.3 . . . . .	19
4.1.3	Die Software: InterNetNews (INN) . . . . .	19
4.2	Vorbereiten des Rechners . . . . .	20
4.3	Installation von INN 1.2 . . . . .	21
4.4	Lokale Erweiterungen und Ergänzungen . . . . .	23
4.4.1	Newsgroups anlegen . . . . .	23
4.4.2	Vorhandene Newsgroups ausdrucken: <code>active2ps</code> . . . . .	25
4.4.3	Volumen-Statistik mit <code>volume</code> . . . . .	28
<b>5</b>	<b>Compilieren der Clients</b>	<b>33</b>
5.1	GNU grep . . . . .	33
5.2	GNUS . . . . .	33
5.3	nn . . . . .	33
5.4	rn . . . . .	34
5.5	TIN . . . . .	35
5.6	XRn . . . . .	35

5.7	XVnews . . . . .	35
<b>6</b>	<b>Client-Installation</b>	<b>37</b>
<b>7</b>	<b>Lokale Newsgroups</b>	<b>39</b>
<b>8</b>	<b>Infos für EDV-Interessierte</b>	<b>41</b>
8.1	Sun . . . . .	41
8.2	Ultrix . . . . .	41
8.3	VMS . . . . .	42
8.4	MS-DOS . . . . .	43
8.5	MS-Windows . . . . .	43
8.6	OS/2 . . . . .	44
8.7	Amiga . . . . .	44
8.8	Atari . . . . .	45
8.9	Grafik . . . . .	45
<b>II</b>	<b>IRC</b>	<b>47</b>
<b>1</b>	<b>Einführung</b>	<b>47</b>
<b>2</b>	<b>Konventionen</b>	<b>49</b>
<b>3</b>	<b>Erste Schritte</b>	<b>51</b>
<b>4</b>	<b>Befehle</b>	<b>53</b>
4.1	Editier-Befehle . . . . .	53
4.2	Allgemeine Befehle . . . . .	53
4.3	Kommunikation und private Konversation . . . . .	54
4.4	Kanäle und Öffentliche Konversation . . . . .	55
4.5	Kanal- und Benutzer-Modi . . . . .	56
4.6	Client to Client Protocol . . . . .	58
4.7	Netzwerk-Befehle . . . . .	59

<b>5</b>	<b>Weitere ircII-Zaubertricks</b>	<b>61</b>
<b>6</b>	<b>Häufig gestellte Fragen</b>	<b>63</b>
6.1	Wo bekomme ich einen IRC-Client her? . . . . .	63
6.2	Welchen Server soll ich nehmen? . . . . .	63
6.3	Ein paar ausgewählte Channels . . . . .	63
6.4	Was ist NickServ? . . . . .	64
6.5	Wo kann ich weitere Informationen finden? . . . . .	64
<b>III</b>	<b>Elm</b>	<b>65</b>
<b>1</b>	<b>What is it?</b>	<b>65</b>
<b>2</b>	<b>Aliases</b>	<b>67</b>
2.1	Manuelles Ändern und Ergänzen von Aliases . . . . .	67
2.2	Aliases ändern mit Elm . . . . .	68
2.3	Beispiel: Verteilier-Liste . . . . .	68
<b>3</b>	<b>Folder</b>	<b>71</b>
3.1	Ablegen von Nachrichten . . . . .	71
3.2	Lesen von Foldern . . . . .	71
<b>4</b>	<b>Filter</b>	<b>73</b>
4.1	Regeln . . . . .	73
4.2	Überprüfen der Regeln . . . . .	74
4.3	Aufruf . . . . .	74
4.4	Beispiel: Aussortieren der Linux-Mailing-List . . . . .	74
<b>IV</b>	<b>Anhang</b>	<b>77</b>
<b>A</b>	<b>Die Datei .newsrc</b>	<b>77</b>

<b>B Mein elmrc :-)</b>	<b>79</b>
<b>C Beispiel-.signatures</b>	<b>81</b>
<b>D Software-Herkunft</b>	<b>83</b>
<b>E Literaturverzeichnis</b>	<b>87</b>



## Teil I

# NetNews

## 1 Was sind NetNews?

Jeder Wissenschaftler kennt das Problem, sich relevante Informationen schnell und unkompliziert zu beschaffen. Im Zeitalter der weltweit vernetzten Computer gibt es neben der Recherche in Online-Datenbanken eine weitere rechnergestützte Informationsquelle, die in Deutschland jedoch noch nicht sehr verbreitet und akzeptiert ist. Es sind dies die sogenannten „NetNews“ (auch „UseNet-News“ oder einfach nur „News“).

„News“ ist ein globales Kommunikationssystem, an dem weltweit über 40.000 Serversysteme von Universitäten, Großforschungseinrichtungen, Firmen und Privatleuten mit schätzungsweise zwei Millionen Benutzern (Lesern) teilnehmen. Im Wesentlichen kann man die NetNews mit einer großen Zeitung vergleichen, die nur aus Leserbriefen besteht, eingeteilt in verschiedene Rubriken, die als „Newsgroups“ bezeichnet werden. 1992 existierten ca. 2000 international verbreitete Newsgroups in 20 formalen Hierarchien sowie ca. 85 deutschsprachige Gruppen. Die Newsgruppen lassen sich grob in drei Bereiche aufteilen:

- Computerbezogene Themen: `COMP.*`
- Wissenschaftliche Themen: `SCI.*`
- Themen aus allen Gebieten des täglichen Lebens (Politik, Freizeit, Religion, Gesellschaft, Kultur, ...): `ALT.*`, `REC.*`, `SOC.*`, ...

Das Grundprinzip im UseNet, so nennt man das Netz der an den NetNews teilnehmenden Rechner, beruht darauf, daß jeder Benutzer eigene Mitteilungen oder Kommentare zu Nachrichten anderer Benutzer in das Netz schicken kann. Auf diese Mitteilungen und Kommentare (sog. „Postings“ oder „Artikel“) können dann wieder andere antworten, so daß eine regelrechte Diskussion über die betreffenden Themen entsteht.

Die Anfänge des UseNet reichen bis in das Jahr 1979 zurück. Seit dieser Zeit hat nicht nur die Anzahl der teilnehmenden Rechner, sondern auch die Menge der transportierten Daten erheblich zugenommen. Die täglich erzeugte Datenmenge beträgt im Moment mehr als 50MB. Die darin enthaltenen Informationen lassen sich nur deshalb sinnvoll nutzen, weil NetNews durch Toplevel-Hierarchien thematisch gegliedert vorliegen, die noch weiter aufgeteilt sind. Als Toplevel-Hierarchie bezeichnet man den ersten Teil des Newsgroup-Namen (bis zum ersten Punkt). So kennzeichnet z. B. `DE` im Namen der Gruppe `DE.COMP.SOURCES.UNIX` die Zugehörigkeit zur Toplevel-Hierarchie `DE`. Die anderen Namensbestandteile `COMP`, `SOURCES` und `UNIX` werden dementsprechend als „Secondlevel-Hierarchie“, „Thirdlevel-Hierarchie“ etc. bezeichnet. Sie geben eine weitere Spezialisierung an: Themenbereich Computer, Quellcodes für Unix.

Man unterscheidet u. a. „globale“ und „nationale“ Toplevel-Hierarchien. Die globalen Hierarchien wurden mit der Absicht angelegt, sie weltweit zu verbreiten; die Kommunikationssprache ist englisch. Die nationalen Toplevel-Hierarchien werden normalerweise innerhalb des jeweiligen

Landes verteilt. Kommunikationssprache ist die Landessprache. Die Anzahl der täglich erzeugten Artikel verteilt sich vor allem auf die Hierarchien REC (Recreation, 31%), COMP (Computer, 24%), ALT (alternative Themen, 18%), SOC (gesellschaftliche Themen 13%) und SCI (Wissenschaft, 5%).

## 2 Konventionen

An einer Diskussion kann sich prinzipiell jeder beteiligen, der technisch dazu in der Lage ist. Eine Einschränkung nach Organisation, Zugehörigkeit zu Randgruppen oder Inhalt des Artikels geschieht nicht. Daß sich trotzdem in einem Netz von zwei Millionen Teilnehmern sinnvolle und konstruktive Diskussionen entwickeln können, resultiert aus der Disziplin der Benutzer. Die meisten halten sich freiwillig an übereinstimmend akzeptierte Richtlinien, die den Umgang miteinander und dem Medium NetNews regeln. So wird unter anderem erwartet, daß jeder die Artikel unter seinem eigenen Namen veröffentlicht und sich nicht hinter Pseudonymen versteckt. Ebenso sind persönliche Angriffe und Beleidigungen („flames“), aber auch kommerzielle Werbung unerwünscht. Diese „Netiquette“ getaufte Richtlinien werden regelmäßig in den Newsgruppen NEWS.ANNOUNCE.NEWUSERS und NEWS.ANSWERS sowie in DE.NEWUSERS in deutscher Sprache für neue Benutzer veröffentlicht. Häufig findet man dort auch Antworten zu häufig gestellten Fragen (Frequently Asked Questions, FAQ), siehe [7], [10].

Die meisten Artikel enthalten am Ende ein kleines persönliches Logo des jeweiligen Autors, die sog. Signature. Sie kann enthalten sein, muß aber nicht. Ihr Inhalt ist verschiedenster Art: Name, Adresse, Anschrift und Telefonnummer sind gebräuchlich, oft werden auch irgendwelche Sprüche oder Zeilen aus Liedern und Gedichten verwendet. Eine Signature ist durchschnittlich 4 Zeilen lang, was im Normalfall ausreicht, um den Autor zu identifizieren. Anhang C enthält einige Beispiele hierzu.

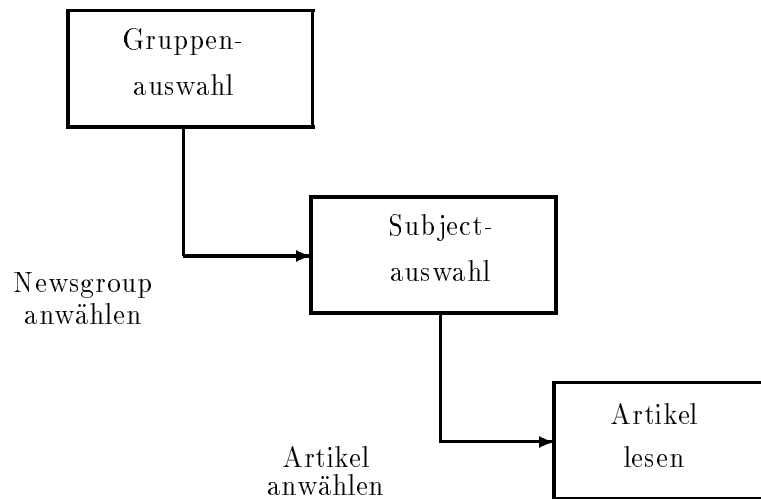
Die Einrichtung neuer Usenet-Gruppen mit nicht-lokaler Verbreitung wird in [4] im Detail für Gruppen der Hierarchien DE.\* und DE.ALT.\* besprochen.

## 3 Funktionsumfang der NewsReader

Ein NewsReader ist ein Programm auf dem eigenen Rechner, mit dem man an NetNews teilnehmen kann.

Dieser Abschnitt beschreibt die nötigen Aktionen, mit denen man sich durch die einzelnen NewsReader manövriert, um News zu lesen.

Das Schema ist immer dasselbe: Aus den angebotenen Gruppen wählt man eine aus, aus den in dieser Gruppe diskutierten Themen (Subjects) wählt man dann wiederum eines aus. Die folgende Grafik veranschaulicht dies:



Entsprechend hierzu Screenshots für Gruppenauswahl:

```

Group Selection (news.uni-regensburg.de 149)          h=help

 1      uni-regensburg.test
 2      uni-regensburg.aktuell
 3      uni-regensburg.probleme
 4      uni-regensburg.netzwerk
 5      uni-regensburg.mensa
 6      uni-regensburg.test.mod
 7      4 comp.os.linux.announce      Announcements impor
 8      3 comp.sys.amiga.announce     Announcements about
 9      comp.sys.sun.announce         Sun announcements a
10     comp.windows.x.announce       X Consortium announ
11     gnu.announce                  Status and announce
-> 12     6 news.announce.newusers    Explanatory posting
13     biz.sco.announce              SCO and related pro
14     1 de.admin.news.announce
15     gnu.emacs.announce            Announcements about

<n>=set current to n, TAB=next unread, /=search pattern, c)atchup,
g)oto, j=line down, k=line up, h)elp, m)ove, q)uit, r=toggle all/unread,
s)ubscribe, S)ub pattern, u)nsubscribe, U)nsub pattern, y)ank in/out
  
```

... Artikelauswahl:

```
news.announce.newusers (6T 6A 0K 0H R)                h=help
->  1 + Introduction to the *.answers newsgroups        Jonathan I. Kamen
    2 + Publicly Accessible Mailing Lists, Part 1/5    Stephanie da Silv
    3 + Publicly Accessible Mailing Lists, Part 2/5    Stephanie da Silv
    4 + Publicly Accessible Mailing Lists, Part 3/5    Stephanie da Silv
    5 + Publicly Accessible Mailing Lists, Part 4/5    Stephanie da Silv
    6 + Publicly Accessible Mailing Lists, Part 5/5    Stephanie da Silv

<n>=set current to n, TAB=next unread, /=search pattern, ^K)ill/select,
a)uthor search, c)atchup, j=line down, k=line up, K=mark read, l)ist thread,
|=pipe, m)ail, o=print, q)uit, r=toggle all/unread, s)ave, t)ag, w=post

*** End of Articles ***
```

... und das Lesen eines bestimmten Artikels:

```
Sun, 18 Apr 1993 07:00:06 news.announce.newusers      Thread 1 of 6
Article 30      Introduction to the *.answers newsgroups No responses
jik@GZA.COM      Jonathan I. Kamens at Geer Zolot Associates

Archive-name: news-answers/introduction
Version: $Id: Introduction,v 1.44 1993/04/18 01:00:32 jik Exp $

Introduction

This is the monthly introductory article for the moderated
newsgroups alt.answers, comp.answers, misc.answers, news.answers,
rec.answers, sci.answers, soc.answers, and talk.answers (hereafter
collectively referred to as "*.answers"). It explains the purpose of
the newsgroups, what kinds of articles should be submitted to them,
how to submit, how to participate in the mailing list for periodic

<n>=set current to n, TAB=next unread, /=search pattern, ^K)ill/select,
a)uthor search, c)atchup, f)ollowup, j=line down, k=line up, K=mark read,
|=pipe, m)ail, o=print, q)uit, r)epl) mail, s)ave, t)ag, w=post

--More--(11%) [1478/13096]
```

Folgende Aktionen sind mit den einzelnen Newsreadern möglich:

- **Start:** Die einzelnen Schritte, die nötig sind, um den NewsReader zu starten.
- **Subscribe/Unsubscribe:** Aus den mehr als 3000 angebotenen Gruppen die auswählen, die man öfter oder regelmäßig lesen möchte. Diese Angaben werden im Homedirectory des jeweiligen Benutzers in der Datei `.newsrc` angelegt<sup>1</sup>. Dort steht auch, welche Artikel einer Gruppe schon gelesen wurden.
- **Newsgroup anwählen:** Wie man aus den vielen Gruppen eine bestimmte auswählt.
- **Artikel auswählen:** Wie man aus den Artikeln einer Gruppe einen bestimmten Artikel auswählt, um ihn zu lesen.
- **Artikel speichern:** Abspeichern eines interessanten Artikels.
- **Alte Artikel lesen:** Es wird zwischen gelesenen und ungelesenen (neuen) Artikeln unterschieden. Wie liest man bereits gelesene (alte) Artikel nochmals?
- **Catch up:** Alle Artikel einer Gruppe als gelesen kennzeichnen.
- **Posten:** Wie stellt man eigene Fragen in einer bestimmte Newsgruppe? Siehe auch [8]: dort werden Sinn und Zweck der einzelnen Headerzeilen erläutert.
- **Followup:** Wie antwortet man auf Fragen anderer? Die beiden Möglichkeiten, per Mail oder per News zu antworten werden angesprochen.
- **Verlassen:** Beenden des jeweiligen NewsReaders.

---

<sup>1</sup>Erläuterungen hierzu sind in Anhang A zu finden.



### 3.1 TIN (Unix)

Die hier gemachten Angaben beziehen sich auf TIN Version 1.1, Patchlevel 9. Der NewsReader macht keinen Gebrauch der X-Window-Oberfläche, paßt sich aber der Größe des jeweiligen `xterm`-Fensters an. Er ist außerdem von den angebotenen Funktionen her sehr vielseitig und kann jedermann eigentlich nur wärmstens empfohlen werden.

- **Start:** Zuerst müssen die beiden Umgebungsvariablen „`NNTPSERVER`“ und „`REPLYTO`“ gesetzt werden. „`NNTPSERVER`“ enthält den Namen des lokalen News-Servers, in unserem Fall `news.uni-regensburg.de`. „`REPLYTO`“ enthält die vollständige eigene E-Mail-Adresse, also z. B. „`feyrer@rrzc1.rz.uni-regensburg.de`“. Am besten setzt man diese Variablen gleich im `.cshrc` (bei `tcsh`, `csch`):

```
setenv NNTPSERVER news.uni-regensburg.de
setenv REPLYTO feyrer@rrzc1.rz.uni-regensburg.de
```

bzw. im `.profile` bei `BASH`, Bourne/Korn-Shell):

```
NNTPSERVER=news.uni-regensburg.de      ; export NNTPSERVER
REPLYTO=feyrer@rrzc1.rz.uni-regensburg.de ; export DOMAIN
```

Wer zum Posten statt dem `vi` den `emacs` verwenden möchte, der muß noch die folgenden beiden Umgebungsvariablen setzen:

```
setenv VISUAL emacs
setenv TERM vt100      # sonst gibt's keine Cursor-Tasten
```

für `csch`/`tcsh` bzw. für `BASH`, Bourne/Korn-Shell:

```
VISUAL=emacs      ; export VISUAL
TERM=vt100        ; export TERM
```

- **Subscribe/Unsubscribe:** Beim erstmaligen Start bekommt man eine kurze Hilfestellung, und anschließend eine Liste aller auf dem Server existierenden NewsGroups, z. Zt. sind dies etwas über 3000. Diese bekommt man ab jetzt bei jedem Start des `TIN` angezeigt. Es ist jedoch sinnvoller, die NewsGroups, an denen man nicht interessiert ist, mit „`u`“ (unsubscribe) abzubestellen. Mit „`U *`“ kann man alle angezeigten Gruppen abbestellen.

Will man später eine neue oder vorher abbestellte Gruppe wieder regelmäßig lesen, so empfiehlt es sich, diese zu subscriben. Dazu ist zuerst die Liste aller Gruppen vom Server neu einzulesen: „`y`“ (Yank in subscribed/unsubscribed), und dann die entsprechende(n) Gruppe(n) mit „`s`“ (Subscribe) neu zu bestellen. Mehrere Gruppen können mit „`S`“, gefolgt von einem Suchmuster, bestellt werden.

Will man nur die bestellten Gruppen sehen und die restlichen ausblenden, so geschieht dies mit „`r`“, nicht bestellte Gruppen werden am Anfang der Zeile mit einem „`u`“ (unsubscribed) gekennzeichnet.

- **Newsgroup anwählen:** Als nächstes kann man sich daranmachen, die Gruppe auszuwählen die man lesen möchte. Wenn in einer Gruppe kein Artikel vorhanden ist oder alle

Artikel einer Gruppe bereits gelesen wurden, so wird diese nicht angezeigt. Will man aber z. B. einen alten Artikel nochmals lesen, dann kann man mit „r“ alle bestellten (subscribed) Gruppen einblenden.

Die Auswahl der gewünschten News-Gruppe geht am einfachsten mit den Cursortasten (rauf/runter), wenn man die Nummer der gewünschten Gruppe weiß (1. Spalte, ganz links), so kann man diese auch direkt eintippen. Mit der Leertaste geht's eine ganze Seite nach vorne, mit „~U“ geht's eine Seite zurück.

Freunde des vi können auch die Tasten „hjkl“ zur Cursorsteuer verwenden und mit „CTRL-f“ und „CTRL-b“ kann man seitenweise vor- und zurückblättern.

Zum Schluß einfach „RETURN“ drücken.

- **Artikel lesen:** Wieder wie beim Auswählen der NewsGroup mit Cursortasten, „hjkl“, „Space“, . . . , und mit „RETURN“ bestätigen. Den nächsten bzw. vorherigen Artikel erhält man mit „n“ bzw. „p“. Will man ein anderes Thema (Subject) wählen, so kommt man mit „q“ zum entsprechenden Auswahlmenü zurück.
- **Artikel speichern:** „s“ drücken, dann angeben ob man einen einzelnen Artikel („a“), den ganzen Gesprächsverlauf (Thread, „t“), alle Artikel, die zu einem Muster (Pattern) passen („P“) oder mit „T“ eine Reihe vorher mit „t“ markierter (tagged) Artikel abspeichern möchte.

Nach der Eingabe des Dateinamens<sup>2</sup> kann man dann angeben, ob der Artikel einen besonderen Inhalt hat, z. B. eine UU-codierte Binärdatei („u“), ein Shell-Archiv („s“), . . . Dies ist jedoch meist nicht der Fall, also einfach mit „n“ oder „RETURN“ bestätigen.

Existiert die angegebene Datei bereits, kann man den Artikel entweder hinten anfügen oder abbrechen.

- **Alte Artikel lesen:** Will man bereits gelesene Artikel nochmals lesen, so kann man sie mit „r“ ein- und wieder ausblenden.

Dies funktioniert auch bei der Auswahl der Newsgroup, da dort ebenfalls nur Gruppen mit neuen (ungelesenen) News aufgelistet werden.

- **Catch up:** Mit „c“ werden — nach Rückfrage — alle Artikel der gewählten Gruppe als gelesen kennzeichnen.
- **Posten:** Bei der Artikel-Auswahl „w“ tippen, dann den Betreff (Subject) des Artikels angeben. Als nächstes wird der Editor<sup>3</sup> gestartet und man kann seinen Artikel tippen. Mit Hilfe der Headerzeilen kann man noch einiges einstellen, [8] gibt hierüber Auskunft.

Dann den Text tippen, zum Schluß den Artikel abspeichern und den Editor verlassen.

Falls man den Artikel doch nicht abschicken will, kann man jetzt noch mit „q“ abbrechen, „p“ versendet ihn in alle Welt!

- **Followup:** Hat man einen Artikel gelesen und möchte darauf antworten, so kann dies entweder an den Autor persönlich (via E-Mail) oder an die ganze Newsgruppe geschehen, indem man ein „Followup“ postet.

In jedem Fall muss man den Artikel angewählt haben. Um dann per E-Mail zu antworten, drückt man einfach „r“ (Reply by mail), es wird dann wiederum der Editor geladen mit

---

<sup>2</sup>Alle relativen Pfadnamen beziehen sich hier auf das Verzeichnis ~News.

<sup>3</sup>vi, falls nicht anders über die Umgebungsvariable „VISUAL“ eingestellt.



dem soeben gelesenen Text als Zitat<sup>4</sup>. Dann die Antwort tippen, zum Schluß wiederum abspeichern und den Editor verlassen. In der darauffolgenden Sicherheitsabfrage kann man wiederum mit „q“ abbrechen oder mit „s“ den Artikel versenden.

Will man öffentlich auf einen Artikel antworten, so kann man mit „f“ ein Followup (Nachfolge) abschicken. Der weitere Ablauf ist derselbe wie beim Posten.

- **Verlassen:** Im Auswahlmenü für die NewsGroups „q“ drücken.

---

<sup>4</sup>Es ist Sitte, zwar den Urheber des Textes zu nennen, jedoch unwesentliche Stellen des Originals zu löschen, um kurz und prägnant zu bleiben.



## 3.2 XRn (Unix)

Beim XRn handelt es sich — im Gegensatz zum TIN — um einen reinen X11-Newsreader, er kann nicht mit einem ASCII-Terminal verwendet werden. Ebenso wie der TIN bietet er eine Vielzahl an Möglichkeiten. Hervorragend ist auch die Hilfsfunktion: Wenn man mit der Maus über einem Button steht, bekommt man angezeigt, was ein Druck auf diesen Button bewirken würde, so daß die Beschriftung des Buttons noch näher erläutert wird.

Das XRn-Fenster ist in zwei Hälften aufgeteilt. In der oberen werden die Gruppen und Themen (Subjects) angezeigt und ausgewählt, im unteren werden die Artikel zum Lesen angezeigt.

- **Start:** Vor dem Aufruf mit „xrn“ muß die Umgebungsvariable „NNTPSERVER“ auf den Namen des NNTP-Servers gesetzt werden. Dies geschieht bei Verwendung von `csch` bzw. `tcsh` am besten in der Datei `.cshrc`:

```
setenv NNTPSERVER news.uni-regensburg.de
```

bzw. im `.profile` bei BASH, Bourne/Korn-Shell):

```
NNTPSERVER=news.uni-regensburg.de ; export NNTPSERVER
```

Außerdem sollte in der Datei `.Xresources` (bei Verwendung von MIT-X11) bzw. unter OpenWindows in `.Xdefaults` die vollständige E-Mail-Adresse angegeben werden. Sie wird für Antworten auf gepostete Artikel benötigt. Der Eintrag sieht in jeder der beiden Dateien wie folgt aus:

```
xrn.replyTo: <EMail>
```

Statt `<EMail>` ist die eigene E-Mail-Adresse einzutragen.

- **Subscribe/Unsubscribe:** Beim ersten Start des XRn bekommt man eine Liste aller auf dem Server existierenden Newsgroups, z. Zt. sind dies etwas über 3000. Da man diese wohl kaum alle lesen will, sollte man die in der oberen Hälfte angezeigten Gruppen die man regelmäßig lesen möchte, mit dem linken Mausknopf markieren und mir den darunterliegenden Buttons „Add first“ als wichtigste Gruppen markieren (werden ab sofort zuerst angezeigt), mit „Add last“ setzt man sie als letzte. Mit „Quit“ kann man diese Auswahl beenden.

Möchte man später neue Gruppen subscriben, so kann man dies mit „All groups“ und dort mit „Subscribe“ machen.

Will man Gruppen, die man regelmäßig liest, abbestellen, so markiert man sie mit der Maus (oberes Fenster) und bestätigt mit „Unsubscribe“.

- **Newsgroup anwählen:** Die in der oberen Fensterhälfte angezeigten Newsgroups können mit Hilfe des links angebrachten Scrollbalkens durchgrblättert werden, durch Anklicken einer Gruppe mit der linken Maustaste wird die entsprechende Gruppe zum Lesen ausgewählt.
- **Artikel anwählen:** Nachdem eine Gruppen zum Lesen angewählt wurde, werden im oberen Fenster die einzelnen Artikel der Gruppe aufgezeigt. Dabei kann wieder der Scrollbalken verwendet werden. Wird ein Artikel durch Druck auf die mittlere Maustaste

angewählt, wo wird er in der unteren Fensterhälfte angezeigt. Zum Scrollen des Artikels existiert wiederum ein eigener Scrollbalken.

- **Artikel speichern:** Wird ein Artikel in der unteren Fensterhälfte angezeigt, so kann er durch einen Mausklick auf „Save“ abgespeichert werden.
- **Alte Artikel lesen:** Will man bereits gelesene Artikel nochmals lesen, so kann man sie mit „List old“ einblenden.
- **Catch up:** Um alle Artikel einer Gruppe als gelesen zu markieren, kann man den Button „Catch up“ drücken.
- **Posten:** Um einen eigenen Artikel abzusenden, klickt man auf „Post“. Es wird ein eigenes Fenster für einen Editor aufgemacht, in dem der Artikel-Rumpf vorgegeben ist. In den Headerzeilen ist zumindest das Thema (Subject) anzugeben, siehe hierzu auch [8]. Wenn im Home-Directory eine Datei `.signature` existiert, so wird ihr Inhalt angehängt.

Wenn man dann seinen Artikel zwischen Header und Signature eingefügt hat (evtl. fertigen Artikel mit „compIncludeFile“ einfügen), dann kann man ihn mit „compSend“ abschicken. Mit „compAbort“ kann abgebrochen werden.

- **Followup:** Will man auf einen Artikel antworten, so gibt es zweierlei Möglichkeiten: Entweder man postet ein Followup, oder man antwortet dem Autor privat, per E-Mail.

Dies geht mit den Buttons am unteren Fensterrand. Mit „Reply“ schickt man seine Antwort per E-Mail, „Followup“ gibt die Antwort über das UseNet weiter, so daß sie auch von anderen Leuten gelesen werden kann.

- **Verlassen:** Mit „Quit“ kann man XRn verlassen.

### **3.3 Andere Betriebssysteme**

Newsreader für MS-DOS und VMS sind in Vorbereitung und werden bei Bedarf auf dem PD-Server `ftp.uni-regensburg.de` abgelegt.



## 4 Server-Installation

### 4.1 Komponenten des News-Servers

#### 4.1.1 Die Hardware: Sun SPARCstation 10

Der Rechner für den News-Server ist eine eigens für diese Aufgabe dedizierte Sun SPARCstation 10, Modell 30. Die interne Festplatte mit 500MB wurde durch eine zweite externe Festplatte mit 1GB ergänzt. Eine nähere Beschreibung dieses Rechnertypen findet man in [18], [12] und [20].

Der Rechner ist mit dem standardmäßig vorhandenen Ethernet-Anschluß an das Campus-weite Ethernet-Backbone angeschlossen. Dies ermöglicht zum einen Verbindungen „nach außen“, um News von anderen Rechnern zu empfangen, desweiteren ermöglicht der Campus-weite Zugriff das Lesen und Posten von News von beliebigen Workstations aus. Ausserdem werden Backups ebenfalls über das Netzwerk auf das Bandlaufwerk der `rrzc1` gemacht.

#### 4.1.2 Das Betriebssystem: SunOS 4.1.3

Der Rechner läuft unter dem Betriebssystem SunOS 4.1.3 (=Solaris 1.1). Solaris 2.1 ist zwar zum Zeitpunkt der Installation bereits verfügbar, aufgrund der Instabilität des Systems wird es aber nicht für den Server-Betrieb verwendet, siehe hierzu auch [19]. Allgemein soll der Umstieg auf Solaris 2.x an der Uni Regensburg nicht vor dem Erscheinen von Solaris 2.2 geschehen.

#### 4.1.3 Die Software: InterNetNews (INN)

Als News-Software wird „InterNetNews“ (INN) verwendet. Es besteht aus einem Usenet-Transportsystem sowie dem Verwaltungssystem für Unix-Systeme, auf denen NNTP<sup>5</sup> für den meisten Usenet-Verkehr eingesetzt wird.

Neben INN existieren noch zwei weitere Usenet-Pakete. B News, auch als B2.11 bekannt, ist die älteste Implementierung und besitzt einige Unzulänglichkeiten, die sich besonders in I/O-Engpässen bei hohem News-Aufkommen äußern. Das 1987 erschienene C News umgeht dies, indem jeweils mehrere Artikel auf einmal verarbeitet werden und so die Verwaltungs- und Protokoll-Dateien im Speicher gehalten werden können.

Als Transportsystem für die News-Artikel verwenden beide Pakete UUCP (Unix to Unix Copy). Dieses ist für den „Store and Forward“-Betrieb, d. h. für den Austausch über Modem ein- bis zweimal am Tag, gedacht. Die Natur des Internet erlaubt es jedoch, Artikel quasi in Echtzeit weiterzuleiten. Dazu eignet sich das NNTP-Protokoll wesentlich besser, es wurde eigens für diesen Einsatz konzipiert. Zur Zeit existiert nur eine frei verfügbare Implementierung des NNTP-Protokolls für Unix, sie ist unter dem Namen „nntp“ bekannt (was wohl etwas Verwirrung schafft: niemand wird `sendmail SMTP` nennen).

Die oben genannten News-Systeme können nun alternativ auf UUCP oder NNTP aufsetzen. Um

---

<sup>5</sup>NetNews Transfer Protocol, in RFC 977 definiert.

dem Echtzeit-Anspruch gerecht zu werden, muß aber wieder jeder ankommende Artikel einzeln verarbeitet werden, was den Performance-Vorteil von C News wieder zunichte macht. Dies und andere Gründe führten zur Implementierung einer NetNews-Software speziell für Rechner am Internet: InterNetNews.

INN besteht aus den folgenden Komponenten:

- *Innd* ist hauptsächlich für neu eintreffende News-Artikel zuständig.
- *innxmit* liest die Dateien mit den weiterzuleitenden Artikeln und bietet diesen den News-feeds an.
- *ctlinnd* steuert den *Innd*.
- *nnrpd* ist ein NNTP-Server für Newsreader-Clients.

Diese 4 Teile bilden zusammen die Transport- und Verwaltungseinheit, mit der ein kompletter, eigenständiger News-Server betrieben werden kann. Eine ausführliche Beschreibung der einzelnen Komponenten findet man in [16].

INN wird in den Newsgroups `NEWS.SOFTWARE.NNTP` und `NEWS.SOFTWARE.B` diskutiert. Außerdem können über die Mailing-List `nnntp-managers@colossus.apple.com` weitere NNTP-Experten erreicht werden. Für eine Aufnahme in die Liste muß eine entsprechende Nachricht an `nnntp-managers-request@colossus.apple.com` geschickt werden.

## 4.2 Vorbereiten des Rechners

Hier sollen kurz die Schritte aufgeführt werden, die nötig sind, um den Rechner hard- und softwaremäßig auf die Installation vorzubereiten bzw. für den News-Betrieb benötigte Programme zu installieren. Die Herkunft der einzelnen Programme ist in Anhang D auf Seite 83 aufgeführt.

- Diverse Hilfsprogramme installieren:
  - GNU `egrep`: Wegen Bugs im `grep/egrep` von Sun muß auf diese — ohnehin leistungsstärkere — Variante zurückgegriffen werden.
  - `perl`: Wird zum kontrollieren der Config-Dateien benötigt, außerdem für diverse Hilfsprogramme *die* Programmiersprache. Sollte IMHO auf jedem Unix-Rechner installiert werden. [14] gilt als Standardwerk zu perl.
  - `patch`: Um den Source-Code diverser Programme auf den neuesten Stand zu bringen.
- Die 1GB-Platte mit zwei Partitionen versehen:
  1. Spool-Partition: sie soll ca. 900 MB umfassen, auf ihr werden die News-Artikel gespeichert werden. Sie muß auf `/var/spool/news` gemountet werden, siehe auch [17] auf Seite 12.
  2. Local-Partition: ca. 100 MB, hauptsächlich für News-Software und temporäre Dateien. Diese muß auf einer eigenen Partition untergebracht sein, da sonst bei voller Spool-Platte keine Artikel gelöscht (expired) werden können. Die Partition wird auf `/usr-/local/news` gemountet, die 100MB sollten auch für Spool-Platten > 1GB ausreichen.
- Auf dem Root-Filesystem werden ca. 10 MB für temporäre Daten benötigt.



- Die Umgebungsvariable „inn“ muß auf die INN-Sourcen zeigen: „setenv inn \$HOME/inn-1.2“

### 4.3 Installation von INN 1.2

Im Folgenden werden die Schritte beschrieben, die nötig sind, um einen News-Server mit INN V1.2 zu installieren. Seitenverweise beziehen sich auf die mitgelieferte Dokumentation [17].

Auf dem News-Server sind im einzelnen die folgenden Schritte nötig, um INN zum laufen zu bringen:

1. User „news“ mit Gruppe „news“ anlegen.
2. **GANZ WICHTIG!** In `$inn/config/files.list` eine Zeile mit dem Eintrag „`../BUILD`“ ergänzen, da sonst dort die Pfade nicht richtig gesetzt werden!!!
3. C-Subst übersetzen und testen ( $\Rightarrow$ [17] S. 5):
 

```
cd $inn/config
cp config.dist config.data
make c quiet
```
4. System konfigurieren: `$inn/config/config.data lt.` [17], S. 18 (Solaris 1.x bzw. SunOS 4.1.3)
5. Übersetzen („more cautious“,  $\Rightarrow$ [17], S. 21)
 

```
cd $inn/config
make quiet
cd ..
```
6. Library ( $\Rightarrow$  [17] S. 22)
 

```
cd $inn/lib
make libinn.a lint
make install
```
7. Frondends ( $\Rightarrow$  [17] S. 23)
 

```
cd $inn/frontende
make all
```
8. Innd ( $\Rightarrow$  [17] S. 23)
 

```
cd $inn/innd
make all
```
9. NetNews Reading Daemon ( $\Rightarrow$  [17] S. 24)
 

```
cd $inn/nnrpd
make all
```
10. Backend Programs ( $\Rightarrow$  [17] S. 24)

```
cd $inn/backends
make all
```

11. Expire (⇒ [17] S. 25)

```
cd $inn/expire
make all
```

12. Script and data files (⇒ [17] S. 26)

```
cd $inn/site
make all
```

13. Installing the System (⇒ [17] S. 27), Verzeichnisse anlegen, Binaries und Scripts installieren, vorher `makedirs.sh` überprüfen:

```
cd $inn
su root
makedirs.sh
make update
```

14. Konfigurationsdateien!

```
cd $inn/site
```

- Folgende Dateien prüfen: `checkgroups`, `rmgroup`, `default`, `sendme`, `docheckgroups`, `sendsys`, `ihave`, `senduname`, `newgroup`, `version`, `parsecontrol`, `writelog`
- Folgende Dateien durchsehen: `innlog.awk`, `scanlogs`, `innstat`, `news.daily`, `tally.-control`, `rc.news`, `tally.unwanted`
- Folgende Dateien *müssen* angepaßt werden:
  - `control.ctl`: u.a. Tabelle aus `newslog(5)`, Seite 1 übertragen
  - `expire.ctl`: Hier eintragen, wie lange die einzelnen Newsgroups gehalten werden sollen/können.
  - `nntpsend.ctl`
  - `hosts.nntp`
  - `inn.conf`
  - `moderators`: Verzeichnis der moderierten Newsgroups und der zugehörigen Moderatoren. Kann einem entsprechenden Posting in `NEWS.ANSWERS` entnommen werden.
  - `newsfeeds`: Darf keine Leerzeilen enthalten!
  - `nntp.access`
  - `passwd.nntp`
- Die Konfigurationsdateien können mit dem perl-Script `inncheck` auf Korrektheit überprüft werden, siehe `news.daily(8)`!

15. `make install`

16. Folgendes in `/etc/syslog.conf` eintragen, siehe auch `$inn/syslog/syslog.conf`:

```

news.err      /var/log/news/news.err
news.crit     /var/log/news/news.crit
news.crit     *
news.crit     /dev/console
news.notice   /var/log/news/news.notice

```

17. „news.daily“ täglich (über „cron“) als Benutzer „news“ laufen lassen.
18. „nntpsend“ alle 10 Minuten (über „cron“ starten, damit Nachrichten zu den Newsfeeds übertragen werden.
19. `/usr/local/etc/inndstart` sollte `suid root` gesetzt werden
20. „active file“ und „history database“ mit BUILD-Script anlegen. Die dabei gestellten Fragen sind wie hier angegeben zu beantworten:
  - Use C version of subst? *y*
  - Have you already built the executables? *y*
  - Do directories exist? *y*
  - Continue with installation? *y*
  - Start subshell? *n*
21. `innwatch` evtl. in `rc.news` eintragen
22. `rc.news` installieren: aus `/etc/rc.local` heraus aufrufen ( $\Rightarrow$  [17] S. 11)
23. That's it!

## 4.4 Lokale Erweiterungen und Ergänzungen

Um diverse Listen etc. zu verwalten und Statistiken zu erstellen, wurde die News-Software durch ein paar selbstgeschriebene Programme ergänzt.

Da es sich bei den anfallenden Aufgaben ausschließlich um „Textverarbeitung“ handelt, ist die Programmiersprache „perl“<sup>6</sup> geradezu prädestiniert für diese Anwendung.

### 4.4.1 Newsgroups anlegen

Da beim ersten Anlauf des Servers nicht bekannt war, welche Newsgroups empfangen werden würden, mußte aus der Datei `/var/log/news/news.notice` ermittelt werden, welche Artikel in nicht-existierende Groups gehörten, diese werden von INN als „unwanted“ klassifiziert. Das perl-Script `mk-unwanted-online` liest die Datei `news.notice` von der Standardeingabe und gibt die entsprechenden Shell-Kommandos aus, die nötig sind, um die jeweilige Gruppe anzulegen.

Damit die Datei `news.notice` nicht nur einmal durchgelesen wird, sondern die dauernd angehängten Einträge verarbeitet werden, wird sie über „`tail -f`“ in `mk-unwanted-online` gepiped:

```
% tail -f /var/log/news/news.notice | mk-unwanted-online | csh
```

---

<sup>6</sup>Practical Extraction and Report Language, siehe [14]

Hier das gesamte perl-Script zu mk-unwanted-online:

```
#!/bin/perl

#
# mk-unwanted-online
#
# Legt Groups an, die in /var/log/news/news.notice als 'unwanted'
# markiert werden. Aufruf:
#
# tail -f /var/log/news/news.notice| mk-unwanted-online | csh
#
while(<>){
    if(/Unwanted newsgroup +\"(.+)\"){
        chop($date='date');
        print STDERR "$date: $1\n";
        print "/usr/local/news/bin/ctlinnd newsgroup $1 y >&/dev/null\n";
    }
}
```

Die ersten News aus Erlangen erreichten uns am Freitag Nachmittag, dieses Script lief daraufhin das ganze folgende Wochenende, um die Artikel nicht zu verlieren. Am folgenden Montag wurden die Gruppen dann gemäß Postings aus NEWS.ANSWERS angelegt, die die aktuell existierenden Gruppen beschrieben.

Ab und an geschieht es immer noch, daß Artikel zu einer Gruppe ankommen, die zwar schon lange existiert, aber noch nicht angelegt wurde. Dies wird auch in der Datei /var/log/news/unwanted.log mitprotokolliert. Das Script mk-unwanted legt die dort verzeichneten Newsgroups an:

```
#!/usr/bin/perl

#
# mk-unwanted
#
# Legt die in "/var/log/news/unwanted.log" gemeldeten Newsgroups
# an. Aufruf:
#
# mk-unwanted | csh
#
open(U,"/var/log/news/unwanted.log") || die " Can't open unwanted.log: $!\n";

while($line=<U>){
    $line =~ s/[^\n]* //;
    chop($line);
    print STDERR "/usr/local/news/bin/ctlinnd newsgroup $line y\n";
    print STDOUT "/usr/local/news/bin/ctlinnd newsgroup $line y >&/dev/null\n";
}
```

#### 4.4.2 Vorhandene Newsgroups ausdrucken: active2ps

Sehr bald interessierte dann, welche Newsgroups überhaupt vorhanden waren bzw. zu welchen Gruppen News gehalten wurden. Dazu existiert im INN-Paket eine Datei `/usr/local/news/active`, das sog. Active-File. Diese ASCII-Datei enthält neben dem Namen jeder Gruppe zusätzliche Verwaltungsinformationen, hier ein kurzer Auszug:

```
alt.binaries.sounds.misc 0000001552 0000001147 y
alt.bitterness 0000000094 0000000074 y
alt.bob-packwood.tongue.tongue.tongue 0000000004 0000000005 y
alt.bogus.group 0000000005 0000000006 y
alt.bonsai 0000000156 0000000124 y
alt.books.anne-rice 0000000215 0000000196 y
```

Da dies natürlich keinem Anwender zugemutet werden kann, wurde das folgende Script `active2ps` geschrieben, das das Active-File in Form einer PostScript-Datei aufbereitet. Auch dies ließ sich mit perl sehr schön bewerkstelligen, der PostScript-Output entspricht übrigens den Adobe-Comment-Conventions V2.0:

```
#!/usr/bin/perl

#
# active2ps
#
# Gibt Liste aller Newsgroups im PS-Format aus.
#

$active      = "/usr/local/news/active";
$output      = "active.ps";

$creator     = "$0 by Hubert Feyrer (feyrer@rrzc1.rz.uni-regensburg.de)";
$user       = "$ENV{'USER'}";
$date       = 'date'; chop($date);
$origin_x   = 20;
$origin_y   = -20;
$page_x     = 800;
$page_y     = 550;
$borderwidth = 3;
$font_hdr   = 'Times-Roman';
$font_txt   = 'Times-Roman';
$size_hdr   = 25;
$size_txt   = 7;
$page      = 1;

$llx        = $origin_x+$borderwidth/2;
$lly        = $origin_y+$borderwidth/2;
$urx        = $origin_x+$page_x-($borderwidth/2);
$ury        = $origin_y+$page_y-($borderwidth/2);
```

```

$cols      = 4;
$colwidth  = ($urx-$llx-2*$borderwidth)/$cols;

$xmid      = $llx+($urx-$llx)/2;
$yup       = $ury-1.4*$size_hdr;
$pg_x      = $page_x+$origin_x;
$pg_y      = $page_y+$origin_y;

$x         = -1;
$y         = -1;

$PSprolog  = <<END
%!PS-Adobe-2.0
%%Title: Active-file
%%Creator: $creator
%%CreationDate: $date
%%For: $user
%%Pages: (atend)
%%BoundingBox: 0 0 $pg_y $pg_x
%%DocumentFonts: $font_hdr $font_txt
%%DocumentNeededFonts: $font_hdr $font_txt
%%EndComments

/font_hdr
  ($font_hdr) findfont
  $size_hdr scalefont
def

/font_txt
  ($font_txt) findfont
  $size_txt scalefont
def

/Portrait {
  grestore
} bind def

/Landscape {
  gsave
  $page_y 0 translate
  90 rotate
} bind def

/Border {
  dup

```

```

    $borderwidth setlinewidth
    $llx $lly moveto
    $urx $lly lineto
    $urx $ury lineto
    $llx $ury lineto
    closepath
    stroke

    font_hdr setfont
    stringwidth pop 2 div
    $xmid exch sub
    $ury $size_hdr sub
    moveto
    show

    $llx $yup moveto
    $urx $yup lineto
    stroke
} bind def

%%EndProlog
END
;

#####

open(OUT,">$output") || die "Can't write to $output: $!\n";
print OUT $PSprolog;

open(A,"sort $active |") || die;

print OUT <<END

%%Page: $page $page
Landscape
(Vorhandene Newsgroups, Seite $page) Border
font_txt setfont
END
;

$x=$llx+1.5*$borderwidth;
$y=$yup-2*$size_txt;
while($group=<A>){
    $group =~s/ .*$/g;
    chop($group);
    print OUT "$x $y moveto ($group) show\n";
}

```

```

$y-=$size_txt;
if($y < $origin_y+$size_txt){
    $x+=$colwidth;
    $y=$yup-2*$size_txt;

    if($x>$urx-0.5*$colwidth){
        $x=$llx+1.5*$borderwidth;
        $page++;
        print OUT "showpage\n\n\n";

        print OUT "%Page: $page $page\n";
        print OUT "Landscape\n";
        print OUT "(Vorhandene Newsgroups, Seite $page) Border\n";
        print OUT "font_txt setfont\n";
    }
}
}
print OUT <<END
showpage

%%Trailer
%%Pages: $page 1
END
;

close(A);
close(OUT);

```

Ein aktueller Ausdruck aller vorhandenen Newsgroups liegt diesem Bericht bei.

#### 4.4.3 Volumen-Statistik mit volume

Von mehreren Leuten wurde gefragt, wieviel News pro Tag ankämen und wie sich diese auf die einzelnen Newsgroups verteilen. Auch dies zu ermitteln ist kein größeres Problem. Da die Newsgroup-Hierarchie 1:1 auf das Dateisystem abgebildet wird, kann hier mit dem „du“-Befehl gearbeitet werden.

So kann nicht nur ermittelt werden, welches News-Volumen in den einzelnen Gruppen ankommt, sondern auch, wie die einzelnen Top-Level-Hierarchien etc. gewichtet sind und entsprechend, wo die meisten News laufen. Dies kann dann sortiert als Hitparade verwendet werden.

Da der „du“-Befehl aber nur angibt, wieviel Daten (in kB) absolut in einer Gruppe sind, ist hier jeweils die Differenz zwischen zwei Tagen zu nehmen: Nachdem um Mitternacht alte News gelöscht werden, wird der „du“-Befehl auf das Spool-Directory angewandt. Tagsüber sammeln sich dann die News in den einzelnen Gruppen an. Bevor um Mitternacht wieder die alten News gelöscht werden, wird nochmals der „du“-Befehl eingesetzt, die Differenz zum Morgen gibt dann



an, wieviele kB News in der einzelnen Gruppe an diesem Tag ankamen.

Dies alles wird im Rahmen des Shell-Scripts `news.daily` erledigt, das auch für das Löschen der alten Artikel sorgt und über „cron“ aufgerufen wird: Zu Beginn des Scripts wird „`volume stop`“ aufgerufen, nach dem Löschen wird mit „`volume start`“ der aktuelle Stand festgehalten.

Nachdem bei „`volume stop`“ der „`du`“-Befehl das zweite mal gelaufen ist, werden die Differenzen zum Morgen errechnet und in einer Datei im Verzeichnis `/var/log/news` abgelegt. Zusätzlich wird nur das Gesamtvolumen hinten an die Datei `/var/log/news/volume.log` angehängt.

Hier nun das vollständige perl-Script „`volume`“:

```
#!/bin/perl

#
# volume
#
# Statistik ueber taegliches News-Volumen erstellen. Nach dem taeglichen
# expire in 'news.daily' wird mit 'du' die Groesse der einzelnen News-
# groups ermittelt. Vor dem naechsten 'news.daily' wieder, die Differenz
# ist dann das Datenvolumen der einzelnen Gruppen.
#
#

$logdir="/var/log/news";           # Pfad fuer Log-Dateien
$tmp1="$logdir/volume.tmp1";      # Spoolgroesse zu Beginn des Tages
$tmp2="$logdir/volume.tmp2";      # Gruppe: Volumen
$tmp3="$logdir/volume.tmp3";      # Mail-Text
$vollog="$logdir/volume.log";     # Format: jjjjmmtt.hhmmss <size/k>
$spool="/news/spool/news";        # Spool-Directory, nicht SLink!!!
$newsadmin="news@news.uni-regensburg.de"; # Zieladresse fuer output
$top=30;                           # MaxNetLoaders

### DEBUG-WERTE
#$logdir="log";                    # Pfad fuer Log-Dateien
#$tmp1="$logdir/volume.tmp1";     # Spoolgroesse zu Beginn des Tages
#$tmp2="$logdir/volume.tmp2";     # Gruppe: Volumen
#$tmp3="$logdir/volume.tmp3";     # Mail-Text
#$vollog="$logdir/volume.log";    # Format: jjjjmmtt.hhmmss <size/k>
#$spool="Spool";                  # Spool-Directory, nicht SLink!!!
#$newsadmin="news@news.uni-regensburg.de"; # Zieladresse fuer output
#$top=5;                           # MaxNetLoaders
### DEBUG-WERTE

%months = ("Jan", "01",
           "Feb", "02",
           "Mar", "03",
           "Apr", "04",
           "May", "05",
```



```

        if($d!=0 || $group!~/\./){
            print TMP2 "$group: $d k\n";
        }
        if($group!~/\./){
            $sum+=$d;
        }
    }
close(DU);
close(TMP2);

# Ausgaben an NewsAdmin mailen
system("echo >>$tmp3");
system("echo 'News-volume between <$d1> and <$d2>: $sum k' >>$tmp3");
system("echo >>$tmp3");
system("echo >>$tmp3");
system("echo 'stop top-volume Newsgroups:' >>$tmp3");
system("echo '-----' >>$tmp3");
system("echo >>$tmp3");
system("sort -nr +1 $tmp2 | head -$top >>$tmp3");
system("echo >>$tmp3");
system("echo >>$tmp3");
system("echo 'Volume per Newsgroups:' >>$tmp3");
system("echo '-----' >>$tmp3");
system("echo >>$tmp3");
system("cat $tmp2 >>$tmp3");

# In der folgenden Zeile den Kommentar entfernen, damit die
# Statistik zusammen mit dem Daily Usenet Report verschickt wird:
#system("mail -s 'News-Volumen: $sum k' $newsadmin <$tmp3");

$date2=~/(\\w+) +(\\w+) +(\\d+) +(\\d\\d):(\\d\\d):(\\d\\d) +(\\w+) +(\\w+) +(\\d\\d\\d\\d)/;
$d=sprintf("%04d%02d%02d.%02d%02d%02d", $9, $months{$2}, $3, $4, $5, $6);
system("mv $tmp3 $logdir/volume-$d-$1");

system("echo '$d $sum k' >>$vollog");

unlink($tmp1);
unlink($tmp2);

exit 0;
}

print "Usage: $0 (start|stop)\n";
exit 1;

```

Hier ein Beispiel des Volumen-Überblicks, der in der Datei (`volume.log`) abgelegt wird:

19930428.000001	60479	k
19930429.000002	78163	k
19930430.000001	67069	k
19930501.000001	77226	k
19930502.000001	45803	k
19930503.000001	34313	k
19930504.000001	69756	k
19930505.000001	76283	k
19930506.000002	78570	k
19930507.000002	71915	k
19930508.000001	79126	k
19930509.000001	41179	k
19930510.000001	36692	k
19930511.000003	38553	k
19930512.000001	85256	k
19930513.000001	78576	k
19930514.000001	69654	k
19930515.000002	72204	k
19930516.000001	41172	k
19930517.000002	36027	k
19930518.000001	72219	k
19930519.000001	76396	k
19930520.000002	90310	k
19930521.000001	75542	k

Es wird das Datum und die Uhrzeit sowie das News-Volumen in kB angezeigt.

Zwei Beispiel der täglichen Volumens-Statistik, vom 20.5.1993 und 21.5.1993, liegen diesem Bericht bei. Aus der ersten Zeile läßt sich erkennen, daß der „du“-Befehl und das Löschen der alten Artikel ca. von 00:00 bis 01:46 gedauert hat.

## 5 Compilieren der Clients

Neben der News-Server-Installation wurden mehrere Clients installiert und ausprobiert, aus denen dann TIN und XRn als empfohlene Newsreader hervorgingen.

Hier soll noch auf ein paar Punkte eingegangen werden, die beim Übersetzen dieser Clients zu beachten waren bzw. welche Änderungen und Anpassungen nötig waren.

### 5.1 GNU grep

Dies ist zwar kein Newsreader, da es aber bei der Installation trotzdem ein kleines Problem gegeben hat, soll dies hier kurz erwähnt werden:

Nach „./configure“ muß im **Makefile** bei **DEFS** noch „-Dconst=“ angefügt werden, da der von Sun ausgelieferte C-Compiler nicht dem ANSI-Standard entspricht.

### 5.2 GNUS

Die im Archiv enthaltenen Lisp-Dateien müssen mit dem beiliegenden **Makefile** erst noch übersetzt werden, da sich dies positiv auf die Arbeitsgeschwindigkeit auswirkt. Da GNUS bei Lucid-Emacs bereits enthalten ist, soll hier kurz auf die Installation unter GNU Emacs<sup>7</sup> eingegangen werden, folgende Schritte sind nötig:

```
# make
# cp *.el *.elc /usr/local/emacs/lisp
# cp *.info /usr/local/emacs/info
```

In der Datei `/usr/local/emacs/info/dir` ist dann noch die folgende Zeile zu ergänzen:

```
* GNUS: (gnus)      NNTP-based Newsreader for GNU- and Lucid-Emacs.
```

Werden die Lisp-Files in einem privaten Verzeichnis installiert, so muß noch die Umgebungsvariable „EMACSLOADPATH“ gesetzt werden, damit die Dateien gefunden werden.

In jedem Fall muß die GNUS-Bibliothek dann (am besten im `.emacs`) mit `(load-library "gnus")` nachgeladen werden. Der Aufruf erfolgt dann interaktiv über `M-x gnus` oder mit dem folgenden `csh`-Alias:

```
% alias gnus      "emacs -f gnus"
```

### 5.3 nn

Als Config-File ist `config.h-sparc` zu wählen, die folgenden Variablen sind dann zu setzen:

- `NNTP_SERVER: /usr/local/lib/nntpserver`

---

<sup>7</sup>The one and only!

- BIN\_DIRECTORY: /usr/local/bin
- LIB\_DIRECTORY: /usr/local/lib
- DB\_DIRECTORY: /var/spool/nn
- INEWS\_PATH: /usr/local/bin/inews
- USR\_MAN\_DIR: /usr/local/man/man1
- SYS\_MAN\_DIR: /usr/local/man/man1
- DAEMON\_MAN\_DIR: /usr/local/man/man8

Nach dem übersetzen mit „make all“ ist die nn-eigene Datenbank wie folgt zu erzeugen:

```
# ./init s
# ./init INIT
```

Der nn-Dämon `nnmaster` muß immer laufen, er wird am besten über „cron“ alle 10 Minuten mit „/usr/local/bin/nnmaster -l -r -C“ aufgerufen; „nnmaster“ aktualisiert dann ständig die eigene Datenbank.

## 5.4 rn

Die Datei `/usr/local/lib/nntpserver` muß den Namen des NNTP-Servers enthalten: `news.uni-regensburg.de`.

Für das Übersetzen muß dann außerdem noch der INN-Quellcode und die `libinn.a` vorhanden sein, im INN-Quellcode muß ein symbolischer Link von `$inn/common` nach `$inn/include` zeigen: „ln -s \$inn/include \$inn/common“.

Beim Aufruf von „Configure“ sind folgende Distribution Prefixes zu setzen:

- local org: `rz`
- organisation: `urgb`
- city: `regensburg`
- state/province: `by`
- country: `de`
- continent: `eunet`

Im Makefile muß dann noch der Wert von `CFLAGS` korrigiert werden:

```
CFLAGS = $(CFLAGS) \
-DCHAR_OK=NNTP_CLASS_OK \
-DCHAR_FATAL=NNTP_CLASS_FATAL \
-D_ERR_NOGROUP=NNTP_NOSUCH_GROUP_VAL
```

Beim Linken muß dann entsprechen mit `libinn.a` anstatt von `clientlib.o` gelinkt werden.

## 5.5 TIN

Im Makefile sind die folgenden Definitionen zu setzen:

```
-DNNTTP_ONLY
-DNNTTP_INEWS
-DNO_NNTP_EXTS
-DNNTTP_INEWS_DOMAIN=\"uni-regensburg.de\"
-DNNTTP_SERVER_FILE=/usr/local/lib/nntpserver
```

Dieser Newsreader enthält eigene NNTP-Routinen, der Server-Code wird nicht benötigt. Übersetzt wird anschließend mit „make bsd“.

## 5.6 XRn

Hier wird wiederum die `libinn.a` benötigt, außerdem ist beim Linken darauf zu achten, daß die richtigen X-Bibliotheken (MIT-X11/OpenWindows) gelinkt werden, da dies sonst zu Fehlern beim Linken und zur Laufzeit führen kann (keine Scrollbalken, ...).

## 5.7 XVnews

In der Header-Datei `xvnews.h` müssen die folgenden `#defines` gesetzt werden:

```
#define ORGANISATION "University of Regensburg, Germany"
#define NNTPSERVER "news.uni-regensburg.de"
#define DOMAIN ".uni-regensburg.de"
```

Übersetzt wird dann mit „make“, nicht mit „xmkmf“!!!





## 6 Client-Installation

Hier wird beschrieben, wie sich jedermann die fertig compilierten Binärdateien der Newsreader TIN und XRn auf seiner Unix-Workstations von SUN, DEC oder SGI installiert kann.

Die Versionen für die einzelnen Betriebssysteme sind am PD-Server <ftp.uni-regensburg.de> im Verzeichnis `/freeware/software/news/install` in den folgenden Dateien vorhanden:

Rechner, Betriebssystem	Dateiname+Pfad
SparcStation, SunOS 4.1.x	SunOS.tar.Z
SparcStation, Solaris 1.x	SunOS.tar.Z
SparcStation, Solaris 2.x	Solaris.tar.Z
Silicon Graphics, Irix	Irix.tar.Z
DEC, Ultrix	Ultrix.tar.Z
MS-DOS	MSDOS.lha
VMS	VMS.BCK

Diese gepackten Binärdateien sind mit „tar“ und „uncompress“ zu entpacken, z. B.:

```
# zcat SunOS.tar.Z | tar xf -
```

Jedes Archiv enthält in einem Verzeichnis die folgenden Dateien:

```
README..... Kurze Erläuterung (Wird beim Installieren angezeigt).
tin..... Ausführbare Datei für TIN (Kommt nach /usr/local/bin/tin).
xrn..... Ausführbare Datei für XRn (Kommt nach /usr/local/bin/xrn).
inews ..... Wird für XRn zum Posten benötigt (Kommt nach /usr/local/bin/inews).
XRn.ad..... Voreinstellungen für XRn (Kommt nach /usr/lib/X11/app-defaults/XRn).
tin.1 ..... Man-Page für TIN (Kommt nach /usr/local/man/man1).
xrn.1 ..... Man-Page für XRn (Kommt nach /usr/local/man/man1).
install.sh . Installationscript
```

Da die Dateien in Systemverzeichnisse kopiert werden, muß die eigentliche Installation unter Root-Kennung erfolgen. Nach dem „su“-Befehl einfach das Installationsscript aufrufen. Fertig!

Nach geglückter Installation kann das gesamte `news`-Verzeichnis gelöscht werden, es wird nicht mehr benötigt.

An dieser Stelle sollte nochmals darauf hingewiesen werden, daß jeder Benutzer vor dem Start der Newsreader ein paar Umgebungsvariablen setzen muß, siehe Beschreibung von TIN und XRn auf Seite 11 bzw. 15.



## 7 Lokale Newsgroups

Momentan sind die folgenden lokalen Newsgroups angelegt, Namen, Struktur, ...sind vorerst für den Testbetrieb, Anregungen hierzu sind dringendst erwünscht!

- `UNI-REGENSBURG.TEST`: Der Name sagt eigentlich alles. Hier können Tests gemacht werden um z. B. zu sehen, welche Header erzeugt werden, ob die Signature korrekt (nicht zweimal) angehängt wurde, ...
- `UNI-REGENSBURG.TEST.MOD`: Dies ist eine moderierte Test-Gruppe, der Moderator ist Hubert Feyrer (`feyrer@rrzc1.rz.uni-regensburg.de`). In moderierte Groups gepostete Artikel werden vom Moderator sortiert und dann als Zusammenfassung in der Newsgroup abgelegt.
- `UNI-REGENSBURG.AKTUELL`: Enthält aktuelle Ereignisse aus allen Bereichen der Uni Regensburg. Wer was interessantes weiß, soll es hierher posten. Diese Gruppe sollte DAS schwarze Brett der Uni Regensburg sein.
- `UNI-REGENSBURG.PROBLEME`: Probleme jeder Art und deren Beseitigung. Daß da momentan nur computer-bezogene Fragen drinstehen, soll niemand abhalten auch über was anderes zu schreiben!
- `UNI-REGENSBURG.NETZWERK`: Networking an der Uni Regensburg.
- `UNI-REGENSBURG.MENSA`: Es ist geplant, hier den Speiseplan der Mensa und der Uni-Pizzeria zu veröffentlichen. (Die Gruppe ist moderiert, damit nicht jeder in aller Öffentlichkeit über schlechte Qualität, zu hohe Preise etc. meckern kann.)
- `UNI-REGENSBURG.INFOSYSTEME`: Zur Diskussion der Infomationssysteme an der Uni Regensburg: News, Gopher, ...

Diese Gruppen werden zur Zeit nur auf dem Campus der Uni Regensburg gehalten, gepostete Artikel verlassen den Campus nicht.

Wenn der Wunsch besteht, neue Gruppen anzulegen, die einer der bestehenden Hierarchien angehören sollen, dann sind die in der jeweiligen Hierarchie geltenden Regeln<sup>8</sup> einzuhalten. [4] beschreibt das Anlegen neuer Gruppen in den Hierarchien `DE.*` und `DE.ALT.*`. Dies muß allerdings in Absprache mit dem Rechenzentrum durchgeführt werden, bitte E-Mail an die News-Administratoren unter `news-admin@news.uni-regensburg.de`.

---

<sup>8</sup>RfD (Request for Discussion), CfV (Call for Vote), ...



## 8 Infos für EDV-Interessierte

Hinter diesem Titel verbergen sich Hinweise auf Newsgroups zu Betriebssystemen, mit denen der eine oder andere vielleicht arbeitet. Deshalb sollen hier gezielt für jedes Betriebssystem interessante Newsgroups aufgelistet werden.

### 8.1 Sun

comp.org.sug	Talk about/for the The Sun User's Group.
comp.sources.sun	Software for Sun workstations. (Moderated)
comp.sys.sun.admin	Sun system administration issues and questions.
comp.sys.sun.announce	Sun announcements and Sunergy mailings. (Moderated)
comp.sys.sun.apps	Software applications for Sun computer systems.
comp.sys.sun.hardware	Sun Microsystems hardware.
comp.sys.sun.misc	Miscellaneous discussions about Sun products.
comp.sys.sun.wanted	People looking for Sun products and support.
comp.unix.solaris	Discussions about the Solaris operating system.
comp.windows.news	Sun Microsystems' NeWS window system.
info.sun-managers	*Sun-managers digest (sun-managers@rice.edu). (Moderated)
info.sun-nets	*Sun-nets digest (sun-nets@umiacs.umd.edu). (Mod.)
umn.comp.sys.sun	Sun Workstations at the University of Minnesota.
umn.cs.system.sun	Discussion/Announcements vis-a-vis the U of MN CS Suns.
umn.local-lists.sun-news	??
umn.net-lists.sun-managers	The „Sun-managers“ mailing list.

### 8.2 Ultrix

comp.unix.ultrix	Discussions about DEC's Ultrix.
umn.local-lists.tc-ultrix-vms	??
vmsnet.networks.tcp-ip.ucx	DEC's VMS/Ultrix Connection product.

## 8.3 VMS

bit.listserv.vmslsv-l	VAX/VMS LISTSERV Discussion List.
comp.os.vms	DEC's VAX* line of computers & VMS.
gnu.emacs.vms	VMS port of GNU Emacs.
news.software.anu-news	VMS B-news from Australian National Univ.
umn.comp.os.vms	VMS operating system at the U of Minnesota.
umn.local-lists.tc-ultrix-vms	??
vmsnet.admin	Administration of the VMSnet newsgroups.
vmsnet.alpha	Discussion about Alpha AXP architecture, etc.
vmsnet.announce	General announcements of interest to all. (Mod.)
vmsnet.announce.newusers	Orientation info for new users. (Moderated)
vmsnet.decus.journal	The DECUServe Journal. (Moderated)
vmsnet.decus.lugs	Discussion of DECUS Local User Groups.
vmsnet.employment	Jobs sought/offered and related issues.
vmsnet.internals	VMS internals, etc., gatewayed to MACRO32 list.
vmsnet.mail.misc	Other electronic mail software.
vmsnet.mail.mx	MX email system, gatewayed to MX mailing list.
vmsnet.mail.pmdf	PMDF email system.
vmsnet.misc	General VMS topics not covered elsewhere.
vmsnet.networks.desktop.misc	Other desktop integration software.
vmsnet.networks.desktop.pathworks	DEC Pathworks desktop integration software.
vmsnet.networks.management.decmcc	DECmcc and related software.
vmsnet.networks.management.misc	Other network management solutions.
vmsnet.networks.misc	General networking topics not covered elsewhere.
vmsnet.networks.tcp-ip.cmu-tek	CMU-TEK TCP/IP package, gatewayed.
vmsnet.networks.tcp-ip.misc	Other TCP/IP solutions for VMS.
vmsnet.networks.tcp-ip.multinet	TGV's Multinet TCP/IP, gatewayed.
vmsnet.networks.tcp-ip.ucx	DEC's VMS/Ultrix Connection product.
vmsnet.networks.tcp-ip.wintcp	The Wollongong Group's WIN-TCP TCP/IP.
vmsnet.pdp-11	PDP-11 hard-/software, gatewayed to info-pdp11.
vmsnet.sources	Source code postings ONLY. (Moderated)
vmsnet.sources.d	Discussion about or requests for sources.
vmsnet.sources.games	Recreational software postings.
vmsnet.sysmgt	VMS system management.
vmsnet.test	Test messages.
vmsnet.tpu	TPU language and applications.
vmsnet.uucp	DECUS uucp software, gatewayed.
vmsnet.vms-posix	Discussion about VMS POSIX.

## 8.4 MS-DOS

<code>clari.nb.ibm</code>	Newsbytes IBM PC World coverage. (Moderated)
<code>comp.archives.msdos.announce</code>	Announcements about MSDOS archives. (Moderated)
<code>comp.archives.msdos.d</code>	Discussion of materials available in MSDOS archives.
<code>comp.binaries.ibm.pc</code>	Binary-only postings for IBM PC/MS-DOS. (Moderated)
<code>comp.binaries.ibm.pc.d</code>	Discussions about IBM/PC binary postings.
<code>comp.binaries.ibm.pc.wanted</code>	Requests for IBM PC and compatible programs.
<code>comp.os.msdos.4dos</code>	The 4DOS command processor for MS-DOS.
<code>comp.os.msdos.apps</code>	Discussion of applications that run under MS-DOS.
<code>comp.os.msdos.desqview</code>	QuarterDeck's Desqview and related products.
<code>comp.os.msdos.misc</code>	Miscellaneous topics about MS-DOS machines.
<code>comp.os.msdos.pcgeos</code>	GeoWorks PC/GEOS and PC/GEOS-based packages.
<code>comp.os.msdos.programmer</code>	Programming MS-DOS machines.
<code>comp.protocols.tcp-ip.ibmpc</code>	TCP/IP for IBM(-like) personal computers.
<code>comp.sys.ibm.pc.demos</code>	Demonstration programs which show programmer skill.
<code>comp.sys.ibm.pc.digest</code>	The IBM PC, PC-XT, and PC-AT. (Moderated)
<code>comp.sys.ibm.pc.games</code>	Games for IBM PCs and compatibles.
<code>comp.sys.ibm.pc.hardware</code>	XT/AT/EISA hardware, any vendor.
<code>comp.sys.ibm.pc.misc</code>	Discussion about IBM personal computers.
<code>comp.sys.ibm.pc.rt</code>	Topics related to IBM's RT computer.
<code>comp.sys.ibm.pc.soundcard</code>	Hardware and software aspects of PC sound cards.
<code>comp.sys.ibm.ps2.hardware</code>	Microchannel hardware, any vendor.
<code>misc.forsale.computers.pc-clone</code>	IBM PC related computer items.
<code>relcom.msdos</code>	MS-DOS software.
<code>umn.comp.sys.ibm</code>	IBM PC and compatibles at the University of Minnesota.

## 8.5 MS-Windows

<code>comp.binaries.ms-windows</code>	Binary programs for Microsoft Windows. (Mod.)
<code>comp.os.ms-windows.advocacy</code>	Speculation and debate about MS Windows.
<code>comp.os.ms-windows.announce</code>	Announcements relating to Windows. (Mod.)
<code>comp.os.ms-windows.apps</code>	Applications in the Windows environment.
<code>comp.os.ms-windows.misc</code>	General discussions about Windows issues.
<code>comp.os.ms-windows.programmer.misc</code>	Programming Microsoft Windows.
<code>comp.os.ms-windows.programmer.tools</code>	Development tools in Windows.
<code>comp.os.ms-windows.programmer.win32</code>	32-bit Windows programming interfaces.
<code>comp.os.ms-windows.setup</code>	Installing and configuring Microsoft Windows.

## 8.6 OS/2

bit.listserv.os2-1	OS/2 Discussions.
comp.binaries.os2	Binaries for use under the OS/2 ABI. (Moderated)
comp.os.os2.advocacy	Supporting and flaming OS/2.
comp.os.os2.announce	Notable news and announcements related to OS/2. (Moderated)
comp.os.os2.apps	Discussions of applications under OS/2.
comp.os.os2.beta	All aspects of beta releases of OS/2 systems software.
comp.os.os2.misc	Miscellaneous topics about the OS/2 system.
comp.os.os2.networking	Networking in OS/2 environments.
comp.os.os2.programmer	Programming OS/2 machines.
comp.os.os2.ver1x	All aspects of OS/2 versions 1.0 through 1.3.

## 8.7 Amiga

alt.sources.amiga	Technically-oriented Amiga PC sources.
alt.sources.amiga.d	Discussion of technically-oriented Amiga PC sources.
alt.sys.amiga.demos	Code and talk to show off the Amiga.
alt.sys.amiga.uucp	AmigaUUCP.
alt.sys.amiga.uucp.patches	Patches for AmigaUUCP.
bit.listserv.i-amiga	Info-Amiga List.
comp.binaries.amiga	Encoded public domain programs in binary. (Moderated)
comp.sources.amiga	Source code-only postings for the Amiga. (Moderated)
comp.sys.amiga.advocacy	Why an Amiga is better than XYZ.
comp.sys.amiga.announce	Announcements about the Amiga. (Moderated)
comp.sys.amiga.applications	Miscellaneous applications.
comp.sys.amiga.audio	Music, MIDI, speech synthesis, other sounds.
comp.sys.amiga.datacomm	Methods of getting bytes in and out.
comp.sys.amiga.emulations	Various hardware & software emulators.
comp.sys.amiga.games	Discussion of games for the Commodore Amiga.
comp.sys.amiga.graphics	Charts, graphs, pictures, etc.
comp.sys.amiga.hardware	Amiga computer hardware, Q&A, reviews, etc.
comp.sys.amiga.introduction	Group for newcomers to Amigas.
comp.sys.amiga.marketplace	Where to find it, prices, etc.
comp.sys.amiga.misc	Discussions not falling in another Amiga group.
comp.sys.amiga.multimedia	Animations, video, & multimedia.
comp.sys.amiga.programmer	Developers & hobbyists discuss code.
comp.sys.amiga.reviews	Reviews of Amiga software, hardware. (Moderated)
comp.unix.amiga	Minix, SYSV4 and other *nix on an Amiga.



## 8.8 Atari

<code>alt.games.lynx</code>	The Atari Lynx.
<code>comp.binaries.atari.st</code>	Binary-only postings for the Atari ST. (Moderated)
<code>comp.sources.atari.st</code>	Source code-only postings for the Atari ST. (Moderated)
<code>comp.sys.atari.8bit</code>	Discussion about 8 bit Atari micros.
<code>comp.sys.atari.advocacy</code>	Attacking and defending Atari computers.
<code>comp.sys.atari.st</code>	Discussion about 16 bit Atari micros.
<code>comp.sys.atari.st.tech</code>	Technical discussions of Atari ST hard/software.

## 8.9 Grafik

<code>alt.binaries.multimedia</code>	Sound, text and graphics data rolled in one.
<code>alt.graphics</code>	Some prefer this to <code>comp.graphics</code> .
<code>alt.graphics.pixutils</code>	Discussion of pixmap utilities.
<code>bit.listserv.ingrafx</code>	Information Graphics.
<code>comp.graphics</code>	Computer graphics, art, animation, image processing.
<code>comp.graphics.animation</code>	Technical aspects of computer animation.
<code>comp.graphics.avs</code>	The Application Visualization System.
<code>comp.graphics.explorer</code>	The Explorer Modular Visualisation Environment (MVE).
<code>comp.graphics.gnuplot</code>	The GNUPLOT interactive function plotter.
<code>comp.graphics.opengl</code>	The OpenGL 3D application programming interface.
<code>comp.graphics.research</code>	Highly technical computer graphics discussion. (Moderated)
<code>comp.graphics.visualization</code>	Info on scientific visualization.
<code>comp.sys.amiga.graphics</code>	Charts, graphs, pictures, etc.
<code>rec.photo</code>	Hobbyists interested in photography.



## Teil II

# IRC

## 1 Einführung

IRC ist ein Netzwerk, in dem sich auf mehreren Kanälen mehrere Benutzer gleichzeitig unterhalten können. Es ermöglicht den Teilnehmern am Internet, sich in Echtzeit zu unterhalten und ist damit ein Ersatz und eine Verbesserung gegenüber „talk“. IRC hat dessen volle Funktionalität, bietet darüber hinaus mehr als zwei Benutzern die Möglichkeit, sich gleichzeitig zu unterhalten (ähnlich einer Konferenzschaltung), und andere Goodies.

Um am IRC teilzunehmen, ruft man das entsprechende IRC-Client-Programm einfach auf: unter Unix „irc“. Auf anderen Betriebssystemen existieren andere, auf die jeweilige Systemumgebung angepaßte Clients.

Wenn man unter einem anderen Pseudonym (Nickname) als dem Login-Namen bekannt sein will, so gibt man dies beim Aufruf an: „irc nickname“. Jeder IRC-Benutzer wählt ein Pseudonym, unter dem die gesamte Konversation geführt wird.

Das Wichtigste am IRC ist, daß man Spaß daran hat, es zu entdecken und zu lernen. Wenn man sich die Zeit nimmt und nicht durcheinanderbringen läßt, dann wird man bald neue Freunde auf der ganzen Welt finden!

IRC basiert auf einem Client-Server-Modell. Clients sind Programme die sich an einen Server anbinden, ein Server ist ein Programm das Daten (Nachrichten) von einem Benutzer-Client zum einem anderen Client befördert. Es existiert eine Reihe von IRC-Clients für verschiedenen Umgebungen (Unix, Emacs, VMS, MSDOS, MV, ...) die es alle erlauben, sich an einen Server anzuschließen. Der Client, von dem hier am meisten gesprochen wird, ist auch am weitesten verbreitet: „ircII“. Andere Clients sind ähnlich, und verstehen oft ircII-Befehle.



## 2 Konventionen

Englisch ist die Sprache, die am IRC am meisten gesprochen und verstanden wird. Da IRC jedoch in vielen Ländern benutzt wird, ist Englisch bei weitem nicht die einzige Sprache. Wenn man eine andere Sprache als Englisch sprechen will (z. B. mit Freunden), dann sucht man sich einen eigenen Kanal und setzt dort das Thema (Topic) entsprechend, um dies anzuzeigen. Anders herum sollte man das Thema eines Kanals beachten, um herauszufinden, ob es irgendwelche Einschränkungen gibt.

Es ist nicht nötig, jeden Teilnehmer eines Kanals persönlich zu grüßen. Ein „Hallo“ oder Ähnliches reicht für gewöhnlich. Auch sollte man nicht erwarten, von jedermann zurückgegrüßt zu werden... Auf einem Kanal mit 20 Teilnehmern würde dies zu einer Bildschirmseite voll mit Hallos führen. Es ist auch nicht nötig, jeden zu grüßen, der einen Kanal betritt. Wenn man dies unbedingt will weil man ihn kennt, so sollte man dies mit einer privaten Nachricht tun. Dasselbe gilt auch für das Verabschieden von einem Kanal.

Es sollte auch angemerkt werden, daß es sehr schlechter Stil ist, die Möglichkeiten eines Clients dazu zu verwenden, automatisch „Hallo“ oder „Auf Wiedersehen“ zu sagen.

Wenn man einen neuen Kanal betritt, sollte man eine Weile zuhören, um einen Überblick davon zu erhalten, was gerade diskutiert wird. Man kann jederzeit mitdiskutieren, sollte aber nicht darauf bestehen, ein eigenes Thema einzubringen, wenn es sich nicht ergibt.

Beachten Sie, daß die Leute am IRC ihre Meinung über Sie nur aufgrund Ihrer Aktionen, Aussagen und Kommentare bilden. Also denken sie, *bevor* Sie lostippen! Wer anstößige Wörter verwendet wird schräg angesehen. Senden Sie nicht Unmengen ungewollter Informationen an einen Kanal oder einen bestimmten Benutzer (etwa via Cut & Paste); Es erhöht die Chancen gewaltig, aus dem Kanal geworfen oder gar vom gesamten IRC verbannt zu werden. Andere unerwünschte Aktionen sind:

- Andere Benutzer ärgern.
- Einen Kanal durch fortwährendes Piepen stören.
- Jedes Benehmen, das die Funktionalität des IRC einschränkt.

Weiterhin sollte man im Hinterkopf behalten, daß die Nachricht, die man einem anderen IRC-Benutzer schickt, über jeden Server zwischen dem Absender und der betreffenden Person läuft. Wenn man jemand einen Brief schickt, dann kann jeder Postbote auf dem Weg den Brief öffnen....

Genauso ist es mit allen Netzwerkdiensten. Jeder IRC-Administrator kann seinen Server so compilieren, daß er alle Nachrichten mitprotokolliert, die über seinen Server gehen. Generell sollte man also keinem Server trauen!

Wie man Direktverbindungen zwischen Clients aufbaut wird später erläutert, siehe „/DCC CHAT“ auf Seite 59. Diese Technik sollte verwendet werden, wenn man verhindern will, daß jemand private Nachrichten am IRC abfängt.



### 3 Erste Schritte

Der am weitesten verbreitete IRC-Client „ircII“ hält alle Hilfstexte online. Wenn man einmal hängenbleiben sollte, einfach „/HELP“ tippen und RETURN drücken. Generell sollte man sich merken:

**Alle ircII-Befehle beginnen mit einem „/“!**

Groß-/Kleinschreibung spielt keine Rolle, Befehle können auch abgekürzt werden: „/SI“ und „/sign <text>“ stehen beide für „/SIGNOFF“ und beenden die IRC-Sitzung.

Alles was nicht mit einem Schrägstrich beginnt wird als Nachricht an jemand interpretiert und in den aktuellen Kanal geschickt.

Wenn man IRC startet, ist man in einem verlassenem Platz, an dem niemand spricht. Man wird dann wohl mit „/JOIN“ einem Kanal beitreten. Eine Liste aller Kanäle wird mit „/LIST“ angezeigt.





## 4 Befehle

Im folgenden Abschnitt werden alle Befehle — ihren Funktionen nach — zusammengefaßt und beschrieben.

### 4.1 Editier-Befehle

- `/HISTORY [<number>]`  
Mit diesem Befehl kann man sich die letzten `number` Befehle anzeigen lassen, die man selber getippt hat.
- `/! [<history-nummer>|<history-muster>]`  
Holt einen zuvor bereits abgeschickten Befehl zurück, er kann mit den im Emacs üblichen Tasten editiert werden:

Taste	Effekt
<code>^P</code>	Letzte Zeile zurückholen.
<code>^N</code>	Nächste Befehlszeile.
<code>^F</code>	Cursor ein Zeichen nach rechts.
<code>^B</code>	Cursor ein Zeichen nach links.
<code>^A</code>	Cursor an Zeilenanfang.
<code>^E</code>	Cursor an Zeilenende.
<code>^D</code>	Zeichen unter Cursor löschen.
<code>^K</code>	Vom Cursor bis zum Zeileende.
<code>^U</code>	Die ganze Zeile löschen.
<code>^L</code>	Bildschirm neu zeichnen.

Die History-Nummer kann mit dem „/HISTORY“-Befehl ermittelt werden. Das History-Muster muß dem Anfang eines bereits gegebenen Befehls entsprechen; Genau wie in der `csh`.

- `/LASTLOG [<anzahl eintraege>|<text> [<von eintrag>]]`  
Listet die letzten `<anzahl eintraege>` Zeilen auf oder alternativ dazu von einer bestimmten Stelle ab.
- `/CLEAR`  
Löscht den Bildschirm.

### 4.2 Allgemeine Befehle

- `/NICK [<nickname>]`  
Hiermit kann man sich ein beliebiges Pseudonym zulegen, unter dem man in Zukunft bekannt ist.

Jeder am IRC sieht dieses Pseudonym, es ist jedoch darauf zu achten, daß die Pseudonyme momentan maximal 9 Zeichen lang sein dürfen. Per Voreinstellung ist das Pseudonym gleich dem Login-Namen. Es kann außerdem mit der Umgebungsvariable „IRC\_NICK“ gesetzt werden.

Es ist nicht erlaubt, daß zwei Leute das selbe Pseudonym verwenden, dies wird von den IRC-Servern überwacht. Wenn man das IRC startet und das beabsichtigte Pseudonym bereits existiert, dann muß man erst ein anderes wählen, bevor man zugelassen wird.

- `/QUIT [<grund>]`  
Beendet die IRC-Sitzung. Man kann auch „/BYE“, „/SIGNOFF“ oder „/EXIT“ verwenden. Wenn ein Grund angegeben wird, so wird er den anderen Leuten am Kanal angezeigt.
- `/HELP [<befehl>]`  
Gibt Hilfestellung zu einem bestimmten Befehl.
- `/WHOIS [[<server>] <pseudonym>]`  
Zeigt Informationen über jemand.

Dies funktioniert jedoch nur, wenn die entsprechende Person mit dem angegebenen Pseudonym gerade am IRC ist. Mit dem „/WHOWAS“ erhält man diese Informationen auch später noch:

- `/WHOWAS [[<server>] <pseudonym>]`  
Zeigt Informationen darüber, wer das angegebene Pseudonym zuletzt benutzt hat, auch wenn es gerade nicht benutzt wird.

Sehr oft führt ein fehlgeschlagenes „/WHOIS“ dazu, daß man „/WHOWAS“ versucht. Die kann mit „/SET AUTO\_WHOWAS ON“ automatisiert werden. „/HELP SET AUTO\_WHOWAS“ liefert nähere Informationen hierzu.

- `/AWAY [<nachricht>]`  
Dieser Befehl hinterlegt eine Nachricht, daß man zur Zeit nicht erreichbar ist und dem IRC keine Beachtung schenkt. Wenn jemand per „/MSG“ eine private Nachricht schickt oder „/WHOIS“ verwendet, so erhält er die entsprechende Nachricht. „/AWAY“ ohne Parameter bedeutet, daß man wieder dabei ist.

### 4.3 Kommunikation und private Konversation

- `/MSG <pseudonym>|<kanal> <text>`  
Schickt eine private Nachricht an den Benutzer mit dem angegebenen Pseudonym. Keiner außer ihm kann diese Nachricht sehen.

Wenn man eine private Nachricht an mehr als eine Person schicken will, so kann man eine Liste von Pseudonymen — durch Kommata getrennt — angeben. (Keine Leerzeichen dazwischen!)

Zwei spezielle Pseudonym sind „,“ (Komma) und „.“ (Punkt). Wenn „,“ angegeben wird, dann wird die Nachricht an die letzte Person geschickt, die eine private Nachricht geschickt hat. Mit „.“ geht die Nachricht an die Person, der man selbst zuletzt eine private Nachricht geschickt hat.

Man kann eine private Konversation betreiben, indem man nur „/MSG“-Befehle verwendet. Andauernd „/MSG <pseudo> <text>“ bzw. „/MSG . <text>“ zu tippen wird aber lästig. Hier kommt der „/QUERY“-Befehl genau richtig!

- `/QUERY [<pseudonym>|<kanal>]`  
Beginnt eine private Konversation mit <pseudonym>.

Der Text, den man tippt und der normalerweise an den Kanal gehen würde, wird nun in Form von privaten Nachrichten an die Person mit dem angegebenen Pseudonym weitergeleitet. Um eine private Konversation abzubrechen reicht „/QUERY“ ohne Parameter.

- /NOTIFY [[-] <pseudonym>

Mit diesem Befehl kann man sich warnen lassen, wenn bestimmte Leute das IRC betreten oder verlassen.

Es kann passieren, daß man Nachrichten von bestimmte Benutzern nicht sehen will. Sei es, weil sie nur dumpf daherlabern oder einen auf die Nerven gehen. Die richtige Antwort auf ein solches Verhalten ist, die Person zu „/IGNORE“n:

- /IGNORE [<pseudonym>|<user@host> [[-]<nachrichtentyp>]]

Unterdrückt Ausgaben der angegebenen Personen auf den Bildschirm. Dabei kann man entweder eine IRC-Pseudonym oder eine Internet-Adresse der Form „User@Host“ angeben. Wildcards sind in jeder Form erlaubt. Die Ausgaben, die unterdrückt werden können beinhalten: private Nachrichten (MSG), Notizen (NOTICE), Öffentliche (PUBLIC) Nachrichten, Einladungen (INVITE), alles (ALL) oder nichts (NONE).

Wenn man vor den zu ignorierenden Nachrichten-Typ ein „-“ setzt, so hebt dies die Ignorierung auf.

## 4.4 Kanäle und Öffentliche Konversation

Auf dem IRC gibt es eine Menge Plätze, an denen man sich aufhalten kann; Diese Plätze werden „Kanäle“ genannt. Man kann die Gespräche auf einem Kanal mit dem Gespräch unter einer Gruppe von Leuten vergleichen: Man sieht/hört alles, was gesagt wird, und man kann auf alles antworten. Was man tippt wird von allen aufgenommen, die zuhören wollen — und jeder, der etwas verpasst wird es nur mitbekommen, wenn es von irgendwem wiederholt wird. Wie im „Richtigen Leben“!

Alle Kanäle am IRC haben Namen: Ein „#“-Zeichen gefolgt von einer Zeichenkette, wie z. B. #C++, #Asians oder #EU-Opers. Gewöhnlich wird der Name des Kanals etwas über die Art der Gespräche oder die auf dem Kanal befindlichen Personen aussagen.

- /LIST [[<flags>] <kanal-maske>]

Listet die vorhandenen Kanäle, die Anzahl der Benutzer und das Thema. Da die Liste normalerweise ziemlich lang ist, kann man sie durch Angabe von Flags kürzen. „/LIST -MIN n“ zeigt zum Beispiel nur die Kanäle an, die mehr als n Benutzer haben. Die Kanal-Maske kann Wildcards enthalten.

- /NAMES [[<flags>] <kanal-maske>

Zeigt die Pseudonyme aller Benutzer auf jedem Kanal an.

- /JOIN [<kanal>]

Setzt den angegebenen Kanal als den aktuellen Kanal. Wird kein Kanal angegeben, so wird der aktuelle Kanal angezeigt. Der „/CHANNEL“-Befehl hat den gleichen Effekt.

Wenn man einen Kanal betritt erhält man nützliche Infos darüber: eine Liste aller Benutzer auf dem Kanal, das Thema, etc.

Sobald man einen Kanal betreten hat, will man oft genauere Informationen über die Leute, die sich auf dem Kanal befinden. Diese bekommt man mit dem „/WHO“-Befehl:

- `/WHO [<kanal>|<wildcard>]`

Gibt Informationen über eine Liste von Benutzern aus, „`/WHO *`“ listet alle Benutzer des aktuellen Kanals auf.

Die erste Spalte enthält den Namen des aktuellen Kanals, das Pseudonym, Status, Real-Name (im Internet-Format: `user@host`) und ein kleiner Kommentar den man mit der Umgebungsvariable „`IRCNAME`“ selbst setzen kann.

Der Status besagt, ob eine Person anwesend („H“) oder abwesend („G“) ist, siehe „`/AWAY`“. IRC-Operateure werden durch einen „\*“ gekennzeichnet, Kanal-Operateure besitzen ein „@“.

Wenn man bereits auf einem Kanal ist, ist es möglich, jemand anders mit dem „`/INVITE`“-Befehl dorthin einzuladen.

- `/INVITE <pseudonym> [<kanal>]`

Lädt die Person mit dem angegebenen Pseudonym auf den angegebenen Kanal ein. Wird der Kanal weggelassen, so wird der aktuelle Kanal verwendet.

Wenn man eine Einladung erhält, so kann man mit „`/JOIN -INVITE`“ der Einladung folgen.

Nun, Sie raten es schon: Wenn man jemand auf einen Kanal einladen kann, dann kann man auch wieder hinauswerfen, etwa, wenn er sich nicht den Konventionen anpaßt, Leute ärgert, ... Dies geht mit dem „`/KICK`“-Befehl:

- `/KICK [<kanal>] <pseudonym>`

Wirft den Benutzer mit dem angegebenen Pseudonym aus dem aktuellen Kanal hinaus. Nur Kanal-Operateure sind dazu privilegiert!

- `/LEAVE <kanal>`

Einen Kanal verlassen.

- `/TOPIC [[<kanal>] <thema>`

Wechselt oder setzt das Thema des angegebenen Kanals.

- `/ME <beschreibung>`

Von Zeit zu Zeit möchte man beschreiben, was man tut oder wie man sich fühlt. Dies geht mit dem „`/ME`“-Befehl; Die korrekte Konvention ist, die Beschreibung mit einem Punkt abzuschließen.

- `/DESCRIBE <pseudonym>|<kanal> <beschreibung>`

Beschreibt einem bestimmten Benutzer auf einem bestimmten Kanal, wie man sich fühlt und was man gerade tut. Ansonsten wie „`/ME`“.

Die Befehle „`/ME`“ und „`/DESCRIBE`“ benutzen das CTCP, ein Client-to-Client-Protokoll, das eingeführt wurde, um bestimmte Aktionen auszuführen, aber nicht von allen IRC-Clients verstanden wird. Wenn man eine Fehlermeldung erhält, dann ist die Beschreibung nicht richtig angekommen.

## 4.5 Kanal- und Benutzer-Modi

Kanäle können zusätzliche Auflagen haben, die mit dem „`/MODE`“-Befehl gesetzt werden können. Um dies zu verstehen sollte man sich klarmachen, daß die erste Person, die einen Kanal betritt,

diesen erschafft und für ihn verantwortlich ist ( $\Rightarrow$ Kanal-Operateur ist). Er oder Sie kann im Folgenden weitere Auflagen erteilen und andere Leute zu Kanal-Operateuren ernennen.

Ein Kanal ist normalerweise öffentlich (*public*). Wenn jemand auf einem öffentlichen Kanal ist, so kann er von allen anderen Benutzern gesehen werden (sofern sein eigener Benutzer-Modus die zuläßt). Jeder kann Benutzer auf öffentlichen Kanälen sehen und sich zu ihnen gesellen („/JOIN“).

Privat (*private*) bedeutet, daß man — obwohl man von jedermann gesehen werden kann — nicht sagen kann, auf welchem Kanal die betreffende Person ist, wenn man nicht gerade auf dem selben Kanal ist. Sie erscheinen einfach als „Prv: \*“ beim „/LIST“ und „/NAMES“. Da die Nummer der möglichen Kanäle in die Milliarden geht, ist dies ziemlich sicher: Alles was andere in Erfahrung bringen können ist, daß man IRCt; Sie können nicht sagen, wo man ist.

Wenn man auf einem geheimen (*secret*) Kanal ist, dann kann niemand, der nicht auf dem selben Kanal ist, sehen, daß man überhaupt am IRC ist. Der eigene Name wird bei „/LIST“-Befehlen nicht angezeigt. Der einzige Hinweis, daß man am IRC ist, besteht darin, daß jeder, der das IRC betritt, erfährt, wieviele Benutzer auf wievielen Servern anwesend sind. Wenn jemand alle Benutzer überprüft und weniger als die angegebene Zahl findet, dann kann er sicher sein, daß sich Leute in geheimen Kanälen aufhalten. Er kann jedoch weder ermitteln, wer diese Leute sind, noch in welchen Kanälen sie sind.

- `/MODE <kanal>|<pseudonym> [[+|-]<modus> [<parameter>]]`  
 Ein „+“- oder „-“-Zeichen bestimmt, ob das Attribut hinzugefügt oder entfernt werden soll. Kanäle können moderiert (*moderated*, nur der Kanal-Operateur kann sprechen), geheim (*secret*), privat (*private*), mit einer Begrenzten Anzahl von Benutzern, anonym (*anonymous*), nur mit Einladung betretbar (*invite-only*), auf ein Thema begrenzt (*topic-limited*), ...sein. Siehe „/HELP MODE“ für eine ausführliche Beschreibung.

Folgende Channel-Modes sind möglich:

Modus	Effekt auf den kanal
<code>b &lt;person&gt;</code>	Person <i>verbannen</i> , <code>&lt;person&gt;</code> als Pseudonym oder <code>user@host</code> .
<code>i</code>	Kanals kann nur mit Einladung betreten werden ( <i>invite-only</i> ).
<code>l &lt;number&gt;</code>	Kanal ist auf max. <code>&lt;number&gt;</code> beschränkt.
<code>m</code>	Kanal ist moderiert.
<code>n</code>	Externe „/MSG“s sind nicht erlaubt.
<code>o &lt;pseud&gt;</code>	Macht <code>&lt;pseudo&gt;</code> zum Kanal-Operateur.
<code>p</code>	Kanal ist privat.
<code>s</code>	Kanal ist geheim ( <i>secret</i> ).
<code>t</code>	Nur Kanal-Operator kann Kanal-Thema setzen.

Der „/MODE“-Befehl kann auch verwendet werden, um die eigenen Parameter, den „user mode“, zu setzen. Diese können mit „/MODE `<eigenes_pseudonym>`“ überprüft werden, manchmal auch mit „/UMODE“. Man beachte, daß „+i“ auf manchen Servern voreingestellt ist, um die zurückgezogenheit der Benutzer nicht zu stören. Dies sollte jedoch kein Problem sein, da jeder Benutzer jederzeit seine eigenen Parameter verändern kann.

Folgende Benutzer-Parameter existieren:

Modus	Effekt auf <code>pseudonym</code>
i	Macht unsichtbar ( <i>invisible</i> ) für andere.
o	IRC-Operator-Status.
s	Server-Notizen empfangen.
w	Prügel ( <i>wallops</i> ) empfangen.

## 4.6 Client to Client Protocol

- `/CTCP <pseudonym> <befehl> [<arguments>]`

Erlaubt es, Client-spezifische Aktionen über das Netzwerk auszuführen. `<befehl>` kann z. B. `VERSION`, `FINGER`, `ACTION`, `ERRMSG`, `CLIENTINFO`, ... sind. `CLIENTINFO` liefert Informationen über alle Befehle, die der Client des Benutzers `<pseudonym>` versteht.

Falls Interesse am CTCP-Mechanismus besteht, so findet man in den Hilfsdateien zu „ircII“ weitere Hinweise. „/HELP CTCP“ kann vielleicht auch weiterhelfen... Das Client-to-Client-Protocol bietet auch einen sehr mächtigen Dienst an: mit seiner Hilfe können Dateien zwischen zwei Personen ausgetauscht werden. Die allgemeine Syntax ist:

- `/DCC <funktion> [<arguments>]`

Um kleine Textdateien zu verschicken ist wahrscheinlich E-Mail die beste Lösung. Trotzdem bietet „ircII“ die Möglichkeit, direkte Client-Verbindungen (Direct Client-Connections, DCC) an, um z. B. Dateien zu empfangen oder zu versenden.

Wenn PseudoA eine Datei an PseudoB schicken möchte, so kann er dies, indem er „/DCC SEND PseudoB dateiname“ tippt.

PseudoB wird benachrichtigt und wenn er die Datei will, so kann er sie mit „/DCC GET PseudoA dateiname“ empfangen. Hier ein Überblick der Möglichkeiten:

- `/DCC SEND <pseudonym> <dateiname>`

Bietet eine Datei-Übertragung via DCC an.

Wenn man gebeten wird, jemand eine Datei zu senden, sollte man genau wissen was man tut. Zum Beispiel darf die Paßwort-Datei nie weitergegeben werden. Dies könnte Crackern illegalen Zugriff auf Ihren Rechner ermöglichen und Sie und Ihren Systemverwalter in Schwierigkeiten bringen!!!

- `/DCC GET <pseudonym> <dateiname>`

Akzeptiert eine Datei-Übertragung, der Sender muß die Datei mit „/DCC SEND“ angeboten haben.

- `/DCC LIST`

Zeigt alle offenen DCC-Verbindungen mit ihrem Typ, Status und dem Pseudonym des Empfängers.

- `/DCC CLOSE <type> <pseudonym> [<argumente>]`

Beendet ein ungewünschtes DCC-Angebot. `<type>`, `<pseudonym>` und `<argumente>` müssen dieselben sein wie von „/DCC LIST“ angegeben.

- `/DCC CHAT [<pseudonym>]`  
`„/DCC CHAT <pseudonym>“` startet eine Direktverbindung zum Client des angegebenen Benutzers, dieser muß mit `„/DCC CHAT“` antworten. Sobald die Verbindung aufgebaut ist, können mit `„/MSG =pseudonym text“` Nachrichten ausgetauscht werden.

## 4.7 Netzwerk-Befehle

Wenn man auf Schwierigkeiten stößt, sollte man sich an den lokalen IRC-Administrator oder einen IRC-Operateur wenden. Der IRC-Administrator hat Zugang zu allen Dateien, die für den IRC-Server wichtig sind. IRC-Operateure wurden von IRC-Administratoren privilegiert, ihre Aufgabe ist es, ein stabiles, schnelles IRC-Netzwerk zu sichern. Im Folgenden wird beschrieben, wie man herausfindet, wer IRC-Administrator und/oder -Operateur ist.

Noch ein Hinweis: Wenn man nach dem Namen eines Servers gefragt wird, kann man auch das Pseudonym einer Person angeben, die an diesen Server angeschlossen ist. Dies erweist sich ab und an als nützlich....

- `/ADMIN [<server>]`  
Zeigt administrative Informationen wie etwa die Internet-Adresse (FQDN), Organisation, den IRC-Administrator und dessen E-Mail-Adresse des angegebenen Servers an. Wird kein Server angegeben, so wird der Server verwendet, an den man gerade angeschlossen ist.
- `/LINKS [[<server>] <wildcard>]`  
Gibt eine Liste mit allen Servern aus, die momentan das IRC-Netzwerk bilden. Wird `<wildcard>` angegeben, so werden nur diese Server aufgelistet.
- `/SERVER [<server>|<server-nummer> [<port-nummer>]`  
Wechselt zum angegebenen Server mit der angegebenen Port-Nummer. Wenn keine Port-Nummer angegeben ist, so wird 6667 als Default verwendet.

Gelegentlich passiert es, daß man schneller zwischen zwei Servern umschalten kann, als das IRC-Netzwerk die Nachricht weiterleiten kann, daß man den alten Server verlassen hat. Man sollte also nicht überrascht sein, wenn man gesagt bekommt, daß das eigene Pseudonym bereits verwendet wird.... Einfach ein paar Sekunden warten und dann das Pseudonym mit `„/NICK“` nochmals setzen.

- `/MOTD [<server>]`  
Zeigt die „Nachricht des Tages“ (Message of the day, MOTD) des angegebenen Servers an. Wird `<server>` weggelassen, so wird der aktuelle Server verwendet.
- `/USERS [<server>]`  
Zeigt die Benutzer an, die momentan am Server-Rechner eingeloggt sind.
- `/DATE [<server>]`
- `/TIME [<server>]`  
Zeigt die aktuelle Uhrzeit und das Datum an. Wenn ein Server angegeben wird, so werden die Daten von diesem Server abgefragt. Die Befehle `„/DATE“` und `„/TIME“` sind identisch.
- `/LUSERS [<wildcards> [<server>]`  
Gibt eine Zusammenfassung aller Server, Operateure und Benutzer, die dem angegebene Suchmuster `<wildcards>` entsprechen, aus der Sicht des angegebenen Servers.

- /TRACE [<server>]  
Zeigt die Server-Verbindungen des angegebenen Servers.
- /STATS c|i|k|l|m|u|y [<server>]  
Zeigt diverse Statistiken des IRC-Servers an.
- /INFO [<server>]  
Zeigt Informationen über die IRC-Erfinder, Programmierer und viele weitere Leute an, die schon lange nichts mehr mit dem IRC zu tun haben.
- /VERSION [<server>]  
Gibt die Versionsnummer des ircII-Clients und des Servers aus.



## 5 Weitere ircII-Zaubertricks

ircII ist kein einfaches Client-Programm, sondern ein ganzes Betriebssystem(!), die dazugehörige Programmiersprache ist so einfach wie sie nur sein kann: einfach Schrecklich! Aber wer sich trotzdem damit befassen möchte, der kann hier nähere Informationen finden...

Unter Unix gibt es folgende Umgebungsvariablen:

Name	Effekt
IRCNAME	Alles, was man gerne anstatt seines richtigen Namens haben möchte
IRCNICK	Das Pseudonym, unter dem man bekannt sein möchte.
IRCPATH	Eine Liste von Verzeichnissen, in denen Scripts gesucht werden.
IRCRC	Welche Datei anstatt von <code>\$HOME/.ircrc</code> verwendet werden soll.
IRCSERVER	Eine Liste von Servern, die verwendet werden sollen.
TERM	Der verwendete Terminal-Typ.

Das Befehls-Zeichen („/“) wird nur benötigt, wenn man Befehle interaktiv eingibt, in Programmen wird es nicht benötigt.

Folgende Befehle sind für die Programmierung interessant, sie können im Online-Manual nachgeschlagen werden:

- „#“: Wie „/COMMENT“, nur kürzer
- „/ALIAS“: Erzeugt Befehls-Alias
- „/ASSIGN“: Erzeugt Variablen
- „/BIND“: Bindet Funktion an Tastenkombination
- „/CD“: Wechselt Arbeitsverzeichnis des ircII-Clients
- „/COMMENT“: Kommentar
- „/ECHO“: Gibt die übergebenen Parameter aus
- „/EXEC“: Erlaubt das Starten von Subprozessen
- „/FLUSH“: Leert Ausgabe-Puffer
- „/IF“: Abfrage-Operator für boole'sche Ausdrücke.
- „/LOAD“: Lädt eine ircII-Script-Datei
- „/ON“: Hinterlegt Aktionen für bestimmte Ereignisse
- „/REDIRECT“: Leitet die Ausgabe eines Befehls um
- „/SAVE“: Speichert alle ircII-Einstellungen in eine Datei
- „/SAY“: Sendet Text an einen Benutzer oder einen Kanal
- „/SEND“: Wie „/SAY“
- „/SET“: Weist einer Variable einen Wert zu
- „/SLEEP“: Ein paar Sekunden warten
- „/TYPE“: Simuliert Tastatureingaben

- „/WAIT“: Wartet, bis der Server fertig ist
- „/WHILE“: Schleifen-Operator
- „/WINDOW“: Erlaubt die Manipulationen mehrerer „Fenster“ in ircII
- „/XECHO“: Wie „/ECHO“, jedoch über Flags steuerbar

Wenn man ircII öfters benutzt, wird man feststellen, daß man zu Beginn immer wieder die gleichen Befehle eingibt. Dies kann man sich sparen: Beim starten sucht ircII im Home-Directory nach der Datei `.ircrc` und arbeitet die darin enthaltenen Befehle ab.

## 6 Häufig gestellte Fragen

### 6.1 Wo bekomme ich einen IRC-Client her?

Auf den Suns des RZ-CIP-Pools (`rrzc1`, ...) sowie der `rrzsg1` ist das IRC-Programm fest installiert, Aufruf mit „irc“.

Für andere Rechnerplattformen sollte die folgende Liste von FTP-Servern nützlich sein:

UNIX ircII	<code>cs.bu.edu</code> <code>nic.funet.fi</code> <code>ftp.informatik.tu-muenchen.de</code>	<code>/irc/clients</code> <code>/pub/unix/irc/ircII</code> <code>/pub/net/irc</code>
VMS	<code>cs.bu.edu</code> <code>nic.funet.fi</code> <code>ftp.informatik.tu-muenchen.de</code>	<code>/irc/clients/vms</code> <code>/pub/unix/irc/vms</code> <code>/pub/net/irc</code>
VM REXX	<code>ftp.informatik.uni-oldenburg.de</code> <code>ftp.informatik.tu-muenchen.de</code> <code>cs.bu.edu</code> <code>nic.funet.fi</code>	<code>/pub/irc/rxirc</code> <code>/pub/net/irc/VM</code> <code>/irc/clients/rxirc</code> <code>/pub/unix/irc/rxirc</code>
MSDOS	<code>cs.bu.edu</code> <code>nic.funet.fi</code>	<code>/irc/clients/msdos</code> <code>/pub/unix/irc/msdos</code>
Macintosh	<code>cs.bu.edu</code> <code>nic.funet.fi</code> <code>sumex.stanford.edu</code> <code>ftp.ira.uka.de</code>	<code>/irc/clients</code> <code>/pub/unix/irc/mac</code> <code>/info-mac/comm</code> <code>/pub/systems/mac</code>
Amiga	<code>f.ms.uky.edu</code>	<code>/pub/amiga/Telecom</code>

Für den Amiga existiert im DNet-Paket ein exzellenter IRC-Client!

### 6.2 Welchen Server soll ich nehmen?

Am besten ist normalerweise der geographisch am nächsten liegende Server. Ist man erstmal am IRC, so kann man sich nach anderen Servern erkundigen. Folgende deutsche Server gewähren Zugang:

- `noc.belwue.de` (129.143.2.1)
- `irc.uni-erlangen.de` (131.188.61.13)
- `eva.fmi.uni-passau.de` (132.231.1.15)
- `irc.informatik.tu-muenchen.de` (131.159.0.198)

### 6.3 Ein paar ausgewählte Channels

In den Kanälen `#Hottub` und `#initgame` halten sich immer Leute auf. `#Hottub` ist als Simulation eines Whirlpools gedacht, `#init` ist ein Non-Stop-„Init“-Spiel. Einfach „/JOIN“en und

ausprobieren!

In Kanälen, die nach deutschen Universitätsstädten benannt sind, wird in der Regel Deutsch gesprochen, z. B. `#Tuebingen`, `#Konstanz`, `#Stuttgart`...

Viele IRC-Operateure halten sich in `#Twilight_Zone` auf, in `#EU-Opers` findet man Operateure aus ganz Europa.

Auf den Kanälen `#linux` und `#linux68k` geht es hauptsächlich um das von Linus Thorvalds entwickelte PD-Unix „Linux“ bzw. dessen Motorola-Portierung.

Auf den Kanälen `#amiga`, `#amiga!` und `#amigager` trifft man Amiga-Besitzer und -Fans an, auf letzterem wird ausschließlich Deutsch gesprochen.

## 6.4 Was ist NickServ?

Um aus dem Hilfstext des NickServ zu zitieren, Sinn und Zweck von NickServ ist es, Verwirrung mit Pseudonymen zu vermeiden. Ein Pseudonym kann zwar nicht fest vergeben werden, NickServ schickt jedoch eine Nachricht an den, der sich unter dem Pseudonym eines eingetragenen IRC-Benutzers anmeldet. Wenn man sich 10 Wochen nicht am IRC blicken läßt, dann verfällt die Zuteilung und das Pseudonym kann wieder von jedermann verwendet werden.

Das NickServ-Passwort kann nur von einem NickServ-Operateur geändert werden. Mit dem Befehl `„/MSG NickServ@service.de OPERWHO“` kann man herausfinden, welche NickServ-Operateure sich zur Zeit am IRC aufhalten. Pseudonyme mit einem `„*“` daneben sind am IRC. Eine kurze Hilfestellung kann mit `„/MSG NickServ@service.de help“` erhalten werden.

## 6.5 Wo kann ich weitere Informationen finden?

Es existiert eine Usenet-Gruppe `ALT.IRC`, in der man mitdiskutieren, Fragen stellen und beantworten kann.

Es existieren auch mehrere Mailing-Lists mit IRC-bezogenen Themen:

Adresse für Bestellung	Themen
<code>operlist-request@eff.org</code>	Server-Code, routing und Protokoll
<code>ircd-three-request@eff.org</code>	<code>ircd 3.0</code> Protokoll
<code>listserv@grasp1.univ-lyon1.fr</code>	Europäische IRC-Operateur-Liste
<code>d12p+@andrew.cmu.edu</code>	<code>ircII</code> -Liste
<code>vmsirc-request@vax1.elon.edu</code>	VMS-IRC-Liste
<code>irchat-request@cc.tut.fi</code>	<code>irchat</code> -Liste

Auf `Operlist` wird Server-Code von aktuellen und älteren Versionen, Routing und das IRC-Protokoll diskutiert. Bei `ircd-three@eff.org` geht es hauptsächlich um Protokoll-Erweiterungen für Version 3.0 des IRC-Deamon-Programms `ircd`, das momentan konzipiert wird. Auf den Listen `ircII`, `vmsirc` und `irchat` tut sich zur Zeit relativ wenig.

## Teil III

# Elm

## 1 What is it?

Elm ist ein Shell-basierender Mail-Client, nutzt jedoch — im Gegensatz zu vielen seiner Vorgänger, siehe [39] — 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 werden<sup>9</sup>.

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
```

---

<sup>9</sup>Die 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 [37].

## 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"
```





## Teil IV

# Anhang

## A Die Datei `.newsrc`

Die Datei `.newsrc` im Homedirectory enthält, welche Newsgroups subscribed bzw. unsubscribed sind und welche Artikel man schon gelesen hat. Dabei existiert für jede am News-Server bekannte Gruppe ein Eintrag pro Zeile.

Die Reihenfolge, in der die Gruppen darin aufgelistet sind, wird auch beim Lesen von den einzelnen Newsreadern respektiert: Die Gruppen, die ganz oben stehen, werden auch zuerst angezeigt. Man kann sich also die Reihenfolge, in der man die Gruppen aufgelistet bekommen möchte mit jedem beliebigen Texteditor selbst einrichten, z. B. alphabetisch sortiert, ... Das ist eigentlich auch der einzige Grund, wieso man sich um die Existenz dieser Datei überhaupt kümmern sollte, alles andere wird von den Newsreadern erledigt.

Bevorzugt man eine alphabetische Reihenfolge, so kann man dies mit der folgenden Befehlsfolge erreichen:

```
% cd $HOME
% sort .newsrc >.newsrc.srt
% mv .newsrc.srt .newsrc
```

Werden neue Gruppen angelegt, so wird man beim Start von TIN gefragt, ob man sie subscriben will. Diese neu angelegten Gruppen werden auf jeden Fall ganz hinten angehängt, da `.newsrc` ja sämtliche Gruppen enthält.

Wie sieht nun eine Zeile dieser Datei aus und welche Informationen enthält sie? Jede Zeile enthält folgende Angaben (von links nach rechts):

- Den Namen der Gruppe
- Ein Kennzeichen, ob die Gruppe subscribed („:“) oder unsubscribed („!“) ist
- Eine Auflistung der bereits gelesenen Artikel

Hier ein paar Beispiele:

```
gnu.g++.help:
gnu.gcc.help: 1-3
alt.dcom.catv!
misc.jobs.contract! 1-114,246,251
```

GNU.G++.HELP ist zwar subscribed („:“), es wurde aber noch kein Artikel gelesen.

GNU.GCC.HELP ist subscribed, die Artikel 1 bis 4 wurden gelesen und werden im Folgenden nicht mehr angezeigt.<sup>10</sup>

---

<sup>10</sup>Sie sind natürlich weiterhin vorhanden und können im „tin“ mit der „r“-Taste sichtbar gemacht werden.

Die Gruppe ALT.DCOM.CATV ist unsubscribed („!“), sie wird bei der Auflistung der Newsgroups nicht angezeigt. Im „tin“ bekommt man mit „y“ eine Auflistung aller existierenden Gruppen.

MISC.JOBS.CONTRACT ist zwar unsubscribed, es wurden aber trotzdem schon ein paar Artikel gelesen. Dies kann zweierlei Gründe haben. Entweder wurden in dieser Gruppe News gelesen, ohne sie zu subscriben, oder man hatte die Gruppe bestellt, gelesen, und dann abbestellt.

Wie gesagt soll diese Information hier nur der Vollständigkeit halber aufgeführt sein. Wem die Reihenfolge, in der die News-Groups aufgelistet werden, egal ist, der kann die Datei getrost ignorieren (aber nicht löschen!).

## B Mein elmrc :-)

Hier soll an einem Beispiel aufgezeigt werden, welche Möglichkeiten Elm zur Konfiguration bietet. [38] 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
```

## C Beispiel-.signatures

Hier sollen einige Beispiel für die Datei `.signature` gezeigt werden. In diesem Zusammenhang sei nochmals explizit auf die entsprechenden Postings in `de.newusers` verwiesen, die sich mit diesem Thema befassen ([9], [5], [10]).

```
Walter Wasser, Wasserkuppenstr.23, 6401 Tuempel, Telefon: +12-781-41505
walter@psygnosis.hohn.sub.org
```

```
\ Kurt 'Heinemann' Kuerzer          \      mni%(0241)25034255102::delphi \
\ Student consultant WINDOWS and networks \ delphi@mulhhu51.bitnet \
\ Rechenzentrum Universitaet Ulm Germany \ delphi@pt.uni-ulm.de \
\ "With mainframes you just can do nonsense, with PCs not even that." \
```

```
      hein kurz      o      hein@bogred.ka.sub.org      voice@94-217-06493
comp.sc.student    _>\ _      h_kurz@ira.uka.de      caesar@irc
karlsruhe germany  (_)<(_)      unkf@kadnui2.bitnet      happy linux user
what good is a photograph of you? everytime i look at it it makes me feel blue
```

```
          ,,,
          (o o)
-----o00--(_)--00o-----
Michael Miramar privat: <mira@hexy.ob.open.de>      (hexy.pott.sub.org)
                        @work: <mira@macrots.uucp> ...!unido.uucp!macrots!mira
PD/Share-(Soft)ware for waffle, msdos, etc... +94 280 669771 [11am-8pm]
Anonymous UUCP -> Login: nuupc Password: nuupc Index: ~/public/index.txt
```

```
_      Tom Jimpson      --      ---
_| ~-  jimpson@wg9.wvii.edu  /\_\  /\_\_\
\, _}  Western Geophysical  \/_/  // / /
 \ (   Houston, Texas      \/_/
```

```
-----
Hubert Feyrer, Bachstr. 40, W-8304 Mallersdorf, GERMANY
Internet: feyrer@rrzc1.rz.uni-regensburg.de
Tel.: +049 (0)8772 / 6084
```



## D Software-Herkunft

Hier soll die Herkunft der installierten Software festgehalten werden. Es wird jeweils der FTP-Server, der Pfad und der Name des Archivs bzw. Verzeichnisses oder der einzelnen Dateien festgehalten. Eine optionale Beschreibung gibt nähere Hinweise zur Installation der Software.

- Elm:  
*FTP-Server:* ftp.uni-paderborn.de  
*Pfad:* /pub/unix/elm/elm-2.4  
*Datei:* elm-2.4.13.tar.Z
- GNU Emacs (The One and Only!):  
*FTP-Server:* rrzc1.rz.uni-regensburg.de  
*Pfad:* /usr2/gnu/emacs  
*Datei:* Alle inkl. Unterverzeichnisse  
*Bemerkung:* Kann binär übernommen werden, muß wieder in /usr2/gnu/emacs installiert werden. Lösung (um Konflikte mit Lucid Emacs zu vermeiden): Installation in /usr-local/gnu/emacs, „ln -s /usr/local /usr2“.
- GNU grep:  
*FTP-Server:* ftp.uni-kl.de  
*Pfad:* /pub/GNU/grep  
*Datei:* grep-1.6.tar.Z  
*Bemerkung:* Aufgrund von Bugs und Unzulänglichkeiten in SunOS 4.1.3 muß die GNU-Variante der Befehle „grep“ und „egrep“ verwendet werden.
- GNU zip:  
*FTP-Server:* rrzc1.rz.uni-regensburg.de  
*Pfad:* ~c9020/bin  
*Dateien:* gzip, gunzip, zcat  
*Bemerkung:* Können binär übernommen werden.
- GNUs:  
*FTP-Server:* liasun3.epfl.ch  
*Pfad:* /pub/gnu/emacs  
*Datei:* Gnus2.tar.Z  
*Bemerkung:* Ergänzung für GNU-Emacs, die enthaltenen Lisp-Dateien sollten vor Gebrauch kompiliert werden. Ist bereits Bestandteil von Lucid Emacs.
- INN V1.2:  
*Datei:* inn-changed.tar.Z  
*Bemerkung:* Gepatchte Version der Uni Passau, um NNTP-Connects für PCs zu ermöglichen.
- INN V1.4:  
*FTP-Server:* ftp.uni-erlangen.de  
*Pfad:* /pub/news/inn  
*Datei:* inn-1.4.tar.Z  
*Bemerkung:* Neueste, ungepatchte Version. Wird erst auf dem neuen News-Server installiert werden.
- Lucid Emacs:

- FTP-Server:* rrzc1.rz.uni-regensburg.de  
*Pfad:* /usr/local/emacs  
*Datei:* Alle inkl. Unterverzeichnisse  
*Bemerkung:* Kann binär übernommen werden, muß in /usr/local/emacs installiert werden.
- MTools:  
*FTP-Server:* rrzc1.rz.uni-regensburg.de  
*Pfad:* ~c9020/bin  
*Datei:* m\*  
*Bemerkung:* Können binär übernommen werden.
  - nn:  
*FTP-Server:* ftp.uni-kl.de  
*Pfad:* /pub/unix/mail+news/nn  
*Datei:* nn-6.4.16.tar.Z
  - patch:  
*FTP-Server:* rrzc1.rz.uni-regensburg.de  
*Pfad:* /phy1/local/bin  
*Datei:* patch  
*Bemerkung:* Kann binär übernommen werden.
  - PERL:  
*FTP-Server:* rrzc1.rz.uni-regensburg.de  
*Pfad:* /phy1/local/bin  
*Datei:* perl  
*Pfad:* /phy1/lib/perl  
*Datei:* Alle  
*Bemerkung:* Kann binär übernommen werden, die lib-Files kommen nach /usr/local/lib/perl, danach „ln -s /usr/local /phy1“.
  - rn:  
*FTP-Server:* ftp.uni-kl.de  
*Pfad:* /pub/unix/mail+news/rn  
*Datei:* rn-4.4.3.tar.Z
  - tcsh:  
*FTP-Server:* rrzc1.rz.uni-regensburg.de  
*Pfad:* /usr/local/bin  
*Datei:* tcsh  
*Bemerkung:* Kann in binärer Form übernommen werden.
  - TIN:  
*FTP-Server:* files1zrz.zrz.tu-berlin.de  
*Pfad:* /pub/news/tin  
*Datei:* tin-1.1.9.tar.Z  
*Bemerkung:* Neueste Version der primary Site für TIN-Releases.
  - XRn:  
*FTP-Server:* ftp.uni-kl.de  
*Pfad:* /usr/unix/mail+news/xrn  
*Datei:* xrn6-18beta2.tar.Z



- xvnews:  
*FTP-Server:* ftp.zrz.tu-berlin.de  
*Pfad:* /pub/news/xvnews  
*Datei:* xvnews-2.0.2.tar.Z
- ZModem:  
*FTP-Server:* ftp.uni-kl.de  
*Pfad:* /pub/unix/communication  
*Datei:* zmodem.tar.Z  
*Bemerkung:* Für Datenübertragung Unix ⇒ Amiga



## E Literaturverzeichnis

- [1] Joachim Astel (achim@jat.nbg.sub.org). Die Newsgruppen der DE.\*-Hierarchie, April 1993. Existiert als Textdatei im PD-Server in `/freeware/software/news/newusers/de-xxx.txt`.
- [2] Joachim Astel (achim@jat.nbg.sub.org). Einführung, April 1993. Existiert als Textdatei im PD-Server in `/freeware/software/news/Newusers/Einfuehrung.txt`.
- [3] Joachim Astel (achim@jat.nbg.sub.org). Einleitung zu DE.NEWUSERS, April 1993. Existiert als Textdatei im PD-Server in `/freeware/software/news/Newusers/Einleitung.txt`.
- [4] Joachim Astel (achim@jat.nbg.sub.org). Einrichtung von Usenet-Gruppen, April 1993. Regeln für die Einrichtung und Entfernung von Usenet-Gruppen: RfD, CfV, ... Existiert als Textdatei im PD-Server in `/freeware/software/news/Newusers/NeueGruppen.txt`.
- [5] Joachim Astel (achim@jat.nbg.sub.org). Flame-Anleitung, April 1993. Satirischer Text, wie man sich "beliebt" machen kann. Existiert als Textdatei im PD-Server in `/freeware/software/news/Newusers/Flaming.txt`.
- [6] Joachim Astel (achim@jat.nbg.sub.org). Glossar, April 1993. Erklärung einiger Ausdrücke des Usenet-Jargons. Existiert als Textdatei im PD-Server in `/newusers/Glossar.txt`.
- [7] Joachim Astel (achim@jat.nbg.sub.org). Häufig gestellte Fragen, April 1993. Existiert als Textdatei im PD-Server in `/freeware/software/news/Newusers/FAQ.txt`.
- [8] Joachim Astel (achim@jat.nbg.sub.org). Headerzeilen, April 1993. Erklärung der Headerzeilen: Subject:, Newsgroups:, Distribution:, ... Existiert als Textdatei im PD-Server in `/freeware/software/news/Newusers/Headerzeilen.txt`.
- [9] Joachim Astel (achim@jat.nbg.sub.org). Netiquette, April 1993. Die wichtigsten Verhaltensmaßregeln in Stichpunkten zusammengefaßt. Existiert als Textdatei im PD-Server in `/freeware/software/news/Newusers/Netiquette.txt`.
- [10] Joachim Astel (achim@jat.nbg.sub.org). SEX mit Schafen und Katzen, April 1993. Ein weiterer satirischer Text, der häufig gestellte Fragen entsprechend behandelt. Existiert als Textdatei im PD-Server in `/freeware/software/news/Newusers/SEX.txt`.
- [11] Joachim Astel (achim@jat.nbg.sub.org). Umlaute im Usenet, April 1993. Ae, ä oder "a? Existiert als Textdatei im PD-Server in `/freeware/software/news/Newusers/Umlaute.txt`.
- [12] Hubert Sieverding. V8 Twin Sparc — Die Systemtechnologie des SPARCsystems 10. *iX*, (7):42–47, Juli 1992. Technische Daten und Beschreibung zu Bussystem und Prozessor der SPARCstation 10.
- [13] Brendan P. Kehoe. Zen and the Art of the Internet, Januar 1992. Behandelt News, telnet, ftp, ... Sehr empfehlenswert!
- [14] Larry Wall and Randal L. Schwartz. *Programmieren in perl*. Carl Hanser Verlag, 1993. Deutsche Ausgabe des bei O'Reilly & Associates erschienenen Originals „Programming in perl“, auch als „Camel-Book“ bekannt.
- [15] Nicolas Pioch (nicolas.pioch@grasp.inso-lyon.fr) and Owe Rasmussen (d1rasmus@dtek.chalmers.se). A short IRC primer, Februar 1993. Liegt als ASCII-, L<sup>A</sup>T<sub>E</sub>X- und PostScript-Datei auf `nic.funet.fi` im Verzeichnis `/pub/unix/irc`.

- [16] Rich Salz. InterNetNews: Usenet transport for Internet sites. In *Summer '92 USENIX — June 8-June 12, 1992 — San Antonio, TX*, pages 93–98, Juni 1992. Als PostScript-Datei am FTP-Server `ftp.uni-erlangen.de` in `/pub/news/inn/inn-usenix.ps.Z` vorhanden.
- [17] Rich Salz (`rsalz@uunet.uu.net`). *Installing InterNetNews*. Open Software Foundation, 1.2 edition, Januar 1993. Bei INN 1.2 enthaltene Installationsanleitung.
- [18] Sebastian von Bomhard. Power-Box — Erster Test der SPARCstation 10. *iX*, (9):44–48, September 1992. Detaillierter Bericht über Eigenschaften der SPARCstation 10 inkl. Benchmarks.
- [19] Susanne Mührke. Brückenschlag mit System. *Workout*, (4):18–24, 1992. Suns Umschwung von BSD auf SVR4 und das Ergebnis.
- [20] Susanne Mührke. Zwischen Hamburg und Haiti. *Workout*, (4):10–17, 1992. Suns neue Rechnerarchitektur als Basis für Multimedia-Anwendungen.
- [21] Syd Weomstein (`elm@DSI.COM`). *answer(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [22] Syd Weomstein (`elm@DSI.COM`). *autoreply(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [23] Syd Weomstein (`elm@DSI.COM`). *checkalias(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [24] Syd Weomstein (`elm@DSI.COM`). *Elm Configuration Guide*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [25] 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.
- [26] Syd Weomstein (`elm@DSI.COM`). *elm(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [27] Syd Weomstein (`elm@DSI.COM`). *fastmail(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [28] Syd Weomstein (`elm@DSI.COM`). *filter(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [29] Syd Weomstein (`elm@DSI.COM`). *frm(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [30] Syd Weomstein (`elm@DSI.COM`). *listalias(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [31] Syd Weomstein (`elm@DSI.COM`). *messages(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [32] Syd Weomstein (`elm@DSI.COM`). *newalias(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [33] Syd Weomstein (`elm@DSI.COM`). *newmail(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [34] Syd Weomstein (`elm@DSI.COM`). *printmail(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.
- [35] Syd Weomstein (`elm@DSI.COM`). *readmsg(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.

- [36] 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.
- [37] 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.
- [38] 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.
- [39] 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.
- [40] Syd Weomstein (elm@DSI.COM). *wnewmail(1L)*. DataComp Systems, Inc., 3837 Byron Road, Huntingdon Valley, PA 19006-2320, 2.4 edition, October 1992.

