

© 1994,1995,1996 Winfried Trümper <truemper@MI.Uni-Koeln.DE>

Trotz Sorgfalt bei der Zusammenstellung dieses Manuskripts kann ich nicht dessen Richtigkeit und Brauchbarkeit garantieren. Daher

BENUTZUNG DES SKRIPTS AUF EIGENE GEFAHR.

Insbesondere ist es bewußt UNVOLLSTÄNDIG, um eine möglichst knappe Darstellung zu erzielen.
Das Ausprobieren der enthaltenen Beispiele kann Ihre Daten unwiederbringlich ZERSTÖREN und/oder Hardware irreperabel BESCHÄDIGEN. Ich übernehme dafür keine Haftung.

Inhaltsverzeichnis

Teil I: Allgemeine Informationen

- 1 Inhaltsverzeichnis
- 2 Über den Crash-Kurs
- 3 Allgemeines zu Linux

Teil II: Eigene Schritte mit Linux

- 5 Rüstzeug und generelle Bemerkungen
- 5 Aufspüren von Dateien und Wörtern
- 6 Erste Schritte mit Linux
- 7 Grundzüge der Dateiwaltung von Linux
- 8 Tips für Anfänger
- 8 Besonderheiten im Dateisystem
- 9 Dateibetrachter und Texteditoren
- 10 Dokumentation
- 11 Erklärende Übersetzung einiger englischer Vokabeln
- 12 Beschaffung von Software
- 13 Installation von Software

Teil III: Wie Linux funktioniert

- 14 Lebenszyklus eines Linux-Systems
- 15 Benutzer und Gruppen
- 16 Bildschirm und Tastatur
- 17 Der "login"-Vorgang
- 18 "passwords"
- 19 Die »bash«
- 20 shell-Skripte
- 21 Prozesse
- 22 Prozeßkontrolle
- 23 "swap"
- 24 Die Datei als Prinzip
- 25 Die Dateien im Verzeichnis '/dev'
- 26 Organisation des Dateibaums
- 27 Montage des Dateibaums

28	Speichermedien
29	Backup

Teil IV: Linux und Internet

30	Netzwerk und Internet
30	Beispiel eines Netzwerks
31	Die Logistik des Internet
31	Konfigurationsdateien
32	Details zur Logistik
33	Konfigurationsdateien für Fortgeschrittene
33	Vergleich von Medien zur Datenübertragung
33	Zugang zum Internet in Deutschland
34	ISDN
35	Ethernet
35	Anbindung via Modem
36	TCP/IP per Modem

Teil V: Der praktische Teil der Internetbenutzung

37	Benutzung anderer Maschinen im Netz und Zubehör
37	Kleine Internet-Namenskunde
38	Elektronische Post
39	Mailinglisten
40	Das "usenet"
41	Kommunikation mit anderen Benutzern
42	Koordination der Serviceleistungen

Teil VI: Sicherheit

43	Sicherheitsaspekte
----	--------------------

Teil VII: Die grafische Oberfläche

45	X-Windows-System
46	Mehr über "X"
47	Konfiguration von X11-Anwendungen

Teil VIII: Wichtige Systemkomponenten

48	Drucken
49	Linux als Print- und Fileserver
50	Es wird Zeit
51	Deutsche Anpassungen
52	Der Betriebssystemkern
53	LILO
54	"init"
55	Planung der Installation
56	Systemadministration_

Über den Crash-Kurs

2.1 Zielgruppe

Die vorliegenden Seiten sind in erster Linie Unterlagen zu meinen 1- und 5-tägigen Einführungskursen zu Linux gedacht, die ich zweimal pro Jahr an der Universität zu Köln halte.

Der Inhalt orientiert sich an dem, was ich in den vergangenen 3

Jahren Anfängern und Interessierten zum Thema Linux erzählt habe. Ich habe versucht, die "Dauerbrenner" in einer möglichst logischen Reihenfolge zu präsentieren und trotz der extrem knappen Darstellung eine gewisse Verständlichkeit zu gewährleisten.

Allzu ausführliche Texte sind nach meiner Auffassung ein Indiz für Unzulänglichkeiten, die mit hohem Zeitaufwand beschrieben und somit zementiert werden, anstatt sie mit einigem Programmieraufwand ein für alle mal zu beheben (man denke an ein kleines Konfigurationsmenü oder an eingebaute Hilfe). Der "midnight commander" (siehe 6.3) ist in dieser Hinsicht ein positives Beispiel, ebenso die Kernel-Hilfetexte (siehe 52.X).

Das Manuskript richtet sich an Linux-Benutzerinnen, die ihre eigenen Systemverwalterinnen sind und die Vorgänge auf ihrem Linux-System im Detail verstehen wollen - oder müssen. Wem eine Systemverwalterin zur Seite steht, die den PC pflegt und mit neuen Programmversionen regelmäßig aktualisiert, wird nur die Teile I, II und V für die tägliche Arbeit benötigen.

Ohne den zugehörigen Vortrag muß man diese Unterlagen selbst zum Leben erwecken, indem man alle Beispiele und Bemerkungen ausprobiert. Wer den Vortrag vermißt oder professionelle Hilfe bei der Installation benötigt, kann mich gegen entsprechende Bezahlung gerne engagieren.

2.2 Notation

- o Beispiele, die wortwörtlich auf der Kommandozeile ausprobiert werden können, sind eindeutig mit den Zeichen » und « eingegrenzt. Dies scheint mir notwendig, da alle anderen Anführungszeichen unter Linux spezielle Bedeutung haben. Programmnamen sind ebenfalls mit diesen Zeichen gekennzeichnet.
- o Beispiele, die sinngemäß ausprobiert werden sollen, sind lediglich mit normalen Anführungszeichen " und " gekennzeichnet, sie gehören nur in diesem Fall nicht zum Beispiel.
- o Datei- und Verzeichnisnamen sind zwischen den Apostroph-Zeichen ` und ' eingeschlossen.
- o URL-Adressen (siehe 37.6) werden ohne besondere Markierungen zitiert, erhalten dafür aber eine eigene Zeile.

2.3 Nutzungslizenz

Dieses Manuskript darf nur zu privaten Zwecken in unveränderter Form vervielfältigt und weitergegeben werden, sowohl in gedruckter als auch in elektronischer Form. Kommerzielle Nutzung und der Vertrieb auf CD-ROM sind NICHT gestattet.

Alle weiteren Rechte vorbehalten.

2.4 Erhältlichkeit

Die neuste Version dieses Textes ist stets unter den URLs

<http://www.mi.uni-koeln.de/linux/crash-kurs/>
<ftp://ftp.uni-koeln.de/doc/linux/crash-kurs>

oder via e-mail an die Adresse "ftpmail@ftp.uni-stuttgart.de" mit der Nachricht ("message body")

```
reply-to DIE_EIGENE_e-mail_ADRESSE_HIER_HINSCHREIBEN
open ftp.Uni-Koeln.DE
get /doc/linux/crash-kurs/DATEINAME
```

Es stehen verschiedene Formate (-> anstelle von DATEINAME eintragen) zur Verfügung:

DATEINAME	Größe	Beschreibung
ckl-3.5.ps.gz	90 kb	postscript-Version für Linux
ckl-3.5.txt.gz	70 kb	Text-Version für Linux
ckl35_ps.zip	90 kb	postscript-Version für DOS
ckl35_txt.zip	70 kb	Text-Version für DOS

Letzteres Verfahren nennt sich "ftpmail" und eignet sich für beliebige Dateien auf beliebigen ftp-Servern; allerdings wartet man auf die Antwort u.U. einen Tag.

2.5 Voraussetzungen

Ausgangspunkt des Manuskripts ist ein ist lauffähiges Linux-System, auf dem die Kommandozeile benutzt werden kann. Obwohl ich persönlich die Debian-Distribution verwende, sollten die Beispiele auch unter anderen Distributionen funktionieren.

Die Installation aller Linux-Distributionen ist heutzutage menügeführt und derart einfach, daß mir der Platz für eine detaillierte Beschreibung der einzelnen Schritte als zu kostbar erscheint.

Stattdessen erscheint es mir sinnvoller, die Planung eines Linux-Systems näher zu erläutern, siehe 26.3 und 55.X._

Allgemeines zu Linux

Glossar:

- o Mit "Server" werden u.a. Computer bezeichnet, die in einem Netz von Computern Daten und Dienstleistungen zur Verfügung stellen.
- o "POSIX" bezeichnet einen Programmierstandard in der Linux-Welt.

3.1 Was ist Linux?

- o "Linux" ist genaugenommen ein Betriebssystemkern (engl. "kernel"), dessen Entwicklung im April 1991 vom finnischen Studenten Linus Torvalds gestartet wurde.
- o Der Name geht auf "Linus' Unix" zurück und zeigt damit an, daß Linux ein unix-artiger Betriebssystemkern ist.
- o Dieser Betriebssystemkern ist nach dem Start des Computers die oberste Instanz und hat im wesentlichen folgende Fähigkeiten:
 - "multiuser": jede Personen, die den Computer benutzt, hat eine eigene "Privatsphäre" (geschützt durch ein Geheimwort)
 - "multitasking": mehrere Aufgaben (engl. "tasks") können quasi gleichzeitig erledigt werden (siehe 21.2)
 - die Ressourcen des Computers (wie Arbeitsspeicher [RAM], Rechenleistung und Netzwerk) werden den konkurrierenden Programmen zugeteilt (d.h. keine Selbstbedienung), dadurch

wird eine gewisse Sicherheit erreicht

- verschiedene Modelle von Computerbestandteilen mit gleicher Funktion (z.B. verschiedene Soundkarten) können von den Programmen in stets gleicher Art und Weise angesprochen werden; eine große Hilfe für Programmierer!

- o Der Betriebssystemkern von Linux ist in der Programmiersprache C geschrieben und liegt im Quelltext vor.
- o Ohne zusätzliche Programme (Software) ist ein Betriebssystemkern ziemlich nutzlos.

3.2 Linux - das Betriebssystem

- o Um die Fähigkeiten des Betriebssystemkerns komfortabel und sinnvoll nutzen zu können, benötigt man mindestens:
 - Programme zur Erkennung und Behebung von Fehlern
 - den Zugriffsschutz, d.h. die Abfrage des Geheimwortes
 - eine Befehlszeile zum Start weiterer Programme
 - ein System zum automatischen Start der oben genannten Programme
- o Dieses absolut notwendigen Komponenten bezeichnet man landläufig als "Betriebssystem" (eine genauere Definition lasse ich aus).
- o Die Installation des Linux-Betriebssystems ist weitgehend automatisiert und menügeführt. _

3.3 Linux-Distributionen

- o "Linux-Distributionen" sind kinderleicht zu installierende Sammlungen von Programmen "für Linux"; sie enthalten ausserdem das Betriebssystem.
- o Linux-Distributionen werden von verschiedenen Interessengruppen in leicht unterschiedlicher und wechselnder Zusammensetzung, bzw. Qualität angeboten.
- o Es gibt keine "offizielle" Linux-Distribution und somit läßt sich von Linux nicht sprechen wie von DOS, Win oder OS:2.
- o Typischerweise befinden sich nach der Installation mehr als 1000 Programme auf der Festplatte des Computers:
 - eine netzwerkfähige, grafische Oberfläche (X11, XFree86)
 - Programmiersprachen (C, C++, phyton, perl, basic, ...)
 - Internet-Browser (netscape, mosaic, arena)
 - Internet-Server (WWW, FTP, smb, e-mail, ...)
 - Text-Editoren
 - das Text-Satzsystem "TeX"
- ... jeweils mit fast unüberschaubar viel Zubehör und mehr Dokumentation als man lesen kann.
- o Tabellenkalkulationen und WYSIWYG-Textverarbeitungsprogramme der herkömmlichen Art sind noch nicht kostenlos erhältlich. (Von Star gibt es aber entsprechende Ankündigungen, siehe URL <http://www.stardiv.de/german/office/preise/so31.html>)

3.4 Woher stammt die ganze Software?

- o Programmieren ist ebenso wie das Malen oder Nähen eine kreative Tätigkeit, die Freude bereitet.
- o Die meiste Software ist nicht speziell für Linux geschrieben worden (sondern der Betriebssystemkern wurde so geschrieben, daß

die meiste Software auf einem PC unter Linux läuft ...).

- o Das Internet und das Amateurradio schaffen die Infrastruktur zur sekundenschnellen Kommunikation der Software-Entwickler; erst dadurch wurden so große Projekte wie Linux überhaupt möglich.
- o Jedoch ist nicht alle Software für Linux kostenlos erhältlich, in letzter Zeit gibt es erfreulicherweise immer mehr Firmen, die ihre Produkte auch für Linux anbieten.

3.5 Zusammenfassung

- o Im allgemeinen Sprachgebrauch wird nicht zwischen dem Betriebssystemkern, dem Betriebssystem oder den Linux-Distributionen unterschieden.
- o Linux wird von einer Fangemeinde aus Programmierern entwickelt, die Linux selbst einsetzen; daher werden Fehler schnell erkannt und behoben; die Entwicklung ist sehr rasant.
- o Linux ist kostenlos, zuverlässig, schnell und POSIX-konform.
- o Technisch gesehen ist Linux eine gut gelungene Umsetzung von (mehr oder weniger) bekannten Konzepten (und damit veraltet ?).

4.1 Konkret: was man mit Linux anstellen kann

- o Workstation
Linux verwandelt einen PC in eine leistungsfähige Workstation. Als Arbeitsumgebung zum Verfassen von TeX-Dokumenten und als Entwicklungsumgebung zum Schreiben von Software hervorragend geeignet. Beispiele: »xwpe« (ähnlich der Oberfläche von TurboC), »emacs« (AucTeX-Modus)
- o Kostengünstiges X-Terminal
Installiert man nur das Grundsystem (OHNE C-Compiler, TeX, usw.) plus XFree86, so erhält man automatisch ein X-Terminal zu einem unschlagbarem Preis.
- o Parallelrechner
PC-Pools lassen sich per Boot-Diskette kurzfristig und ohne Änderungen an den bestehenden Betriebssystemen in einen Parallelrechner verwandeln (Info's bei Andreas Erdmann <erdmann@mi.uni-koeln.de>). Die Rechenleistung einiger Dutzend zusammenarbeitender P90-PCs ist beachtlich.
- o WWW-Server
Mittlerweile gibt es ein reichhaltiges Angebot an Programmen, mit denen sich WWW-Server unter Linux realisieren lassen. Der "apache-httpd" ist der meistgebrauchte Server; mein persönlicher Favorit ist der "Spinner-httpd".
- o Allgemeine Server-Funktionen
Nahezu jede Server-Funktion läßt sich mit Linux realisieren. Sei es als File-Server für DOS/Win-PCs, als e-mail HUB, FTP-Server, Firewall ...

4.2 Nachteile von Linux

- o Die Fähigkeit zum Verwalten mehrerer Benutzer bedeutet einen höheren Verwaltungsaufwand (in jeder Hinsicht).
- o Hinter Linux und den meisten seiner Programme steht ein Konzept, ohne dessen Kenntnis man hilflos ist.
- o Linux ist nicht besser oder schlechter als die herkömmlichen Betriebssysteme für den PC, sondern hat eine andere Zielgruppe;

deswegen lohnen sich der Aufwand für die Installation und die Einarbeitungszeit nicht in jedem Fall.

4.3 Welche Hardwarevoraussetzungen benötigt Linux/i386?

Theoretisches Minimum: 386 SX, 1MB RAM, 5 MB Festplatte
Akzeptables Minimum: 386 DX/40 mit Coprozessor (!), 8 MB RAM
Akzeptables System: 486 DX/66, 16MB RAM, 600 MB Festplatte
Sehr schnelles System: P90, 32MB RAM, 1GB Festplatte

Weitere Plattformen: Atari, Amiga, MAC, DEC-Alpha, Sparc_

4.4 Rechtlicher Status

- o Da Linux aus sehr vielen Komponenten mit unterschiedlichem Copyright besteht, läßt sich keine einheitliche Aussage machen.
- o Üblicherweise behalten sich die Autoren die Rechte an ihren Werken vor, erlauben aber die Nutzung unter der Auflage der "General Public Licence" (GPL), die im wesentlichen aussagt:
 - die Software DARF weitergegeben werden, aber nur unter denselben Bedingungen
 - die Software kommt ohne irgendwelche Garantien, eine Haftung ist ausgeschlossen
 - es dürfen bei Verkauf lediglich Kosten für den Vertrieb berechnet werden (CD-ROM, Disketten, Papier, Verpackung)
 - es werden KEINE Lizenz-, Registrations- oder Benutzungsgebühren fällig
 - insbesondere müssen stets die Quelltexte verfügbar sein
- o Die GPL wurde von der "Free Software Foundation" (FSF) entworfen und dient als Copyright für deren GNU-Programme (»gcc«, »emacs«, ...); manchmal wird sie auch als "Copyleft" bezeichnet.
- o Nicht alle Linux-Programme unterliegen der GPL, daneben gibt es noch BSD-artige Copyrights, das Copyright des MIT oder die "Artistic Licence" von »perl«.
- o "Freeware" und "Shareware" haben als Copyright fast keine Bedeutung unter Linux.

4.5 Erhältlichkeit

- o Linux ist erhältlich
 - auf CD-ROM
 - im Internet via FTP, NFS oder AFS
- o In den Anfangstagen war Linux auch auf Disketten erhältlich; mittlerweile hat Linux einen solchen Umfang, daß ca. 100 Disketten notwendig wären ...

4.6 Geschichtliches

- o 1991 kauft sich der finnische Student Linus Torvalds einen 386er PC (auf Ratenzahlung ?), um mit den Möglichkeiten des 80386 Prozessors zu experimentieren.
- o Aus den ersten Versuchen wird schnell ein einfacher, aber brauchbarer multitasking-Betriebssystemkern.
- o Zunächst wurde Linux unter Minix, einem Lehrbetriebssystem von A. Tannenbaum ("Linux is obsolete"), entwickelt.
- o Nachdem der C-Compiler »gcc« und weitere wichtige Entwicklungswerkzeuge unter Linux liefen, konnte Linux unter Linux weiterentwickelt werden.

- o Der Betriebssystemkern wurde von Grund auf neu programmiert.
- o Die Weiterentwicklung fand und findet unter Beteiligung von vielen interessierten Programmierern im Internet statt; Linus ist und war nie der einzige Entwickler._

Rüstzeug und generelle Bemerkungen

Es ist nicht unbedingt notwendig, sich jeden Punkt vor dem Weiterlesen zu merken; man muß im Zweifelsfalle nur wissen, daß er hier steht.

5.1 Allgemeine Regeln für den Umgang mit Linux

- o Alle Befehle werden ohne Nachfrage ausgeführt, also Vorsicht.
- o Regelmässig Sicherheitskopien anlegen.
- o Befehle zur Systemverwaltung sind root (siehe 15.5) vorbehalten.
- o Es wird i.d.R. zwischen Gross- und Kleinschreibung unterschieden.

5.2 Gebräuchliche Abkürzungen für Tastenkombinationen

- o Notation: Zeichen oder Wörter zwischen < und > symbolisieren eine Taste auf der Tastatur des Computers.
- o Wird eine beliebige Taste gleichzeitig mit einer der folgenden Tasten gedrückt, so spricht man von einer Tastenkombination.

Beschriftung	Abkürzungen	Bedeutung
/	shift- S-	Umschaltung auf Großbuchstaben; engl. "shift"
Strg (ctrl)	^ C-	"Steuerung", "control"
Alt	Alt- A-	????
<- ->	Tab-	Tabulatortaste (Einrückung)
<> (Diamant)	Meta- M-	????
` ` (Lücke)	SPC	Leerzeichen, engl. "space"
Esc	ESC	"Escape"-Taste
<--	BS	Backspace (linkes Zeichen löschen)

- o Wird die <Alt>-Taste nicht erkannt, so kann man diese durch Vorausschicken von <ESC> simulieren (NICHT gleichzeitig drücken).
- o Die Funktion der <Meta>-Taste wird oft von der <Alt>-Taste übernommen.
- o Die Tastenkombination werden abgekürzt durch das Voranstellen der Tastenbeschriftungen oder der Kürzel.

Bsp.: <Strg-c> (drücke Tasten <Strg> und <c> gleichzeitig)
 <C-c> ist äquivalent zu <Strg-c>
 <Strg-Alt-Entf> rebootet Linux (also Vorsicht!)

5.3 Verbreitete Funktionen einiger Tasten (-kombinationen)

<q>	<Esc>	"quit", "escape", Programm verlassen
<h>	<?>	"help", Hilfestellung anzeigen
</>	<s>	Text durchsuchen
<SPC>		eine Seite weiterblättern

<Strg-c>	Programm beenden
<Strg-d>	"end of file" (EOF), Eingabe beenden
<Strg-q>	Ausgabe fortsetzen (Anhalten mit <Strg-s>)
<Strg-l>	Bildschirm neu zeichnen (auffrischen)_

Falls einmal die Angaben in diesem Manuskript nicht auf den eigenen Dateibaum zutreffen: DON'T PANIC. Man kann sich behelfen:

5.4 Aufspüren von Dateien und Wörtern

- »find /etc -name "xinitrc" -type f -print«
sucht nach der Datei 'xinitrc' im '/etc'-Ast und zeigt alle Vorkommen mit dem kompletten Pfadnamen an; ('-type f' um nur Dateien [engl. "files"] zu durchsuchen).
- »locate xinitrc«
wie »find«, greift jedoch auf eine find-Datenbank zurück, wodurch die Suche extrem beschleunigt wird. Die notwendige Datenbank wird (als Systemverwalterin root) mit dem Befehl »su -l nobody -c updatedb« erzeugt. Nicht auf allen Systemen vorhanden, aber unbedingt vorzuziehen!
- »grep -i loadkeys /etc/rc.boot/*«
zeigt die Dateien im Verzeichnis '/etc/rc.boot' an, in denen der Befehl »loadkeys« ausgeführt wird ('-i' ignoriert Gross-/Kleinschreibung).
- »grep -i loadkeys `find /etc -name "*" -type f -print`«
ist eine Kombination von »find« und »grep«.
- »strace -eopen,stat -f -o /tmp/dump emacs«
ruft den »emacs« auf und protokolliert die Zugriffe auf (Konfigurations-)Dateien in der Datei '/tmp/dump' ('-f', um auch alle Kind-Prozesse anzuzeigen [siehe 21.4]; '-eopen,stat' für die Dateizugriffe).

5.5 Suchmuster

- o Suchmuster dienen zur Formulierung dessen, was man sucht.
Bsp.: `search*` (alle Wörter, die mit 'search' beginnen)
 - o Es gibt hauptsächlich zwei Arten von Suchmustern:
 - "stringsearch" (einfache Wortsuche)
 - "regular expressions" (reguläre Ausdrücke)
 - o Beide sind nicht 100%ig standardisiert.
 - o "stringsearch" erlaubt meist nur eine Suche nach Teilworten; ein Stern steht für beliebig viele "fehlende" Zeichen.
Bsp.: `search*` oder `*.txt`
 - o "regular expressions" erlauben sehr komplexe Suchmuster und sind unter Linux weit verbreitet.
 - o Obgiges Beispiel sieht mit "regular expressions" so aus:
Bsp.: `search.*` oder `.*\.txt`
 - o Der Punkt steht für ein beliebiges Zeichen, der Stern für "vorheriges Zeichen beliebig oft".
 - o Ein Hut `^` steht für "Anfang der Zeile", `\$` für das Ende.
 - o Dokumentation zu regulären Ausdrücken findet man mit den Befehlen »man grep«, »man regexp« (Programmiersprache TCL), »man sed«.
- Erste Schritte mit Linux

Die folgenden Ausführungen findet man auf den hinteren Seiten nochmals, und zwar mit mehr Details, als an dieser Stelle angebracht wären.

6.1 Das login-Prompt

- o Der Zugang zu einem Linux-PC ist durch einen Namen ("login") und ein Geheimwort ("password") geschützt.
- o Hat man beides eingegeben, so startet Linux eine Kommandozeile, auf der man arbeiten kann.

6.2 Die Kommandozeile

- o Ist man auf der Kommandozeile angelangt, so findet man einen schwarzen Block oder einen nervös blinkenden Unterstrich vor; beide werden als "Cursor" bezeichnet und markieren die Stelle, an der die eingetippten Zeichen erscheinen.
- o Die Kommandozeile beginnt etwa so:

```

truemper@FeilSurf:~/kurs3> _
'-----' '-----' '-----'
      |           |           |
"login-name"     |           Verzeichnis, auf das sich die
                  |           Kommandos beziehen
Name des Linux-PCs

```

- o Den Bereich vom Anfang der Zeile bis zum Cursor bezeichnet man als "Prompt". Man kann es selbst definieren, z.B.:

```
»PS1='\u@\h:\w> '«
```

('\u' = "username", '\h' = "hostname", '\w' = "working directory")

- o Die Kommandozeile wird aus historischen Gründen als "shell" ("Muschel") bezeichnet, diese Bezeichnung wird im folgenden verwendet.
- o Die shell dient zum Aufruf der Programme (siehe 19), mit denen man eigentlich arbeiten will. Ein typischer Programmaufruf sieht etwa so aus:

```

truemper@FeilSurf:~/kurs3> less -MMI vortrag3.txt
'----- Prompt -----' '---' '---' '--- Datei -'
                        |       |
                        Programmname Optionen

```

6.3 Alternativen zur Kommandozeile

- o Für Anfänger ist der »mc« ("midnight commander") eine große Hilfe; er ersetzt die Kenntnis vieler Linux-Kommandos.
- o Ein "upgrade" von alten 2.x-Versionen ist lohnenswert; siehe URL

<http://stekt.oulu.fi/~jtklehto/mc/>

6.4 Optionen

- o Die Informationen über das was, wen und wie gibt man nach dem Programmnamen an.
- o "Optionen" beginnen meist mit einem '-' und beeinflussen die Grundfunktionen des Programms.

Bsp.: »less -MMI«

- o Zwei Formate zur Angabe von Optionen sind verbreitet:

- herkömmliche Kurzform: '-Option'

Bsp.: »less -?« für das Anzeigen der Hilfe

- GNU-Format: '--Option'

Bsp.: »less --help« für das Anzeigen der Hilfe

- o Die Reihenfolge kann entscheidend sein (z.B. beim C-Compiler).
- o ' -- ' (Doppel-Minus alleine) bedeutet "jetzt folgen keine Optionen mehr"

Bsp.: »open -s -- bash -login« (siehe 16.3.4)

6.5 Argumente und Parameter

- o "Argumente" und "Parameter" sind Werte oder (Datei-)Namen, die Programmen mitgeteilt werden sollen.

Bsp.: »less vortrag3.txt«

- o ' - ' (Minus) anstelle eines Dateinamens bedeutet "die Daten werden über den Ein- oder Ausgabekanal gelenkt" (siehe 19.4).
- o Leer- und Sonderzeichen in Parametern müssen gequotet werden.

6.6 Quoting (dt.: "Anführung")

- o Unter "quoting" versteht man die Befreiung eines Zeichens von einer speziellen Bedeutung (nicht nur auf die Kommandozeile bezogen).
- o "Das Zeichen ; trennt Sätze; ein ...": Menschen können aus dem Zusammenhang erkennen, daß der erste Strichpunkt Punkt keine Sätze trennt ... dem Computer muß dies mitgeteilt werden!

- o Der Backslash ('\') wirkt nur auf das unmittelbar nachfolgende Zeichen [insbesondere dient er NICHT zum Aufbau von Verzeichnisnamen wie unter DOS].

Bsp.: »echo Das Zeichen \; trennt Sätze\; ...«

- o Anführungszeichen schützen Leer- und Sonderzeichen, die nicht der Ersetzung unterliegen.

Bsp.: »echo "Das Zeichen ; trennt Sätze; ..."«
»echo "Hallo Welt; Hallo \$LOGNAME"«

- o Apostroph-Zeichen schützen alles.

Bsp.: »echo 'Hallo Welt; Hallo \$LOGNAME'«

Grundzüge der Dateiwaltung von Linux

7.1 Dateien (engl. "files")

- o Dateien sind die Träger der Information (Texte, Bilder, usw.)
- o Zulässige Dateinamen:
 - Länge: 1-255 Zeichen
 - Format: kein spez. Format erforderlich (z.B. "DOS 8+3")
- erlaubte Zeichen: '/' ist kein zulässiges Zeichen für Datei- und Verzeichnisnamen; die Beschränkung auf a-z,A-Z,0-9,.,_,+,- ist nicht notwendig, aber sinnvoll.

7.2 Elementare Kommandos zur Handhabung von Dateien

Aktion	Prgramm	Beispiele
auflisten	»ls«	("list") »ls« »ls -l« »ls -la \$HOME more«

ausgeben	»cat«	("concatenate")	»cat /proc/cpuinfo«
kopieren	»cp«	("copy")	»cp /etc/profile .«
verschieben	»mv«	("move")	»mv profile /tmp«
löschen	»rm«	("remove")	»rm -i /tmp/profile«
klassifizieren	»file«		»file /etc/magic«
vergleichen	»diff«, »cmp«		siehe 13.2

7.3 Verzeichnisse (engl: "directories")

- o Benutzersicht: Verzeichnisse sind hierarchisch organisierte Ablagefächer für Dateien; sie können selbst wieder Verzeichnisse enthalten. Zulässige Namen: siehe 7.1.
- o Den Weg zu einem in der Hierarchie tieferliegendem Verzeichnis wird mit '/' gebildet.
Bsp.: `/usr/local/bin/'
- o Besondere Verzeichnisse:
 - .
 - .. Verzeichnis, in dem man sich gerade befindet
 - übergeordnetes Verzeichnis
 - ~ das eigene, private Verzeichnis
 - / oberstes Verzeichnis (siehe 26.X)

7.4 Umgang mit Verzeichnissen

Aktion	Prg.	Beispiele

erzeugen	»mkdir«	("make directory") »mkdir /tmp/verz«
wechseln:	»cd«	("change directory") »cd /« »cd -«, »cd +«
	»pushd«	»pushd /tmp«
	»popd«	»popd«
anzeigen	»ls«	("list") »ls -la« (list long)
verschieben	»mv«	("move")
löschen	»rmdir«	("remove directory") »rmdir /tmp/verz«
	»rm -rf«	(ohne Nachfrage und rückstandlos)_

7.5 Standard Zugriffs- und Ausführungsrechte

Abb. 1: Erläuterung der Ausgabe von »ls -ld ~«

```

Typ          Eigentümer  Gruppe  Grösse  Datum   Zeit    Name
|           |           |       |       |       |       |
|           |           |       |       |       |       |
|           |           |       |       |       |       |
d rwx r-x --- 9 truemper li-adm 1024 Mar 8 13:40 /home/truemper
|           |           |       |
|           |           |       |   +- Anzahl der "hard-links" auf diese Datei,
|           |           |       |   bzw. Anzahl der direkten Unterverzeichnisse
|           |           |       |
|           |           |       |   +++----- Rechte der Allgemeinheit ("others")
Rechte des  -+++   |       |   421
Eigentümers 421   |       |
("user")    +-----+-----+ Rechte der Gruppe ("group")
421

```

Im obigen Beispiel können nur der Eigentümer und Mitglieder der Gruppe "li-adm" in das Verzeichnis `/home/truemper' wechseln und dort auf Dateien und das Inhaltsverzeichnis zugreifen. Ausserdem

kann nur der Eigentümer neue Dateien erzeugen oder alte löschen.

- o Zugriffsrechte ändern: »chmod« ("change mode")
Bsp.: »chmod u=rwx,go=rx ~« (u=user,g=group,o=others)
»chmod 755 ~« (7=4+2+1, 5=4+1)
- o Zugriffsrechte für "zukünftige" Dateien festlegen: »umask«
Bsp.: »umask 027« (ohne weiteres nur vorübergehend
wirksam; siehe 20 und 19.2)

Kürzel	Recht	Erlaubt bei Dateien ...	Erlaubt bei Verzeichnissen ...
-		kein Recht	kein Recht
r	read	Ansehen der Datei (Leserecht)	Ansehen des Inhalts eines Verzeichnis
w	write	Verändern von Dateien (Schreibrecht)	Löschen aller Dateien, Erzeugen von Dateien
t		keine Wirkung	nur eigene Dateien können gelöscht werden (`/tmp')
x	execute	Ausführen einer Datei als Programm	Betreten des Verzeichnisses
s		siehe 22.6	Siehe »man 5 mount«, Option 'grpuid'

Kürzel Dateityp (unvollständige Aufzählung; siehe 24.X)

-	Datei	(engl.: "file")
d	Verzeichnis	(engl.: "directory")
l	Verweis	(engl.: "link")_

Tips für Anfänger

8.1 Kopieren von Dateien

- o Der Befehl »cp« kopiert Dateien innerhalb des Linux-Dateibaums.
Ganze Verzeichnisse können mit der Option '-a' kopiert werden.
Bsp.: »cp -a /etc/skel /tmp«
- o Auf fremde Maschinen innerhalb eines Computernetzwerks (z.B.
Internet) kopieren: »rcp« ("remote-copy", siehe 37.2).
Bsp.: »rcp localhost:/etc/profile . «
- o Mit »tar« kann man Archive und Backups erzeugen (nicht beschränkt
auf Bänder).
Bsp.: »tar -Mczf /dev/fd0 ~/mail«

In diesem Beispiel wird das Verzeichnis `~/mail' auf Disketten
('-M' für mehrere) gesichert und dabei komprimiert ('z').

- o "1:1"-Kopien von ganzen Datenträgern oder teilweise Kopien von
Dateien können mit »dd« erzeugt werden.
Bsp.: »dd if=/dev/fd0 of=Diskette_7«

8.2 Umgang mit DOS-Dateien

- o Der Oberbegriff für alle Werkzeuge zum Umgang mit DOS-Dateien
ist »mtools«.
Doku.: »man mtools«
- o Die Zuordnung von Laufwerksbuchstaben zu Geräten (Floppy,
Festplatte) wird in der Datei `~/etc/mtools.ref' konfiguriert.

Bsp.: »mdir a:«

- o Datenträger im DOS-Format können auch dauerhaft in den Dateibaum von Linux integriert werden (siehe 27.X).

Bsp.: »mkdir /floppy;\nmount -t msdos /dev/fd0 /floppy«

8.3 Format der Textdateien

- o Die Textdateien der verschiedenen Betriebssysteme unterscheiden sich im Zeichensatz und dem Zeilenende.

	Zeichensatz	Zeilenende	
Linux	ISOLATIN-1	LF	(= "linefeed", Zeilenvorschub [neue Zeile], <Strg-J>)
MAC		CR	(= "carriage return", Wagenrücklauf [Anfang der Zeile], <Strg-M>)
DOS	IBMP	CR LF	

- o Umwandlung mit »dos2unix«, bzw. »unix2dos« oder »recode ibmp:lat1« bzw. »recode lat1:ibmp«.

Bsp.: »recode lat1:ibmp vortrag.txt«_

Besonderheiten im Dateisystem

8.4 Gerätedateien

- o Hardware wird theoretisch nur über spezielle Dateien im Verzeichnis '/dev' angesprochen, siehe 24.1 und 25.
- o Mithilfe der Zugriffsrechte für Dateien (s. vorherige Seite) kann man also auch den Zugriff auf die Hardware bestimmten Benutzern vorbehalten. Beispiele: Modem, Floppy.

8.5 Betriebssysteminterna

- o Das proc-Verzeichnis enthält imaginäre Dateien (read- und cat-only) mit wichtigen Informationen über die Hardware, das Betriebssystem und die laufenden Programme.

Bsp.'e: »cat /proc/cpuinfo« (Hardware)
»cat /proc/version« (Betriebssystem)
»cd /proc/1; ls -l« (Programm »init«)

- o Die Datei in '/proc' belegen keinen Platz auf der Festplatte.

8.6 Verweise

- o Soll dieselbe Datei mehrere Namen erhalten oder wird eine Datei in verschiedenen Verzeichnissen gebraucht, so legt man statt einer Kopie einen Verweis (engl. "link") auf die betreffende Datei an.

Bsp.: »ls -l /usr/bin/X11/X«
»ls -l /usr/bin/mkdir«
»ls -ld /usr/adm«

- o "hardlink": ist mit Original IDENTISCH; nur für gewöhnliche Dateien und innerhalb derselben Partition möglich.

Bsp.: »echo "Hallo Welt" > hallo_welt;\"

ln hallo_welt hello_world«

- o "symbolischer link": wird zur Laufzeit ausgewertet , d.h. beim Zugriff auf den link wird überprüft, ob der Zugriff auf die original-Datei überhaupt möglich, bzw. erlaubt ist.

Bsp.: »ln -s mail Mail«

- o Es gibt "relative" und "absolute" symbolische links:

- "absolut": Verweis bzgl. Wurzel-Verzeichnis `/'
Bsp.: »ln -s /etc/profile .profile«

- "relativ": Verweis bzgl. des aktuellen Verzeichnisses
Bsp.: »ln -s ../../etc/profile .profile«_

Dateibetrachter und Texteditoren

Man will bei einem Texteditor kein WYSIWYG, sondern lediglich die reine Information erfassen, z.B. um eine Konfigurationsdatei zu ändern.

9.1 Lesen von (langen) Textdokumenten

- o Die Programme »more«, »less« und »most« dienen zum seitenweisen Anzeigen von Texten, sie sind kleiner und schneller als normale Editoren.
- o Nur »less« und »most« lassen sich intuitiv mit den Cursorstasten bedienen.
- o Mit der Umgebungsvariable 'PAGER' bestimmt man, welches dieser Programme zum Anzeigen von Text benutzt werden soll.

Bsp.: »export PAGER="less -MMI"«

- o Auch »more« wertet diese Einstellung aus und startet u.U. eins der anderen Programme.

9.2 Programme zum lesen verschiedener Dateiformate

Dateiendungen	Betrachter	Beispiel
.txt	more	»more German-HOWTO.txt«
.gz	zmore	»zmore German-HOWTO.txt.gz«
		»zcat German-HOWTO.txt.gz more«
.man	.1 man	»man -l German-HOWTO.man«
.dvi	xdvi	»xdvi German-HOWTO.dvi« (X11)
.ps	ghostview	»ghostview German-HOWTO.ps« (X11)

Dekomprimieren von `.gz'-Dateien mit dem Befehl »gunzip«

Bsp.: »gunzip German-HOWTO.dvi.gz«

9.3 Übersicht verbreitete Texteditoren (unvollständig!)

Name	Anfänger	Leistung	Besonderheiten
ae	o++	o+	nett, mit Auflistung der Tastenbelegung
ed	o--	o	Notfalleditor, einzeilig (!)

ee	o++	o+	einfach, winzig, menügeführt
elvis	o--	o++++	the king of vi's
emacs	o-	o+++++	nur Kaffekochen kann er nicht, riesig
joe	o+	o++	W*rd\$tar-kompatibel
mars_e	o+	o++	komplett in Deutsch, nur X11
pico	o+	o+	Editor des e-mail-Programms »pine«
vim	o--	o++++	vi-improved
we	o+	o++	ähnlich dem TP-Editor

9.4 »vi«-ähnliche Editoren

- o »vi« ist der Standard-Editor auf den meisten unix-artigen Systemen, er ist NICHT intiutiv bedienbar.
- o Verschiedene Ausgaben des »vi«: »elvis«, »nvi«, »vim«, ... (?).
- o Der »vi« unterscheidet zwischen Kommando- und Eingabemodus.
- o Tastenbelegung:

im Kommandomodus:

Tasten	Funktion
x	Zeichen löschen
dd	Zeile löschen
:w	Datei sichern
:q	»vi« beenden
i	in Eingabemodus wechseln

im Eingabemodus:

Tasten	Funktion
Esc	Kommandomodus einschalten

9.5 Der Editor »emacs«

- o »emacs« ist das Schweizer Taschenmesser unter den Texteditoren; er ist ebenfalls nicht intiutiv bedienbar.
- o Konfigurationsdateien sind `~/usr/lib/emacs/site-lisp/site-start.el' (globale Einstellungen) und `~/.emacs' (persönlich).
- o Jede Datei wird in ein eigenes Fenster geladen, das als "Buffer" bezeichnet wird.
- o Die vorletzte Zeile des Bildschirms (invertierte Darstellung; Bezeichnung: "mode-line") enthält Informationen über die Datei und den eingestellten Bearbeitungsmodus; Bsp.:

```
--**--Emacs: vortrag.txt          (Text Fill)--50/865--L865--19%-----
```

- o Die letzte Zeile des Bildschirms wird "minibuffer" genannt und dient der Eingabe von Kommandos oder dem Dialog mit dem Benutzer; Bsp.:

```
Save file ~/kurs3/vortrag.txt? (y, n, !, ., q, C-r or C-h)
```

- o Deutsche Kurzbeschreibung unter

<http://www.inf.tu-dresden.de/~srl/projects/emacs-refcard/>

Tasten	Funktion	Tasten	Funktion
C-g	Vorgang abbrechen	C-x C-s	Buffer speichern
C-s	Wort suchen	C-x C-c	Beenden/Verlassen
C-_	Undo	C-x C-f	Datei laden
C-w	Text ausschneiden	C-x C-i	Datei einfügen
C-SPC	Marke setzen	C-x 1	Fenster schliessen

C-y	Text einfügen	C-h m	Bearbeitungsmodus
F10	Menü (emacs-19.30)		anzeigen_

Dokumentation

Es kann nicht oft genug betont werden, daß die vorhandene Dokumentation den größten Teil der "Anfängerfragen" beantworten; meist besteht sie nämlich aus nichts anderem als "Anfängerfragen" (natürlich samt den gesammelten Antworten).

Wer diese Informationsquellen ignoriert, der macht sich selbst (!) und anderen Menschen unnötige Mehrarbeit. In diesem Sinne: RTFM (siehe 41.7)

10.1 Dateinamen und Bezeichnungen, die Hilfe versprechen

- o `FAQ': "Frequently Asked Questions" ist eine Sammlung von häufig gestellten Fragen samt Antworten
- o `HOWTO': eine überarbeitete FAQ (gleiche Information, Frage/Antwort-Charakter entfällt)
- o `Guide': ein deutlich erweitertes HOWTO
- o `README': allgemeine Hinweise zu einem Programm
- o `INSTALL': Installationsanleitung
- o `.c': Programmtext (enthält häufig unmittelbar am Anfang oder am Ende Hinweise)

10.2 Wo findet man Dokumentation?

- o Auf der eigenen Festplatte in den Verzeichnissen

`/usr/doc' (nach Programmen sortiert) und
`/usr/src/Linux/Documentation'.

- o Im WWW: einen Rundumschlag zum Thema Linux bietet das "Linux Documentation Project" (LDP); erreichbar unter den URLs:

<http://www.jura.uni-sb.de/LDP/linux.html>
<http://www.uni-paderborn.de/Linux/mdw>
<http://www.cs.TU-Berlin.DE/ftp/pub/Linux/mdw/>

- o In der newsgroup "comp.os.Linux.answers" (siehe 40.1).
- o Im Buchhandel.
- o Auf "sunsite-mirrors" (siehe 12.3) im Verzeichnis `doc'.

- o In den FAQ-Archiven z.B. unter
ftp://www.uni-paderborn.de/FAQ/comp.os.Linux.answers

In diesen Archiven findet man nicht nur die Linux-spezifische Dokumentation, sondern auch Informationen über PC-Hardware, PC-Software, andere Betriebssystemen, Programmiersprachen usw.

Die Gliederung erfolgt nach newsgroups (siehe 40.6), d.h. allgemeinere Information findet man z.B. unter

ftp://www.uni-paderborn.de/FAQ/comp.sys.ibm-pc.hardware
ftp://www.uni-paderborn.de/FAQ/comp.unix.questions
ftp://www.uni-paderborn.de/FAQ/comp.windows.x.announce_

10.3 "manual-pages":

- o "man-pages" sind per »man«-Befehl abrufbare Programmbeschreibungen.
Bsp.: »man 1 mtools« (siehe 8.2)
»tkman &« (unter X11)

- o Referenzen auf man-pages werden im Format "manpage(sektion)" angegeben.
 Bsp.: »mtools(1)«
- o Stichwortsuche in den Titeln der man-pages mit der Option '-k':
 Bsp.: »man -k user«
- o Am Schluss einer »man-page« findet man oft Beispiele und Tips, sowie die Namen der Konfigurationsdateien.
- o Thematische Gliederung der man-pages (Zahlen bezeichnen Sektion):

1: Benutzerprogramme	2: Betriebssystemroutinen
3: C Bibliotheksfunktionen	4: Geräte
5: Beschreibung (Format) der Konfigurationsdateien	
6: Spiele	7: Vermischtes
8: Systemverwaltung	n: Neues (?)

Insbesondere die Details zu den in diesem Skript mehr beiläufig erwähnten Programmnamen kann man in den man-pages nachlesen.

10.4 Programmparameter, die eine Kurzbeschreibung anzeigen

- o '-?', '-h', '--help', sind die verbreitetsten Optionen.
 Bsp.: »ls --help«
- o '-v' oder '-V' spucken meist die Versionsnummer aus.
 Bsp.: »gcc -v«
- o Manche Programme zeigen die Kurzbeschreibung bei Fehlbedienung automatisch an.

10.5 GNU-TeXinfo

- o "GNU-TeXinfo" ist eine auf Hypertext basierende Dokumentation.
- o Die Bedienung ist nicht intuitiv; die eingebaute Hilfe sollte als erstes gelesen werden ("Alt-x describe-mode <return>").
- o Start des Programms mit dem Befehl 'info' oder im Editor 'emacs' mit <Strg-h> i.

10.6 Bücher und Zeitschriften

Eine Liste mit deutschsprachiger Literatur rund um Linux ist erhältlich unter

<http://home.pages.de/~joey/Linux/Dynamic/Buecher/>

Darüberhinaus erscheinen mir erwähnenswert:

- o "Linux-Journal", SSC Inc. (Verlag), engl., monatlich (sehr breites Spektrum, oft verkannt)
- o "Essential System Administration" (Frisch)
- o "Linux Installation and getting started" (Welsh)
- o "Running Linux" (Welsh/Kaufmann)
- o "Crash-Kurs Linux" (Trümper)_

Erklärende Übersetzung einiger englischer Vokabeln

Die englische Original-Dokumentation verwendet viele Wörter, die nicht in Wörterbüchern zu finden sind, bzw. die dort nicht mit der im technischen Zusammenhang üblichen Bedeutung zu finden sind.

Die folgende Übersicht soll eine kleine Hilfestellung geben.

Die im Internet gebräuchlichen Akürzungen sind im Abschnitt 41.7 wiedergegeben.

Englisch	dt. Übersetzung mit Erklärung
abort	Programm oder Verbindung abbrechen; "Press ctrl-c to abort"
ambiguous	mehrdeutig
announcement	Ankündigung
argument	Wert (Einstellung), der einem Programm übergeben wird
bracket	eckige Klammer [oder]
branch	Zweig, Alternative
character	Zeichen
comment	Kommentar
compile	Übersetzen; z.B. "C-Programme in ausführbaren Maschinencode compilieren"
concatenate	hintereinanderhängen
consider	betrachten
deadlock	nicht aufhebare Blockierung; etwas hat sich festgefressen
default	Vorgabewert, Standardeinstellung
device	Gerät; der Zugriff auf Geräte erfolgt unter Linux fast ausschließlich über Dateien, die sogenannten Gerätedateien im Verzeichnis <code>/dev</code> ; meist bezieht sich "device" auf eine dieser Dateien
directory	Verzeichnis
environment	Umgebung; bezieht sich auf die Einstellungen, die einem Programm mit Umgebungsvariablen (environment variables) übergeben wird
execute	ausführen (eines Programms oder Befehls)
file	Datei
file descriptor	Dateizugriffsnummer; programminterne Nummer für eine Datei
font	Zeichensatz
forward	weiterleiten (meist im Zusammenhang mit e-mail)
header	Vorspann
host	Maschine, Computer, Rechner
I/O	Ein-/Ausgabe
interface	Schnittstelle zwischen <ul style="list-style-type: none"> - Mensch und Maschine (z.B. grafische Oberflächen) - Netzwerkkarte und Betriebssystemkern
kernel	Betriebssystemkern; der heilige Teil von Linux
key	Taste
keyboard	Tastatur
library	Zusammenfassung von kompilierten C-Routinen zu einer Bibliothek, die dann aus jedem anderen Programm heraus benutzt werden kann
link	Verweis; anstatt eine Datei zu kopieren, kann man auch einen Verweis anlegen; dies spart Platz und erlaubt einige Tricks
load	laden, Beladung, Ausnutzung; in Zusammenhang mit der Auslastung einer Maschine gebraucht
lock	Sperre, Riegel (meist gewollt)
match	Übereinstimmung
mount	montieren, zusammensetzen; stets im Zusammenhang mit dem Verzeichnisbaum, der durch das Kommando <code>»mount«</code> zusammengesetzt wird
network	Computernetz(werk)
news	
OS	Betriebssystem (Operating System)
obsolete	überflüssig, veraltet
overflow	bei einer mathematischen Operation ist ein Ergebnis aufgetreten, daß wegen seiner Größe nicht mehr in die dafür vorgesehene Variable paßt
parenthesis	runde Klammer

path	Pfad; meist als Begriff für den kompletten Verzeichnisnamen zu einer Datei gebraucht
pipe	Röhre, Veranschaulichung der Weiterleitung von Daten von einem zum nächsten Programm; das sogenannte "pipe-symbol" ist das Zeichen (senkr. Strich)
port	
preference	Vorlieben (bei Konfiguration)
press	Aufforderung zum Drücken einer Taste
prompt	Bereitschaftsmeldung (oder -zeichen)
proprietary	
quote	Anführungszeichen oder Zitat
range	Bereich; von ... bis
remote	im Sinne von "die Gegenseite der Verbindung"
reply	antworten (auf eine e-mail oder einen Artikel)
resolve	Ermittlung (Auflösung) der numerischen IP-Adresse, die zu einem Maschinennamen gehört
sequence	Sequenz, Aufeinanderfolge
server	Computer, der anderen Computern Dateien oder Dienstleistungen zur Verfügung stellt
socket	
source	meist Begriff für die Programmtexte (Quellcode, Sourcecode) benutzt; "may the source be with you"
space	Leerzeichen, Platz ("disk space, the final frontier")
stack	Stapel
string	Zeichenkette
subject	Thema, Betreff (bei e-mails)
synopsis	zusammenfassende Übersicht
template	Vorlage, Muster, Schablone, Schema
trace	Ablaufverfolgung
type	Tippen Sie ein
underflow	Unterlauf, z.B. "der Unterlauf des Rheins" ;-)
usage	Gebrauch, Gebrauchsanleitung
verbatim	wortwörtlich
wildcard	Joker, meist erfüllt der Stern * diese Funktion, z.B. zur unvollständigen Angabe von Dateinamen_
Beschaffung von Software	

An einigen Stellen erwähne ich Software, ohne genauer zu sagen, woher man diese beziehen kann und wie man diese installiert. Eine erfolgsversprechende Antwort wäre im Einzelfall etwa so lang wie die folgenden zwei Seiten. Die einzige Möglichkeit zur Lösung des Dilemmas besteht also in der Offenlegung des dahinterstehenden, allgemeinen Prinzips.

12.1 Überblick über die verfügbare Software

- o "Linux Documentation Project" (LDP) (siehe 10.2)
- o Bunte Zusammenstellungen von Hinweisen

<http://www.fokus.gmd.de/linux/linux-prog.html>

- o Naturwissenschaften, Mathematik, Informatik

http://chpc06.ch.unito.it/Linux/Linux_old.html
<http://www.fu-berlin.de/~clausi>

- o Hochleistungs-Systeme mit Linux

<http://www.cs.cornell.edu/home/mdw/hpc/hpc.html>

- o "The Linux Midi + Sound Pages"

<http://www.digiserve.com/ar/linux-snd/>

- o Die "Linux Software Map" (LSM) ist eine allgemeine, aber unvollständige Übersicht mit einheitlichen Beschreibungen den Programmpaketen, man findet sie
 - in Dateiform als "LSM.Datum.gz" auf sunsite-mirrors
 - im WWW als Datenbank, z.B. unter der URL
 - <http://www.boutell.com/lsm/lsmbykey.cgi>
 - <http://www.iki.fi/liw/linux/cola.html>
 - <http://www.bib.rwth-aachen.de/lsm.html>
- o Einen Überblick über die erhältlichen kommerziellen Programme gibt das 'Commercial-HOWTO'.

12.2 Ankündigungen von Neuerungen und Korrekturen

- o Ankündigungen von allgemeinem Interesse kann man ca. wöchentlich in der newsgroup "comp.os.Linux.announce" nachlesen (siehe 40.6)
- o Die e-mail Verteiler der Entwicklergruppen (siehe 39.X) enthalten Hinweise auf Programmteile, die noch nicht für die allgemeine Benutzung geeignet sind._

12.3 Fundorte für Software

- o Software wird im Internet auf sogenannten "ftp-Servern" gelagert (siehe 37.5).
- o Folgende ftp-Server haben für die Linux-Gemeinde besondere Bedeutung:

sunsite.unc.edu	die Sammelstelle für Linux-Software schlechthin
tsx-11.mit.edu	ALPHA- und BETA-Software (siehe 13.4)
ftp.funet.fi	neuste Kernelversionen und -patches
- o Viele ftp-Server in Deutschland spiegeln diese Softwarearchive (mit nächtlicher Aktualisierung); Bezeichnung: "mirror-sites".
- o Ausserdem sind die Linux-Bereiche dieser ftp-Server auch auf CD-ROMs gepreßt im (Buch-)Handel erhältlich.
- o Nicht alles findet man auf tsx-11 oder sunsite; das ganze Internet ist voll mit Programmen, die nicht speziell für Linux geschrieben sind, die aber einwandfrei unter Linux laufen.

12.4 Zugriff auf ftp-Server

- o Solange man die brandaktuellen Versionen nicht wirklich benötigt, sollte man das Internet schonen und CD-ROMs verwenden.
- o Der Zugriff auf FTP-Server ist besonders einfach mit den Programmen »mc« ("midnight commander", <F9> "FTP link") oder »netscape«.
- o Falls diese nicht zur Verfügung stehen: »ftp« benutzen.

Aufruf: "ftp FTP-Server"

Name zum Anmelden: "ftp" oder "anonymous"

Password: die eigene Email-Adresse

Kommandozeile: »dir« Dateinamen auflisten

»cd« Verzeichnis wechseln

"get zB.tgz" Datei `zB.tgz' herunterladen

```
"put zB.tgz"      Datei `zB.tgz' aufladen
»quit«           ftp-Sitzung beenden
```

- o Internet-Knigge: nach Möglichkeit nur nahe FTP-Server benutzen.

12.5 Navigation und Suche

- o Dateien mit Namen `ls-lR' enthalten eine Liste der Namen aller vorrätigen Dateien auf einem ftp-Server.
- o Um im Internet komfortabel nach Software zu suchen, eignen sich die Programme »archie« oder »xarchie«; diese basieren auf Inhaltsverzeichnissen vieler großer ftp-Server.
Bsp.: »archie -h archie.th-darmstadt.de mc |more«
»xarchie &«_

Installation von Software

13.1 Auspacken von "roher" Software

- o Programmpakete sind i.d.R. mit »tar« gepackt und mit »gzip« komprimiert, und enden mit ».tar.gz' oder kurz ».tgz'; dieser Typ von Dateien kann mit dem Befehl »tar xzvf« ausgepackt werden.

```
Bsp.: »tar xzvf dosemu0.60.tgz«
      ' :: `
Datei auspacken -' :: ` - die folgende Datei (file)
dekomprimiere ---' `--- sag was du machst (verbose)
```

(Nur Testen: statt 'x' ein 't' verwenden.)

- o Die weniger weit verbreitete Verpackungsart »shar« ("shell archive") wird mit dem Befehl »unshar« entpackt.
- o entpacken in den Verzeichnissen `/usr/local/src' oder `~/src'; evtl. vorher Unterverzeichnisse anlegen

13.2 Korrekturen und Flickwerkzeug

- o Sind (kleinere) Korrekturen an Programmen notwendig, so werden meist nur die notwendigen Änderungen am Quelltext verteilt, nicht das komplette Programm (drastische Reduzierung der Netzbelastung).
- o Diese Korrekturen werden als "patches" (dt. "Flicken") bezeichnet.
- o Erzeugung von »patches« mit »diff«:

```
Bsp.: "diff -u Datei.alt Datei.neu > Unterschiede"
```

- o Anwenden der patches mit dem Programm »patch«:

```
Bsp.: "cat Unterschiede | patch -p0"
```

13.3 Das Programm »make«

- o Der überwiegende Teil der Software für Linux liegt im Quelltext ("roh") vor und muß i.d.R. noch übersetzt werden ("kochen").
- o Das Kommando »make« übersetzt/erstellt automatisch (nur) die fehlenden Teile eines Programms und fügt sie zusammen.
- o Wie was zu tun ist erfährt »make« aus der Datei `Makefile'.
- o Liegt dem Programm kein `Makefile' bei, so gelten die Regeln:
 - Für X11-basierte Programme wird das `Makefile' automatisch durch den Befehl »xmkmf« erzeugt.
 - Für GNU-Programme kann ein `Makefile' automatisch durch den Befehl »./configure« erstellt werden.

- o Meist reicht das Kommando »make«, um ein Programm so weit zu bauen, daß man es testen kann.
- o Die genauen Schritte entnimmt man `INSTALL' und/oder `README'.
- o Durch »make install« wird das Programm i.d.R. installiert.
- o Empfohlene Einstellungen im Makefile: "PREFIX=/usr/local" oder "BINDIR=/usr/local/bin" (weitere entsprechend)_

13.4 Entwicklungsstadien

- o Grundsätzlich ist keine Software fehlerfrei.
- o "ALPHA" ist ein Hinweis auf Software, die noch in der Entwicklung steckt und noch nicht ausreichend getestet worden ist; die Wahrscheinlichkeit von schwerwiegenden Programmfehlern leicht erhöht. (Nicht zu verwechseln mit der Computerarchitektur gleichen Namens.)
- o "BETA" kennzeichnet Software, die das ALPHA-Stadium (hoffentlich) hinter sich hat und nun einer intensiven Testphase unterzogen wird.
- o "stabil" (engl. "stable") wird Software genannt, wenn die BETA-PHASE hinreichend lang war und innerhalb eines gewissen Zeitraums keine neuen Fehler gefunden wurden.

13.5 Der GNU C-Compiler "gcc"

- o Um aus C-Quelltexten ein lauffähiges Programm zu produzieren, steht auf den meisten Linux-Systemen der GNU C-Compiler zur Verfügung.

```
Bsp.: "gcc -O2 -I/usr/X11R6/lib -L/usr/X11R6/include
      -o zB zB.c"
```

Option Beschreibung

```
-----
-O      Optimierung des erzeugten Programms (Werte: 0, 1, 2 ,3)
-o      Name der Ausgabedatei (des lauffähigen Programms)
-I      Verzeichnis mit zusätzlich benötigten header-Dateien
-L      Verzeichnis mit zusätzlich benötigten Programmbibliotheken
-l      Archivname der zusätzlich benötigten Programmbibliotheken
      (Bsp.: `libc.a' <-> Archiv `c' <-> Option '-lc')
```

- o Die Vereinbarung von Konstanten, Variablen, Funktionen, usw. wird meist in sogenannten "header-Dateien" vorgenommen (enden mit `.h' und liegen meist in "include"-Verzeichnissen).
- o Während des Übersetzens werden die header-Dateien durch "#include"-Anweisungen in den Quelltext eingebunden.
- o Nur die Verzeichnisse `/usr/include' und `/usr/local/include' werden vom gcc automatisch nach solchen Dateien durchsucht.
- o Um das Rad nicht jedesmal neu erfinden (lies: "programmieren") zu müssen, sind häufig benutzte Funktionen in fix und fertig kompilierten Archiven verfügbar.
- o Diese Archive nennt man auch "Bibliotheken" (engl. "libraries"), sie werden vom gcc in den Verzeichnissen `/lib', `/usr/lib' und `/usr/local/lib' erwartet und tragen die Endung `.a' oder `.sa'.
- o Es gibt eine speziell für 586er-Prozessoren optimierte Version des gcc unter der URL
<http://www-iss.mach.uni-karlsruhe.de/pcg/>
- o die meisten Fehlermeldungen entstehen durch fehlende '-I', '-L', oder '-l' Optionen, z.B.: "undefined reference to"
- o "ld: Output file requires shared library `libc.so.4'" läßt sich beheben, indem man die Option '-g' entfernt oder '-lc' hinzunimmt._

Lebenszyklus eines Linux-Systems

Alle Programme und Vorgänge werden am Ende des Manuskripts im Detail besprochen.

Glossar:

- o "booten": Start eines Betriebssystems
- o "hochfahren": umgangssprachlicher Ausdruck für booten
- o "Sektor": 512 byte großer Datenblock; kleinste Einheit zur Organisation eines Datenträgers

14.1 Der Instinkt der Hardware

- o Einschalten des Computers
 - alle Komponenten gehen in einen Ausgangszustand
 - Aktivierung der BIOS-Routinen
- o BIOS (Basic Input/Output System)
 - in einem Chip auf der Hauptplatine des PCs dauerhaft gespeichert
 - Konfiguration durch das "BIOS-Setup"
 - lädt automatisch die Anweisungen aus dem allerersten Sektor der Diskette oder der Festplatte

14.2 Der erste Sektor eines Datenträgers

- o Hier befinden sich Anweisungen zum Start eines Betriebssystems.
- o Linux kann in diesen Bereich den Boot-Manager »lilo« ("Linux Loader") installieren, der die Wahl zwischen verschiedenen Betriebssystemen erlaubt (siehe 53.X).
- o Bedienung des Boot-Managers »lilo«:
 - Taste <Alt> zum Aktivieren des Boot-Prompts
 - Taste <Tab> listet die Namen der bootbaren Betriebssysteme auf (vom Benutzer konfiguriert)
 - mit Parametern läßt sich der Bootvorgang beeinflussen
- o Linux kann auch durch die Boot-Manager der anderen Betriebssysteme gebootet werden.



(Die Diskette soll in dieser Abbildung symbolisch für einen Datenträger stehen. Details Seite 53)_

14.3 Der Linux-Betriebssystemkern

- o Ziel der vorhergehenden Aktionen ist es, den Betriebssystemkern von Linux in den Arbeitsspeicher zu laden und zu starten.
- o Der Betriebssystemkern soll alle Ressourcen des Computers (Rechenleistung, Arbeitsspeicher, Netzwerkkomponenten) zwischen

- den konkurrierenden Programmen (-> multitasking) aufteilen.
- o Dazu muß der Betriebssystemkern diese Ressourcen suchen, finden und initialisieren; die meisten Meldungen rühren daher.
- o Mit dem Programm »dmesg« kann man diese Meldungen nach dem Booten nochmal anzeigen lassen.
- o Nachdem dies erfolgt ist, wird von einer bestimmten Partition ("root-Partition") das Programm »init« ausgeführt.

14.4 Das Ur-Programm »init«

- o »init« ist Vorfahre aller anderen Programme.
- o Die eigentliche Arbeit überlässt »init« shell-Skripten in `/etc/init.d/` oder `/etc/rc.d/`; diese entsprechen einer verzweigten `autoexec.bat` von DOS.
- o Wichtige Vorgänge sind:
 - Überprüfung der Festplatte; sind deren Daten stimmig?
 - Zusammenfassung aller lokaler (physikalisch) Speichermedien zu einem großen Dateibaum
 - Aktivierung von swap (Auslagerungsdatei bei nicht ausreichendem Arbeitsspeicher [RAM])
 - Laden zusätzlicher Treiber, sogenannter Module
 - Aktivierung des Netzwerks
 - Belegung der Tastatur und Einstellung einer Schriftart
 - Starten wichtiger (ständig aktiver) Systemprogramme
- o Ist das System bereit, so erscheinen auf den Textkonsolen login-Aufforderungen und evtl. wird die graphische Oberfläche gestartet.

14.5 Laufender Betrieb

- o Was jetzt passiert ist Sache des Benutzers.

14.6 Vor dem Abschalten des Computers

- o Spätestens nach 497 Tagen ununterbrochenem Betrieb muß Linux angehalten werden (Überlauf des 32-bittigen jiffies-Zählers).
- o Während des Betriebs werden nicht alle Daten sofort bei ihrer Erzeugung auf der Festplatte gespeichert; man muß dies also vor dem Ausschalten sicherstellen.
- o Dies geschieht mit den Programmen »halt« oder »reboot« bzw. im Mehrbenutzer-Betrieb mit »shutdown now«.
- o Der "Informatikergriff" <Strg-Alt-Entf> kann auch benutzt werden.

Benutzer und Gruppen

15.1 Benutzer

- o Linux unterscheidet die Benutzer des Computers anhand eines max. 8-stelligen Namens, der "login-id" (oder auch "username").
- o Vor dem Zugriff auf ein Linux-System muß sich der Benutzer mit der login-id und einem Geheimwort ("password") ausweisen.
- o Eine Zugangsberechtigung zu einem Linux-Rechner wird "account" (dt. "Konto") genannt.
- o Die Daten eines Benutzers sind mit dessen Namen versehen und können so von den Daten anderer Benutzer unterschieden werden.
- o Jedem Benutzer ist ein privater Bereich im Verzeichnisbaum zugewiesen, das sogenannte home-Verzeichnis (engl.: "home-directory")
 - Bsp.: »echo \$HOME«

- o Arbeit in einem Fenster beenden mit den Befehlen »logout« oder »exit«.

15.2 Programme zur Manipulation der account-Daten

Programm	Beschreibung
»passwd«	password ändern
»chsh«	"shell" ändern ("change shell")
»chfn«	vollen Benutzernamen ändern (»change full name«)

- o Die Auswahlmöglichkeiten für »chsh« werden durch die Datei `/etc/shells` begrenzt.
- o Neben dem vollen Benutzernamen kann mit »chfn« auch noch die Telefon- und Raumnummer des Büros und die private Telefonnummer abgespeichert werden (im "GECOS-Feld").
- o Evtl. wird die Wahl des passwords auf "sichere" Geheimworte begrenzt.

15.3 Die Benutzerdatenbank `/etc/passwd`

- o Die Benutzerdatenbank `/etc/passwd` ist von jedermann lesbar.
- o Neben der login-ID bekommt der Benutzer noch zwei (mehr oder weniger willkürliche) Nummern zugewiesen, die "user-ID" und die "group-ID".
- o Für interne Zwecke wird stets die "user-ID" verwendet, nicht die login-ID.
- o Das GECOS-Feld ist für Kommentare gedacht und enthält meist den vollen Benutzernamen.
- o Zentrale Verwaltung der Benutzerdatenbank bei mehreren Rechnern mit "NIS" (vormals "YP") oder "NYS" (Linux-Version von "NIS+").
- o Beispiel eines Eintrags in `/etc/passwd`:

```
truemper:LkesGuNvQ:998:970:Winfried Truemper,,,,:/home/truemper:/bin/bash
|           |           |           |           |           |
login-ID    password    user-ID    group-ID    GECOS-Feld    shell
|           |           |           |           |
home-Verzeichnis_
```

15.4 Gruppen

- o Gruppen dienen dem gemeinsamen Zugriff mehrerer Benutzer auf dieselben Dateien (bzw. Geräte).
- o Namen von Gruppen dürfen ebenfalls nur 8 Zeichen lang sein (eine häufige Fehlerquelle).
- o Jeder Benutzer ist automatisch Mitglied der in der Datei `/etc/passwd` angegebenen Gruppe.
- o »chgrp« (change group; Gruppenzugehörigkeit einer Datei ändern; nur ausführbar von root oder Mitgliedern der Gruppe).
- o Zusätzliche Mitgliedschaften werden in der Datei `/etc/groups` eingerichtet; das Format ist:

```
floppy:*:25:truemper,winni
|           |           |           |
Name der Gruppe    login-IDs der Mitglieder
|           |           |
password    group-ID (interne Nummer der Gruppe)
```

15.5 Die Systemverwalterin "root"

- o Die Systemverwalterin root (auch "Superuserin" genannt) darf ALLES, sie hat die user-ID 0 und die group-ID 0 und wird unter der

login-id "root" geführt.

- o Insbesondere die Konfiguration des Systems ist root vorbehalten, dazu gehören:
 - Anlegen von Benutzern und Gruppen
 - Konfiguration von Hardware
 - Festlegung der Betriebsparameter
- o Die meisten Programme in den Verzeichnissen `/sbin`, `/usr/sbin` und `/usr/local/sbin` sind nur von root ausführbar; aus diesem Grund sind diese Verzeichnisse bei gewöhnlichen Benutzern auch nicht im Suchpfad für ausführbare Programme (`$PATH`, siehe 19.6) enthalten.
- o Programme zur Benutzerverwaltung

Name	Beschreibung
<code>>chown<</code>	Eigentümer einer Datei ändern ("change owner") Bsp.: <code>>chown -R mueller /home/mueller<</code>
<code>>adduser<</code>	neuen Benutzer anlegen
<code>>groupadd<</code>	neue Gruppe anlegen
<code>>sudo<</code>	begrenzte Vergabe von root-Rechten an Benutzer

15.6 Benutzerstatistiken und -informationen

- o wer macht was, wo und seit wann: `>w<`, `>who<`
- o wer war zuletzt da: `>last -n 10<`
- o der gläserne Benutzer: `>finger<`
Bsp.: `>finger -l $LOGNAME<`
- o `>acct<`, `>sac<`: Auslastung des Rechners durch einen Benutzer über einen bestimmten Zeitraum
- o `>pacct<`: wieviele Seiten hat ein Benutzer gedruckt?_

Bildschirm und Tastatur

16.1 Das Problem

- o Jedes Programm und jeder Benutzer braucht eine Ein- und Ausgabemöglichkeit, d.h. multiuser/multitasking erfordert mehr Tastaturen und Monitore, als physikalisch vorhanden sind.
- o Unter Linux kann man bis zu 4 VGA-Monitore an den PC anschliessen, sowie serielle Terminals, aber selbst das würde nicht ausreichen, wäre zu teuer, unkomfortabel und zu aufwendig.
- o Die Lösung: virtuelle Konsolen und (virtuelle) Terminals.

16.2 Virtuelle Konsolen

16.2.1 Grundlegendes

- o Die "Konsole" ist die Kombination aus direkt an die Workstation angeschlossener Tastatur und dem Monitor.
- o "virtuelle Konsolen" sind den Programmen vorgetäuschte Konsolen, die vom Benutzer wahlweise auf der Konsole eingeblendet werden können.
- o Eigenschaften:
 - hin- und herschalten mit `<Alt-Ctrl-F1>` - `<Alt-Ctrl-F8>`
 - Zurückblättern mit `<Shift-Bild>` möglich (aus Sicherheitsgründen nicht mehr nach Wechsel der Konsole)
 - Cut & Paste möglich mit "Maustreiber" `>gpm<`
 - virtuelle Konsolen sind text- und grafikfähig
 - VESA Powersave-Modus wird unterstützt (siehe Datei

```
`/usr/src/Linux/drivers/char/vesa_blank.c')
```

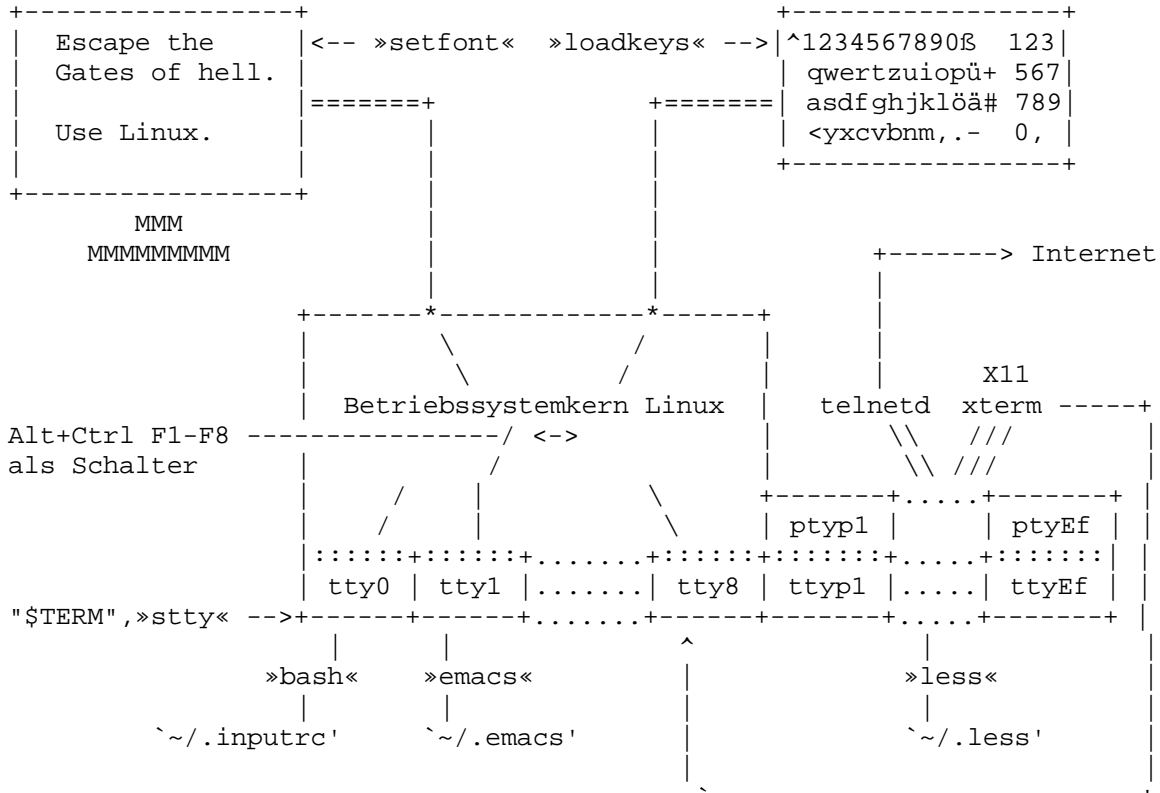
- o zugehörige Gerätedateien sind `/dev/tty0' - `/dev/tty8' (siehe 24.1)
- o Linux kann maximal 63 virtuelle Konsolen verwalten, vorkonfiguriert sind aber nur 8; beides reicht ebenfalls noch nicht aus.
- o Aktivierung der login-Aufforderungen in `/etc/inittab'
- o Programme auf "leeren" (unbenutzten) virtuellen Konsolen starten mit dem Programm »open« (Paket `dynamic-vt-1.1.tgz')
Bsp.: »open -sl«

16.2.2 Der Bildschirm

- o Der Befehl »setfont« erlaubt die Wahl einer anderen Schriftart durch Laden einer Datei aus dem Verzeichnis `/usr/lib/kbd/consolefonts'.
Für Deutschland wählt man `latlu-16.psf'.
Bsp.: »setfont latlu-16.psf« (`/etc/rc.boot/console')
- o Höhere Auflösungen im Textmodus durch
 - den Eintrag "vga=ask" in `/etc/lilo.conf' (siehe 53.X)
 - das Programm »resizecons« (Experimente notwendig)
 - das Paket "SVGATextMode" (evtl. flimmerfreies Bild)
- o Photo eines Textbildschirms ("screen shot") möglich (siehe 25.6)
Bsp.: »cat /dev/vcs1 > screen_dump«

- o Erscheinungsbild der virtuellen Konsolen ändern: »setterm«

Bsp.: »setterm -foreground green -store -clear«_



16.2.3 Die Tastatur

- o Der Befehl »loadkeys« lädt eine Tastaturbelegung aus dem Verzeichnis `/usr/lib/kbd/keytables'.
- o Die Einstellungen durch »loadkeys« betreffen alle virtuellen Konsolen [insbesondere auch das "X-Window-System"].

- o für deutsche PC-Tastaturen: `de-latin1-noadkeys' oder `de'
Bsp.: »loadkeys de-latin1-noadkeys«

16.3.1 Terminals

- o Zur besseren Ausnutzung der sündhaft teuren Großrechner wurden früher massenhaft "intelligente" Textmonitore an jene angeschlossen; diese bezeichnet man als "Terminals".
 - o Als Terminals verhalten sich heutzutage:
 - die Linux-Konsole (mit erweiterten Eigenschaften ...)
 - Fenster unter X11 (Programme »xterm« oder »rxvt«)
 - »telnet«-Sitzungen von fremden Rechnern
 - »screen«-Sitzungen
 - o zugehörige Gerätedateien: `/dev/tty0' - `/dev/ttyEf' (siehe 24.1)
- [???? Controlling Terminals, Daemonen]_

16.3.2 Konfiguration der Terminaleigenschaften

- o Der Typ des Terminals wird durch die Umgebungsvariable "TERM" (siehe 19.6) angezeigt; gängige Werte sind:

"linux":	die Linux-Konsole
"xterm":	Fenster mit Kommandozeile unter X-Windows
"vt100":	Standardwert ("grüner Textmonitor")
- o »stty«: grundlegende Eigenschaften konfigurieren
Bsp.: »stty -a«
»stty rows 10 cols 40« (10 Zeilen, 40 Spalten)
»stty erase ^? susp ^z start ^q stop ^s werase ^w«
»stty erase ^h« (falls <-| nicht funktioniert)
- o Terminal für Nachrichten sperren: »mesg n« (dieses Kommando ändert die Zugriffsrechte an der zugehörigen Gerätedatei)

16.3.3 Einstellungen zurücksetzen

- o Das Kommando »reset« versetzt ein Terminal in einen "brauchbaren" Zustand (notfalls "blind" eintippen).
- o Drücken der Tastenkombination <Strg-L> frischt den Bildschirm auf.
- o »clear« löscht den Bildschirm (Vorsicht: nur der sichtbare Ausschnitt wird gelöscht, nicht nach oben herausgerutschte Zeilen).

16.3.4 Zubehör

- o Virtuelle Terminals: das Programm »screen« stellt die gleichen Funktionen wie die Linux-Konsole zur Verfügung (oder war es umgekehrt?); darüberhinaus kann man es "schlafen" legen oder an ein anderes Terminal "umziehen".
- o Mit »ttsnoop« kann root Benutzer ausspionieren (der Gebrauch dieses Programms kann strafbar sein).

16.4 Programmierung

- o Die Fähigkeiten des Terminaltyps sind beschrieben durch
 - die Datei `/etc/termcap' ("Terminal Capabilities") und
 - die alphabetisch in Verzeichnissen geordneten Beschreibungen unter `/usr/lib/terminfo'.

- o Jedem Programm ist es selbst überlassen, ob und mit welcher Methode es diese Informationen auswertet.
- o `~/inputrc' konfiguriert "readline"-basierte Programme.
- o "ncurses": C-Bibliothek zur Bildschirmsteuerung
- o »tput«: elementare Bildschirmsteuerung aus shell-Skripten heraus
- o »dialog«: Dialog-Boxen für shell-Skripte (hübsch, aber für komplexere Programmen zu unkomfortabel in der Bedienung UND der Programmierung)

Bsp.: »dialog --yesno "Ja oder Nein?" 7 30«_

Der "login"-Vorgang

17.1 Grundlegendes

- o Vor dem Zugriff auf einen Linux-Rechner muß sich der Benutzer mit einem Namen und einem Geheimwort ausweisen (es gibt aber noch andere Verfahren).
- o Dieses Ausweisen nennt man in der Umgangssprache "einloggen"; engl. "logging in" oder kurz "log in" (dt.: "log Dich ein").
- o Der umgekehrte Vorgang, das Abmelden (nie vergessen!) wird als "ausloggen" oder "log out" bezeichnet.

17.2 Das login-Prompt

- o Das login-Prompt ist die Aufforderung an den Benutzer, die login-id einzugeben.

Bsp.'e: »FeilSurf login: « (Konsole/telnet)
 »annex username: « (Terminalserver)
 »Name (localhost:truemper): « (FTP)

- o Programme zur "Erzeugung" von login-Prompts

Name	Begrüßungstext	Beschreibung
getty	/etc/issue	Standard für Textkonsolen
mgetty	/etc/issue	Fax, PPP oder login via modem
xdm	/etc/X11/xdm/Xresources	X11-login (siehe 45.X)
telnetd	/etc/issue.net /etc/issue	telnet-login (Tür zum Internet)
ftpd	/etc/ftpd/ftpaccess	

- o Die gettys werden in der Datei `/etc/inittab' aktiviert und von »init« verwaltet.
- o Ist der Benutzernamen eingegeben, so wird i.d.R. das Programm »login« gestartet.

17.3 Das Programm »login«

- o »login« führt alle notwendigen Schritte zum Anmelden des Benutzers durch (Konsole und telnet); von "getty" und Verwandten erhält es den Namen des Benutzers.
- o login-Zugriffssperren:

- `/etc/usertty' [Wird die Datei überhaupt ausgewertet?]
 - `/etc/passwd' Gibt es eine Benutzer mit diesem Namen?
 - `/etc/nologin' Warum dürfen sich keine Benutzer einloggen?
(Kann zum Sperren des Systems verwendet werden.)
 - `/etc/securetty' Darf sich root an diesem Terminal einloggen?_
- o Autorisierung: Abfrage des passwords durch »login«
("password-prompt").
 - `/etc/passwd' Stimmt das password? (Vergleich der verschlüsselten Versionen)
- o Diverses
 - `~/hushlogin' Will der Benutzer ein login ohne viel Tam-Tam?
- o Logbücher: vermerken, daß sich der Benutzer eingeloggt hat
 - `/var/run/utmp' Logbuch
 - `/var/log/wtmp' Logbuch
 - `/var/log/lastlog' Logbuch
- o Abschliessende Aktionen
 - Übereignung des Terminals an den Benutzer
 - Ausgabe der Datei `/etc/motd' ("Message of the day")
 - Start der in `/etc/passwd' angegebenen shell

"passwords"

18.1 Grundlegendes zu passwords

- o Unter Linux wird grundsätzlich nur mit den verschlüsselten Versionen der passwords gearbeitet - das original-password kennt nur der Benutzer.
- o Das password wird erst nach der Eingabe von »login« verschlüsselt, d.h. bei login-Vorgängen via Internet wird es zunächst im Klartext übertragen.
- o Da die Datei `/etc/passwd' von jedermann lesbar ist, sind es auch die verschlüsselten passwords.
- o Durch Verschlüsseln von Wörtern und anschließendem Vergleich kann man passwords also ERRATEN; aus diesem Grund gibt es "einfache" und "schwere" passwords.
- o "shadow-passwords" ist ein Verfahren zum Aufbewahren der (verschlüsselten!) passwords in einer nicht öffentlich lesbaren Datei (siehe `Shadow-Password-HOWTO').

18.2 Verschlüsselungstechnik

- o Die Standard-Verschlüsselungsverfahren benutzen max. 8 Zeichen des eingegebenen passwords, d.h. "Grundlage" und "Grundlagen" werden nicht unterschieden.
- o Von jedem Zeichen werden nur die ersten 7 Bit verwendet, d.h. "Määnder" und "Mdander" sind gleich.
- o Somit ergeben sich unter Abzug der nicht eintippbaren Zeichen '^@' und '^?' nur $126^8 = 63.527.879.748.485.376$ Möglichkeiten.
- o Das verschlüsselte password ist stets 13 Zeichen lang und kann sogar in passwd-Dateien anderer Rechner verwendet werden.
- o weitere Informationen siehe "crypt(3)" (manual-page)_

18.3 password vergessen: was tun?

- o root kann das password eines Benutzers mit dem Programm »passwd« ändern.
- o Hat root ihr password vergessen, so muß man rebooten und entweder
 - am LILO-Prompt mit "linux single" in den single-user Modus booten (nur bei veralteten Systemen möglich)
 - mit den Installationsdisketten oder Rettungsdisketten booten (ohne tatsächlich zu installieren) und sich so Zugang zur Datei `/etc/passwd' zu verschaffen, um das password von root zu löschen (-> kein password).

18.4 Wahl eines "guten" passwords

- o Das Programm »npasswd« läßt leicht zu erratende passwords erst gar nicht zu (Alternative zu dem herkömmlichen »passwd«-Programm).
- o Ein password muß 2 Bedingungen erfüllen:
 - es muß einfach zu merken sein, damit man es nicht notieren braucht
 - es darf nicht mit Standard-Software wie »crack« zu erraten sein
- o Daher scheiden als password aus:
 - alle Dinge, die in Verbindung mit eigenen Person stehen (Adresse, Telefonnummer, Autokennzeichen, Lieblingsgericht, Name des Haustieres, usw.)
 - alle bekannten Wörter und Abkürzungen, die nur leicht oder in gängiger Form abgewandelt werden
- o Übrig bleiben Akronyme: bilde einen Satz und nimm als password jeweils das erste Zeichen eines jeden Wortes (oder ähnlich)

Bsp.: "Ich esse gerne frisch geröstete Pfirsichkerne"
 --> "IchegfgP" ("Ich", um 8 Buchstaben zu erhalten)
- o Allzu offensichtliche Akronyme sollte man aber ebenfalls nicht wählen (Sprichworte, Titel von Musik-Stücken oder Filmen, usw)

18.5 "Knacken" von passwords

- o Die gängigsten 1.000.000 Wörter, Romanfiguren, Städtenamen, usw. lassen sich auf einem P90 in weniger als 60 Minuten durchprobieren; deren beliebtesten Abwandlungen innerhalb von 24h.
- o »crack« ist ein Programm zum Erraten von passwords, es bedient sich eines (möglichst umfangreichen) Wörterbuchs und einer Reihe von Regeln zur Abwandlung von Wörtern.
- o Der Einsatz von »crack« kann strafbar sein und ist daher mit äußerster Vorsicht zu geniessen._

Die »bash«

19.1 Allgemeines zu shells

- o Es gibt unterschiedliche shells (z.B. »tcsh«, »pdksh«, »zsh«), unter Linux ist die »bash« ("bourne again shell") weit verbreitet.
- o Die folgende Beschreibung bezieht sich nur auf die »bash«.

19.2 Konfiguration

- o Konfigurationsdateien für die interaktive Benutzung werden wie

shell-Skripte geschrieben:

/etc/profile	(globale Konfiguration)
~/.bash_login, ~/.profile	(pers. Konfiguration)
~/.bash_logout	(vor dem Verlassen)

- o Konfigurationsdatei, die nur shell-Skripte beeinflusst

~/.bashrc	(pers. Konfiguration)
-----------	-----------------------
- o Tastenbelegung und Anzeige von Umlauten:

~/.inputrc	(keine Kommentare!)
------------	---------------------

19.3 Eingabehilfen

- o Kommandozeileneditor: die eingegebenen Kommandos lassen sich ähnlich wie in einem Editor bearbeiten; siehe Anleitung zum Editor emacs (siehe 9.5).
- o Vergangenheitsbewältigung: die shell merkt sich alle ausgeführten Befehle ("command line history"):
 - per Cursortasten kann man herumstöbern
 - gezieltes Suchen mit <Strg-R>
- o Tippfaulheit auch ohne Maus möglich:
 - einmal <Tab> vervollständigt Programm- und Dateinamen (engl: "completion"), falls eindeutig möglich
 - zweimal <Tab> zeigt alle Möglichkeiten an

19.4 Eingabe- und Ausgabeumleitung

- o Jedes Programm öffnet 3 "Kanäle" (Dateien, siehe 25.13) für die folgenden Zwecke

Eingabe (0), Ausgabe (1), Fehlermeldungen (2)

- o Ohne weitere Vorkehrungen sind diese automatisch mit dem gerade verwendeten Terminal verbunden.
- o Eingabeumleitung (aus Datei in Programm) mit '<'

Bsp.: »mail truemper < viele_Gruesse«

- o Ausgabeumleitung (von Programm in Datei) mit '>'

Bsp.: »cat /proc/cpuinfo > meinSystem«
»cat /proc/meminfo >> meinSystem« (Anhängen)
- o Fehlerumleitung mit '>'

Bsp.: »make zImage 2> Fehler«
»make zImage &> Fehler« (inkl. Ausgabeumleitung)_

19.5 Röhren zur Datenweitergabe

- o Bei einer Röhre (engl. "pipe") erhält das nachfolgende Programm die Ausgabe des vorhergehenden Programms als Eingabe.

Bsp.: »cat viele_Gruesse mail truemper«

- o Röhren können aus mehreren Stücken bestehen.

Bsp.: »cat /etc/passwd sort more«

- o Rückgabewert: der des letzten Befehls in der Röhre (siehe 19.7)

19.6 Umgebungsvariablen

- o Umgebungsvariablen beeinflussen sowohl die Kommandozeile selbst, als auch die in ihr gestarteten Programme:
 - setzen von Variablen innerhalb der shell durch Zuweisung

Bsp.: »PS1='\h \u \w Kommandozeile:' «
--

- setzen mit Vererbung an Kindprogramme durch »export«
Bsp.: »export TERM=vt100«

- o Anzeige aller gesetzten Variablen mit »set« oder »env«
- o Löschen mit »unset«
Bsp.: »unset PATH«
- o wichtigste Variable: »PATH«, der Suchpfad für ausführbare Programme
Bsp.: »export PATH="/bin:/usr/bin:~/bin"«
»which ls« (bezieht sich auf \$PATH, vgl. 20.1)

19.7 Rückgabewerte

- o Jedes Programm gibt nach Beendigung eine Zahl an die bash zurück, die Auskunft über aufgetretene Fehler gibt.
- o Die Umgebungsvariable '\$?' enthält diesen Wert - bis zum Ausführen des nächsten Befehls (!)
- o Der Rückgabewert '0' bedeutet "kein Fehler".

19.8 Kommandolisten

- o Man kann mehrere Kommandos hintereinander angeben (verketteten), nur muß man der shell mit speziellen Zeichen anzeigen, wo ein Kommando endet und wo das nächste beginnt.
Bsp.'e: »make dep && make clean && make zImage« (siehe 52.X)
- o Braucht ein Programm eine Möglichkeit zur Eingabe, so muß bei Verwendung von '&' diese notfalls per Umleitung in eine Datei gegeben werden (auch für die Ausgabe empfohlen).
Bsp.: »make zImage &> MakeMeldungen &<«

Zeichen	Bedingung	Fehler-Rückgabewert
&	keine	stets 0
;	vorheriger Befehl beendet	des zuletzt ausgef. Befehls
&&	vorheriger Befehl erfolgreich	----- " -----
	-- " -- Befehl mißlungen	----- " -----

19.9 Ausführen von Kommandos und Ersetzungen

- o Vor dem Ausführen einer Zeile mit Kommandos führt die shell Erweiterungen und Ersetzungen im eingegebenen Text durch.

	erweitert/ersetzt durch	Beispiel
a{d,c,b}e	Kombinationen	»echo a{d,c,b}e«
~	home-Verzeichnis	»echo ~« »cd ~root«
\$var	Wert der Variablen "var"	»echo \$PATH«
\$(bfhl)	Ausgabe des Befehls "bfhl"	»rm \$(find /tmp -atime +3)«
`bfhl`	wie oben	»rm `find /tmp -atime +3`«
\$((x+y))	Wert der Operation x+y	»echo \$((1+1))«
*	mögliche Dateinamen	»ls -l /dev/lp*«
?	einzelne Zeichen von Dateinamen	
[abc]	Zeichen von Dateinamen	»ls -l /dev/lp[12]«

- o Nach dem Anwenden obiger Regeln werden alle selbst nicht gequoteten Vorkommen der Zeichen '\', '`', and '`' entfernt.
- o Die sicherste Methode zur Weiterverwendung der Ausgabe eines

Befehls als Argumente für einen weiteren Befehl ist das Programm
»xargs«; Nachteil: langsamer

```
Bsp.: »find /tmp -atime +3 -print0 | xargs -r0 rm --«
```

[Letzteres ist NICHT möglich mit dem Kommando

```
»find /tmp -atime +3 | rm -- «,
```

da die Ausgabe von »find« (eine Liste mit Dateien) nicht als
Eingabe an »rm« übergeben werden soll, sondern als Argument]

19.10 Aliase

- o Umbenennung von Kommandos (z.B. bei DOS-Entzugerscheinungen):

```
Bsp.: »alias del='rm -i'«
```

- o Abkürzung von Kommandos:

```
Bsp.: »alias wech='clear; logout'«
```

- o Erzeugung komplexer Kommandos:

```
Bsp.: »alias play='cat $@ > /dev/audio'«
```

- o Aliase sind nur im interaktiven Modus wirksam (d.h. nicht in
shell- Skripten).

19.11 Kontrollstrukturen (nur Schemata)

- o Abgezählte Wiederholungen:

```
for name [ in word; ] do list ; done
```

- o Bedingte Ausführung:

```
if list then list [ elif list then list ]\  
... [ else list ] fi
```

- o Fallunterscheidung:

```
case word in [ pattern [ | pattern ] ... ) list ;; ]
```

- o Schleife:

```
while list do list done_
```

20 Shell-Skripte

- o Alle Kommandos können auch in eine Datei geschrieben werden,
die dann Zeile für Zeile von der shell abgearbeitet wird.
- o Eine solche Datei sollte mit "#!/bin/sh" beginnen.
- o Anwendungsbereich: kleinere Aufgaben der Systemverwaltung.
- o Alternativen: andere Skriptsprachen (»python«, »perl«)
(Skripte zum Booten müssen mithilfe der bash realisiert werden!).
Bsp.:

```
-----8<----- hier schneiden -----8<-----  
#!/bin/sh  
# Dies ist ein Kommentar  
echo "Sie riefen das Programm mit folgenden Parametern auf: $@"  
  
# Eingabe eines Satzes  
echo -n "$0: Bitte geben Sie etwas ein: "; read $Eingabe  
  
# Auswertung der Eingabe  
if [ "$Eingabe" = "" ]; then  
    echo "Sie haben nichts eingegeben, Sie Faulpelz!"  
else  
    echo "Sie haben eingegeben: $Eingabe"  
fi  
  
# Welche Gerätedatei gehört zum Druckeranschluß?  
for i in /dev/lp*; do tunelp $i; done  
  
# Auswahl des Standard-Druckers anhand der
```

```

        # Gruppenzugehörigkeit
GROUPID=`id -g`
case $GROUPID in
"20900")
    PRINTER=declaser;;
"21000")
    PRINTER=klaser;;
esac
export PRINTER

        # Einmal <Strg-D> drücken zum Ausloggen
ignoreeof=1; export ignoreeof

        # Zugriffsrechte für neu erzeugte Dateien (invertiert,
        # "022" bedeutet "755"); dauerhafte Einstellung wird nur
        # über `/etc/profile' oder `~/.bash_profile' erreicht!
umask 022

        # Einige tcsh-Funktionen für die bash
which () { type -path "$@"; }; export -f which
setenv () { export $1="$2"; }; export -f setenv
unsetenv () { unset "$@"; }; export -f unsetenv
-----8<----- hier schneiden -----8<-----

```

- o "debug"-Modus wird durch "set -x" (Parameter '-x') eingeschaltet.
- o Werden Skripte oder Funktionen mit Optionen aufgerufen, so bezeichnen
 - '\$@' alle Optionen
 - '\$0' - '\$9' die ersten zehn Optionen (0=Programmname)_

Prozesse

21.1 Programme

- o Um den Inhalt einer Datei (Skript-Text oder Maschinencode) ausführen zu können, muß das Zugriffsrecht 'x' ("executable", siehe 7.5) gesetzt sein.
 - Bsp.: »chmod a+x Mein_Programm« (z.B. Skript aus 20)
- o Mithilfe der Zugriffsrechte kann die Ausführung eines Programms bestimmten Benutzern oder Benutzergruppen vorbehalten werden.
- o Ein Programm wird nach dem Start als "Prozeß" bezeichnet (gebräuchlich ist auch engl. "task").

21.2 Prozesse

- o Unter Linux können mehrere (bis zu 512) Prozesse quasi gleichzeitig ausgeführt werden [tatsächlich kann eine CPU natürlich nur ein einziges Programm abarbeiten; durch ein schnelles Abwechseln bei der Programmausführung entsteht die Illusion der Gleichzeitigkeit (Vergleich: Fernsehen)].
- o Jedes Programm erhält beim Start eine eindeutige Nummer im Bereich von 4-32767, die sogenannte "process id", kurz "PID".
- o Mögliche Zustände für Prozesse sind z.B. R ("running"), S ("sleeping"), Z ("zombie"), etc.
- o "zombies" sind bereits beendete Prozesse, deren Daten aber noch für andere Prozesse zur Verfügung stehen müssen.

21.3 Hintergrundprozesse und Daemonen

- o Ein "Hintergrundprozeß" ist mit keinem Terminal direkt verbunden.

- o Der Befehl »nohup« leitet die Ausgabe eines Programms in eine Datei ('nohup.out') um und läßt es mit Priorität 5 starten.
Bsp.: »nohup make zImage &« (siehe 52.X)
- o Ständig laufende Programme werden als "Daemonen" bezeichnet; diese schalten sich von selbst in den Hintergrund und benötigen keine Eingaben vom Benutzer (und verursachen auch keine Ausgabe).
- o Fehlermeldungen der Daemonen werden im Verzeichnis '/var/log/' protokolliert (abhängig von '/etc/syslog.conf').
- o Die Programmendung 'd' weist auf "daemon" hin.

21.4 Verwandtschaftsverhältnisse

- o »init« ist der Vorfahre aller Prozesse.
- o Startet ein Prozeß einen weiteren Prozeß, so bezeichnet man den ersten als Eltern- und den zweiten als Kindprozeß, den Vorgang des selbst als "forken".
- o "fork-bomben" sind Prozesse, die in einer Endlosschleife sich selbst neu starten, sie legen das System u.U. sofort still; dagegen werden unter '/usr/src/linux/include/linux/tasks.h' Vorkehrungen getroffen.

Prozeßbaum (gekürzte Ausgabe von »pstree«)

init--lpd	Druckmanager ("line printer daemon")
-gpm	Maustreiber (Textkonsolen)
-3*[getty]	login-Aufforderungen (Textkonsolen)
-cron	startet Programme zu einstellbaren Zeiten
-xinetd	internet-super-daemon (erweiterte Version)
-kflushd	regelmäßiges Leeren von Datenpuffern
-klogd	protokolliert Meldungen des kernels
-kswapd	Steuerung des Auslagerns von RAM
-rpc.portmap	ermöglicht netzwerkweite Prozeduren
-syslogd	protokolliert Meldungen versch. Programme
-update	
-bash---pine	e-mail-Programm auf der Kommandozeile
-bash---emacs	Texteditor »emacs« auf Kommandozeile

21.5 Binärformate

- o Ein "Binärformat" ist die Art und Weise der Organisation des Maschinencodes in einer Programmdatei.

Format Beschreibung

a.out	altes Binärformat von Linux
ELF	Das "Executable and Linking Format" bietet Programmierern wenige, aber enorme Vorteile; die Benutzer profitieren nur indirekt davon (anders ausgedrückt: bemerken es nicht); siehe 'ELF-HOWTO'.
iBCS2	"Intel Binary Compatibility Standard" ist zur Ausführung von Programmen einiger kommerzieller Un*xen notwendig.

21.6 Speicherverwaltung

- o Linux lädt in 4kb-Blöcken nur die Programmteile in das RAM, die tatsächlich benutzt werden (ganz schön geschickt, was?).
- o Reicht das RAM nicht zur Ausführung eines oder mehrerer Prozesse aus, so lagert Linux längere Zeit nicht benutzte Programmteile oder Daten auf speziell dafür vorgesehenen Bereichen ("swap-devices") auf der Festplatte aus (Fachausdruck: "paging", auch: "swapping").

- o Prozesse, die unerlaubt auf Speicherbereiche (anderer Prozesse) zugreifen wollen, werden SOFORT mit der Fehlermeldung "SEGMENTATION FAULT" ("SIGSEGV", "segmentation violation") abgebrochen.

21.7 Nach einem Programmabsturz ...

- o Programmabstürze verursachen i.d.R. einen letzten Schnappschuß des Programms, der in Form der Datei `core' in das aktuelle Verzeichnis geschrieben wird.
- o Mit dem Kommando »ulimit« kann man die Erzeugung von core-Dateien unterbinden (bzw. begrenzen).
- o core-Dateien können u.U. wiederbelebt werden (als Programme weiterlaufen) oder zur Fehleranalyse mit einem debugger untersucht werden._

Prozeßkontrolle

22.1 Informationen über Prozesse

- o Im Verzeichnis `/proc' werden Informationen über die Prozesse zur Verfügung gestellt; die Programme »procinfo«, »ps«/»tkps«, »pstree«, »top« und »fuser« werten diese Informationen aus.
- o Übergibt man einem Programm ein Argument, so ist mit »ps« und Verwandten sichtbar; deshalb verwendet man passwords nicht als Argument für Programme.
- o Der Daemon »rpc.rstatd« stellt Statistiken über die Auslastung der Maschine im Netz zur Verfügung.
- o In der Datei `/etc/inetd.conf' sind evtl. weitere, unerwünschte Informationsdienste konfiguriert (»netstat«, »sysstat«, »finger«).
- o »time« kann zum Starten von Programmen verwendet werden, deren Ressourcenverbrauch pro Programmablauf man ermitteln will.
Bsp.: »time -v wc /proc/ksyms«

22.2 Detaillierte Ablaufverfolgung

- o Höchste Detailstufe erreicht man nur mit einem debugger; dessen Bedienung geht über den Anspruch dieses Manuskripts hinaus.
- o »strace« kann zum Starten von Programmen verwendet werden, deren Systemaufrufe protokolliert werden sollen.
Bsp.: »strace -f -eopen,stat cat /dev/null«

22.3 Signale

- o Das Programm »kill« beeinflusst laufende Programme durch Signale.
- o Liste aller Signale mit dem Befehl »kill -l«; Auswahl:

Nr.	Name	Taste	Funktion
1	HUP		hangup (Modem hat aufgelegt)
2	INT	<Strg-c>	interrupt (unterbrechen)
3	QUIT	<Strg-\>	quit (verlassen)
9	KILL	<Strg-u>	kill (gewaltsam beenden)
15	TERM		terminate (beenden)
18	CONT		continue (Programm fortsetzen)
19	STOP	<Strg-z>	stop (Programm anhalten)

- o Zuordnung von Tasten zu Signalen mit dem Befehl »stty«
- o »kill -KILL« kann weder abgefangen, noch ignoriert werden.

- o »kill -TERM« ist der Standard und läßt die Chance zum Beenden wichtiger Vorgänge (Speichern, usw.).
Bsp.: »kill \$PPID«
- o Die Bedeutung der Signale kann von Programm zu Programm etwas unterschiedlich sein.
- o Signale senden unter Angabe des Programmnamens mit »killall«
Bsp.: »killall -STOP emacs«
»kill -STOP %emacs"«
- o Beenden durch Anklicken: »xkill«_

22.4 Seid nett zueinander

- o Prozesse können unterschiedlich nett zu anderen Prozessen sein, wenn es um die Zuteilung von Prozessor-Leistung (Rechenzeit) geht.
- o Statt Nettigkeit wird auch die Bezeichnung "Priorität" verwendet (umgekehrte Bedeutung!).
- o Die Nettigkeit wird durch Zahlen von -20 (keine Rücksicht auf andere) bis 20 (vollkommene Bescheidenheit) angegeben, dazwischen lassen sich alle (ganzzahligen) Abstufungen wählen.
- o Die gewöhnliche Nettigkeit liegt bei 0.
- o Nettigkeiten <0 sind wegen ihrer Gefährlichkeit nur von root einstellbar; Vorsicht!
- o Starten eines Programms mit Nettigkeit ungleich '0':
Bsp.: »nice -n 10 xdos«
- o Nachträgliches Verändern der Nettigkeit: »renice«
- o Das nachträgliche Herabsetzen der Nettigkeit ist nur der Superuserin root erlaubt.

22.5 Beschränkung der Ressourcen

- o RAM, Festplattenplatz und Prozessorleistung können durch Fehlfunktion von Prozessen in unbegrenztem Umfang angefordert werden und dadurch kann ein Linux-PC unbenutzbar werden.
- o Eine extrem großzügige Beschränkung der Ressourcen ist angebracht.
- o "disk quotas": Beschränkung des Festplattenplatzes pro Benutzer

22.6 Programme mit speziellen Rechten

- o Um Benutzern einen kontrollierten Zugriff auf Daten zu ermöglichen, kann man unter Linux Programme mit speziellen Ausführungsrechten versehen.
- o "suid": set-user-id-on-execution, statt 'x' (siehe 7.5 und 21.1) wird ein 's' für den Eigentümer des Programms gesetzt; daraufhin wird das Programm stets im Namen seines Eigentümers und mit seinen Rechten ausgeführt, EGAL welcher Benutzer es startet (vorausgesetzt, er darf es überhaupt starten)

Bsp.: »ls -l /usr/bin/passwd /etc/passwd«

obwohl '/etc/passwd' für die Benutzer nur lesbar ist, kann jeder Benutzer seine Daten in dieser Datei mit dem Kommando »passwd« ("suid-root") ändern

- o "sgid": set-group-id-on-execution, statt 'x'
[entsprechende Funktion wie bei »suid«-Programmen]

VORSICHT: diese Arten von Programmen sind potentielle Sicherheitslücken !

Da sich Skripte (z.B. shell-Skripte) unter Linux austricksen

lassen (der Inhalt ist kurz nach dem Start vom Benutzer austauschbar), ignoriert der Betriebssystemkern das s-bit für Skripte wohlweislich. (Ausnahme: »perl«-Skripte)._

"swap"

Da das physikalisch vorhandene RAM ("Arbeitsspeicher", "memory") zum gleichzeitigen Betrieb vieler Programme selten ausreicht, bietet Linux die Möglichkeit, spezielle Bereiche der Festplatte wie zusätzliches RAM zu behandeln, diese nennt man "swap-devices".

23.1 Bezeichnungen

"virtual memory" das durch swap-devices bereitgestellte "RAM"

"swapping": vollständiges Auslagern eines Programme in das "virtual memory" (diese Technik findet heutzutage fast keine Verwendung mehr, der Begriff wird aber trotzdem benutzt)

"paging": seitenweises Auslagern von unbenutzten Programmteilen (in Blöcken zu je 4kb)

23.2 Philosophie der Speicherverwaltung

- o "virtual memory" ist aufgrund der Mechanik der Festplatten sehr viel langsamer als RAM (> Faktor 1000), es ist also immer nur eine Notlösung.
- o Das RAM soll immer für irgendeinen Zweck verwendet werden, ungenutztes ("freies") RAM ist Geldverschwendung.
- o Ein kleiner Teil des RAM ("idle memory") muß allerdings für Situation freigehalten werden, in denen kein paging möglich ist (mind. 190 kbyte).
- o Ungenutztes RAM wird automatisch zur vorübergehenden Beschleunigung von Festplattenoperationen (Speicher, Lesen) eingesetzt, d.h. wiederholt angeforderte Dateien werden im RAM bereitgehalten.
- o Diese Funktion ("buffering") wird sofort stark eingeschränkt, sobald RAM von Prozessen benötigt wird.
- o Wird ein Programm zwei- oder mehrmals gestartet, so wird die bereits im Speicher befindliche Kopie (oder die vorhandenen Teile) zum Starten des Programms benutzt ("cache").
- o "shared libraries" ermöglichen den gemeinsamen ("shared") Zugriff verschiedener Programme auf die gleichen Funktions-Bibliotheken ("libraries").

23.3 Statistiken zur Speicherbenutzung

- o »free« und »vmstat« zeigen den "freien" und benutzten Speicher an.
- o Genauere Statistiken erhält man auf der Textkonsole durch Drücken von <Strg-Rollen> oder <Shift-Rollen>.

	total	used	free	shared	buffers	cached
Mem:	15104	14876	228	10508	628	6148
Swap:	20124	3848	16276_			

23.4 Erzeugung von "virtual memory"

- o Da die Daten von den äußerten Bereichen (=erste Partionen) einer Festplatte am schnellsten gelesen werden können, sollte man dort

auch die "swap-devices" plazieren.

- o Als "swap-devices" eignen sich leere Dateien oder leere Festplattenpartitionen mit einer Größe von 40kb - 128 MB (PC)

Erzeugung von 8 MB "virtual memory":

```
-----8<----- hier schneiden -----8<-----
# Erzeugen einer 8MB großen, leeren Datei
dd if=/dev/zero of=/var/swapfile bs=1k count=8192
mkswap /var/swapfile # "Formatierung"
swapon /var/swapfile # Einschalten
-----8<----- hier schneiden -----8<-----
```

- o "swap-devices" können in gewissen Grenzen im laufenden Betrieb ein- und ausgeschaltet werden.
Bsp.: »swapoff /var/swapfile«
- o Für alle "swap-devices", die in der Datei `/etc/fstab` aufgeführt sind, erfolgt beim booten eine automatische Aktivierung.
- o Als Größe sollte man ca. die doppelte Größe des RAM wählen.

23.5 swap-Verhalten

- o Für spezielle Einsatzzwecke kann es sinnvoll sein, die Voreinstellungen bzgl. des swap-Verhaltens des kernels zu ändern.
- o Die gewünschten Einstellungen können dem Boot-Manager LILO übergeben werden; siehe `/usr/src/linux/mm/swap.c`. (Wer diese Datei nicht versteht, sollte auch nicht an den Werten herumfummeln.)
- o [? Darf die Priorität des kswapd herabgesetzt werden ?]
- o Per "RAID" lassen sich "swap-devices" auf verschiedenen Festplatten zu schnelleren Einheiten zusammenfassen._

Die Datei als Prinzip

24.1 Gerätedateien

- o Der Zugriff auf Hardware erfolgt (theoretisch) nur über Dateien.

```
Bsp.: »cat /dev/mouse«
```

```
(nur auf Textkonsole und ohne laufenden »gpm«
zeigt Mausbewegungen an; Abbruch mit <Strg-c>)
```

- o Diese speziellen Dateien heissen "Gerätedateien" und liegen im Verzeichnis `/dev`; die engl. Bezeichnung ist "devices".
- o Vorteile:
 - Zugriffsrechte für Hardwarekomponenten
 - einheitliche Form von Zugriffsrechten
- o Es wird unterschieden zwischen
 - "character devices": zeichenorientierte Geräte
 - "block devices": blockweise arbeitende Geräte
- o Ausnahmen bilden die Netzwerkkarten (z.B. Ethernet, isdn), die nicht über Dateien angesprochen werden
- o Von DOS bekannt: LPT1:, COM1:, NUL:
- o Erzeugung fehlender Gerätedateien mithilfe des shell-Skripts

```
»/dev/MAKEDEV«  
Bsp.: »cd /dev; ./MAKEDEV tty9«
```

24.2 Sockets

- o "sockets" sind Übergabepunkte für Informationen (sie stellen eine der drei Möglichkeiten dar, wie Prozesse miteinander kommunizieren können).
Bsp.: `/dev/printer' (siehe 48.X)
- o Die Verbindung zwischen zwei Übergabepunkten kann innerhalb der eigenen Maschine ("Unix domain sockets") oder über das Internet geschehen ("Internet domain sockets").
- o Herumspielen mit sockets: Paket »socket-1.1«

24.3 Named Pipes

- o "named pipes" sind die dritte Möglichkeit, mit der Prozesse untereinander kommunizieren können (die "erste" Möglichkeit nennt sich übrigens "shared memory" [gemeinsam genutzter Speicher]).
- o Eine "named pipe" nennt man auch "fifo" (first-in-first-out).
Bsp.: »mkfifo rohr; echo "hallo_welt" > rohr« (wartet)
»cat rohr«
- o Bildlich gesprochen sind "named pipes" fest eingegrabene Röhren (siehe 19.5).

Die Dateien im Verzeichnis `/dev'

Nur nicht von den mehr als 700 Dateien in diesem Verzeichnis abschrecken lassen! Immerhin kann man sie mit einfachen Schemata der folgenden drei Seiten leicht überblicken.

Das Format der Übersicht ist:

Datei	ggf. Nummerierung	Bemerkungen, Pr*d*ktnamen
25.1	Parallele Schnittstellen (siehe `Printing-HOWTO')	
lp \times	$\times = \{ 0 \dots 2 \}$	1. - 3. "Druckeranschluß"
"lp" steht für "line printer" ("Zeilendrucker")		
25.2	Serielle Schnittstellen (siehe `Serial-HOWTO')	
ttyS \times	$\times = \{ 0 \dots 3 \}$	serielle Schnittstellen Nr. 1-4 (unter DOS: COM1 - COM4)
ttyS \times	$\times = \{ 4 \dots 31 \}$	nur bei zusätzlicher Hardware, den sogenannten "multiport boards"
ttyI \times	$\times = \{ 0 \dots 16 \}$	Modem-Emulation des ISDN-Treibers
cuax	$\times = \{ 0 \dots 31 \}$	wie "ttyS"
cui \times	$\times = \{ 4 \dots 31 \}$	wie "ttyI"
25.3	Busmäuse (siehe `Busmouse-HOWTO')	
atibm, atimouse		ATI-XL (Grafikkarte)
inportbm, msmouse		Inport (ATI, M*crosoft)

jmouse	????
logibm, logimouse	Logitech
psaux, psmouse	PS2

25.4 Sound (siehe `Sound-HOWTO')

audio	SUN-ähnliche audio-Schnittstelle
pcaudio	dito f. eingeb. Lautsprecher
dsp	"digital sampling device" ("rohe Audiodaten")
pcsp	dito f. eingeb. Lautsprecher
midi0x x = { 0 .. 3 }	MIDI-Schnittstelle
mixer	Mixer
pcmixer	dito f. eingeb. Lautsprecher
patmgrx x = { 0,1 }	Patch-Manager
sequencer	direkter Zugriff auf MIDI, FM, und GUS
sndstat	Daten zur vorhandenen (bzw. erkannten) Hardware_

25.5 Terminals (siehe 16.3.1)

tty	Kontrollterminal eines Prozesses
ttyx x = { p0..pf, ..., z0..zf A0..Af, ..., E0..Ef }	Terminal-Anschlüsse aus der Sicht des Benutzers
ptyx x = { p0 .. Ef }	Terminals aus Sicht der Prozesse

25.6 Konsolen (siehe 16.2.1)

ttyx x = { 0 .. 63 }	virtuelle Konsolen
vcsx x = { 0 .. 63 }	Inhalt von "ttyx"
vcsax x = { 0 .. 63 }	dito inkl. Bildschirmattribute

25.7 Diskettenlaufwerke (engl. "floppy disk", kurz "fd")

fdx x = { 0 .. 8 }	generisches Gerät, automatische Erkennung der Diskettenkapazität
fd0CompaQ	für CompaQ-Laufwerke
fd0cbm1581	für Commodore 1581-Disketten
fd0Dx x = 360 .. 1040	DD-Disketten
fd0Ex x = 2880 .. 3840	ED-Disketten
fd0Hx x = 1440 .. 1920	HD-Disketten

| \ |
 | \ |
 | \| - Kapazität der Diskette
 | \| --- Diskettentyp
 | \| ----- Nr. des Laufwerks (0=a: 1=b: ...)

25.8 Bandlaufwerke mit Diskettenanschluß (siehe `Ftape-HOWTO')

rftx x = { 0 .. 3 }	Bandlaufwerk mit Rückspulen
nrftx x = { 0 .. 3 }	dito ohne automatisches Rückspulen

"rft" steht für "rewinding floppy tape"
"nrft" steht für "not rewinding floppy tape"

25.9 (E)IDE Festplatten (engl. "hard disks", kurz "hd")

hda	gesamte erste IDE Festplatte
hdax x = { 1 .. 16 }	Partitionen der ersten IDE

```

| \
|  \- Nr. der Partition
|---- Nr. der Festplatte (a = erste, b = zweite, ...)

```

25.10 Sonderfunktionen

```

full          ?????
kmem          Abbild des
log           Socket f. »syslogd«
mem           ?????
null          Schwarzes Loch, schluckt alles
ram           ?????
xconsole      ?????
zero          Liefert unbegrenzt viele Nullen_

```

25.11 CD-ROMs mit herstellerspezifischen Schnittstellen (siehe `CDROM-HOWTO')

```

aztcd0        Aztech CD268A, Orchid CD-3110,
               Okano/Wearnes CDD110, Conrad TXC
cdu31a        Sony CDU31A/CDU33A
cdu535        SONY CDU-535/531
cm206cd       CM206 Philips/LMS
gscd0         Goldstar R420
isp16         ISP16, MAD16, Mozart
lmscd         LMS/Philips CD205/225/202
mcd           Mitsumi
mcdx×        × = { 0,1 }
               Mitsumi, multisession-fähig
optcd0        Optics Storage 8000 AT
sbpcd×        × = { 0 .. 3 }
               Sound-Blaster OHNE IDE/ATAPI, also
               die meisten Matsushita, Panasonic,
               Creative, Longshine, Kotobuki und
               TEAC Laufwerke
sjcd          SANYO CDR-H94A
sonycd        ?????

```

25.12 SCSI Geräte

```

st×          × = { 0,1 }
nst×         × = { 0,1 }
               SCSI-Bandlaufwerk/Streamer
               dito, ohne autom. Zurückspulen

sra          SCSI-CDROM mit ID 0
sga          generisches SCSI-Gerät mit ID 0
sda          gesamte SCSI Festplatte mit ID 0
sdax         × = { 1 .. 15 }
               Partitionen der SCSI Festplatte
               mit ID 0
| \
|  \- Nr. der Partition
|---- SCSI-ID (a = 0, b = 1, ..., h = 7)

```

```

"st" steht für "scsi tape"
"sr" steht für "scsi rom"
"sg" steht für "scsi generic"
"sd" steht für "scsi disk"

```

25.13 Verweise im Verzeichnis `/dev'

```

:0           -> console           siehe 16.2
cdrom        -> mcdx0             (Beispiel)
console      -> tty0             siehe 16.3.1
core         -> /proc/kcore       Abbild des kernels
fd           -> /proc/self/fd     Verz. mit "file descriptor"

```

ftape	-> rft0	Floppy tape
kcore	-> /proc/kcore	Abbild des kernels
modem	-> cua2	(Beispiel)
mouse	-> ttyS0	(Beispiel)
sbpcd	-> sbpcd0	SoundBlaster-CDROM
stdin	-> fd/0	Standard-Eingabe
stdout	-> fd/1	Standard-Ausgabe
stderr	-> fd/2	Kanal für Fehlermeldungen
nftape	-> nrft0	Floppy tape_

Organisation des Dateibaums

Glossar:

"File System": Dateisystem, Dateibaum, Verzeichnisbaum

26.1 Grundlegendes

- o keine Laufwerksbuchstaben, stattdessen ein einziger, aus den verschiedenen Massenspeichern zusammengesetzter Verzeichnisbaum
- o Auch Daten anderer Betriebssysteme können in diesen Baum integriert werden (Prinzip des virtuellen Dateissystems).
- o drei große, ineinandergeschachtelte Hierarchien:

/usr/local	selbst installierte Software
/usr	Lieferumfang der Distribution
/	zum Booten unbedingt benötigte Programme

(jeder Baum jeweils ohne den vorherigen Ast)

- o drei weitere wichtige Äste:

/usr/X11R6	(Standardaufteilung)
/usr/lib	(kunterbuntes Durcheinander)
/var	(kein '/usr/var' oder '/usr/local/var' !)

26.2 Der "Linux File System Standard"

- o Der "Linux File System Standard" (FSSTND) enthält Richtlinien zum Aufbau des Dateibaums; erhältlich per von
<ftp://tsx-11.mit.edu/pub/Linux/docs/Linux-standards/fsstnd>
- o Der Standard richtet sich nur an Personen, die Software "für Linux" zusammenpacken und auf ftp-Servern hinterlegen.

26.3 "Disk space ... the final frontier"

Verzeichnis	Platzbedarf		Bemerkung
	min.	empf.	
/	15	20	abhängig von der Distribution
/usr	300	500	grundsätzlich zu klein
/tmp	10	20	unkritisch
/home	20	100	pro Benutzer mind. 10 MB
/var	20	60	TeX-fonts, e-mail, Druckaufträge
swap	10	20	siehe 23.X
=====			
gesamt:	375	720	

- o Die verschiedenen Äste lassen sich auch auf mehrere Festplatten

oder Partitionen verteilen._

Beispiel für einen Verzeichnisbaum (unvollständig):

```
-----  
/                               Wurzel des Verzeichnisbaums  
|                               (engl. "root-directory")  
bin/                             Programme, von Benutzern ausführbar  
boot/                            Kopien der Boot-Konfiguration, Betriebssystemkern  
dev/                             Gerätedateien ("devices")  
etc/                             Konfigurationsdateien ("et cetera")  
home/                            Privatverzeichnisse der Benutzer  
lib/                             Bibliotheken mit Programmroutinen  
|                               und sonstige Datensammlungen  
|   modules/                     ladbare Treiber des Betriebssystemkerns  
|  
lost+found/                       Reste verlorengegangener Dateien  
mnt/                              vorübergehend/testweise montierte Speichermedien  
proc/                             Informationen über den Zustand des Systems  
root/                             Privatverzeichnis der Systemverwalterin (root)  
sbin/                             Programme, nur von root ausführbar  
tmp/                              Verzeichnis für temporäre Dateien  
usr/  
|  
|   X11R6/                       X Window System, Version 11 Ausgabe 6  
|   bin/                         Programme  
|   doc/                         Dokumentation  
|   lib/                           
|   include/                     Variablendeklarationen f. C-Bibliotheken  
|   local/  
|   |  
|   |   bin/  
|   |   doc/  
|   |   lib/  
|   |   include/  
|   |   sbin/  
|   |   src/  
|  
|   man/                         elektronische Handbücher ("manual pages")  
|   src/                         Quelltexte ("sources")  
|   |  
|   |   Linux/                   Quelltexte zu Linux (Betriebssystemkern)  
|  
var/                              veränderliche Dateien ("variable")  
|  
|   local/  
|   lock/                        Reservierung/Blockierung von Gerätedateien  
|   log/                         Logbücher und administrative Informationen  
|   run/  
|   spool/                       Druckaufträge, e-mail, News, TeXfonts, ...  
|   |  
|   |   cron/                   Konfiguration für die Zeitschaltuhren  
|   |   lpd/                    Druckaufträge  
|   |   mail/                   Postfächer für e-mail  
|   |  
|   |   tmp/_  
Montage des Dateibaums  
-----
```

- o Der Befehl »mount« setzt den Verzeichnisbaum (siehe 26.1) aus den Datenträgern oder Teilen davon zusammen.
- o Das Ausführen des »mount«-Befehls wird kurz als "mounten" bezeichnet.

Bsp: »mount -t msdos /dev/fd0 /mnt«
 | | |
 Typ Gerät "wohin" (»mount point«,
 Einhängepunkt)

- o Konfiguration des »mount«-Befehls in »/etc/fstab« (siehe »man 5 fstab«); diese Datei wird während des Bootvorgangs automatisch "ausgeführt". Beispiel:

```

/dev/cdrom /cdrom          iso9660 defaults,user,ro 0 0
`-----' `-----'    `-----' `-----'
|          |          |          |          |
Gerät     Einhängpunkt Dateisystemtyp Optionen "dump" "passno"

```

- o Zur Bedeutung von "dump" und "passno" siehe 29.? und 54.?.
- o Die Reihenfolge in »/etc/fstab« ist entscheidend; der Ast »usr« muß gemountet sein, bevor »usr/local« gemountet werden kann.
- o Das "mounten" kann gewöhnlichen Benutzern mit der Option 'user' erlaubt werden (evtl. 'user,noauto').
- o Das Kommando »umount« ist die Säge für den Dateibaum; es arbeitet vor dem Abschalten des Computers den Inhalt der Datei »/etc/fstab« in umgekehrter Reihenfolge ab.

Bsp.: »umount /cdrom«

27.2 Das "virtual file system" (VFS)

- o Ein Dateisystem ist die Art und Weise, mit der man Daten auf Festplatten, CD-ROMs, usw. organisiert, bzw. speichert.
- o Das VFS ist das einzige Dateisystem für Linux - zumindest erscheint dies dem Benutzer so.
- o Der Zugriff auf tatsächlich existierende Dateisystemtypen (für Linux, DOS, OS/2) geschieht über Treiber (im Betriebssystemkern).
- o Nicht vorhandene Eigenschaften werden durch das VFS vorgegaukelt (z.B. Eigentümer und Zugriffsrechte für DOS-Dateien).
Bsp.: »ls -l /mnt /floppy /cdrom«
- o mac/amiga-Datenträger können von der PC-Hardware (angeblich?) nicht verarbeitet werden, da hilft auch kein VFS!_

27.3 Einige Dateisysteme, die Linux verarbeiten kann

Typ	für Zugriff auf	Typ	für Zugriff auf
ext2	Linux (standard)	hpfs	OS2 (readonly)
iso9660	CD-ROMs	msdos	DOS
minix	Minix	vfat	win95
nfs	Netzwerk	ncp	novell-Server
sysv	System-V	smb	WfW-shares
ufs	Sun/Nextstep/FreeBSD (readonly)		

- o Das "mounten" der Dateisystemtypen 'ncp' und 'smb' erfordert z.Zt. ein spezielles »mount«-Kommando.

Bsp.: »ncpmount -S NovServ /mnt«

27.4 Das "extended-2" Dateisystem

- o Ein Datenträger (oder Bereiche davon) erhält durch den Befehl »mke2fs« das Format des extended-2 Dateisystems und kann dann entsprechend "gemountet" werden. Der Vorgang wird als "Formatieren" bezeichnet.

Bsp.: »mke2fs -c /dev/fd0«

- o Die Betriebsparameter für das extended-2 Dateisystem können als Parameter bei der Formatierung angegeben werden oder nachträglich mit dem Befehl »tune2fs«.

Bsp.: »tune2fs -m 10 -gid li-adm /dev/fd0«

(In diesem Beispiel ist 10% des Platzes für die Gruppe "li-adm" reserviert.)

- o Fehler auf dem Dateisystem können mit dem Programm »fsck« (hoffentlich) behoben werden.

Bsp.: »fsck -M /dev/fd0«

27.5 "extended-2" Dateiattribute (Linux spezifisch)

- o »chattr« zum Setzen der erweiterten Attribute
- o »lsattr« zum Anzeigen derselbigen

Kürzel	Attribut	Bedeutung
-		Attribut nicht gesetzt
a	append	Daten werde an Datei wird nur angehängt
c	compress	Daten werden automatisch komprimiert
d	no-dump	Datei wird beim "dumpen" ignoriert
i	immutable	Datei unveränderbar (selbst für root)
s	secure-delete	Löschen überschreibt Datei mit Nullen
S	synchronous	wie Option 'sync' des »mount«-Befehls
u	undelete	Löschen kann rückgängig gemacht werden_

Speichermedien

Warnung:

auswechselbare Speichermedien vor dem Wechseln unbedingt mit dem Befehl »umount« demontieren, ansonsten droht Datenverlust!

28.1 Diskettenlaufwerke

- o Der Umgang mit DOS-Disketten ist unter 8.2 beschrieben.
- o Das Paket `fdutils-4.2' enthält Programme zum Konfigurieren der Diskettenlaufwerke, sowie zum Formatieren von Disketten. Nicht zuletzt die Dokumentation der exotischen Formate ist interessant.

- o Formatieren von DOS-Disketten mit den Befehlsfolgen

»fdformat /dev/fd0; mformat a:«

oder

»superformat /dev/fd0«

28.2 Festplatten ((E)IDE oder SCSI)

- o Linux kann Festplatten beliebiger Grösse ansprechen, ein Eintrag im BIOS-Setup ist NICHT notwendig.
- o Die Festplatte wird zur Aufnahme bestimmter Äste des

Verzeichnisbaums (oder gar zur Aufnahme anderer Betriebssysteme; *schauder*) in Bereiche unterteilt, diese nennt man Partitionen.

- o Die Partitionierung der Festplatte wird mit dem Programm »fdisk« (menügeführt: »cfdisk«) vorgenommen
Bsp.: »fdisk -l«
- o Eine Partition, die die Wurzel des Verzeichnisbaums '/' enthält, wird als "root-partition" bezeichnet (mehrere möglich).
- o Die "root-partition" und der "kernel" müssen innerhalb der ersten 1024 Cylinder auf der 1. oder 2. EIDE-Platte angelegt werden. (Bei SCSI-Festplatten ist dies egal.)

28.3 (E)IDE

- o "IDE" war jahrelang DIE Technik für Festplatten-Controller in PCs und wurde von "EIDE" ("enhanced IDE") abgelöst.
- o EIDE unterstützt bis zu 4 Geräte pro Controller und erlaubt höhere Datentransferraten als IDE.
- o Neben Festplatten kann man mittlerweile auch (Atapi-)CD-ROMs und (Atapi-)Bandlaufwerke an den EIDE-Controller anschliessen.

28.4 CD-ROMs

- o Zum Betrieb von CD-ROMs stehen folgende Techniken zur Verfügung:
 - ATAPI CD-ROMs oder SCSI CD-ROMs
 - herstellerabhängige Controller
- o Nähere Informationen zu den herstellerabhängigen Techniken findet man im Verzeichnis `/usr/src/linux/Documentation/cdrom'`.

28.5 SCSI

- o "SCSI" steht für "Small Computer Systems Interface" und ist eine Schnittstelle zum Anschluß von speziellen
 - Festplatten
 - CD-ROM Laufwerken
 - Bandlaufwerken
 - DATs und Scannern (beide werden über die "generic-devices" angesprochen)
- o Pro SCSI-Controller können bis zu 7 solcher Geräte angeschlossen werden; jedes erhält eine eindeutige Nummer; das 8. Gerät (und dessen Nummer) ist der SCSI-Controller.
- o Das SCSI-Verbindungskabel ist 50-polig, die maximale Länge des Kabels beträgt 3 m (gemessen vom ersten zum letzten Gerät).
- o An beiden Enden des Kabels müssen sogenannte Endwiderstände angebracht sein; dies kann mit externen Steckern geschehen oder per Aktivierung evtl. in die Geräte eingebauter Widerstände.
- o SCSI-Festplatten sind NICHT grundsätzlich schneller als IDE-Festplatten, sondern eher etwas teurerer.

28.6 Toaster

- o Die Herstellung eigener CD-ROMs erfordert das Paket »mkisofs-2.0« und eine Änderung des kernels (siehe CD-Write mini-HOWTO).
- o Das Verfahren besteht aus zwei Schritten, dem Erzeugen eines 1:1

Abbilds der CD und dem eigentlichen "Brennen".

```
Bsp.:  »mkisofs -r -o CD_Abbild /home/linux \  
      cdwrite /dev/sgf < CD_Abbild «
```

(Sicherstellen, daß '/dev/sgf' der CD-Schreiber ist! Ansonsten erfolgt "Formatierung" der Festplatte.)

- o Vor dem "Brennen" kann man die so erzeugte Datei per "loopback" als CD mounten und testen.
- o Der Schreibvorgang darf von keinem anderen Prozeß unterbrochen werden; ansonsten ist die CD unbrauchbar. Es sollten also nur die allernotwendigsten Programme gestartet werden und das Netz abgeschaltet sein (evtl. neuen "run-level" anlegen [siehe 55.X])_

Backup

[Habe kein DAT/Tape, daher bleibt diese Seite vorerst leer.]

—

—
Netzwerk und Internet

Warnung:

Mit dem Anschluß an das Internet oder allgemein an ein Netzwerk setzt man sich potentiellen Bösewichten aus.

Man muß vorher unbedingt alle accounts mit passwords schützen und einen Blick in die Dateien '/etc/inetd.conf' (siehe 42.1) und '/etc/X11/Xaccess' (siehe 45.X) werfen.

30.1 Linux und Internet

- o Das Internet ist ein weltweites Netzwerk von Computern.
- o Das Übertragungsprotokoll ist "transmission control protocol / internet protocol" (TCP/IP).
- o Das Netzwerk von Linux ist das Internet.

30.2 Grundzüge der Netzwerkkonfiguration

- o Unbedingt nur die vom Netzbetreiber zugewiesenen Werte verwenden.
- o Absolutes Muß für die Anbindung an ein Netz:

"ip-address":

Internet-Adresse der eigenen Maschine; vergleichbar mit einer Postleitzahl

- o Wenn man das gesamte Internet erreichen will braucht man ein(en)

"router" oder "gateway":

eine Maschine, die die Wege zu den anderen Maschinen kennt

- o Weil sich niemand die Nummern merken will:

"hostname":

der Name der eigenen Maschine (Kurzform)

"domainname":

Internet-Gebietsname (domain, Domäne)

"full qualified domain name" (FQDN):

hostname plus domainname ergeben den "vollen" oder

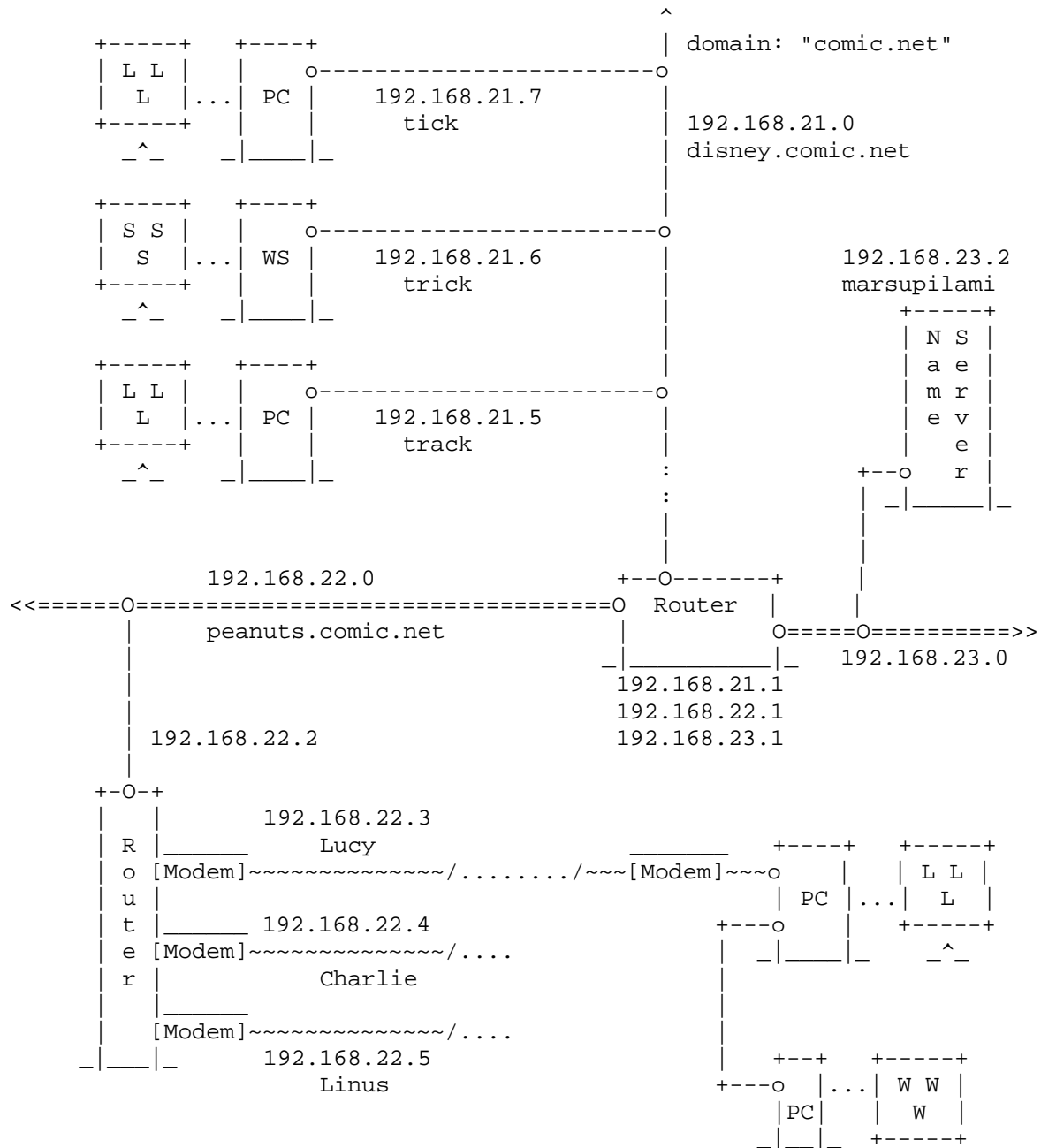
"langen" Maschinennamen.
 "nameserver":
 Internet-Adresse einer Maschine, die alle anderen
 Namen und Internet-Nummern kennt

30.3 Warum IP-Adressen UND hostnamen?

- o IP-Adressen sind zu kompliziert für den Benutzer (und auch schlecht zu merken).
- o "hostnamen" sind zu kompliziert für die Hardware der router.
- o In Kombination können auch mehrere, physikalisch getrennte Teilnetze zu einem logischen Verbund werden._

Beispiel eines Netzwerks

Bemerkung: die verwendeten IP-Adressen sind im Internet ungültig
 (siehe 31.1)



Hinweis zur Abbildung: man kann fast NICHTS daraus lernen, sie dient nur der Illustration.

Eine kleine Namenskunde befindet sich auf Seite 37._

Die Logistik des Internet

31.1 Adressen und Netze

- o Jede Maschine (Workstation, PC, Netzknoten) im Internet besitzt mind. eine numerische Adresse im Format

a.b.c.d (a, b, c und d im Bereich von 1 bis 254)

- o Die Adresse 127.0.0.1 bezieht sich stets auf die eigene Maschine (falls das "loopback-device" benutzt wird, siehe 33.1), sie kann zu keinen anderen Zwecken gebracht werden, als zur Definition des "das sind wir selbst".
- o Folgende Adreßbereiche sind zum Aufbau privater, nicht mit dem Internet verbundener Netzwerke freigegeben:

10.b.c.d 172.16.c.d 192.168.c.d

(diese Adressen werden vom Internet ignoriert!)

- o Man spricht von Class-A, -B, -C Netzwerken.
- o Auf Hardware-Adressen wird hier nicht eingegangen.

31.2 Namen und "Adressen-Server"

- o Fast jede Maschine im Internet besitzt neben der numerischen Adresse (Postleitzahl) mind. einen sybolischen Namen (Wohnort).
- o "nameserver" kennen die zu den symbolischen Namen gehörenden IP-Adressen und umgekehrt.
- o Abfrage der nameserver mit »nslookup« (keine man-page, dafür ein 'help'-Befehl auf dem nslookup-prompt).
- o "DNS" ist die Abkürzung für das "domain name system" und bezeichnet das System der nameserver.

31.3 Routen auf der Datenautobahn

- o »route« zeigt an, wie der Betriebssystemkern Datenpakete an die verschiedenen Empfänger verschickt.
- o »traceroute« dient zur Verfolgung des kompletten Weges, den Datenpakete zu einem bestimmten Empfänger einschlagen.

Bsp.: »traceroute nic.funet.fi«

- o "router" leiten Datenpakete an entfernte Maschinen und sorgen für den Datenaustausch zwischen Teilnetzen.
- o U.U. müssen die Pakete bis zum Ziel über mehrere router geleitet werden; darauf hat man aber keinen Einfluß.
- o "router" oder "gateway" wird meist synonym verwendet, wobei ein "gateway" eigentlich grundverschiedene Netze miteinander verbindet._

Konfigurationsdateien

Die tatsächlich notwendige Konfiguration beschränkt sich auf die folgenden Dateien. Hat man kein lokales Ethernet (, sondern nutzt man z.B. nur "ppp"), so entfällt sogar der zweite Teil von `/etc/init.d/networks'.

- o ``/etc/hostname'`: hostname der eigenen Maschine
- o ``/etc/resolv.conf'` enthält mindestens die IP-Adresse eines nameservers (Schlüsselwort "nameserver" voranstellen)

```
Bsp.:
-----8<----- hier schneiden -----8<-----
nameserver 192.168.23.2
domain    disney.comic.net
-----8<----- hier schneiden -----8<-----
```

- o ``/etc/init.d/network'` (Slackware: ``/etc/rc.d/inet1'`)

- Diese Datei ist ein shell-Skript (ändert sich bald).
- »ifconfig« gibt den Netzwerkschnittstellen IP-Adressen.
- Der Befehl »route« dient zum Festlegen der Datenwege.

```
Bsp.:
-----8<----- hier schneiden -----8<-----
# Konfiguration für "lucy.comic.net"
# Aktivierung des internen loopback-Netzes
ifconfig lo 127.0.0.1

# Der Weg zu uns selbst führt über uns selbst.
route add 127.0.0.1

# Aktivierung der Ethernet-Karte (setzt den
# Treiber für die entsprechende Karte im kernel
# voraus).
ifconfig eth0 192.168.22.2 \
broadcast 192.168.22.255 \
netmask 255.255.255.0

# Alle Maschinen im gleichen Netz werden direkt
# angesprochen.
# Dieser Eintrag ist notwendig, um den router zu
# erreichen, der die restliche "Arbeit" übernimmt.
route add -net peanuts.comic.net

# Der ganze Rest wird an den router geschickt.
# "route add 192.168.22.1" steckt bereits in
# obiger Zeile .
route add default gw 192.168.22.1

-----8<----- hier schneiden -----8<-----
```

—
Details zur Logistik

32.1 Mehr zu IP-Adressen

- o Um Datenpakete im großen Umfang und möglichst schnell an verschiedenste Empfänger schicken zu können muß das Format der Adressen besonders gut von Computern oder speziellen elektronischen Schaltkreisen zu verarbeiten sein.
- o Daher steckt das Konzept zur Verarbeitung der IP-Adressen im Binärformat (symbolische Namen zu analysieren kostet zuviel Zeit!)

- o Maschinen-intern entsprechen dem üblichen, dezimalen Format der IP-Adressen ("a.b.c.d") pro Zahl 8-bits:

192.168.22.3 entspricht 11000000.10101000.00010110.00000011

- o In dieser simplen Form kann die Adresse von sehr primitiven und enorm schnellen Logik-Bausteinen analysiert werden.

33.2 Lokale Netze

- o Bei der Planung eines lokalen Netzes hat man 2 Ziele:
 - direkter Datenverkehr zwischen allen lokalen Maschinen ohne Umweg über den router (unnötige Belastung)
 - Begrenzung der "Reichweite" der Datenpakete auf das kleinste mögliche Teil-Netz (aus Sicherheitsgründen und zum Schutz anderer Teil-Netze vor unnötiger Belastung)
- o Die Einstellungen zum Umsetzen dieser Ziele sind:

Bezeichnung	Beschreibung
network-address	Adresse des eigenen Teil-Netzes, Vergleich mit IP-Adresse gibt an, ob ein Datenpaket zum lokalen Netz gehört oder nicht
subnet-mask	Bitmaske zum Herausstanzen der Netzwerk-Adresse aus der IP-Adresse einer Maschine
address 192.168.22.3	11000000.10101000.00010110.00000011
netmask 255.255.255.0	11111111.11111111.11111111.00000000
=====log. AND =====	
network 192.168.22.0	11000000.10101000.00010110.00000000

In diesem Beispiel bilden alle 254 Maschinen mit der Adresse "192.168.22.XXX" ein gemeinsames Netzwerk.

- o Darüberhinaus gibt es noch die "broadcast-address", die eine Adresse für Nachrichten "an alle" im Teil-Netz ist.
- Konfigurationsdateien für Fortgeschrittene

- o `/etc/resolv.conf` kann enthalten:
 - Internet-Gebietsname (Schlüsselwort "domain" voranstellen)
 - IP-Adresse der nameserver (max. 3) (Schlüsselwort "nameserver" voranstellen)
 - Suchpfad für kurze Maschinennamen

```
Bsp.:
-----8<----- hier schneiden -----8<-----
domain    disney.comic.net
nameserver 192.168.23.2
nameserver 192.168.26.172
search    peanuts.comic.net franquin.comic.net
-----8<----- hier schneiden -----8<-----
```

- o `/etc/hosts`: Namen und Adressen wichtiger Maschinen; zur Umgehung

oder als lokale (!) Ergänzung des nameservers

Bsp.:

```
-----8<----- hier schneiden -----8<-----
127.0.0.1      localhost tick.comic.net tick
192.168.23.2   huba hubahuba huba.comic.net
192.168.22.2   lucy lucy.comic.net
-----8<----- hier schneiden -----8<-----
```

- o `/etc/host.conf` legt fest:
 - Reihenfolge: erst `/etc/hosts` oder erst DNS befragen?
 - Sind mehrere Namen für die eigenen Maschine erlaubt?
 - o `/etc/networks`: Nummern aller lokalen Netzwerke (ähnlich wie `/etc/hosts` für Netzwerkadressen)
 - o `/etc/hosts.allow`, `/etc/hosts.deny`: Welche Maschinen dürfen auf welche Dienste zugreifen?
 - o `/etc/inetd.conf`: Welche Dienste werden überhaupt angeboten?_
- Vergleich von Medien zur Datenübertragung
-

33.1 Von Linux unterstützte Medien

Name	Modul/Treiber	Beschreibung
Amatuerradio	AX.25	Datenübertragung via Funk
Arcnet	arcnet	"besseres" Ethernet
ATM	????	????
Parallelport	plip	Zweckentfremdung des Druckeranschlusses
Ethernet	kartenabhängig	Koaxialkabel (50 Ohm), ähnlich wie Radio- und Fernsehanschlußkabel
ISDN	????	????
loopback	loop	ein reiner Software-Trick, verbindet den Rechner intern mit sich selbst
Modem	ppp, slip	Byte für Byte per Telefon
Serieller Port	ppp, slip	Bit für Bit über Klingeldraht
Tokenring	ibmtr	Netzwerk mit Ringstruktur, Transfer via Token geregelt; zuverlässig

33.2 Leistung

Medium	Reichweite	Teilnehmer (*)	Durchsatz (kb/s)
Amatuerradio	einige km	????	1.2
Arcnet	650 m	ca. 50	310, 12500
ATM	????	????	????
Parallelport	2 m	2	????
Ethernet	300 m	ca. 50	1250, 12500
ISDN	-	-	> 8
loopback	0	1	> 1000
Modem	-	2	1.6 - 3.7
Serieller Port	30 m	2	1.6 - 11
Tokenring	100 m	8	500, 2000

(*)= ohne "zusätzliche" Hardware

33.3 Kosten

Medium	Kosten (ab DM)		Gebühren (ca. DM)
	Karte	Kabel	
Amatuerradio	????	????	????
Arcnet	????	????	????
ATM	????	????	????
Parallelport	-	20	0
Ethernet	70	2/m	????
ISDN	200	-	ISDN-Gebühren
loopback	0	0	0
Modem	140	-	Telefongebühren
Serieller Port	-	10	0
Tokenring	????	????	????_

Zugang zum Internet in Deutschland

Der Reichtum des Internet ist von den meisten Menschen nahezu alleine aus dem Spaß an der eigenen Kreativität erschaffen worden. Wer vom Internet ein optimales Medium zum Konsumieren von bunten Bildern und anderen aufregenden Sachen erwarten, wird daher bitter enttäuscht sein. Ohne eigenes Zutun wird man nicht ständig unterhalten (wie man das etwa vom Fernsehen gewohnt ist). Ebenso wenig eignet sich das Internet als Ersatz für die herkömmlichen Informationsquellen (Bücher, Filme, CD-ROMs), sondern höchstens als Ergänzung. Vor allem ist das Internet nämlich ein Kommunikationsmedium.

33.4 Wozu braucht man einen Internet-Zugang?

- o um mit anderen Internet-Teilnehmern in direkten Kontakt zu treten (z.B. um e-mail auszutauschen oder um am usenet teilzunehmen)
- o um ganz vorne bei der Entwicklung von Linux dabei zu sein (aktiv oder passiv)

33.5 Alternativen zum Internet-Zugang

- o Viele der natürlichen Reichtümer des Internet sind mittlerweile auf preisgünstigen CDs verfügbar
 - Abzüge der wichtigsten FTP-Server (sunsite/tsx-11)
 - tausende interessanter WWW-Seiten
 - Archive der newsgroups (auf CD aber relativ uninteressant; rechtlich umstritten)
- o Man sollte auf Aktualität Wert legen (und sich nicht von falschen Behauptungen irreführen lassen, nur die alten Versionen der Programme liefen stabil).
- o CDs sind schneller als jedes Modem ...

33.6 Besondere Internet-Anbieter (nicht unbedingt billig!)

- o "Individual Network e.V." (IN) Dachverband bundesweiter Betreibergemeinschaften, die Privatpersonen einen Internetzugang anbieten (starke regionale Ausrichtung, ca. DM 20-40/Monat)

Individual Network e. V., Scheideweg 65, D-26121 Oldenburg
IN-Info@Individual.NET (e-mail Adresse für Informationen)

- o "Verein zur Förderung der privat betriebenen Datenkommunikation e.V." (Vz* oder Sub-Netz)

????

- o §Verein zur Förderung eines Deutschen Forschungsnetzes e.V." (DFN-Verein), Zusammenschluss der Universitätsnetze; auch für Privatpersonen zugänglich (ab ca. DM 40/Monat)

DFN-Verein, Pariser Straße 44, 10707 Berlin
<http://www.shuttle.de/> (Anschluß für Privatpersonen)_

ISDN

[Habe kein ISDN, halte den Platz aber schonmal frei.]

—

—
 Ethernet

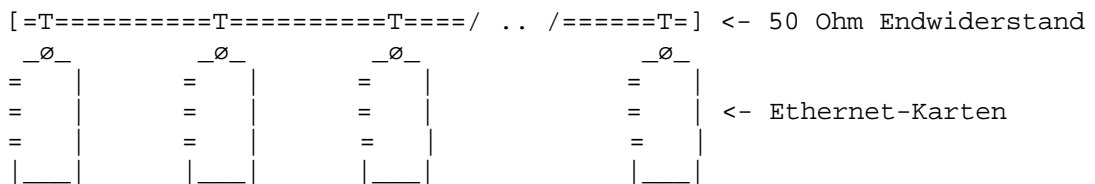
Bei Vernetzung mittels Ethernet ist pro Rechner eine "Ethernetkarte" (Einsteckkarte oder vergleichbar) erforderlich. Es gibt verschiedene Techniken, die im folgenden beschrieben werden.
 Für Privatpersonen ist ein auf Ethernet basierender Internetanschluß derzeit zu teuer, ein lokales Netz kann allerdings interessant sein.

35.1 "thick ethernet"

- o Diese Technik wird heute fast nicht mehr verwendet.

35.2 "thin ethernet"

- o Vorteile: schnell, preiswert, einfach
- o Nachteile: primitive Technik, etwas empfindlich
- o Das Netzkabel ist abgeschirmtes Koaxialkabel (wie beim Fernseher), Verbindungsstücke und -kupplungen sind in BNC-Technik ausgeführt.
- o Die Einsteckkarten werden mit T-förmigen Zwischenstücken an das Netzkabel angeschlossen.
- o Die Enden MÜSSEN mit Endwiderständen abgeschlossen werden.



35.3 "twisted pair"

[Entfällt aus Zeitmangel.]

—
 Anbindung via Modem

35.4 Modems

- o ein "Modem" empfängt Daten von der seriellen Schnittstelle und verschickt sie über die Telefonleitung (oder Klingeldraht)
- o "externe" Modems werden zwischen serieller Schnittstelle und Telefon geschaltet
- o "interne" Modems besitzen eigene serielle Schnittstellen und werden als Karte in den Computer gesteckt
- o ttyS0 - ttyS3 entsprechen COM1 - COM4 von DOS
- o Folgendes shell-Skript findet evtl. die zugehörige Gerätedatei:

```

-----8<----- hier schneiden -----8<-----
for i in 0 1 2; do
    if (chat -v -t 5 "" ATZ "" \\d "" ATZ \
        OK "" > /dev/ttyS$i < /dev/ttyS$i)
    then; echo "Modem on /dev/ttyS$i (= COM${i+1})"
    else; echo "No Modem found on ttyS$i"; fi; done
-----8<----- hier schneiden -----8<-----

```

- o "Baud" gibt an, wie oft sich das Trägersignal pro Sekunde ändert; mehr als 2400 Baud (entspricht 2.4kHz) sind über eine normale Telefonleitung kaum zu schaffen
- o "cps" ist die Anzahl der übertragenen Zeichen pro Sekunde ("chars per second"); durch geschickte Manipulation des Trägersignals ("Phasen-Modulation") kann diese Zahl höher sein, als "Baud"
- o Werden die Daten vor dem Übertragen im Modem komprimiert, so können noch höhere Geschwindigkeiten erzielt werden; allerdings lassen sich nicht alle Daten gleich gut komprimieren.
- o Bei Datenkompression (z.B. 1:4) muß die serielle Schnittstelle ein Vielfaches der Daten heranschaffen, z.B. 4*14.400 cps = 57600 cps; in den Linux-Standards sind aber nur 38400 cps als maximale Geschwindigkeit vorgesehen.
- o Der Befehl »setserial« liefert die höheren Übertragungsraten, wenn ein Programm "38400 bps" verlangt.


```

      Bsp.:  »setserial /dev/ttyS2 spd_hi« (57600 bps)
           »setserial /dev/ttyS2 spd_vhi« (115200 bps)
      
```
- o Konfiguration der seriellen Schnittstellen mit dem Befehl »setserial«,


```

      Bsp.:  »setserial /dev/ttyS2 irq 5«
      
```
- o Obige Befehle findet man in der Datei `/etc/rc.boot/0setserial'.

35.5 Standards

Name	Übertragungstechnik
V.2	300 baud
V.22bis	2400 baud
V.32	9600 baud
V.32bis	14400 baud
V.34, V.FC	28800 baud +Leitungstörungen senken die Übertragungsrate nur vorübergehend
V.42bis	Datenkompression bis max. 1:4_

TCP/IP per Modem

36.1 Übertragungstechniken

Name	Protokoll	Bemerkung
PPP	point-to-point protocol	weitgehend automatische Konfiguration, modern

SLIP	serial line internet protocol	weit verbreitet, veraltet
CSLIP	compressed SLIP	Kompression der Verwaltungsinformation (nicht der Daten)
slirp	????	
term	????	

36.2 PPP

- o "PPP" ist eine Technik zum Übertragen von Internet-Datenpaketen (TCP/IP) via Modem.
- o Eine ausführliche Beschreibung zu PPP findet man unter der URL

`http://www.mi.uni-koeln.de/linux/ppp`
- o Beide Seiten der Übertragungstrecke (Modem privater PC <-> Modem im Internet-Rechner) müssen PPP beherrschen (beim Internet-Anbieter nachfragen!).
- o Um unter Linux PPP nutzen zu können, muß der Betriebssystemkern (siehe 52.X) den PPP-Treiber enthalten (fest oder als Modul).
- o Je höher die kernel-Version, desto höher muß i.d.R. auch die Version des verwendeten PPP-Pakets sein.
- o Im PPP-Paket für Linux sind drei Programme enthalten:

```

»chat«  Das chat-Programm baut die Verbindung auf.
»pppd«  der PPP-Daemon konfiguriert und betreibt die PPP-
        Verbindung, er läuft auf beiden Seiten der
        Verbindung
»pppstats«  Statistiken über die Auslastung der Verbindung

```

36.3 Der Verbindungsaufbau

- o Das Programm »chat« erwartet bestimmte Antworten vom Modem und kann mit Befehlen und Wörtern antworten.

```
Bsp.:  »chat ' ATZ OK ' < /dev/modem > /dev/modem«
```

- o Meist dient es dem Wählen und dem Einloggen.
- o Mehrere Wählversuche und/oder Telefonnummern müssen durch ein kleines shell-Skript probiert werden (``/etc/ppp/ppp-connect'`).

```

-----8----- hier schneiden -----8-----
ABORT          BUSY
ABORT          NO ANSWER
'              ATDT888333777
CONNECT       '
ogin:         winfried
assword:      top-secret
-----8----- hier schneiden -----8-----

```

In diesem Beispiel wird mit dem Tonwahlverfahren die Nummer "888333777" gewählt und der Benutzer "winfried" mit dem password "top-secret" eingeloggt. Mit dem Parameter '-f' übergibt man dem »chat«-Programm ein solches Skript.

36.4 Der "pppd"

- o Konfigurationsdateien im Verzeichnis ``/etc/ppp'`

```

`chatfile'      Schlüsselwörter, die beim Verbindungsaufbau
                erwartet werden und die zugehörigen Antworten
`chap-secrets'
`ip-down'
`ip-up'
`options'      Betriebsparameter, die mit der Gegenseite
                vereinbart werden sollen (keine Garantie)
`pap-secrets'
`ppp-connect'  zum Aufbau der Verbindung

```

- o Der "pppd" kann auf als Server für eingehende Verbindungen arbeiten.

[Fortsetzung PPP entfällt aus Zeitmangel.]

—
Benutzung anderer Maschinen im Netz und Zubehör

37.1 Herkömmliche Programme

- o Terminalsitzungen: »telnet«
 Bsp.: »telnet localhost«
- o Datentransfer: »ftp«
 Bsp.: »ftp localhost«

37.2 Berkeley r-utilities

- o Die »r-utitlies« erlauben die Arbeit mit anderen Maschinen ohne die Eingabe eines Namen und Geheimwortes.
- o Stattdessen erfolgt die Autorisierung über Einträge in den Dateien

Datei	Format einer Zeile
/etc/hosts.equiv	"-Rechnernamen", "+Rechnernamen"
~/rhosts	"Rechnername", "Rechnername Benutzer"

- o Vorteile:
 - passwords werden nicht durch das Netz geschickt
 - bequemer
- o Nachteile:
 - Systemverwalter der anderen Maschine erhalten Zugang
 - ein anderer Rechner kann unter falschem Namen Zugang bekommen

o Übersicht

Funktion	Programm	Beispiel
telnet	»rlogin«	»rlogin localhost«
Programm starten	»rsh«	»rsh localhost -- ls -l /«
Dateien kopieren	»rcp«	»rcp v.txt localhost:v.tex«
X11-Programme starten (bauen auf »rsh« auf)	»xon«, »xrsh«	»xon localhost netscape«

37.3 Sicherheit auf allen Wegen: »secure shell«

[Entfällt aus Zeitmangel.]

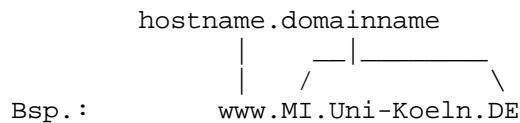
37.4 Zubehör

- o Netz-Radar: »ping«
 Bsp.: »ping localhost«
- o Mitschnitt des Datenverkehrs: »tcpdump«
 Bsp.: »tcpdump -i lo«
- o »tkined« ist ein universelles Programm zum Beobachten des
 Netzes mit einer grafischen Oberfläche_

Kleine Internet-Namenskunde

37.5 Grundlegendes

- o um auf Rechner im Internet zuzugreifen bedient man sich
 symbolischer in der Form

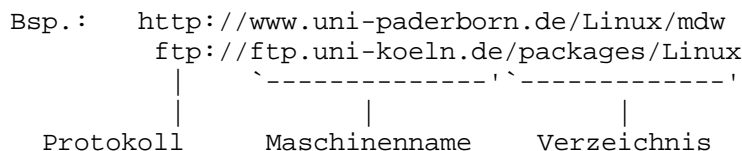


- o Der hostname ist bei öffentlich zugänglichen Dienstleistungen
 in vielen Fällen dem Zweck der Maschine angepaßt, der domainname
 gibt Auskunft über die Organisation oder Firma.
- o Mit diesem Wissen kann man die Adressen der meisten Informations-
 server im Internet erraten.

Dienst	Protokoll	Beispiel	Bescheibung
archie	????	archie.th-darstadt.de	Suche in den Inhaltsverz. der ftp-Server
ftp	ftp	ftp.uni-koeln.de	siehe 12.3
news	nntp	news.uni-koeln.de	siehe 40.1
www	http	www.uni-koeln.de	World Wide Web

37.6 Das URL-Format

- o speziell für die WWW-Browser wurde das URL-Format entwickelt, das
 die Adressierung aller Informationen, Dateien und andere Ressourcen
 im Internet in einem einheitlichen Format erlaubt
- o in einer URL-Adresse stecken die folgenden Informationen:
 - Protokoll
 - voller Name der betreffenden Maschine
 (- evtl. Portnummer [Nummer der Eingangstüre])
 - Dateiname (+Verzeichnisname) oder password



37.7 Kleine Namenskunde (Endungen)

Kürzel	Bedeutung	Kürzel	Bedeutung
ac	academic	de	Deutschland
u-	Universität	com	Firma
fi	Finnland	edu	Ausbildungseinrichtung
		net	Netzwerk

38.1 Grundlegendes

- o Elektronische Post wird als "e-mail" ("electronic mail") bezeichnet.
- o e-mail wird i.d.R. sofort zugestellt, und das weltweit
- o SMTP: "simple mail transport protocol"
- o "emaillieren": mit Email überziehen
- o Eine e-mail setzt sich aus folgenden Feldern zusammen:

From:	Absender
Date:	Datum ("des Poststempels")
Reply-To:	Rückadresse (wenn ungleich Absender)
To:	an wen ist die e-mail gerichtet
Subject:	worum geht es in der e-mail ("Betreff")
Cc:	Kohledurchschlag ("carbon copy") an ...
Bcc:	Durchschlag ohne Wissen des Empfängers ("blind carbon copy") an ...
Fcc:	Kopie für die eigenen Unterlagen ("file carbon copy") in die Datei ...
Recieved:	Stationen im Internet, über die die e-mail weitergeleitet wurde

- o Zusammen bilden oben genannte Felder den "header" der e-mail.
- o Der eigentliche Text wird als "body" oder "message text" bezeichnet.

38.2 Adressen

- o e-mail-Adressen haben die Form

"Name@Anschrift"

- o `@` wird "at" ausgesprochen (engl. "bei")
- o Als "Name" kommen in Frage:
 - der login-Name des Benutzers (max. 8 Buchstaben!)
 - ein alias für den login-Namen (`/etc/aliases` auf der Empfängermaschine)
- o Als "Anschrift" kommen in Frage:
 - der Rechnername einer im Internet existierenden Workstation
 - der Name einer Sammelstelle für alle Rechner, auf denen der Benutzer arbeitet (oft der domainname)
- o "postmaster" ist der Ansprechpartner bei Problemen mit der e-mail
Bsp.: "postmaster@localhost"
- o e-mail-Adressen muß man kennen, es gibt keine vollständigen Adressbücher oder eine andere zuverlässige Einrichtung, um an e-mail-Adressen zu kommen
- o Der X.500-Dienst ist ein kleines Adressbuch für e-mail-Adressen
Bsp.: »finger truemper@X500.Uni-Koeln.DE«
- o WWW-Interface zum X.500-Dienst unter der URL

<http://www.uni-koeln.de/dienste/x500.html>

38.3 Programme zum Lesen und Schreiben von e-mail

Programmname	Eigenschaften
--------------	---------------

pine einfach, komfortabel, Speicherfresser

elm	Standard-Programm, unkomfortabel
emacs	"Alt-x rmail"; nur Kaffekochen kann er nicht ...
tkmail	mit graphischer Oberfläche; einfach

- o Unterschrift/Brieffuß: Datei `~/signature' (wird von vielen Programmen automatisch an die ausgehende e-mail angehängt)

38.4 Text, Anlagen und Beilagen

- o grundsätzlich kann man per mail nur ascii-Text verschicken (also keine Umlaute, Bilder, Programme, usw.)
- o zwei Verfahren zur Codierung beliebiger Daten in (dann unsinnigen) ascii-Text sind verbreitet:

uuencode	"usenet encode" (?); zum Decodieren muß die e-mail in eine Datei gespeichert werden, die dann von »uudecode« gelesen wird Bsp.: »uuencode lnx.gif lnx.gif > lnx.uu« »uudecode saved_e-mail.txt«
----------	---

MIME	"multipurpose internet mail extensions", eine Art Standard für "multimedia"-mail; wird von vielen Programmen automatisch decodiert
------	--

- o Beilagen werden als "attachement" bezeichnet
- o `/etc/mailcap' gibt an, welche Programme zum Betrachten der Beilagen aufgerufen werden sollen (falls das e-mail-Programm dies nicht selbst kann)

38.5 Verschicken von e-mail

- o e-mail-Programme reichen die e-mail weiter an
 - die Daemonen »smtpd« oder »sendmail« auf dem eigenen Rechner oder
 - an einen "SMTP-Server" (unkonventionell: "mail-Server"), d.h. einem Rechner, auf dem diese Daemonen laufen
- o eine Empfangsbestätigung erfolgt i.d.R. NICHT (und man sollte dies auch tunlichst unterlassen, z.B. wegen der Mailinglisten)
- o ist eine e-mail überhaupt nicht zustellbar, so wird sie mir einer kurzen Notiz zurück an den Absender geschickt
- o ist eine e-mail vorübergehend nicht zustellbar (z.B. Netzausfall), so wird sie im Verzeichnis `/var/spool/mqueue' ("mail queue") gespeichert und in regelmässigen Abständen wird die Zustellung erneut versucht (max. 4 Tage lang)

38.6 Empfang und Zustellung von e-mail

- o die Daemonen »smtpd« oder »sendmail« nehmen mail aus dem Internet entgegen und führen folgende Aktionen durch:
 - durch einen Eintrag in der Datei `/etc/aliases' wird die e-mail auf andere Maschinen oder tatsächlich existierende Benutzer umgeleitet werden
Bsp.: »Winfried_Truemper: truemper«
 - falls ein Nachsendeauftrag in der Datei `~/forward' des Empfängers (= `~`) besteht, wird die e-mail an die darin enthaltene e-mail-Adresse geschickt

Bsp.: "truemper@MI.Uni-Koeln.DE"
"|/usr/bin/vacation"

- ansonsten wird die e-mail im Verzeichnis ``/var/spool/mail'` zur Abholung durch den Benutzer hinterlegt, und zwar tragen die "Briefkasten"-Dateien die Namen der Benutzer
- o Gängige e-mail-Programme gucken automatisch in diese Briefkästen.
- o mit dem Programm »popmail« kann man (mit dem eigenen password) die e-mail aus den Briefkästen anderer Rechner herausholen (z.B. um sie "offline" zuhause oder unter DOS zu lesen)
- o weniger verbreitet, aber sicherer und komfortabler ist die "imap"-Technik: die Verwaltung der mail verbleibt auf dem Server

38.7 Zubehör

- o Briefkastenflaggen:
 - »xbiff« einfachstes Modell
 - »xpbfiff« zeigt ausserdem Absender und Betreff in einem extra-Fenster an
 - »maileds« Tastatur-LEDs werden zur Benachrichtigung über neue e-mail zweckentfremdet
- o automatische e-mail "Beantworter"
 - »vacation« schickt dem Absender die Nachricht aus der Datei `~/vacation.msg` zu; z.B. bei (im voraus bekannter,) mehrtägiger Abwesenheit
 - »BCR MailHandler« schickt auf Anforderung bestimmte Dateien zu
- o Verschiedenes
 - »procmail« leistungsfähiger Filter zum automatischen Sortieren, Löschen, Weiterverarbeiten von e-mail ("vacation forte")
 - »splitdigest« erzeugt aus "digest" wieder einzelne mails
 - »from« Kurzübersicht der Neueingänge
 - »pgp« Briefsiegel und Verschlüsselung
 - »super-cite« Zitate im emacs
- o MIME-decoder
 - »mimedec.c«
 - »mpack«

Mailinglisten

39.1 Grundlegendes

- o "Mailinglisten" verteilen e-mail automatisch an eine grössere Anzahl von Personen.
- o Sie dienen der Kommunikation via e-mail, ohne die Teilnehmer mit der Verwaltung der Adressen der anderen Teilnehmer zu belasten.
- o Zu jeder Mailingliste existieren zwei e-mail Adressen:
 - eine "Steueradresse" bei der man den Bezug der Mailingliste abonnieren und stornieren kann
 - die "Verteileradresse" schickt alle e-mail, die an sie gerichtet wird, an die Abonnenten weiter
- o nicht alle Mailinglisten sind öffentlich zugänglich
- o viele öffentliche Mailinglisten werden archiviert und die Archive sind per WWW lesbar, daher IST ES NICHT UNBEDINGT NOTWENDIG, EINE MAILINGLISTE ZU ABONNIEREN
- o es gibt gerade für Linux einige Mailinglisten mit einigen hundert

Beiträgen pro Woche; Vorsicht!

- o um diesem Problem zubegegnen und die wertvollen Informationen trotzdem einer breiten Masse zugänglich zu machen, gibt es »digest« (ein subjektives "best of" einer Mailingliste)

39.2 »majordomo«

- o die meisten mailinglisten sind mit dem Programm »majordomo« realisiert
- o die Befehle für die majordomo-Steueradresse müssen formlos im e-mail Text ("message body") stehen, NICHT im Betreff ("subject")
- o einige der majordomo-Befehle:

»help«	schickt eine e-mail zurück, in der die Bedienung und alle Befehle des majordomo erklärt werden
»lists«	listet die Namen aller Mailinglisten auf, die von dem betreffenden Rechner verwaltet werden
»subscribe«	gefolgt vom Namen einer Mailingliste richtet ein Abonnement für diese ein
»unsubscribe«	gefolgt vom Namen einer Mailingliste löscht das Abonnement für diese

39.3 Linux-spezifische Mailinglisten

- o »Yggdrasil Computing« organisiert (lies: bezahlt) für die Linux-Gemeinde die Umleitung aller wichtigen Mailinglisten in entsprechende newsgroups. Dadurch ist ein gefahrloses und kostengünstiges Mitlesen für alle möglich.
Informationen unter
????

- o Man sollte als normaler Benutzer die newsgroup der "linux.*"-Hierarchie dem Abonnement der mailinglisten vorziehen._

Das »usenet«

40.1 »usenet« und »newsgroups«

- o eine »newsgroup« beinhaltet e-mail-artige Diskussionsbeiträgen zu einem Oberthema; diese werden als »Artikel« bezeichnet
- o i.a. kann sich jeder an der Diskussion beteiligen, d.h. Artikel lesen und verfassen
- o in »moderierten« newsgroups bestimmt ein Moderator, welche Artikel in der newsgroup erscheinen (z.B. um Mißbrauch zu verhindern)
- o die Gesamtheit aller Rechner, die das Lesen und Schreiben von Artikeln erlauben, nennt man »usenet«
- o der Lebensnerv von Linux war und ist das usenet

40.2 Organisation der newsgroups

- o newsgroups sind hierarchisch nach Themen geordnet
- o Unter-Hierarchien werden durch Punkte voneinander getrennt
Bsp.: »comp.os.Linux.announce«
- o Entstehung und Auflösung durch demokratische Wahl der usenet-Teilnehmer (siehe newsgroup »de.newusers«)

40.3 Infrastruktur des usenet

- o die Artikel werden auf festen Wegen (Universitäten, Organisationen) durch das usenet gereicht, entweder einzeln und effektiv per NNTP (»network news transport protocol«) oder in großen Paketen via uucp (»unix-to-unix copy«)
- o die Laufzeit kann einige Tage betragen

40.4 newsreader

- o »newsreader« sind Programme zum Lesen und Verfassen von Artikeln
- o »threaded newsreader«: fassen Frage und Antworten unter einer Überschrift zusammen (unbedingt empfehlenswert)
- o Umgebungsvariablen zur Konfiguration von newsreadern:
 - »NNTPSERVER« hostname des NNTP-Servers
 - »ORGANIZATION« Organisation (z.B. »home sweet home«)
- o einige newsreader
 - »xskim« kostensparend und ohne Aufwand news via Modem lesen
 - »tin« nicht besonders aufregend, aber einfach und funktionell
 - »Gnus« der Overkill, nur für Mutige; Infos unter der URL <http://www.ifi.uio.no/~larsi>
- o ohne ständige Netzanbindung (»offline«) news lesen und schreiben mit »skim«, »slurp«, »newsflash«, »postit« (zum Selberbasteln) oder eine fertige Beschreibung von:

40.5 usenet-Jargon ftp://pople.pc.uni-koeln.de/pub/news.ps_

- »CfD«: "call for votes" (Aufruf zur Stimmabgabe)
- »crossposting«: Artikel, die an mehrere newsgroups gleichzeitig gerichtet sind; i.d.R. ist dies absolut UNERWÜNSCHT
- »flame«: Artikel, die die Diskussion kräftig anheizen (meist unsachlich und verletzend)
- »follow up«: Artikel mit direktem Bezug auf einen vorhergehenden (entspricht dem "reply" der e-mail)
- »nettiquette«: Knigge für das Internet, Netz-Benehmen
- »off-topic«: themenfremd
- »posting«: Artikel in den news
- »RfD«: "request for discussion" (Aufruf zur Diskussion über die Neueinrichtung einer newsgroup)

40.6 Archive des usenets

- o bevor man im usenet eine Frage stellt, sollte man einen Blick in die Archive werfen; evtl. ist die Frage dort samt zugehöriger Antwort verzeichnet
- o die »FAQs« der einzelnen newsgroups findet man auf verschiedenen FTP-Servern (in Verzeichnisse mit Namen der newsgroups ein sortiert)
 - <ftp://ftp.uni-paderborn.de/FAQ> (Deutschland)
- o »DejaNews« kann per WWW nach Stichworten durchsucht werden:
 - <http://www.dejanews.com/forms/dnsetfilter.html>

[dies ist nicht die Startseite, sondern die Seite mit dem Filter: »*Linux*« eingeben]

40.6 Einige newsgroups und Hierarchien

alt.* alternate newsgroups

```

comp.*           Computer-bezogene Themen
comp.lang.*     Programmiersprachen
comp.os.*       Betriebssysteme ["operating systems"]
comp.os.Linux.* Linux-spezifische Themen
comp.os.Linux.advocacy Linux gegen den Rest der Welt
comp.os.Linux.announce Ankündigungen (moderiert)
comp.os.Linux.answers Dokumentation (moderiert)
comp.unix.*
linux.*         siehe 39.3
misc.*         miscellaneous (Verschiedenes)
news.*
rec.*          recreation (Hobby und Freizeit)
sci.*         science (Wissenschaft)
soc.*         society (Gesellschaft)
talk.*        talk (Unterhaltung)
de.*          deutschsprachige newsgroups, Wiederholung der
              englischen Original-Struktur

de.comp.os.Linux
de.newusers
de.talk.jokes  Witze
de.test       Zum Ausprobieren und Testen der newssoftware_

```

Kommunikation mit anderen Benutzern

Hinweis:

um sich vor einer Notizenflut anderer Benutzer zu schützen, kann man mit dem Kommando »mesg n« deren Empfang abstellen (betrifft »write«- und »talk«-ähnliche Programme)

41.1 Benachrichtigen

- o »write«: Nachricht direkt auf Bildschirm des anderen schreiben
- o »wall«: dito, aber an alle
- o »rusers«, »rwho«, »finger«: wer ist wo zu finden

41.2 Palaver in Echtzeit: »talk«, »ntalk«, »ytalk«

- o »ytalk« ist die komfortabelste Version (»ESC« aktiviert Menü)
Bsp.: »mesg y; ytalk \$LOGNAME«
- o mit SUN-Maschinen gibt es hinsichtlich talk oft Probleme, aber auch sonst ist der talk-Dienst häufig "ausser Betrieb"

41.3 Online-Diskussionsforen: »irc«

- o »irc« steht für "Internet Relay Chat" (aber: »man ircII«)
- o in den verschiedenen Diskussionsforen wird LIVE diskutiert; der Stil ist eher chaotisch
Bsp.: »irc -c Linux irc.informatik.rwth-aachen.de«
- o Befehle beginnen mit dem Zeichen `/' (slash)
Bsp.: `/'quit', `/'help'

41.4 Fax

- o die Programmpakete »mgetty+sendfax«, »efax«, »hylafax« erlauben das Verschicken und Empfangen von Faxen via (Fax-)Modem
- o gegenüber e-mail hat Fax unter Linux nur eine geringe Bedeutung

41.5 Spiele

Spiele für Linux basieren i.d.R. auf einem Netzwerk und erlauben mindestens das austauschen von kurzen Nachrichten; auf möglichst realistische Grafiken wird keinen Wert gelegt. Eine Auswahl :

- o »mud«: textbasiertes Rollenspiel, meist im Stil von
»Dungeons&Dragons«, Informationen in den newsgroups
»rec.games.mud.announce« und »de.alt.mud«
- o »xpilot«: Raumschiff-Kriegsspiel in 2D-Welt, Team-Modus möglich;
Information via »telnet xpilot.cs.uit.no 4400«
- o Schach: »xboard« (Oberfläche f. »gnuchess«); Kontaktaufnahme zum
Schach-Server mit »xboard -ics True -icshost chess.lm.com«_

41.6 4 goldene Regeln für den Umgang mit anderen Netzteilnehmern

- o schreib nichts, was Du dem anderen nicht auch ins Gesicht sagen würdest; am anderen Ende der Leitung sitzt ein Mensch wie Du und ich, kein gefühlsloser Computer
- o wenn Dir jemand Beleidigungen an den Kopf wirft: mach Dir nichts draus, manche Menschen verlieren vor dem Bildschirm total das Gefühl für den anderen Menschen (also: nicht direkt totschiagen!)
- o laß Dir unaufgefordert zugesandten Werbemüll und anderen Ramsch nicht gefallen
- o ohne präzise Informationen KANN Dir nicht geholfen werden - oder Du zahlst wie bei kommerziellen hotlines kräftig dafür, daß man Dir die Würmer einzeln aus der Nase zieht: "Ist Ihr Computer eingeschaltet?" Kommt nicht von ungefähr.
- o zieh Dich warm an, sprech nicht mit fremden Leuten und räum' endlich Dein Zimmer auf

41.7 Häufig vorkommende Abkürzungen

AFAIK as far as I know (soweit ich weiss)
ASAP as soon as possible (so schnell wie möglich)
BTW by the way (übrigens; wo wir gerade dabei sind)
cu see you (wir sehen uns)
FYI for your information (zur Kenntnisnahme)
gr&d grinning and ducking (grinst und duckt sich)
IMHO in my humble opinion (meiner bescheidenen Meinung nach)
in my honest opinion (meiner ehrlichen Meinung nach)
LOL laughing out loud (lautes Lachen)
MORF man or female? (Mann oder Frau?)
OIC oh, I see (ich verstehe)
ROTFL rolling on the floor, laughing (die Person wälzt sich vor Lachen am Boden)
RSN real soon now (schon bald)
RTFM "read the fucking manual" (ein gängiger Hinweis, sich mit dem Handbuch auseinanderzusetzen)
THX thanks (Danke)

weitere unter der URL:
????

41.8 Smilies (Seite um 90° im Uhrzeigersinn drehen)

:-) Standard-Smiley
;-) zwinkender Smiley
:-/ zweifelnder Smiley
:-(unglücklicher oder verärgertes Smiley
:-< trauriger Smiley

:~0 diesem Smiley steht der Mund vor Schreck offen
(~: Benutzer ist Linkshänder

weitere unter der URL:

????

im Buchhandel: »Smileys«, Sanderson/Dougherty, O'Reilly-Verlag
Koordination der Serviceleistungen

42.1 Server

- o »Server« sind Daemonen oder Maschinen, die anderen Programmen oder Maschinen Daten oder Dienstleistungen zur Verfügung stellen
- o es gibt mehr als 50 Daemonen mit Server-Funktion für Linux
- o würden diese Programme stets gleichzeitig laufen, bräuchte man dafür alleine mehrere MB Arbeitsspeicher (RAM)

42.2 Der Internet-Super-Daemon »inetd«

- o der Daemon »inetd« startet bei Anforderung von Netzwerkdiensten die speziell notwendigen Daemonen
- o Konfiguration durch '/etc/inetd.conf'
- o die meisten Dienste sind freigegeben, also Vorsicht!
- o alle Daemonen, deren Programmnamen mit »in.« beginnt, müssen vom »inetd« gestartet werden
- o ein simple Zugriffsbeschränkung wird durch den »tcpd« gewährleistet, der vor dem Aufruf der eigentlichen Daemonen die Gegenseite der Verbindung genau überprüft; Konfiguration mit
 - '/etc/hosts.allow': wer darf was
 - '/etc/hosts.deny': wer darf was nicht
- (siehe » man hosts_access«)
- o »xinetd« ist eine erweiterte Version des »inetd«

42.3 Standalone-Daemonen

- o viele Daemonen brauchen zum Start einige Sekundenbruchteile Zeit, um alle Konfigurationsdateien zu lesen und sich zu initialisieren
- o werden solche Daemonen intensiv genutzt (man denke an den http-Daemon für WWW-Anfragen), so ist der Umweg über den »inetd« (startet den daemon für jede Anfrage neu) viel zu langsam
- o stattdessen müssen solche Daemonen als eigenständige Prozesse laufen; z.B. »sendmail« und »httpd«
- o Daemonen, die sich selbst verwalten können, bezeichnet man als »standalone«
- o alle Daemonen, deren Programmnamen mit »in.« beginnt, müssen vom »inetd« gestartet werden
- o Zugriffsregelungen: individuell

42.4 Die RPC-Funktionen

- o »RPC« steht für »Remote Procedure Call« und ist der Oberbegriff für eine Reihe von ????
- o die »RPC-Daemonen« sind meist erkennbar an vorangestelltem »rpc.« im Programmnamen (Ausnahmen z.B.: »ypbind« »yp serv«)
- o Zugriffsregelungen: individuell und wie bei »tcpd«

—
Sicherheitsaspekte

—

—

—

—
X-Windows-System

45.1 Grundlegende Eigenschaften von X11

- o "X11" ist der Unterbau der grafischen Benutzeroberfläche von Linux. Es wurde in den 80er Jahren am "Massachusetts Institute of Technology" (MIT) entwickelt und wegen seiner enormen Bedeutung dem dafür gegründeten "X Consortium" übereignet.
- o Das reine "X11" stellt lediglich grafische Informationen dar, die es von anderen Programmen erhält.
- o Die Verwaltung der Fenster, deren Dekoration und Aussehen wird vom Windowmanager (Fenstermanager) gesteuert.

-----8<----- hier schneiden -----8<-----

```
/usr/bin/X11/X > X-log 2>&1 &
```

```
# Ein grauer Bildschirm erscheint.  
# Zurückwechseln mit <Strg-Alt-F1>
```

```
xterm -ls -sb -display :0.0 &
```

```
# Scheinbar passiert nichts. Die virtuellen Konsolen nach X  
# absuchen und die Maus in das neue Fenster platzieren.
```

```
/usr/bin/X11/fvwm &
```

-----8<----- hier schneiden -----8<-----

45.2 Netzwerkfähigkeiten

- o "X11" ist netzwerkfähig. Es vergibt an den Monitor eine Adresse, über den er im gesamten Netzwerk (z.B. Internet) angesprochen werden kann.
- o Das Format der Monitor-Adresse ist

```
track.comic.net:0.0  
`-----' | |  
          | | Nr. des Monitors (bei Verteilung  
hostname  | | des Bilds auf mehrere Monitore)  
          | |  
          | | Nummer des "X11"
```

- o Wird 'X' mehrmals gestartet so erhalten die daraus entstehenden Oberflächen anstelle der Endung ':0.0' die Endungen ':1.0', ':2.0' usw.

```
Bsp.: »/usr/bin/X11/X :1 «
```

- o Anhand der Umgebungsvariable "DISPLAY" wird Programmen mitgeteilt, wie die Adresse des Monitors lautet, den sie benutzen sollen.

Bsp.: »echo \$DISPLAY«

- o Die Option '-display' überschreibt den Wert von "DISPLAY".

Bsp.: »xterm -display localhost:0.0«_

45.3 Zugriffskontrolle

- o Die Netzwerkfähigkeiten von X11 machen eine Zugriffskontrolle unbedingt notwendig. Dabei geht es NICHT NUR um das unberechtigte Öffnen von Fenstern, sondern vielmehr muß verhindert werden, daß die Vorgänge in den Fenstern ABGEHÖRT werden können (man denke an die Eingabe eines passwords). Das Abhören ist nämlich mit primitivsten Mitteln möglich, siehe »man xev«.
- o Unter Linux regeln z.Zt. zwei Verfahren den Zugriff auf die eigene X-Sitzung, die sich etwa so einordnen lassen:
 - Erlaubnis für alle Programme von einer bestimmten Maschine - ohne Unterscheidung der Benutzer
 - Erlaubnis für alle Programme, die einen bestimmten Ausweis vorlegen können
- o Das Programm »xhost« legt fest, von welchen Maschinen jeder Benutzer zugreifen darf. Man sollte mit diesem Programm keinerlei Erlaubnis erteilen, auch nicht den Benutzern der eigenen Maschine (s.o.).

Bsp.: »xhost - « (Verbot für alle Maschinen)
- o Der Standard unter Linux ist die Methode "MIT-MAGIC-COOKIE-1", bei der Programme einen Ausweis (üblicherweise in `~/Xauthority`) vorlegen müssen. Die Datei, die diesen Ausweis enthält darf natürlich nur für den Eigentümer les- und schreibbar sein (das Ausführungsrecht muß NICHT gesetzt werden).
- o Dieser Ausweis kann z.B. mit »xauth« weitergegeben werden (siehe auch »xon« oder »xrsh«).

45.4 Windowmanager

- o "virtual-Windowmanager" vervielfachen die nutzbare Oberfläche; der max. sichtbaren Ausschnitt ist natürlich durch den Monitor festgelegt.
- o Unter Linux wird meist der »fvwm«-Windowmanager eingesetzt.

Übersicht Windowmangager

Name	Konfigurationsdateien	Beschreibung
»fvwm«	`/etc/X11/fvwm/system.fvwmrc' `~/fvwmrc'	"Free Virtual Window Manager"; vielseitig
»mwm«		"Motif Window Manager" (kommerzielles Programm)
»olvwm«	`/usr/openwin/lib' `~/openwin-init'	"Open Look Virtual Window Manager"; stilvoll
»twm«	`/etc/X11/twm/system.twmrc' `~/twmrc'	"Tab Window Manager"; sehr schlicht_

Mehr über "X"

46.1 Begriffe

"X-Server": das Programm, das die Grafikbefehle der anderen Programme in ein Bild auf dem Monitor umwandelt.

"root-window": der "Hintergrund".

"cut&paste": markieren und an anderer Stelle einfügen.

46.2 X für Linux

- o Es gibt unter Linux mehrere Programme, die die Funktion von "X11" übernehmen; sie entsprechen dem Release 6 (X11R6):
 - "XFree86" vom "XFree86 Project, Inc."
 - "AcceleratedX" von der Firma "X Inside Inc." ist schneller als "XFree86" (noch schneller?!), aber nicht kostenlos

46.3 XFree86, Version 3.1.2

- o Um die Hardware möglichst optimal auszunutzen, wurden für die verschiedenen Familien von Grafikkarten (z.B. "S3", "Mach") spezielle Versionen des XFree86 geschrieben.
Bsp.: »dir /usr/X11R6/bin/X*«
- o Trotzdem ist noch eine Konfiguration notwendig, da z.B. der Monitor ebenfalls optimal ausgenutzt werden soll, andererseits nicht überlastet werden darf.
- o Zur Erzeugung der Konfigurationsdatei `/etc/X11/XF86Config` kann das Programm `»/usr/X11R6/bin/xf86config«` benutzt werden.
- o XFree86 übernimmt beim Start die Tastenbelegung der Textkonsolen ... mit Ausnahme der Taste `<AltGr>`. Bei der Konfiguration muß die Frage
Do you want to enable these bindings for the Alt keys?
mit 'y' beantwortet werden, um auch diese zu aktivieren.

Tastenkombination	Funktion
<code><Strg-Alt-BackSpace></code>	beendet XFree86 sofort (Notbremse)
<code><Strg-Alt-Numpad+></code>	nächsthöhere Auflösung wählen
<code><Strg-Alt-Numpad-></code>	nächstniedrige Auflösung wählen
<code><Strg-Alt-F1></code>	zurück auf die Textkonsole wechseln

- o XFree86 benötigt eine "leere" virtuelle Konsole; ohne weitere Angaben wird die erste "leere" Konsole beschlagnahmt.
Bsp.: `»/usr/X11R6/bin/X vt4«_`

46.5 Starten von X11

- o Wegen der Netzwerkfähigkeiten gibt es mehrere Möglichkeiten zum Starten und Arbeiten mit X11:

Prg.	Konfigurationsdatei	Beschreibung
startx	<code>/etc/X11/xinit/xinitrc'</code>	Aufruf nach dem Einloggen an virtueller Textkonsole
openwin	<code>/usr/openwin/lib/Xinitrc'</code>	dito; startet "Open Windows"
xdm	<code>/etc/X11/xdm/Xsession'</code>	login-Fenster für direkten

Zugang zum X11

»X -query localhost«	nur X auf der eigenen Maschine, alles andere auf "localhost"
»X -indirect localhost«	Liste von Maschinen anfordern, die eine X-Sitzung erlauben

46.6 Der "X-Display Manager"

- o »xdm«, der "X-Display Manager", sorgt für einen ständigen Betrieb des XFree86 auf der Konsole und läßt ein login-Fenster erscheinen (kein Anmelden an der Textkonsole erforderlich).
- o Konfigurationsdateien des »xdm« im Verzeichnis `/etc/X11/xdm/`:

Beschreibung	Datei
Namen der Konfigurationsdateien	<code>`xdm-config'</code>
Aussehen und Eigenschaften des login-Fensters	<code>`Xresources'</code>
Selbst zu verwaltende X-Sitzungen	<code>`Xservers'</code>
Welche Maschinen dürfen mit der Option <code>'-query'</code> auf uns zugreifen?	<code>`Xaccess'</code>

- o Folgende shell-Skripte verwalten die X-Sitzung

Beschreibung	Datei
Zubehör zum login-Fenster starten	<code>`Xsetup'</code>
Nach dem Einloggen des Benutzers	<code>`Xstartup', `GiveConsole'</code>
Start der eigentlichen Sitzung	<code>`../Xsession'</code>
Nach Beenden der Sitzung	<code>`Xreset', `TakeConsole'</code>

— Konfiguration von X11-Anwendungen

X11-Anwendungen lassen sich konfigurieren über

- (1) die Kommandozeile
- (2) den per »xrdb« geladenen Einstellungen ("X-Ressourcen")
- (3) Umgebungsvariablen
- (4) Konfigurationsdateien im Verzeichnis `/usr/lib/X11/app-defaults'`

Die Priorität der Einstellungen entspricht obiger Nummerierung.

Die per »xrdb« geladenen Einstellungen werden (nur) für die aktuelle Sitzung in einer X11-internen Datenbank gespeichert. Programme können diese Datenbank auslesen und die Einstellungen übernehmen.

Die Standardkonfigurationen der X11-Programme findet man im Verzeichnis `/usr/lib/X11/app-defaults'`; diese sollten nicht geändert werden, sondern nur als Beispiel dienen. Stattdessen kann man die Einstellungen mit Einträgen in den Dateien

`~/Xresources'` oder `~/Xdefaults'` (privat)
`/etc/X11/Xresources'` (global)

überschreiben. Die Dateien werden mit dem Befehl »xrdb« in die X11-interne Xresources-Datenbank geladen.

Bsp.: »xrdb -merge ~/.Xresources«

Beispiele:

```
-----8<----- hier schneiden -----8<-----
*Background: lightgrey
xwpe.font: 10x20
Mosaic*HomeDocument: http://www.rrz.uni-koeln.de/
Emacs*cursorColor: red
XDvi.paper: a4
XDvi*geometry: 800x720
XTerm*ttyModes: intr ^c quit ^\\ kill ^u erase ^?
XTerm.VT100.Translations: #override \n\
<Key>Home: string(0x01) \n\
<Key>End: string(0x05) \n\
<Key>Delete: string(0x04) \n\
<Key>BackSpace: string(0x7f) \n\
Ctrl<Key>Left: string(0x1b) string("b")\n\
Ctrl<Key>Right: string(0x1b) string("f")\n
-----8<----- hier schneiden -----8<-----_
```

—
Drucken

48.1 Ansteuerung der Hardware

- o Der Betriebssystemkern stellt lediglich Gerätedateien zur Verfügung, über die direkt mit dem Drucker kommuniziert werden kann.
Er hat NICHT die Funktion eines Druckmanagers zum Verwalten und aufbereiten der Daten.
Bsp.: »cat /proc/cpuinfo > /dev/lp1«
- o Herausfinden des "richtigen" Gerätedatei:
Bsp.: »for i in /dev/lp*; do tunelp \$i; done«
- o Konfiguration der Gerätedateien: »tunelp« (tune-lineprinter)
- o I.d.R. sind Drucker vom Werk aus auf den Zeichensatz von DOS eingestellt (siehe 8.3). Entweder (falls möglich) legt man einen entsprechenden Mini-Schalter am Drucker um, oder man muß für die Konvertierung von Texten mit »unix2dos« sorgen (im Druckerfilter ergänzen).
- o Der Betriebssystemkern muß den Treiber für die parallele Schnittstelle enthalten ("Parallel Printer Support").
Bsp.: »insmod lp«

48.2 Druckmanager

- o Druckmanager übernehmen die folgenden Funktionen
 - Einrichtung von symbolischen Namen für die einzelnen Drucker
 - Entgegennahme der zu druckenden Daten (siehe unten)
 - Zwischenspeicherung der Daten bis zur Bereitschaft des Druckers (man denke an mehrere Druckaufträge)
 - Aufruf des "Druckertreibers" (ein Konvertierungsprogramm für die Pixelsprache des jeweiligen Druckers)
 - Übermittlung der Daten an den Drucker

- o Die Konfiguration und die Vergabe von symbolischen Namen wird in der Datei `/etc/printcap` vorgenommen.

```
-----8<----- hier schneiden -----8<-----
octopuss|lpt1|lp|PostScript|DeskJet 3050:\
      :lp=/dev/lp1:\
      :sd=/var/spool/lpd/octopuss:\
      :if=/usr/local/bin/deskjet-treiber:\
      :mx#0:sh:
-----8<----- hier schneiden -----8<-----
```

In diesem Beispiel erhält der Drucker an `/dev/lpt1` die Namen "octopuss", "lpt1" und "lp". Die Druckaufträge für ihn werden im Verzeichnis `/var/spool/lpd/octopuss` zwischengespeichert (engl. "spoolen"). Das Programm `/usr/local/bin/deskjet-treiber` setzt die Daten in die Pixelsprache des Druckers um.

- o Zur Realisierung eines Druckmanagers stehen die Pakete »Berkeley lpd« und »LPRng« (etwas flexiblerer/komplexer) zur Verfügung.
- o Der zugehörige Daemon trägt den Namen »lpd« und benötigt ein Netzwerk (das »loopback« reicht bereits).
- o Steuerung mit »lpc« ("line printer control").

48.3 Druckerfilter und Druckertreiber

- o Ein Druckerfilter ist ein Programm, das vom Druckmanager zur Aufbereitung der zu druckenden Daten aufgerufen wird. Mein persönlicher Favorit ist das Programmpaket "magicfilter" (da einfach zu installieren, transparent und flexibel).
- o Druckerfilter erkennen den Typ der Daten anhand charakteristischer Zeichen an bestimmten Positionen (z.B. "GIF" zu Beginn der Datei bei GIF-Bildern):

- Reiner Text und Daten in der Pixelsprache des Druckers werden nicht weiter verändert und an den Drucker geschickt.
- Sinnvoll druckbare Daten werden mithilfe verschiedener Programme (z.B. "gunzip", "pbmplus-utilities", "dvips", usw.) in das postscript-Format überführt.
- Alle anderen Daten werden als "nicht druckbar" zurückgewiesen.

- o Das Programm »gs« ("ghostscript") enthält Druckertreiber für die Umsetzung von postscript in das Pixelformat vieler handelsüblicher Drucker. (Nur notwendig, wenn man keinen postscript-fähigen Drucker besitzt.)

- o Liste der eincompilierten Druckertreiber: `»gs -h | more«`

- o Ein Druckerfilter führt genau die Befehle aus, die man auch per Hand auf der Kommandozeile geben würde, um eine Datei in die Pixelsprache des eigenen Druckers zu konvertieren. Beispiel:

```
-----8<----- hier schneiden -----8<-----
zcat ckl-3.5.ps.gz | gs -q -dSAFER -dNOPAUSE -sDEVICE=djet500\
-sOutputFile=- - > /dev/lp1
-----8<----- hier schneiden -----8<-----
```

In diesem Beispiel wird die Datei `ckl-3.4.ps.gz` zunächst decomprimiert (»zcat«) und dann für einen Deskjet-500 Drucker aufbereitet.

48.4 Liste nützlicher Konvertierungsprogramme

[Kommt alles, nur Geduld.]

48.5 Druckaufträge aus der Sicht des Benutzers

Aktion	Prg.	Beispiel
Datei drucken	»lpr«	»lpr -h ckl-3.5.ps«
Warteschlange zeigen	»lpq«	»lpq -l «
Aufträge zurückziehen	»lprm«	»lprm -«

- o Die Auswahl des Druckers geschieht nach folgenden Regeln:
 - Der erste Drucker in `"/etc/printcap"` ist der Standard-Drucker
 - Die Umgebungsvariable `"PRINTER"` kann einen anderen Drucker als Voreinstellung angeben.
 - Mit der Option `'-P'` überschreibt man die ersten beiden Einstellungen, bezieht sich aber nur auf den aktuellen Befehl.
Bsp.: `»lpr -Pkrake ckl-3.5.ps«`
- o Ist in der Konfigurationsdatei `"/etc/printcap"` eine Grenze für die maximale Größe eines Druckauftrags angegeben (`"mx#1000"` o.ä.), so müssen grössere Dateien mit der Option `'-s'` gedruckt werden. In diesem Fall wird im spool-Verzeichnis lediglich ein symbolischer link angelegt, d.h. die Datei darf bis zur Erledigung des Drucks nicht angefaßt werden.
Bsp.: `»lpr -s ckl-3.5.ps«`
- o `»mpage«`, `»atp«`, `»gslp«`, `»a2ps«`, `»x2ps«`, `»genscript«` sparen Papier, indem sie 2 (oder mehr) Seiten Text verkleinert auf eine Seite drucken (im postscript-Format).

48.7 Zugriffsrechte

- o Beschränkung der Druckerbenutzung auf eine Benutzergruppe:
`":rg=studenten:\"` (beschränkt Zugriff auf "studenten").
- o Freigabe der Drucker für andere Maschinen im Netz: `"/etc/hosts.lpd"` und `"/etc/hosts.equiv"`
- o Platzmangel, ab dem keine Druckaufträge mehr entgegengenommen werden: `"/var/spool/lpd/octopus/minfree"`

48.6 Drucker im Netz

- o Um Drucker anderer Linux-Rechner im Netz anzusprechen, läßt man das Feld für die Gerätedatei in der Datei `"/etc/printcap"` leer und trägt zusätzlich ein:

```
-----8<----- hier schneiden -----8<-----
      :lp=:\
      :rm=marsupilami:\
      :rp=gutenberg:\
-----8<----- hier schneiden -----8<-----
```

Wobei "marsupilami" der Name des anderen Rechners und "gutenberg" der Name des Druckers am anderen Rechner ist._

Linux als Print- und Fileserver

Es wird Zeit

50.1 Die Uhrzeit

- o Unter Linux gibt es "drei verschiedene Uhrzeiten":
 - die der auf der PC-Hauptplatine installierten CMOS-Uhr,
 - die Linux-interne Uhrzeit (Systemzeit) und
 - die von verschiedenen »Uhren« angezeigte Zeit

	»clock -u -w«		»date -s«	
CMOS-Uhr (Mainboard) (GMT)	<----->	Linux Systemzeit (GMT)	<----->	Benutzer-Uhr (MET)
	»clock -u -s«		»date«, »xclock«	

- o Die Option '-u' des clock-Befehls bewirkt eine unveränderte Übernahme der CMOS-Zeit als Systemzeit (üblicherweise während des Bootvorgangs).

50.1 Zeitzonen

- o Die ganze Welt ist in Zeitzonen eingeteilt; der "Nullpunkt" ist willkürlich auf "Greenwich" (England) festgelegt worden.
- o "Greenwich Mean Time" (GMT) ist die sogenannte Normalzeit, aus der sich alle anderen Uhrzeiten berechnen lassen.
- o Die Regeln zur Umrechnung befinden sich unter Linux im Verzeichnis `/usr/lib/zoneinfo`.
- o Zu den Regeln gehört z.B. die automatische Umstellung von Sommer- auf Winterzeit.

- o Die Datei `/usr/lib/zoneinfo/localtime` legt die Zeitzone eines Linux-Systems fest; Programme wie »date« und »xclock« berechnen daraus die Uhrzeit. Oftmals ist diese Datei ein link.

- o Die Umgebungsvariable "TZ" kann von den Benutzern zur Wahl einer anderen Zeitzone benutzt werden.
Bsp.:
»export TZ=Asia/Tokyo; xclock -title Tokio &&
»export TZ=America/New_York; xclock -title "NY" &&

(Die Beispiele setzen voraus, daß der Windowmanager die Titel der xclock-Fenster nicht unterdrückt!)

- o Ist Linux das einzige Betriebssystem auf dem Rechner, so setzt man die Systemzeit auf »GMT« (Greenwich Mean Time) und legt einen Verweis von `localtime` auf `MET` an.
- o Ansonsten genau umgekehrt (d.h. Regeln werden nicht beachtet).

WARNUNG: vor dem Setzen der Zeitzone alle anderen Arbeiten beenden, d.h. Dateien schliessen und alle Programme beenden! Ansonsten können Dateien mit »kaputten« Zeitmarken versehen werden und ziemlich üble, »nicht erklärbare« Fehler verursachen. _

50.3 Synchronisation der Uhrzeit

- o Zum einmaligen Abgleich der Uhrzeit mit anderen Maschinen im Netz kann »netdate« benutzt werden.

Bsp.: »netdate trick tick track«

- o Um die Uhrzeit fortlaufend mit allen anderen Maschinen im Netz abzugleichen dient der Daeomon »xntp«.
Dabei kommt es weniger auf die exakte Uhrzeit, als auf eine möglichst einheitliche Zeit aller Maschinen an.

50.4 "cron"

- o Der daemon »cron« startet Programme immer wiederkehrend in regelmäßigen Zeitabständen.
- o Jeder Benutzer kann mit dem Programm »crontab« eine individuelle Tabelle mit cron-Anweisungen erstellen

Bsp.: »crontab -e«

```
-----8<----- hier schneiden -----8<-----
# 5:30 Uhr das Modem zurücksetzen
30 5 * * * modem_reset

# Erinnerung: jeden Mittwoch den Drucker aussaugen
0 8 * * Wed mail truemper < Drucker_saugen

# | | | | |
# | | | | +- Wochentag 0-7 (0 oder 7 =Sonntag)
# | | | +---- Monat 1-12
# | | +----- Tag 1-31
# | +----- Stunde 0-23
# +----- Minuten 0-59

-----8<----- hier schneiden -----8<-----
```

- o Dokumentation der Syntax in »man 5 crontab«
- o Verursachen die durch »cron« gestarteten Programme eine Ausgabe, so wird diese als mail an den Benutzer geschickt.

50.5 Die Programme »at« und »batch«

- o »at« führt Programme einmalig zu einem bestimmten Zeitpunkt aus.
Bsp.: »at -f ftp_get_file.sh 7am Mar 10 «

In diesem Beispiel werden die Kommandos aus der Datei
`ftp_get_file.sh' am 10. März um 7:00 Uhr ausgeführt.

- o Die Liste aller Aufträge wird mit »atq« angezeigt; einzelne Aufträge können mit »atrm« gelöscht werden.
- o »batch« führt Programme aus, sobald der Rechner nicht ausgelastet ist.

Deutsche Anpassungen

—

—

Linux

52.1 Der Betriebssystemkern

- o Zur Aufgabe des Betriebssystemkerns siehe 3.1.
- o Der Betriebssystemkern wird als »kernel« bezeichnet.
- o Bezugsquellen für die neusten kernel-Quelltexte unter 12.3.

- o Die verschiedenen Versionen des Betriebssystemkerns werden in aufsteigender Reihenfolge nummeriert.
- o Kernelversionen unter "2.1" erzeugen i.d.R. schwere Minderwertigkeitsgefühle.

52.2 Begriffe

"kernel-panic": Notbremse des Betriebssystemkerns zur Verhinderung von Folgefehlern. Hält das Linux-System sofort und ohne Vorwarnung an. Selten.

"Code-Freeze": Ankündigung von Linus, in die BETA-Phase zu gehen, um den diesjährigen "stabilen" Kernel herauszugeben. Ruft allgemeine Torschlußpanik hervor, um noch möglichst viele neue Funktionen einzubauen.

52.3 Treiber und Module

- o "Treiber" verleihen dem kernel Fähigkeiten wie "Lesen von DOS-Dateisystemen" oder "Ansteuern von ATAPI-CDROMs".

Befehl	Angezeigte Übersicht
»cat /proc/devices«	Geräte (block- und zeichenorientiert)
»cat /proc/filesystems«	Dateisysteme
»cat /proc/net/dev«	Netzwerkschnittstellen

- o "Module" sind Treiber, die im laufenden Betrieb in den kernel eingebunden werden können.
- o Bei der Erzeugung eines neuen kernels kann man wählen, ob ein Treiber fester Bestandteil des kernels sein soll, oder als Modul zur Verfügung stehen soll.

Programm	Funktion	Beispiel
»lsmod«	Liste aller geladenen Module anzeigen	»lsmod«
»insmod«	Modul in den kernel laden	»insmod lp«
»rmmod«	Modul aus kernel entfernen	»rmmod lp«

- o `/etc/modules` listet alle Module auf, die beim booten geladen werden sollen.
- o Automatisches Laden von Modulen "bei Bedarf": »kerneld«

52.4 Konfiguration und Erzeugung eines Betriebssystemkerns

- o Voraussetzung für die Erzeugung eines lauffähigen Betriebssystemkerns sind
 - der C-Compiler gcc plus Zubehör und
 - die Quelltexte zum Betriebssystem (z.B. im Verzeichnis `/usr/src`).
- o Das Auspacken unterscheidet sich nicht von dem anderer "Software", siehe 13.1.
 - Bsp.: »cd /usr/src; mv linux linux-old; \
 - tar -xzvf linux-1.3.77.tar.gz«
- o Auch die "Updates" lassen sich wie gewohnt einspielen, siehe 13.2.
 - Bsp.: »cd /usr/src; for i in patches/*; do \

```
zcat $i | patch -p0«
```

- o Die Konfiguration wird im Verzeichnis ``/usr/src/linux'` mit den Befehlen
- o Jede Einstellung kann modifiziert werden und ist genau beschrieben.
- o Nach der Konfiguration wird der kernel mit der Befehlsfolge

```
»make dep && make clean && make zImage«
```

erzeugt. Er steht danach als Datei ``zImage'` im Verzeichnis ``/usr/src/Linux/arch/i386/boot'` zur Verfügung.

- o Die Beschreibung der Installation dieser Datei folgt.
- o Gibt man abschliessend den Befehl

```
»make modules; make modules_install«,
```

so werden die Module erzeugt und im Verzeichnis ``/lib/modules'` installiert (und werden dort von den modules-Programmen gefunden).

Hier ein Wort von unserem großen Meister:

1.3.x kernels are development kernels, but the more people that use them, the merrier. People should be aware of the fact that it's a development kernel, but they shouldn't be discouraged from using them as long as they know what it means:

- if you see a bug, report it. To me, to the mailing lists, and possibly even to the newsgroups. If you're running an old kernel, people will usually tell you to upgrade to the latest kernel and then try it again, so you might want to do that before being told.
- if you get bitten by a bug, tough luck. I'll do my very best to get the bug fixed, but the one thing I won't do is feel sorry for you. In fact, I might ask you to do it all over again, just to get more information. I'm a heartless bastard.

So do not try to discourage people from trying development kernels: just tell them what that implies. Lots of people will then be happier to use the stable kernels, but if everybody is chicken, we won't get any bugreports or fixes._