

Benutzerebene eines Rechners

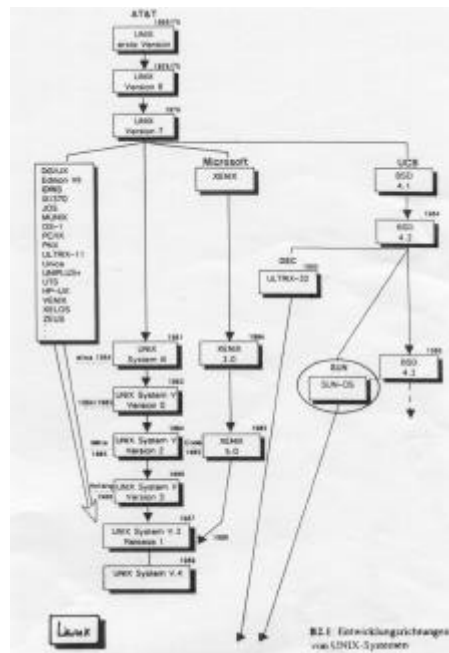
1. **Betriebssystem UNIX**
2. **Internet**
3. **Texteditor VI**

Das Betriebssystem UNIX

- Aufgaben eines Betriebssystems
generell: Verwaltung und Zuteilung der Ressourcen eines Rechners

wesentliche Aufgaben:

- Kontrolle und Verwaltung der Betriebsmittel
- Bereitstellung einer logischen Sicht auf die Betriebsmittel
- Zuteilung der Betriebsmittel
- Ausführen von Benutzerprogrammen
- Ablaufkontrolle aller Prozesse



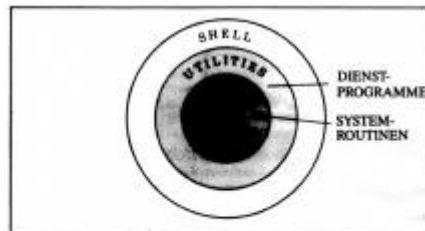
(aus J.Gulbins 1988)

Kennzeichen von UNIX

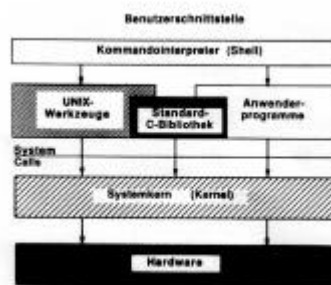
- Mehrbenutzer-Betriebssystem (multi-user)
- Time-sharing Betriebssystem (Verwaltung mehrerer asynchroner Prozesse)
- Geräteunabhängiges, hierarchisches Dateikonzept
- Gute Portabilität
- Dialogorientierung (Kommandointerpreter)
- Fließband-Technik (pipelining)

Strukturierung von UNIX

- **Schalen-/Schichtenkonzept**
 - Hardware
 - Betriebssystem-Kern (*kernel*)
(Prozessverwaltung und -kommunikation, Speicherverwaltung, Ein-/Ausgabe-Verwaltung, Gerätetreiber, ...)
 - Dienst- und Anwenderprogramme
(Bibliothek mit Standardfunktionen (in C), Compiler, Editoren, ...)
 - Kommandointerpreter (*shell*)



Struktur eines Computersystems



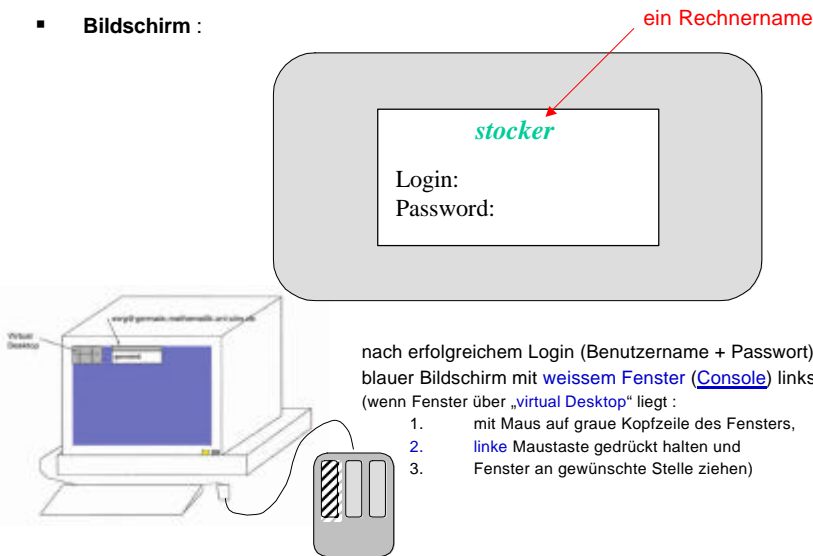
Systemstruktur von UNIX

(aus P.Domann et al.1988)

Rechnersitzungen -- An- und Abmeldung

Anmelden (Login)

- Bildschirm :



nach erfolgreichem Login (Benutzername + Passwort) ...
blauer Bildschirm mit **weissem Fenster (Console)** links oben
(wenn Fenster über „virtual Desktop“ liegt :

1. mit Maus auf graue Kopfzeile des Fensters,
2. linke Maustaste gedrückt halten und
3. Fenster an gewünschte Stelle ziehen)

Arbeitsfenster

Öffnen eines Arbeitsfensters



1. mit Maus auf blauen Hintergrund-Bereich,
 2. rechte Maustaste kurz drücken oder gedrückt halten
- ➔ Auswählenü **Workspace**
Auswahl **Hosts**
 - ➔ Auswählenü **Hosts**
Auswahl **Xterm**
 - ➔ Auswählenü **Xterm**
Auswahl **thales** (Bsp. eines lokalen Rechners)



sog. „prompt“

Abmelden (Logout)

Abmelden (Logout)

▪ **Bildschirm :**

1. mit Maus auf blauen Hintergrund-Bereich,
 2. **rechte** Maustaste kurz drücken oder gedrückt halten
- Auswahlmnü **Workspace**
Auswahl **Exit**



Online-Dokumentation

- **man** <kommandoname>

man = manual

Auflistung der zu einem Kommando gespeicherten Informationen und Optionen (Namen, Beschreibungen, Parameter, etc)

Ausgabe: bereitet die Ausgabe zum seitenweisen blättern vor.

verlassen mit **q** (quit).

Dateien und Dateisysteme

- Dateikonzept

Datei: sequentielle Folge von Datenworten (Bytes)

Dateiarten: 1) normale Dateien (*files*; Texte, Programme, Daten,...)

2) Dateiverzeichnisse (*directories*; auch Kataloge)

3) Gerätedateien (*special files*, für physik. Geräte)

Ansprechen (Addressieren) einer Datei/Verzeichnis

- ihren direkten Namen (im enthaltenden Verzeichnis)
- ihren absoluten Namen, d.h. (vollständigen) Pfadnamen
- relativ zur Position an der man sich im Dateibaum gerade befindet

Dateien (Files)

- Zusammenfassung einer Folge von Zeichen (Bytes) zu einer Einheit
- Eine Datei wird permanent auf der Festplatte gespeichert.
- Eine Datei hat einen Namen und einen Besitzer.
- Dateinamen sollen sprechend sein (Vorsicht bei Sonderzeichen!!!)
- Fast alles - auch Drucker, Bildschirm, etc - ist eine Datei.
- Mit einer Datei sind weitere Informationen verbunden.

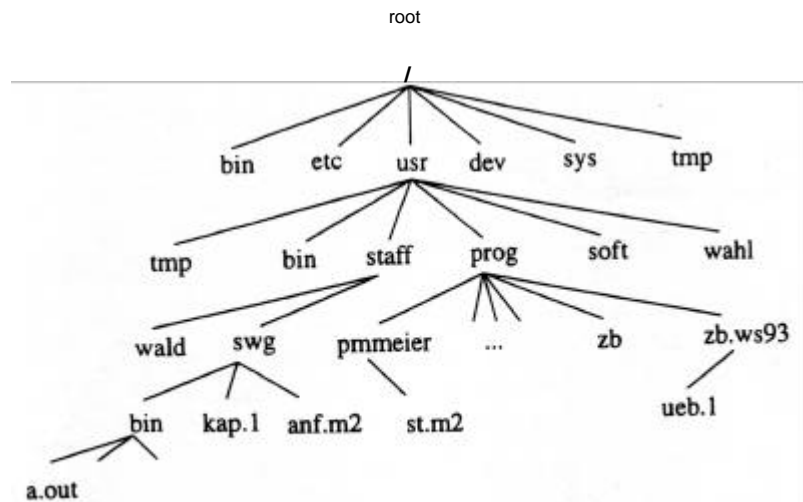
Kataloge / Directories

- Zusammenfassung von Dateien (und Katalogen) zu einer Einheit.
- Ein Katalog ist selbst eine Datei, deren Inhalt i.w. die Namen der enthaltenden Dateien (und Kataloge) ist.
- Kataloge dienen ähnlich wie Ordner zur Zusammenfassung von „gleichartigen“ Dateien.
- Das UNIX-Dateisystem besteht (von außen betrachtet) aus hierarchisch angeordneten Katalogen.
- Der „oberste Katalog“ (sog. **root**) hat einen festen Namen: / (Schrägstrich)

Verzeichnisse

- Heimat-Verzeichnis (home directory)
automatisch zugewiesenes Verzeichnis (Ordner), auf dem man sich nach dem Anmelden befindet.
- Spezielle Namen
 - . aktuelles Verzeichnis auf dem man gerade steht.
 - .. Vorgänger-/Elternverzeichnis (ist eindeutig!)
 - <name>/ Nachfolger-/Kindverzeichnis (durch <name> eindeutig)

Verzeichnisbaum



Verzeichnisbaum-Kommandos

pwd	Name/Pfad des aktuellen Verzeichnisses (print working directory)
cd	Ändern des Arbeitsverzeichnisses (change directory)
cd .	bleibt im aktuellen Verzeichnis.
cd ..	zum Elternverzeichnis des aktuellen Verzeichnisses.
cd <name>/	zum angegebenen Unterverzeichnis, ausgehend vom aktuellen Verzeichnis.
mkdir <name>	anlegen eines neuen Unterverzeichnisses
rmdir <name>	löschen eines Unterverzeichnisses (das angesprochene Verzeichnis muß leer sein)

Dateiausgabe (Format: command <fn1> <fn2>..)

- cat** Ausgabe oder Konkatenation von Dateien auf Monitor.
- more** seitenweise (formatierte) Ausgabe von Dateien auf Monitor.
- head, tail** Ausgabe der ersten/letzten Zeilen von Dateien auf Monitor.
- grep** Suchen einer angeg. Zeichenfolge in einer Datei.
Format: grep "<such-string>" <fn1> <fn2>
- lp** Ausgabe von Dateien auf dem Drucker
Datei muß im **Postscript** Format sein
- a2ps** Ausgabe einer Text-Datei auf dem Drucker
Text wird hierzu konvertiert.

ASCII-Zeichen

D	O	H	Z	D	O	H	Z	D	O	H	Z
0	000	00	nul	27	033	1B	esc	54	066	36	6
1	001	01	soh	28	034	1C	fs	55	067	37	7
2	002	02	stx	29	035	1D	gs	56	070	38	8
3	003	03	etx	30	036	1E	rs	57	071	39	9
4	004	04	eot	31	037	1F	us	58	072	3A	:
5	005	05	enq	32	040	20		59	073	3B	;
6	006	06	ack	33	041	21	!	60	074	3C	<
7	007	07	bel	34	042	22	"	61	075	3D	=
8	010	08	bs	35	043	23	#	62	076	3E	>
9	011	09	ht	36	044	24	\$	63	077	3F	?
10	012	0A	lf	37	045	25	%	64	100	40	@
11	013	0B	vt	38	046	26	&	65	101	41	A
12	014	0C	ff	39	047	27	'	66	102	42	B
13	015	0D	cr	40	050	28	(67	103	43	C
14	016	0E	so	41	051	29)	68	104	44	D
15	017	0F	si	42	052	2A	*	69	105	45	E
16	020	10	dle	41	053	2B	+	70	106	46	F
17	021	11	dc1	44	054	2C	,	71	107	47	G
18	022	12	dc2	45	055	2D	-	72	110	48	H
19	023	13	dc3	46	056	2E	.	73	111	49	I
20	024	14	dc4	47	057	2F	/	74	112	4A	J
21	025	15	nak	48	060	30	0	75	113	4B	K
22	026	16	syn	49	061	31	1	76	114	4C	L
23	027	17	etb	50	062	32	2	77	115	4D	M
24	030	18	can	51	063	33	3	78	116	4E	N
25	031	19	em	52	064	34	4	79	117	4F	O
26	032	1A	sub	53	065	35	5	80	120	50	P

Informationen über Dateien

ls Inhalt (*listing*) eines Verzeichnisses

Optionen (siehe **man ls**):

- a alles anzeigen auch versteckte Dateien
- l lang, d.h. detaillierte Dateiinformatioenen
- x spaltige Anzeige
- F Name von Verzeichnissen (d) werden in Form <name>/ angezeigt
- s Größe der einzelnen Dateien in Blöcken à 512 Byte

Ändern von Dateien und Dateiattributen I

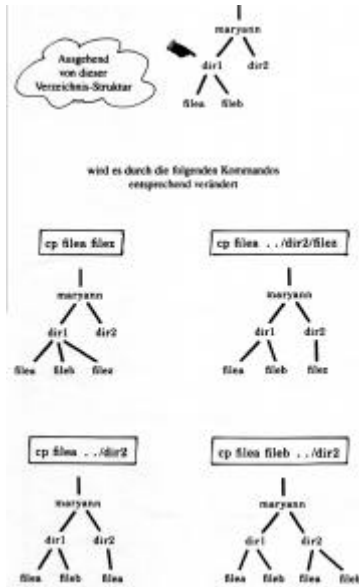
chmod Änderung der Zugriffsrechte (Mode) einer Datei.
(siehe auch **man chmod**)

cp Kopieren von Dateien
Format: cp <source1> ... <destination>

- Existiert Datei <destination> nicht, so wird diese angelegt.
- Existiert <destination> bereits, so wird deren Inhalt überschreiben

rm Löschen von Dateien.
Format: rm <fn1> <fn2> ...
VORSICHT! : Was weg ist, ist weg!

Beispiel: für cp-Befehl



Ändern von Dateien und Dateiattributen II

mv

Änderung des Namens einer Datei

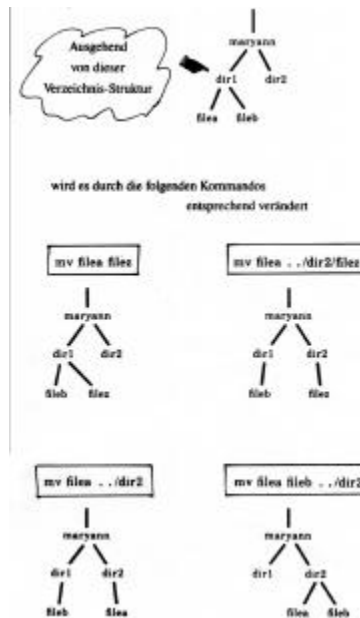
Format: `mv <source> <destination>`

`mv <source1> <source2> ... <destination>`

- Umbenennen einer Datei oder eines Verzeichnisses
- `<destination>` = Verzeichnis, dann werden `<source1> <source2> ...` nach `<destination>` verschoben

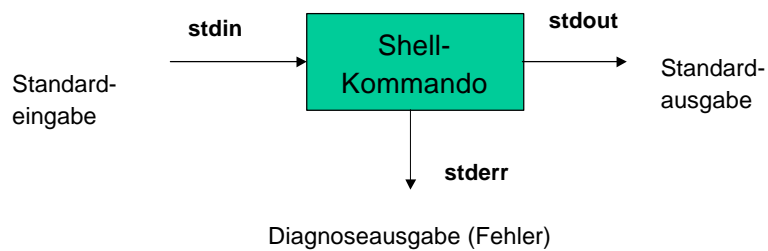
Der **mv**-Befehl wird häufig zum Umhängen/Verschieben von Dateien und Verzeichnissen im Dateibaum verwendet.

Beispiel: für mv-Befehl



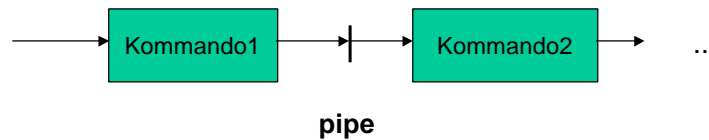
Kommandointerpreterebene (Shell)

- allgemein:
 - Kommandosprache in einem Interpreter
 - (einfache) Programmiersprache



stdin : Eingabe von Tastatur
stdout, stderr : Ausgabe auf Bildschirm

Pipelines: Konkatenation von Kommandos



| Umlenkung der Standardausgabe von Kommando 1
in die Standardeingabe von Kommando 2
....

Beispiel:

ls|sort -r absteigend sortierte Liste der Dateien
im aktuellen Verzeichnis

Umlenken der Standardein-/ausgabe

- < Umlenken der Standardeingabe aus einer angegebenen Datei.
Bsp.: **wc < <f1>**
zählt Zeilen, Worte und Zeichen der Text-datei
- > Umlenken der Standardausgabe in eine angegebene Datei.
existiert die Datei nicht, wird diese angelegt.
existiert die Datei, so wird deren Inhalt überschrieben!
- >> hängt neuen Inhalt am Dateiende an!
Bsp.: **ls -l > liste**
ls -l >> liste
- 2> Umlenken der Diagnoseausgabe
Bsp: **wc -x <f1> 2> error.doc** (-x als Option ungültig!)

Prozesse

- Mit jedem eingegebenen Kommando wird ein Prozeß gestartet; während der Ausführung bleibt die Standardeingabe blockiert!
- Starten von **Hintergrund**prozessen
 - am Ende der Kommandozeile: **&** (kaufm. Und-Zeichen).
Anzeige der Prozeßnummer (PID, process-id)
 - während der Laufzeit des Prozesses kann man im Vordergrund weiterarbeiten (= Standardeingabe ist offen)
- Welcher Prozeß arbeitet gerade?
ps process status
Ausgabe: PID TTY TIME COMMAND
87 console 0:02 sh
101 console 0:00 ps
- Abbrechen eines Prozesses
 - 1) Prozeß im Vordergrund: ctrl-C
 - 2) Prozeß im Hintergrund: kill -9 <PID>

Sonderzeichen der Shell

- Kommandozeile ...
 - wird zunächst auf bestimmte Zeichen(kombinationen) untersucht.
 - Kommando wird ausgeführt und mit Argumenten(=Optionen) versorgt.
- Dateinamen
 - * (leere) Zeichenfolge (0 bis n Zeichen) als Platzhalter gültiger Namensfragmente
 - ? Genau ein Zeichen als Platzhalter
- Quotierungen: ` ` , " " , ` ` , \
- Variable
 - \$ anschließende Zeichenkette wird als Variable betrachtet und hierdurch ersetzt.
Bsp.: **\$HOME** – HOME enthält den Pfadnamen des Heimatkatalogs.

Mail-Kommunikation

- Einsatz: Verschicken von Nachrichten an verschiedene Benutzer des (UNIX)-Systems.
- verschickte Nachrichten landen in einem Briefkasten. (MAIL-BOX)
- Der Inhalt ist vom jeweiligen Adressaten abrufbar.
- Das Vorhandensein von (neuen) Nachrichten wird nach dem Anmelden (login) angezeigt.
You have mail. oder **You have new mail.**

Verschicken einer E-Mail

- Direkte Erstellung einer Nachricht.
...% **mail** <login-name des Adressaten>
 subject: <stichwort>
 .
 . direkt eingegebener Text
 .
 ctrl-D oder .
...%
- vorhandene Nachricht
...% **mail** <login-name des Adressaten> < message

Lesen, Speicher, Antworten

- ...% mail

.
. Ausgabe der letzten eingegangenen Nachrichten

? Kommando Prompt-Zeichen von mail

Kommandos:

? Liste der möglichen Kommandos

s speichern der vorhandenen Nachrichten

M<user> mail an Adressaten

Beenden von mail

- q, ctrl-D

zu löschende Nachrichten werden entfernt, die anderen abgelegt;
Rückkehr zur **Shell**

- X

direkte Rückkehr zur **Shell**; ohne den Status zu ändern

mail-Kommandos I

d [message list]	delete messages
e [message list]	edit messages
f [message list]	show from lines messages
h	print out active message headers
m [message list]	mail to specific users
n	goto and type next message
pre [message list]	make messages go back to system mailbox
q	quit, saving unresolved messages in mailbox
r [message list]	reply to sender
R [message list]	reply to sender and all recipients
t [message list]	type messages

mail-Kommandos II

top [message list]	show top lines of messages
u [message list]	undelete messages
v [message list]	edit messages with display editor
w [message list] file	append messages to file
x	quit, do not change system mailbox
z [-]	display next [previous] page of headers
!	shell escape

pine - ein Mail-Client

- ASCII-orientiertes Programm zum lesen und schreiben von Mails (Alternative: **mutt**)
- wichtigste Menüpunkte:
 - Compose Message (neue Mail schreiben)
 - Folder List (zeigt Mailverzeichnisse)
 - => INBOX (hier sind die neuen Mails drin)
 - => sent-mail (die Mails die man selber verschickt hat)
 - => ...
- im unteren Teil des Fensters sieht man welche Tasten für welchen Befehl gedrückt werden müssen (^ bedeutet die **ctrl**-Taste)
- wichtige Befehle:
 - m** zum Hauptmenue
 - q** Beenden (mit **y(es)** bestätigen)
 - Control + X** versenden der Mail (in **Compose Message**)
- **nbiff** zeigt an ob neue Mails gekommen sind.
mit **Open** wird **pine** geöffnet (Hinweis: Man befindet sich dann gleich in der INBOX, nicht im Hauptmenue!)

Hauptmenü von pine

```
ssh thales
PINE 4.21L2  MAIN MENU  Folder: INBOX  No Messages

?  HELP          -  Get help using Pine
C  COMPOSE MESSAGE -  Compose and send a message
I  MESSAGE INDEX -  View messages in current folder
L  FOLDER LIST   -  Select a folder to view
A  ADDRESS BOOK  -  Update address book
S  SETUP         -  Configure Pine Options
Q  QUIT         -  Leave the Pine program

Copyright 1989-1999.  PINE is a trademark of the University of Washington.
[Folder: "INBOX" opened with 0 messages]
? Help
OTHER CMDS [ListFldrs] PrevCmd ReINotes
NextCmd KBlock
```

Mail-Verzeichnisse

```
ssh thales
PINE 4.21L2  FOLDER LIST                               Folder: INBOX  No Messages
INBOX

[ALL of folder list]
? Help      Main Menu  PrevFldr  PrevPage  Add      Rename
OTHER CMDS [View Fldr] NextFldr  Spc      NextPage  Delete   WhereIs
```

INBOX

```
ssh thales
PINE 3.95  FOLDER INDEX                               Folder: INBOX  Message 2 of 2 NEW
+ 1 Oct 24 To: schwedi@mathem (351) Tutorien
+N 2 Oct 24 To: schwedi@mathem (351) Tutorien

? Help      Main Menu  PrevMsg  PrevPage  Delete   Reply
OTHER CMDS [ViewMsg] NextMsg  Spc      NextPage  Undelete Forward
```

Compose Message



nbiff

Start



neue mails



Nützliche Kommandos

cat	Dateinhalt ausgeben
cd	Katalog wechseln
cp	Datei kopieren
cut	aus einer Datei mit Zeilenstruktur Teile "herausschneiden"
date	Datum und Uhrzeit ausgeben
echo	Argumente ausgeben
egrep	wie grep, nur mächtiger (reguläre Ausdrücke)
fgrep	nach fest vorgegebener Zeichenfolge suchen, schnell
find	Nach Dateien in einer Katalog-Hierarchie suchen
grep	Zeichenfolge suchen
head	Dateianfang ausgeben
ls	Kataloginhalt ausgeben
man	INFORMATIONEN ZU EINEM KOMMANDO
mail	e-mail versenden
mailx	e-mail ansehen u. bearbeiten (mit help)
mkdir	Katalog anlegen
more	ähnlich zu pg, nur komfortabler
mv	Datei umbenennen
paste	zwei Dateien zeilenweise zusammenfügen
pg	Seitenweise ausgeben
rm	Datei entfernen (!!!)
rmdir	leeren Katalog entfernen
sort	Datei sortieren
tail	Dateiende ausgeben
tr	in einer Datei durchgehend ein Zeichen durch ein anderes ersetzen
wc	Zeilen, Worte und Zeichen zählen
who	Liste der angemeldeten Benutzer ausgeben

Internet

Ausgangspunkt: Verteilte Kommunikations-Systeme (Netze)

ARPANET (Advanced research Projects Agency)

Prinzip:

- Netze von Rechnern (Vermittlungsrechner (Router), Zielrechner)
- Vermittlung und Betrieb (Paketvermittlung)
 - Nachricht von A nach B
 - Zerlegung in Pakete mit Zieladresse B
 - Versendung der Pakete über möglicherweise verschiedene Verbindungen

Prinzip und Organisation

- **Protokolle**
 - Festlegung von Konventionen und Regeln über den Austausch von Informationen zw. Kommunikationspartnern (hier Computern)
- **Ziel:** vollständige, fehlerfreie und möglichst effektive Datenübertragung
 - TCP/IP heute ein Standard
 - TCP : Transmission Control Protocol
 - IP: Internet Protocol

**Internet = Gesamtheit aller Netzwerke und Computer
die über TCP/IP- Verbindungen erreichbar sind.**

Prinzip und Organisation

- **Client-Server Prinzip**
 - **Server** (Diener, Anbieter)
Programm auf einem Rechner A, das einen bestimmten Dienst anbietet und der von einem anderen Nutzer abgerufen werden kann.
Bsp.: ftp-, news-, mail-,www-Server
 - **Client** (Klient, Kunde)
(Anwender-)Programm auf einem Rechner B, das den Dienst anfordert.
- **Angebot von Internet-Diensten**
Provider (Anbieter,Versorger) = ISP (Internet Service Center)

Firmen, Vereine oder Gemeinschaften, die diesen Service anbieten und verwalten.

Internet-Dienste

- **TELNET**
Zugriff auf andere Rechner (Dialogzugriff).
- **FTP**
Übertragen (Kopieren) von Dateien zwischen verschiedenen Rechnern.
- **E-MAIL**
Austausch von Nachrichten.
- **NEWS**
Diskussion- und Informationsmedium zu vielen verschiedenen Themen.
- **WWW**
Multimediales hyperlink-basiertes Informationssystem und Schnittstelle den anderen Internet-Diensten.

Client-Programme unter UNIX

- telnet (für TELNET)
- ftp (für FTP)
- mail, pine (für E-MAIL)
- netscape (für WWW)

Adressen und Namen im Internet

Prinzip der Adreßsuche

1. Anwender (client) stellt Suchanfrage
2. Übersetzung der symbolischen Rechneradresse in die zugehörige numerische Internet-Adresse. (zentralen Adreßliste verwalten die Internetnamen)
3. Zentrale Adreßlisten werden für Domain-bereiche geführt.
4. Paketvermittlung der Anfrage und Erreichen des Zielrechners mit Server-Programm
5. Internetseite wird dem Client übermittelt.

IP-Adressen

Länge: 32 Bit = 4 Byte à 8 Bit

Format von IP-Adressen: **a.b.c.d**

a,b,c,d \in [1,254]

0 und 255 haben Sonderrolle

Beispiele gültiger Internet-Adressen:

141.41.49.2

194.163.245.145

Aufbau von Internet-Adressen I

- Symbolische Namen anstelle numerischer Internet-Adressen!
- Organisation:
 - Jede an Internet angeschlossene Einrichtung betreibt einen eigenen Name-Server
 - In jedem Land gibt es einen Haupt-Domain-Name-Server (verteilte Rechnerdatenbank mit Namen/Adreßeinträgen)

Aufbau der Domain-Namen

host.[subdomain.[subdomain.[...]]domain.topleveldomain

Aufbau der Domain-Namen:

host.[subdomain.[subdomain.[...]]]domain.topleveldomain



Top-Level-Domain:

- „Ur“-Internet (in USA)
edu, com, gov, ...
- Internationale Länderkennzeichnungen
de - Deutschland
uk - Großbritannien
usw.

Adressierungsschema URL (Uniform Resource Locator)

Beispiel: <http://www.informatik.uni-ulm.de>

Vor dem Doppelpunkt steht das zu verwendende Protokoll (z.B. ftp, http,...).

E-mail

- **Verwendung:** Austausch von Nachrichten zwischen 2 Benutzern.
- **Adresse:**
 - Bezeichnung des Adressaten
 - Internet-Adresse
 - Allgemeine Form:
[username@domainname](#)
- **Adressierungsschema** (vereinfacht!)
 - Schicke Nachricht zum Rechner **domainname** und dort an die Mailbox **username**

Telnet

- **Verwendung:** Fernbedienung eines entfernt gelegenen Rechners per *remote login*. (Telnet ist ein Internet-Protokoll, mit dem man sich bei einem anderen Rechner zum Dialogbetrieb anmelden kann!)
- **Fern-Anmeldung:**
 - Eingabe des Zielrechners:
telnet domainname
 - Wenn der Zielrechner zugreifbar ist, so erscheint die übliche Anmeldeprozedur:
login:
Password:

WWW (World Wide Web)

- **Verwendung:** Integration aller bisherigen Dienste.
- **Architektur:**
 - Protokoll HTTP (Hyper Text Transfer Protocol)
Betrieb der Verbindung zwischen WWW-Server und WWW-Client
 - Sprache HTML (Hyper Text Markup Language)
Erstellung von WWW-Dokumenten
 - Client (=Browser)
- **Aufruf:**
netscape &

Der Editor vi

- Texteditor (lat. Edire) ist ein Programm zur Texterstellung und Textbearbeitung
- **vi** ist der Standardeditor unter **UNIX**
- **vi** (engl. *visual*) baut auf den älteren Editoren **ed** und **ex** auf.
- **vi** ist bildschirmorientiert, d.h. es können gleichzeitig mehrere zeilen eines Textes angezeigt werden.
- **vi** bietet umfangreiche Funktionen zur ...
 - Cursor-Bewegung
 - Textbearbeitung
 - Textsuche und -ersetzung
 - Kopieren, Verschieben, Ersetzen von Textteilen
 - Konfiguration des Editors

Aufruf und Beenden des vi

Kommandosyntax:

vi [optionen] <f1> <f2>....

meist

vi <filename>

Speichern und Ausstieg:

ZZ (im Kommandomodus)

:wq (im ex-Modus)

bedingter Ausstieg (falls keine Änderungen gemacht wurden)

:q (im ex-Modus)

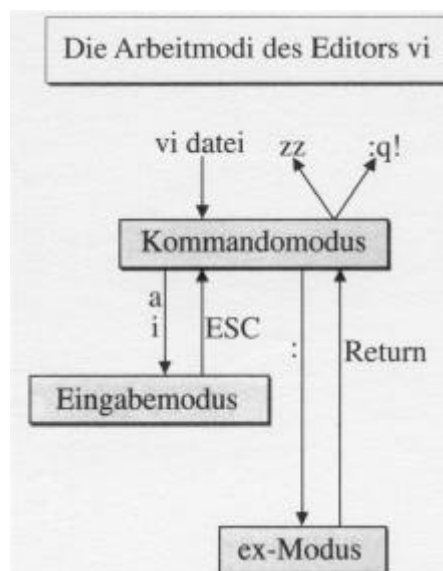
Unbedingter Ausstieg (Änderung werden nicht gespeichert)

:q! (im ex-Modus)

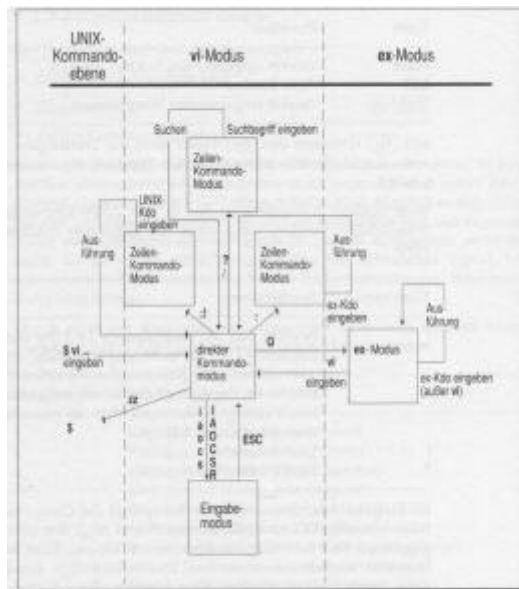
Arbeitsmodi des vi

- Kommandomodus
 - Eingaben sind Kommandos
- Eingabemodus
 - Eingaben sind Texte, die in einer Datei gespeichert werden.
- Kommandozeilenmodus (ex-Modus)
 - Eingaben sind Spezialkommandos; abgeleitet vom Vorläufer-Editor **ex**
 - Abschluss mit **return**-Taste
- Wechsel vom Eingabemodus in den Kommandomodus mit **ESCAPE**-Taste (**Esc**).
- Nach Aufruf des **vi** im Kommandomodus

Übersicht vi-Modi



Übersicht vi-Modi



vi-Kommandos Aufruf/Ausstieg

Aufruf / Ausstieg	
vi	name wird beim Ausstieg (Schreiben) angegeben
vi name	Aufruf der Datei <i>name</i> am Anfang
vi +n name	Aufruf der Datei <i>name</i> bei Zeile <i>n</i>
vi + name	Aufruf der Datei <i>name</i> am Ende
vi name1,name2,...	Aufruf der Datei <i>name1</i> , weiter mit :n
vi -t tag	Aufruf der Datei, die <i>tag</i> enthält
vi +/muster name	Aufruf der Datei <i>name</i> mit Positionierung auf erste Zeile die <i>muster</i> enthält
view name	nur Lesen der Datei <i>name</i>
ZZ	Schreiben und Ausstieg
:w	Schreiben
:q	Ausstieg (bedingt: falls keine Änderungen)
:q!	unbedingter Ausstieg
:wq	Schreiben und Ausstieg

vi-Kommandos Texteingabe

Kommando	Wirkung
a	<i>append</i> : nach der Stelle, auf der der Cursor steht, werden die im Folgenden eingegebenen Zeichen eingefügt
A	nach dem Zeilenende einfügen (Zeile: s.u.) - wie \$a
i	<i>insert</i> : vor der Stelle auf der der Cursor steht, werden die im Folgenden eingegebenen Zeichen eingefügt
I	Am Zeilenanfang einfügen - wie ^i
o	neue Zeile nach aktueller Zeile und dort a
O	neue Zeile vor aktueller Zeile und dort a

Cursorbewegungen I

Kommando	Bedeutung
h	ein Zeichen nach links, max. bis Zeilenanfang
j	ein Zeichen nach unten (nächste Zeile)
k	eine Zeile nach oben
l	ein Zeichen nach rechts, max. bis Zeilenende
4l	4 Zeichen nach rechts ähnlich für h j k
H	zum Bildschirmumfang
M	zur Bildschirmmitte
L	zum Bildschirmende (unterste Zeile auf dem Bildschirm)
+	nächste Zeile
-	vorhergehende Zeile
return	wie +
~	auf erstes Nicht-Leerzeichen
0	Zeilenanfang
\$	Zeilenende
w	zum Anfang des nächsten Wortes
W	wie w, . wird als Worttrenner ignoriert
b	(back) zum Anfang des vorausgehenden Wortes (ohne Sonderzeichen)
B	wie b, analog zu W
e	zum nächsten Wortende
E	wie e, analog W
)	auf nächsten Satzanfang
(auf vorigen Satzanfang
fx	zum nächsten x
Fx	auf letztes x
tx	vor nächstes x
Tx	vor letztes x
:	wiederholt letztes f F t T

Cursorbewegungen II

Kommando	Bedeutung
ctrl-f	um eine Bildschirmseite weiter (<i>forward scrolling</i>)
ctrl-b	um eine Bildschirmseite zurück (<i>backward scrolling</i>)
ctrl-d	um eine halbe Bildschirmseite weiter (<i>down</i>)
ctrl-u	um eine halbe bildschirmseite zurück (<i>up</i>)
ctrl-e	um eine Zeile vorwärts
ctrl-y	um eine Zeile rückwärts
z+	neue Seite mit der aktuellen Zeile oben
z.	neue Seite mit der aktuellen Zeile in der Mitte
z-	neue Seite mit der aktuellen Zeile unten
:\$	zum Dateiende
:1	zum Dateianfang (erste Zeile)
:5	zur fünften Zeile
:zahl	zur <i>zahl</i> -ten Zeile
:/string/	zur nächsten Zeile, die <i>string</i> enthält

Zeilennummern sichtbar :**set nu**
 Nummerierung wieder aus :**set nonu**

Textbearbeitung

Kommando	Bedeutung
x	Zeichen, auf dem der Cursor steht, löschen
dw	Wort löschen (ab der Cursor-Position)
3dw	die nächsten drei Worte löschen
dd	Zeile, in der der Cursor steht, löschen
3dd	die nächsten drei Zeilen löschen
d\$	vom Cursor bis Zeilenende löschen
D	wie d\$
-.2,7d	Zeile 2 bis Zeile 7 löschen
.,5d	von aktueller Zeile bis Ende alles löschen (wie dG)
d)	lösche nächsten Satz
dG	alles von aktueller Pos. bis Dateiende löschen
u	letztes Kommando rückgängig machen
:r file	Inhalt von <i>file</i> nach aktueller Zeile einfügen
yy	(<i>yy</i>)k) Zeile in einen Puffer schreiben (siehe p)
3yy	aktuelle und folgende 2 Zeilen (also 3) in den Puffer schreiben
p	(<i>par</i>) Puffer nach aktueller Zeile einfügen
P	wie p , nur vor der aktuellen Zeile
J	aktuelle Zeile durch folgende verlängern (zusammenfügen)
cw	change word: am Wortende erscheint ein \$, Wort kann überschrieben werden
c\$	am Zeilenende erscheint ein \$. Zeichen bis zum Zeilenende werden überschrieben
o/sr/1sr2/	erstes Vorkommen von <i>sr1</i> in der aktuellen Zeile wird durch <i>sr2</i> ersetzt (<i>substitute</i>)
o/sr/1sr2/g	alle Vorkommen von <i>sr1</i> in der aktuellen Zeile werden durch <i>sr2</i> ersetzt (<i>substitute global</i>)
:1,\$o/sr/1sr2/	im gesamten Text (1,\$) wird in jeder Zeile das erste Vorkommen von <i>sr1</i> durch <i>sr2</i> ersetzt
:1,\$o/sr/1sr2/g	im gesamten Text (1,\$) werden alle Vorkommen von <i>sr1</i> durch <i>sr2</i> ersetzt
3,7w file	Text von der 3. bis 7. Zeile in Datei <i>file</i> schreiben

Textsuche und Textersetzung

- Zu suchender Text entweder **direkt** oder über ein **Muster**.
- Schrägstrich **/** : nächstes Vorkommen aufsuchen
- Fragezeichen **?** : letztes Vorkommen aufsuchen

- **direkt** : d.h. fester Text wird eingegeben

Beispiel: **:/informatik**

suche die nächste Stelle, an der die Zeichenkette

informatik *vorkommt.*

Textsuche und Textersetzung

Muster

- Eckige Klammern "**[]**" ein Zeichen von einer Aufzählung.
- Punkt "**.**" steht für ein beliebiges Zeichen.
- Stern "*****" beliebige viele Wiederholungen der vorstehenden Zeichenkette.
- **^** steht für Zeilenanfang; **\$** steht für Zeilenende.
- **\<** Wortanfang und **\>** Wortende.
- Bei Sonderzeichen **^ \$? . * [] \ < >** Backslash **** voranstellen.

Literatur

- H. Herold: *UNIX Grundlagen*, Addison Wesley
- J. Gulbins: *UNIX*, Springer Verlag
- M. Wielsch: *LINUX*, Data Becker
- F. Schweiggert: *Allgemeine Informatik*, Vorlesungsunterlagen