FH Bielefeld
University of
Applied Sciences

Vorlesung

Betriebssysteme

Teil 2
Einführung und Shell

Dozent

Prof. Dr.-Ing.

Martin Hoffmann

martin.hoffmann@fh-bielefeld.de

Raum D206

Ringstr. 94

FH Bielefeld
University of
Applied Sciences

Ziele der heutigen Vorlesung

- Betriebssysteme, Historie und Hintergründe
- Vorbereitung für das Praktikum: Shell
- Die verschiedenen Architekturen von Betriebssystemen kennenlernen
- Spezielle Betriebsarten wie Teilnehmer- und Teilhaberbetrieb, Application-Server-Betrieb und Terminalserver-Betrieb kennenlernen

FH Bielefeld
University of
Applied Sciences

Inhalt der Vorlesung

- Historie und Hintergründe
- Shell
- Betriebsmodi
  - Kern- und Benutzermodus
  - Hardware Grundlagen, Schutzebenen, Systemaufrufe
- Aufbau des Betriebssystems
  - Monolithisch, geschichtet, …
  - Unix, Linux, Mac OS, Windows, Android

FH Bielefeld
University of
Applied Sciences

Einführung: Aufgaben

Aufgaben eines Betriebssystems:

- **Verwaltung** der Betriebsmittel (Ressourcen)
  - *Prozess*verwaltung (erzeugen, löschen, zuteilen, synchronisieren)
  - *Speicher*verwaltung
  - Verwaltung des *Dateisystems*
  - Verwaltung von *Geräten*
  - Verwaltung der *Benutzer*
- **Abstraktion** von der Hardware
  - Die Eigenschaften der Hardware werden vor dem Benutzer verborgen.
  - Die Benutzung der Hardware wird durch eine einheitliche Schnittstelle gewährleistet.
  - Die Hardware stellt zusammen mit dem Betriebssystem eine *abstrakte Maschine* dar, auf der die Benutzerprogramme aufsetzen.

FH Bielefeld
University of
Applied Sciences

Einführung: Historie

Historie von Betriebssystemen:

- erste Rechnergeneration (ca. 1945-1955):
  - **Kein** Betriebssystem (‚*Single Purpose Computers*‘),
  - Programmierung über Steckbrett oder Lochstreifen... keine Programmiersprachen.
- zweite Generation (ca. 1955-1965):
  - **Stapelverarbeitung**, einfache Job Control.
  - Programmiersprachen: Assembler, Fortran...
- dritte Generation (1965-1980):
  - Mehrbenutzer- und **Multiprogramm**betrieb, Steuerung über Terminal (Tastatur und Bildschirm).
  - Programmiersprachen: C, Pascal…
- vierte Generation (ab ca. 1975):
  - **Interaktive** Systeme mit grafischer Benutzeroberfläche, verteilte Betriebssysteme Netzwerkbetriebssysteme,
  - Multiprozessorsysteme. Objektorientierte Programmiersprachen

FH Bielefeld
University of
Applied Sciences

Microsoft und Windows

- Gründung Microsoft 1975
- Erfolg: MS-DOS („the day Gary Kildall went flying“)
- 100.000 Mitarbeiter, 75 Mrd $ Umsatz

MICROSOFT

- 95% Marktanteil auf PCs und Notebooks

FH Bielefeld
University of
Applied Sciences

Unix

Bell Labs:

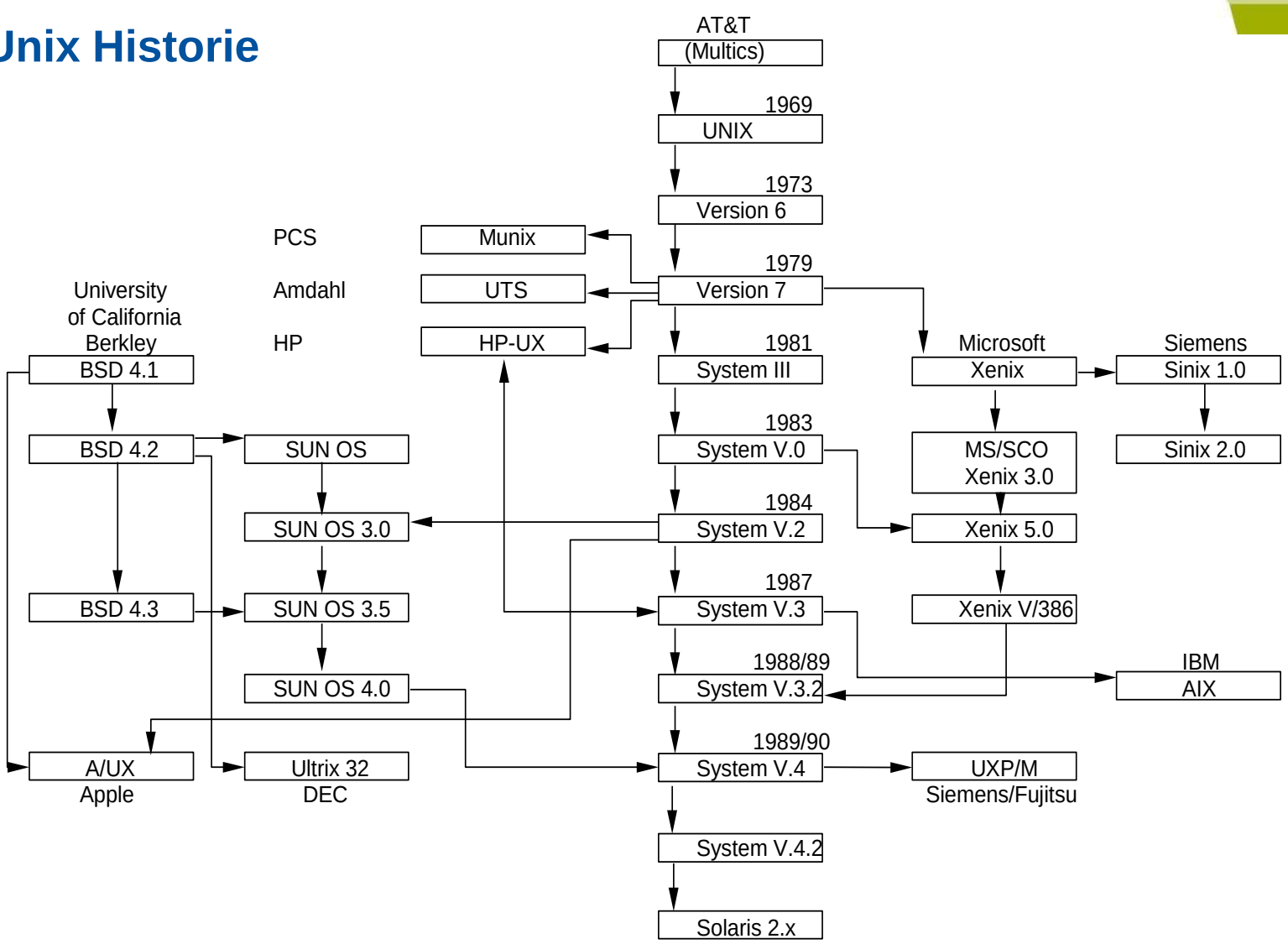
Ken Thompson
(1. Unix Version 1969, Sprache Assembler, später B),

Dennis Ritchie
(Weiterentwicklung der Sprache B zu C),
Brian Kernighan

Bild: Dennis Ritchie (stehend) und Ken Thompson bei der UNIX-Portierung auf die PDP-11 an 2 Teletype 33 Terminals (1970)

FH Bielefeld
University of
Applied Sciences

Unix Historie



Linux

- Linus Torvalds 1991
  - Unix Betriebssystem für IBM PC
  - Tanenbaums Minix

Title	Duties
Chief programmer	Performs the architectural design and writes the code
Copilot	Helps the chief programmer and serves as a sounding board
Administrator	Manages the people, budget, space, equipment, reporting, etc.
Editor	Edits the documentation, which must be written by the chief programmer
Secretaries	The administrator and editor each need a secretary
Program clerk	Maintains the code and documentation archives
Toolsmith	Provides any tools the chief programmer needs
Tester	Tests the chief programmer's code
Language lawyer	Part timer who can advise the chief programmer on the language

FH Bielefeld
University of
Applied Sciences

Linux

FH Bielefeld
University of
Applied Sciences

Smartphones

- Smartphones
  - Umfrage Betriebssysteme WS 2012/2013
    - Windows 2
    - Android 25
    - iOS 4
    - Symbian 0
    - Palm OS 0
  - Umfrage Betriebssysteme SoSe 2014
    - Windows 2
    - Android 22
    - iOS 8
    - Symbian 0
    - Palm OS 0
  - Umfrage Betriebssysteme SoSe 2015
    - Windows 1
    - Android 30
    - iOS 4
    - Symbian 0
    - Palm OS 0

FH Bielefeld
University of
Applied Sciences

Einführung: Arten

Einteilung von Betriebssystemen:

- Nach Betriebsart:
  - Stapelverarbeitung (batch processing)
    - Programme werden einzeln gestartet und verarbeitet.
    - Klassische Großrechnerbetriebssysteme.
  - Time-Sharing-Betriebssystem
    - Die Rechenleistung wird in Zeitscheiben aufgeteilt.
    - Interaktives Arbeiten (auch mehrerer Benutzer) ist möglich.
  - Echtzeitbetriebssystem
    - Garantierte Antwortzeiten können für Prozesse angegeben werden.
- Nach Rechnerarchitektur
  - Einprozessorsystem
  - Multiprozessorsystem
  - Verteiltes System (Rechenknoten haben eigene CPU und eigenen Speicher)

FH Bielefeld
University of
Applied Sciences

Einführung: Arten (Forts.)

Einteilung von Betriebssystemen (Forts.):

- Nach *Einsatzgebiet*
  - **Großrechnersysteme**
    - Bsp. OS/390 (IBM)
  - **Server**
    - Solaris (SUN), Linux, BSD, Windows Server (Microsoft)
  - **Multiprozessorsysteme**
    - spezielle Varianten von Windows, UNIX (Solaris)
  - **Personalcomputer**
    - Windows (XP, Vista, 7), MacOS, Linux
  - **Handheldcomputer / Smartphones**:
    - Android, iOS, Symbian, Blackberry, Windows Mobile, Palm OS
  - **Eingebettete Systeme**:
    - QNX, VxWorks
  - **Sensorknoten**:
    - TinyOS, Java VM

FH Bielefeld
University of
Applied Sciences

Codeumfang von Betriebssystemen

Jahr	AT&T	BSD	Minix	Linux	Solaris	Win NT
1976	V6, 9K					
1979	V7, 21K					
1980		4.1, 38 K				
1982	Sys III, 58 K	4.2, 98 K				
1984		4.3, 179 K				
1987	SVR3, 92 K		1.0 13 K			
1989	SVR4, 280 K					
1991				0.01, 10 K		
1993		Free 1.0, 235 K				3.1, 6 M
1994		4.4 Lite, 743 K		1.0, 165 K	5.3, 850 K	3.5, 10 M
1996				2.0, 470 K		4.0, 16 M
1997			2.0, 62 K		5.6, 1.4 M	
1999				2.2, 1 M		
2000		Free 4.0, 1.4 M			5.8, **2.0 M**	2000, **29 M**
2007						Vista, **50 M**

Windows 7: 70 M

Vgl. auch Tanenbaum, 2002: K = 1.000 LOC, M = 1000.000 LOC

FH Bielefeld
University of
Applied Sciences

Testfrage

- Nennen Sie die Basis- oder Kernelfunktionalitäten eines BS!
- Mögliche Antworten:

  Prozessmanagement

  Dateimanagement

  Behandlung von Echtzeitereignissen

  Bestimmung der Prozessreihenfolge

  Userverwaltung

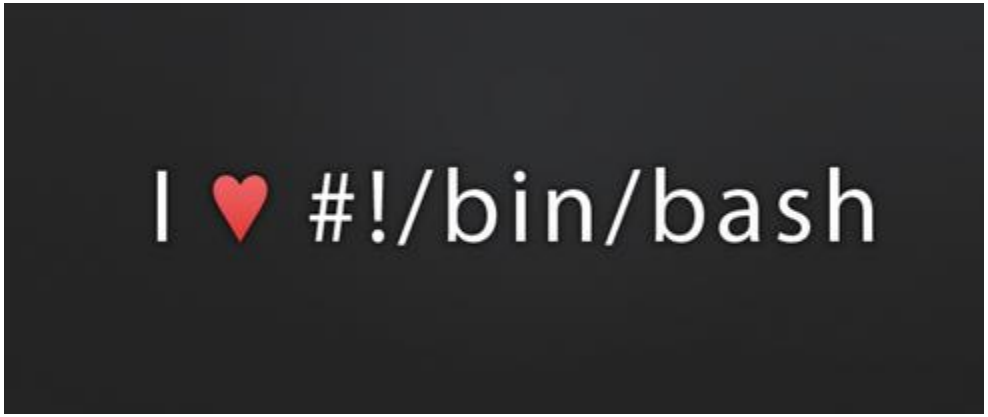
  Verwaltung des virtuellen Speichers

  Login

  Nachrichtentransport

FH Bielefeld
University of
Applied Sciences

Fingerübung: Linuxkonsole



FH Bielefeld
University of
Applied Sciences

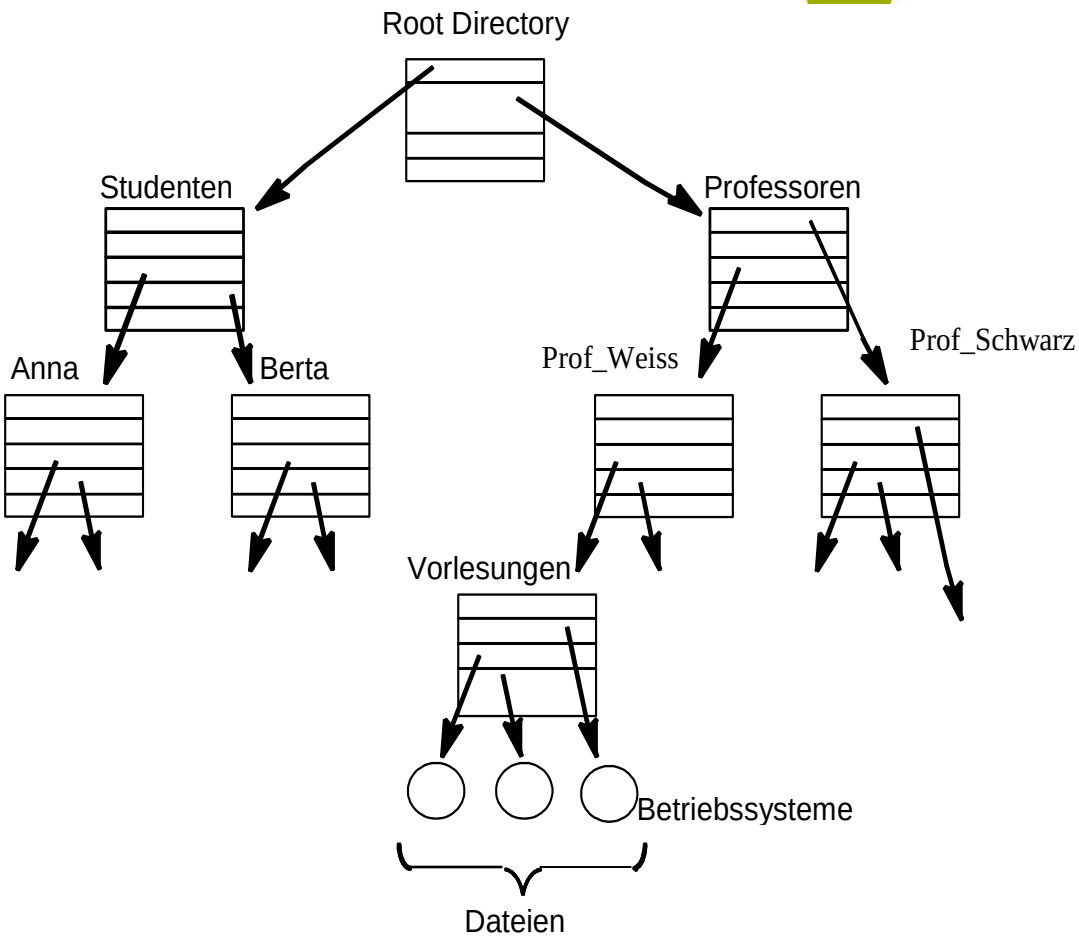
Dateien

- Eine UNIX-Datei ist eine Zeichenfolge (Bytefolge).
- Strukturierung durch den Benutzer
- 4 Dateiarten:
  - normale Dateien: Programme, Texte
  - Verzeichnisdateien (Directories): enthalten Verweise auf Dateien und weitere Verzeichnisdateien
  - Gerätedateien (special files)
  - Pipes

FH Bielefeld
University of
Applied Sciences

Verzeichnisse

- Dateien werden in Verzeichnissen (directories) abgelegt. Directories können weitere Directories enthalten: Hierarchie
- Das „höchste“ Directory heißt Root Directory.



File-System (File-Baum)

FH Bielefeld
University of
Applied Sciences

Adressierung von Dateien

- Durch Angabe ihres Pfadnamens (path name), d. h. alle Directories, ausgehend von root, die bis zu der gesuchten Datei durchlaufen werden müssen.
- z.B. **`/Professoren/Prof_Weiss/Vorlesungen/Betriebssysteme`**
- Absolute Pfadnamen beginnen mit ”/”, also bei root.
- Genau ein Directory ist jeweils als ”Working Directory” definiert.
- Relative Pfadnamen beginnen im Working Directory.
  z. B. Working Directory = **`/Professoren`**:
  rel. Pfadname: **`Prof_Weiss/Vorlesungen/Betriebssysteme`**

FH Bielefeld
University of
Applied Sciences

Testfrage

- Pfadnamen: Das Working Directory sei auf /homes/schroeder/fischer eingestellt. Wohin weist der Pfadname "/homes/stoiber/merkel„

- Mögliche Antworten:

  /homes/schroeder/fischer

  /homes/schroeder/fischer/stoiber/merkel

  /homes/stoiber/merkel

  /homes/stoiber/merkel/schroeder/fischer

FH Bielefeld
University of
Applied Sciences

Schutzmechanismen

- Jede Datei, jedes Directory erhält einen 9-bit-Code (Schutzbits, Protection Code):

R (read): lesen
W (write): schreiben
X (execute): ausführen (für Verzeichnis: suchen)

- Eine „1“ bedeutet: das entsprechende Recht wird gegeben.
- Beispiel: 111 101 001 heißt: rwx r-x --x

  Eigentümer darf lesen, schreiben, ausführen
  Gruppe darf lesen und ausführen
  alle anderen dürfen nur ausführen.
- Überprüfung der Zugriffsberechtigung beim Öffnen der Datei.

FH Bielefeld
University of
Applied Sciences

Testfrage

- Eine Datei habe die Schutzbits "111 101 001". Sie sind nicht der Eigentümer, gehören aber zur Gruppe. Dürfen Sie schreiben?

- Mögliche Antwort:

  ja

  nein

FH Bielefeld
University of
Applied Sciences

cd, pwd

- Wie kommt man mit dem cd-Kommando ...
  - in sein home-Verzeichnis?
  - in das übergeordnete Verzeichnis?
  - in das root-Verzeichnis?

- Wie kann das aktuelle Arbeitsverzeichnis angezeigt werden?

FH Bielefeld
University of
Applied Sciences

Lösung zu cd, pwd

- cd
- cd ..
- cd /

- pwd

FH Bielefeld
University of
Applied Sciences

man

- Besorgen Sie sich Informationen zum man-Kommando.
  - Welches Programm wird zum Anzeigen der man-Pages verwendet (Pager)?
  - Wie kann man auf einer man-Page
    - rückwärts blättern?
    - nach einem Begriff suchen?
    - zur nächsten Fundstelle springen?
  - Wie verlässt man die man-Page?
  - Wie können Sie das man-Kommando verwenden, um herauszufinden, welche Kommandos das UNIX System für die Uhrzeit vorsieht?

FH Bielefeld
University of
Applied Sciences

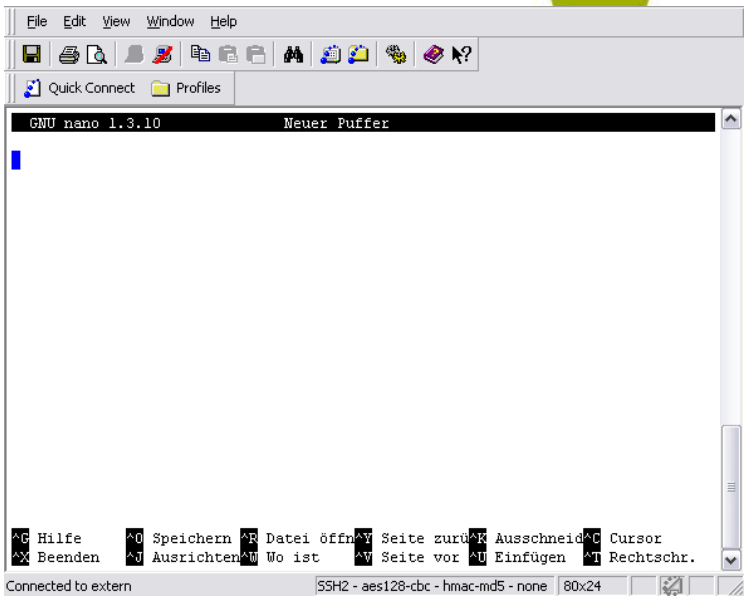
Lösung man

- man man
  - liefert Informationen zum man-Kommando
- Pager: less;
  - Hilfe dazu über h
- CTRL-B
  - zurückblättern
- /<Suchbegriff>
  - sucht in der man-Page nach dem Suchbegriff
  - n Springt zur nächsten Fundstelle
- q verlässt die man-Page
- man -k <Schlüsselwort>
  - liefert Informationen zu einem gegebenen Schlüsselwort
  - z.B. man -k time
    liefert Informationen zu Funktionen für die Zeitmanipulation.

FH Bielefeld
University of
Applied Sciences

Texteditor

- Editor "nano" in der Shell
  - Sonderbefehle
    - Beenden des Editors mit Nachfrage zum Speichern
      - STRG-X
    - Speichern
      - STRG-O
    - Laden
      - STRG-R
    - Suchen
      - STRG-W
    - usw.



FH Bielefeld
University of
Applied Sciences

ls

- Das Kommando ls hat verschiedene Optionen. Benutzen Sie das Kommando man, um festzustellen, mit Hilfe welcher Option das
Folgende am Bildschirm ausgeben werden kann
  - alle, auch die mit . beginnenden Dateien eines Verzeichnisses
  - Dateien eines Verzeichnisses in unsortierter Reihenfolge (in der die Dateien auf der Festplatte gespeichert sind)
  - Langformat Auflistung mit Dateityp, Zugriffsrechten, Anzahl von Hardlinks, Besitzername, Dateigröße u.s.w.
  - die horizontale Auflistung von mit Komma getrennten Dateien.

FH Bielefeld
University of
Applied Sciences

Lösung zu ls

- -a
  - do not ignore entries starting with .
- -f
  - do not sort, enable -aU, disable -lst
- -l
  - use a long listing format
- -m
  - fill width with a comma separated list of entries
- -u
  - with -lt: sort by, and show, access time
    with -l: show access time and sort by name
    otherwise: sort by access time

mkdir, cp, mv, rm, rmdir

- Erstellen Sie das Verzeichnis texte in Ihrem Home-Verzeichnis.
  - Erstellen sie mittels nano eine Datei Text.txt mit beliebigem Inhalt
  - Kopieren Sie sich die Datei Text.txt ins Verzeichnis texte.
- Erstellen Sie das Unterverzeichnis text1 im Verzeichnis texte.
  - Verschieben Sie die Datei Text.txt ins Verzeichnis text1.
- Erstellen Sie das Unterverzeichnis text2 im Verzeichnis texte.
  - Kopieren Sie die Datei Text.txt aus dem Verzeichnis text1 ins Verzeichnis Text2.
- Löschen Sie die Datei Text.txt aus dem Verzeichnis text1.
- Löschen Sie das Verzeichnis text1.

FH Bielefeld
University of
Applied Sciences

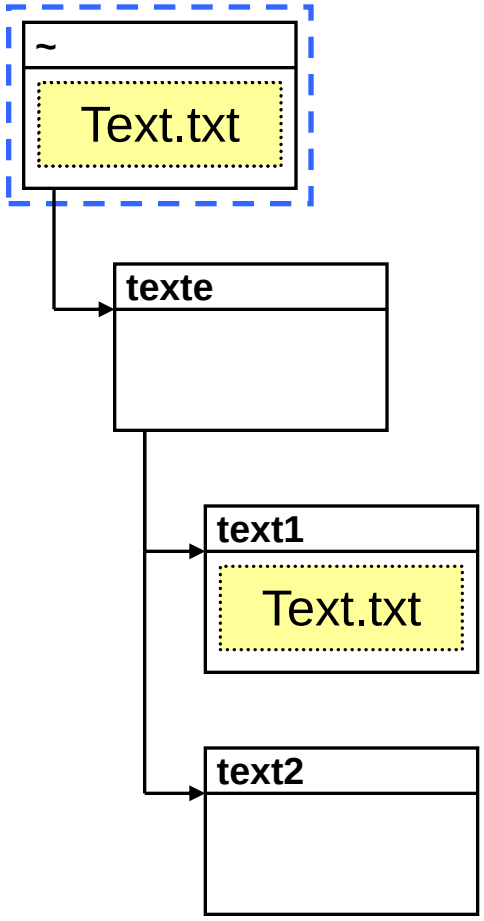
Lösung 1 zu mkdir, cp, mv, rm, …

- mkdir texte
- nano Text.txt
- cp Text.txt texte
- cd texte
- mkdir text1
- mv Text.txt text1
- mkdir text2
- cp text1/Text.txt text2
- cd text1
- rm Text.txt
- cd ..
- rmdir text1

FH Bielefeld
University of
Applied Sciences

Lösung 2 zu mkdir, cp, mv, rm, …

- mkdir texte
- nano Text.txt
- cp Text.txt texte
- cd texte
- mkdir text1
- mv Text.txt text1
- mkdir text2
- cp text1/Text.txt text2
- cd text1
- rm Text.txt
- cd ..
- rmdir text1



FH Bielefeld
University of
Applied Sciences

Ausgabeumlenkung cat, more, sort

- Lenken Sie die Ausgabe des Kommandos ls -f /etc in die Datei lsf.txt um.
  - Geben Sie die Datei lsf.txt am Bildschirm aus.
  - Geben Sie die Datei lsf.txt seitenweise am Bildschirm aus.
  - Geben Sie die sortierte Datei lsf.txt am Bildschirm aus
    - Alphabetisch
    - Umgedreht alphabetisch
    - Sortiert sowie seitenweise

FH Bielefeld
University of
Applied Sciences

Lösung zu Ausgabeumlenkung cat, more, sort

- ls –f /etc > lsf.txt
- cat lsf.txt
- more lsf.txt
- cat lsf.txt | sort | more
- cat lst.txt | sort –r | more
- sort lsf.txt | more

FH Bielefeld
University of
Applied Sciences

Einführung: Modi

Kern- und **Benutzermodus**:

- Zum Schutz des Betriebssystems (und damit des Rechners) vor fehlerhaften (oder böswilligen) Applikationen und zum Schutz der Applikationen untereinander haben moderne leistungsfähige Prozessoren ein Privilegiensystem
  - Betriebssystem läuft im **Kernmodus** (*kernel mode*, *supervisor mode*),
  - die Applikationen im **Benutzermodus** (*user mode*)

	Modus	
Privileg	Benutzermodus	Kernmodus
Ausführbare Maschinenbefehle	begrenzt	alle
Hardwarezugriffe	nein	Vollzugriff
Zugriff auf MMU	nein	ja
Zugriff auf Systemcode bzw. Daten	keiner / lesend	ja

Usermodus und Kernelmodus

- Usermodus (Benutzermodus)
  - Ablaufmodus für Anwendungsprogramme
  - Kein Zugriff auf Kernel-spezifische Code- und Datenbereiche
- Kernelmodus
  - Privilegierter Modus
  - Dient der Ausführung der Programmteile des Kernels
  - Schutz von Datenstrukturen des Kernels
- Umschaltung über spezielle Maschinenbefehle
- Aktueller Modus steht in einem Statusregister

FH Bielefeld
University of
Applied Sciences

Hardware-Grundlagen am Beispiel der Intel-Architektur

- Beispiel: x86-Architektur
  - Schutzkonzept über vier Privilegierungsstufen (Ring 0 - 3)
  - Prozess läuft zu einer Zeit in einem Ring
  - Meist werden aus Kompatibilitätsgründen zu anderen CPUs nur zwei Ringe unterstützt:

  - Ring 0: Kernelmodus (privilegiert, Zugriff auf Hardware möglich)
  - Ring 3: Usermodus (nicht privilegiert)
  - Übergang von Ring 3 nach Ring 0 über privilegierte Operation (int-Befehl)
  - Anmerkung: Ab x64/IA64 werden nur noch zwei Ringe unterstützt

FH Bielefeld
University of
Applied Sciences

Schutzebenen

- Die Schutzstufe eines Programms ist im **PSW** codiert.
- Wechsel zwischen den Schutzstufen (Ringen) nur mit CALLS möglich. Kein direkter Einsprung in niedrigeren Ring!
- Wechsel von Ring 3 nach Ring 0 über Kontextwechsel (Trap, Unterbrechung)

Schutzstufe	Typische Nutzung
Ring 0	Betriebssystemkern (Kernel)
Ring 1	Systemaufrufe
Ring 2	Shared Libraries
Ring 3	Benutzerprogramme

FH Bielefeld
University of
Applied Sciences

Einführung: Systemaufrufe

Betriebssystemaufrufe

Um aus dem Benutzermodus in den Kernmodus zu gelangen, gibt es folgende Möglichkeiten:

- **Hardware Unterbrechung** (*HW Interrupt*) , z.B.
  - Echtzeit-Uhr (Timer)
  - Anforderung eines E/A Gerät
  - Hardware Fehler (z.B. Stromversorgung, Speicherfehler…)
- **Software Unterbrechung** (*SW Interrupt*, *Trap*)
  - System Aufruf (*System Call*, *Supervisor Call*)
  - Software Fehler (z.B. Zugriff auf ungültige Adresse, ungültiger Befehl, Division durch 0 …)

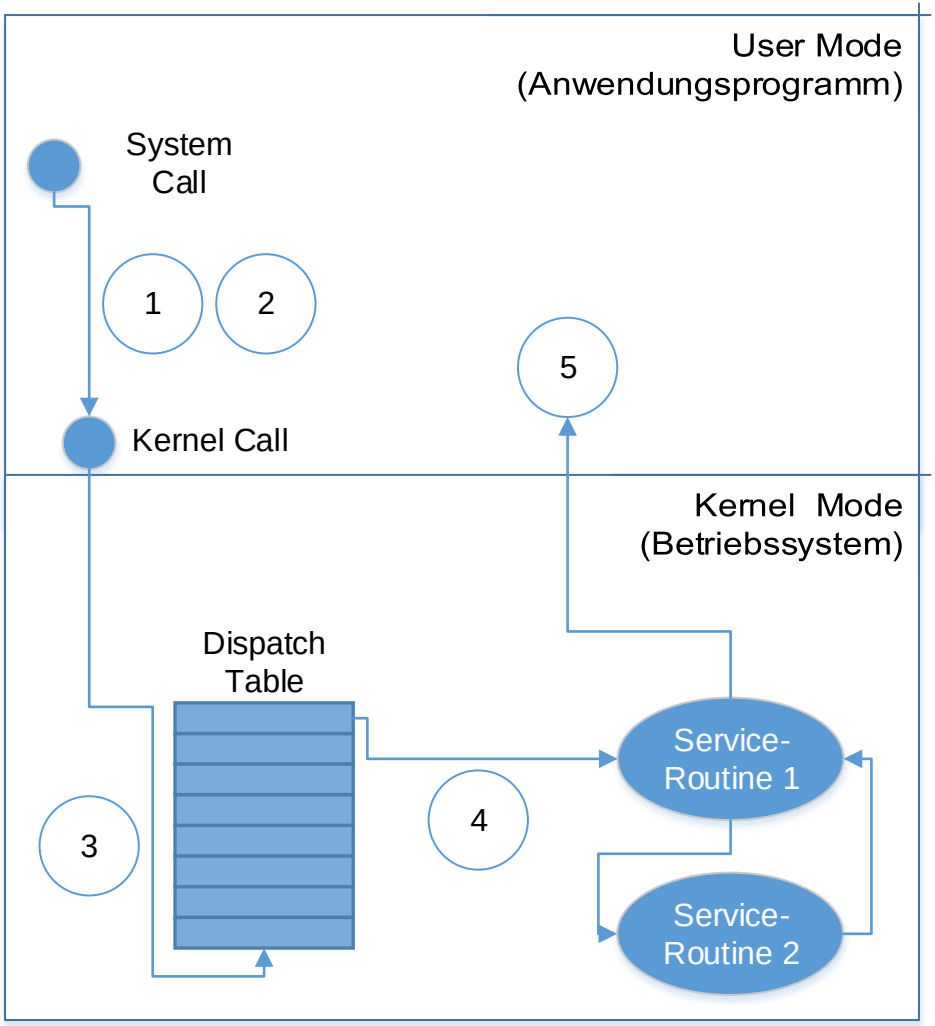
FH Bielefeld
University of
Applied Sciences

Einführung: Systemaufrufe (Forts.)

- Aufteilung in Benutzerebene (*user*) und Systemebene (*kernel*)
- Ablauf:

1. Anwender-Programm benötigt einen BS-Service: System Call.
2. Parameter werden in Übergabebereich platziert.
3. Steuerung wird an den Systemkern übergeben: Kernel Call (auch: Supervisor Call).
4. Kernel: identifiziert Service-Routine und ruft sie auf.
5. Service-Routine läuft ab und gibt Ergebnis an den Auftraggeber (Anwender-Programm) zurück.

- Zweiteilung (user - kernel) ergibt ungenügende Strukturierung.



FH Bielefeld
University of
Applied Sciences

Testfrage

- Zusammenarbeit zwischen Anwenderprogramm und BS: Welches sind die Ziele des Kernel-Call-Mechanismus?
- Mögliche Antworten:

  Erhöhung der Geschwindigkeit

  Vereinfachung der Programmierung

  Schutz von Speicherbereichen

  Einsparung von Energie

  Senkung der Taktfrequenz

  Entkopplung von User und System

FH Bielefeld
University of
Applied Sciences

Vorlesung

Vielen Dank für Ihre Aufmerksamkeit

Dozent

Prof. Dr.-Ing.

Martin Hoffmann

martin.hoffmann@fh-bielefeld.de

Raum D206

Ringstr. 94