

Betriebssysteme

Robert Kaiser

(HTTP: <http://www.cs.hs-rm.de/~kaiser>
EMail: robert.kaiser@hs-rm.de)

Wintersemester 2020/2021

1. Einführung



<https://www.freebsd.org/de/docs.html>

Einführung

- ① Was ist ein Betriebssystem?
- ② Geschichte der Betriebssysteme
- ③ Geschichte von UNIX
- ④ UNIX-Eigenschaften
- ⑤ Zusammenfassung

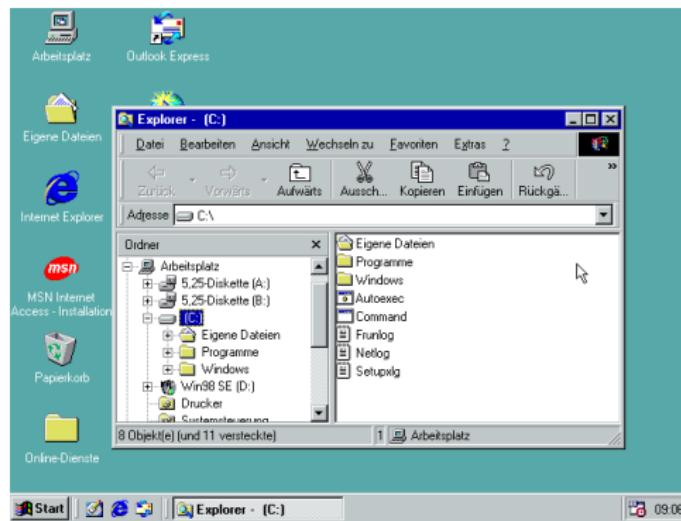
Was ist ein Betriebssystem?



<https://www.nullstelle.de/fun/bilder/computer/linux.jpg>

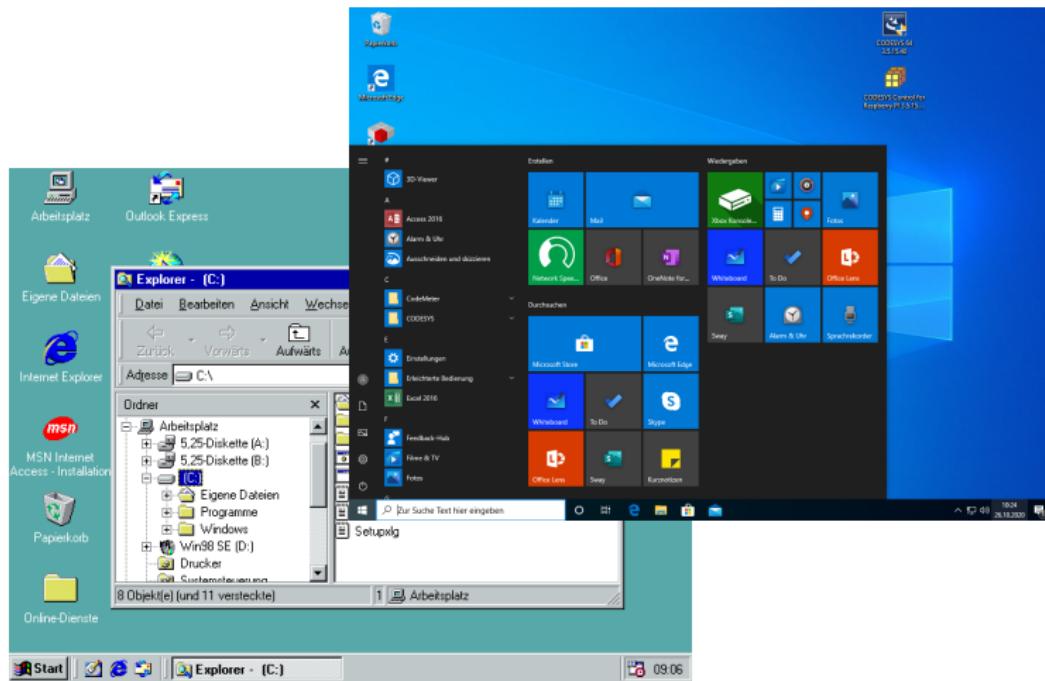
Woran erkennt man es?

GUI Design / Funktionalität?



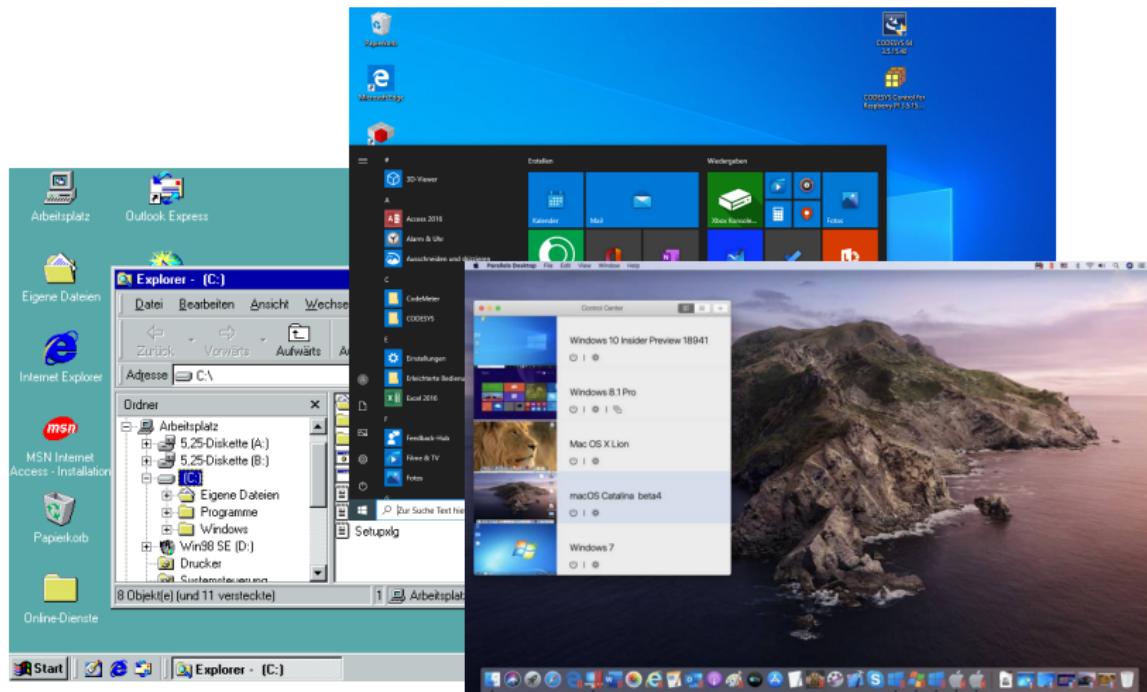
Woran erkennt man es?

GUI Design / Funktionalität?



Woran erkennt man es?

GUI Design / Funktionalität?



Papierkorb

Firefox

Google Chrome

Mozilla Thunderbird

WinSCP

Adobe Acrobat DC

Adobe Creative Cloud

1_Einfuehrung-slides.pdf - Adobe Acrobat Pro DC

Start Werkzeuge 1_Einfuehrung-slid... x

3 (7 von 13) 109% ⋮ Freigeben

Werkzeuge durchsuchen

- PDF-Datei erstellen
- Dateien zusammenführen
- PDF bearbeiten
- PDF-Datei exportieren
- Seiten verwalten
- Zur Überprüfung senden **NEU**
- Kommentieren
- Ausfüllen und unterschreiben
- Scan & OCR
- Schützen
- Mehr Werkzeuge

Woran erkennt man es?

GUI Design / Funktionalität?

© R. Kaiser, Hochschule RheinMain BS WS 2020/2021 1 - 3

Dateien in der Document Cloud speichern und freigeben Weitere Infos

10:36 26.10.2020



Your PC ran into a problem and needs to restart. We're just collecting some error info, and then we'll restart for you.

20% complete



For more information about this issue and possible fixes, visit <https://www.windows.com/stopcode>

If you call a support person, give them this info:

Stop code: CRITICAL_PROCESS_DIED

Wann sieht man es?

Wenn es „crasht“?

:(
:(

Your PC ran into a problem and needs to restart. We're just collecting some error info, and then we'll restart for you.

20% complete

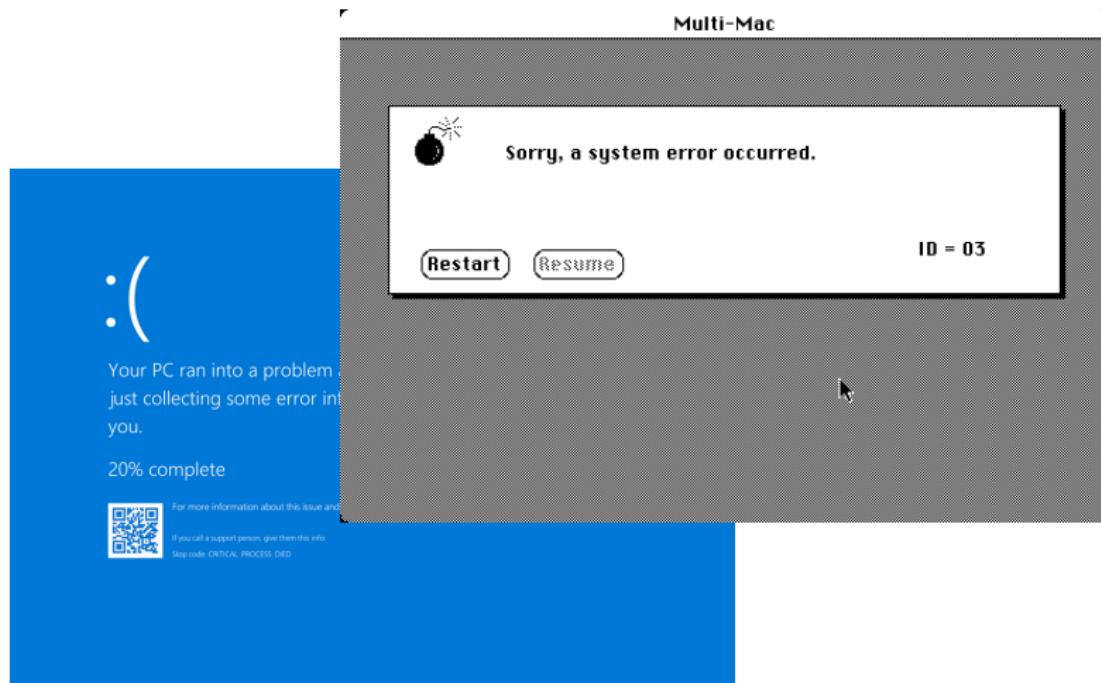


For more information about this issue and possible fixes, visit <https://www.windows.com/stopcode>

If you call a support person, give them this info:
Stop code: CRITICAL_PROCESS_DIED

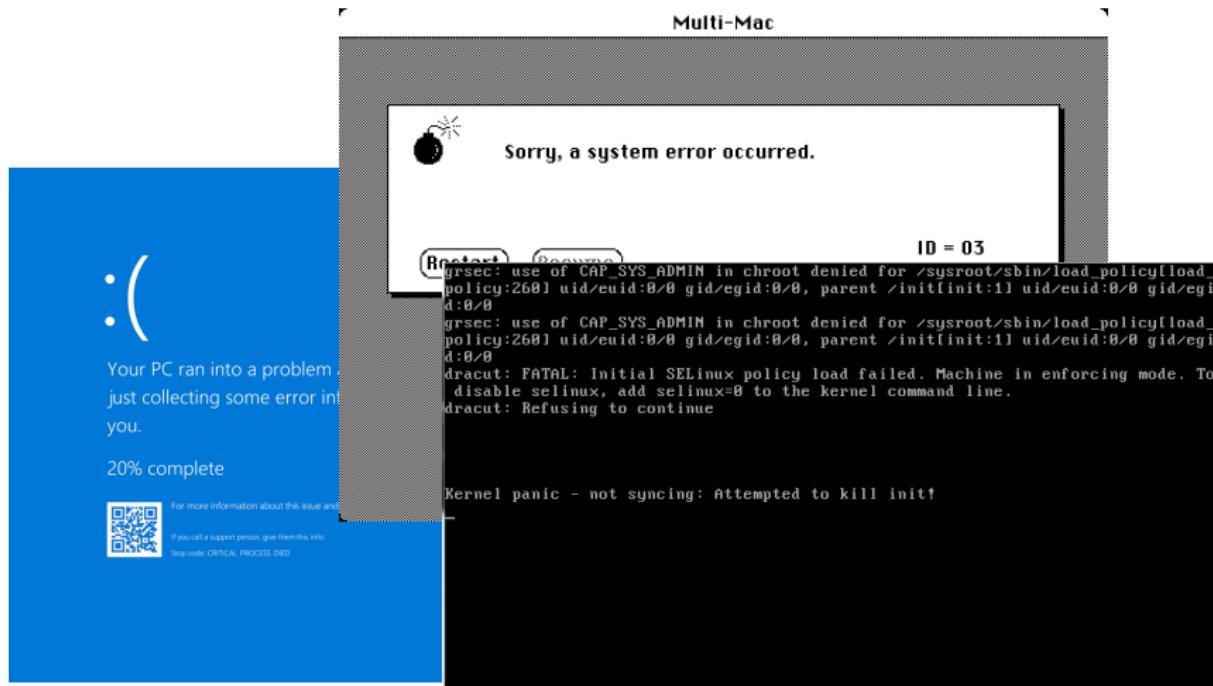
Wann sieht man es?

Wenn es „crasht“?

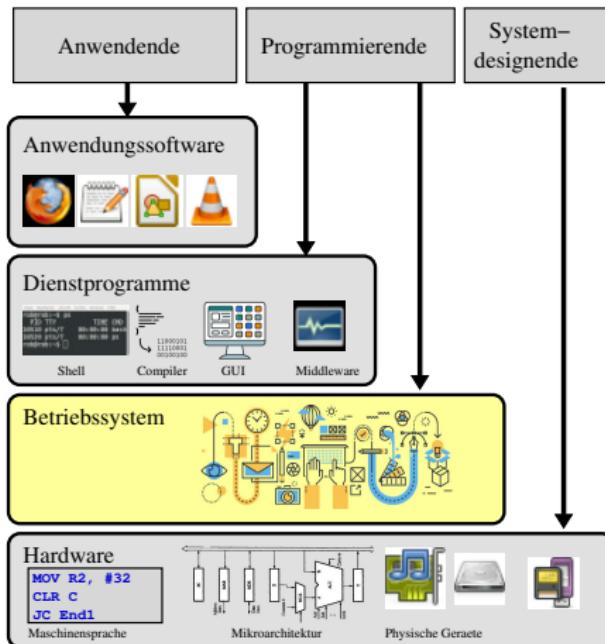


Wann sieht man es?

Wenn es „crasht“?

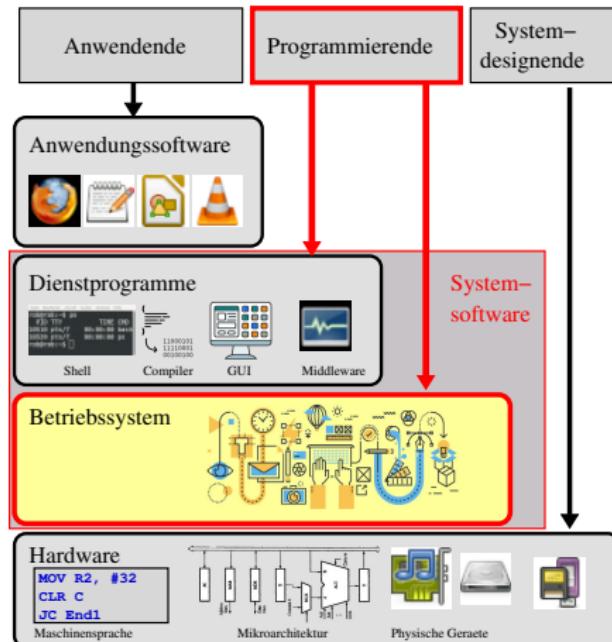


Allgem. Schichtenmodell eines Rechensystems



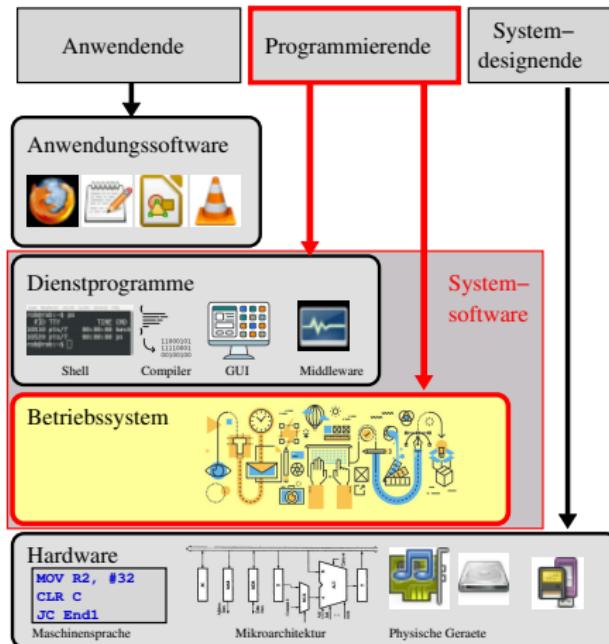
- Anwendungssoftware: der „Nutzen“ des Rechners
- Systemsoftware: Zum Betrieb des Rechners erforderlich („notwendiges Übel?“)
- Betriebssystem: Grundlegendstes Systemprogramm

Allgem. Schichtenmodell eines Rechensystems



- Anwendungssoftware: der „Nutzen“ des Rechners
- Systemsoftware: Zum Betrieb des Rechners erforderlich („notwendiges Übel?“)
- Betriebssystem: Grundlegendstes Systemprogramm

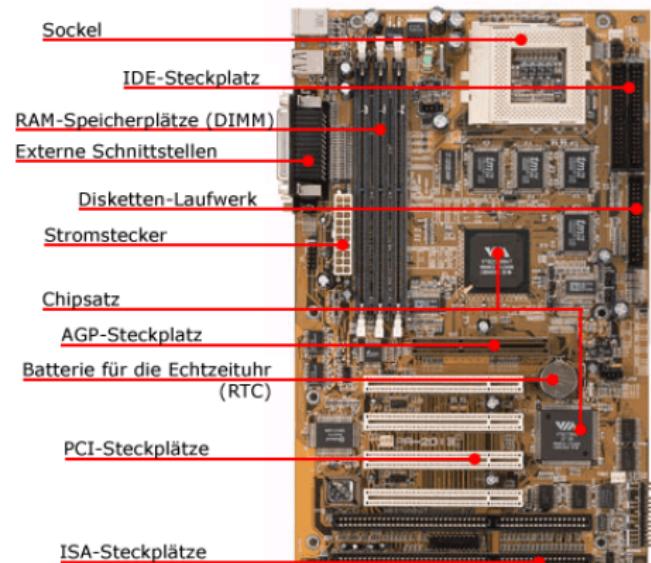
Allgem. Schichtenmodell eines Rechensystems



- Anwendungsoftware: der „Nutzen“ des Rechners
- Systemsoftware: Zum Betrieb des Rechners erforderlich („notwendiges Übel?“)
- Betriebssystem: Grundlegendstes Systemprogramm

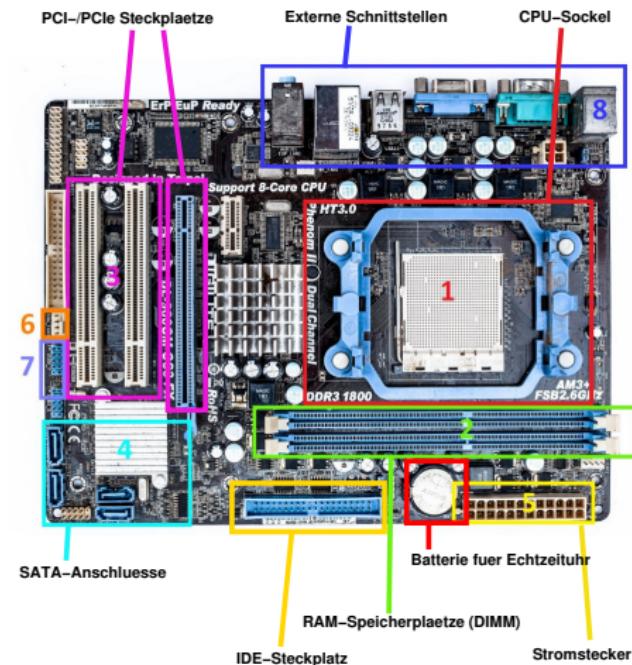
Komponenten eines Rechners

- Prozessor(en) („CPUs“)
- Hauptspeicher („Core Memory“)
- Bildschirmschnittstelle
(Grafikkarte, Terminal)
- Massenspeicher
(Festplatte, Flash-Speicher)
- Netzwerkschnittstelle(n)
- Uhren
- Soundchip
- Drucker, Webcam,



Komponenten eines Rechners

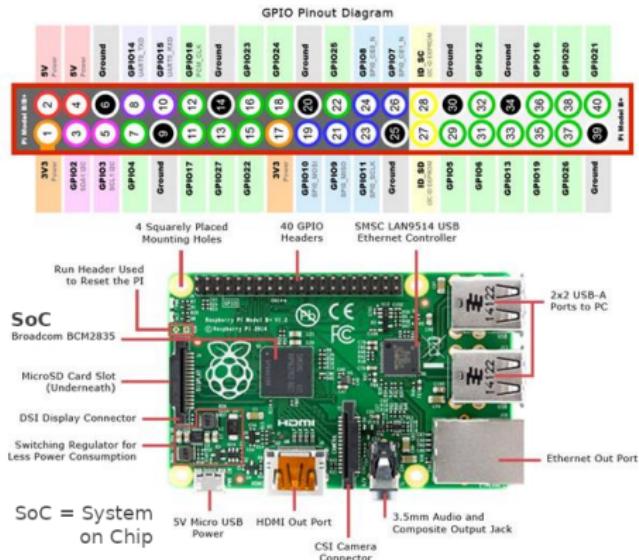
- Prozessor(en) („CPUs“)
- Hauptspeicher („Core Memory“)
- Bildschirmschnittstelle
(Grafikkarte, Terminal)
- Massenspeicher
(Festplatte, Flash-Speicher)
- Netzwerkschnittstelle(n)
- Uhren
- Soundchip
- Drucker, Webcam,



Komponenten eines Rechners



- Prozessor(en) („CPUs“)
 - Hauptspeicher („Core Memory“)
 - Bildschirmschnittstelle
(Grafikkarte, Terminal)
 - Massenspeicher
(Festplatte, Flash-Speicher)
 - Netzwerkschnittstelle(n)
 - Uhren
 - Soundchip
 - Drucker, Webcam,



Komponenten eines Rechners

- Das Erstellen von Programmen zur Verwaltung und möglichst optimalen Nutzung dieser Komponenten ist **schwierig**.
- Aufgrund des rapiden technischen Fortschritts sind solche Programme zudem sehr kurzlebig.
- Beispiel: Festplatten-Zugriff:

ATA Befehlssatz (Auszug)¹

ATA Command	Instruction			OP
...
READ SECTOR(S)	RdSec	RDSK	RS	20h
READ SECTOR(S) WITHOUT RETRIES	RdSecN	RDSKN	RSN	21h
READ DMA QUEUED EXT	RdDMAQEx	RDQX		26h
...
SEEK		SEEK		70h
CFA TRANSLATE SECTOR	CfaTransSec	CFTS		87h
EXECUTE DEVICE DIAGNOSTIC	Diagnose	DIAG	DIAG	90h
...

¹<https://ulinktech.com/wp-content/uploads/2020/05/ATA-Command-Table-in-OpCode-Order.pdf>

Komponenten eines Rechners

- Das Erstellen von Programmen zur Verwaltung und möglichst optimalen Nutzung dieser Komponenten ist **schwierig**.
- Aufgrund des rapiden technischen Fortschritts sind solche Programme zudem sehr kurzlebig.
- Beispiel: Festplatten-Zugriff:

ATA Befehlssatz (Auszug)¹

Wer will schon mit solch einer Schnittstelle arbeiten, um Informationen langfristig zu speichern oder darauf zuzugreifen ?!?

ATA Command	Instruction			OP
...
READ SECTOR(S)	RdSec	RDSK	RS	20h
READ SECTOR(S) WITHOUT RETRIES	RdSecN	RDSKN	RSN	21h
READ DMA QUEUED EXT	RdDMAQEx	RDQX		26h
...
SEEK		SEEK		70h
CFA TRANSLATE SECTOR	CfaTransSec	CFTS		87h
EXECUTE DEVICE DIAGNOSTIC	Diagnose	DIAG	DIAG	90h
...

¹<https://ulinktech.com/wp-content/uploads/2020/05/ATA-Command-Table-in-OpCode-Order.pdf>

Rechner aus Anwendersicht

System-Software stellt Dienste bereit, z.B.:

- **Dateiverwaltung** (Dateien / Ordner öffnen / lesen / schreiben, Suchfunktionen, Zugriffsschutz, ...)
- **Ein-/Ausgabemöglichkeiten** (Tastatur / Maus, Bildschirm / Drucker, ..)
- **Programmierumgebung** (Editor, Compiler, Assembler, Linker, Debugger, Bibliotheken,..)
- Evtl. **Mehrbenutzerfähigkeit** (Gemeinsame Nutzung „teurer“ Gerätschaften)
- **Netzwerkzugang** (Internet, (W)LAN, ..)
- ...

BS als „virtuelle Maschine“

Der Funktionsumfang (**Befehlssatz**) der realen Maschine wird erweitert

- Ziele dabei:
 - ▶ Komplexität der „nackten“ Hardware verstecken
 - ▶ Abstrakte Schnittstellen auf hohem Niveau
 - Leichter zur verstehen und auch langlebiger
- Die Vorstellung solcher Abstraktionen und ihrer Realisierung im Betriebssystem ist Gegenstand der Vorlesung.

BS als Betriebsmittelverwalter

Definition: Betriebsmittel (BM, engl. *resources*)

Alle zuteilbaren und nutzbaren Hardware- und Software- Komponenten eines Rechensystems

- Aufgabe des Betriebssystems: Geordnete, kontrollierte Zuteilung („Allokation“) der BM an die um sie konkurrierenden Programme.
- Gemeinsame Nutzung von Betriebsmitteln (z.B. Drucker)
- Schutz vor unberechtigter Benutzung, Geheimhaltung von Informationen, wenn ihr Benutzer dies wünscht
- Vermitteln im Falle von Konflikten
- Abrechnen der Kosten der Betriebsmittelnutzung (*accounting*)

Zusammenfassung

Definition: Betriebssystem

Ein Betriebssystem ist ein Programm, das alle Betriebsmittel eines Rechensystems verwaltet und ihre Zuteilung kontrolliert und den Nutzern des Rechensystems eine virtuelle Maschine offeriert, die einfacher zu verstehen und zu programmieren ist, als die unterlagerte Hardware.



Zusammenfassung

Definition: Betriebssystem

Ein Betriebssystem ist ein Programm, das alle Betriebsmittel eines Rechensystems verwaltet und ihre Zuteilung kontrolliert und den Nutzern des Rechensystems eine virtuelle Maschine offeriert, die einfacher zu verstehen und zu programmieren ist, als die unterlagerte Hardware.



Zusammenfassung

Definition: Betriebssystem

Ein Betriebssystem ist ein Programm, das alle Betriebsmittel eines Rechensystems verwaltet und ihre Zuteilung kontrolliert und den Nutzern des Rechensystems eine virtuelle Maschine offeriert, die einfacher zu verstehen und zu programmieren ist, als die unterlagerte Hardware.



Zusammenfassung

Definition: Betriebssystem

Ein Betriebssystem ist ein Programm, das alle Betriebsmittel eines Rechensystems verwaltet und ihre Zuteilung kontrolliert und den Nutzern des Rechensystems eine virtuelle Maschine offeriert, die einfacher zu verstehen und zu programmieren ist, als die unterlagerte Hardware.



Zusammenfassung

Definition: Betriebssystem

Ein Betriebssystem ist ein Programm, das alle Betriebsmittel eines Rechensystems verwaltet und ihre Zuteilung kontrolliert und den Nutzern des Rechensystems eine virtuelle Maschine offeriert, die einfacher zu verstehen und zu programmieren ist, als die unterlagerte Hardware.



Zusammenfassung

Definition: Betriebssystem

Ein Betriebssystem ist ein Programm, das alle Betriebsmittel eines Rechensystems verwaltet und ihre Zuteilung kontrolliert und den Nutzern des Rechensystems eine virtuelle Maschine offeriert, die einfacher zu verstehen und zu programmieren ist, als die unterlagerte Hardware.



Zusammenfassung

Definition: Betriebssystem

Ein Betriebssystem ist ein Programm, das alle Betriebsmittel eines Rechensystems verwaltet und ihre Zuteilung kontrolliert und den Nutzern des Rechensystems eine virtuelle Maschine offeriert, die einfacher zu verstehen und zu programmieren ist, als die unterlagerte Hardware.



Zusammenfassung

Definition: Betriebssystem

Ein Betriebssystem ist ein Programm, das alle Betriebsmittel eines Rechensystems verwaltet und ihre Zuteilung kontrolliert und den Nutzern des Rechensystems eine virtuelle Maschine offeriert, die einfacher zu verstehen und zu programmieren ist, als die unterlagerte Hardware.



Geschichte der Betriebssysteme

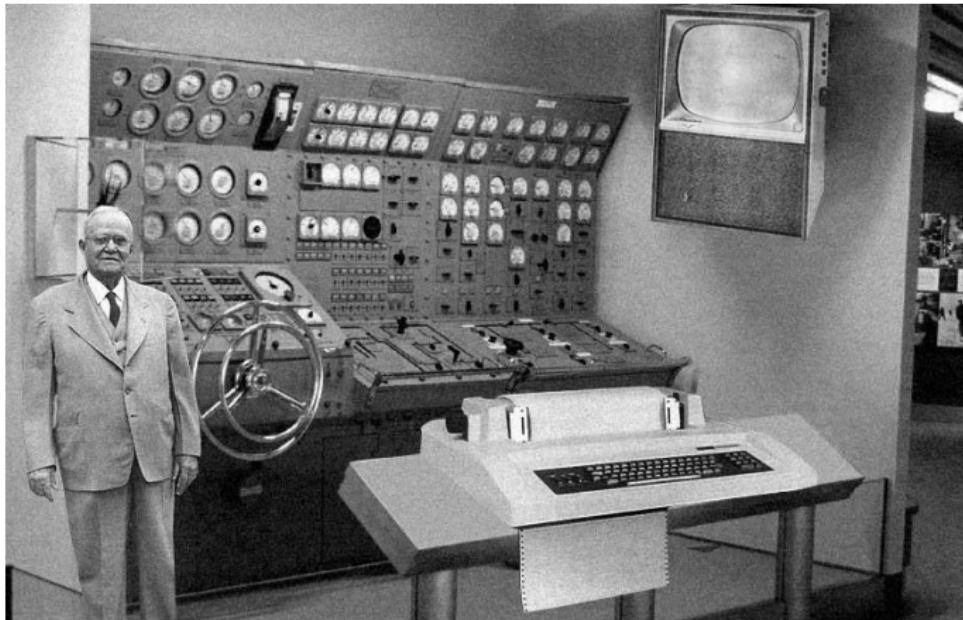
Entwicklung in etwa parallel zu Rechnergenerationen:

- (1945-1955): Röhren und Steckkarten
- (1955-1965): Transistoren und Stapelverarbeitung
- (1965-1980): ICs und Mehrprogrammbetrieb
- (1980-1990): PCs und Netzwerkbetriebssysteme
- (1990-heute): ...?

Generation (1945-1955)

- USA: Howard Aiken (Harvard), John von Neumann (Princeton), J.P.Eckert, W. Mauchley, (UPenn)
- Deutschland: Konrad Zuse
- Eine Gruppe von Personen kümmert sich um Entwurf, Bau, Programmierung, Betrieb und Wartung des Rechners
- „Programmieren“ = Verdrahten von Steckkarten oder absolute Maschinensprache (**keine** Programmiersprachen, nicht einmal Assembler).
- Ab Anfang der 50er Jahre: Benutzung von Lochkarten als Ersatz für Steckkarten.

Generation (1945-1955)



<https://www.snopes.com/Fact-check/1954-home-computer/>

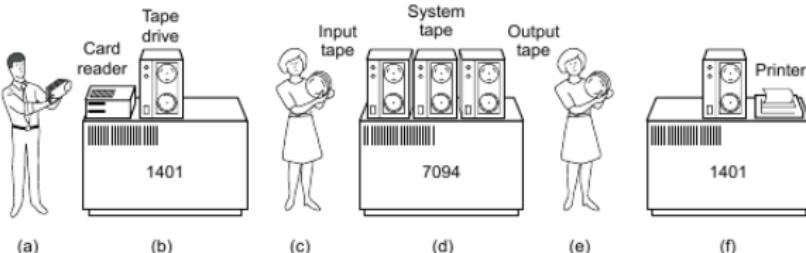
... Betriebssystem?!?

Generation (1955-1965)



- Durch die Einführung der Transistorstechnologie werden Rechner zuverlässig genug, um an Kunden verkauft zu werden
- Unterscheidung zwischen Entwicklern, Herstellern, **Operateuren*Innen**, Programmierern und Wartungspersonal
- Zunächst Ausführung einzelner „Jobs“ (= Programm oder Menge von Programmen) in Form von Lochkartenstapeln mit hohem Anteil manueller Arbeiten
- Rationalisierung des Betriebs (engl.: *operating*) durch Einführung des „Stapelverarbeitungsbetriebs“ (*batch system*) ...

Stapelverarbeitung

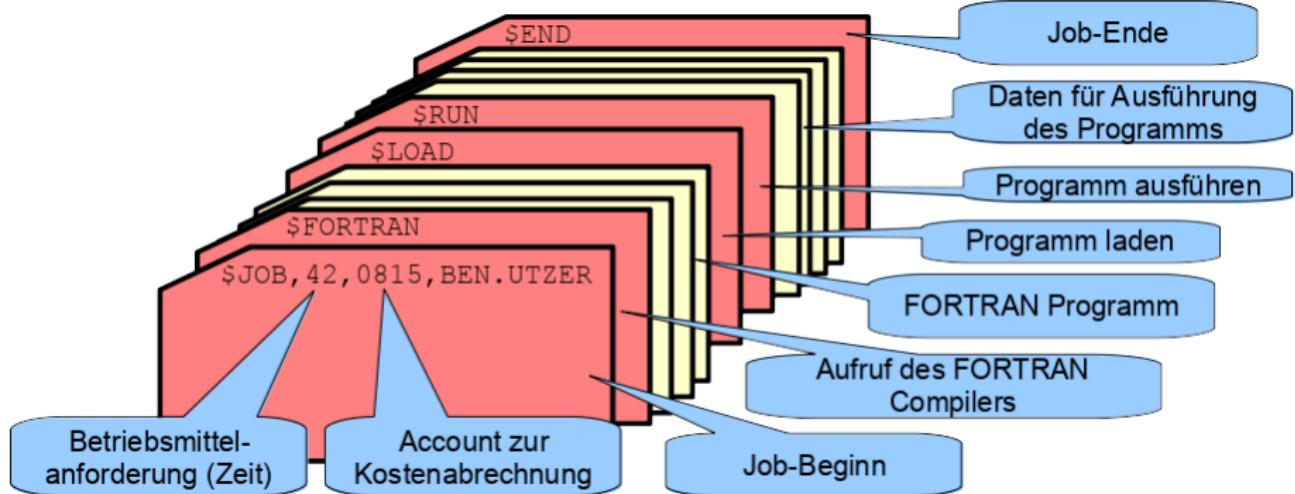


Aus [Tanenbaum]:

Moderne Betriebssysteme

- (a) Programmierende bringen Lochkartenstapel zur Lesestation
(z.B. IBM 1401: relativ preiswerter Rechner für kommerzielle DV)
- (b) Lesestation sammelt Kopien (1:1) auf Magnetband
- (c) OperateurIn trägt Magnetband von Zeit zu Zeit zum Hauptrechner
- (d) Hauptrechner (z.B. IBM 7094 mit mehreren Bandstationen) erzeugt für das Eingabeband ein Ausgabeband mit Ergebnissen aller Programme. Die Abarbeitung geschieht eigenständig durch ein „Monitorprogramm“.
- (e) OperateurIn trägt Ausgabeband zurück
- (f) Drucken des Ausgabebandes unabhängig vom Hauptrechner

Stapelverarbeitung



- Kontrollkarten sind Vorläufer der heutigen Job-Kontrollsprachen und der Kommandointerpreter
- Das Monitorprogramm zur Abarbeitung der Job-Folge ist der Vorläufer eines Betriebssystems

Generation (1965-1980)

- Einführung von *Rechnerfamilien* (Urvater IBM /360) mit gleichem Befehlssatz aber Unterschieden hinsichtlich Preis und Leistung ⇒ Rechner sind „softwarekompatibel“: Dasselbe Programm läuft auf allen Rechnern der Familie.
- Gleichermaßen für wissenschaftliche wie kommerzielle Berechnungen geeignet
- Besseres Preis/Leistungsverhältnis durch integrierte Schaltkreise (ICs)
- Anforderung: Die gesamte Software (auch das Betriebssystem) sollte auf allen Modellen der Familie lauffähig sein.
- Verschiedenste Anforderungen an das BS: E/A-orientierte Anwendungen vs. numerische Berechnungen, etc.

⇒ Komplexes Betriebssystem ...

Generation (1965-1980)

- Einführung von *Rechnerfamilien* (Urvater IBM /360) mit gleichem Befehlssatz aber Unterschieden hinsichtlich Preis und Leistung ⇒ Rechner sind „softwarekompatibel“: Dasselbe Programm läuft auf allen Rechnern der Familie.
- Gleichermaßen für wissenschaftliche wie kommerzielle Berechnungen geeignet
- Besseres Preis/Leistungsverhältnis durch integrierte Schaltkreise (ICs)
- Anforderung: Die gesamte Software (auch das Betriebssystem) sollte auf allen Modellen der Familie lauffähig sein.
- Verschiedenste Anforderungen an das BS: E/A-orientierte Anwendungen vs. numerische Berechnungen, etc.

⇒ Komplexes Betriebssystem ...

Eigenschaften von OS/360

- Mehrere Zehnerpotenzen größer als frühere Stapelverarbeitungsprogramme
- Millionen Zeilen Assemblercode
- Von Tausenden von Programmiererenden entwickelt
- Tausende von Fehlern
- Fortlaufende Releases notwendig

Niemals „fehlerfrei“...

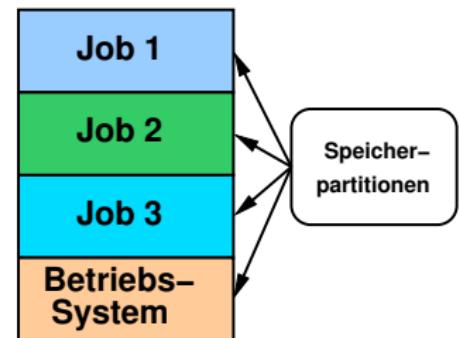


Mehrprogrammbetrieb

- Neue Schlüsseltechnik (Multiprogramming, Multitasking)
- Ziel: Vermeiden von Wartezeiten für „teuren“ Prozessor während E/A-Vorgängen, besonders relevant in kommerziellen Anwendungen.

- Lösung:

- ▶ Gleichzeitiges Bereithalten mehrerer Jobs im Hauptspeicher („Partitionierung“)
- ▶ Anstatt auf E/A zu warten:
Umschalten auf anderen Job



⇒ Neue Anforderung: Schutz der Speicherbereiche
(wurde später hinzugefügt)

Mehrprogrammbetrieb (2)



Variante: Timesharing

- Drang nach kurzen Antwortzeiten
- Jeder Benutzer hat über ein eigenes Terminal on-line Zugang zum System
 - Schnelle Reaktion des Systems auf Benutzereingaben (vorrangig behandelt), Stapelverarbeitung „im Hintergrund“
 - Speicherschutz unerlässlich (vgl. CTSS (MIT, [Corbato 1962]) auf modifizierter IBM 7094)

Timesharing

Das Beispiel: MULTICS

- MULTplexed Information and Computing System
- Zu Beginn (1965) gemeinsames Projekt von MIT, Bell Labs, General Electric, brach auseinander wegen technischer Schwierigkeiten
- Ziel: Hunderte von Nutzern gleichzeitig im Dialogbetrieb auf GE645.
- Letztendlich nur im MIT und einigen andern Einrichtungen produktiv eingesetzt
- **Aber:** bedeutender Einfluss auf nachfolgende Systeme (UNIX!)

Aufkommen der Minicomputer

- Weniger Speicher, geringere Leistung als Großrechner zu einem Bruchteil des Preises
- Beginn 1961: DEC PDP-1
 - ▶ 4K 18-bit Worte
 - ▶ „nur“ \$120.000,- (5% vom Preis einer IBM 7094)
- Für bestimmte, nicht-numerische Aufgaben sogar noch schneller als eine 7094
- Schnelle Abfolge von (**nicht** zueinander kompatiblen) PDP-Rechnern bis zur PDP-11 (1970)

„Mini-“Computer



<https://www.bell-labs.com/usr/dmr/www/picture.html>

Ken Thompson und Dennis M. Ritchie an einer PDP-11 (ca. 1972)

4. Generation (1980-1990)

- Personal Computer als Individuen zugeordnete Werkzeuge („Workstation“)
- Getrieben durch LSI und VLSI-Entwicklung²: preiswert, zugleich aber leistungsstark wie Mini- oder Großrechner
- Netzwerke für Kommunikation und Kooperation
- Dominante Betriebssysteme: MS-DOS und UNIX
- Netzwerkbetriebssysteme erlauben Zugang zu anderen Rechnern, Dateitransfer, gemeinsame Benutzung von Informationen (z.B. TCP/IP Netzwerk Utilities, Network File System)

²LSI = Large Scale Integration, VLSI = very Large Scale Integration

4. Generation (1980-1990)

- Hohe Grafikleistung, benutzerfreundliche Oberflächen (z.B. Apple Macintosh) erschließen neue Benutzergruppen, die keine eigentlichen Rechnerkenntnisse besitzen (müssen).



<https://commons.wikimedia.org/w/index.php?curid=51833>



<https://photos5.appleinsider.com/gallery/29370>

47319-000-3x2-Apple-History-Mac-launch-xl.jpg



https://upload.wikimedia.org/wikipedia/commons/c/c3/Silicon_Graphics_Indy.jpg



https://www.wikiwand.com/en/Sun_Ultra_series

Heute

Neue Anwendungen gekennzeichnet durch

- steigende Komplexität
 - ⇒ Probleme bei der Sicherheit
- neu geforderte Funktionalitäten
 - ▶ Web-Anbindung
 - ▶ Multimedia
 - ▶ Sicherheit (*security und safety*)
 - ▶ Fehlertoleranz/Robustheit
 - ▶ Echtzeitfähigkeit
 - ▶ Skalierbarkeit

Hardware-Trends

- Bis ca. 2000 gab es regelmäßige Steigerungen der Rechenleistung durch höhere Taktfrequenzen
- Physikalische Grenzen sind hier mittlerweile erreicht
 - ▶ Ausweg: on-Chip Parallelität (z.B. Multicore-Prozessoren)
- Vernetzung: drahtlos, mobile Knoten, höhere Bandbreiten
 - ▶ Mobile / Cloud Computing
- Speicher
 - ▶ Durch neue Technologien (z.B. NVMe-Speicher) werden große, nicht-flüchtige Hauptspeicher möglich

⇒ **Viele neue Herausforderungen, auch für Betriebssysteme**

Aktuelle Themen

- **Symmetrisches Multiprocessing (SMP):** Das Betriebssystem unterstützt symmetrische Multiprozessorsysteme, (alle Prozessoren sind gleichwertig).
→ ist heute meist gegeben
- **Asymmetrisches Multiprocessing (AMP):** Das Betriebssystem unterstützt heterogene Multiprozessorsysteme.
 - ▶ Spezialprozessoren (z.B. GPUs) zur Performance-Steigerung
 - ▶ Kombination von leistungsfähigen mit energieeffizienten Prozessoren (z.B. ARM big.LITTLE)
- **Virtualisierung:** Mehrere BS-Schnittstellen auf einem System:
 - ▶ Unterstützung für Vielfalt von Anwendungen, Ausführbarkeit ohne Änderung.
 - ▶ Konsolidierung (nicht nur) für Rechenzentren
 - ▶ Fehlerlokalität (*Multi Level Secure Systems*)
- **Große, persistente Speicher:** In-Memory Datenbanken, Festplatten bald überflüssig?
- **Echtzeitfähigkeit:** In Kombination mit den oben Genannten.

Aktuelle Trends

- Übergang zu **verteilten Betriebssystemen**: Das gesamte Netzwerk erscheint Benutzern wie ein einziges System.
- **Cloud Computing**: Virtualisierung von Rechenleistung im Internet
- Drang nach **standardisierten Schnittstellen**
(APIs = *Application Programming Interfaces*)
- **Hochverfügbarkeit**
- **Energieeffizienz** („Green IT“)
- Unterstützung für die **Administration großer Systeme**, Netzwerk und Anwendungen (Integriertes System-, Netzwerk- und Anwendungs-Management).

Heutiges Spektrum an BS

- **Mainframe-Betriebssysteme:** Massendatenhaltung, hohe E/A-Raten, „legacy“ Anwendungen. (z.B. OS/390, BS2000)
- **Server-Betriebssysteme:** Bereitstellung gemeinsam genutzter Betriebsmittel und Dienste. (z.B. UNIX / Linux, Windows Server)
- **Arbeitsplatzrechner- (Desktop-) Betriebssysteme:** (z.B. Windows, MacOS, Linux)
- **Echtzeit-Betriebssysteme:** Gesicherte Einhaltung von Zeitschranken für die Verarbeitung. (z.B. VxWorks, QNX, RTLinux)
- Betriebssysteme für **eingebettete Systeme**: i.d. R. Produkte mit stark eingeschränktem Betriebsmittelumfang (Handys, PDAs, SmartCards, ABS, ...).
Oft auch **Echtzeiteigenschaften** gefordert. (z.B. Symbian, Linux, ...)

Beispiele: Embedded Systems



<https://de.ccm.net>

Smartphone:
Android Linux



<https://www.raspberrypi.org/products/raspberry-pi-zero>

RasPi Zero: Linux



<https://avm.de/produkte/fritzbox/fritzbox-7590/>

Router: Linux

Geschichte von UNIX



Geschichte von UNIX



vitra. Produkte Downloads

Unix Chair

Antonio Citterio, 2010



Geschichte von UNIX



vitra. Produkte Downloads

Unix Chair

Antonio Citterio, 2010



Kleine UNIX-Historie

- 1965 - AT&T Bell Labs, General Electric und MIT beginnen mit Entwicklung des Betriebssystems MULTICS („MULTIplexed Information and Computing System“, s.o.): gute Ideen, wenig Erfolg
- 1969 – Ken Thompson entwickelt UNICS („UNiplexed Information and Computing System“) auf PDP-7 (4KB Speicher!)
- Ken Thompson, Dennis Ritchie und andere entwickeln UNIX für PDP-11
- Dateisystemaufbau und -Schnittstelle, Kommandointerpreter (Shell) und anderes entstammen MULTICS-Ideen
- 1973 - Reimplementierung in C (von Ritchie entworfen)
- 1974 - Meilenstein-Papier zu Unix von Ritchie und Thompson in CACM
- Einfachheit und Eleganz (später: ACM Turing Award)



Kleine UNIX-Historie (2)

- Schnelle Verbreitung im akademischen Bereich, da das System mit allen Quellen ausgeliefert wurde.
 - ▶ Version 6 in 1976
 - ▶ Bis 1977 an ca. 500 Institutionen vergeben
 - ▶ Version 7 (Portables UNIX) in 1978, mit Variante 32/V für DEC VAX
- Zahlreiche, divergierende Versionen
- Zeitweise Spaltung AT&T System V und Berkeley System Distribution (BSD) (s.u.), ab 1998: Versuch der Vereinheitlichung
- POSIX (Portable Operating System Interface for UUNIX)
 - ▶ durch das anerkannte, unabhängige IEEE-Gremium fand POSIX Beachtung als Standard 1003.1
- Heute: Mehrere, weitgehend POSIX-konforme Versionen (Linux, *BSD, Solaris (Sun), AIX (IBM), ..)

Berkeley UNIX

- UCB (*University of California at Berkeley*) bekam früh AT&T V 6
- eigene Weiterentwicklungen als 1BSD, 2BSD, 3BSD (*Berkeley Software Distribution*) abgegeben,
- 1979: 3BSD für VAX mit Virtual Memory, Demand Paging (u. A. durch William („Bill“) Joy, später SUN Mitgründer)
- 4BSD-Entwicklung für VAX mit zahlreichen Verbesserungen
 - ▶ ab 1979 4BSD-Entwicklung mit Förderung durch DARPA (*Defense Advanced Research Agency*).
 - DARPA-Net: **Ursprung des Internet**
 - ▶ Höhepunkt 4.2BSD in 1983: Neues Virtual Memory Interface, Device Driver Interface, Fast File System, Stabilere Signalverarbeitung, Socketbasierte Interprozesskommunikation, Netzwerkstandards: TCP/IP im Kern und Netzwerk-Utilities (rlogin, telnet, rcp, ftp, rsh).
- Heute: 4.3BSD/4.4BSD mit Quellcode allgemein zugänglich.
FreeBSD, OpenBSD, NetBSD, Dragonfly BSD
- MacOS X und Darwin basieren auf 4.4BSD



System V: das „kommerzielle“ UNIX

- AT&T Bell Labs:

- ▶ Zusammenfassung versch. Varianten bis 1982 zu „UNIX System III“
- ▶ ab 1983 System V als Produkt mit offiziellem Support und fortlaufenden Releases:
 - ★ V.2 in 1984 (100.000 Installationen),
 - ★ Release V.3 in 1987 mit viel Einfluss von 4.2BSD UNIX.

- Novell:

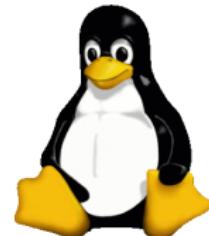
- ▶ 1993: kauft den UNIX-Teil von AT&T (300 Mio \$)
- ▶ SUN kauft sich 1994 von Novell frei (100 Mio \$)
- ▶ 1995: Verkauf der UNIX System V-Quellen an SCO
- ▶ Caldera übernimmt SCO (2001) und nennt sich SCO Group (2002)
- ▶ 2003: SCO-Caldera beschuldigt verschiedene Linux-Firmen und IBM des Diebstahls geistigen Eigentums (Intellectual Property)

Linux

- 1991: Der Finnische Informatikstudent Linus Torvalds beginnt mit der Entwicklung eines freien UNIX-Kerns entsprechend der Single UNIX Specification
- Gewinnt im Server-Markt stark an Bedeutung
- Attackiert Windows im Desktop-Bereich
- Unterstützung durch führende Hersteller (z.B. IBM, HP, ...)
- Aufgrund der freien Verfügbarkeit und wachsender Betriebsmittel auch erfolgreich im embedded-Bereich



- Heute das populärste Betriebssystem nach Windows (und MacOS?)
- Große Vielfalt an Distributionen, oft zugeschnitten auf das Einsatzgebiet (Desktop / Server / Embedded / ..)



Eigenschaften von UNIX

- Mehrbenutzer- und Mehrprogrammbetrieb
(*multi-user / multi-tasking*)
- Hierarchisches Dateisystem
 - ▶ Eine Wurzel: /
 - ▶ Physische Geräte sind Teil des Dateibaums
z.B. /dev/tty0 → serielles Terminal³
- Hohe Übertragbarkeit, dadurch verfügbar vom PDA
bis zum Großrechner
- größtenteils in C geschrieben
- Mächtige Kommandosprache („Shell“): einfache Bausteine, aber
flexible Verknüpfungsmöglichkeiten



³TTY: Abkürzung für *Teletype* = Fernschreiber

Eigenschaften von UNIX

- Mehrbenutzer- und Mehrprogrammbetrieb
(multi-user / multi-tasking)
- Hierarchisches Dateisystem
 - ▶ Eine Wurzel: /
 - ▶ Physische Geräte sind Teil des Dateibaums
z.B. /dev/tty0 → serielles Terminal³
- Hohe Übertragbarkeit, dadurch verfügbar vom PDA bis zum Großrechner
- größtenteils in C geschrieben
- Mächtige Kommandosprache („Shell“): einfache Bausteine, aber flexible Verknüpfungsmöglichkeiten



³TTY: Abkürzung für *Teletype* = Fernschreiber

Eigenschaften von UNIX

- Mehrbenutzer- und Mehrprogrammbetrieb
(*multi-user / multi-tasking*)
 - Hierarchisches Dateisystem
 - ▶ Eine Wurzel: /
 - ▶ Physische Geräte sind Teil des Dateibaums
z.B. /dev/tty0 → serielles Terminal³
 - Hohe Übertragbarkeit, dadurch verfügbar vom PDA bis zum Großrechner
 - größtenteils in C geschrieben
 - Mächtige Kommandosprache („Shell“): einfache Bausteine, aber flexible Verknüpfungsmöglichkeiten

³TTY: Abkürzung für Teletype = Fernschreiber

Beispiel: Shell-Kommandos

- Gegeben: Datei mit Städtenamen (einer je Zeile)
- Frage: Wie oft kommt „Wiesbaden“ vor?
- Folgende einfachen Kommandos gibt es schon:
 - ▶ grep sucht in seiner Eingabe nach Zeilen, die den Suchbegriff enthalten
 - ▶ wc (*word counter*) zählt Zeilen / Wörter / Buchstaben in seiner Eingabe
 - ▶ mit | kann man Aus- und Eingabe zweier Kommandos verbinden (*pipe*)
- Lösung: `grep "Wiesbaden" datei | wc -l`

Beispiel: Shell-Kommandos

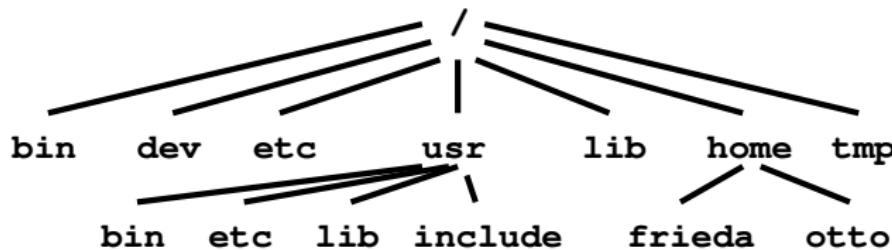
- Gegeben: Datei mit Städtenamen (einer je Zeile)
- Frage: Wie oft kommt „Wiesbaden“ vor?
- Folgende einfachen Kommandos gibt es schon:
 - ▶ grep sucht in seiner Eingabe nach Zeilen, die den Suchbegriff enthalten
 - ▶ wc (*word counter*) zählt Zeilen / Wörter / Buchstaben in seiner Eingabe
 - ▶ mit | kann man Aus- und Eingabe zweier Kommandos verbinden (*pipe*)
- Lösung: grep "Wiesbaden" datei | wc -l



Baukastenprinzip



Datebaum



- / Wurzel des Dateisystems
- /bin, /usr/bin ausführbare Programme, Kommandos
- /dev Device-Dateien, direkter Zugriff auf angegeschlossene Geräte (Scanner, Drucker, Platten...)
- /etc Konfigurationsdateien (Paßwörter, Netzkonfig, ...)
- /lib C-Bibliotheken
- /home Benutzerverzeichnisse
- /tmp Arbeitsverzeichnis für temporäre Dateien

Einige wichtige Kommandos

- cd aktuelles Verzeichnis wechseln
- rm Datei löschen (*remove*)
- mkdir Verzeichnis anlegen
- rmdir Verzeichnis löschen (*remove directory*)
- ps Prozeßliste ausgeben
- mv Datei verschieben / umbenennen (*move*)
- cat Datei(en) ausgeben
- ls Dateien auf/listen (ls -l für mehr Infos)
- man Online-Manual abrufen (z.B. man ls)
- mount Einbinden von Dateibäumen

Zusammenfassung

- ① Sichtweise eines Betriebssystems:
 - ▶ Betriebsmittelverwalter
 - ▶ (erweiterte) virtuelle Maschine
- ② Geschichtliche Entwicklung, dabei Begriffe:
 - ▶ Stapelbetrieb (*Batch Processing*)
 - ▶ Mehrprogrammbetrieb (*Multiprogramming*)
 - ▶ Dialogbetrieb (*Timesharing*)
- ③ Geschichte von UNIX
 - ▶ MULTICS → UNICS → *BSD, System V, Linux
- ④ UNIX-Eigenschaften