

Systeme I:
Betriebssysteme

Kapitel 2

Überblick Betriebssysteme

Wolfram Burgard



Überblick Betriebssysteme

- Aufgabe von Betriebssystemen
- Historische Entwicklung von Betriebssystemen
- Unterschiedliche Arten von Betriebssystemen

Aufgabe von Betriebssystemen

- Betriebssystem = Computerprogramm mit zwei grundsätzlichen Funktionen
 - Abstraktion der tatsächlich vorhandenen Hardware
 - Beispiel: Manipulation des Dateisystems (z.B. Erstellen/Löschen von Dateien), Abstraktion der Hardware (Festplatte)
 - Verwaltung von Systemressourcen

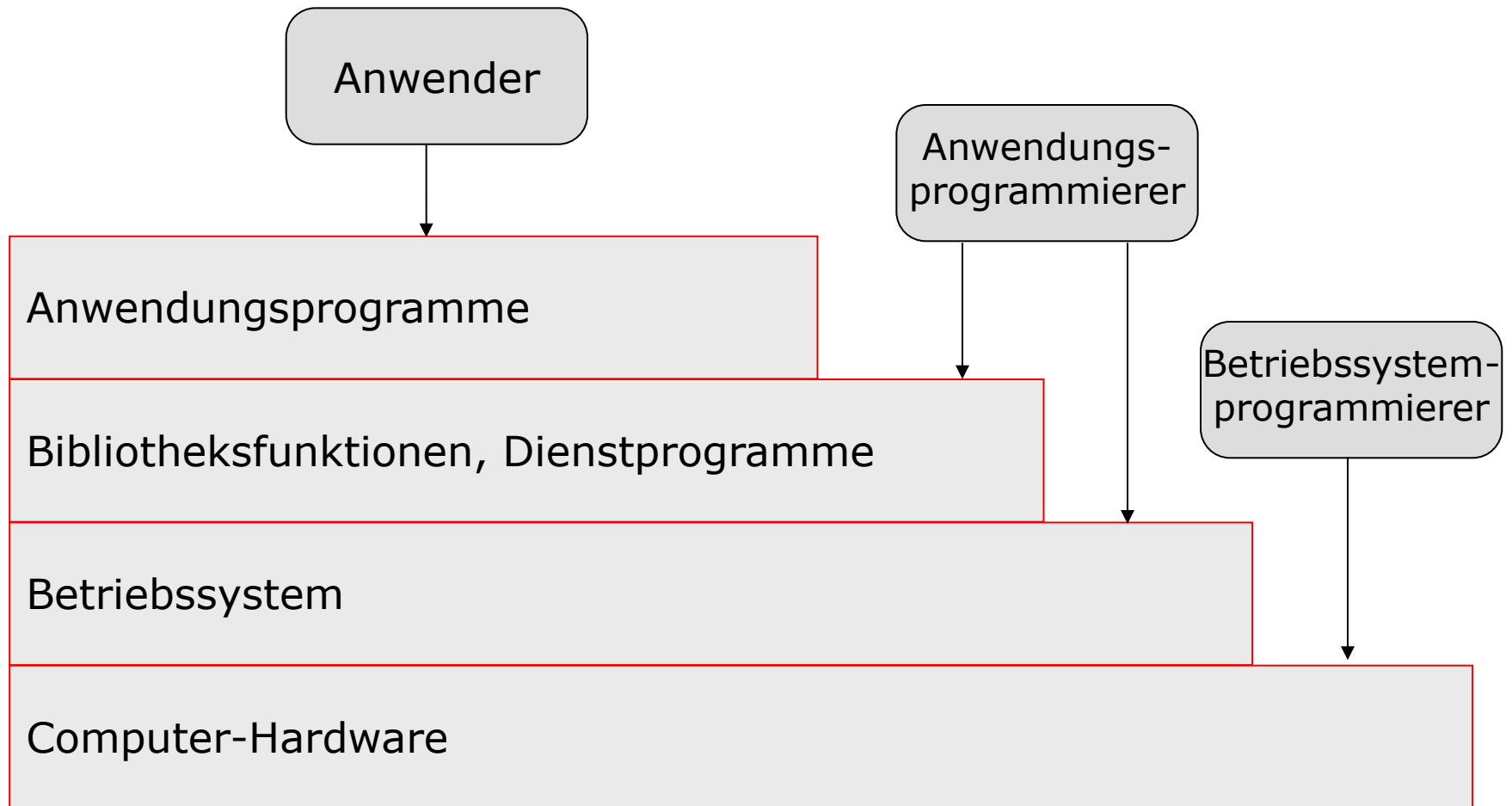
Bereitstellen einer erweiterten Maschine

- **Ziel:** Abstraktion von detailliertem Verhalten des zugrunde liegenden Rechners auf möglichst hohem Niveau
- **Verstecken realer Hardware-Eigenschaften** vor dem Benutzer

Abstrakte Schnittstelle

- Das Betriebssystem stellt dem Programmierer eine **abstrakte Programmierschnittstelle** zur Verfügung
- Das Betriebssystem bietet einen Satz von Kommandos (Systemaufrufe), über die z.B. auf Ein-/Ausgabegeräte zugegriffen werden kann
- **Komfortabel für den Programmierer**

Betriebssystem als Schnittstelle



Dienstprogramme, Anwendungsprogramme

- Schnittstelle zwischen Betriebssystem und Dienstprogrammen nicht immer klar definiert
- Typische **Dienstprogramme**:
 - Compiler
 - Editoren
 - Kommandointerpreter (sog. Shell)
- Beispiele für **Anwendungsprogramme**:
 - Textverarbeitung, Tabellenkalkulation, Datenbankanwendungen

Verwaltung von Systemressourcen

- **Ziel:** Verwaltung aller Bestandteile eines komplexen Systems (Betriebsmittel)
- Beispiele: Prozessoren, Speicher, Platten, Netzwerkschnittstellen, Drucker etc.
- Betriebssystemaufgabe: Geordnete und kontrollierte **Zuteilung der Betriebsmittel** an konkurrierende Prozesse / Benutzer

Betriebssystem als Ressourcenmanager (1)

- „Gerechte“ Zuteilung von gemeinsam genutzten Betriebsmitteln
- Auflösung von Konflikten bei der Betriebsmittelanforderung
- Schutz verschiedener Benutzer gegeneinander (z.B. Zugriffskontrolle bei Dateien)
- Effiziente Verwaltung von Betriebsmitteln

Betriebssystem als Ressourcenmanager (2)

- Fehlererkennung, Fehlerbehandlung
 - Hardware: Gerätefehler
 - Software: Programmfehler
- Ressourcenverwaltung in zwei Dimensionen:
 - **Zeit**: Verschiedene Benutzer erhalten Betriebsmittel nacheinander
 - **Raum**: Verschiedene Benutzer erhalten verschiedene Teile einer Ressource (z.B. Hauptspeicher)

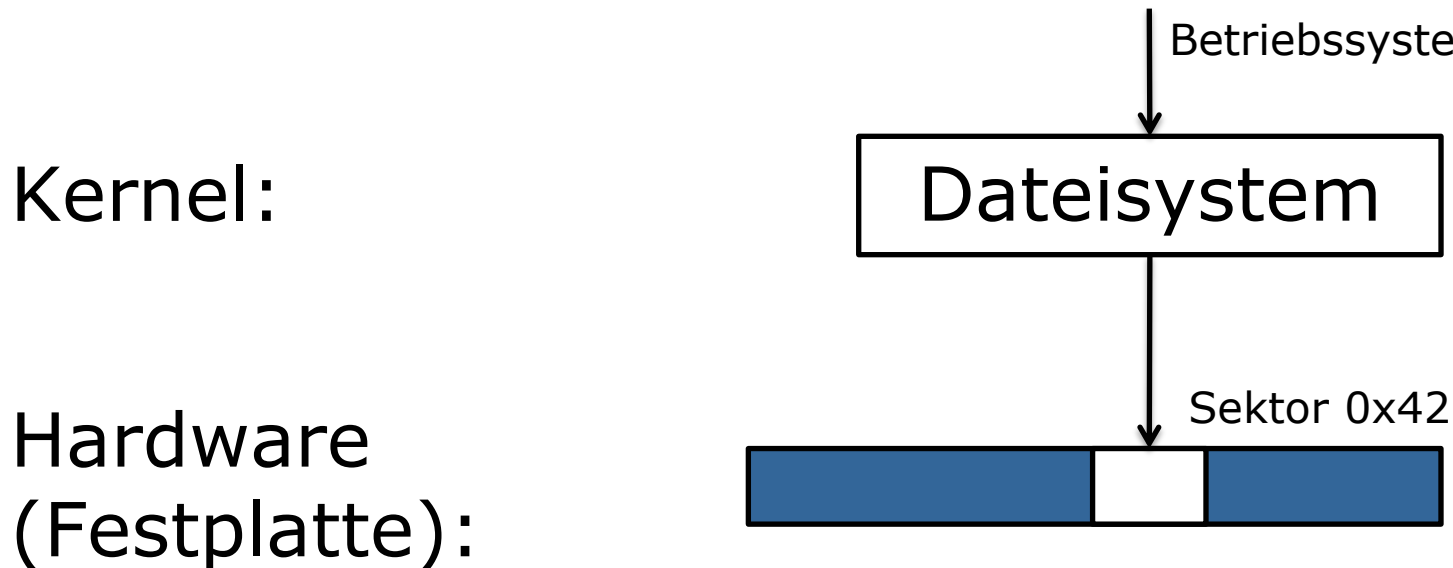
Der Kernel

- Der Kernel (Kern) des Betriebssystems wird in einem privilegierten Modus ausgeführt
 - Zugriff auf Ein- und Ausgabegeräte
 - Speicherschutz veränderbar
- Benutzerprogramme müssen Systemaufrufe (system calls) ausführen, um diese Dienste zu nutzen
 - Mit dem Befehl `TRAP` wird vom Benutzer- in den Kernmodus umgeschaltet
 - Nach Abarbeiten des Systemaufrufs übergibt das Betriebssystem die Kontrolle zurück an das Benutzerprogramm

Betriebssystemaufruf aus Anwendung

- Anfrage des Benutzers nach Inhalt einer Datei muss vom Betriebssystem beantwortet werden

Öffne Datei `/home/someuser/Documents/systeme1.txt`

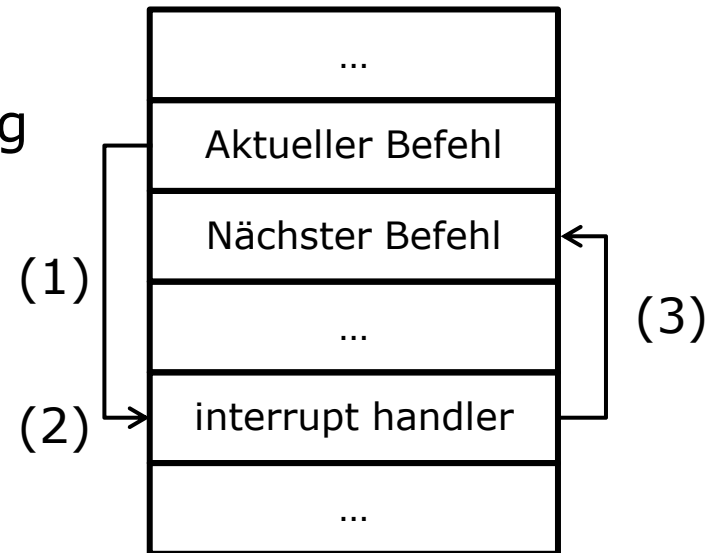


Zugriff auf Ein- und Ausgabegeräte

- Es existieren verschiedene Methoden, auf Ein- und Ausgabegeräte zuzugreifen
 1. „synchroner“ Aufruf: der Systemaufruf fragt in einer Endlosschleife ab, ob die Operation beendet ist ⇒ aktives Warten
 2. Gerät ruft eine Interrupt-Routine auf, so bald es fertig ist
 3. Direct Memory Access über speziellen DMA-Chip

Betriebssystemaufruf durch Interrupts

- Geräte-Controller signalisiert dem Interrupt-Controller eine abgeschlossene Operation
- wird der Interrupt behandelt, so wird
 1. Die aktuelle Programmausführung unterbrochen
 2. Eine Unterbrechungsroutine (interrupt handler) ausgeführt
 3. Die ursprüngliche Programmausführung fortgesetzt



Erweiterbarkeit, Entwicklungsfähigkeit von Betriebssystemen

- **Änderungen** des Betriebssystems erforderlich durch z.B.
 - Neue Hardware
 - Neue Protokolle
 - Korrekturen (z.B. Schließen von Sicherheitslöchern)
- Eigenschaften eines Betriebssystems
 - **Modular** und **klar strukturiert** aufgebaut
 - Gut **dokumentiert**

Historische Entwicklung von Betriebssystemen

Verschiedene Entwicklungsstadien

- Serielle Systeme
- Einfache Stapelverarbeitungssysteme
- Mehrprogrammfähige Stapelverarbeitungssysteme
- Timesharing-Systeme
- Systeme mit graphischen Benutzeroberflächen
- Netzwerkbetriebssysteme
- Verteilte Betriebssysteme

Historische Entwicklung von Betriebssystemen (2)

Einfache Stapelverarbeitungssysteme (1955, IBM)



Historische Entwicklung von Betriebssystemen (2)

Einfache Stapelverarbeitungssysteme (1955, IBM)

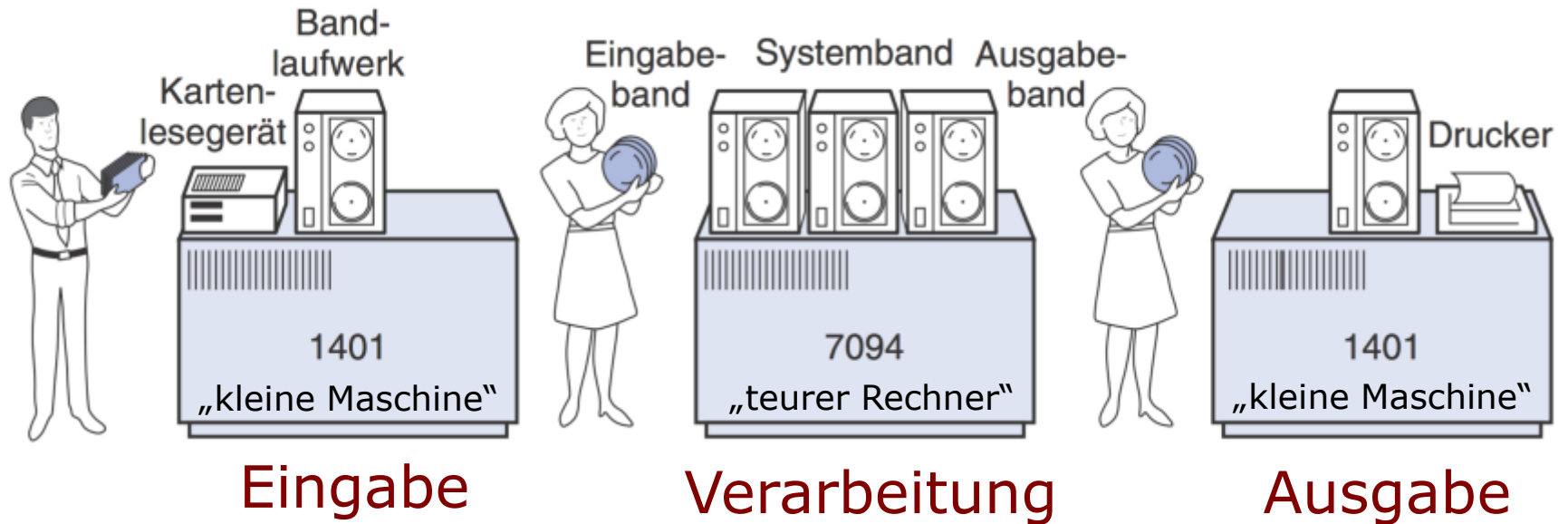
- Unterscheidung von Programmentwicklern und Operateuren, welche die Rechner („Mainframes“) betrieben
- Entwicklung auf Papier (in FORTRAN, Assembler), Stanzen auf Lochkarten
- Sammlung von Programmen (Jobs) auf Lochkarten
- Einlesen der gesammelten Jobs durch kleinen Rechner, Speichern auf Band

Historische Entwicklung von Betriebssystemen (2)

Einfache Stapelverarbeitungssysteme (1955, IBM)

- Abarbeitung des Bandes durch Hauptrechner
 - Einlesen des ersten Jobs, Ausführen des ersten Jobs
 - Ausgaben auf zweites Band
 - Dann weiter mit zweitem Job etc.
- **Serielle Abarbeitung** der Jobs gesteuert durch kleines Softwareprogramm, genannt **Monitor**
- Ausgabe der Ergebnisse für Programmierer durch kleinen Rechner

Historische Entwicklung von Betriebssystemen (2)



Historische Entwicklung von Betriebssystemen (3)

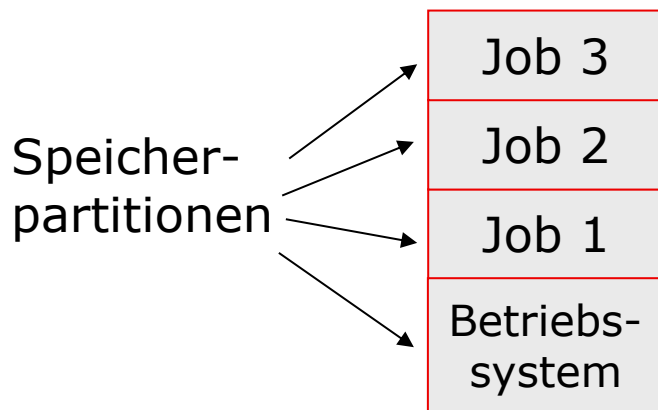
Mehrprogrammfähige Stapelverarbeitungssysteme (ab 1965)

- Verarbeitung durch einzigen Rechner (IBM 360 sowie Nachfolger)
- **Spooling**
 - Einlesen von Jobs auf Lochkarten, danach Speichern auf Platte
 - Nach Beenden eines Jobs: Laden eines neuen Jobs von Platte
- Mehrprogrammfähigkeit bzw. **Multi-programmierung**, um Wartezeiten bei E/A zu reduzieren

Historische Entwicklung von Betriebssystemen (3)

Mehrprogrammfähige Stapelverarbeitungssysteme (ab 1965)

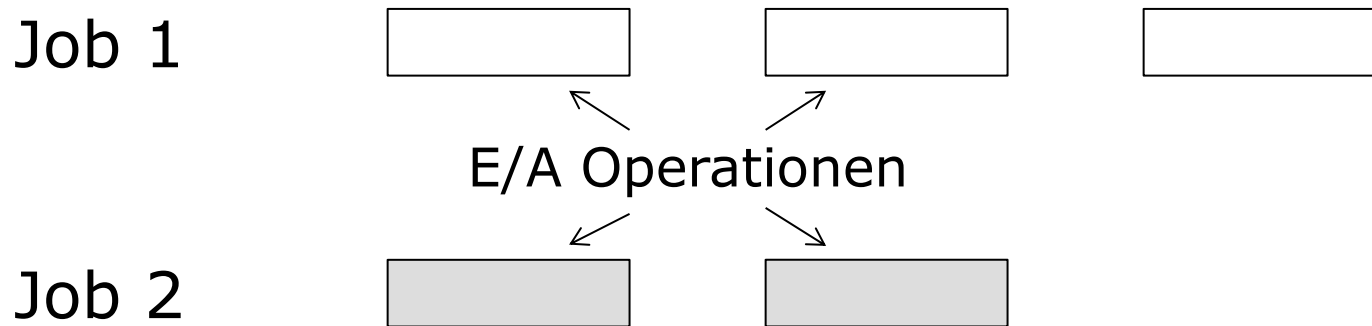
- Viel Rechenzeit wird verschwendet durch Warten der CPU auf Beendigung von E/A Operationen
- Führe aus Effizienzgründen Jobs nicht streng sequentiell aus



- Aufteilung des Speichers in mehrere Bereiche
- Eigene Partition pro aktivem Job
- Wartezeiten auf Beendigung von E/A-Operationen genutzt durch Rechenzeit für andere Jobs

Historische Entwicklung von Betriebssystemen (3)

Mehrprogrammfähige Stapelverarbeitungssysteme (ab 1965)



Multiprogrammierung: Job 1 und Job 2 verschränkt



Historische Entwicklung von Betriebssystemen (4)

Timesharing-Systeme (ab Mitte 60er)

- Bisheriger Nachteil: Kein interaktives Arbeiten mehrerer Benutzer möglich
- Timesharing-Systeme: Online-Zugang zum System für alle Benutzer
- **Idee:** Interaktives Arbeiten eines Benutzers erfordert nicht die komplette Rechenzeit eines Rechners
- Bei **schnellem Umschalten** bemerkt der Einzelnutzer nicht, dass er die Maschine nicht für sich allein hat

Historische Entwicklung von Betriebssystemen (4)

Vergleich

Stapelverarbeitung	Timesharing
Maximale Prozessornutzung (Betreiberwunsch)	Minimale Antwortzeit (Benutzerwunsch)
Befehle in Jobsteuersprache	Interaktive Kommandos

Historische Entwicklung von Betriebssystemen (5)

Systeme mit grafischen Benutzeroberflächen (80(60)er Jahre)

- GUI (Graphical User Interface): Fenster, Icons, Menüs, Mauszeiger
- Zuerst übernommen durch Apple Macintosh
- Später durch Microsoft Windows
 - 1985-1995: Grafische Umgebung, aufsetzend auf MS-DOS
 - ab Windows 95: Betriebssystem und GUI stark miteinander verschränkt

Historische Entwicklung von Betriebssystemen (5)

Systeme mit grafischen Benutzeroberflächen

Unix / LINUX:

- GUI als Aufsatz auf das Betriebssystem
- X-Window-System: Grundlegende Funktionen zur Fensterverwaltung
- Komplette GUI-Umgebungen basierend auf X-Window: z.B. KDE, GNOME

Historische Entwicklung von Betriebssystemen (6)

Netzwerkbetriebssysteme (Mitte 80er)

- Benutzer kennt mehrere vernetzte Rechner
- Einloggen auf entfernten Rechnern möglich
- Datenaustausch möglich
- Auf Einzelrechnern: Lokales Betriebssystem, lokale Benutzer
- Netzwerkbetriebssystem = „normales Betriebssystem mit zusätzlichen Fähigkeiten“

Historische Entwicklung von Betriebssystemen (7)

Verteilte Betriebssysteme

- Mehrere vernetzte Rechner
- Erscheinen dem Benutzer wie Einprozessorsystem
- Datenspeicherung und Programmausführung **verteilt** auf mehreren Rechnern
- Verwaltung automatisch und effizient durch Betriebssystem
- **Probleme**: Nachrichtenverzögerungen, Dateninkonsistenz

Historische Entwicklung von Betriebssystemen (8)

Aktuell

- Betriebssysteme für Mehrkern-Prozessoren
- Aufteilung der Prozesse auf vorhandene Kerne
- Eigene Recheneinheiten, Zugriff auf gemeinsame Ressourcen
- Theoretisch n -fache Rechenleistung bei n Kernen (abhängig von der Parallelisierung der Software)

Arten von Betriebssystemen (1)

Mainframe-Betriebssysteme

- Betriebssysteme für Großrechner
- Einsatz: Webserver, E-Commerce, Business-to-Business
- Viele Prozesse gleichzeitig mit hohem Bedarf an schneller Ein-/Ausgabe
- Sehr hohe Ein-/Ausgabebandbreite
- Beispiel: IBM OS/390, z/OS

Arten von Betriebssystemen (1)

Mainframe-Betriebssysteme

- Drei Arten der Prozessverwaltung:
 - **Batch-Verfahren/Stapelverarbeitung**: Erledigung von Routineaufgaben ohne Benutzerinteraktion (Schadensmeldungen, Verkaufsberichte)
 - **Transaktionsverfahren/Dialogverarbeitung**: Große Anzahl kleiner Aufgaben von vielen Nutzern (Überweisungen, Flugbuchungen)
 - **Timesharing**: Quasi-parallele Durchführung vieler Aufgaben durch mehrere Benutzer (Anfragen an Datenbank)

Arten von Betriebssystemen (2)

Server-Betriebssysteme

- Betriebssysteme für sehr große PCs, Workstations oder auch Großrechner
- Einsatz: z.B. Internetanbieter
- Viele Benutzer gleichzeitig über Netzwerk bedienen
- Zuteilung von Hard- und Softwareressourcen
- Beispiele: NetBSD (Unix), Windows Server

Arten von Betriebssystemen (3)

PC-Betriebssysteme

- Betriebssysteme für Personalcomputer
- Meist nur 1 Benutzer (oder wenige über Netzwerk)
- Einsatz: Programmierung, Textverarbeitung, Spiele, Internetzugriff, ...
- Mehrere Programme pro Benutzer quasi-parallel
- Aufteilung der Prozesse auf vorhandene Kerne
- Zuteilung der Systemressourcen
- Beispiele: Linux, Windows, Mac OS X

Arten von Betriebssystemen (4)

Echtzeit-Betriebssysteme

- Einhalten harter Zeitbedingungen (vs. im Durchschnitt schnell)
- Einsatz: z.B. Betriebssysteme zur Steuerung maschineller Fertigungsanlagen (z.B. Autos)
- Aktion in einem fest vorgegebenen Zeitintervall (in **jedem** Fall, garantierte Deadlines)
- Beispiele: VxWorks, OSEK-OS

Arten von Betriebssystemen (5)

Betriebssysteme für Eingebettete Systeme

- Eingebettete Systeme = „Computer, die man nicht unmittelbar sieht“
- Einsatz: Fernseher, Mikrowelle, Mobiltelefon, Auto, ...
- Meist Echtzeitanforderungen
- Wenig Ressourcen:
 - Kleiner Arbeitsspeicher
 - Geringer Stromverbrauch
- Beispiele: QNX, Windows CE, Windows Phone, iOS, Android

Zusammenfassung

- Betriebssystem = Softwareprogramm
- Abstrakte Schnittstelle zum Rechner
- Verwaltet Systemressourcen
- Historische Entwicklung in mehreren Stadien
- Verschiedene Arten von Betriebssystemen aufgrund verschiedener Anforderungen in unterschiedlichen Anwendungsgebieten
- Moderne Betriebssysteme: Timesharing-Systeme mit Mehrprogrammbetrieb (plus zusätzliche Eigenschaften)