

Übersicht: Einführung

- ✱ Was ist ein Betriebssystem?
- ✱ Großrechnersysteme
- ✱ Mehrprozessorsysteme
- ✱ Arbeitsplatzrechnersysteme
- ✱ Verteilte Systeme
- ✱ Cluster-Systeme
- ✱ Echtzeitsysteme
- ✱ Handheld-Systeme/Palmtop-Systeme
- ✱ Formen der Datenverarbeitung

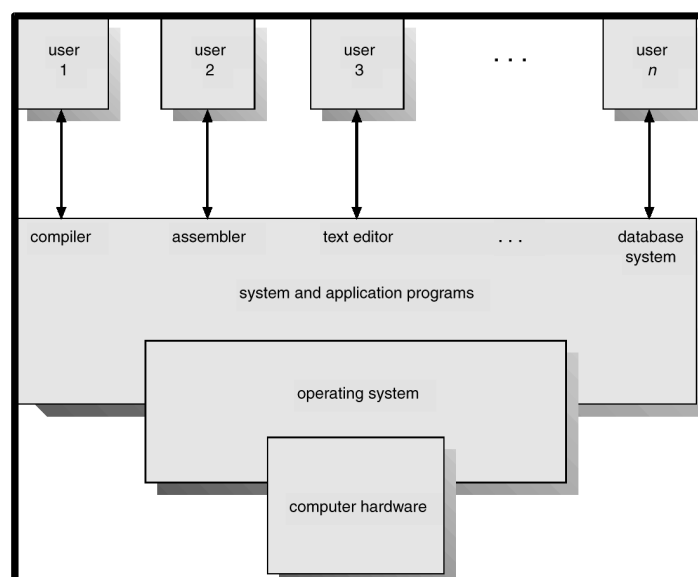
Was ist ein Betriebssystem?

- ✱ Ein Programm, das als Vermittler zwischen dem Anwender und der Computer-Hardware vermittelt.
- ✱ Ziele von Betriebssystemen:
 - ◆ Ausführung von Anwenderprogrammen
 - ◆ Vereinfachung der Lösung von Anwenderproblemen
 - ◆ Leichte Bedienbarkeit des Computersystems
 - ◆ Effiziente Nutzung der Computer-Hardware

Bestandteile von Computersystemen

1. **Hardware** – Bereitstellung grundlegender Betriebsmittel (Prozessor, Speicher, Ein-/Ausgabegeräte)
- ✱ **Betriebssystem** – steuert und koordiniert die Nutzung der Betriebsmittel für die verschiedenen Anwendungsprogramme der verschiedenen Anwender
- ✱ **Anwendungsprogramme** – definieren, wie das zu bearbeitende Problem mit Hilfe der Betriebsmittel gelöst wird (Compiler, Datenbanksysteme, Textverarbeitung, Spiele usw.)
4. **Anwender** (Menschen, Maschinen, andere Computer)

Abstrakte Sicht der Bestandteile



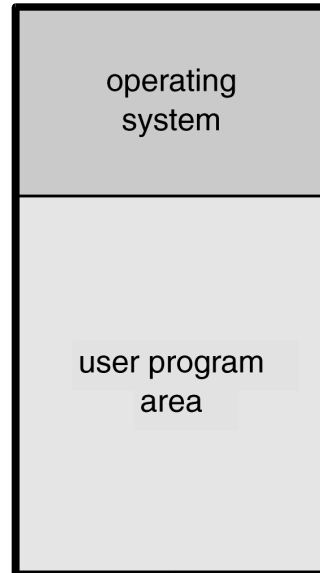
Definitionen von Betriebssystemen

- ✱ **Betriebsmittelzuteiler** – verwaltet Betriebsmittel und teilt diese zu
- ✱ **Steuerprogramm** – steuert die Ausführung der Anwendungsprogramme und der Operationen der Ein-/Ausgabegeräte
- ✱ **Kernel** – das Programm, das immer läuft (alle anderen Programme sind damit Anwendungsprogramme)

Großrechnersysteme

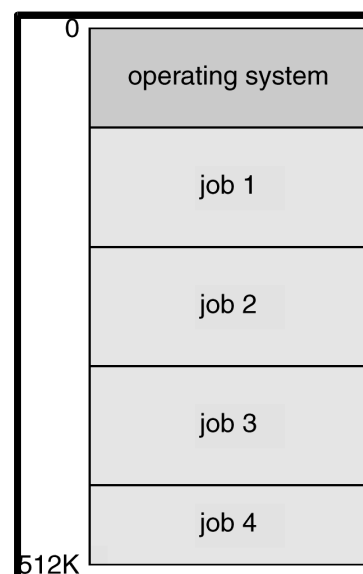
- ✱ **Reduzierung der Einrichtezeit** durch “Bündeln” (batching) ähnlicher Aufträge (jobs)
- ✱ **Automatische Ablaufsteuerung der Aufträge** – automatische Übertragung der Kontrolle von einem Auftrag zum nächsten. Erstes rudimentäres Betriebssystem.
- ✱ **Speicherresidenter Monitor**
 - ◆ Kontrolle liegt zu Beginn beim Monitor
 - ◆ Kontrolle wird an einen Auftrag abgegeben
 - ◆ Nach Beendigung des Auftrags erhält der Monitor die Kontrolle zurück

Speicheraufteilung eines einfachen Stapelverarbeitungssystems



Stapelverarbeitung im Mehrprogrammbetrieb

Mehrere Aufträge werden im Speicher gehalten. Der Prozessor wechselt zwischen diesen Aufträgen hin und her.



Benötigte Betriebssystemfähigkeiten beim Mehrprogrammbetrieb

- ✧ Bereitstellung von Ein-/Ausgabe-Routinen
- ✧ Zuordnung von Geräten zu Aufträgen
- ✧ Speicherverwaltung – das Betriebssystem muss den verschiedenen Aufträgen Speicher zuordnen
- ✧ Prozessor-Scheduling – das Betriebssystem muss zwischen den verschiedenen, ausführungsbereiten Aufträgen auswählen
- ✧ Schutz vor Programmfehlfunktionen (Übergriffen eines Auftrags auf einen anderen, Endlosschleifen usw.)

Mehrbenutzersystem (Time-Sharing Systems) – Interaktive Benutzung

- ✧ Eine interaktive Kommunikationsmöglichkeit zwischen dem Anwender und der Computersystem wird bereitgestellt, die den Zugriff auf Programme und Daten erlaubt. Nach der Abarbeitung eines Kommandos wird das nächste Benutzerkommando erwartet.
- ✧ Der Prozessor wird in schneller Abfolge zwischen verschiedenen Aufträgen, die sich im Speicher und auf Festplatte befinden, hin und her geschaltet. (Nur Aufträge im Speicher erhalten den Prozessor.)
- ✧ Ein Auftrag wird in den Hauptspeicher ein- oder auf Festplatte ausgelagert.

Arbeitsplatzrechnersysteme / Personal-Computer

- ✱ Personal-Computer – Computersysteme, die ausschließlich einem einzigen Benutzer zur Verfügung stehen
- ✱ Ein-/Ausgabegeräte – Tastatur, Maus, Monitor, kleiner Drucker, ...
- ✱ Komfortable Bedienung, schnelle Reaktionszeit
- ✱ Konzepte größerer Betriebssysteme können verwendet werden (z.B. Time-Sharing). Andere Aspekte u.U. weniger wichtig (z.B. Prozessor-Auslastung). Ausführung verschiedener Betriebssysteme möglich (Windows, MacOS, UNIX, Linux)

Mehrprozessorsysteme I

- ✱ Mehrprozessorsysteme besitzen mehrere, eng gekoppelte Prozessoren
- ✱ *Eng gekoppelt* – Prozessoren nutzen gemeinsam Hauptspeicher und Systemtakt. Die Kommunikation zwischen den Prozessoren findet üblicherweise über den gemeinsam genutzten Speicher statt.
- ✱ Vorteile von Mehrprozessorsystemen:
 - ◆ Erhöhter Durchsatz
 - ◆ Verbessertes Preis/Leistungsverhältnis
 - ◆ Höhere Zuverlässigkeit
 - stufenweiser Leistungsverlust (graceful degradation)
 - Ausfallsicherheit (fail-soft systems)

Mehrprozessorsysteme II

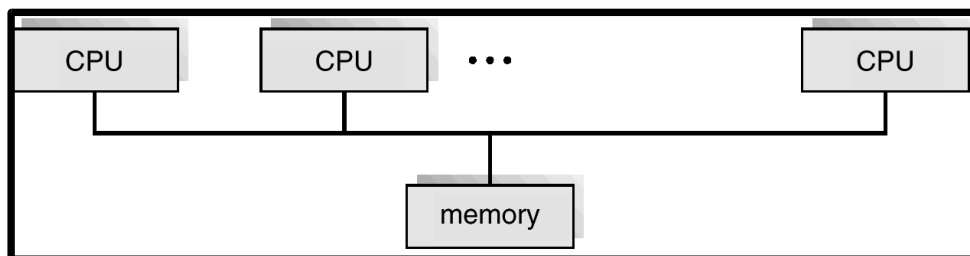
✦ Symmetric multiprocessing (SMP)

- Auf jedem Prozessor läuft das identische Betriebssystem
- Mehrere Prozesse können ohne Leistungsverlust ablaufen
- Die meisten modernen Betriebssysteme unterstützen SMP

✦ Asymmetric multiprocessing

- Jeder Prozessor hat eine spezielle Aufgabe. Ein Master-Prozessor verteilt Aufgaben an die anderen (möglicherweise spezialisierten) Slave-Prozessoren.
- Eher verbreitet bei sehr großen Systemen

Architektur bei symmetrischen Mehrprozessorsystemen



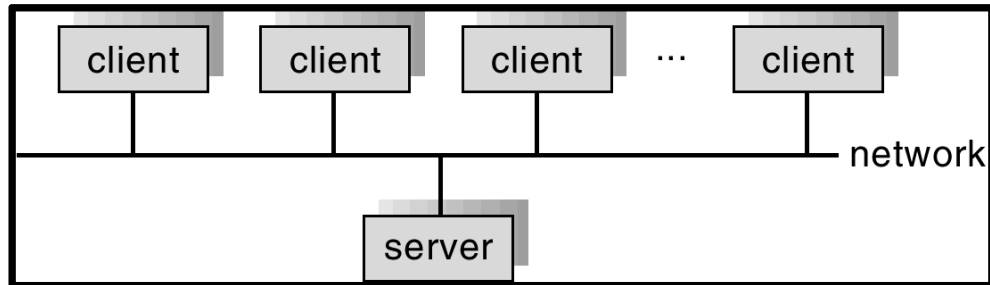
Verteilte Systeme I

- ✧ Verteile die Aufgabe auf verschiedene, lose gekoppelte Prozessoren
- ✧ *Lose gekoppelt* – jeder Prozessor hat seinen eigenen Hauptspeicher und Systemtakt. Die Prozessoren kommunizieren über unterschiedliche Kommunikationsleitungen (z.B. Hochgeschwindigkeitsbusse, Netzwerkleitungen)
- ✧ Vorteile von verteilten Systemen:
 - ◆ Gemeinsame Nutzung von Ressourcen
 - ◆ Erhöhte Rechenleistung – Verteilung von Teilaufgaben
 - ◆ Höhere Zuverlässigkeit

Verteilte Systeme II

- ✧ Voraussetzung: Netzwerk-Infrastruktur
- ✧ Verwendung von Local-Area-Networks (LAN) oder Wide-Area-Networks (WAN) möglich
- ✧ Client-Server-Systeme oder Peer-to-Peer-Systeme

Allgemeine Struktur von Client-Server-Systemen



Cluster-Systeme

- ✱ Beim Clustering werden mehrere Systeme zusammengeschaltet, um gemeinsam eine übergeordnete Aufgabe zu erfüllen. Häufig greifen dabei die beteiligten Systeme auf ein gemeinsames Speichermedium zu.
- ✱ Vorteile: Leistungssteigerung, erhöhte Erreichbarkeit
- ✱ *Asymmetrisches Clustering*: Die verschiedenen Systeme erfüllen unterschiedliche Aufgaben. (Beispiel: Ein Server führt die Anwendung aus, ein anderer überwacht den ersten und übernimmt bei einem Ausfall.)
- ✱ *Symmetrisches Clustering*: Die verschiedenen Systeme erfüllen eine gleichartige Aufgabe. (Beispiel: Mehrere Systeme bearbeiten Anfragen an einen Webserver.)

Echtzeitsysteme I

- ✱ Systeme mit klar definierten zeitlichen Rahmenbedingungen
- ✱ Häufiger Einsatz zur Steuerung dedizierter Anwendungen (z.B. Steuerung von Fahrzeugen, Robotern, medizinischen Instrumenten)
- ✱ Echtzeitsysteme können in Hard- oder Software realisiert sein.

Echtzeitsysteme II

- ✱ Hardware-Echtzeitsysteme
 - ◆ Wenig oder kein Sekundärspeicher; Daten werden im Kurzzeitspeicher oder in einem Festspeicher (Read-Only-Memory) gespeichert.
 - ◆ Schwer vereinbar mit Mehrbenutzersystemen, von gängigen Betriebssystemen nicht unterstützt.
- ✱ Software-Echtzeitsysteme
 - ◆ Nützlich für Echtzeit-Anwendungen, die fortgeschrittene Betriebssystemfunktionen benötigen (z.B. Multimedia und Virtual-Reality-Anwendungen).
 - ◆ Nur begrenzte Einsatzmöglichkeit im industriellen Umfeld

Handheld-Systeme/ Palmtop-Systeme

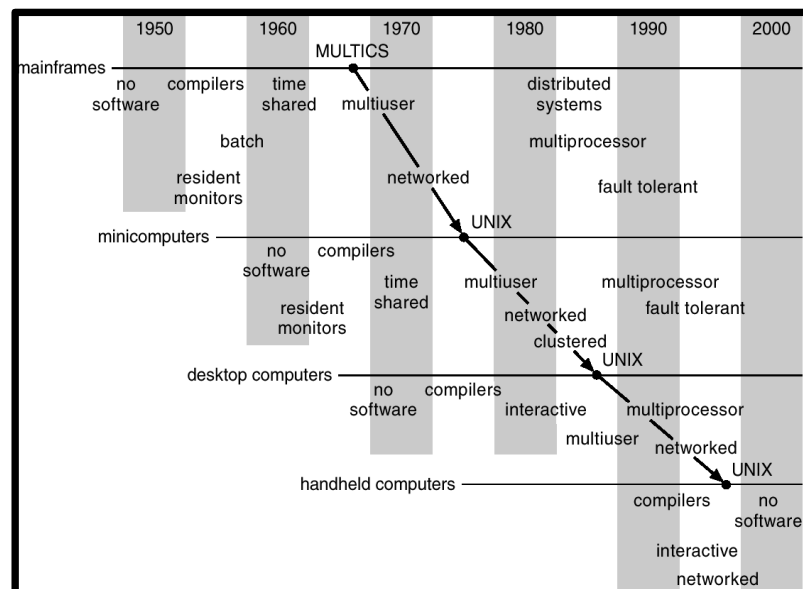
✧ Beispiele für Handheld-Systeme:

- ◆ Personal Digital Assistants (PDAs)
- ◆ Moderne Mobilfunk-Telefone

✧ Besonderheiten:

- ◆ Wenig Speicher
- ◆ Langsame Prozessoren
- ◆ Kleines Display

Migration von Betriebssystemkonzepten und -funktionen



Formen der Datenverarbeitung

- ✱ **Traditionelle Datenverarbeitung:** Einsatz von Computern in Rechenzentren und Büros
- ✱ **Netzwerk-/Web-basierte Datenverarbeitung:** Nutzung von weltweit verfügbaren Daten und Diensten, Einbindung von privaten Rechnern, mobiler Zugang
- ✱ **Embedded Computing:** Spezialisierte Programme/Betriebssysteme in einer Vielzahl von Geräten (Waschmaschinen, Autos, Videorekordern usw.)