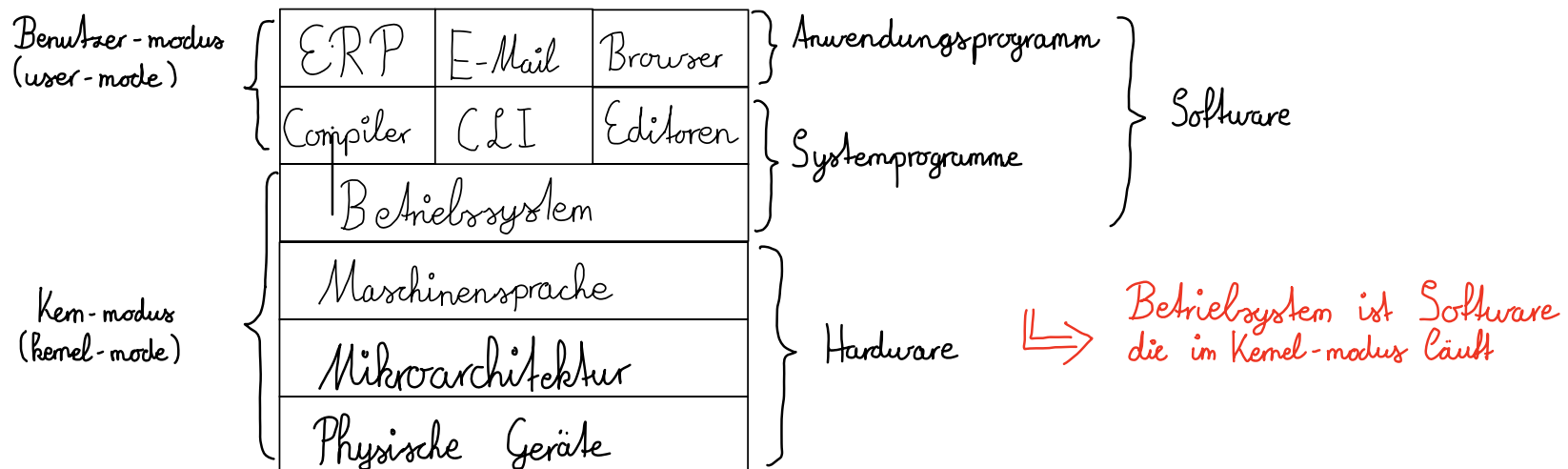


1. Einführung

- einfaches Schichtenmodell:



Aufgaben des Betriebssystems

1. Abstraktion der Hardware

- stellt Systemcalls für Anwendung bereit zum Zugriff auf Hardware

→ Betriebssystem realisiert eine einheitliche Sicht auf die Hardware für Anwendungen

2. Verwalten von Ressourcen

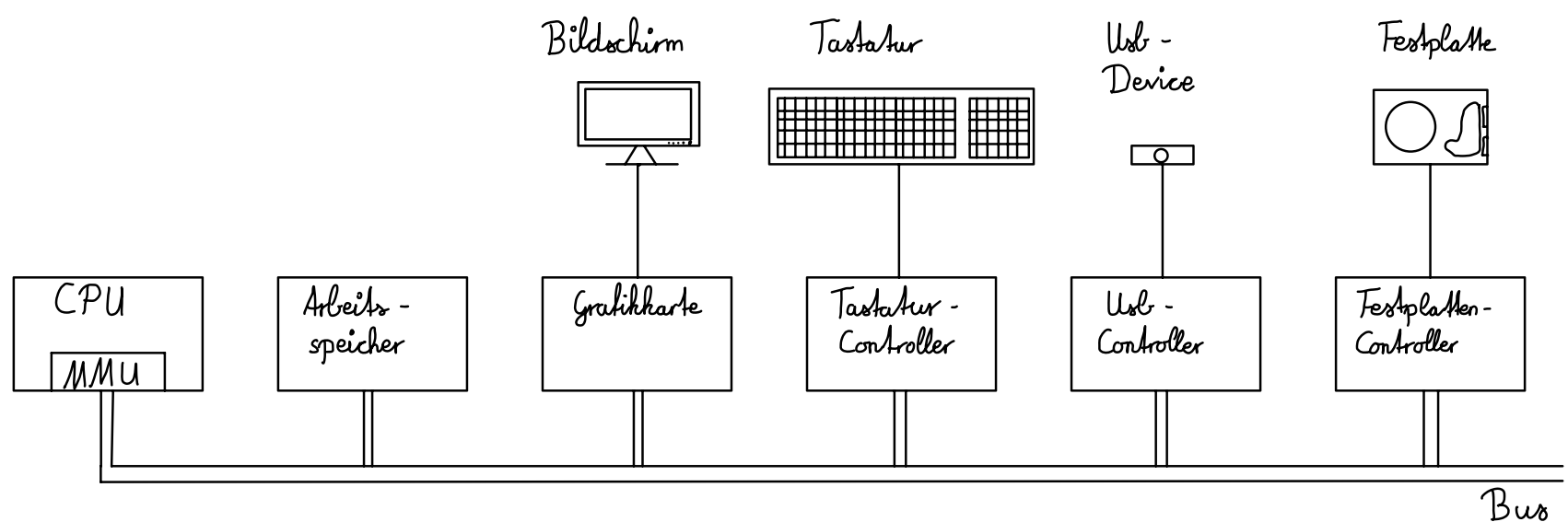
- Anwendungen brauchen Ressourcen um ausgeführt zu werden
- Leistungsfähige Rechner laufen im **Mehrprozess-** und **Mehrbenutzerbetrieb**

→ Betriebssystem verteilt die Ressourcen gerecht und sichert die
Anwendungen und Benutzer gegeneinander

↳ Betriebssystem ist Mittler zwischen Anwendung und Hardware

Anwendungen können somit nicht direkt auf Hardware zugreifen

- einfaches Modell eines PC



Arten von Betriebssysteme

- Großrechner
- Server
- Multiprozessor
- Personalcomputer
- Handheld-Computer
- Eingekerkelte Systeme
 - Auto, Fernseher, MP3-Player,...
 - führt nur vertrauenswürdige Software aus
 - Patchen durch den User nicht möglich
- Sensorknoten
- Echtzeitsysteme
 - Zeit ist essentiell bei Ressourcenvergabe (z.B. Steuerungsanlage)
- Smart Cards / Chipkarten

Überblick Computer-Hardware

Prozessor:

- Befehlsatz abhängig von CPU-Typ
- Laden von Befehlen dauert länger als Ausführen
↳ Optimieren durch Register auf der CPU

Spezialregister:

- Befehlszähler (Program Counter, PC)
 - enthält Speicheradresse des nächsten Befehls
- Kellerregister (stack pointer)
 - zeigt auf das Ende des aktuellen Kellers / Stacks
 - ablegen der Frames für jede angesprungenen und nicht beendete Prozedur
 - Eingabeparameter, lokale Variablen
- Programmstatuswort (PSW)
 - Status-Bits, CPU-Priorität, Modus
- Systemcall als kontrollierter Moduswechsel

E/A-Geräte:

- Integration in Computern durch Controller-Ansatz
- Spezielle Hardware (oft eigene Mikroprozessoren)
 - steuert das Gerät weitgehend autonom
 - kann Interrupts senden
- Geräte-Treiber
 - Software, die mit Controller kommuniziert
 - muss im Kernmodus laufen, also Teil des BS

1. Aktives Warten (Busy-wait) // eher schlecht da CPU belegt ist mit warten

2. Interrupt

3. DMA-Chip (Direct Memory Access)

Aufbau von Betriebssystemen

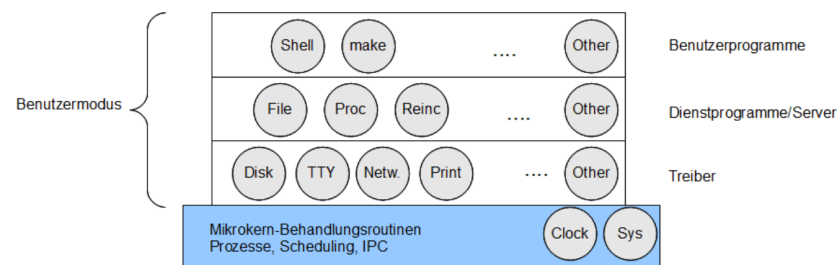
- Monolithische Systeme
 - Häufigste Form
 - Eine große ausführbare Datei
 - jede Funktion ist für andere Funktionen sichtbar

Monolithisch heißt nicht ohne Struktur:

- Hauptprogramm ruft Dienstprozedur auf
- Dienstprozeduren führen Systemaufrufe aus
- Hilfsfunktionen unterstützen Dienstprozeduren

• Mikrokernel :

- Sowenig wie möglich im Kernmodus laufen lassen
 - größere Stabilität, da nur wenige Programme das Gesamtsystem zu Fall bringen können

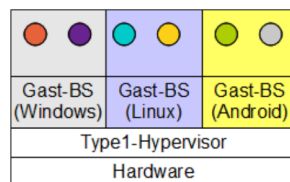


- Reincarnations-Server überprüft Dienstprogramme und startet bei Fehlern die Dienste wieder neu
- Einsatz in Echtzeit, industriellen, avionischen und militärischen Systemen

• Virtuelle Maschinen

– Hypervisor Type 1

- Entspricht VMM



– Hypervisor Type 2

- Teil eines BS
- Gastgebersystem kann noch Programme laufenlassen

