

# Was ist ein Betriebssystem?

„Stellt euch vor, euer Computer ist ein Theater. Die Hardware ist die Bühne, die Schauspieler, die Requisiten. Aber ohne Regisseur läuft nichts sinnvoll ab – das Betriebssystem ist dieser Regisseur.“

Dann erkläre ich in einfachen Punkten:

## Aufgaben eines Betriebssystems:

- **Speicherverwaltung** (RAM, Swap)
- **Prozessverwaltung** (Multitasking, Scheduling)
- **Dateiverwaltung** (Lesen/Schreiben)
- **Geräteverwaltung** (Treiber, I/O)
- **Benutzerverwaltung & Rechte**

## Speicherverwaltung (RAM, Swap)

### Ziel:

Das Betriebssystem muss den **Hauptspeicher (RAM)** sinnvoll auf Prozesse aufteilen und Engpässe durch **Auslagerung (Swap)** überbrücken.

### Aufgaben:

- Zuteilung von RAM an Programme (z. B. wenn du ein Spiel startest, bekommt es z. B. 2 GB RAM zugewiesen)
- Verhindern, dass Programme in fremden Speicherbereich schreiben („Speicherschutz“)
- Auslagerung in die **Swap-Datei/Partition**, wenn RAM voll ist → langsamer, da auf Festplatte

### Beispiel:

Du öffnest 5 Programme gleichzeitig. Der RAM reicht nicht mehr aus. Das Betriebssystem lagert inaktive Programmteile auf die Festplatte (Swap) aus, um den aktiven Programmen weiter RAM zu geben.

## Prozessverwaltung (Multitasking, Scheduling)

### Ziel:

Das OS verwaltet, **welche Prozesse wann und wie lange Rechenzeit bekommen**, auch wenn scheinbar alles gleichzeitig läuft.

### Aufgaben:

- Erzeugen und Beenden von Prozessen
- Zuteilen von **CPU-Zeit** (Scheduling)
- Verwaltung von **Zuständen**: bereit, laufend, wartend etc.
- Prozessprioritäten setzen

### Beispiel:

Du hast Chrome, Spotify und Word geöffnet. Die CPU kann nur einen Prozess gleichzeitig ausführen, aber durch **schnelles Umschalten (Scheduling)** merkst du nichts davon – alles läuft flüssig.

## Dateiverwaltung (Lesen/Schreiben)

### Ziel:

Zugriff auf **Dateien und Ordner auf Speichermedien** ermöglichen, strukturieren und absichern.

### Aufgaben:

- Verwaltung der physischen Speicherblöcke (Sektoren, Cluster)
- Dateioperationen: Öffnen, Lesen, Schreiben, Löschen
- Zugriffskontrolle: Wer darf was lesen oder schreiben?
- Pflege des **Dateisystems** (z. B. NTFS, ext4)

### Beispiel:

Du speicherst ein Word-Dokument. Das Betriebssystem bestimmt **wo genau auf der Festplatte** die Datei liegt und aktualisiert die Dateitabelle.

## **Geräteverwaltung (Treiber, I/O)**

### **Ziel:**

Die Kommunikation zwischen **Hardwaregeräten** (z. B. Drucker, Tastatur, USB-Stick) und Programmen sicherstellen.

### **Aufgaben:**

- Verwendung von **Gerätetreibern** für spezifische Hardware
- Bereitstellung von **Schnittstellen (APIs)** für Programme
- Verwaltung von **Eingabe-/Ausgabeprozessen (I/O)**: z. B. Pufferung, Priorisierung

### **Beispiel:**

Du steckst einen USB-Stick ein. Das Betriebssystem lädt den passenden Treiber, erkennt das Gerät und stellt es als Laufwerk E:\ im Explorer bereit.

## **Benutzerverwaltung & Rechte**

### **Ziel:**

Mehrere Benutzer verwalten und dabei **Zugriffsrechte und Sicherheit** gewährleisten.

### **Aufgaben:**

- Benutzerkonten und Passwörter verwalten
- Gruppenrechte (z. B. Admin, Standardbenutzer)
- Zugriff auf Dateien, Geräte und Programme steuern
- Authentifizierung & Autorisierung

### **Beispiel:**

In Windows kannst du als Admin Software installieren – ein Standardbenutzer kann das nicht. Das verhindert z. B. Schadsoftware-Installation durch unbefugte Nutzer.

## **Zusammengefasst als Merksatz für die Prüfung:**

„Ein Betriebssystem verwaltet **Speicher, Prozesse, Dateien, Geräte und Benutzer** – also alles, was einen Rechner zum Arbeiten bringt.“

## Arten von Betriebssystemen

Ich stelle die verschiedenen Typen gegenüber:

Typ	Beispiele	Eigenschaften
<b>Desktop-OS</b>	Windows, macOS, Linux	GUI, für Endnutzer
<b>Server-OS</b>	Windows Server, Linux, Unix	Headless, stabil, netzwerkfähig
<b>Mobile-OS</b>	Android, iOS	Touch-Oberfläche, ARM-Architektur
<b>Embedded-OS</b>	RTOS, VxWorks, Raspbian	Ressourcenarm, auf Geräte zugeschnitten

## Dateisysteme – was ist das überhaupt?

„Ein Dateisystem ist wie ein Bücherregal mit Inhaltsverzeichnis. Es sagt dem Betriebssystem, **wo und wie** Daten gespeichert werden.“

Dann erkläre ich die Kernaufgaben:

- Strukturierung des Speichers in Dateien und Ordner
- Verwaltung von freien/verwendeten Blöcken
- Zugriffskontrolle und Berechtigungen

## Gängige Dateisysteme im Vergleich

Ich zeige eine Tabelle mit Vergleich:

Dateisystem Plattform		Max. Dateigröße	Besondere Merkmale
FAT32	Win, Linux	4 GB	Kompatibel, aber veraltet
exFAT	Win, Linux, macOS	16 EB	Für USB-Sticks, große Dateien
NTFS	Windows	16 EB	Rechteverwaltung, Journaling
ext4	Linux	1 EB	Journaling, weit verbreitet

Dateisystem Plattform		Max. Dateigröße	Besondere Merkmale
APFS	macOS	8 EB	Snapshots, optimiert für SSDs
XFS, Btrfs	Linux Server	Sehr groß	Moderne Features, Skalierbarkeit