

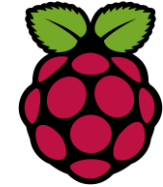
Einführung in die Betriebssysteme

Martin Spörl

Grundlagen

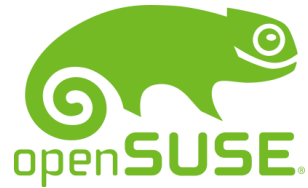
Grundlagen

Betriebssysteme & Distributionen



Mac[™]OS

ubuntu[®]



KALI LINUX



Definition "Betriebssystem"

Welches sind die Hauptaufgaben eines Betriebssystems?

Rechnersystem zugänglich machen

- Komplexe, unterschiedliche Hardware vereinheitlichen
- Kommunikation steuern (Netzwerk)
- Hardware schützen
- Fehler „abfangen“



abstrakte / virtuelle Maschine

Ressourcen verwalten

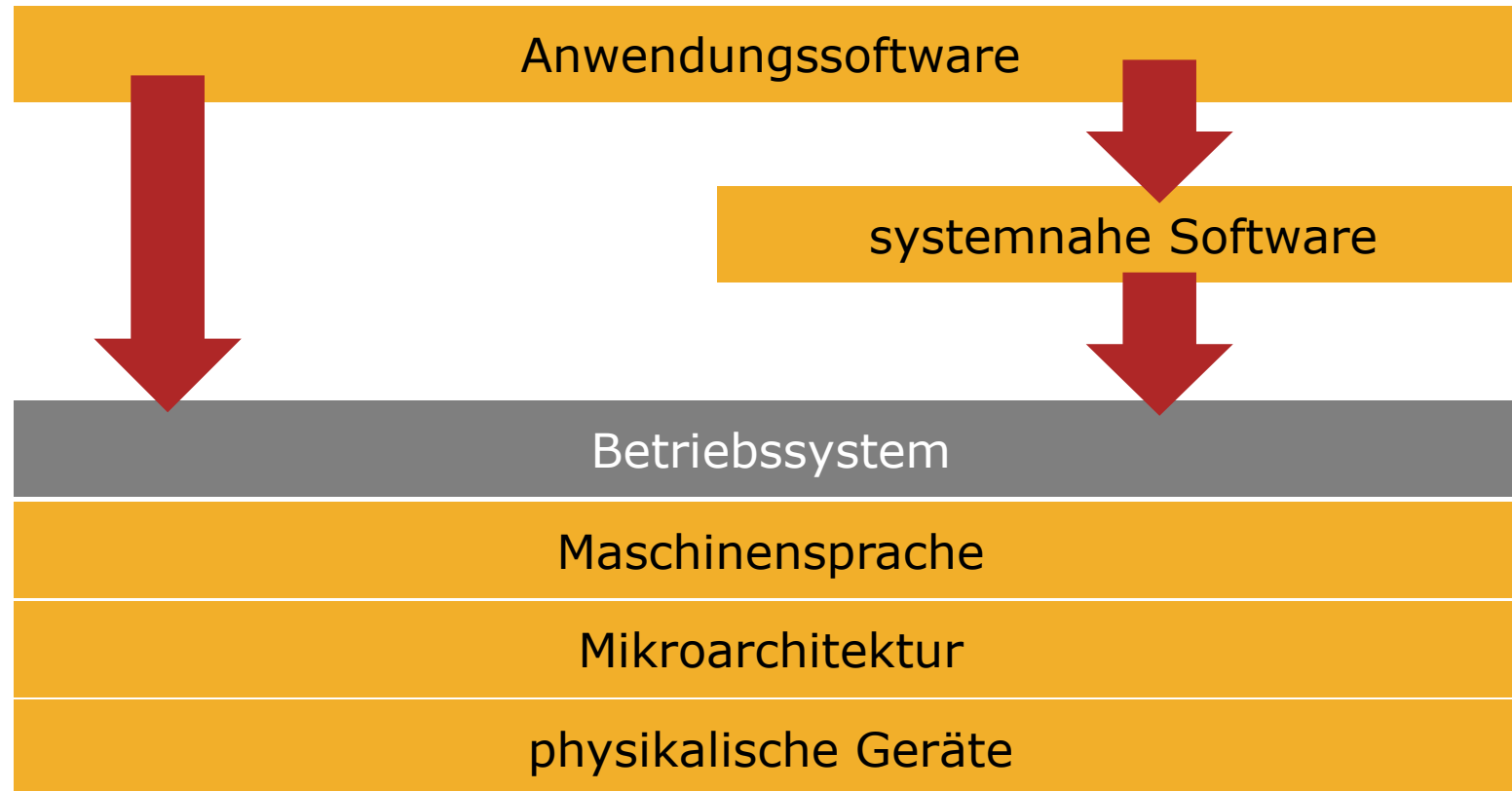
- Einheitlicher & geschützter Zugriff auf Arbeitsspeicher
- Zugriff auf Festplatten
- Aufgaben des Prozessors verwalten



Ressourcenmanager

Definition "Betriebssystem"

Einordnung in das Rechnersystem



Arten von Betriebssystemen I

Klassifizierung nach Anforderungen

Nutzeranzahl

- Single-User-OS
- Multiuser-OS

Anzahl unabhängiger Tasks

- Single-Task-OS
- Multitasking-OS

Kommunikation

- OS für Stapelverarbeitung
- Interaktives OS
- Autonomes OS

Verteilung

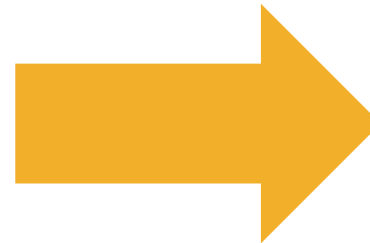
- lokales OS
- verteiltes OS

Arten von Betriebssystemen II

Klassifizierung nach Anforderungen

Zielarchitektur Bedienkonzept

- Embedded OS
 - Mobile OS
 - Echtzeit OS
 - Mainframe OS
 - PC OS
 - Server OS
- Grafisches OS
 - Konsolen OS



Unterschiedliche
Anwendungsfälle
erfordern
unterschiedliche
Betriebssysteme

Geschichte der Betriebssysteme

1945: Z4 „erster kommerziell nutzbarer Computer“

1954: erste Programmiersprache „Fortran“

Bis 1964: Hardware mit dediziertem OS

1964: IBM liefert mit OS/360 erstes modell-übergreifendes OS

Anfang 1970er: Xerox entwickelt erste GUI

Ende 1970er: erste Heimcomputer u. a. mit UI (Commodore, C64, AmigaOS)

1980: erste Festplatten erscheinen und ermöglichen Random Access

1984: macOS erscheint als erstes kommerzielles Betriebssystem

1991: Geburtsjahr von Linux

1992: Windows 3.1 erscheint = Durchbruch für UI auf Computern

1993: Windows NT 3.1 für die HAL ein

2008: Google veröffentlicht Android

Exkurs: Komplexität von OS

- 1996 - Windows NT 4.0
 - ca. 8 Mio. Codezeilen
- 1999 - Linux 2.2
 - ca. 1 Mio. Codezeilen
- 2000 - Windows NT 5.0 („Windows 2000“)
 - ca. 30 Mio. Codezeilen
 - Viele Sicherheits- & Netzwerkdienste ergänzt
- 2001 - Windows NT 5.1 („Windows XP“)
 - ca. 40 Mio. Codezeilen
 - höhere Grafik-Anforderungen, daher Sprunghafter anstieg
- 2003 - Linux 2.6
 - ca. 8 Mio. Codezeilen
- 2005 - MacOS X 10.4
 - Ca. 86 Mio. Codezeilen
- 2007 - Windows Vista
 - ca. 50 Mio. Codezeilen
- 2009 - Android
 - ca. 12 Mio. Codezeilen
- 2009 - Windows 7
 - ca. 40 Mio.
 - Bereinigung des Codes
- 2015 - Linux 4.2
 - ca. 20 Mio. Codezeilen

Zum Vergleich: Google
Suchdienst hat ca. 2 Mrd.
Codezeilen (ca. 86TB)

Exkurs: UNIX

Die größte OS Familie?

- In den 1970er entwickelt
- Zusammen mit C entwickelt
- War ursprünglich ein von AT&T (damals noch Bell Laboratories) entwickeltes OS
- Basis für sehr viele Betriebssysteme
- Damit das am einflussreichste & erfolgreichste OS der Geschichte!
- Heutiger Standard „POSIX“
- Bekannteste Ableger
 - Linux
 - HP-UX
 - (Free-)BSD

Architekturen I

Monolithisches System

- Alles redet mit allem
- Kein Schutz für Informationen
- Typisch für historische Betriebssysteme
- Sehr effizient
- Beispiel: Linux

Geschichtetes System

- Kommunikation nur zwischen benachbarten Schichten
- Kann ineffizient werden
- In der Praxis kaum / gar nicht anzufinden

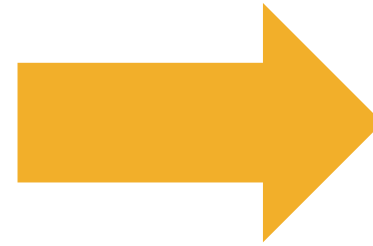
Mikroarchitektur

- Funktionalität wird außerhalb des Kernels ausgelagert
- Kern verwaltet Kommunikation zwischen Client & Server
- Beispiel: Mac OS X

Architekturen II

Modulares System

- Zentraler Kern mit verschiedenen ladbaren Modulen
- Module können andere Module nachladen
- Objektorientierter Ansatz
- klare Schnittstelle zwischen einzelnen Modulen
- Schichtprinzip um grundlegende Module „nach oben hin“ zu erweitern



Mischung der Architekturen erweist sich als effizientestes System