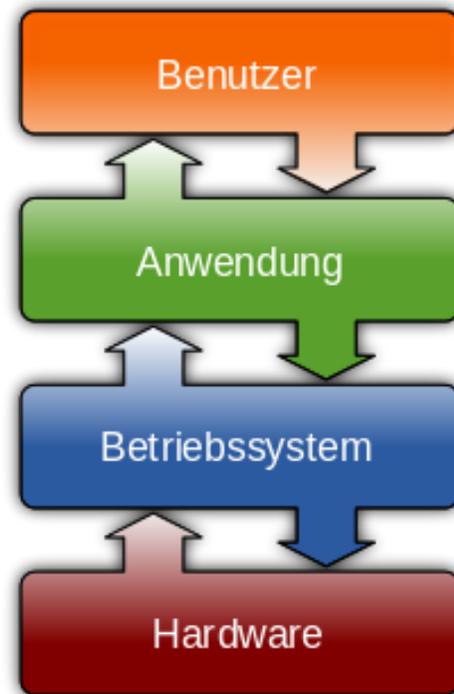


Hardware und Software eines Computers



[Wikipedia]

Was ist ein Computer?

Ein Computer oder Rechner ist ein Apparat, der Daten mithilfe einer programmierbaren Rechenvorschrift verarbeiten kann.

[Wikipedia]



Hardware

Hardware eines Rechners

- Mainboard
- Prozessor (CPU)
- Arbeitsspeicher (RAM)
- Grafikeinheit (GPU)
- Stromversorgung
- Netzwerkanschluss (LAN, WLAN)
- Anschlüsse für Peripherie



- Zentrale Platine eines Computers
- Montageplätze für andere Komponenten

Anschlüsse für Peripherie, Netzwerk

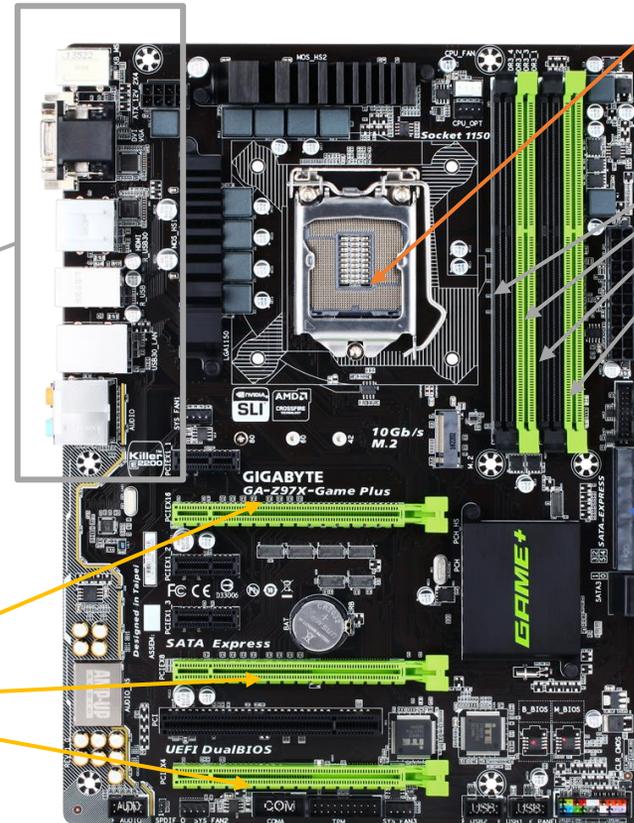
- (mehrere) USB-Buchsen
- Sound, Mikrofon
- Netzwerk (LAN)

GPU-Sockel (PCI Express)

CPU-Sockel

RAM-Sockel

Anschlüsse für
Festplatten



Central Processing Unit (CPU, Prozessor)

- „Zentrale Recheneinheit“
- Wichtige Angaben zur Leistung:
 - Anzahl Kerne
 - Taktfrequenz
 - Hyperthreading (ja/nein)

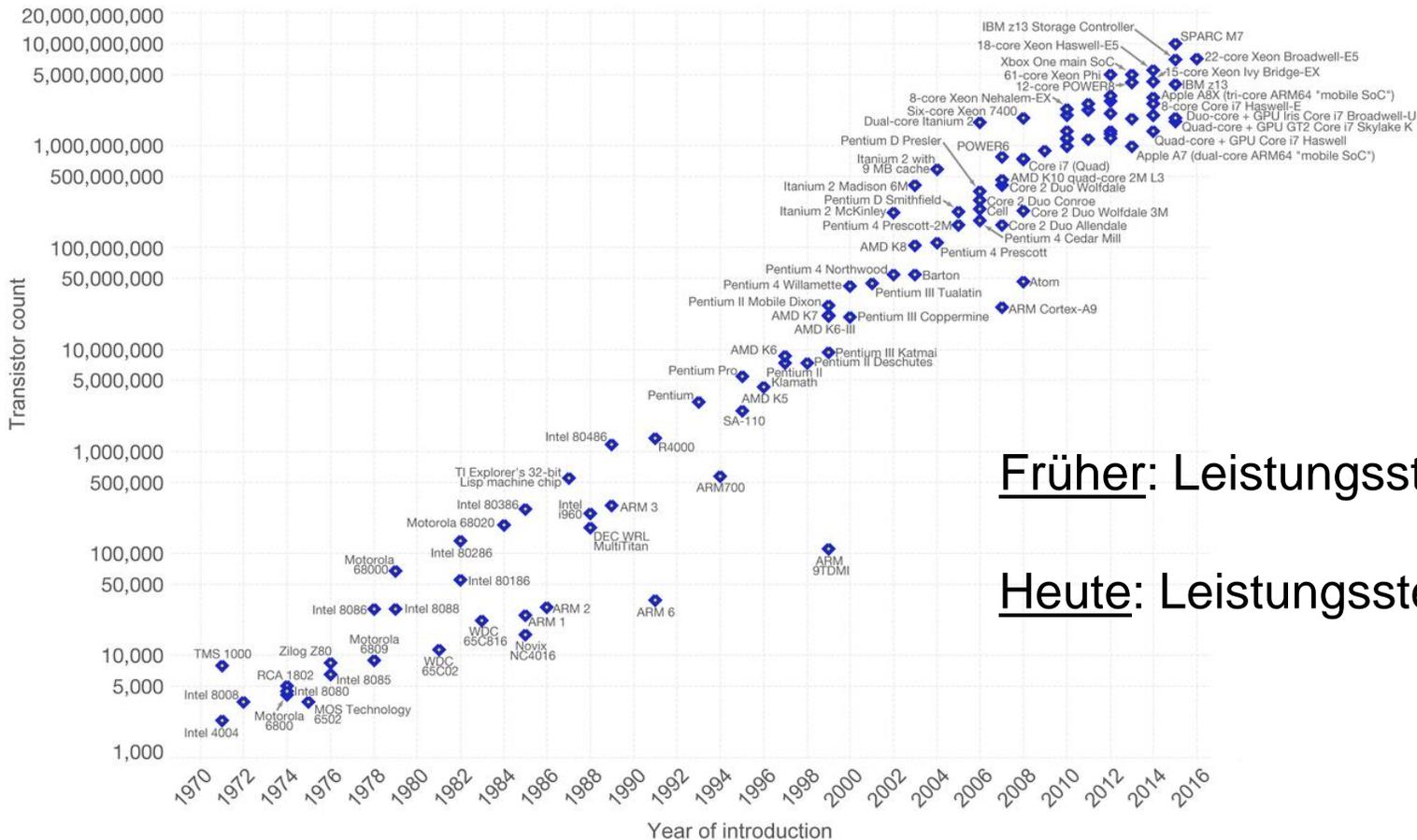


- Zwei wesentliche Architekturen:
 - Von-Neumann: Keine Trennung zwischen Speicher für Daten und Programmcode
 - Harvard: Daten und Programmcode sind strikt voneinander getrennt

Die Komplexität integrierter Schaltkreise verdoppelt sich etwa alle 18 Monate

Moore's Law – The number of transistors on integrated circuit chips (1971-2016) 

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important as other aspects of technological progress – such as processing speed or the price of electronic products – are strongly linked to Moore's law.



Früher: Leistungssteigerung der CPU

Heute: Leistungssteigerung durch Parallelisierung

Speicher: Arbeitsspeicher vs. Festplatte (externer Speicher)



	Arbeitsspeicher (RAM)	Festplatte SSD	Festplatte HDD
Geschwindigkeit	+++	+	0
Größe	Heute: 8 bis 16 GB	Heute: 4 TB möglich	Bis 16 TB
Anwendung	Daten, die die CPU braucht	(Dauerhafte) Speicherung größerer Daten	

Software

- Software, die den Betrieb eines Computers ermöglicht
 - Starten und Beenden des Betriebs
 - Programme laden und beenden
 - Verwaltung des Arbeitsspeichers
 - Zuteilung des Prozessors an die verschiedenen Programme
 - Verwaltung der Ein-/Ausgabegeräte (I/O) (→ Festplatten, Maus/Tastatur, Netzwerk, ...)
 - Überwachen von Berechtigungen (Nutzer/Administrator)
 - Abstraktion der Komplexität der (unterschiedlichen) Hardware
 - ...

Funktionelle Komponenten eines Betriebssystems

- Prozess-Management
- RAM-Management (Arbeitsspeicher)
- Dateiverwaltung (für externen Speicher)
- I/O-Unterstützung (Input/Output)
- Schutzmaßnahmen
- Netzwerk
- Kommandozeilen-Interpreter
- ...

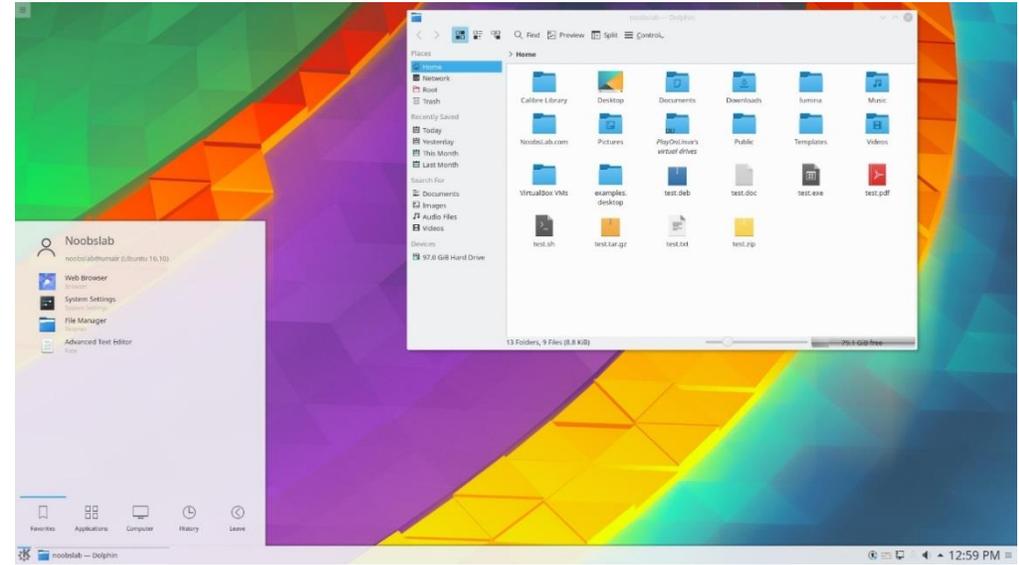
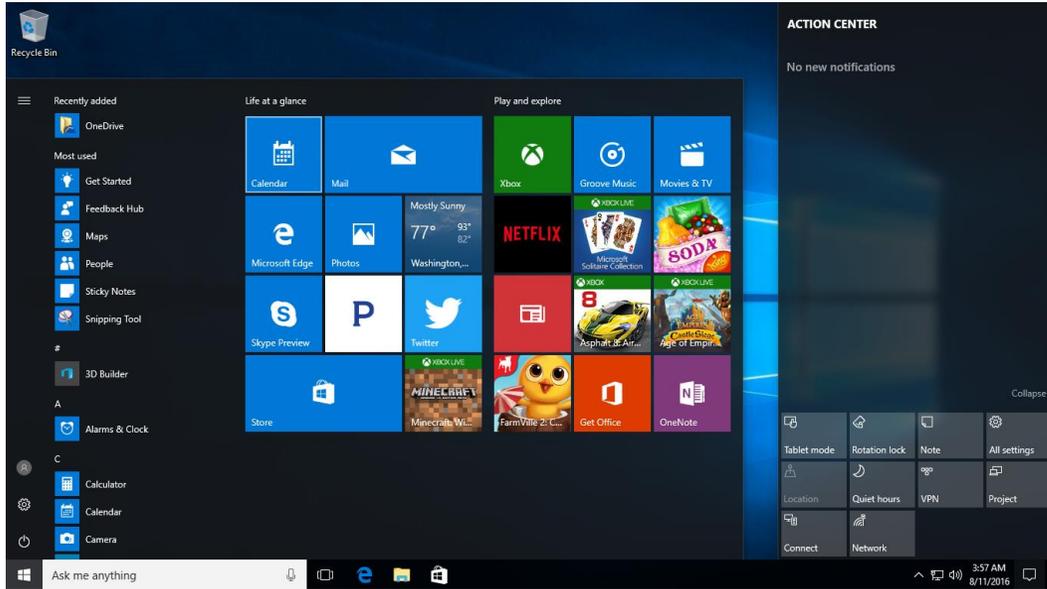


Betriebssystem vs. Benutzeroberfläche (GUI)

- Betriebssystem
 - Stellt elementare Dienste bereit
 - Abstrakte Schnittstelle zur Hardware
 - Verwaltet Betriebsmittel
- (Graphische) Benutzeroberfläche
 - Stelle komfortable Schnittstelle zu Anwendungsprogrammen und Systemdiensten bereit
 - Setzt Dienste „bildlich“ um (Mülleimer, Verschieben von Dateien)
 - Desktop-Metapher (Nachbildung eines Schreibtisch)
- Oft werden Betriebssystem und Benutzeroberfläche verwechselt, da für den Nutzer (v. a. bei MS Windows) kein Unterschied gemacht wird

Beispiele für Betriebssysteme bzw. deren GUIs

Microsoft Windows



Ubuntu (Linux) mit KDE-Desktop

Apple macOS



ie Informa

```

root ~ # ping google.com
PING google.com (74.125.95.103) 56(84) bytes of data:
64 bytes from iw-in-f103.1e100.net (74.125.95.103): icmp_seq=1 ttl=47 time=15.3 ms
^C
--- google.com ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 15.453/15.453/15.453/0.000 ms
root ~ # ls
Desktop README
root ~ # cd /
root / # ls
bin dev home lost+found mnt proc sbin srv tmp var
boot etc lib media opt root shared sys usr
root / # pacman -Ss pidgin
extra/libpurple 2.6.6-1
IM library extracted from Pidgin
extra/pidgin 2.6.6-1
Multi-protocol instant messaging client
extra/pidgin-encryption 3.0-3
A Pidgin plugin providing transparent RSA encryption using NSS
extra/purple-plugin-pack 2.6.3-1
Plugin pack for Pidgin
    
```

Linux CLI

Maschinensprache

- Hochsprache (Python)

Assembly vs. machine code

Machine code bytes	Assembly language statements
	foo:
B8 22 11 00 FF	movl \$0xFF001122, %eax
01 CA	addl %ecx, %edx
31 F6	xorl %esi, %esi
53	pushl %ebx
8B 5C 24 04	movl 4(%esp), %ebx
8D 34 48	leal (%eax,%ecx,2), %esi
39 C3	cmpl %eax, %ebx
72 EB	jnae foo
C3	retl

Instruction stream

```
B8 22 11 00 FF 01 CA 31 F6 53 8B 5C 24  
04 8D 34 48 39 C3 72 EB C3
```

```
def quicksort(Liste,unten,oben):  
    if (unten < oben):  
        pivot = (unten+oben)//2  
        teiler = zerlege(Liste,unten,oben,pivot)  
        quicksort(Liste,unten,teiler-1)  
        quicksort(Liste,teiler+1,oben)  
    return Liste
```

Einteilung Programmiersprachen: Konzept

- Imperativ (Basic, Pascal, C)
- Objektorientiert (Java, C++, C#, **Python**, Scala)
- Funktional (LISP, Haskell)
- Regelbasiert / Logisch (Prolog)
- Datenbank („Relational“) (**SQL**)

nicht im „engeren“ Sinne Programmiersprachen:

- Auszeichnungssprachen (**HTML, XML, JSON**)

Einteilung Programmiersprachen: Einsatzzweck

- Kleine Skripte (Python, Perl)
- (Größere) Programme mit GUI, Spiele (C++, C#, Java)
- Webserver (PHP, Python, Java/Scala)
- Mathematik (R, Mathematica, MatLab)

Name Dropping: Was ist ...

- Car2Car, Car2X
- Spracherkennungssoftware (Beispiele)
- Smart Home
- Internet of Things (IoT)
- Industrie 4.0
- Cloud Computing
 - IaaS, PaaS, SaaS
 - Public Cloud, Private Cloud, Community Cloud