

Programmiermethoden und -werkzeuge 1

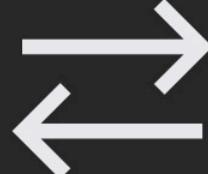
Woche 3 - Shell

Jochen Hosenfeld

jochen.hosenfeld@informatik.hs-fulda.de

Fachbereich Angewandte Informatik

October 31, 2025

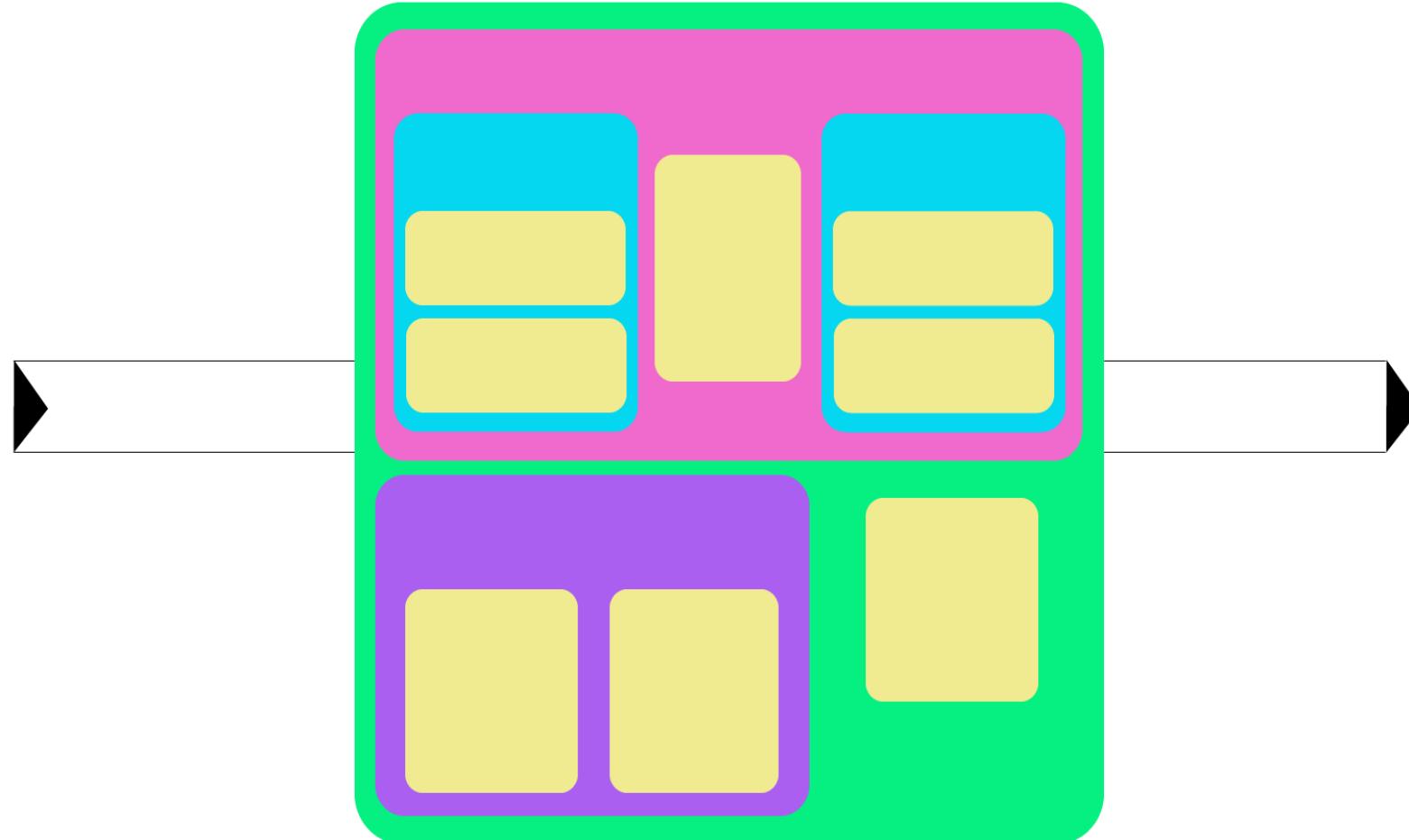
Do  

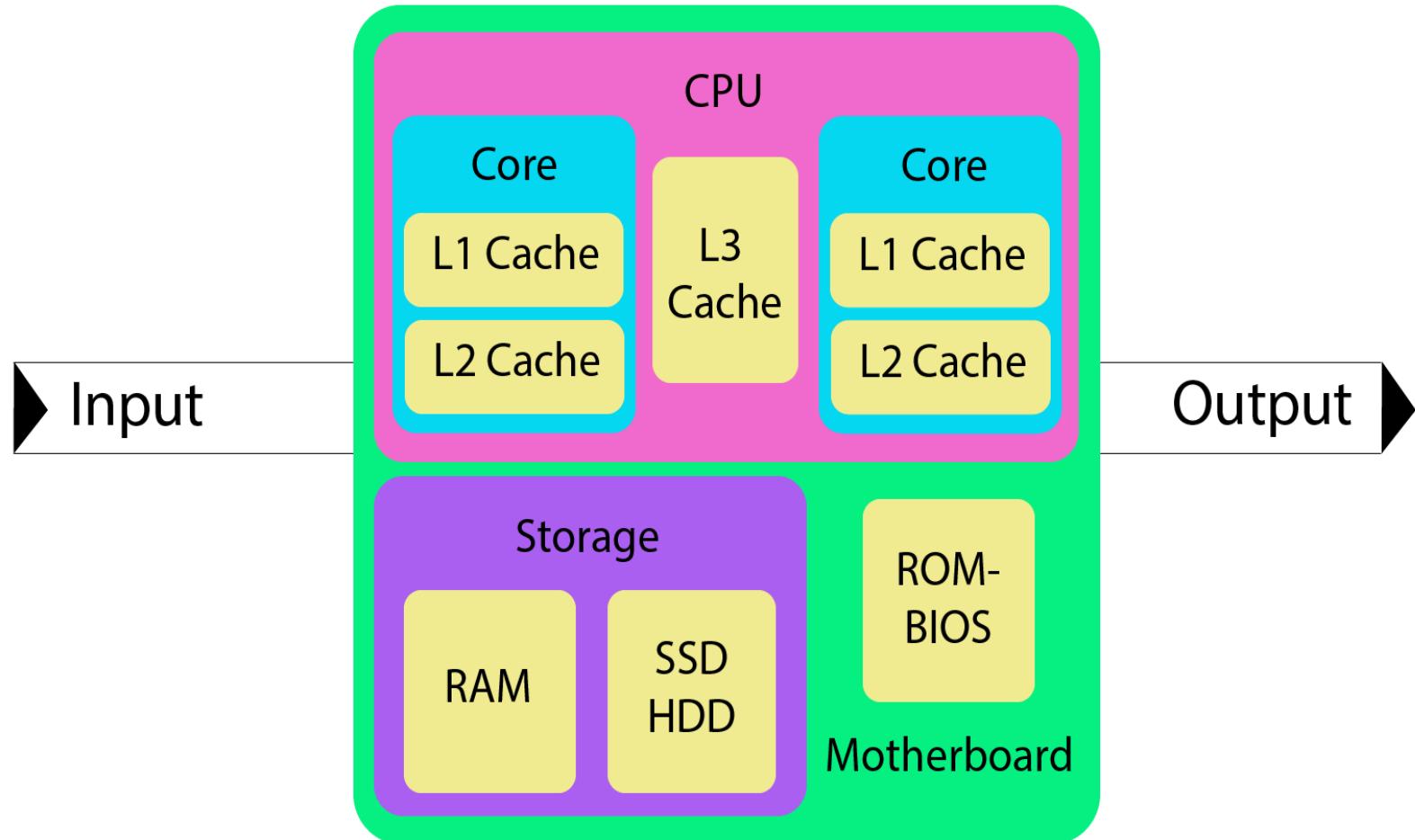
Do  

Wie Computer funktionieren - Arbeitsfläche

Lösung auf nächster Seite



Wie Computer funktionieren



Wie Computer funktionieren

| Komponente | Bedeutung | Beschreibung |
|-------------|--|---|
| CPU | Central Processing Unit | Zentrale Recheneinheit des Computers |
| L1 Cache | Level 1 Cache | Schnellster Zwischenspeicher in der CPU |
| L2 Cache | Level 2 Cache | Etwas größer und langsamer als L1 Cache |
| L3 Cache | Level 3 Cache | Gemeinsamer Cache für mehrere CPU-Kerne |
| RAM | Random Access Memory | Kurzzeitspeicher |
| SSD | Solid State Drive | Schneller, persistenter Datenspeicher |
| HDD | Hard Disk Drive | Mechanischer Datenträger mit Magnetscheiben |
| ROM-BIOS | Read Only Memory – Basic Input Output System | Enthält Startprogramm des Computers |
| Motherboard | Hauptplatine | Zentrale Verbindung aller Komponenten |

Mensch -> Computer

Brainstorming-Aufgabe: Welche Eingabegeräte gibt es?



Betriebssysteme

Operating Systems (OS)

- Windows
 - Microsoft
 - Desktops, Laptops und Server
 - Software-Ökosystem
- macOS
 - Apple
 - Fokus auf UX und Integration mit Apple-Produkten
- Linux
 - Open-Source
 - Verschiedene Distributionen wie Ubuntu, Fedora, Debian

Betriebssysteme

Operating Systems (OS)

- Android
 - Google, basiert auf Linux
 - Smartphones und Tablets
- iOS
 - Apple
 - iPhone und iPad
 - Geschlossenes Ökosystem -> Sicherheit und Benutzerfreundlichkeit
 - Variante iPadOS: größerer Bildschirm und Stiftbedienung

~~Betriebssysteme~~

- Mikrocontroller (z. B. Arduino)
 - Programmcode läuft direkt auf der Hardware
 - spezifische Aufgaben wie Sensorabfrage und Gerätesteuerung
- Eingebettete Systeme (Embedded Systems)
 - Routern, Autos, Haushaltsgeräten
- Firmware/BIOS
 - Initialisierung von Hardwarekomponenten vor dem Laden eines Betriebssystems
 - Rudimentäre Eingaben und Ausgaben sowie Systemstart

Betriebssysteme



~~OGRES ARE LIKE ONIONS~~

Schichten

| Schicht | Beschreibung |
|----------------------------------|--|
| Hardware | Physische Komponenten des Computers wie Prozessor, Speicher, Peripherie |
| HAL (Hardware-Abstraction Layer) | Stellt eine einheitliche Schnittstelle zur Hardware bereit und versteckt Hardware-Details vor dem Betriebssystem |
| Treiber | Kleine Hilsprogramme, die einzelne Hardwaregeräte ansprechbar machen |
| Kernel (Betriebssystemkern) | Verwalten von Hardwarezugriff und Ressourcen, Koordination der Programme |
| Management | Organisiert Prozesse, verteilt CPU-Zeit, Speicher und Eingabegeräte |
| Anwendungen | Benutzerprogramme, die Betriebssystem und Hardware nutzen |

Kernel

1. Start, Steuerung und Beendigung von Programmen
2. Zuteilung und Schutz des Arbeitsspeichers
3. Verbindung und Steuerung der Hardwaregeräte
4. Verwaltung von Dateien und Verzeichnissen
5. Sicherheit und Zugriffskontrolle

Betriebssysteme

Gruppenaufgabe: Finden Sie in einer Kleingruppe heraus, welches Betriebssystem Ihre Sitznachbarn verwenden und warum sie sich dafür entschieden haben.



Kernel <-> Shell

Shell

- 1977-79: Bourne Shell (sh)
- 1978: C shell (csh)
- 1983: KornShell (ksh)
- 1989: Bourne-Again Shell (bash)
- 1990: Z Shell (zsh)
- 1997: Debian's Almquist shell (dash)
- 2006: PowerShell

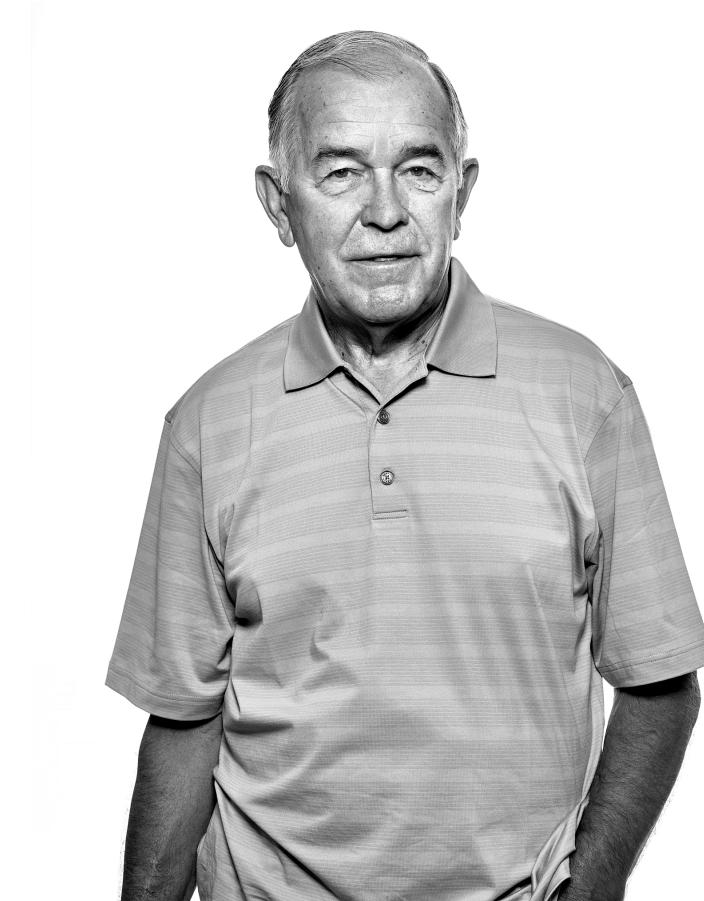
Shell

- 1977-79: Bourne Shell (sh)
- 1978: C shell (csh)
- 1983: KornShell (ksh)
- 1989: Bourne-Again Shell (bash)
- 1990: Z Shell (zsh) -> macOS Standard
- 1997: Debian's Almquist shell (dash) -> Effizienz und Schnelligkeit
- 2006: PowerShell -> Windows Standard

Bourne-Again Shell



Brian Fox



Stephen R. Bourne

Bash ist ...

- ein Programm
 - Ausführbare Software
- ein Interpreter
 - Liest Code und führt ihn aus
- eine Programmierumgebung
 - Bietet spezifische Tools und Funktionen für die Entwicklung und Ausführung von Bash-Skripten und Shell-Kommandos

Interpreter vs. Compiler

Interpreter

- Übersetzt Code Zeile für Zeile
- Echtzeitübersetzung während der Ausführung
- Ermöglicht schnelles Testen
- Langsamere Programmausführung

Compiler

- Übersetzt gesamten Code auf einmal
- Erzeugt ausführbare Datei vor der Ausführung
- Schnellere Programmausführung
- Änderungen erfordern Neukompilierung

Shell, Bash, Terminal, Prompt

Befehle

```
johos@AI-HOSENFELD-NB:~$  
date  
echo hello  
echo $PATH  
which echo  
/bin/echo $PATH
```

Slash / und Backslash \

The image displays a 2x3 grid of character pairs, each consisting of a backslash (\) and a forward slash (/). The pairs are arranged in two rows: the first row contains pairs where the backslash is on the left and the forward slash is on the right, while the second row contains pairs where the forward slash is on the left and the backslash is on the right. Each pair is composed of three vertical segments: a top segment connecting the two characters, a middle segment connecting them, and a bottom segment connecting them. The characters are rendered in white against a dark background.

| | | | | | |
|---|---|---|---|---|---|
| \ | / | / | \ | / | \ |
| \ | (|) | / | \ | (|

Slash /

- Pfadtrennung in Linux und macOS
- “root”
- startet einen “absoluten” Pfad
(/usr/local/bin)

Backslash \

- Pfadtrennung in Windows
- Ein “root” für jede Disk Partition (C:\)
- Escape-Funktion für Sonderzeichen

Pfade

```
1 /          root  
2  
3 .          aktuelles Verzeichnis  
4  
5 ..         Parent-Verzeichnis  
6  
7 .../.../... Parent-Parent-Parent-Verzeichnis
```

Navigation

```
1  pwd          print working directory  
2  
3  cd /home    aktuelles Home-Verzeichnis  
4  
5  ls           list
```

Options

```
1 ls --help    "help" option
2
3 ls -l        long listing format
```

Rechte

```
1 johos@AI-HOSENFELD-NB:~$ ls -l
2 total 8
3 drwxr-xr-x 3 johos johos 4096 Aug 21 16:29 dev
4 -rw-r--r-- 1 johos johos    72 Sep  3 16:39 file.txt
```

Rechte

```
1 johos@AI-HOSENFELD-NB:~$ ls -l
2 total 8
3 drwxr-xr-x 3 johos johos 4096 Aug 21 16:29 dev
4 -rw-r--r-- 1 johos johos    72 Sep  3 16:39 file.txt
5
6 # d = directory
7
8 # r = read
9 # w = write
10 # x = execute
```

Rechte

```
1 johos@AI-HOSENFELD-NB:~$ ls -l
2 total 8
3 drwxr-xr-x 3 johos johos 4096 Aug 21 16:29 dev
4 -rw-r--r-- 1 johos johos    72 Sep  3 16:39 file.txt
5
6 # d = directory
7
8 # r = read
9 # w = write
10 # x = execute
11
12 # 1. Gruppe: Besitzer
13 # 2. Gruppe: Gruppe
14 # 3. Gruppe: alle anderen
```

Verwaltung

- Create
 - Read
 - Update
 - Delete
- > CRUD-Operationen



Spezieller User: "root" bzw. "super user"

Verwaltung

- Create
 - Read
 - Update
 - Delete
- > CRUD-Operationen



Spezieller User: "root" bzw. "super user"

- hat (fast) uneingeschränkten Zugriff
- hohes Risiko, Systemfehler zu verursachen
- nicht automatisch als "root" anmelden
- besser: Befehle mit Rechten des "super users" ausführen

sudo -> “do” as “su”

sudo -> “do” as “su”

```
1 johos@AI-HOSENFELD-NB:~$ ls -l
2 total 8
3 drwxr-xr-x 3 johos johos 4096 Aug 21 16:29 dev
4 -rwxr-xr-x 1 johos johos    72 Sep  3 16:39 file.txt
5 -rwxr-xr-x 1 root  root     0 Oct 11 16:20 only_root.txt
6 johos@AI-HOSENFELD-NB:~$ chmod 755 only_root.txt
7 chmod: changing permissions of 'only_root.txt': Operation not permitted
```

sudo -> “do” as “su”

```
1 johos@AI-HOSENFELD-NB:~$ ls -l
2 total 8
3 drwxr-xr-x 3 johos johos 4096 Aug 21 16:29 dev
4 -rwxr-xr-x 1 johos johos    72 Sep  3 16:39 file.txt
5 -rwxr-xr-x 1 root  root     0 Oct 11 16:20 only_root.txt
6 johos@AI-HOSENFELD-NB:~$ chmod 755 only_root.txt
7 chmod: changing permissions of 'only_root.txt': Operation not permitted
8 johos@AI-HOSENFELD-NB:~$ sudo chmod 755 only_root.txt
```

sudo -> “do” as “su”

```
1 johos@AI-HOSENFELD-NB:~$ ls -l
2 total 8
3 drwxr-xr-x 3 johos johos 4096 Aug 21 16:29 dev
4 -rwxr-xr-x 1 johos johos    72 Sep  3 16:39 file.txt
5 -rwxr-xr-x 1 root  root     0 Oct 11 16:20 only_root.txt
6 johos@AI-HOSENFELD-NB:~$ chmod 755 only_root.txt
7 chmod: changing permissions of 'only_root.txt': Operation not permitted
8 johos@AI-HOSENFELD-NB:~$ sudo chmod 755 only_root.txt
9 johos@AI-HOSENFELD-NB:~$ ls -l
10 total 8
11 drwxr-xr-x 3 johos johos 4096 Aug 21 16:29 dev
12 -rwxr-xr-x 1 johos johos    72 Sep  3 16:39 file.txt
13 -rwxr-xr-x 1 root  root     0 Oct 11 16:20 only_root.txt
14 johos@AI-HOSENFELD-NB:~$
```

CRUD -Operationen

```
1 # Create
2 mkdir      make directories
3 touch      create empty files or update the timestamps
4
5 # Read
6 cat        concatenates and displays the content of files
7 less       interactively viewing (large) text files
8 head       output the first part of files
9 tail       output the last part of files
10
11 # Update
12 mv        move (rename) files
13
14 # Delete
15 rm        removes files or directories
```

Verwaltung und Dokumentation

```
1 mv           rename/remove  
2  
3 cp           copy  
4  
5 mkdir        make new directory  
6  
7 man          manual (q to exit manual)
```

Windows-Subsystem für Linux (WSL)

Einrichten einer WSL-Entwicklungsumgebung