

# All About Memory

Lennart Armbrust

Freie Universität Berlin



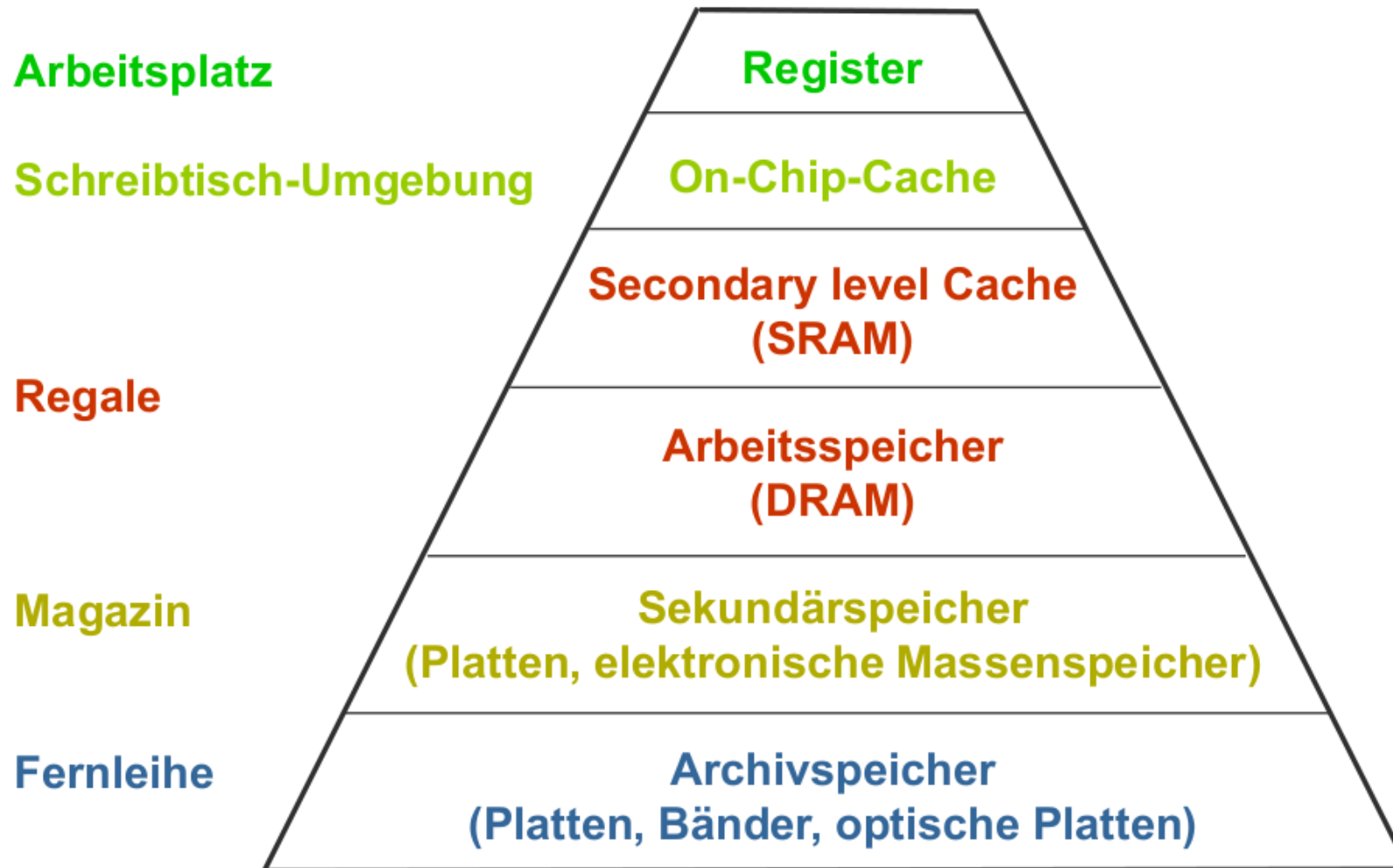
# Einführung

# Gliederung

- Verschiedene Speichertypen
  - Zugriffszeiten
- Die Hardware
- Speicherhierarchie
  - Virtualisierung
- Unterschiede je nach Art des Rechners
- Verbesserungsmöglichkeiten

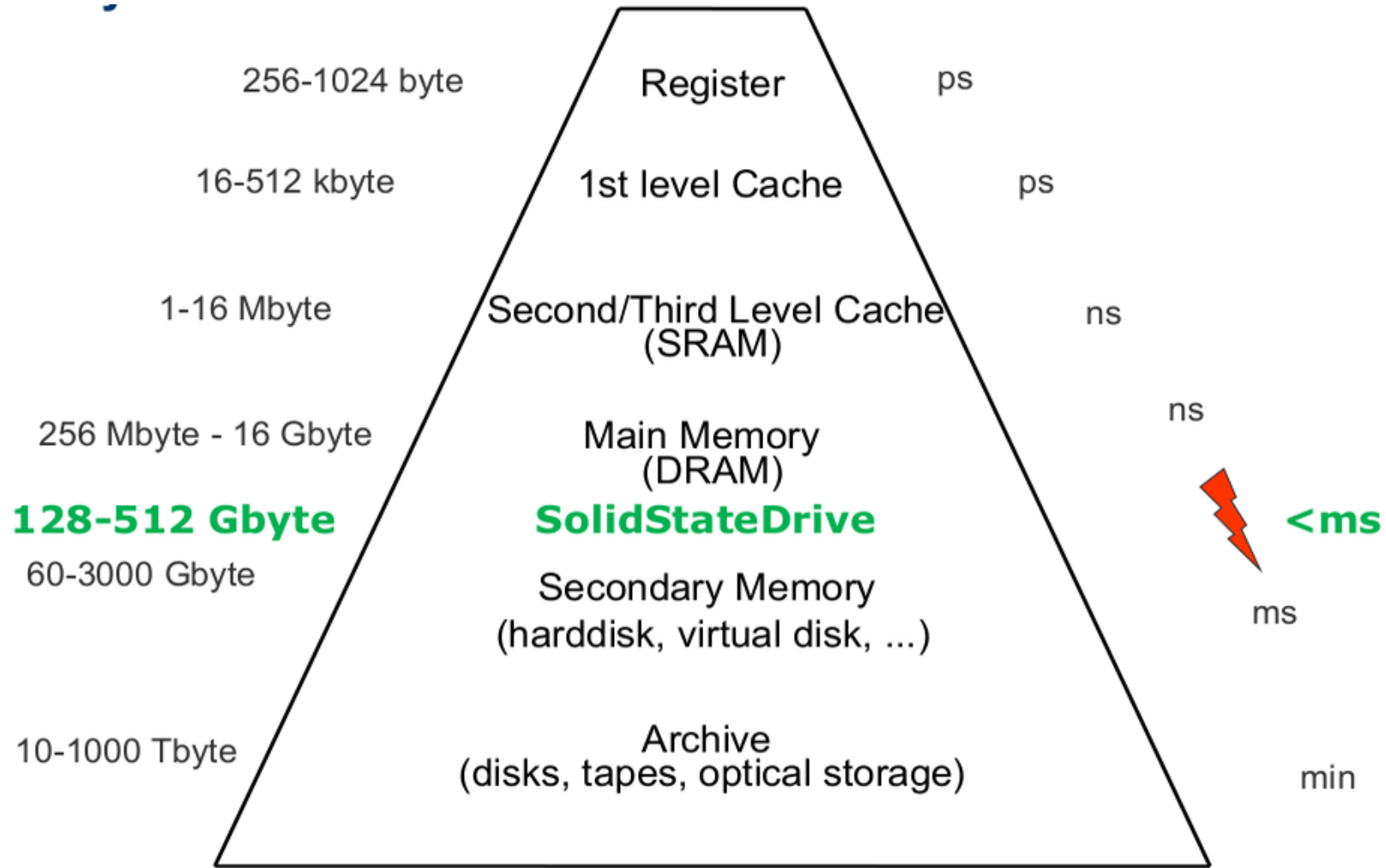
# Verschiedene Speichertypen

# Verschiedene Speichertypen



„TI II: Computer Architecture, Memories“, S. 4, Prof. Jpchen Schiller, FU Berlin

# Verschiedene Speichertypen



„TI II: Computer Architecture, Memories“, S. 7, Prof. Jpchen Schiller, FU Berlin

# Die Hardware

# Die Hardware: Register

- Arbeitsfläche / “Schreibtisch” des Prozessors
- Verschiedene Register im Prozessor:
  - General Purpose
  - Spezialregister:
    - Instruction Pointer
    - Befehlsregister
    - Flagregister
    - ...

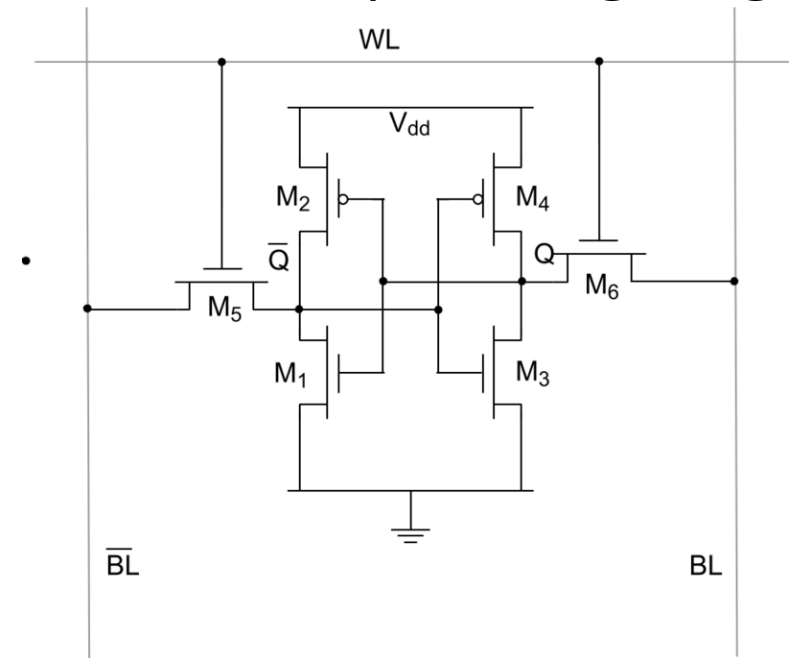


# Die Hardware: Caches

- Lösungsansatz für den Von-Neumann-Flaschenhals
- Oft nachgefragte bzw. für ein Programm benötigte Daten werden direkt im Prozessor abgelegt
  - => man muss nicht “aus dem Prozessor raus”
- Verschiedene Ebenen und Realisierungsformen:
  - Direct-mapped
  - Vollasoziativ
  - N-fach satzassoziativ
- Wenn Cache voll: Ersetzungsstrategien:
  - First In First Out, Least Recently Used, Least Frequently Used, ...

## Die Hardware: SRAM

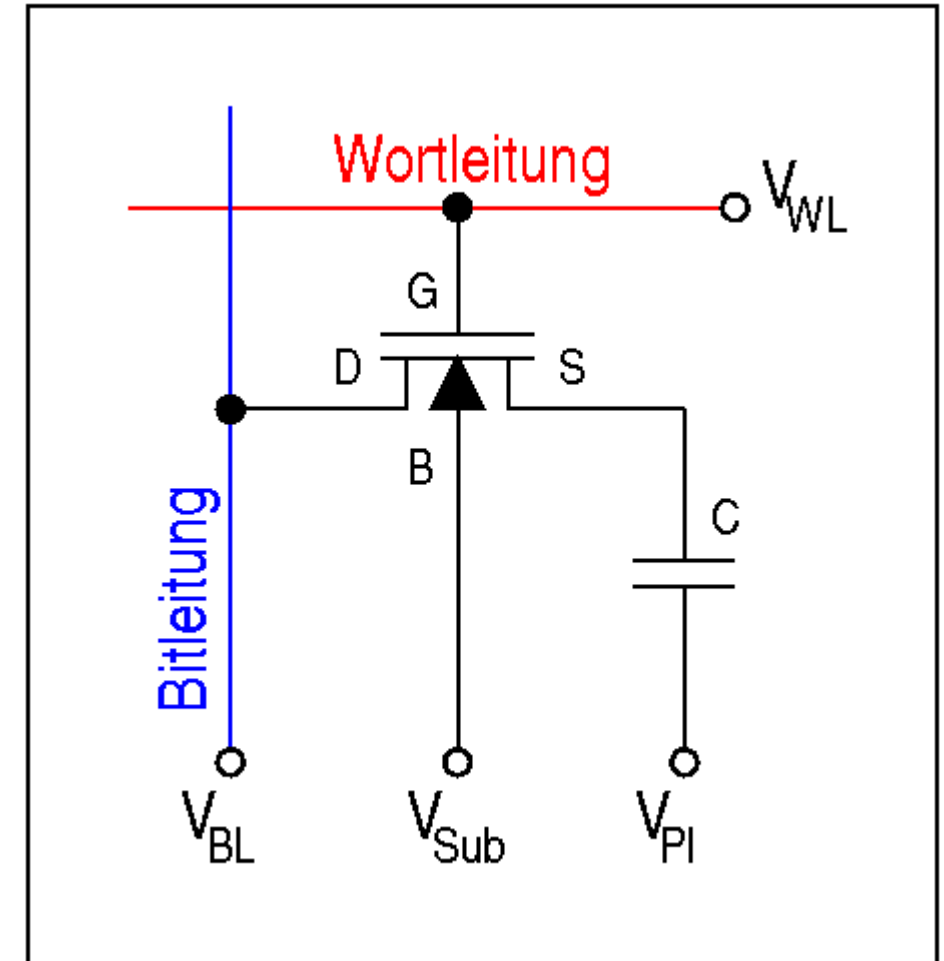
- “Static Random Access Memory”
- Genutzt für Cache
- Flüchtig
- Einmal gespeicherter Wert bleibt erhalten, bis Betriebsspannung wegfällt
- 6 Transistoren pro Bit
- Wenn nicht zugegriffen wird:  
sehr geringer Stromverbrauch



Abelsson, [CC BY-SA 3.0](#)

# Die Hardware: DRAM

- “Dynamic Random Access Memory”
- Genutzt für Arbeitsspeicher
- Flüchtig
- Muss periodisch aufgefrischt werden
- 1 Kondensator und 1 Transistor pro Bit



JürgenZ., [CC BY-SA 3.0](#)

# Die Hardware: SSD

- “Solid State Drive”
- Basiert auf Halbleitern
- Nichtflüchtig
- Hohe Geschwindigkeit für Massenspeicher
- Energieeffizient



D-Kuru, [CC BY-SA 3.0](#)

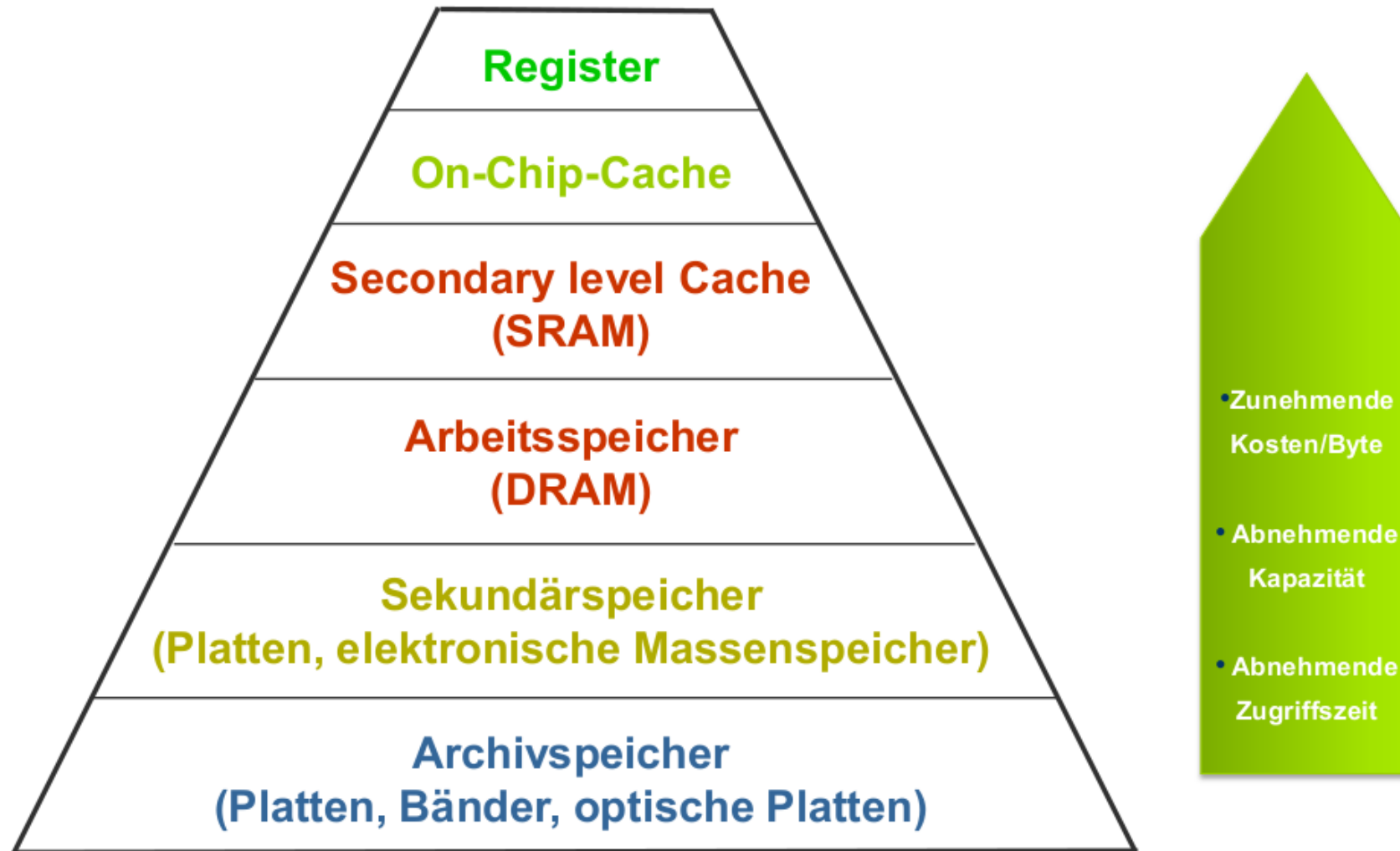
# Die Hardware: HDD

- “Hard Disk Drive”
- Basiert auf Magnetscheiben
- Nichtflüchtig
- Hohe Kapazität
- Günstig
- Bewegliche Teile: Anfällig für Stöße



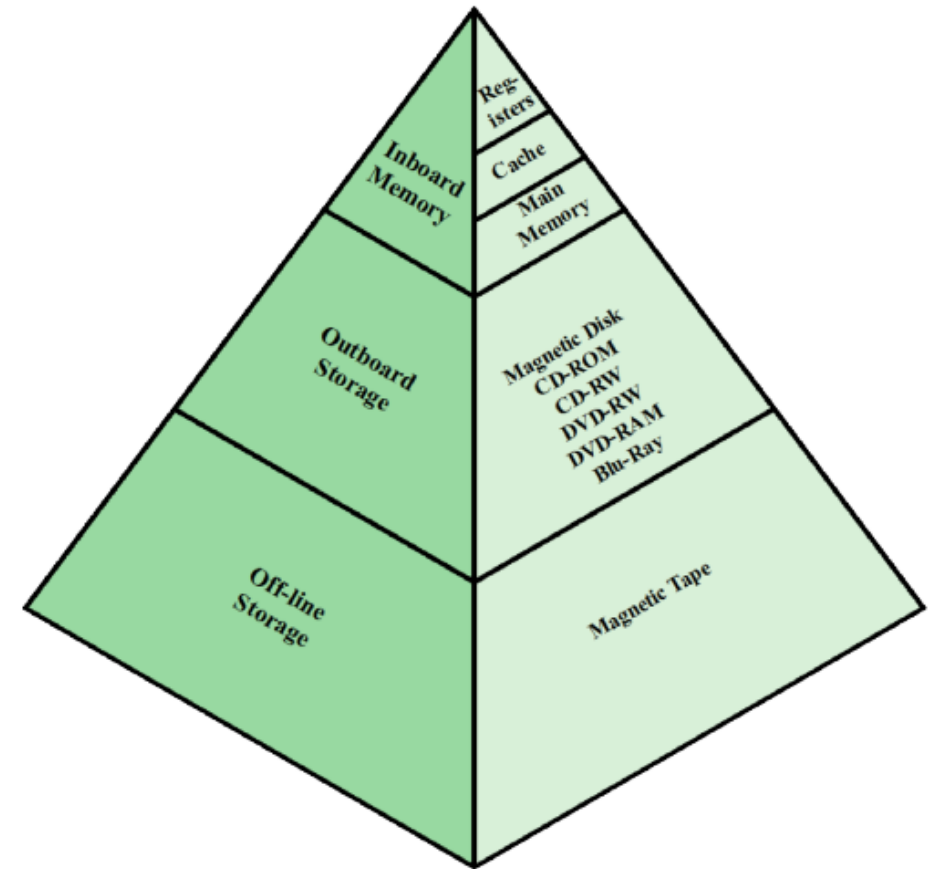
Eric Gaba, [CC BY-SA 3.0](https://creativecommons.org/licenses/by-sa/3.0/)

# Die Hardware: Vergleich

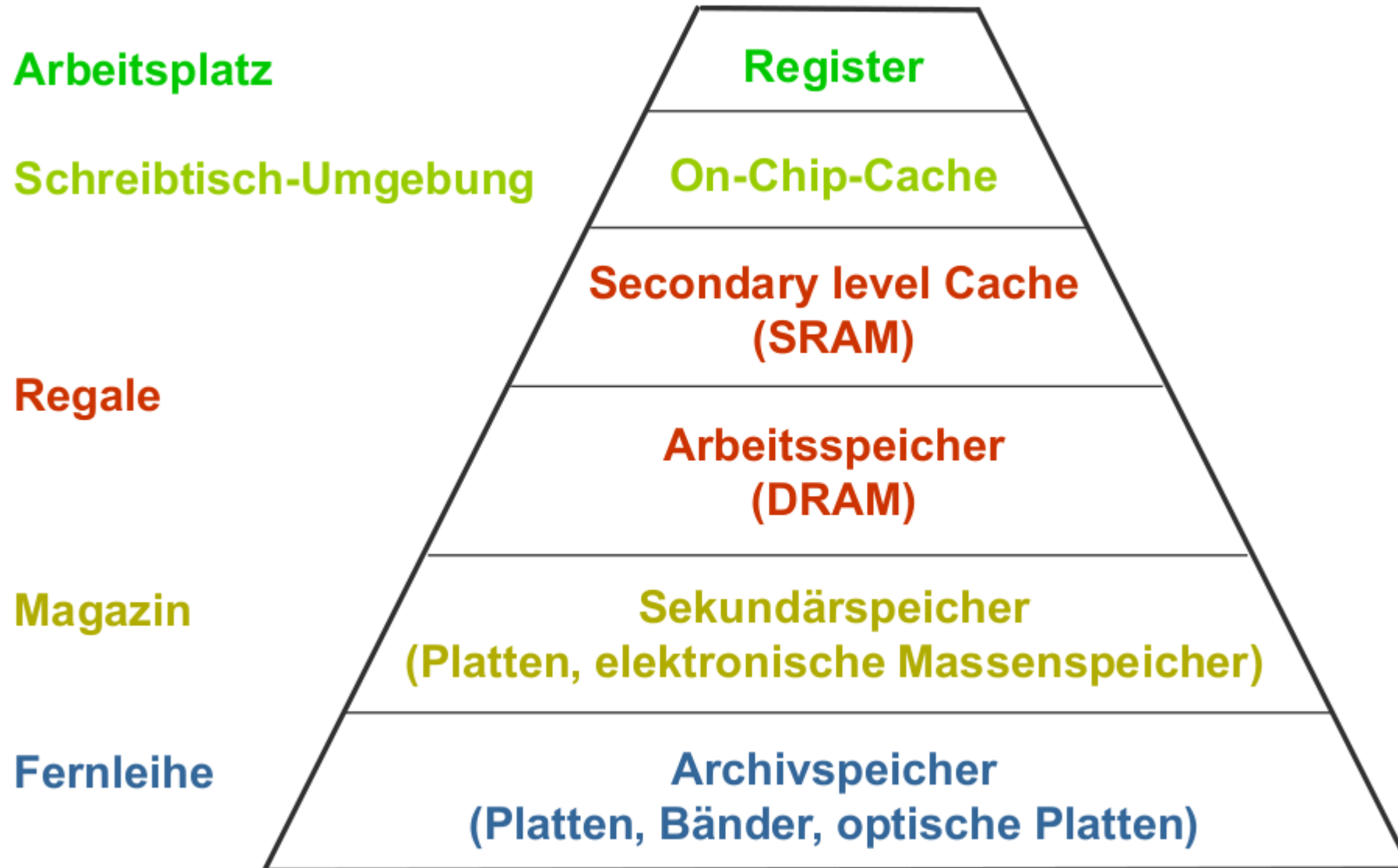


„TI II: Computer Architecture, Memories“, S. 5, Prof. Jpchen Schiller, FU Berlin

# Speicherhierarchie



# Speicherhierarchie

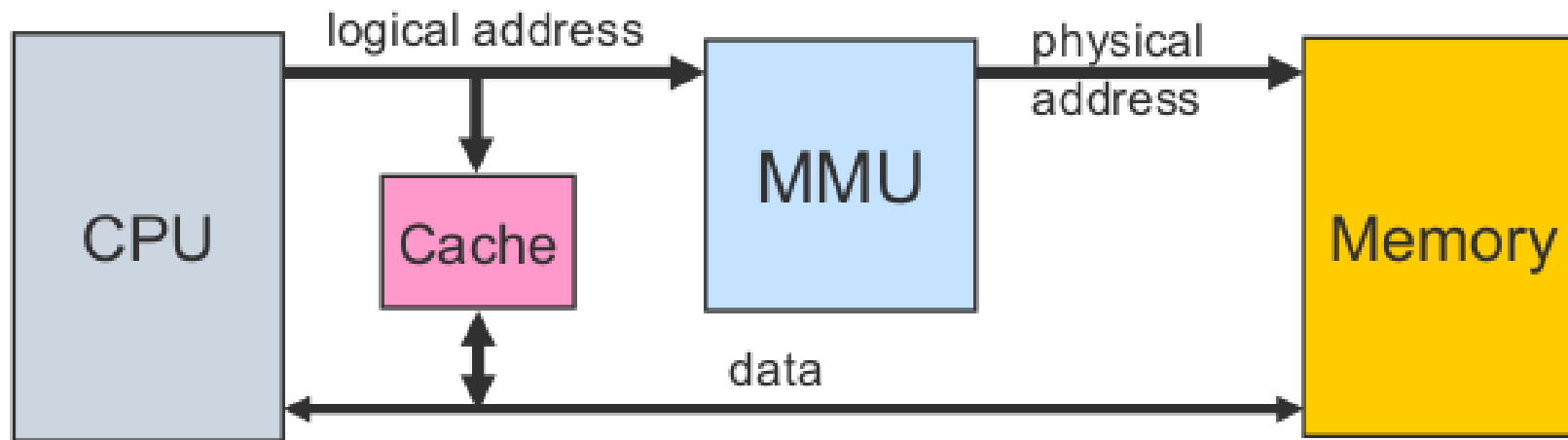




# Speicherhierarchie

- Man möchte:
  - Riesigen sehr schnellen Speicher
- Da das nicht geht, hierarchisiert man Speicher
- Wirkt wie ein großer, zusammenhängender Speicher, wenn:
  - Lokalität der Programmverarbeitung gegeben
  - Umlagerung der Information rechtzeitig (Umlagerungsstrategien)
  - Inhomogenität des Speichersystems für Benutzer nicht sichtbar (Virtueller Speicher)

# Speicherhierarchie: Virtualisierung



„TI II: Computer Architecture, Memories“, S. 102, Prof. Jpchen Schiller, FU Berlin

# Speicherhierarchie: Virtualisierung

- Speicher wirkt für Programme/Benutzer völlig homogen
- Jedes Programm bekommt seinen eigenen Speicherbereich
- Weiterer Vorteil: Sicherheit => MMU sorgt dafür, dass Programme in ihrem Speicherbereich bleiben

# Unterschiede nach Art des Rechners

# Unterschiede

- Je nach Anwendung teilweise sehr spezielle Designs
- Z. B. Kleine Embedded Systems ohne Virtualisierung

# Verbesserungsmöglichkeiten

# Verbesserungsmöglichkeiten

- Speicherhierarchisierung trägt im Vergleich zu homogenem Speicher zu Effizienz bei
  - Großer Speicher & Schneller Speicher sind so zusammen möglich
  - Trägt zu effizienter und ökonomischer Speichernutzung bei
- Hardwarenahe Ansätze ermöglichen weniger Stromverbrauch bzw. erhöhte Energieeffizienz
  - SSDs  $\Leftrightarrow$  HDDs
  - Auffrischen von DRAM verbraucht viel Energie