



**TECHNISCHE
UNIVERSITÄT
DRESDEN**

Faculty of Computer Science Institute of Systems Architecture, Operating Systems Group

EINFÜHRUNG UND BAUSTEINE

MICHAEL ROITZSCH

Organisatorisches

Vorlesung

- Webseite und Mailingliste
- Fragen zur Organisation: Till Smejkal <till.smejkal@tu-dresden.de>
- Fragen zum Inhalt: Michael Roitzsch <michael.roitzsch@tu-dresden.de>
- Fragen und Feedback auch gern nach der Vorlesung

Übung

- Aufgaben auf der Webseite
- Einschreibung per jExam ab Freitag Nachmittag
- zweite Vorlesungswoche: praktische Übungen als Konsultation im Rechnerpool
- ab dritter Vorlesungswoche: reguläre Übungsgruppen

Themen der Vorlesung

- Vermittlung einer integrierten Sicht auf Rechner-Systeme (Hardware, kryptographische Verfahren, Mathematik, maschinennahes Programmieren, Anwendung, ...)
- Grundlagen der Systemarchitektur, Betriebssysteme
- Umgang mit Parallelität
- Umgang mit Betriebsmitteln
- Abwägungen zwischen Zielen (Trade Off)

Literatur

Modern Operating Systems

Andrew S. Tanenbaum

Prentice Hall

Distributed Systems, Concepts and Design

Coulouris, Dollimore, Kindberg

Addison Wesley

Publikationen

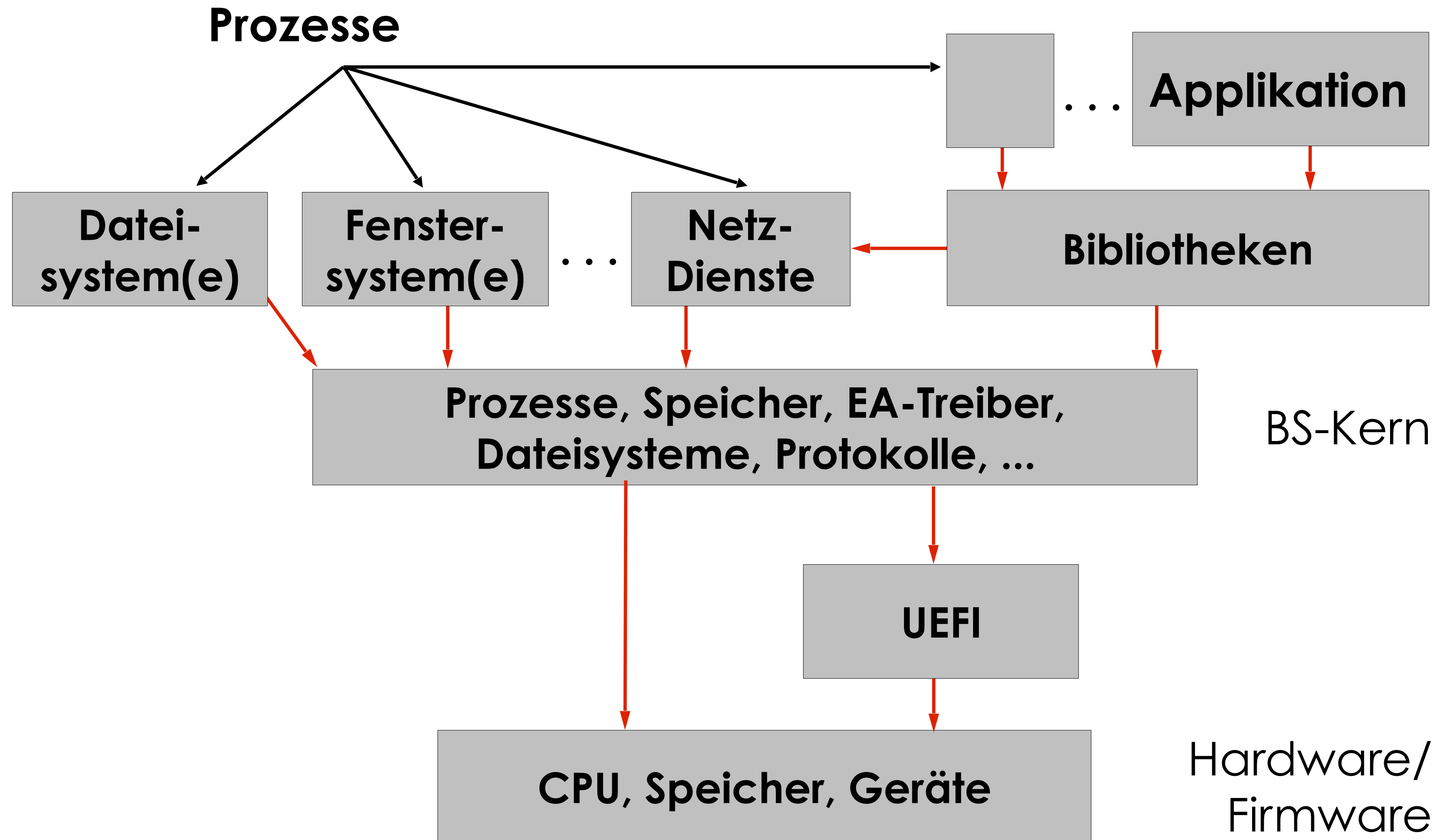
Betriebssystem: Begriffsbestimmung

- Summe derjenigen Programme, die als residenter Teil einer EDV-Anlage für den Betrieb der Anlage und für die Ausführung der Anwenderprogramme erforderlich ist.
— *Lexikon der Informatik, 1991*
- das wichtigste Systemprogramm, es kontrolliert die Ressourcen des Rechners und ist die Basis für die Entwicklung der Anwendungsprogramme
— *TANENBAUM*
- Das Betriebssystem wird gebildet durch die Programme eines digitalen Rechensystems, die zusammen mit den Eigenschaften der Rechenanlage die Basis der möglichen Betriebsarten des digitalen Rechensystems bilden und insbesondere die Abwicklung von Programmen steuern und überwachen.
— *DIN 44300*

Betriebssystem: Begriffsbestimmung

- Betriebssystem ist ein Programm, das als Zwischenmedium zwischen Nutzer und Hardware tritt.
Zweck: Umgebung zur Programmausführung schaffen
Ziel: bequeme Nutzung, effizientes Betreiben
— *SILBERSCHATZ*
- Komponente eines Rechensystems zur Steuerung von Prozessen und zur Verwaltung von Betriebsmitteln mit dem Ziel, eine einfache Nutzung und einen effizienten Betrieb zu ermöglichen.

System-Struktur



System-Struktur: Ein-/Ausgabe



Aufgaben eines Betriebssystems

Sichere und effiziente Ausführung von Benutzerprogrammen:

- Isolation der Anwendungen
- Bereitstellung von Betriebsmitteln (CPU, Speicher, IO, ...)
- Schnittstelle zu Geräten
- Persistente Datenhaltung (z. B. Dateien)
- Kommunikation, Vernetzung
- Virtualisierung (virtueller Speicher, virtuelle Maschinen, ...)
- Benutzer- und applikationsfreundliche Schnittstelle zur Hardware

Aufgaben eines Betriebssystems

Sicherstellung wichtiger Systemeigenschaften:

- Effizienz in der Nutzung von Betriebsmitteln (auch Energie)
- Sicherheit (gegen Angriffe von innen und außen)
- Fehlertoleranz (HW-Ausfälle, SW-Fehler, ...)
- Messen, Abrechnen („Accounting“)
- Echtzeitfähigkeit
- Skalierbarkeit

Weitere Ziele beim Entwurf eines Betriebssystems:

- Einfachheit, Wartbarkeit, Erweiterbarkeit
- Komfort, Kompatibilität, Funktionsumfang

Klassifikationskriterien für Betriebssysteme

- Aufgabenbereich/Betriebsart
 - Stapelverarbeitung („Batch“): Supercomputer, Big Data
 - Interaktives Endgerät (Laptop, Desktop, Tablet, Handy)
 - Interaktiver Serverbetrieb (Suchmaschine, Soziale Netze)
 - Eingebettete Systeme
- Architektur (Mikrokern – monolithisch, ...)
- Verteilung (Einzelrechner, Cluster, Data Center, „Cloud“)

Grundlegende Bausteine

- Threads
- Adressräume
- Prozesse und Kommunikation
- Dateien
- Betriebssystem-Kern und Dienst-Prozesse
- E/A-Treiber

Definition: Thread

Eine selbständige

- ein sequentielles Programm ausführende
- zu anderen Threads parallel arbeitende
- von einem Betriebssystem zur Verfügung gestellte Aktivität.



Gegenüberstellung: Funktionen und Threads

Funktionen P & Q

- Q beginnt, wenn P beendet
- P und Q werden von einem Thread ausgeführt

Threads P & Q

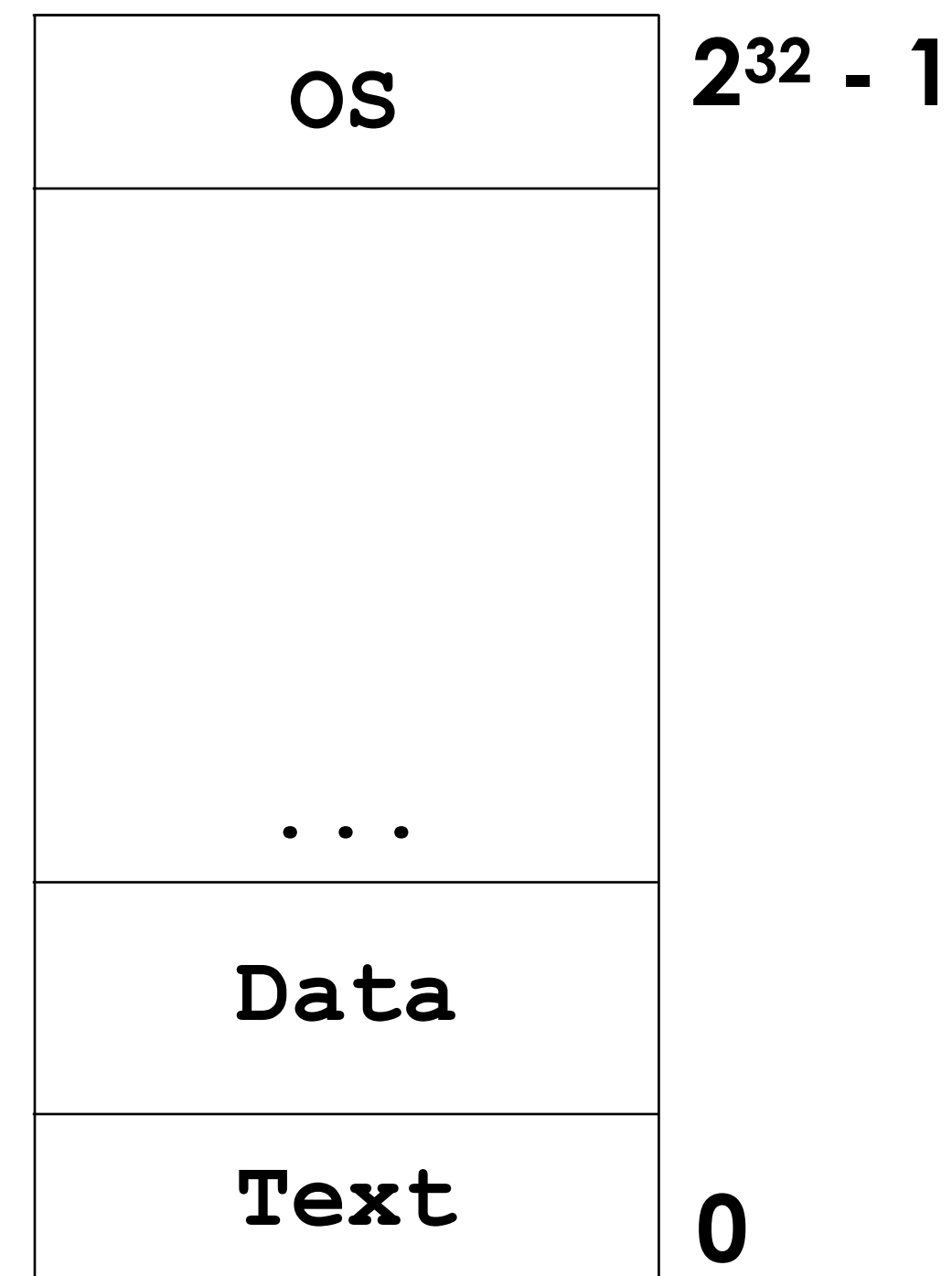
- keinerlei Aussage zur Ausführungsreihenfolge der Instruktionen von P und Q
- P,Q – wenn einmal gestartet – sind selbständig

Definition: Adressraum

- Menge aller Daten und Instruktionen, auf die „direkt“, d. h. mittels **load/store/...** HW-Instruktionen und über den PC zugegriffen werden kann
- z.B.: primitive CPU, d.h. Zugriff nur über ein Adressregister A (32 Bit)

```
ld R, A
//lade den Wert, welcher
//im Speicher an der
//Adresse steht, die im
//Register A angegeben ist,
//ins Register R

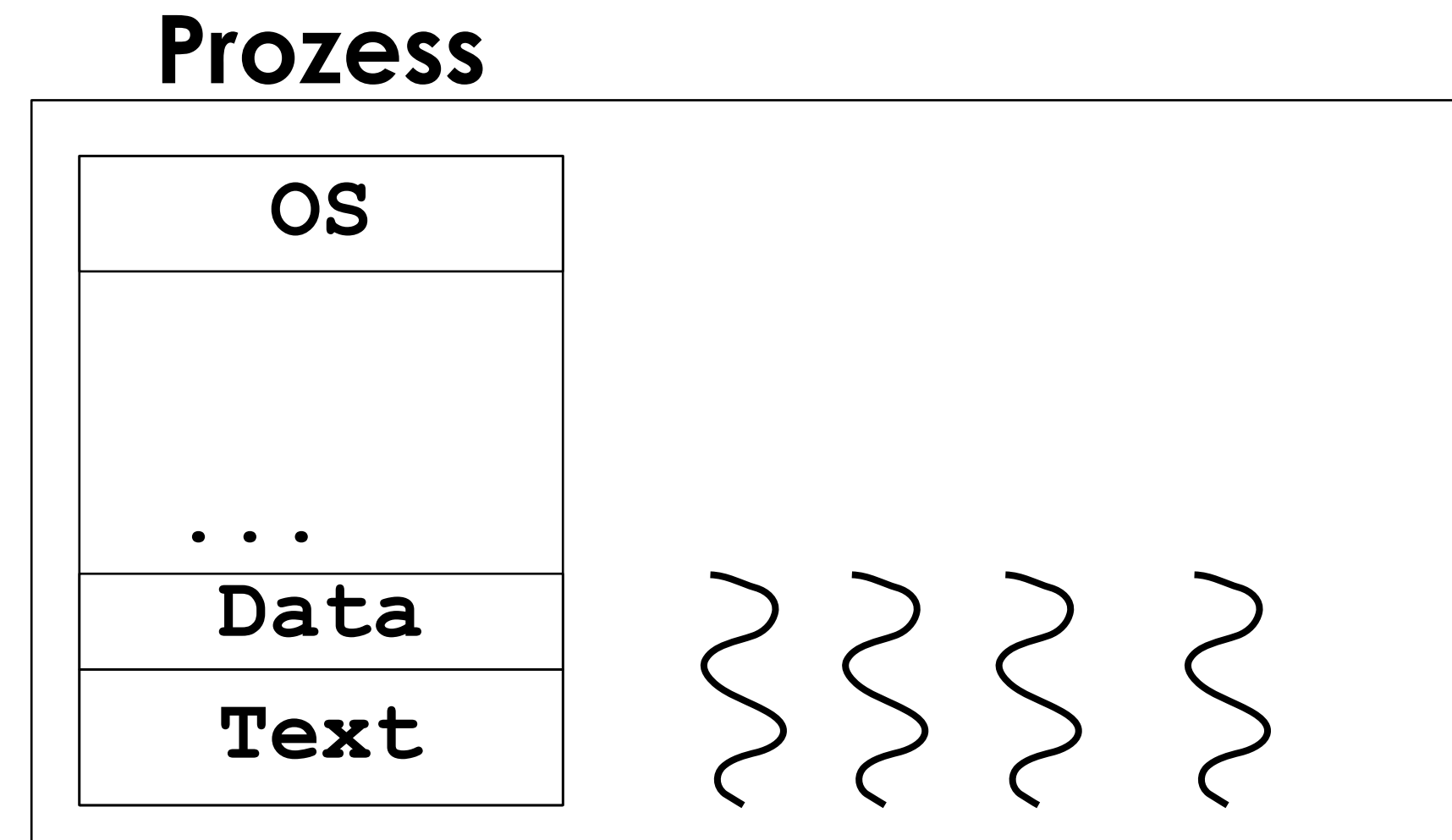
st R, A
//analog schreibend
```



Definition: Prozess

Eine Einheit aus

- einem Adressraum und
- mindestens einem Thread



Weitere Eigenschaften

- Einheit, der Betriebsmittel zugeordnet werden
- Repräsentant von Benutzern innerhalb des Betriebssystems

Begriffsvielfalt ...

... in Betriebssystemen und Programmiersprachen

- MACHs Task ist ein „multithreaded“ Prozess
- ADAs Task ist ein Thread

... in Bibliotheken, Lehrbüchern und Hardware

- „Thread“ manchmal nur im Kontext von „multithreaded“ Prozessen
- „User Level Threads“ vs. Threads, die von einem Betriebssystem zur Verfügung gestellt oder unterstützt werden
- Hardware-Architektur-Kontext: mehrere eigenständig arbeitende Instruktionsströme (Hyperthreading, GPUs)

Wichtig: Begriffsklärung sicherstellen

Beispiele für den Einsatz von Prozessen

- Mehrere Benutzer gleichzeitig auf einem Rechner
 - jeder Benutzer wird durch mehrere Prozesse repräsentiert
- Explizite Parallelarbeit eines Benutzers
 - mehrere Programme gestartet
- Bereitstellung von Diensten durch Hintergrundprozesse
- Auslagerung einer schützenswerten Unteraufgabe in eigenen Prozess

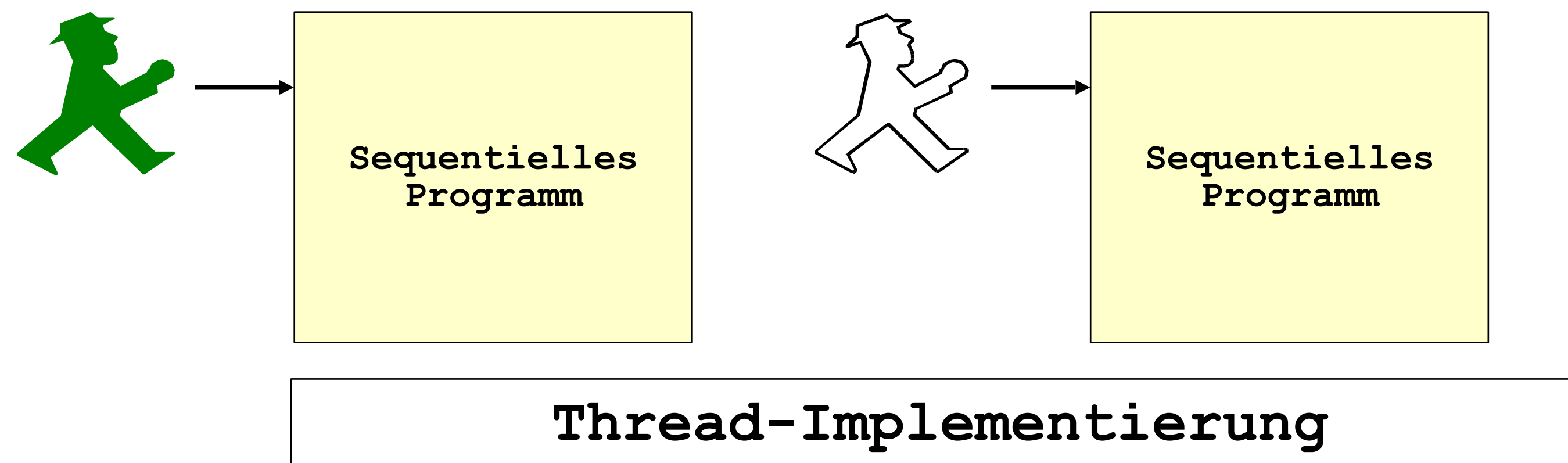
Beispiele für den Einsatz von Threads

- Explizite Parallelarbeit eines Benutzers, z. B.
 - parallele Bearbeitung von Matrizen
 - parallele kombinatorische Suche
- Umgang mit Asynchronität, z. B.
 - Tastatureingabe
 - Unterbrechungsbearbeitung
 - Ein-/Ausgabe
- Hilfsmittel zur Strukturierung komplexer Programme
 - jeder Klient eines Dienstes wird durch einen Thread repräsentiert

Mehrere Threads auf einem Rechner

Implementierung

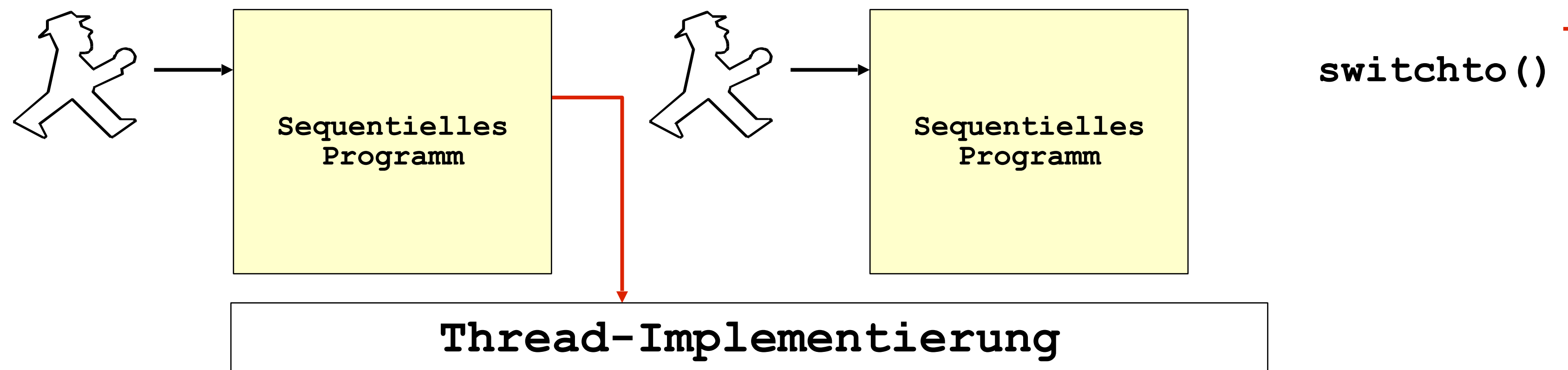
- Zu jedem Zeitpunkt kann auf einer CPU nur ein Thread laufen
- Jeder Thread erhält hin und wieder die CPU:
Thread-Umschaltung



Mehrere Threads auf einem Rechner

Implementierung

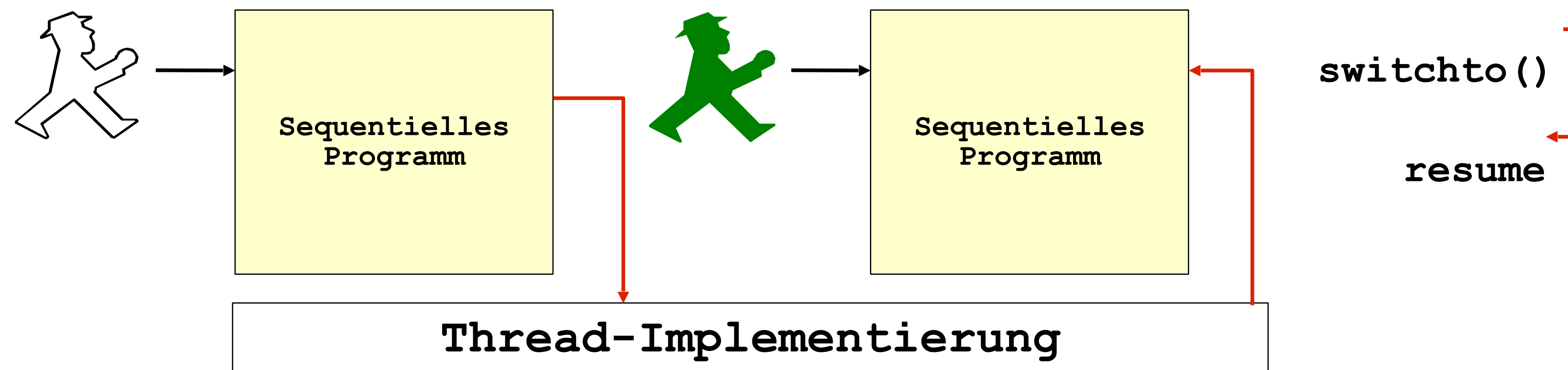
- Zu jedem Zeitpunkt kann auf einer CPU nur ein Thread laufen
- Jeder Thread erhält hin und wieder die CPU:
Thread-Umschaltung



Mehrere Threads auf einem Rechner

Implementierung

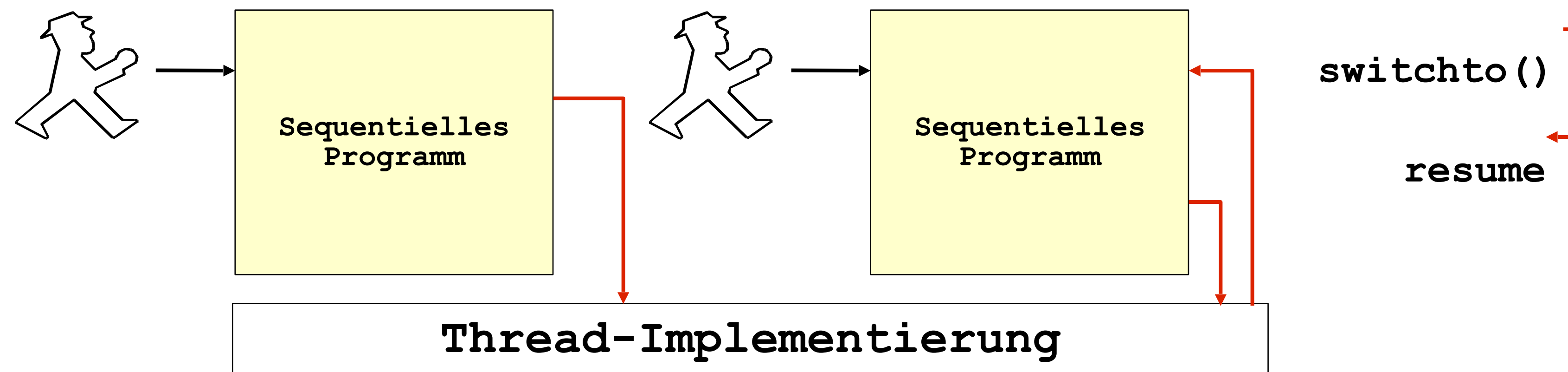
- Zu jedem Zeitpunkt kann auf einer CPU nur ein Thread laufen
- Jeder Thread erhält hin und wieder die CPU:
Thread-Umschaltung



Mehrere Threads auf einem Rechner

Implementierung

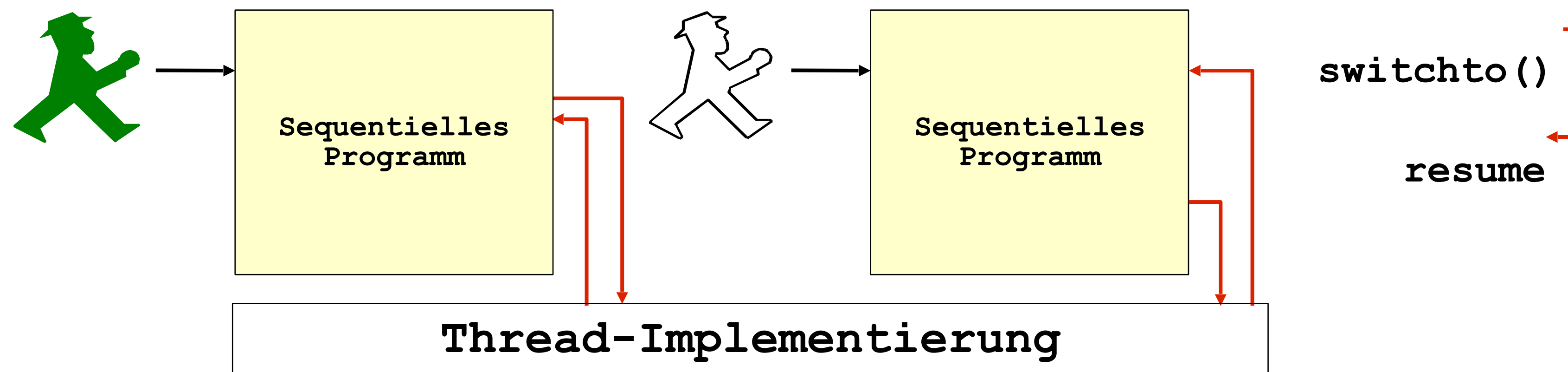
- Zu jedem Zeitpunkt kann auf einer CPU nur ein Thread laufen
- Jeder Thread erhält hin und wieder die CPU:
Thread-Umschaltung



Mehrere Threads auf einem Rechner

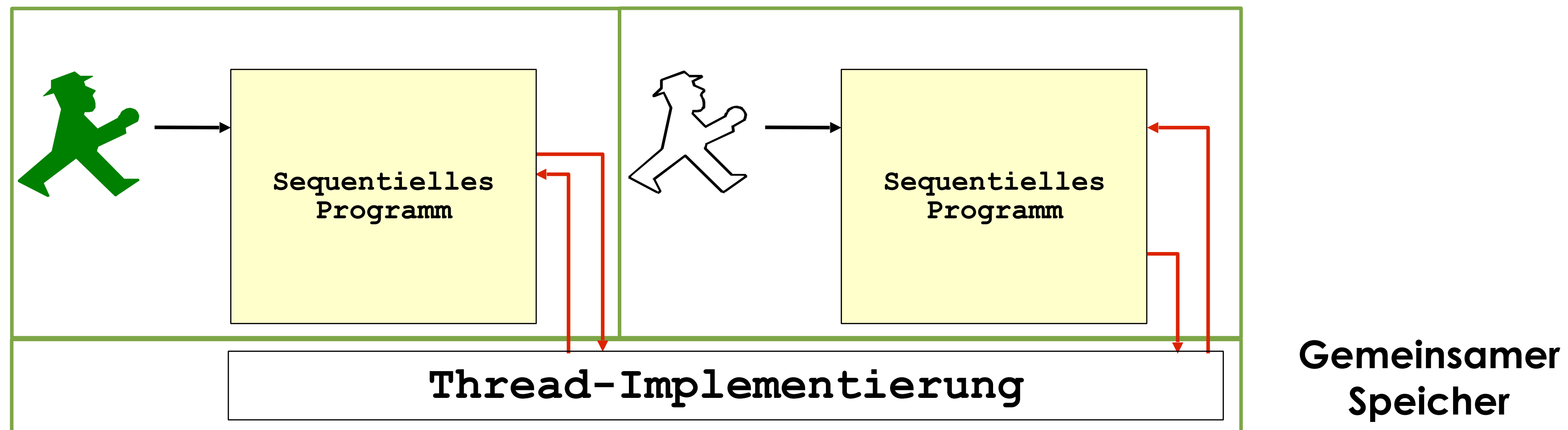
Implementierung

- Zu jedem Zeitpunkt kann auf einer CPU nur ein Thread laufen
- Jeder Thread erhält hin und wieder die CPU:
Thread-Umschaltung



Rechner und Adressräume

- Einfachste Variante: nur ein Adressraum
- Mehrere Adressräume:
 - bei Umschaltung zwischen Threads mit verschiedenen Adressräumen müssen auch diese umgeschaltet werden
 - benötigt HW-Unterstützung (Memory Management Units)



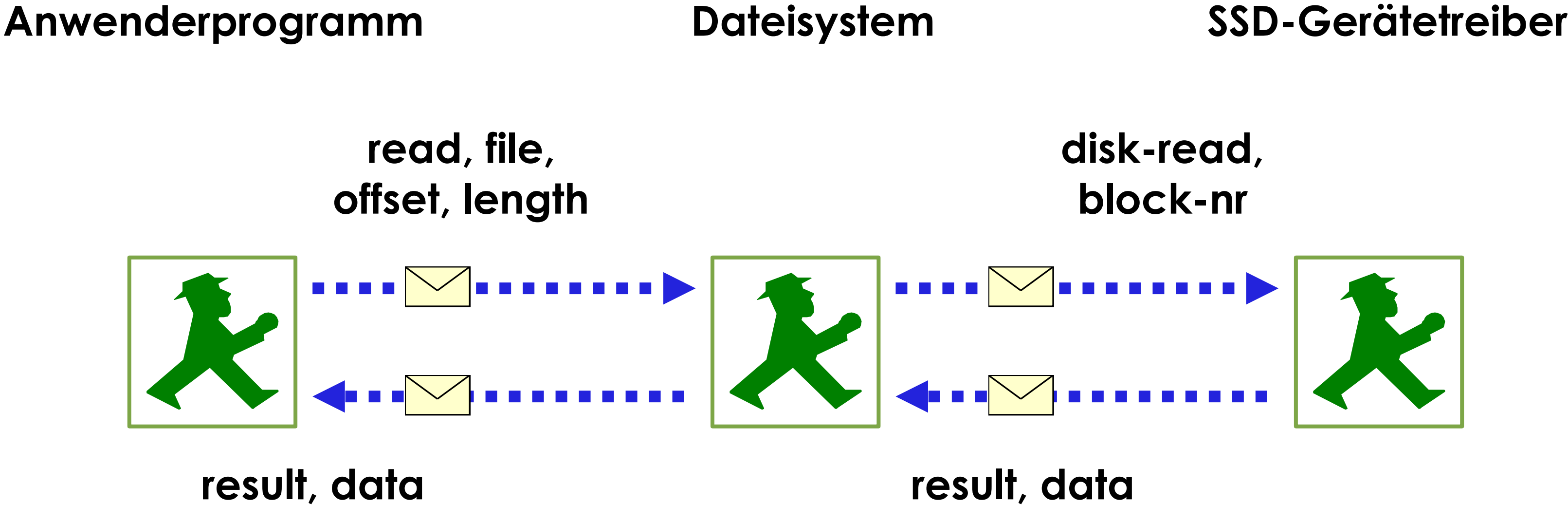
Wozu separate Adressräume?

- Anwenderprogramme sehen bei jeder Ausführung die gleichen Adressen (virtueller Adressraum)
- größere Adressräume für Anwenderprogramme (virtuelle Speicherverwaltung)
- Schutz der Daten verschiedener Prozesse voreinander

Dateien

- Behälter für persistente Speicherung von Daten
- Persistentes Speichermedium (Disk, Flash)
- Operationen:
open, close, read, write, sync
- Typischerweise vom Betriebssystemkern bereitgestellt
- Implementierung auch als Dienst in einem Prozess möglich

Prozesse und Dateien



Definition: Betriebssystemkern

Teil des Betriebssystems, der

- von allen Threads jederzeit genutzt werden kann und
- im privilegierten Modus des Prozessors („kernel mode“) läuft

Enthält mindestens („Mikrokerne“):

- Thread- und Adressraum-Implementierung
- elementare Unterbrechungsbehandlung
- Kommunikationsprimitive

Meistens auch („monolithische“ Systeme):

- Ein-/Ausgabe-Treiber, Protokolle
- Speicherverwaltung, Dateisysteme, ...