

# Systeme I: Betriebssysteme

## **Kapitel 7** **Scheduling**

Wolfram Burgard



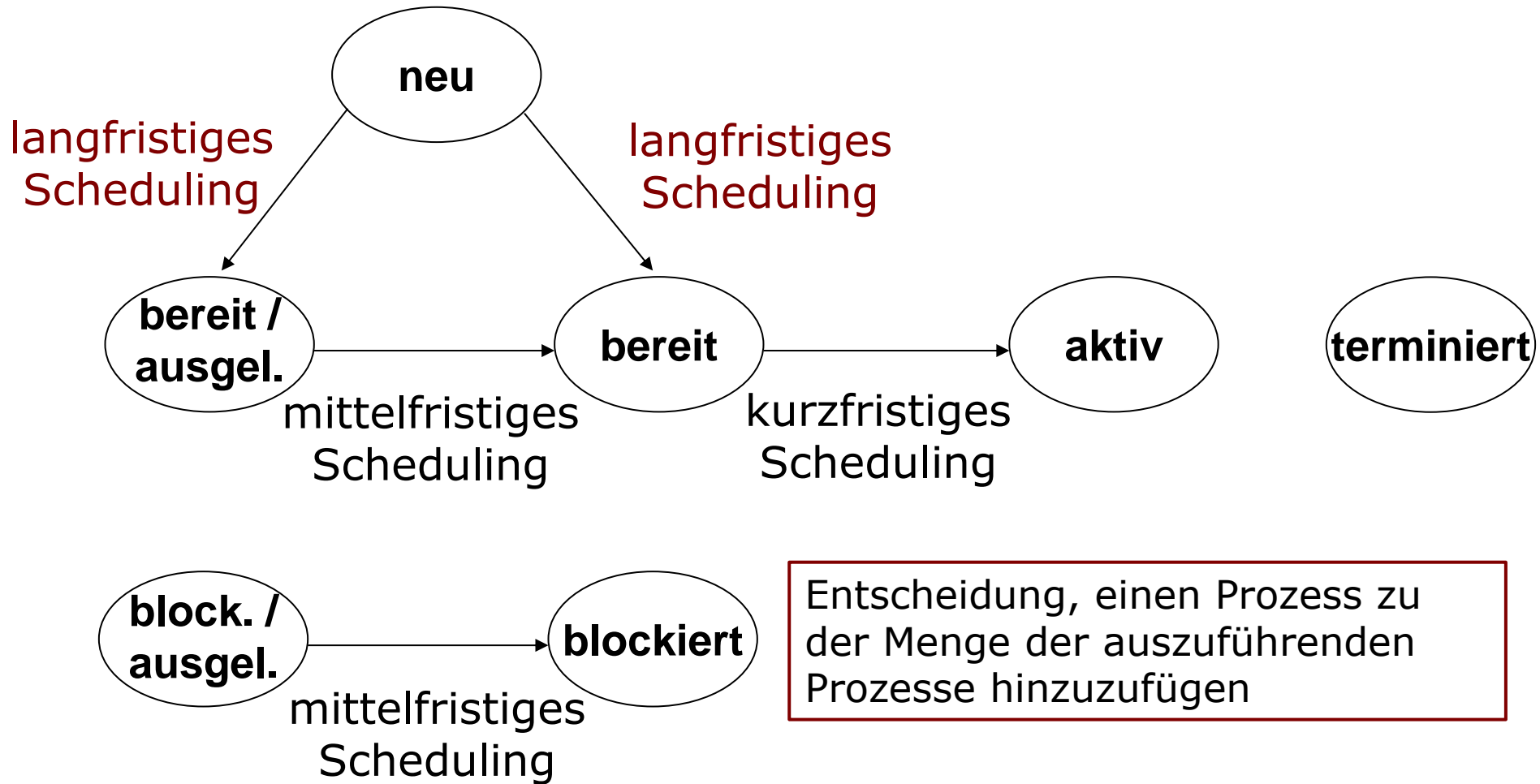
# Inhalt Vorlesung

- Aufbau einfacher Rechner
- Überblick: Aufgabe, Historische Entwicklung, unterschiedliche Arten von Betriebssystemen
- Verschiedene Komponenten / Konzepte von Betriebssystemen
  - Dateisysteme
  - Prozesse
  - Nebenläufigkeit und wechselseitiger Ausschluss
  - Deadlocks
  - **Scheduling**
  - Speicherverwaltung

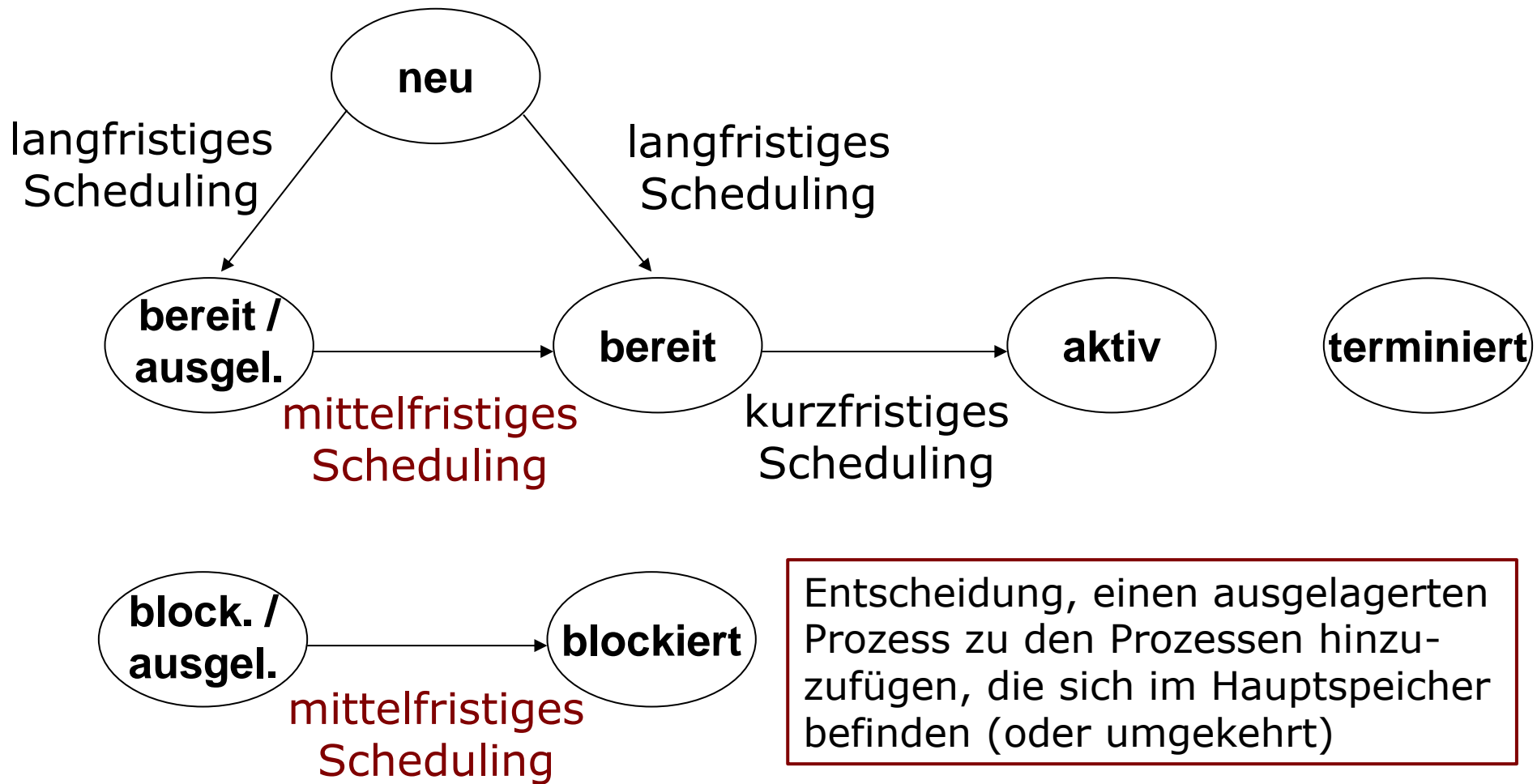
# Einführung

- Mehrprogrammsystem: Im Hauptspeicher werden mehrere Prozesse verwaltet
- Jeder Prozess wird entweder vom Prozessor bearbeitet oder wartet auf ein Ereignis
- Scheduling: Betriebssystem muss entscheiden, welche Prozesse auf den CPU-Kernen Rechenzeit beanspruchen dürfen

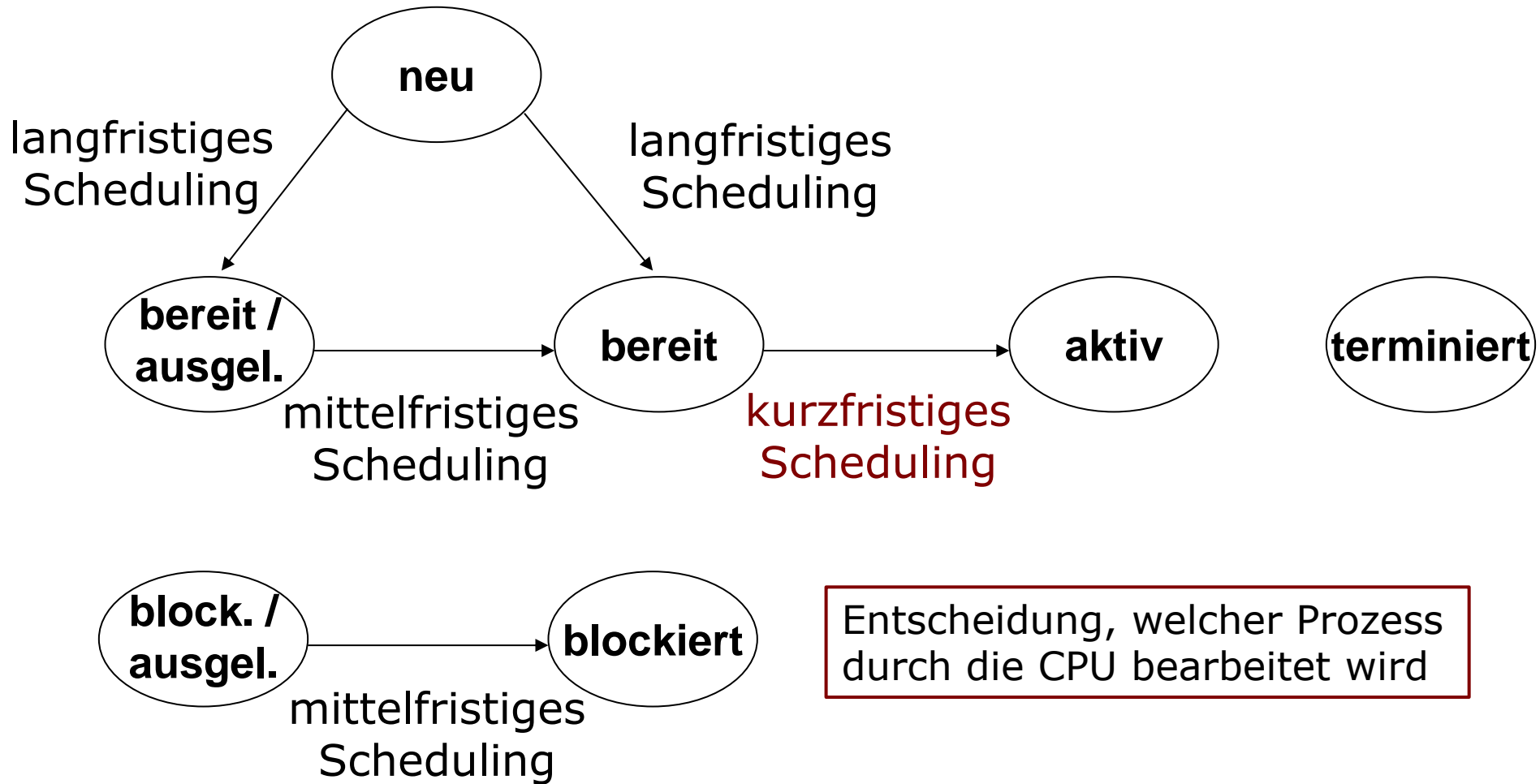
# Drei Arten von Scheduling



# Drei Arten von Scheduling



# Drei Arten von Scheduling



# Kurzfristiges Scheduling

- Rechenzeit wird Prozessen so zugewiesen, dass "optimale Performance" erreicht wird
- Verschiedene Scheduling-Algorithmen existieren für verschiedene Optimierungsziele
- Beachte: Kontextwechsel kosten Rechenzeit

# Kriterien für das kurzfristige Scheduling (1)

- **Benutzerorientiert:**
  - Minimale Antwortzeit bei interaktivem System
  - Minimale Zeit zwischen Eingang und Abschluss eines Prozesses (Durchlaufzeit)
  - Gute Vorhersagbarkeit (unabhängig von Systemauslastung ähnliche Zeit)
- **Systemorientiert:**
  - Maximale Anzahl von Prozessen, die pro Zeiteinheit abgearbeitet werden (Durchsatz, z.B. pro Stunde)
  - Maximale CPU-Auslastung (aktive Zeit)



# Durchsatz vs. Durchlaufzeit

- Durchsatz: Anzahl der Prozesse, die vom System z.B. pro Stunde erledigt werden
- Durchlaufzeit: Zeit von Start bis Abschluss
- Hoher Durchsatz heißt nicht unbedingt niedrige Durchlaufzeit
- Für Benutzer ist eher niedrige Durchlaufzeit interessant

# Kriterien für das kurzfristige Scheduling (2)

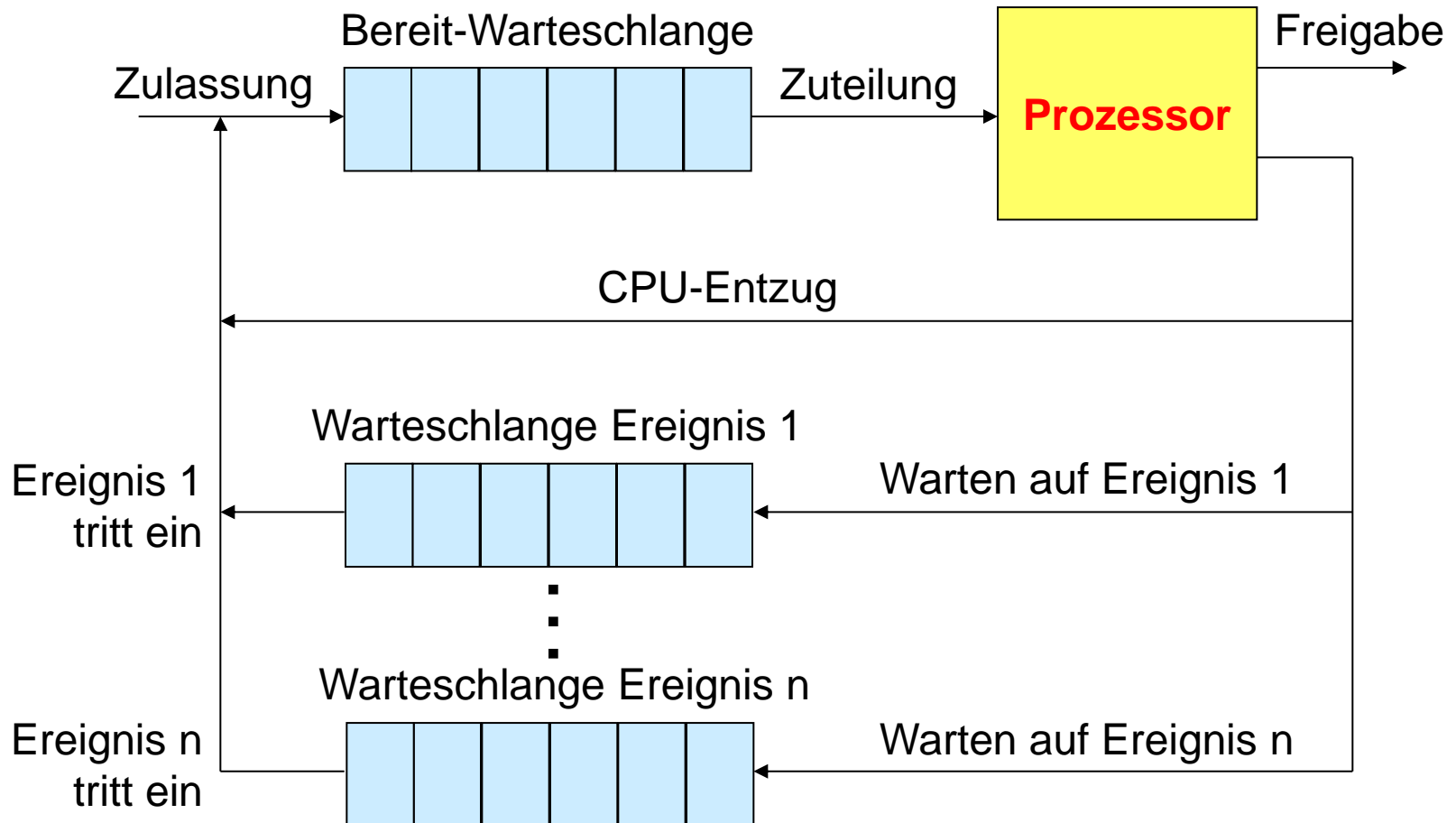
- **Allgemein:**
  - Fairness: Jeder Prozess erhält CPU irgendwann
  - Prioritäten müssen eingehalten werden
  - Effizienz: Möglichst wenig Aufwand für Scheduling selbst
- **Echtzeitsysteme:**
  - Vorhersehbares Verhalten
  - Einhalten von Deadlines

# Kriterien für das kurzfristige Scheduling (3)

- Abhängigkeiten zwischen den Kriterien
- Beispiel:
  - Kurze Antwortzeit: Viele Wechsel zwischen Prozessen
  - Aber dann: Niedrigerer Durchsatz und mehr Aufwand durch Prozesswechsel
- Scheduling-Strategie muss Kompromiss schließen

# Erinnerung: Warteschlangen

Warteschlangen für bereite Prozesse und für Prozesse, die auf Ereignisse warten

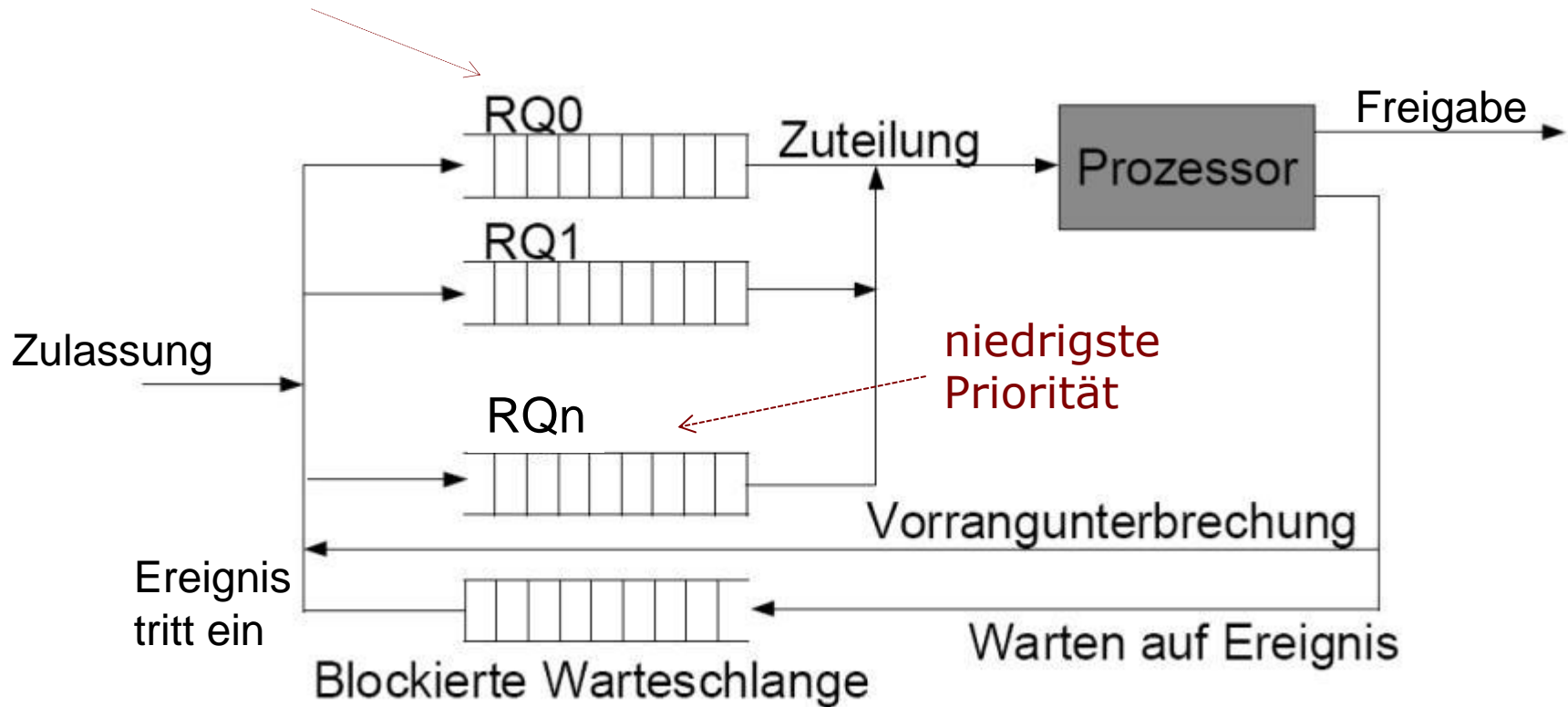


# Prioritäten (1)

- Prioritäten: Mehrere Warteschlangen mit bereiten Prozessen verschiedener Priorität
- Bei Entscheidung der Ablaufplanung: Scheduler beginnt mit der Warteschlange, die bereite Prozesse enthält und die höchste Priorität hat
- Innerhalb Warteschlange: Scheduling-Strategie

# Prioritäten (2)

höchste Priorität



# Prioritäten (3)

- Bereiter Prozess in Warteschlange mit höchster Priorität erhält Rechenzeit
- Problem: Verhungern von Prozessen mit geringer Priorität
- Lösung: Ändere Prioritäten entsprechend Alter (später mehr dazu)

# Scheduling-Algorithmen: Prozessauswahl

- Auswahlfunktion legt fest, welcher der bereiten Prozesse als nächstes aktiv wird
- Basierend auf Prioritäten oder auch Ausführungseigenschaften
- Drei Größen von Bedeutung:
  - $w$  (Wartezeit auf CPU seit Erzeugung)
  - $e$  (bisher verbrauchte CPU-Zeit)
  - $s$  (insgesamt benötigte CPU-Zeit, geschätzt)



# Scheduling-Algorithmen: Zeitpunkt der Auswahlentscheidung

- **Nicht-präemptives Scheduling:**  
CPU kann einem Prozess nur entzogen werden, wenn er beendet oder blockiert ist
- **Präemptives Scheduling:**  
Aktueller Prozess kann vom Betriebssystem unterbrochen werden, wenn dies richtig erscheint

# First Come First Served (FCFS)

- Nicht-präemptive Strategie
- **Strategie:**  
Wenn ein Prozess beendet oder blockiert ist: Bereiter Prozess, der schon am längsten wartet, wird aktiv
- Auswahlfunktion:  $\max(w)$
- Implementiert durch einfache Warteschlange

# Beispiel FCFS

Prozess	Erzeugungszeit	Benötigte Zeit s
P0	0	23
P1	5	3
P2	13	4



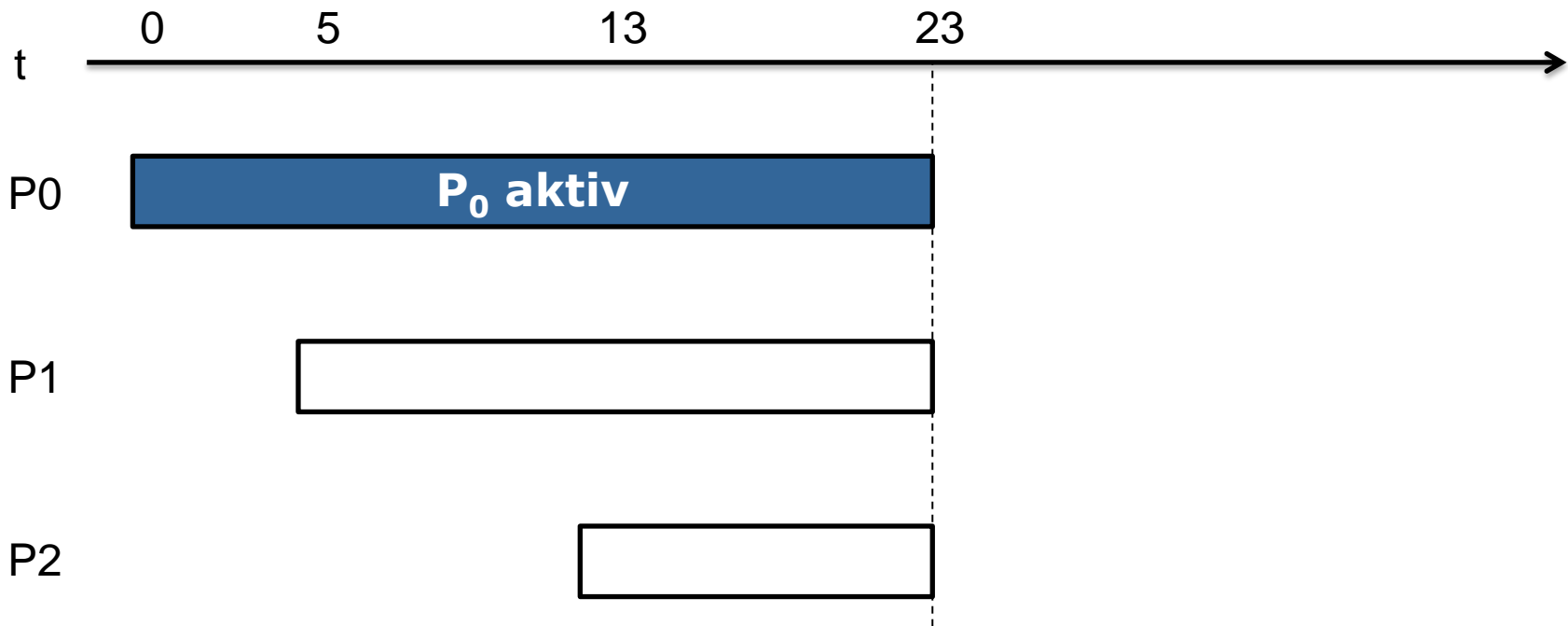
P0

P1

P2

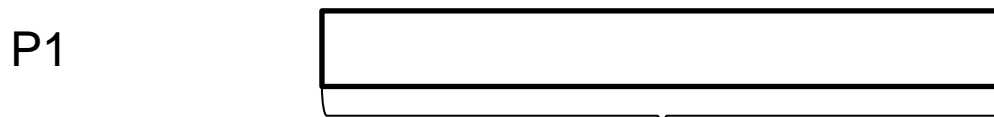
# Beispiel FCFS

Prozess	Erzeugungszeit	Benötigte Zeit s
P0	0	23
P1	5	3
P2	13	4



# Beispiel FCFS

Prozess	Erzeugungszeit	Benötigte Zeit s
P0	0	23
P1	5	3
P2	13	4



$w_1=18$

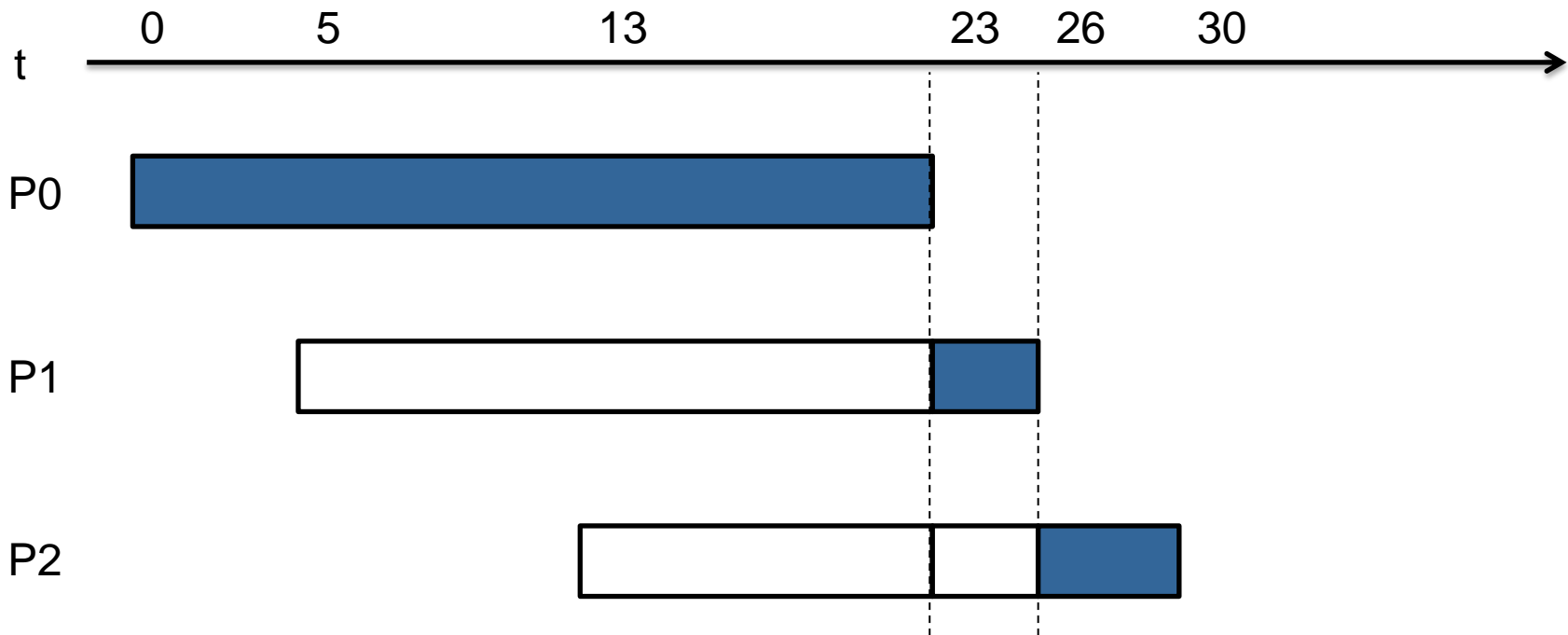


$w_2=10$

Auswahlstrategie:  
 $\max(w)$

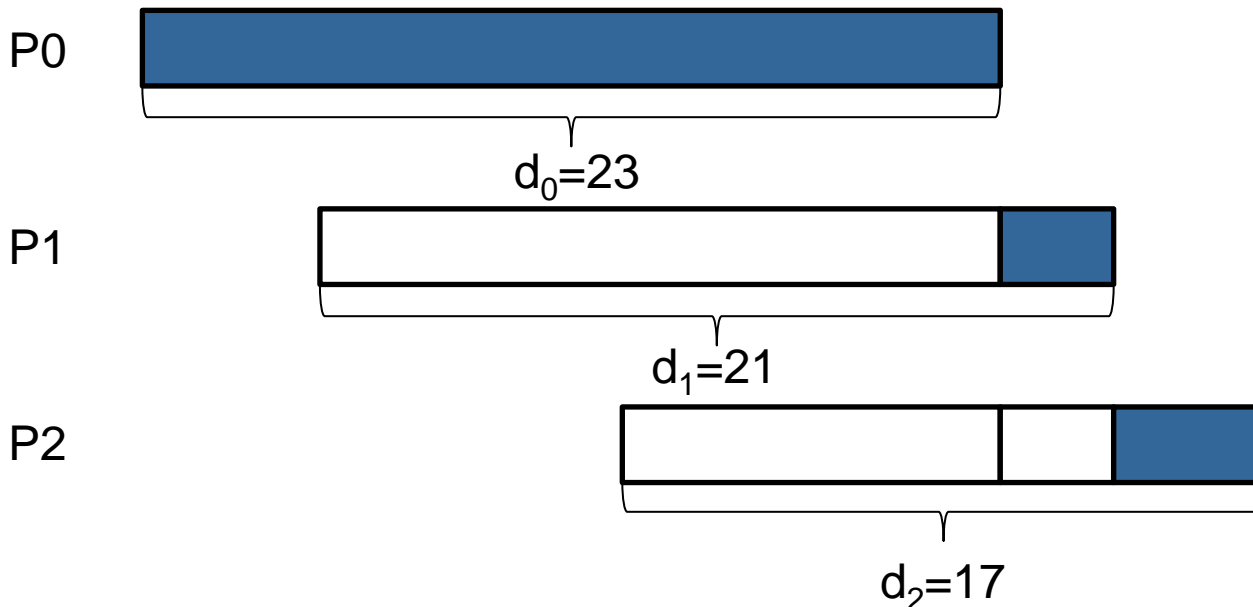
# Beispiel FCFS

Prozess	Erzeugungszeit	Benötigte Zeit s
P0	0	23
P1	5	3
P2	13	4



# Beispiel FCFS - Durchlaufzeiten

Prozess	Erzeugungszeit	Benötigte Zeit s
P0	0	23
P1	5	3
P2	13	4



Mittlere Durchlaufzeit:  
 $d^*=20,3$