

Scheduling

Peter Puschner

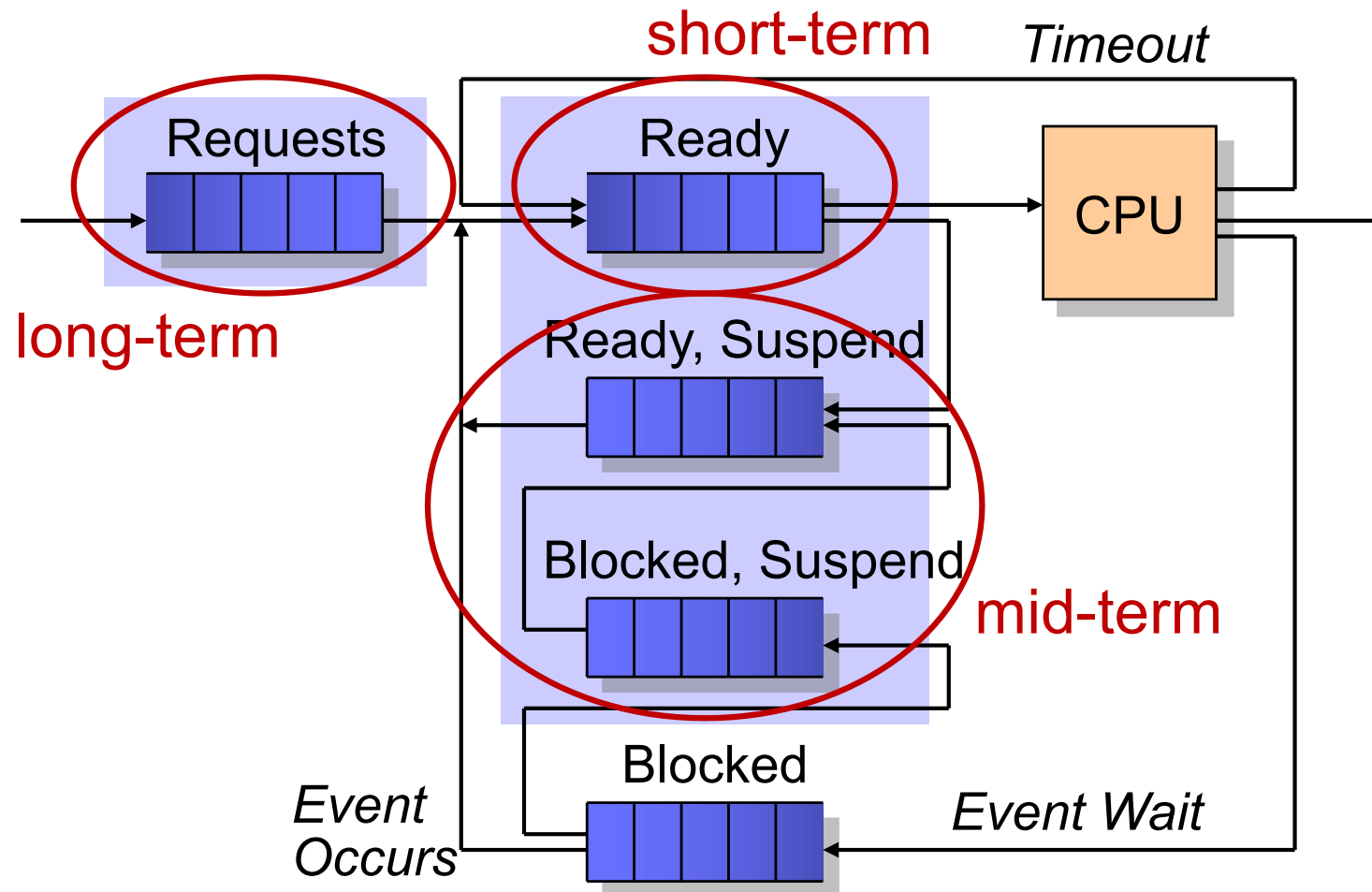
Institut für Technische Informatik

peter@vmars.tuwien.ac.at

Aufgaben des Scheduling

- Bestimmt die Abarbeitungsreihenfolge der Prozesse auf dem Prozessor
- verschiedene Optimierungsziele
 - Durchsatz
 - Prozessorauslastung
 - Fairness
 - Response Time
 - Einhalten von Deadlines
 - ...

Schedulingebenen



Schedulingebenen

- Long-term Scheduling
 - Kreierung von Prozessen
 - bestimmt Grad der Parallelität
 - Mix von CPU- und I/O-intensiven Prozessen
- Medium-term Scheduling
 - Ein- und Auslagern von Prozessen (Memory Management)
- Short-term Scheduling
 - bestimmt nächsten Prozess zur Ausführung

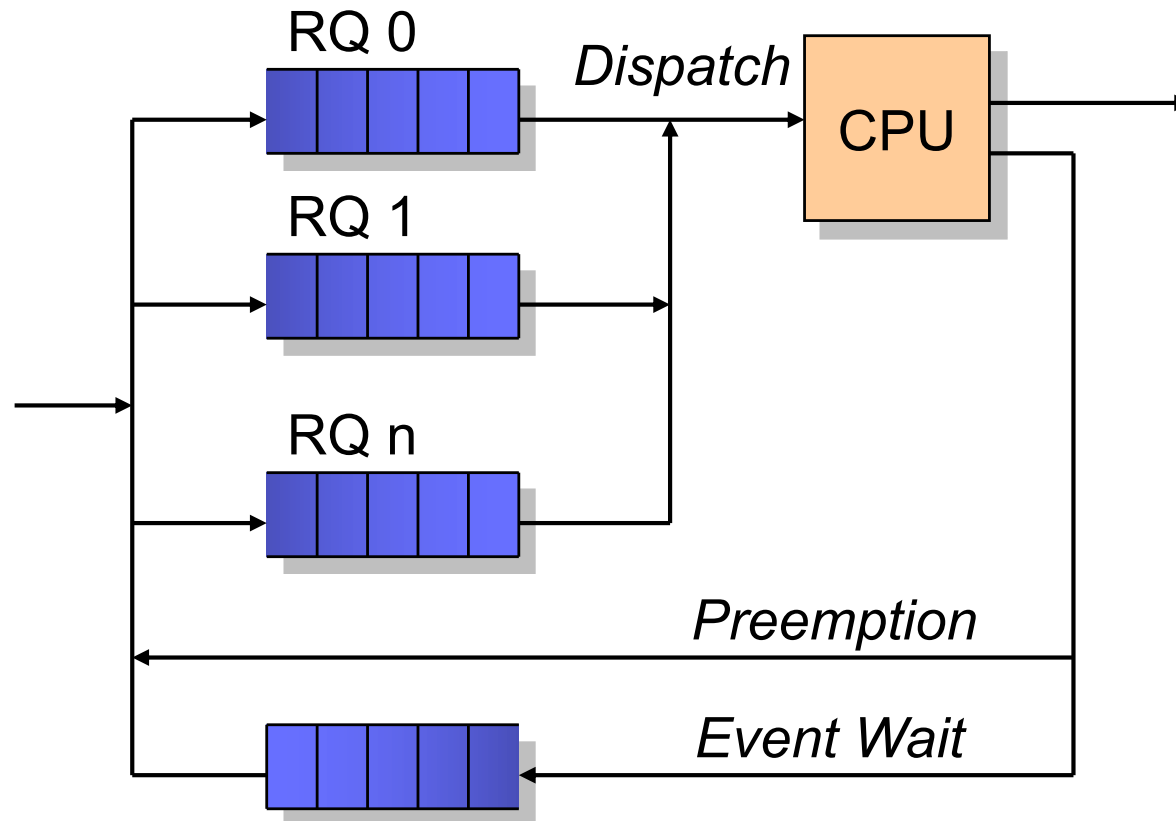
Short Term Scheduling

- Welcher Prozess soll als nächster ausgeführt werden?
- *CPU Scheduler* oder *Dispatcher*
- Aktivierung des Dispatchers, wenn Prozessumschaltung angebracht sein kann:
 - System Calls und Traps
 - I/O-Interrupt, Signale
 - Uhr-Interrupt

Scheduling-Kriterien

	<i>User-Oriented</i>	<i>System-Oriented</i>
<i>Performance</i>	Response Time, Turnaround Time, Deadlines	Throughput, Processor Utilization
<i>Other</i>	Predictability	Fairness, Resource Balance, Priorities

Verwendung von Prioritäten



- Achtung: Starvation!

Schedulingstrategien

- **Selection Function**

Auswahl des nächsten auszuführenden Prozesses

- bisherige Verweildauer
- bisherige/gesamte Ausführungszeit
- Fertigstellungszeitpunkt (Deadline)

- **Decision Mode**

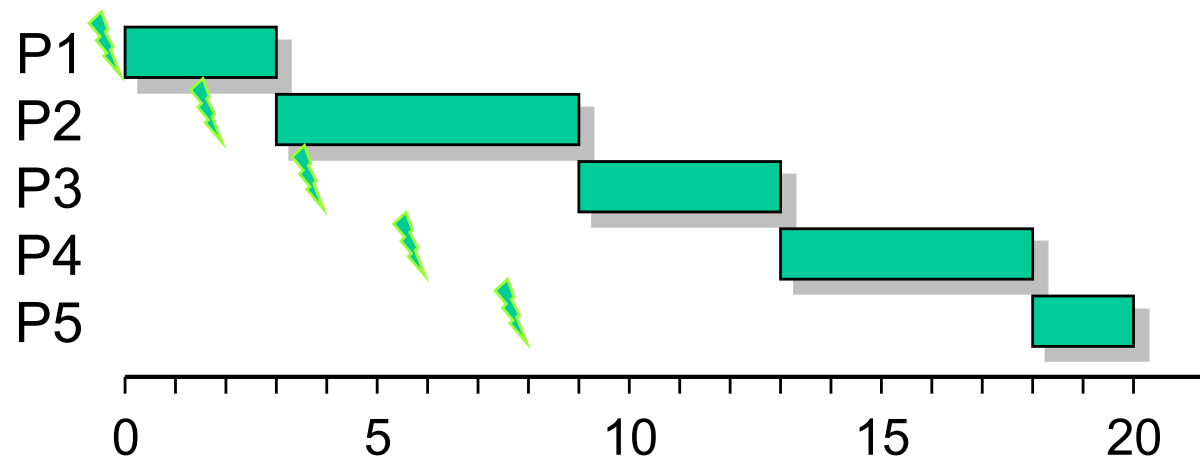
- Non-preemptive: keine externe Unterbrechung von Prozessen durch BS
- Preemptive: Unterbrechung von Prozessen durch BS möglich


Task-Set für Beispiele

<i>Prozess</i>	<i>Arrival Time</i>	<i>Service Time</i>
<i>P1</i>	0	3
<i>P2</i>	2	6
<i>P3</i>	4	4
<i>P4</i>	6	5
<i>P5</i>	8	2

First Come First Served (FCFS)

- Selection Function: Auswahl des Prozesses, der bereits am längsten in der Ready Queue verweilt
- Decision Mode: non-preemptive



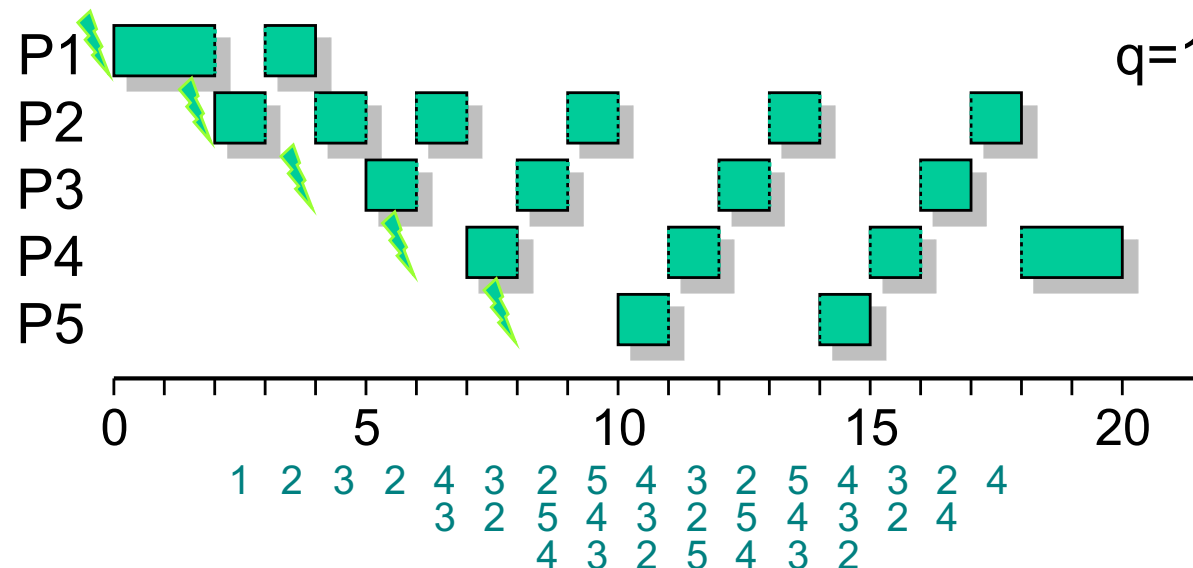
 ... task arrival

First Come First Served

- begünstigt:
 - lange Prozesse (Normalized Turnaround Time)
 - CPU-intensive Prozesse
(CPU-Monopolisierung durch Prozesse ohne I/O)
- schlechte Auslastung von CPU und I/O
- selten „pures“ FCFS

Round Robin (RR, Time Slicing)

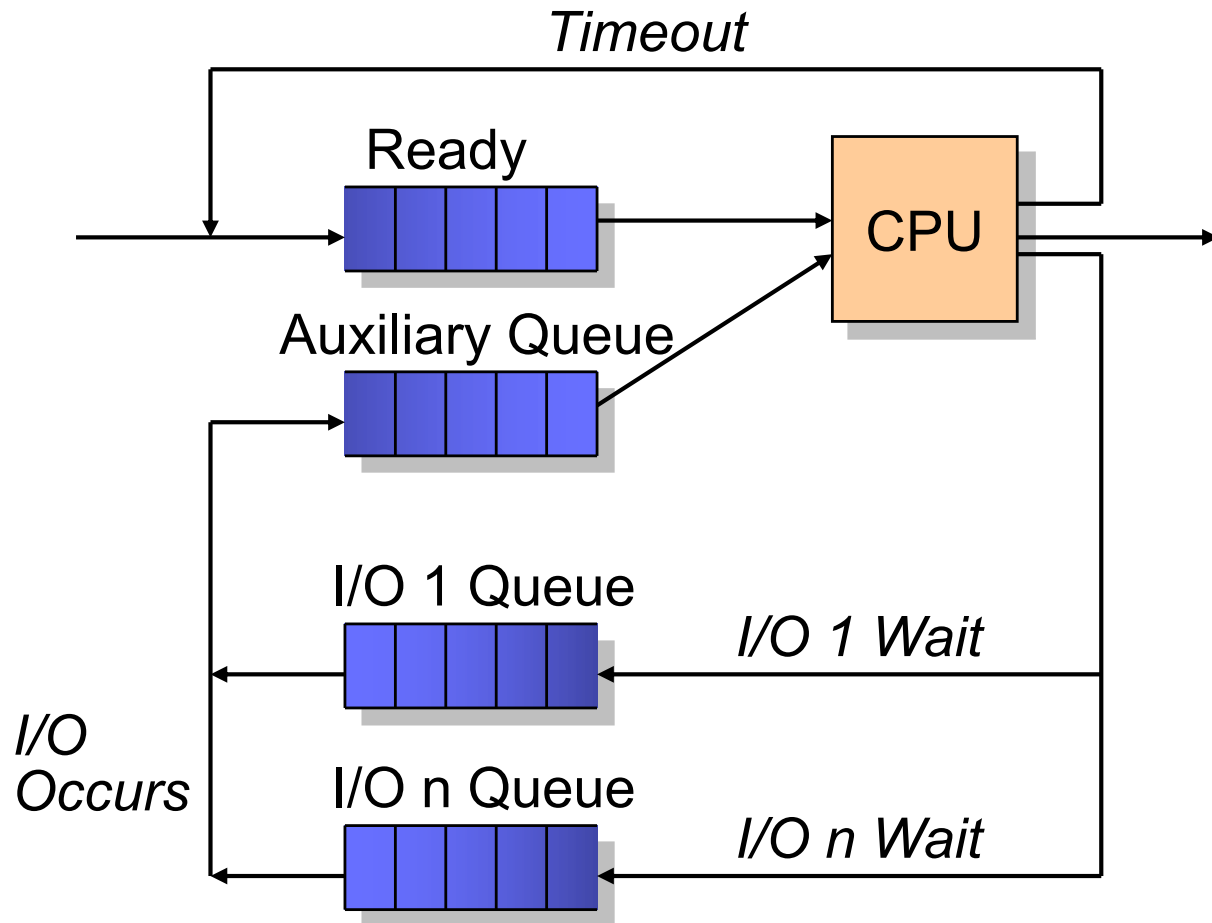
- Selection Function: siehe FCFS
- Decision Mode: preemptive
- Zeitscheiben gleicher Länge werden zyklisch an Prozesse vergeben (Clock Interrupt)



Round Robin

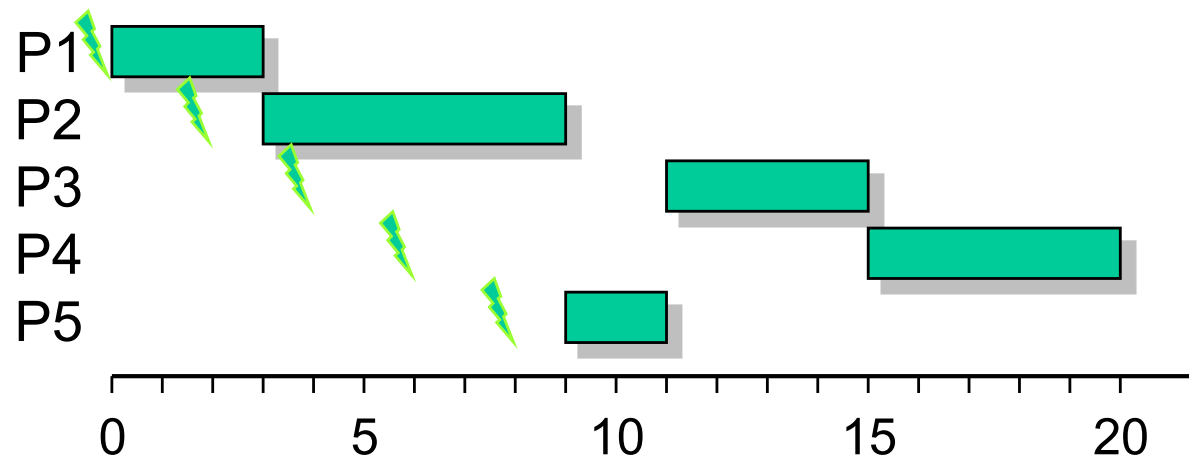
- sinnvolle Zeitscheibenlänge:
 - viel länger als Clock Interrupt + Scheduling
 - etwas länger als eine typische Interaktion
 - benachteiligt I/O-intensive Prozesse
 - I/O intensive Prozesse schöpfen Zeitscheiben nicht voll aus; werden während des Blockierens von CPU-intensiven Prozessen „überholt“
- ⇒ Virtual Round Robin
- ⇒ Auxiliary Queue mit höherer Priorität

Virtual Round Robin



Shortest Process Next (SPN)

- Selection Function: Prozess mit kürzestem erwarteten CPU-Burst zuerst
- Decision Mode: non-preemptive



Shortest Process Next

- bessere Response Times als FCFS
- größere Variabilität der Response Times
 - Verzögerung langer CPU-intensiver Prozesse
- Probleme:
 - Schätzung der Abarbeitungszeiten
 - Starvation langer Prozesse
 - nicht für interaktiven Betrieb geeignet (CPU-intensiver Prozess, der früh ankommt, kann CPU monopolisieren)

Shortest Remaining Time (SRT)

- Selection Function: wie SPN
- Decision Mode: preemptive
- kürzere Prozesse werden fair behandelt
- nicht so viele Interrupts wie bei RR
- Protokollieren der Service Times notwendig
- Starvation möglich

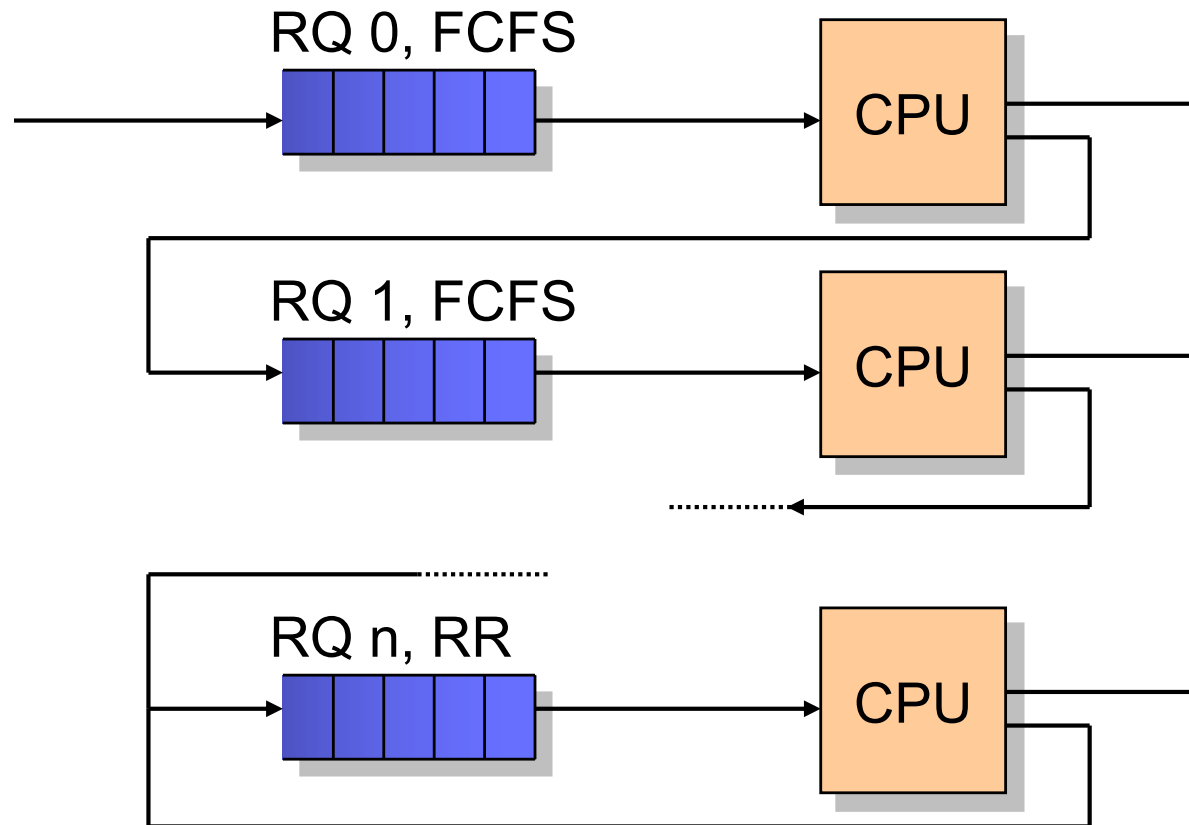
Highest Response Ratio Next

- Selection Function
 - $RR = (w+s) / s$
 - w ... gesamte bisherige Wartezeit
 - s ... geschätzte Service Time
 - Auswahl des Prozesses mit größtem RR
- Decision Mode: non-preemptive
- Eigenschaften:
 - kürzere Prozesse werden fair behandelt
 - keine Starvation
 - Schätzung der Service Times notwendig

Feedback Scheduling

- Selection Function: basiert auf bisheriger Ausführungszeit
 - je mehr CPU Time ein Prozess bisher konsumiert hat, desto niedriger wird seine Priorität (dafür längere Zeitscheiben)
 - eigene Queue für jede Prioritätsstufe
- Decision Mode: preemptive
- Starvation möglich
 - Abhilfe: Anheben der Priorität nach best. Zeit

Feedback Scheduling (2)



Real-Time Scheduling

- Deadlines
- „fast is not real-time“
- Deadlines: hard versus soft real-time
- Prozesse oft periodisch

- Scheduling: typischer Weise preemptive, statische versus dynamische Prioritäten
- Schedulability Tests: Überprüfung, ob alle Tasks rechtzeitig beendet werden können.

Earliest Deadline First (EDF)

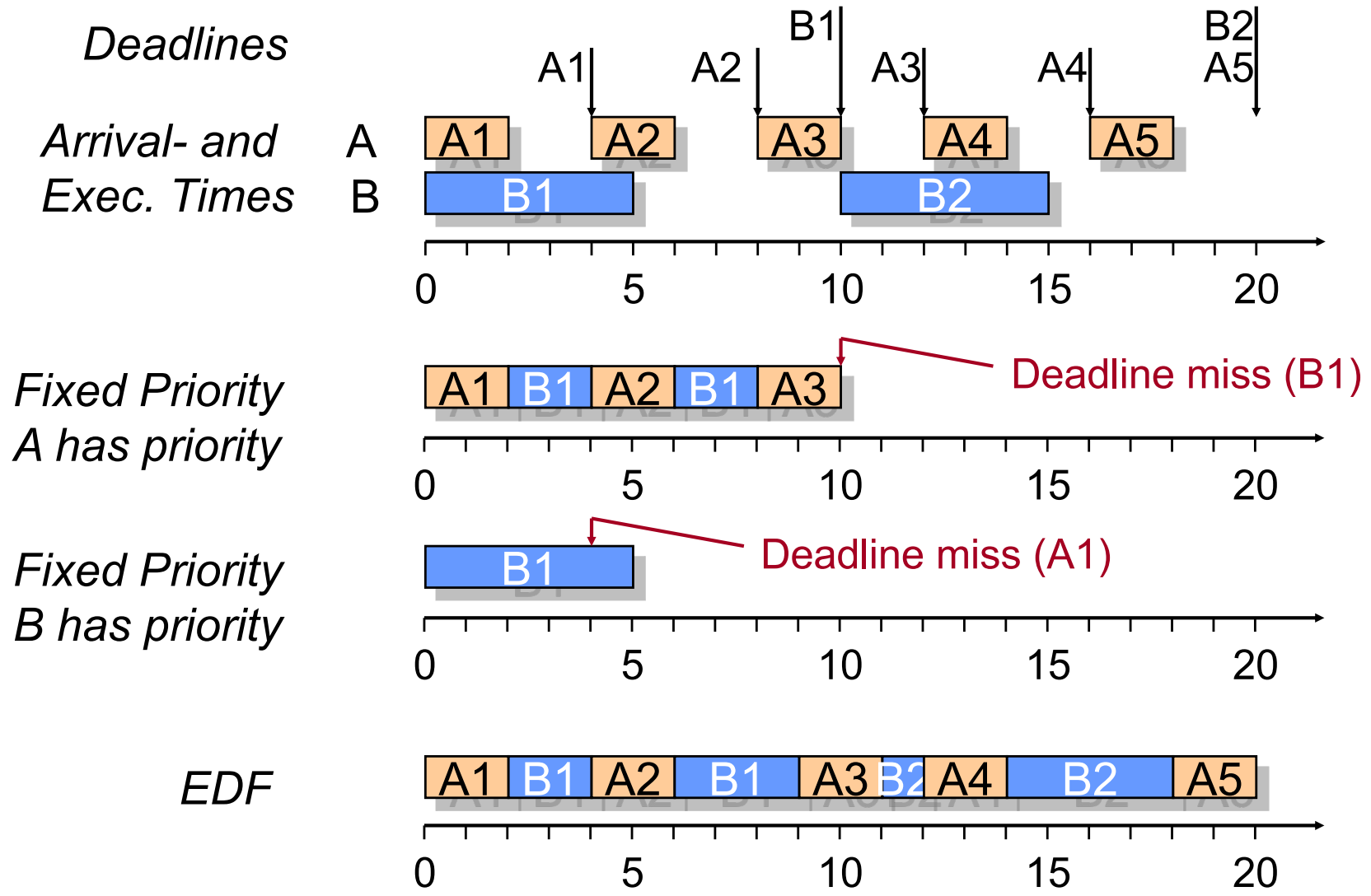
- Selection Function: Task mit frühester Completion Deadline zuerst
- Decision Mode: preemptive
- Minimiert Anzahl der Deadline Misses
- Schedulability Test

Task Deadline = Periode T_i , Ausführungszeit C_i

Notwendige und hinreichende Bedingung:

$$\sum C_i / T_i \leq 1$$

Beispiel: fixe Prioritäten vs. EDF



Zusammenfassung

- Scheduling erfolgt auf verschiedenen Ebenen
- Eine Vielzahl von Kriterien bestimmen die Wahl der Schedulingstrategie
 - benutzerorientiert vs. systemorientiert
 - quantitativ vs. qualitativ
- Schedulingstrategien
 - FCFS, RR, SPN, SRT, HRRN, Feedback (Laufzeit, Verweildauer, I/O-Verhalten)
 - Real-Time Scheduling: EDF