



SYSGO
REAL-TIME SOLUTIONS

Ein Scheduler für alle Fälle

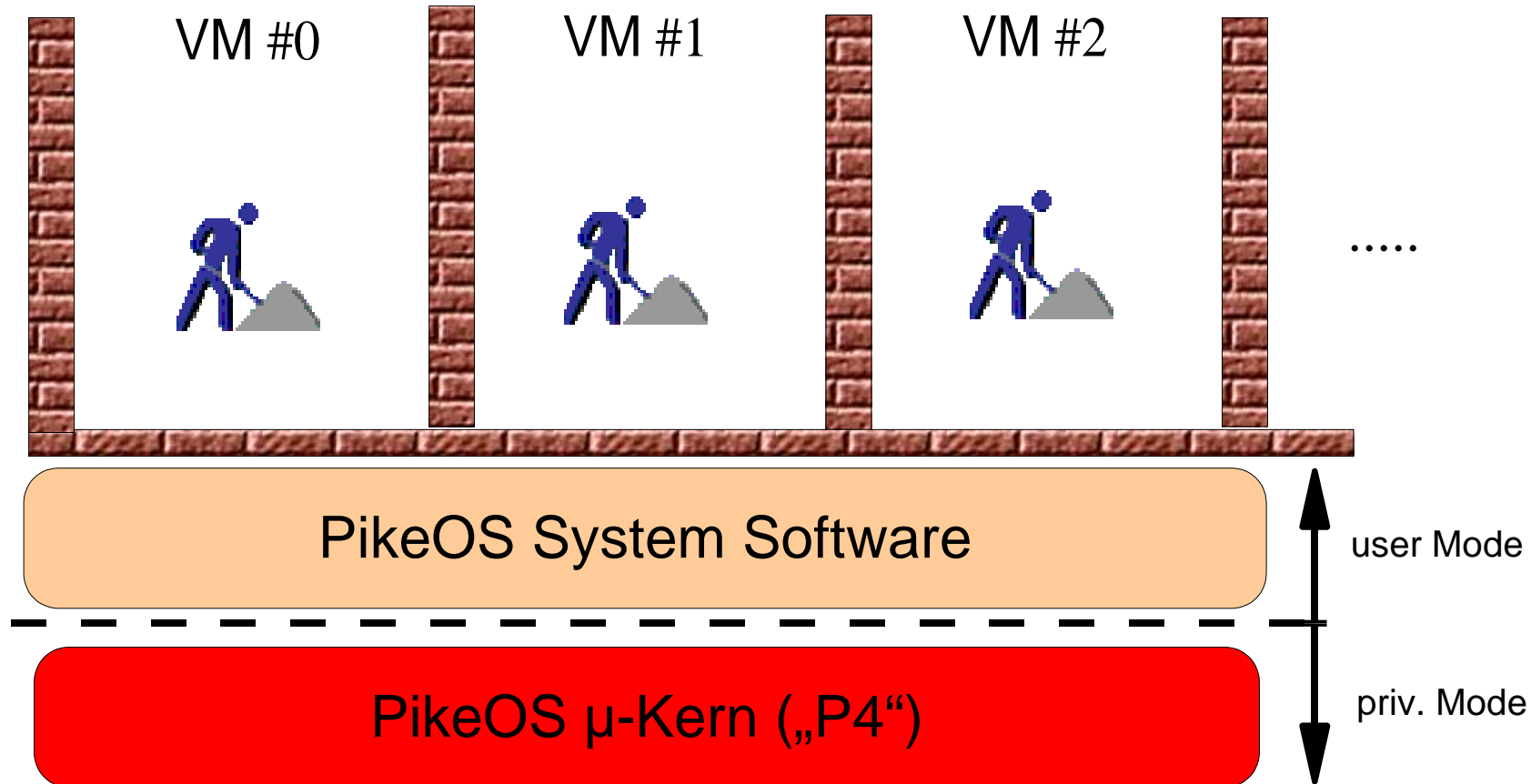
Robert Kaiser, SYSGO AG

Am Pfaffenstein 14
D-55270 Klein-Winternheim
Tel. +49 (0) 6136 9948-0
Fax. +49 (0) 6136 9948-10

PikeOS: multiple VM Umgebung



SYSGO
REAL-TIME SOLUTIONS



PikeOS: multiple VM Umgebung



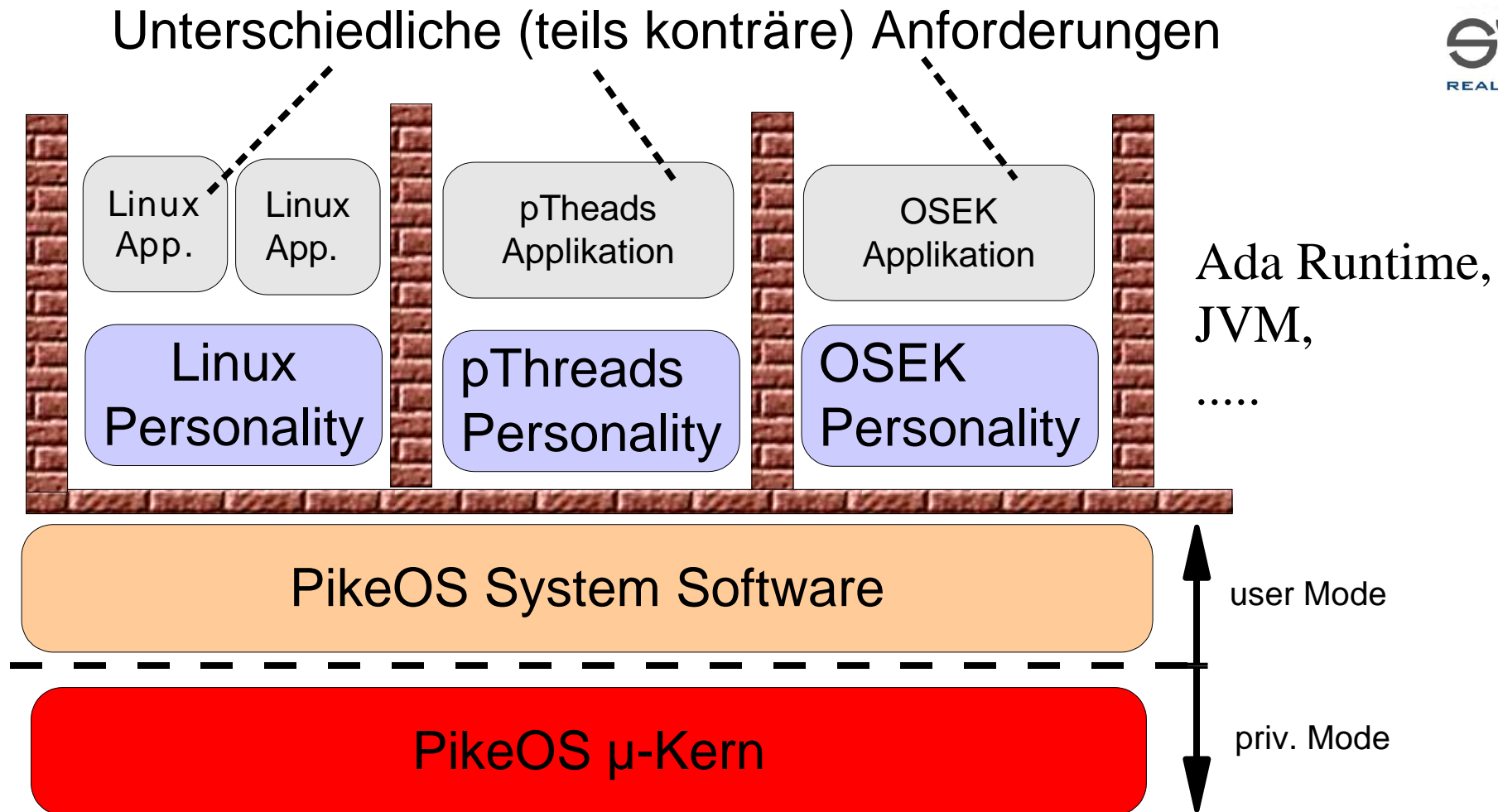
SYSGO
REAL-TIME SOLUTIONS

- ❑ **Statische Verteilung der *Betriebsmittel* auf VMs:**
 - Speicher
 - Zugriff auf Geräte (Register/Interrupts)
 - Kern-Speicher
 - Rechte (Inter-VM Kommunikation, Priv. Aufrufe, ..)
 - Rechenzeit
- ❑ **Dadurch sichere Entkopplung der VMs**
- ❑ **Minimale Trusted Code Base:**
 - μ Kern
 - System Software

PikeOS: Personalities



SYSGO
REAL-TIME SOLUTIONS





SYSGO
REAL-TIME SOLUTIONS

Vielfalt an Zeitanforderungen

□ Nicht-Echtzeit (z.B. Linux)

- keine Fristen (Deadlines)
- „faire“ Prozessorzuteilung
- möglichst gute Prozessorausnutzung

□ Harte Echtzeit (z.B. pThreads, OSEK OS):

- Fristen sind einzuhalten

- 2 Unterklassen:

1) Zeitgetrieben/Periodisch

- Zyklisch wiederholter Rechenbedarf (Zeitpunkte bekannt)
- worst-case Ausführungszeit bekannt

2) Ereignisgetrieben

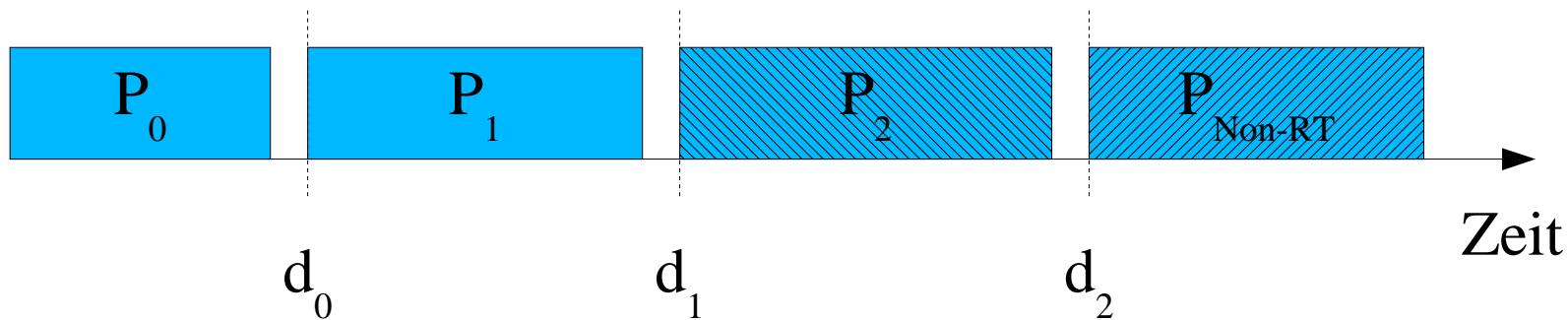
- Rechenbedarf entsteht durch externe Ereignisse (Zeitpunkte unbekannt)
- worst-case Ausführungszeit bekannt



SYSGO
REAL-TIME SOLUTIONS

Zeitgetriebenes Scheduling

- ❑ **Feste Zeitschlitz für Prozesse**
(i.A. zyklische Wiederholung der Zeitschlitz in einem Zeitrahmen)
- ❑ **so bemessen, daß Zeitanforderungen eingehalten werden**
- ❑ **Berücksichtigung von ereignisgetriebenen Prozessen für den worst case Fall (d.h. Alle Ereignisse treten ein)**
- ❑ **Weitere, feste Zeitschlitz für Nicht-Echtzeitprozesse**



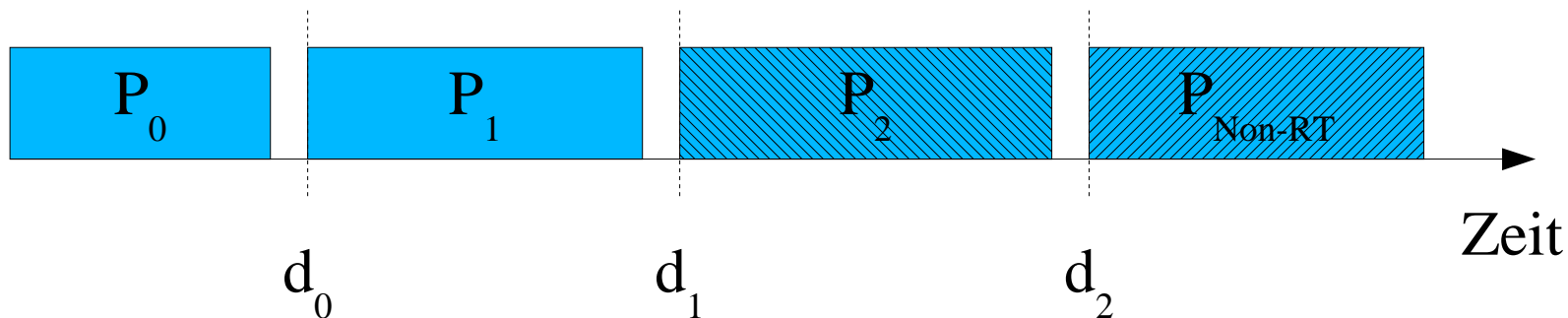


SYSGO
REAL-TIME SOLUTIONS

Zeitgetriebenes Scheduling

□ Vorteile/Nachteile:

- + ideal für zeitgetriebene Prozesse
- + wohlgeordneter Ablauf (keine überflüssigen Kontextwechsel)
- + Zeit jedes Prozesses individuell begrenzt
- hohe worst-case Antwortzeit für ereignisgetriebene Prozesse
- für ereignisgetriebene Prozesse: schlechte Prozessorausnutzung

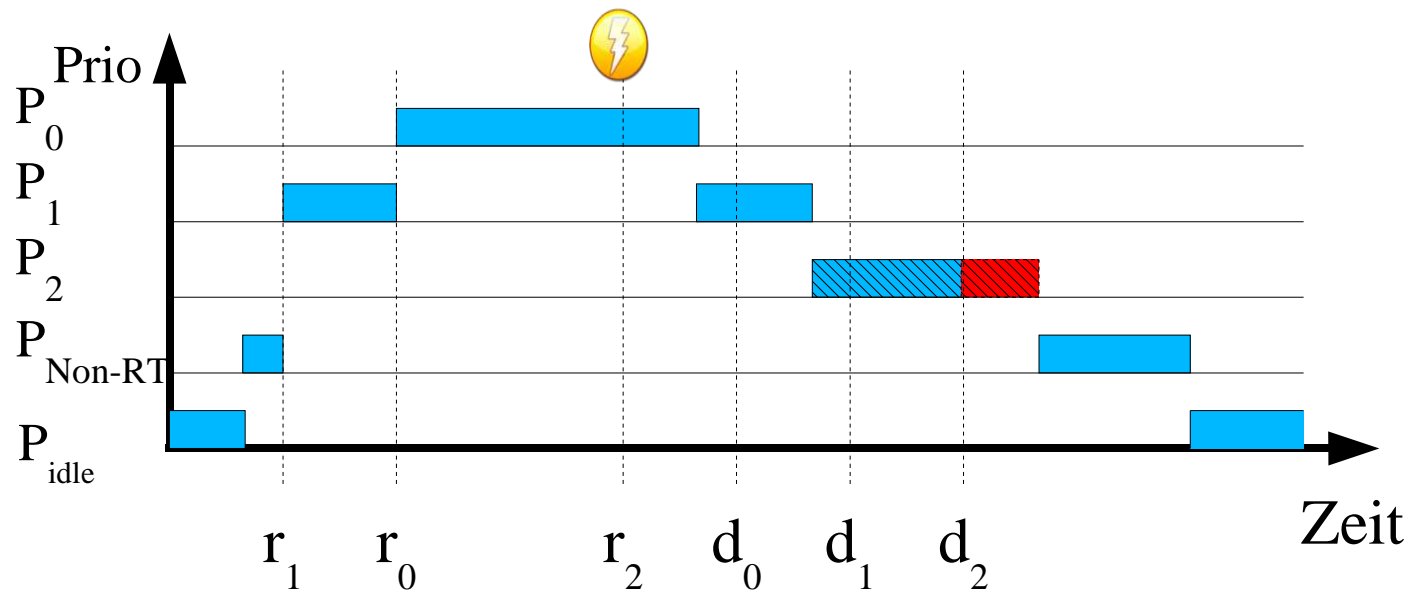


Prioritätsverfahren



SYSGO
REAL-TIME SOLUTIONS

- (z.B. RTAI, RTLinux)
- Zuteilung des Prozessors anhand von Prioritäten
 - Prioritäten entsprechend der Frist, z.B. kurze Deadline -> hohe Priorität
- niederpriorie Prozesse durch hochpriorie unterbrechbar
- niedrigste Priorität für Nicht-Echtzeitprozesse



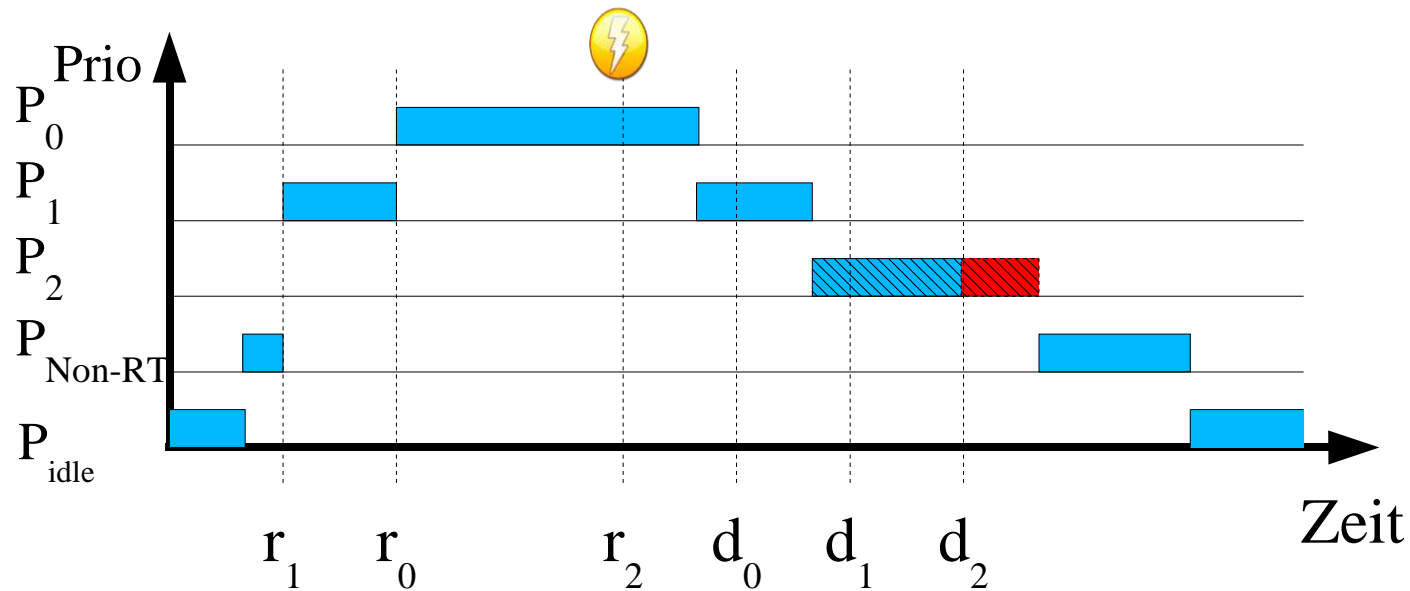
Prioritätsverfahren



SYSGO
REAL-TIME SOLUTIONS

□ Vorteile/Nachteile:

- + Prozessor für hochpriorie Prozesse stets verfügbar
- + von Echtzeitprozessen nicht genutzte Rechenzeit ist für Nicht-Echtzeitprozesse nutzbar -> gute Prozessorauslastung
- Overhead durch überflüssige Kontextwechsel
- Hochpriorie Prozesse können den Prozessor beliebig lange belegen



Prioritätsverfahren



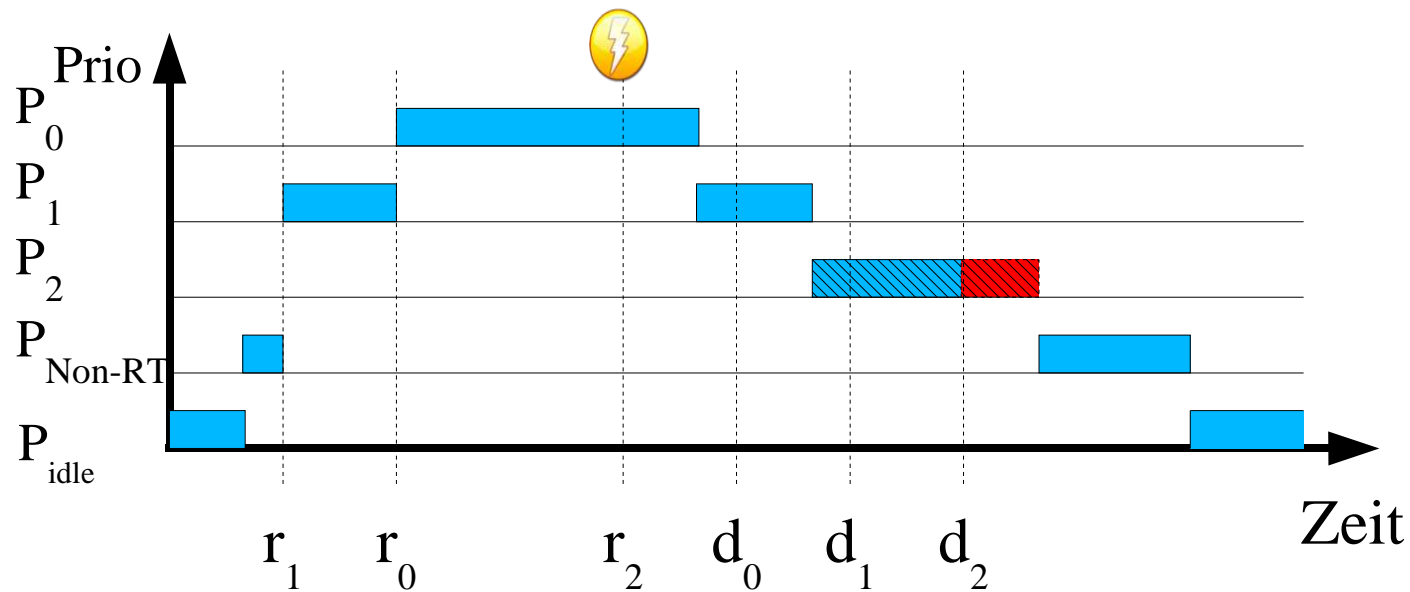
SYSGO
REAL-TIME SOLUTIONS

□ „Vertrauensproblem“:

Prozess gegebener Priorität muss allen höher priorisierten Prozessen vertrauen (DoS-Attacken möglich)

=> „Zwangskopplung“ von Priorität und Vertrauenswürdigkeit.

□ Für PikeOS nur innerhalb einer VM akzeptabel!

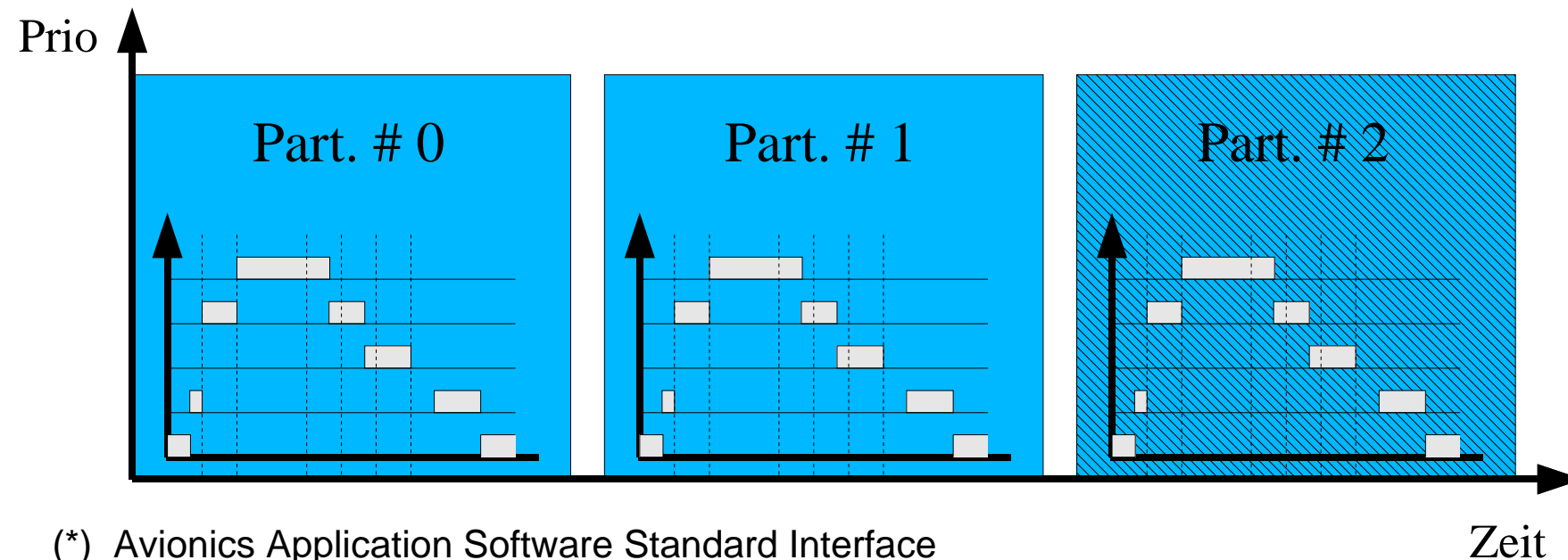




SYSGO
REAL-TIME SOLUTIONS

ARINC 653(*) Scheduler

- Zweistufiges Verfahren
- Zeitschlitz („Zeitpartitionen“), in denen prioritätsgesteuert Gruppen von Prozessen arbeiten



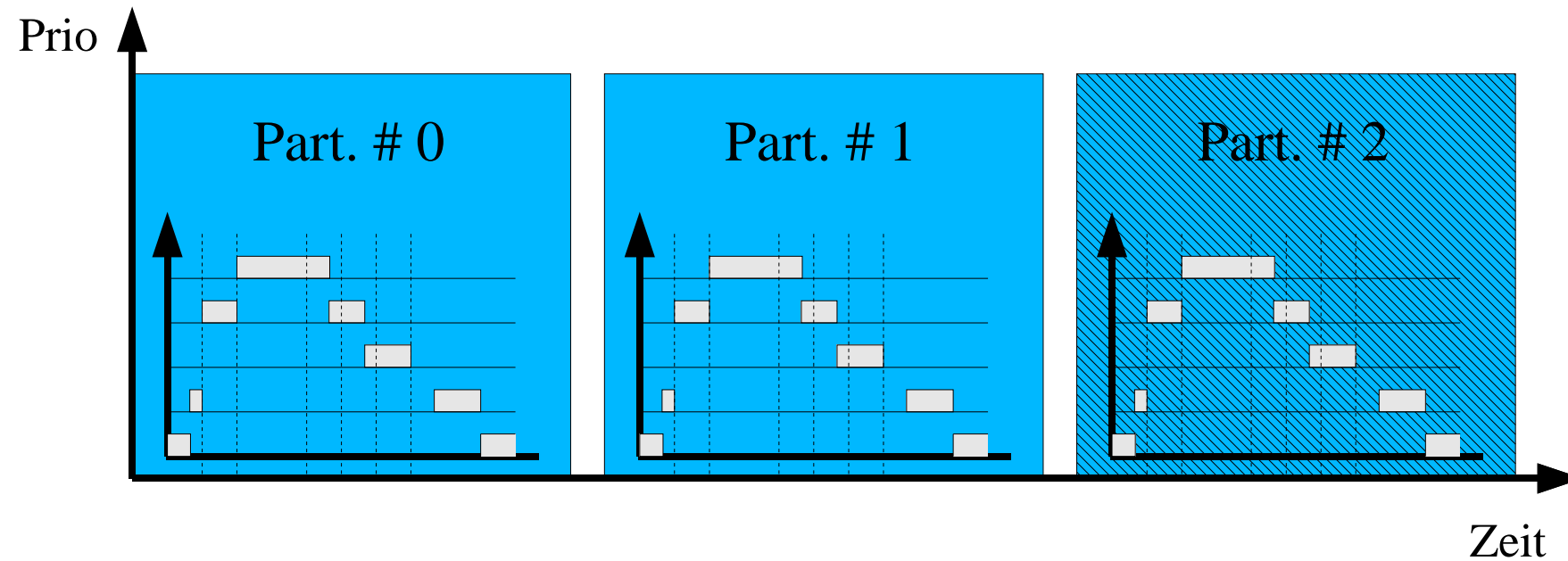
(*) Avionics Application Software Standard Interface
ARINC Specification 653, January 1997.



SYSGO
REAL-TIME SOLUTIONS

ARINC 653 Scheduler

- ❑ **Sichere Entkopplung der Zeitpartitionen**
- ❑ **Aber:**
 - **Schlechte Prozessorausnutzung**
 - **Hohe worst case Antwortzeit**



PikeOS-Verfahren



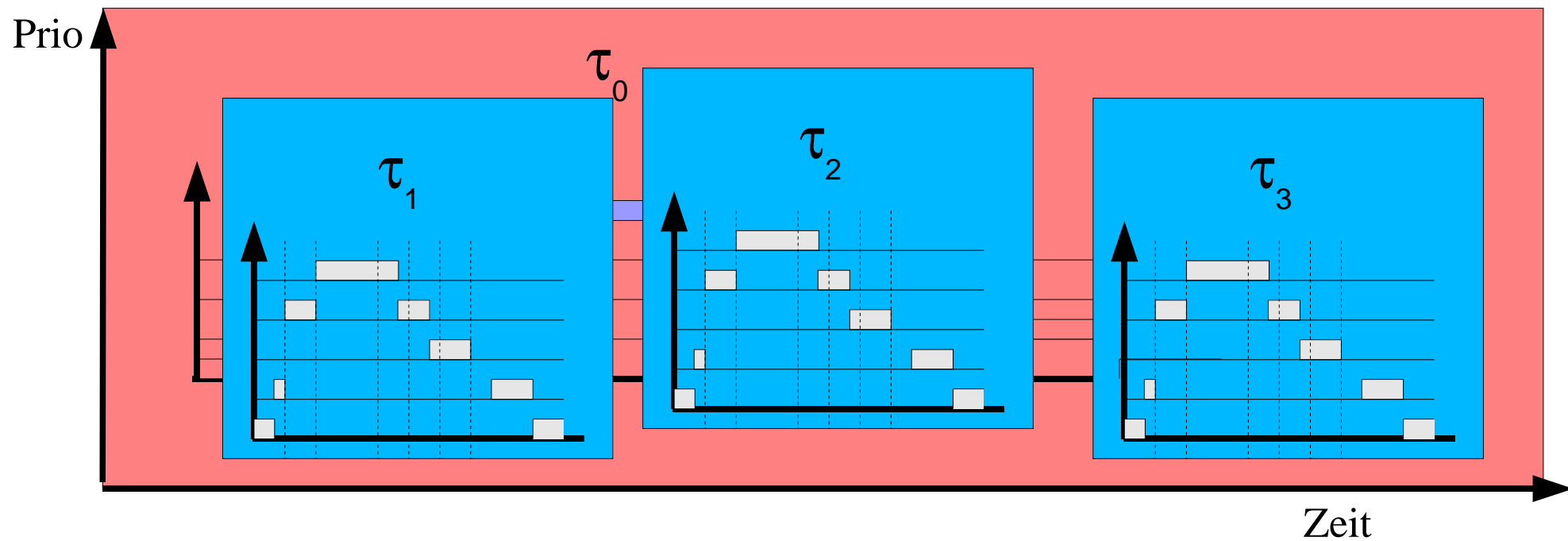
SYSGO
REAL-TIME SOLUTIONS

- Erweiterung des ARINC 653-Verfahrens

- N Zeitpartitionen $\tau_0 \dots \tau_N$

- Davon zwei gleichzeitig aktive Zeitpartitionen:

τ_0 : immer aktiv, τ_{Current} : ($0 < \text{Current} < N$), jeweils für vorgegebene Zeit

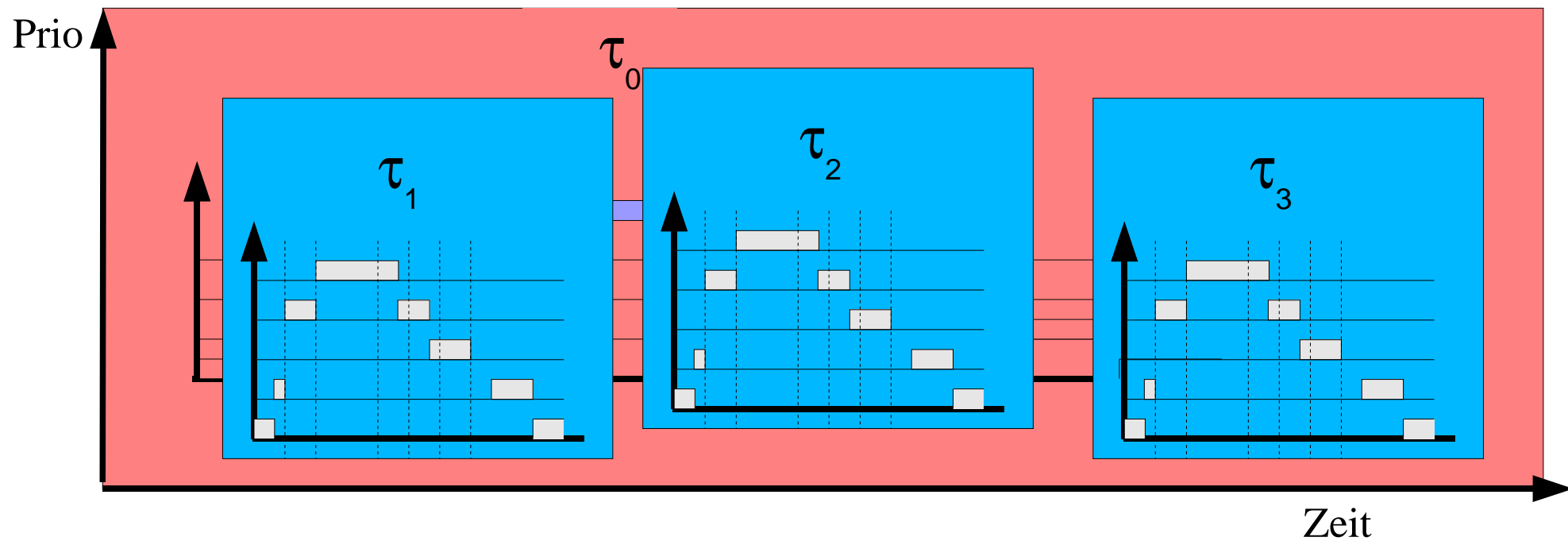


PikeOS-Verfahren



SYSGO
REAL-TIME SOLUTIONS

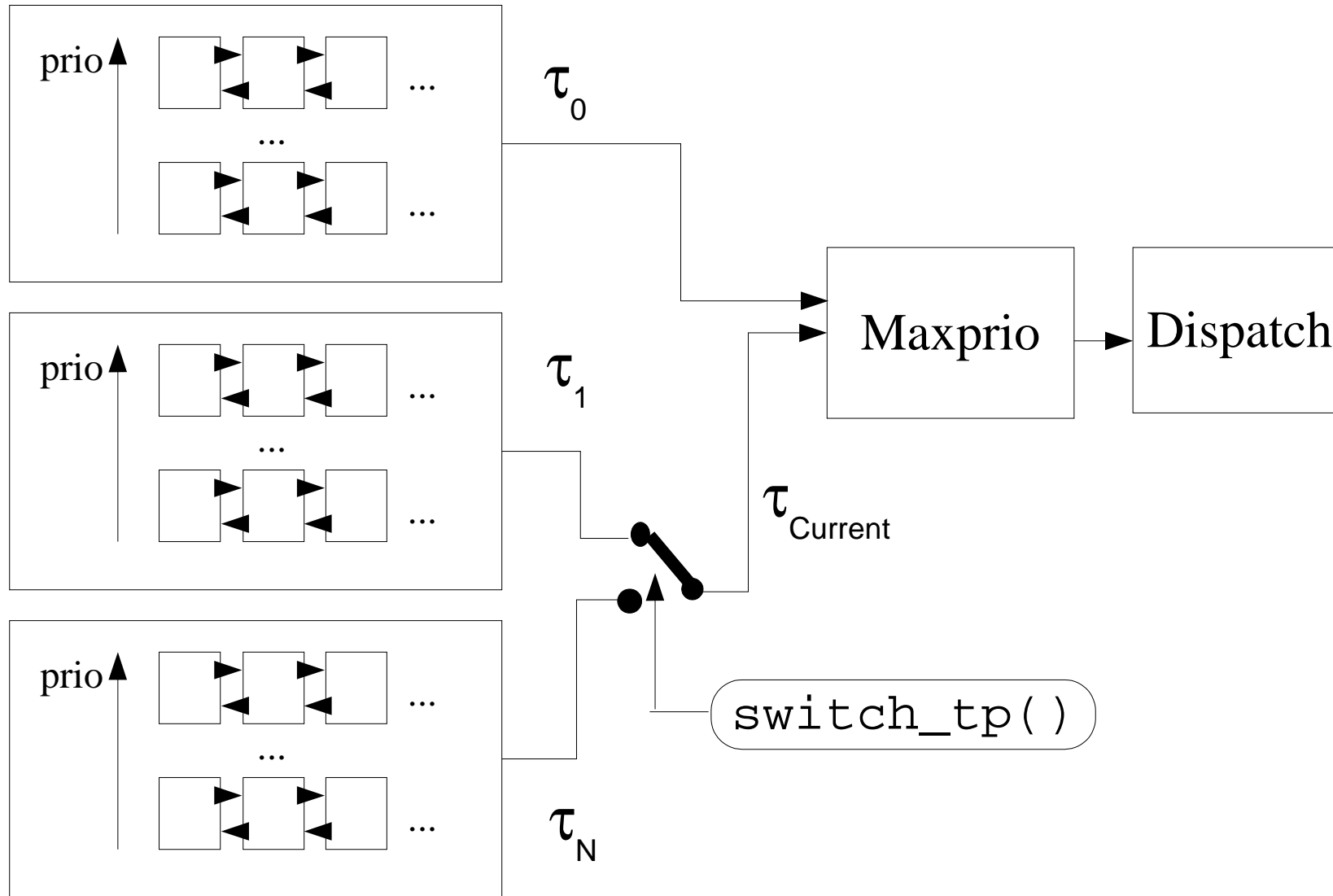
- Scheduler wählt einen Prozess aus $\tau_0 \cup \tau_{\text{Current}}$, entscheidet anhand der Priorität



Implementierung



SYSGO
REAL-TIME SOLUTIONS



Bedeutung der Zeitpartitionen



SYSGO
REAL-TIME SOLUTIONS

□ τ_{Current} : „Vordergrundpartition“

- **Fest zugesichertes Zeitkontingent**
- **Typisch: „Prioritätsband“**

□ τ_0 : „Hintergrundpartition“

- **niederprior: Nutzung übrig gebliebener Zeit für Nicht-Echtzeitprozesse (z.B. Linux)**
- **hochprior: Partitions-übergreifende Services**
 - **Device Treiber**
 - **Watchdog/Health Monitor**
 - **Partition Scheduler**



SYSGO
REAL-TIME SOLUTIONS

Implementierung

- ❑ **Npart x Nprio Ready queues**
- ❑ τ_{Current} **umschaltbar über Systemaufruf:**
- ❑ **`switch_tp_and_wait(TauNext, duration)`**
- ❑ **Zeitbasis: oneshot Timer (ggf. separate HW)**
- ❑ **Zeitpartitionsumschaltung im User-space**
 - Partition Scheduler: Hochpriorer Thread in τ_0
 - Einfaches Austauschen von Schedules
- ❑ **Geringer Overhead (1 syscall pro Partitionsumschaltung)**
 - Minimale Zeitschlitze $\sim 250\mu\text{s}$ bei vertretbarem Overhead(*)

(*) ca. 10% bei MPC 5200@400MHz



SYSGO
REAL-TIME SOLUTIONS

Vor/Nachteile

- ❑ gute Prozessorauslastung möglich
- ❑ keine Zwangskopplung von Vertrauen und Priorität
- ❑ deterministisch
- ❑ Problem: hohe worst case response wie bei zeitgetriebenen Verfahren, d.h. 1 Zyklus
- ❑ Mögliche Auswege:
 - Verlagern in τ_0 (möglich!), aber u.U. Vertrauensproblem
 - kurze Zeitschlitze (bis 250 μ s) -> „proportional share“
 - kritische Partitionen mehrfach im Zyklus

Zusammenfassung



SUSGO
REAL-TIME SOLUTIONS

- vielseitiges Verfahren**

Echtzeit- & Nicht-Echtzeitprozesse, ereignis- oder zeitgetrieben

- leicht verständlich, leicht Implementierbar**

- deterministisch**

- gute Prozessorauslastung**

- zum Patent angemeldet**