



Betriebssysteme | J.1



Franz J. Hauck | Institut für Verteilte Systeme, Univ. Ulm



Bild von Mike by Pexels

J | Verklemmungen Betriebssysteme



Franz J. Hauck | Institut für Verteilte Systeme, Univ. Ulm

Überblick

Überblick der Themenabschnitte

- **A** – Organisatorisches
- **B** – Zahlendarstellung und Rechnerarithmetik
- **C** – Aufbau eines Rechnersystems
- **D** – Einführung in Betriebssysteme
- **E** – Prozessverwaltung und Nebenläufigkeit
- **F** – Dateiverwaltung
- **G** – Speicherverwaltung
- **H** – Ein-, Ausgabe und Geräteverwaltung
- **I** – Virtualisierung
- **J** – Verklemmungen
- **K** – Rechteverwaltung



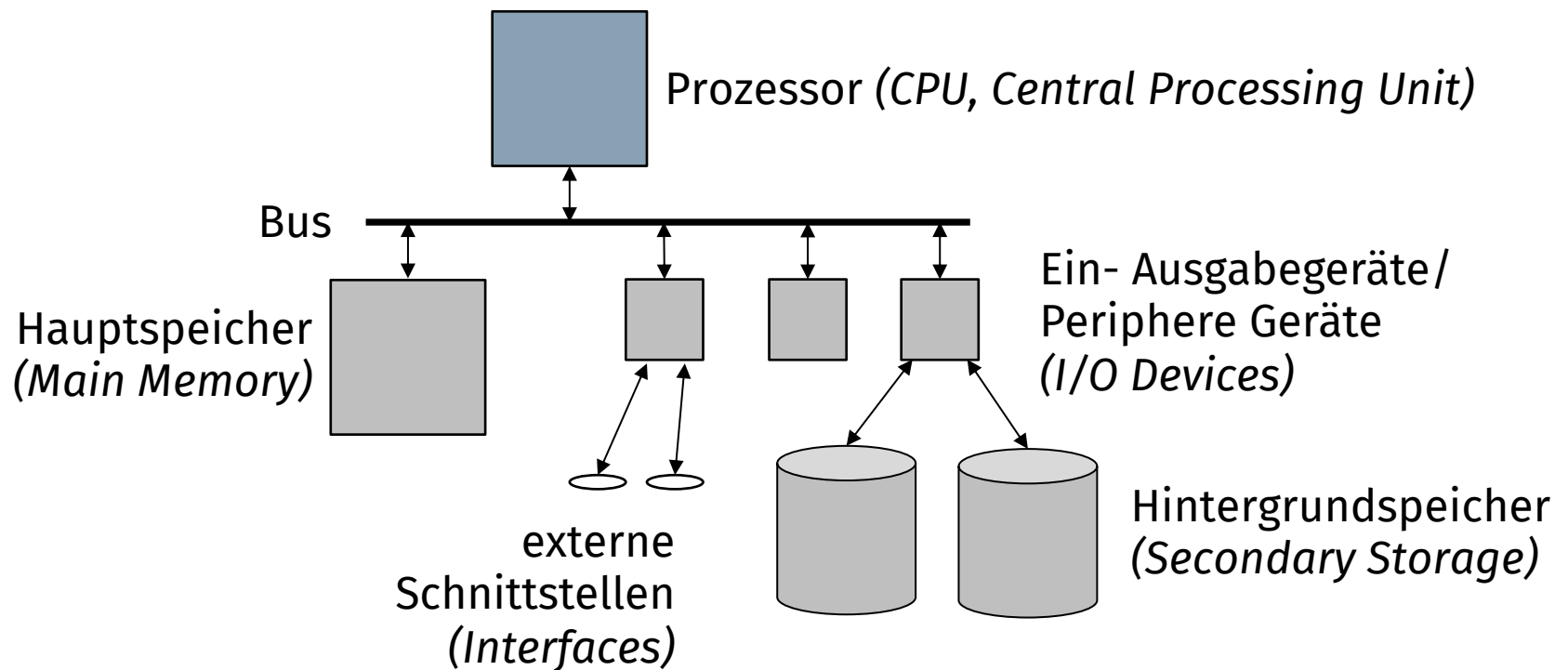
Inhaltsüberblick

Verklemmung

- Begriff
 - Motivation, Voraussetzung
 - Betriebsmittel, Betriebsmittelgraph
- Vermeidung
- Verhinderung
- Erkennung
- Erholung

Einordnung

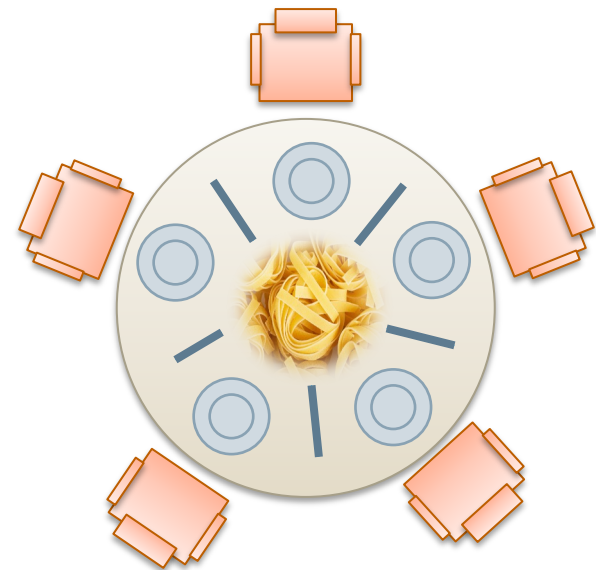
Betroffene physikalische Ressourcen



Motivation

Dinierende Philosophen

- Philosophen denken oder essen
 - *“The life of a philosopher consists of an alternation of thinking and eating.”* (Dijkstra, 1971)
- zum Essen zwei Gabeln nötig
 - jeweils eine Gabel zwischen zwei Philosophen



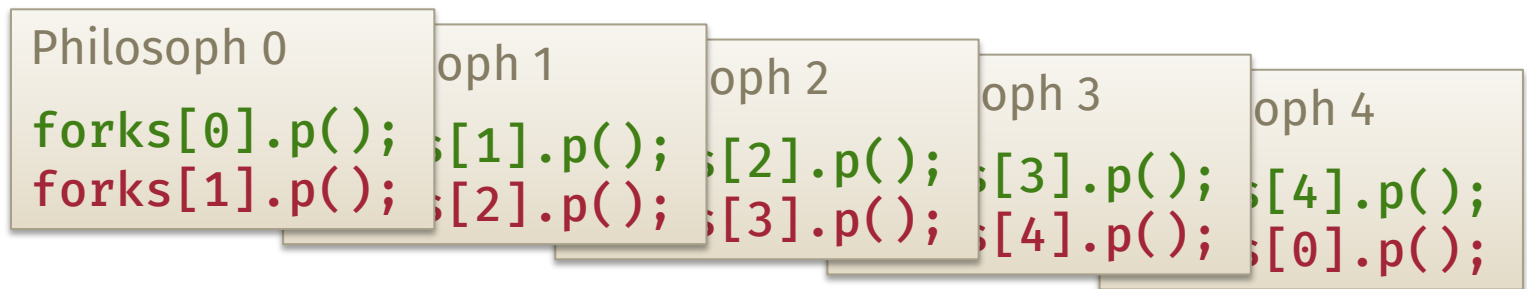
- Philosophen können verhungern, wenn sie sich „dumm“ anstellen!

Motivation (2)

Problem der Verklemmung (Deadlock)

■ Beispielsituation:

- alle Philosophen nehmen **zuerst linke** Gabel und versuchen **dann rechte** Gabel aufzunehmen
 - Implementierung durch Semaphore



- zweite Operation (**rot**) blockiert in allen Philosophen

→ System ist **verklemmt**: Philosophen warten alle auf ihre Nachbarn

Motivation (3)

Problem der Verklemmung (Deadlock)

■ Problemkreise:

- Vermeidung und Verhinderung von Verklemmungen
- Erkennung und Erholung von Verklemmungen

Problem der Verklemmung (Livelock)

- Prozesse laufen, machen aber keinen substantiellen Fortschritt
 - hier nicht betrachtet

Betriebsmittel / Ressourcen

Beispiele

- CPU, Drucker, Geräte (Platten, CD-ROM, Floppy, Audio, usw.)
- virtuelle Betriebsmittel der Anwendung oder des Betriebssystems, z.B. Gabeln der Philosophen

Unterscheidung von Typ und Instanz

- **Typ** definiert ein Betriebsmittel eindeutig
- **Instanz** ist eine Ausprägung des Typs
 - Anwendung benötigt **eine Instanz** eines Typs, **egal welche**
 - z.B. **CPU**: Anwendung benötigt eine von mehreren gleichen CPUs
 - z.B. **Drucker**: Anwendung benötigt einen von mehreren gleichen Druckern
 - z.B. **Gabeln**: jede Gabel ist eigener Typ

Betriebsnutzung

Nutzung erfolgt in drei Schritten

- **Anfordern** des Betriebsmittels (Belegung)
 - blockiert evtl. falls Betriebsmittel nur exklusiv benutzt werden kann
 - z.B. **Gabelaufnahme**: nur exklusiv
 - z.B. **Öffnen einer Datei**: exklusiv oder nicht-exklusiv
- **Nutzen** des Betriebsmittels
 - z.B. **Gabel**: Philosoph kann essen
 - z.B. **Datei**: Anwendung kann lesen und schreiben
- **Freigeben** des Betriebsmittels
 - z.B. **Gabel**: Philosoph legt Gabel wieder zwischen die Teller
 - z.B. **Datei**: Datei wird geschlossen

Voraussetzungen

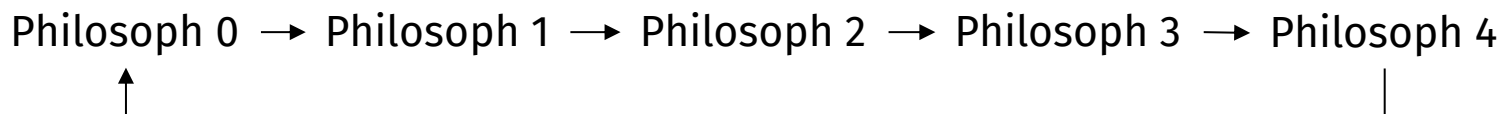
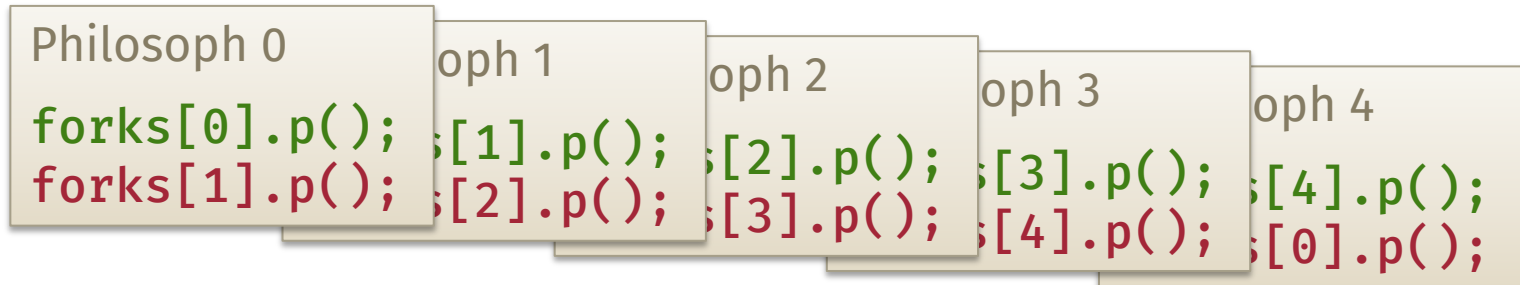
Voraussetzungen für Verklemmung (Deadlock)

- vier **notwendige** Bedingungen
 - **exklusive Belegung**
 - mindestens ein Betriebsmitteltyp exklusiv belegbar
 - **Nachforderungen** von Betriebsmittel möglich
 - ein Prozess hält bereits Betriebsmittel und fordert weiteres an
 - **kein Entzug** von Betriebsmitteln möglich
 - Betriebsmittel können nicht zurück gefordert werden bis der Prozess sie wieder freigibt
 - **zirkuläres Warten**
 - Ring von Prozessen, in dem jeder auf ein Betriebsmittel wartet, das der Nachfolger im Ring besitzt

Voraussetzungen (2)

Beispiel der Philosophen

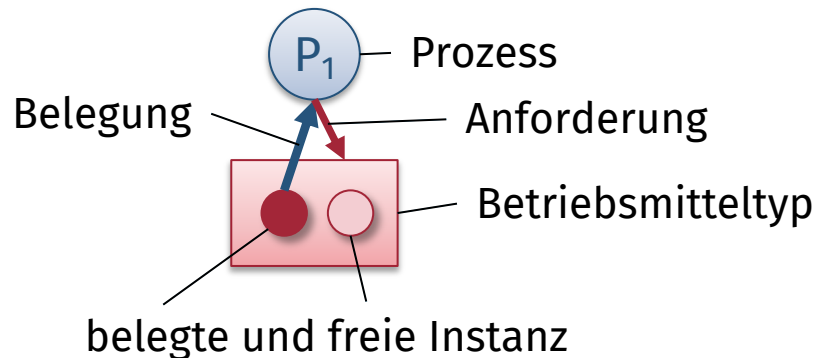
- exklusive Belegung: **ja**
- Nachforderungen von Betriebsmittel möglich: **ja**
- Entzug von Betriebsmitteln: **nicht vorgesehen**
- zirkuläres Warten: **ja**



Betriebsmittelgraph

Veranschaulichung von Belegungen und Anforderungen

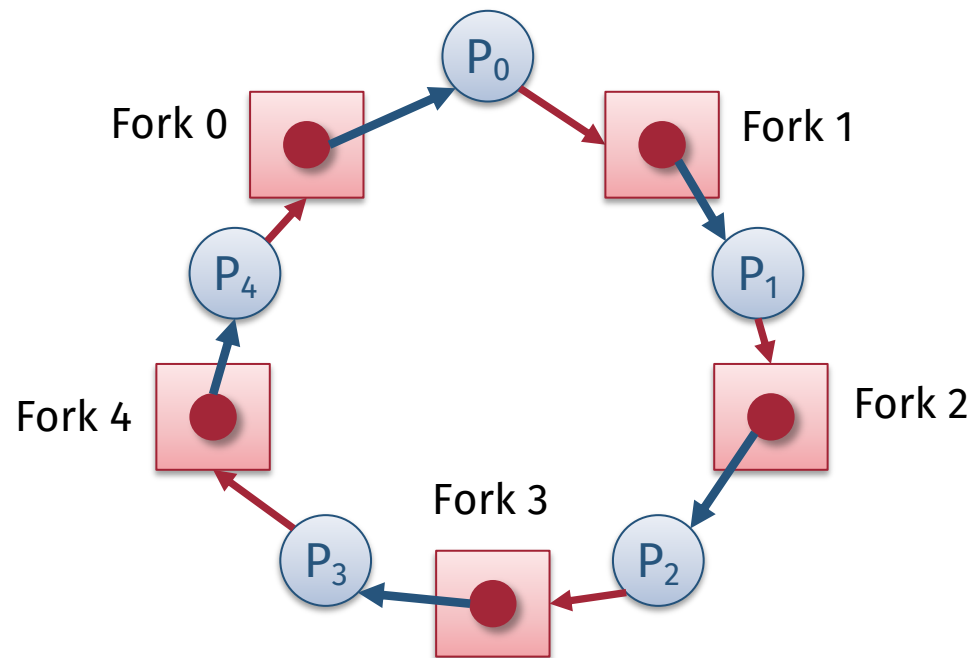
- nur exklusive Belegungen betrachtet



- ableitbare Regeln
 - kein Zyklus im Graph → keine Verklemmung
 - Zyklus im Graph und nur eine Instanz pro Typ → Verklemmung
 - Zyklus ist notwendige Bedingung

Betriebsmittelgraph (2)

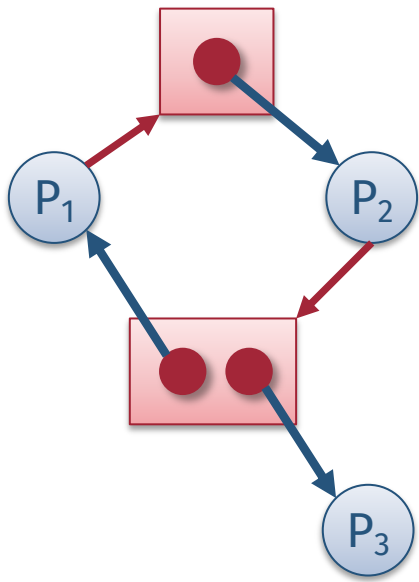
Beispiel fünf Philosophen



- Zyklus und eine Instanz pro Typ → **Verklemmung**

Betriebsmittelgraph (3)

Beispiel mit Zyklus ohne Verklemmung



- sobald P_3 Betriebsmittel zurückgibt wird Zyklus aufgelöst



Betriebssysteme | J.2



Franz J. Hauck | Institut für Verteilte Systeme, Univ. Ulm

Inhaltsüberblick

Verklemmung

- Begriff
 - Motivation, Voraussetzung
 - Betriebsmittel, Betriebsmittelgraph
- Vermeidung
- Verhinderung
- Erkennung
- Erholung

Vermeidung von Verklemmungen

Ansatz zur Vermeidung

- Vermeidung der notwendigen Bedingungen (*Avoidance*)
- **exklusive** Betriebsmittel
 - in der Regel **nicht vermeidbar**
- **Nachforderung** von Betriebsmitteln
 - Ansatz: alle Betriebsmittel **auf einmal anfordern**
 - z.B. **beide Gabeln** in atomarer Belegung (beide oder keine)
 - Nachteile
 - **ungenutzte** aber belegte Betriebsmittel vorhanden
 - **Aushungerung** möglich: Betriebsmittel nie verfügbar

Vermeidung von Verklemmungen (2)

Ansatz zur Vermeidung (fortges.)

■ kein Entzug von Betriebsmitteln

● Ansatz: Entzug erlauben

- mit Belegung werden gehaltene Betriebsmittel freigegeben und gleich wieder zusammen mit neuen angefordert
- während Warten auf neue Betriebsmittel werden bisher gehaltene frei für andere
- möglich für CPU und Speicher (durch Auslagerung)
- unmöglich für Drucker und ähnliche

■ Zyklus

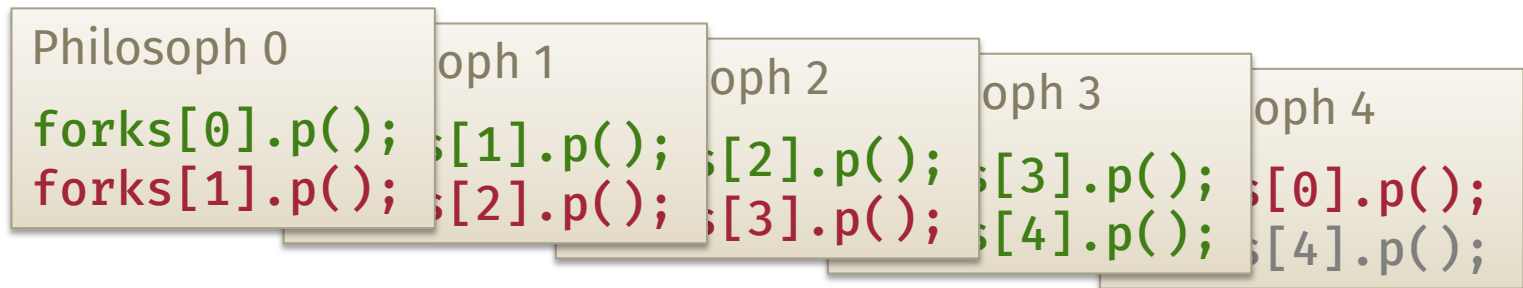
- Ansatz: totale Ordnung in der Betriebsmittel ausschließlich angefordert werden dürfen

Vermeidung von Verklemmungen (3)

Zyklusvermeidung

■ Beispiel fünf Philosophen

- Anforderung der Gabeln nur in aufsteigender Gabelnummern



- kein Zyklus mehr möglich
- keine Verklemmung mehr möglich

Vermeidung von Verklemmungen (4)

Zyklusvermeidung (fortges.)

■ Realisierung

- **Fehlermeldung** bei Verletzung der Ordnung
 - Rückgabe der Betriebsmittel
 - Anforderung in richtiger Reihenfolge
- **nicht immer realisierbar**
 - z.B. weil nächste Anforderung nicht bekannt und Rückgabe mitten in der Operation nicht möglich

Verhinderung

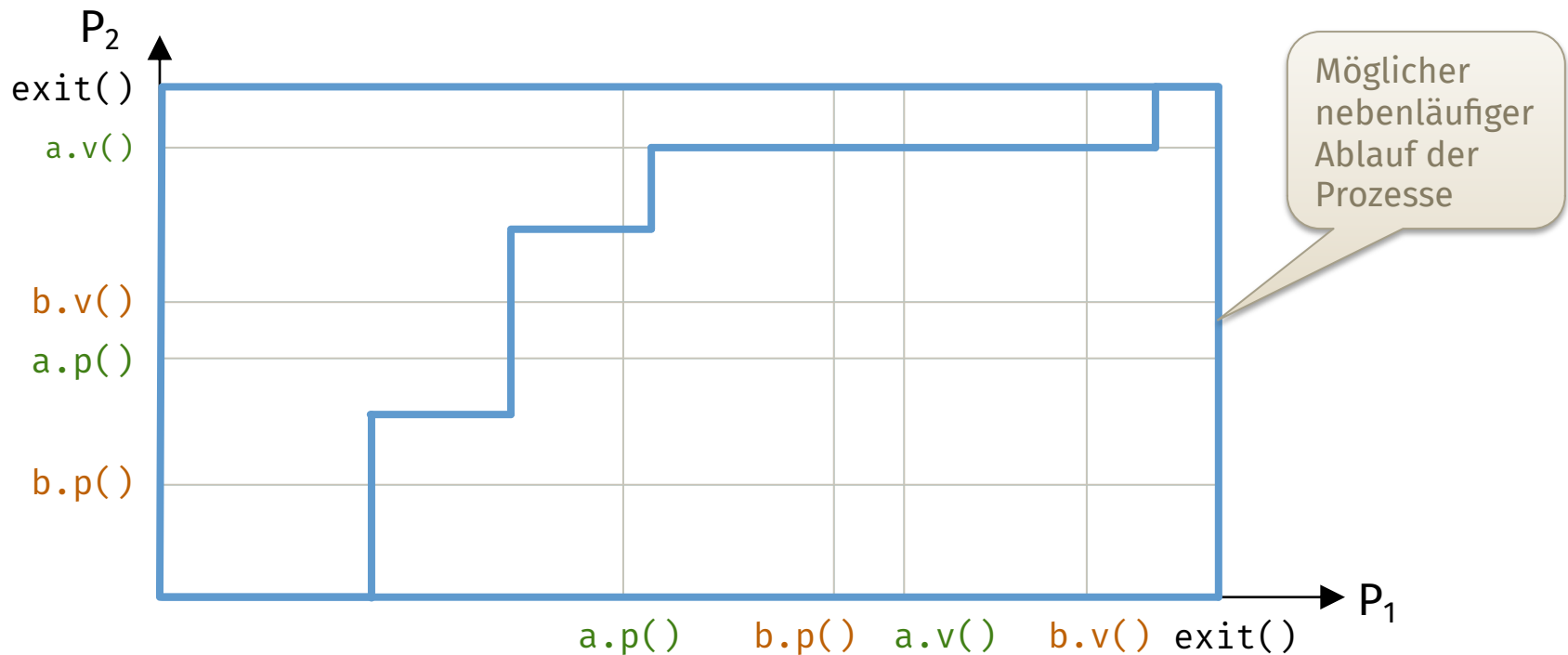
Ansatz zur Verhinderung

- Systemzustand, in dem Verklemmung entstehen könnte, wird vermieden (*Prevention*)
 - **Voraussetzung:** bekannt, welche Betriebsmittel angefordert und freigegeben werden und in welcher Reihenfolge
- **sicherer** Zustand
 - System gerät nicht in Verklemmung
- **unsicherer** Zustand
 - System wird sich verklemmen läuft aber noch
- **unmöglicher** Zustand
 - z.B. Semantik von Semaphoren verletzt
 - zwei Prozesse haben P-Operation durchgeführt

Verhinderung

Beispiel

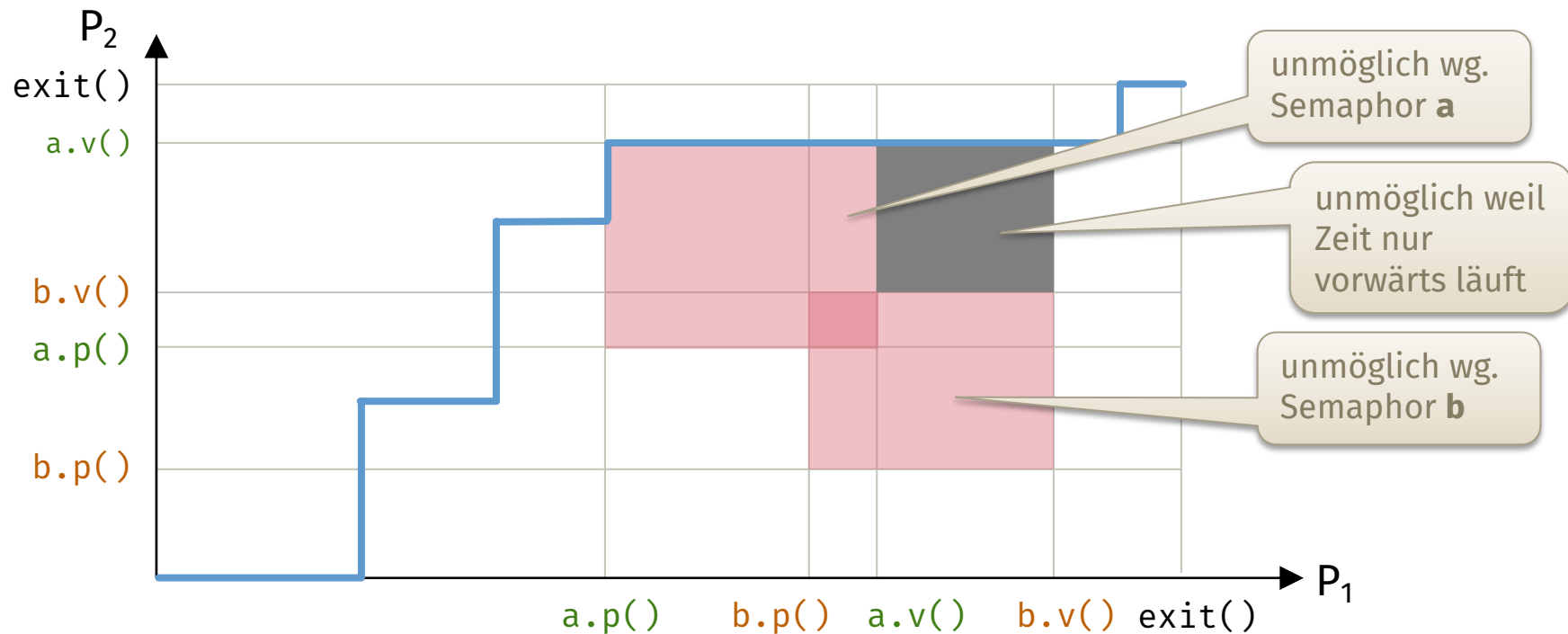
- zwei Prozesse mit Belegung zweier Semaphore A und B



Verhinderung (2)

Beispiel

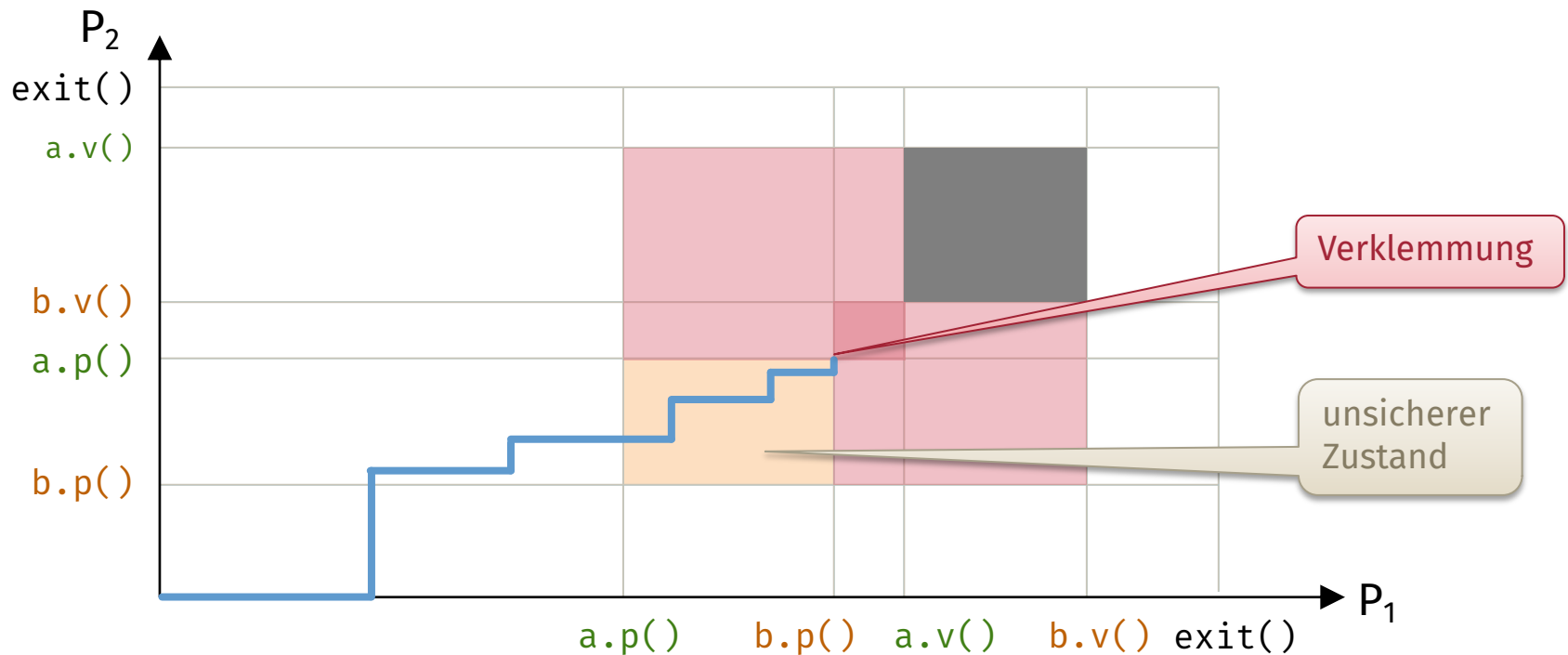
- zwei Prozesse mit Belegung zweier Semaphore A und B



Verhinderung (3)

Beispiel

- zwei Prozesse mit Belegung zweier Semaphore A und B



Verhinderung (4)

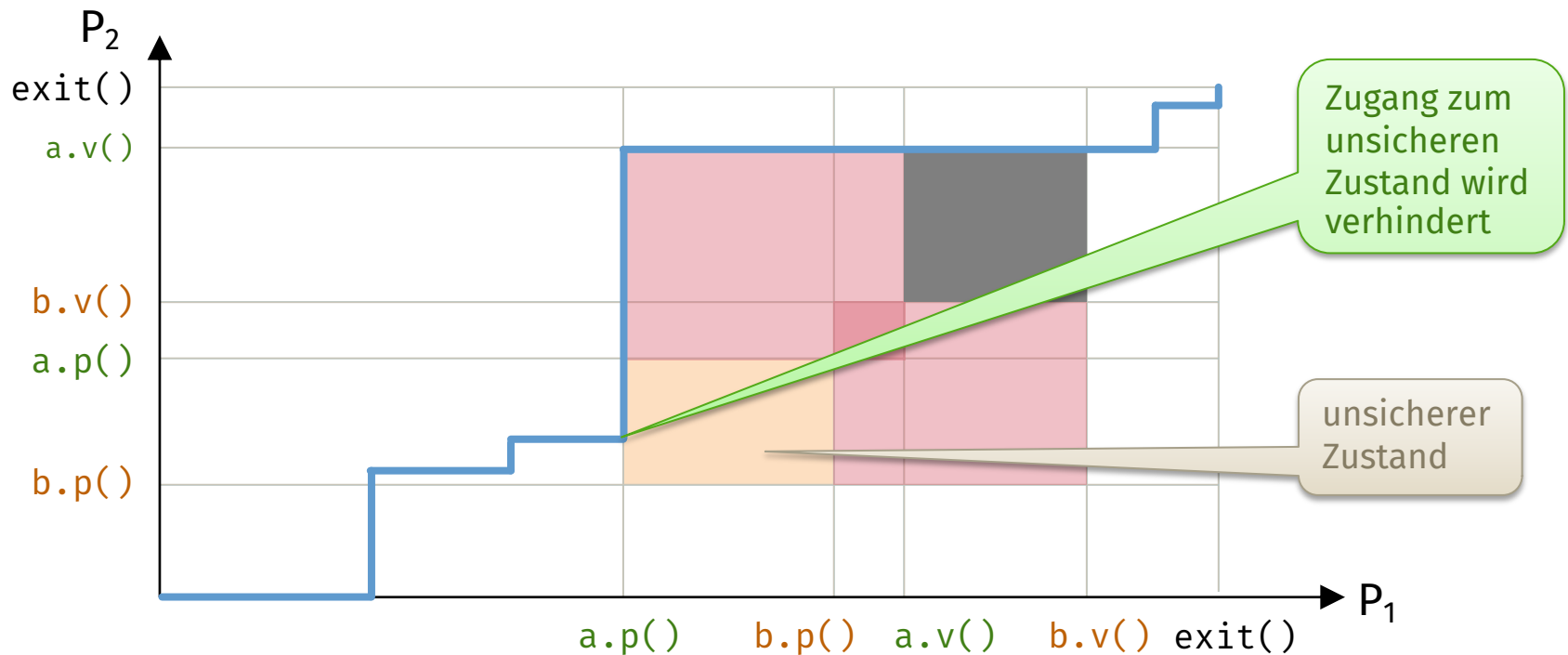
Ansatz zur Verhinderung

- Erkennung der unsicheren Zustände
 - System verhindert Eintritt in unsicheren Zustand
 - z.B. mit Hilfe des Bankers-Algorithmus

Verhinderung (3)

Beispiel

■ Ablauf mit Verklemmungsverhinderung



Erkennung

Erkennung von Verklemmungen

- Zykluserkennung im Betriebsmittelgraph
 - kann auf Wartegraph reduziert werden
 - Wartegraph zeigt, welcher Prozess auf welchen anderen wartet
 - nur für Betriebsmittel mit einer Instanz pro Typ möglich
- graphische Reduktionsmethode
 - anwendbar auf Betriebsmittel mit mehreren Instanzen pro Typ
 - hier nicht näher behandelt

Erkennung (2)

Überlegungen zum Einsatz

- Einsatz sehr rechenzeitaufwändig
 - Verklemmungen eher selten
 - zu häufiger Einsatz: **Verschwendung** von Ressourcen
 - zu seltener Einsatz: **Durchsatz sinkt, kein Fortschritt**
- mögliche Vorgehensweisen
 - Einsatz, wenn Belegung **blockiert**
 - **periodisch** mit niedriger Rate
 - Einsatz, wenn Durchsatz oder Auslastung **sinkt**

Erholung

Erholung von Verklemmungen

- Verklemmung erkannt: Was tun?
 - manuelle Beseitigung
 - System erholt sich selbständig
- Abbrechen von Prozessen
 - terminierte Prozesse geben ihre Betriebsmittel wieder frei
 - alle verklemmten Prozesse abbrechen: **großer Schaden**
 - einen Prozess abbrechen und Erkennung wiederholen
- Probleme:
 - **Verlust** von berechneter Information
 - **Rücksetzbarkeit** von bisherigen Effekten des Prozesses
 - Konsistenz von Daten

Einsatzgebiete

Einsatz von Antiverklebungsmaßnahmen

- in **Betriebssystemen**
 - interne Vermeidung durch Ordnung auf Betriebsmitteln
- in **Anwendungsprozessen**
 - bisher keine Unterstützung durch das Betriebssystem
- in **Datenbanksystemen**
 - typischerweise ausgefeilte Verklebungserkennung und -auflösung
 - Rücksetzbarkeit von Datenbanktransaktionen unterstützt Erholung

Inhaltsüberblick

Verklemmung

- Begriff
 - Motivation, Voraussetzung
 - Betriebsmittel, Betriebsmittelgraph
- Vermeidung
- Verhinderung
- Erkennung
- Erholung