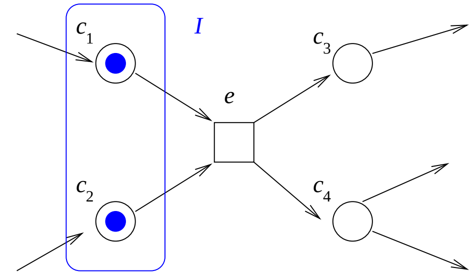
Petri Netze (PN)

- Neben Prozess-Algebra geläufigste Modellierung *nebenläufiger* Systeme
  - seit 60er Jahren untersucht, in Form von **Activity Diagrams** in UML
  - auch wieder: **asynchron** kommunizierende Prozesse (Protokolle/Software)
- Modellierungs- und Verifikationstools existieren
- **Theorie:** Umfangreiche theoretische Untersuchungen
  - Endlichkeit, Deadlock, ...
- Aus der Praxis kommende Erweiterungen
  - Daten, Färbung, Hierarchie, quantitative Aspekte, ...

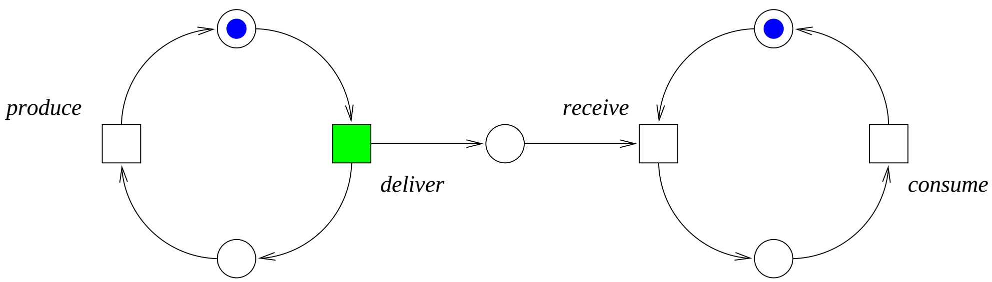
Condition-Event-Netze (CEN)

Definition Ein CEN $N = (C, I, E, G)$ besteht aus Bedingungen C, Anfangsmarkierung $I \subseteq C$, Events E und Abhängigkeiten $G \subseteq (C \times E) \,\dot\cup\, (E \times C)$



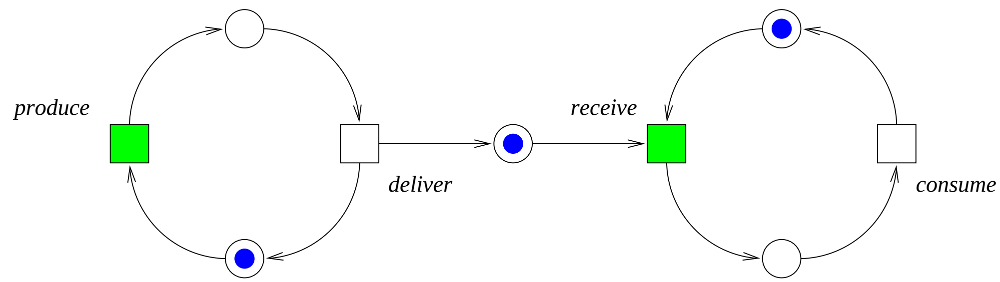
- oftmals $\rightarrow$ statt G
- kann interpretiert werden als *bipartiter* Graph oder ...
- ... Hypergraph mit Multi-Quellen- bzw. Multi-Ziel-Kanten E

Producer Consumer CEN: Anfangsmarkierung



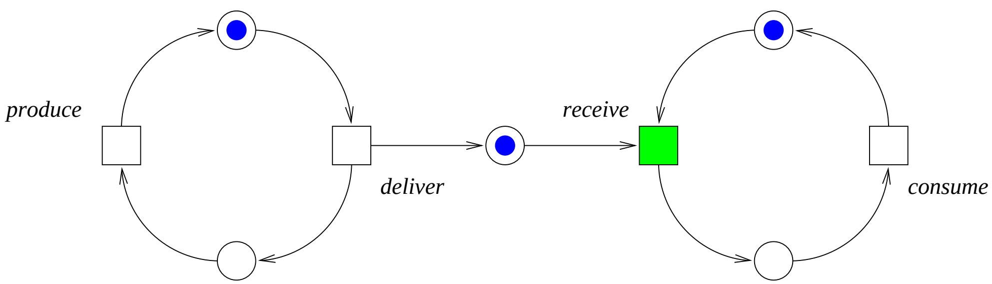
nur ein Event/Transition kann feuern

Producer Consumer CEN: Ausgeliefert



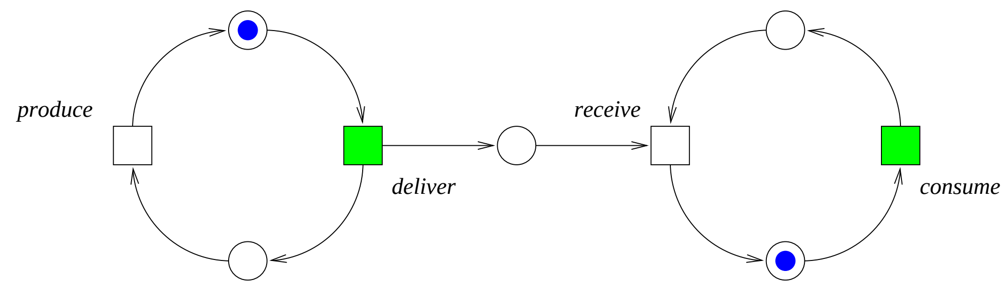
zwei Events/Transitionen können feuern

Producer Consumer CEN: Erneut Produziert



Ziel-Condition von *deliver* besetzt

Producer Consumer CEN: Konsumiert



wiederum Auswahl aus zwei möglichen Events

Semantik eines CEN als LTS

Definition Geg. CEN $N = (C, I, E, G)$. Das LTS $L = (S, \{I\}, \Sigma, T)$ zu N ist definiert durch

$$S = \mathbb{P}(C) \qquad \Sigma = E$$

$$\begin{array}{llll} T(C_1, e, C_2) \ \text{gdw.} & G^{-1}(e) \subseteq C_1 & \text{Vorbedingungen erfüllt} & (1) \\ & G(e) \cap C_1 = \emptyset & \text{Nachbedingungen unerfüllt} & (2) \\ & C_2 = (C_1 \backslash G^{-1}(e)) \cup G(e) & \text{Zustandsänderung} & \end{array}$$

$G(e) \;=\;$ Nachbedingungen von Event e (oder auch $e \rightarrow$)

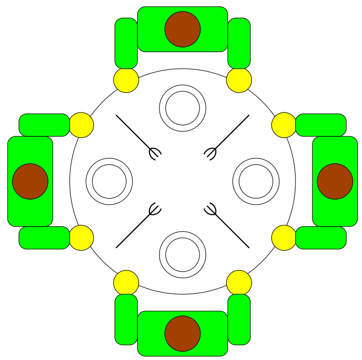
$G^{-1}(e) \;=\;$ Vorbedingungen von Event e (oder auch $\rightarrow e$)

Bemerkungen zur Semantik von CEN

- die Zustände $M \in \mathbb{P}(C)$ des LTS werden auch als Markierungen des CEN bezeichnet
- ein Event e ist **ausführbar** in M wenn $M \xrightarrow{e} \neq \emptyset$
- eine Markierung $M \in \mathbb{P}(C)$ ist ein **Deadlock**
  - gdw. M eine *Sackgasse* im LTS darstellt
  - gdw. kein Event in M ausführbar ist
  - gdw. alle Events *disabled* sind
  - gdw. $\forall e \in E[M \xrightarrow{e} = \emptyset]$
- ein CEN hat einen Deadlock gdw. ein Deadlock erreichbar ist

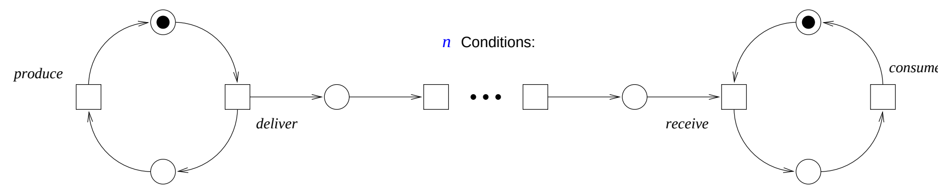
Beispiel Dining Philosophers

n Philosophen, n Gabeln, n Teller

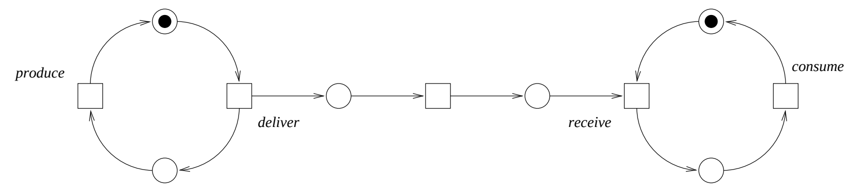


Philosophen Denken und Essen abwechselnd
Essen geht nur mit zwei Gabeln,
welche nicht gleichzeitig vom Tisch genommen werden können

Kapazitäten



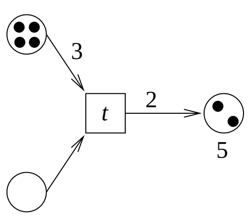
Puffer der Kapazität n



Puffer der Kapazität 2

Place Transition Net (PTN)

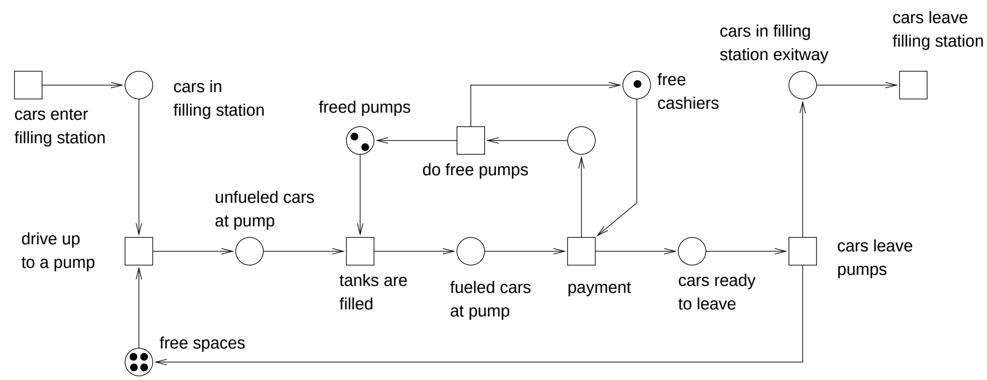
Definition Ein PTN $N = (P, I, T, G, C)$ besteht aus Stellen P, Anfangsmarkierung $I : P \to \mathbb{N}$, Transitionen T, Verbindungsgraph $G \subseteq (P \times T) \dot\cup (T \times P)$, und Kapazität $C : P \dot\cup G \to \mathbb{N}_\infty$.



- Kapazität der Kanten G ist endlich und sogar 1 falls nicht explizit angegeben
- Kapazität der Stellen P kann auch ∞ sein und ist es auch falls nicht explizit angegeben
- CEN lassen sich als PTN interpretieren mit konstanter Kapazität $C \equiv 1$

Filling Station

nach [W. Reisig, *A Primer in Petri Net Design, 1992*]



Geg. PTN $N = (P, I, T, G, C)$.

Definition Transition $t \in T$ kann in Zustand/Markierung $M : P \to \mathbb{N}$ feuern gdw.

$C((p,t)) \leq M(p)$ für alle $p \in G^{-1}(t)$ und

$C((t,q)) + M(q) \leq C(q)$ für alle $q \in G(t)$.

Definition Transition $t \in T$ führt $M_1 : P \to \mathbb{N}$ nach $M_2 : P \to \mathbb{N}$ über gdw.

t kann in M_1 feuern und $M_2 = M_1 - M_- + M_+$ mit

$$M_-(p) = \begin{cases} C((p,t)) & p \in G^{-1}(t) \\ 0 & \text{sonst} \end{cases} \qquad M_+(p) = \begin{cases} C((t,p)) & p \in G(t) \\ 0 & \text{sonst} \end{cases}$$

Definition Das LTS $L = (S, \{I\}, \Sigma, T_L)$ zu N ist definiert durch

$S = \mathbb{N}^P \qquad \Sigma = T \qquad$ und $\qquad T_L(M_1, t, M_2)$ gdw. t führt M_1 nach M_2 über