Petri-Netze / Eine Einführung

Manuel Hertlein

Seminar Systementwurf
Lehrstuhl Theorie der Programmierung

Carl Adam Petri

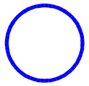
- am 12. Juli 1926 in Leipzig geboren
- Studium der Mathematik
- 1962 Promotion zum Doktor der Naturwissenschaft
- Titel der Dissertationsschrift: "Kommunikation mit Automaten"
- über die Modellierung, Analyse und Simulation von verteilten, dynamischen Systemen mit nebenläufigen und nicht-deterministischen Vorgängen
- Entwicklung von Regeln aus der Automatentheorie für ein neues Systemmodell - „Petri-Netz“

Was sind Petri-Netze?

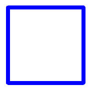
- stellen einen Formalismus zur Beschreibung nebenläufiger, kommunizierender Prozesse dar
- verallgemeinern wegen der Fähigkeit, nebenläufige Ereignisse darzustellen, die Automatentheorie

- Erfolg durch ihre einfache und anschauliche Natur
- zugleich aber auch formal und ausdrucksstark
- mit starken Bezug zur Praxis (Vielzahl von Systemen beschreibbar)

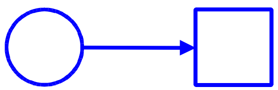
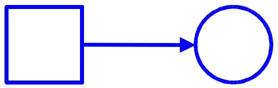
Aufbau eines Petri-Netzes: Plätze

- passive Komponenten
- repräsentieren Objekte / Substantive
- können Dinge
  - lagern *(Bücherbestand)*
  - speichern (*Kartei entliehener Bücher*)
  - sichtbar machen (*MA verfügbar*)
  - Zustände repräsentieren (*MA beschäftigt*)
- grafische Darstellung: 

Aufbau eines Petri-Netzes: Transitionen

- aktive Komponenten
- repräsentieren Verben / Methoden
- können Dinge
  - erzeugen (*Leihschein ausfüllen*)
  - transportieren (*Buch ausgeben*)
  - verändern (*Karteikarte ablegen*)
  - vernichten (*Leihstatus löschen*)
- grafische Darstellung: 

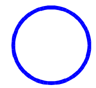
Aufbau eines Petri-Netzes: Kanten

- gedankliche (abstrakte) Beziehungen zwischen Komponenten, d.h.
  - Logische Zusammenhänge
  - Zugriffsrechte
  - Kausale Ordnung
- <u>keine</u> Komponente des Systems
- grafische Darstellung: $\longrightarrow$
- Beziehungen:
  -  : (wird) konsumiert
  -  : (wird) produziert

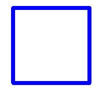
Definition Petri-Netz

- N = (P,T,F)

  - P : nicht leere Menge von Plätzen
    P = {p1,p2,...,p|P|}



  - T : nicht leere Menge von Transitionen (Übergängen)
    T = {t1,t2,...,t|T|}

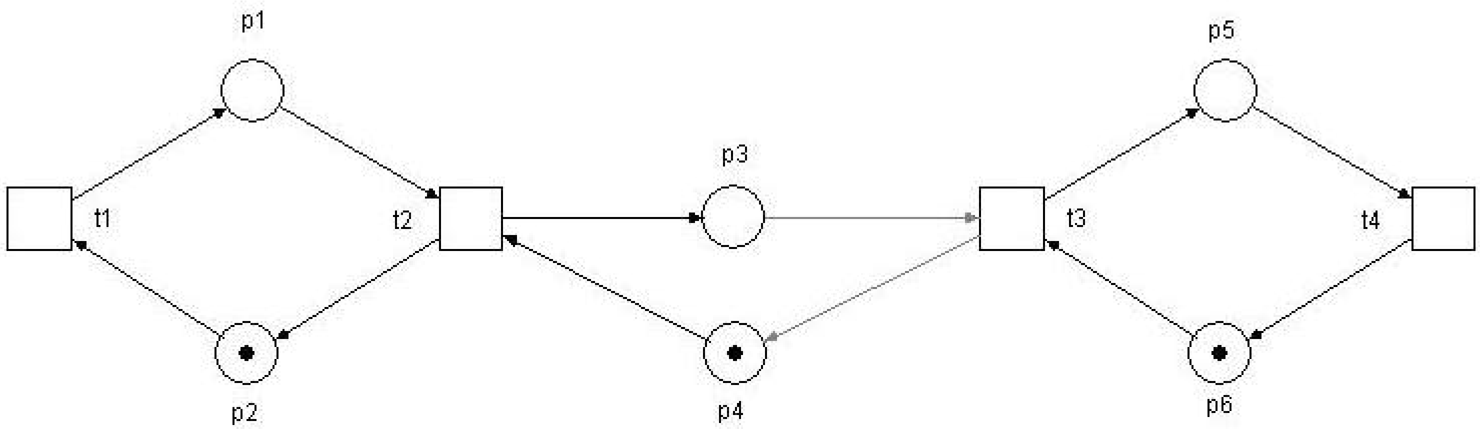


  - F : nicht leere Menge der Kanten (Flussrelation)
    $F \subseteq (P \times T) \cup (T \times P)$

  - Die Mengen der Plätze P und Transitionen T sind disjunkt.

  (bipartiter und gerichteter Graph)

Beispiel: Definition Petri-Netz



- N = (P,T,F)

  wobei

  - P = {p1, p2, p3, p4, p5, p6}
  - T = {t1, t2, t3, t4}
  - F = {{p1, t2}, {p2, t1}, {p3, t3}, {p4, t2}, {p5, t4}, {p6, t3}, {t1, p1}, {t2, p2}, {t2, p3}, {t3, p4}, {t3, p5}, {t4, p6}}

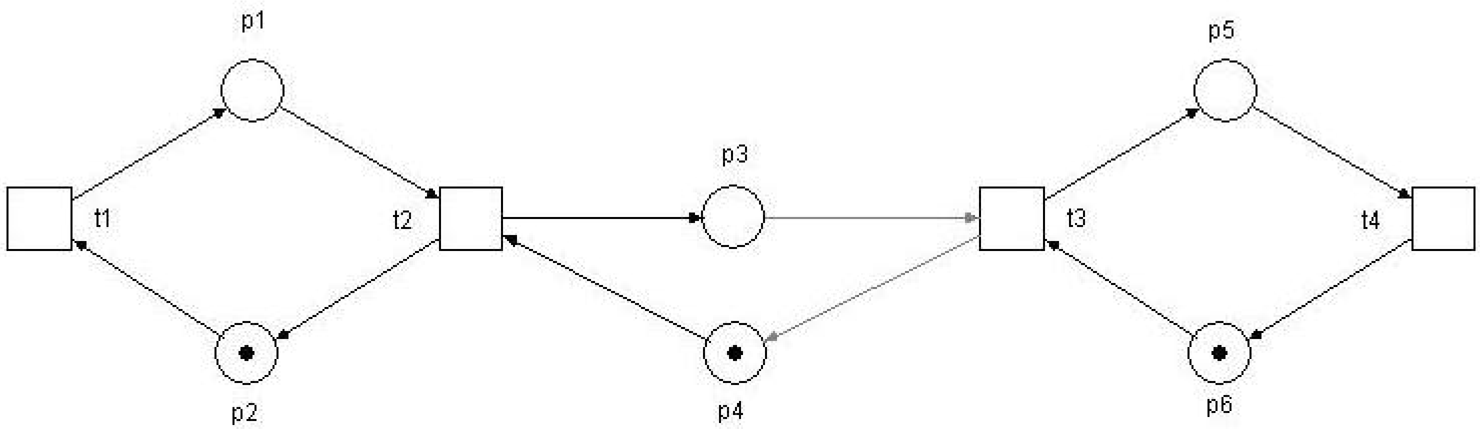
Eigenschaften von Petri-Netzen

- $N = P \cup T$ und aFb steht für $(a, b) \in F$

- F^{-1}: inverse Relation
  - $aF^{-1}b$ *genau dann, wenn* bFa
- F^{+}: transitive Hülle
  - $aF^{+}b$ *genau dann, wenn* $aFc_1Fc_2...c_nFb$ für $c_1,..,c_n \in N$
- F^{*}: reflexiv-transitive Hülle
  - $aF^{*}b$ *genau dann, wenn* $aF^{+}b$ oder $a=b$

- Folgende (Teil-)Mengen sind für $a \in N$ definiert:
  - Vorbereich $\bullet a = \{ b \in N \mid (b,a) \in F \}$
  - Nachbereich $a\bullet = \{ b \in N \mid (a,b) \in F \}$

Beispiel: Eigenschaften von Petri-Netzen



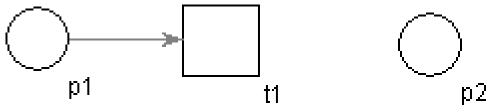
- Für alle a, b ∈ N gilt: aF*b
- Für alle p ∈ P gilt: •p und p• haben genau ein Element.
  - Beispiel: •p1 = {t1} und p1• = {t2}
- Für alle t ∈ T gilt: |•t| = |t•|
  - Beispiel: •t2 = {p1, p4} und t2• = {p2, p3}

Isomorphismus von Petri-Netzen

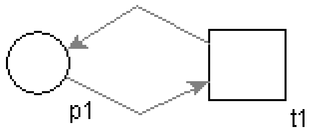
- Zwei Petri-Netze N und N' sind isomorph *genau dann, wenn* es eine Bijektion $\beta : N \rightarrow N'$ zwischen ihren Elementen gibt, so das gilt:
  - $\beta(P_N) = P_{N'}$
  - $\beta(T_N) = T_{N'}$
  - xF_Ny *genau dann, wenn* $\beta(x)F_{N'}\beta(y)$

Spezielle Strukturen (1)

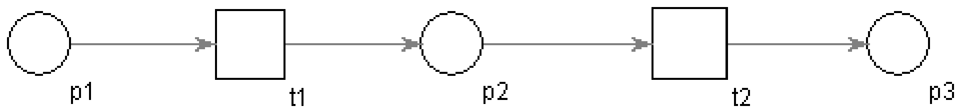
- $x \in N$ heisst <u>isoliert</u> *gdw.* $\bullet x \cup x\bullet = \emptyset$



- $x, y \in N$ bilden einen <u>Loop</u> *gdw.* xF_Ny und yF_Nx

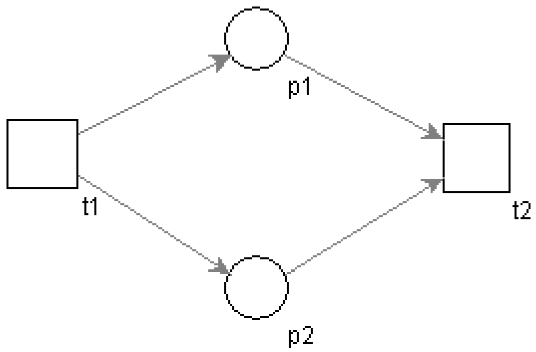


- x und y sind <u>*losgelöst voneinander*</u> *gdw.*
  $(\bullet x \cup \{x\} \cup x\bullet) \cap (\bullet y \cup \{y\} \cup y\bullet) = \emptyset$
  - p1, p3 losgelöst voneinander
  - p1, p2 <u>nicht</u> losgelöst voneinander

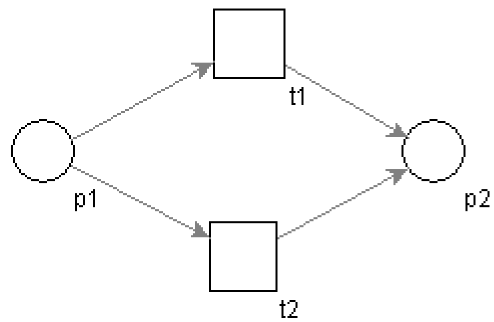


Spezielle Strukturen (2)

- N heisst <u>A-einfach</u> ($A \subseteq N$) *gdw.* für alle $x, y \in A$ gilt:
  $\bullet x = \bullet y$ und $x \bullet = y \bullet \rightarrow x = y$



T_N-einfach
<u>nicht</u> P_N-einfach



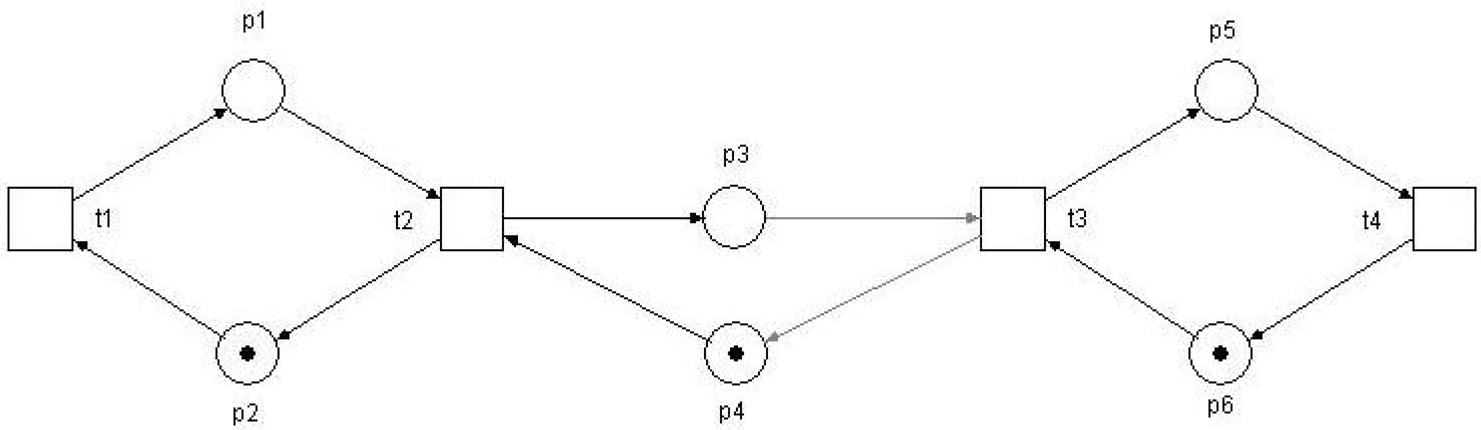
P_N-einfach
<u>nicht</u> T_N-einfach

  - in einem A-einfachen Netz ist jeder Platz/jede Transition eindeutig durch ihren Vor- und Nachbereich bestimmt

- N ist <u>einfach</u> *gdw.* N N-einfach ist

Spezielle Strukturen (3)

- N ist <u>zusammenhängend</u> *gdw.* für alle x, y $\in$ N gilt: $x(F \cup F^{-1})^*y$

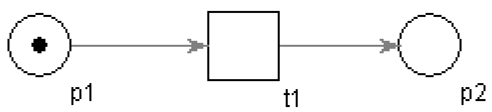


- N ist <u>stark zusammenhängend</u> *gdw.* für alle x, y $\in$ N gilt: xF^*y

Labeling

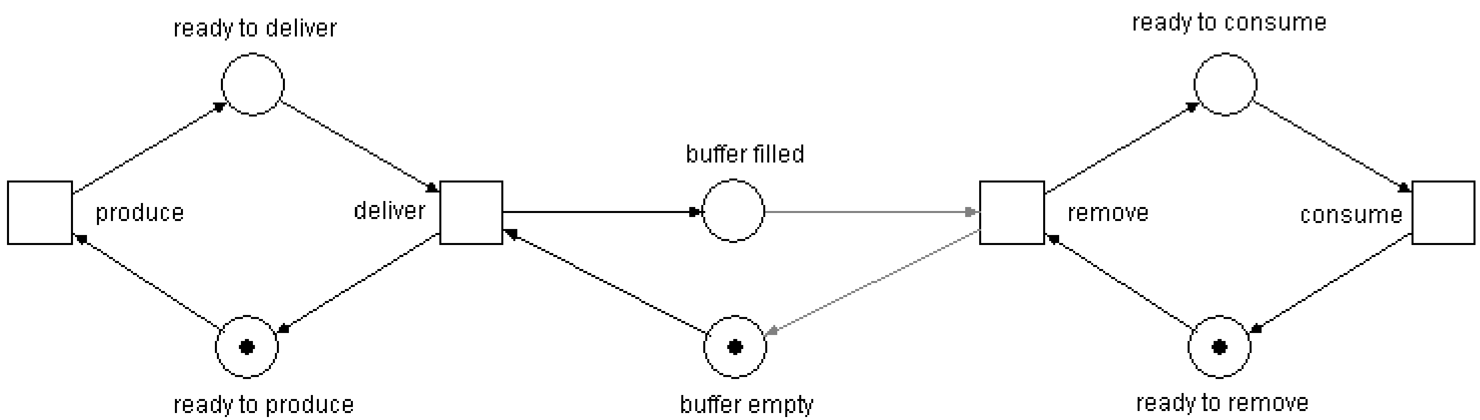
- Labeling von Plätzen $l_1: P_N \rightarrow A$
- Labeling von Transitionen $l_2: T_N \rightarrow A$
- Labeling von Elementen $l_3: P_N \cup T_N \rightarrow A$
- Labeling von Kanten $l_4: F_N \rightarrow A$

- Beispiel:
  - Die Punkte in den Kreisen repräsentieren ein Labeling von Plätzen:
    $l: P_N \rightarrow \{0, 1\}$
    mit $l(p) = 1$ gdw. ein Platz durch einen Kreis mit einem Punkt repräsentiert wird



Dynamik in Petri-Netzen (1)

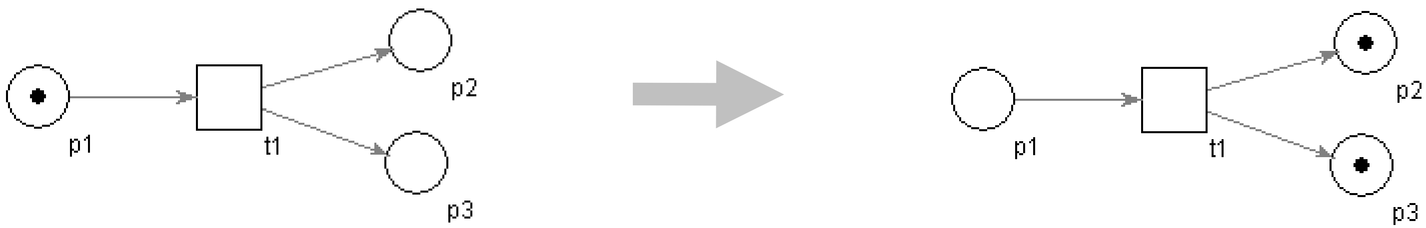
- Plätze können verschiedene Zustände annehmen
- können sich in Abhängigkeit von Zuständen benachbarter Plätze und Transitione ändern
- grafisch werden Zustände durch Marken dargestellt



Dynamik in Petri-Netzen (2)

- Transition heisst *aktiviert* (kann schalten), wenn
  - alle *Vorbedingungen erfüllt* sind, d.h. auf allen Plätzen im Vorbereich Marken liegen
  - alle *Nachbedingungen unerfüllt* sind, d.h. alle Plätze im Nachbereich leer sind

- Wenn eine Transition *schaltet*, werden in einem Schritt
  - Alle seine *Vorbedingungen unerfüllt*, d.h. von allen Plätzen im Vorbereich werden die Marken entfernt
  - Alle seine *Nachbedingungen erfüllt*, d.h. auf alle Plätze im Nachbereich werden Marken eingefügt

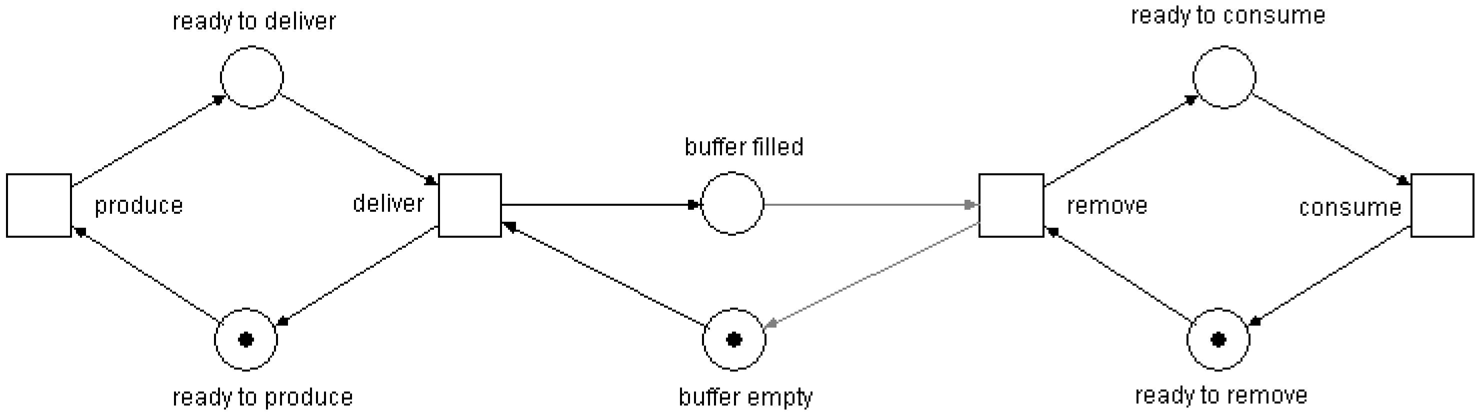


Definition: Zustand und Zustandsänderung

- Jede Untermenge $a \subseteq P_N$ von Plätzen heisst (globaler) Zustand von N
- Zustand a aktiviert eine Transition $t \in T_N$ *gwd.* $\bullet t \subseteq a$ und $(t\bullet \setminus \bullet t) \cap a = \emptyset$
- Sei $a \subseteq P_N$ ein Zustand und $t \in T_N$ eine Transition.
  Dann ist $eff(a, t) := (a \setminus \bullet t) \cup t\bullet$ der Effekt der durch das Schalten der Transition t im Zustand a entsteht.
- Sei $t \in T_N$ mit Aktivierung in einem Zustand $a \subseteq P_N$.
  Dann heisst das Tripel (a, t, eff(a, t)) ein Schritt in N.
  Notation: $a \xrightarrow{t} eff(a, t)$

Beispiel: Zustand und Zustandsänderung

- Zustand: a = {*ready to produce, buffer empty, ready to remove*}
- aktive Transition: *produce*



- nach Schalten von *produce* aktive Transition: *deliver*

Demonstration

- Demonstration eines Consumer-Producer-Netzes
- Java-Tool JARP: http://jarp.sourceforge.net/

Ende

Vielen Dank für die Aufmerksamkeit!