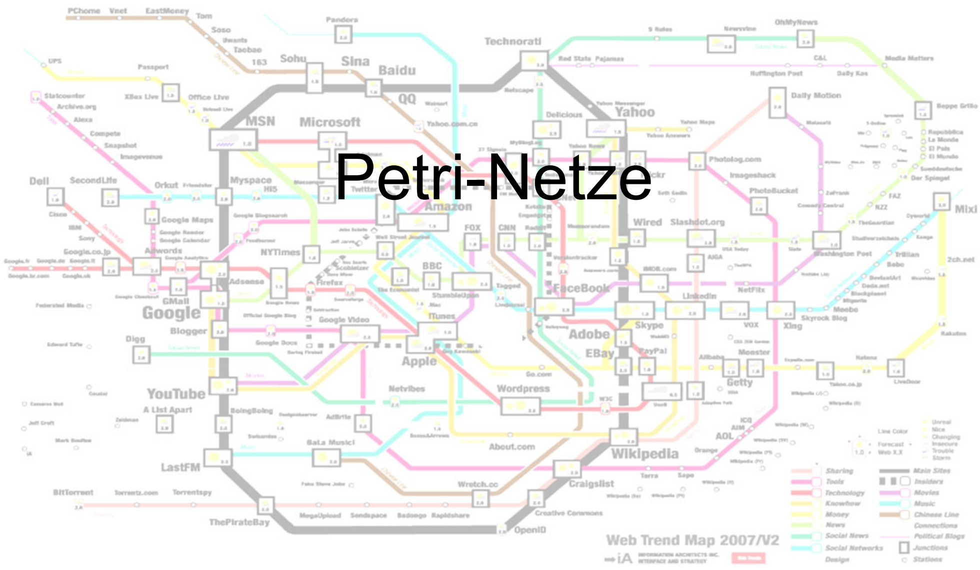
Petri-Netze



Nebenläufigkeit

Regeln
R1: ab → aa
R2: ba → bb

Es wird ein *-Termersetzungssystem betrachtet und nur die direkten, möglichen Nachfolgezustände auf das Eingabewort.

Wort
abcba

Hier sind R1 und R2 anwendbar und auch nebenläufig anwendbar; daher besteht eine echte Nebenläufigkeit.

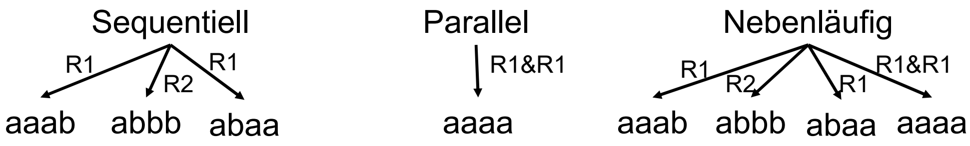
Sequentiell	Parallel	Nebenläufig
$abcba \xrightarrow{R1} aacba$, $abcba \xrightarrow{R2} abcbb$	$abcba \xrightarrow{R1\&R2} aacbb$	$abcba \xrightarrow{R1} aacba$, $abcba \xrightarrow{R2} abcbb$, $abcba \xrightarrow{R1\&R2} aacbb$

Bei der Nebenläufigkeit entstehen also (direkte Folge-)Zustände, die bei einem sequentiellen System oder auch parallelem System nicht entstehen können!
Z.B. bei Verwendung von Threads auf einem Rechner können bestimmte Zustände nicht erreicht werden, die unter Verwendung von Prozessen auf verschiedenen Rechnern möglich sind!

Nebenläufigkeit

Wort
abab

Hier sind R1 und R2 anwendbar, aber nicht nebenläufig anwendbar; daher besteht keine Nebenläufigkeit.



Das die Nebenläufigkeit als direkte Folgezustände der Vereinigung aus den Folgezuständen der sequentiellen und parallelen Verarbeitung entspricht, liegt an der Verwendung von nur zwei Regeln. Bei drei oder mehr Regeln wäre dem nicht so, weil bei der parallelen Verarbeitung alle anwendbaren Regeln gleichzeitig angewendet werden, wogegen bei der nebenläufigen Verarbeitung eine beliebige Untermenge davon zunächst nur angewendet werden kann!

Petri-Netze

*Ein **Petri-Netz** besteht aus zwei Teilen, von denen der eine als statischer Teil die logische Struktur des abzubildenden Systems modelliert, während der andere als dynamischer Teil das Verhalten des Systems im Zeitablauf wiedergibt.*

*Der **statische Teil** eines Petri-Netzes ist ein bipartiter gerichteter Graph. Dessen Ecken werden als S-Ecken, bzw. T-Ecken bezeichnet und entsprechend der Form der Buchstaben durch Kreise, bzw. Rechtecke dargestellt.*

*Für eine bestimmte Ecke v heißen die Anfangsecken der gerichteten Kanten mit der Endecke v der **Vorbereich •v** von v und die Endecken der gerichteten Kanten mit der Anfangsecke v der **Nachbereich v•** von v.*

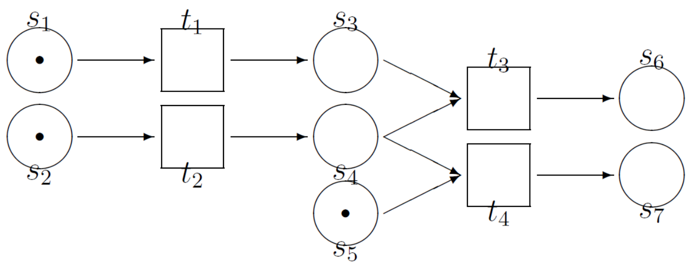
*Der **dynamische Teil** eines Petri-Netzes besteht aus (höchstens einer / mehrere (un-) unterscheidbare) Marken. Dies sind Markierungen der S-Ecken, die sich nach festgelegten Regeln ändern.*

*Eine Ecke t heißt **aktiviert**, wenn jede Ecke aus •t eine Mindestanzahl von Marken trägt und keine Ecke aus t• ihre Maximalzahl von Marken trägt. Unter dem **Schalten** einer aktivierten Ecke t versteht man die Verminderung der Anzahl der Marken bei jeder Ecke aus •t und die Erhöhung der Anzahl der Marken bei jeder Ecke aus t• um jeweils eine festgelegte Anzahl.*

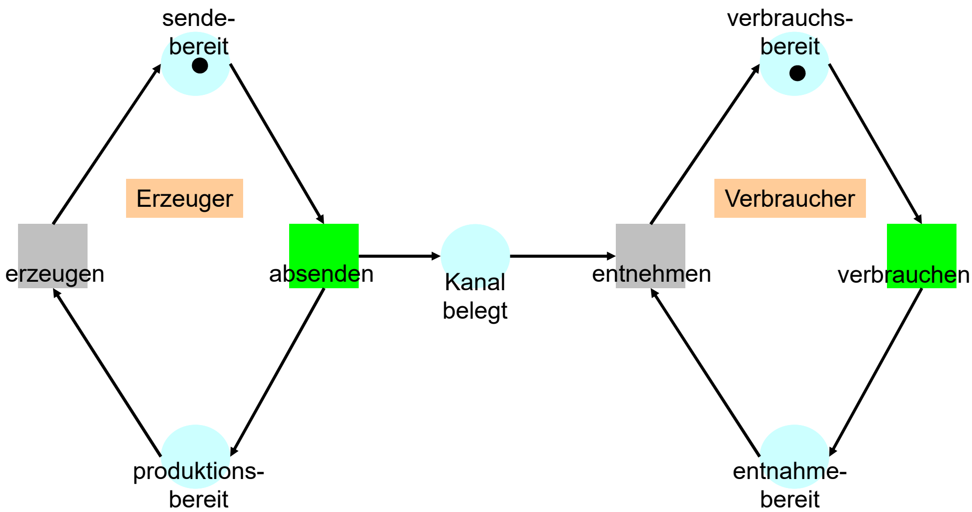
Petri-Netze

*Falls durch die Markierungen nicht eindeutig festgelegt ist, welche T-Ecke schalten darf und dass durch die willkürliche Auswahl einer zu schaltenden Ecke unterschiedlich markierte Netze entstehen, so stehen die entsprechenden T-Ecken in **Konflikt** miteinander. Durch die Möglichkeit der Auswahl wird der entsprechende Ablauf nicht deterministisch!*

*Falls ein Konflikt erst dadurch entsteht, dass zwei nicht miteinander in Konflikt stehende T-Ecken in einer bestimmten zeitlichen Reihenfolge schalten, spricht man von einer **konfusen Situation**.*



Petri-Netze: Visuelle Elemente



Bedingungs/Ereignis-Systeme

*In dem einfachsten Typ werden die **S-Ecken als Bedingungen** interpretiert, deren Markierung Auskunft darüber gibt, ob die jeweilige Bedingung erfüllt (=markiert) oder nicht erfüllt (=nicht markiert) ist. Die **T-Ecken stellen Ereignisse** dar, die genau dann stattfinden dürfen, wenn alle Bedingungen des Vorbereichs erfüllt und alle Bedingungen des Nachbereichs nicht erfüllt sind. Das Stattfinden (=Schalten) eines Ereignisses t wird im Netz dadurch dargestellt, dass alle Ecken aus •t ihre Markierungen verlieren und alle Ecken aus t• markiert werden.*

*Die Menge aller gleichzeitig erfüllten Bedingungen eines Netzes heißt ein **Fall** dieses Netzes. Als **Schritt** bezeichnet man den Übergang von einem Fall zu einem anderen Fall durch das Eintreten eines Ereignisses oder durch das gleichzeitige Eintreten mehrerer Ereignisse.*

*Ein Bedingungs/Ereignis-System heißt **zyklisch**, falls ausgehend von einem beliebigen (sinnvollen) Fall jeder andere (sinnvolle) Fall durch endlich viele Schritte erreicht werden kann.*

*Ein Bedingungs/Ereignis-System heißt **lebendig**, falls ausgehend von einem beliebigen (sinnvollen) Fall für ein beliebiges Ereignis t ein Fall herbeigeführt werden kann, so dass t aktiviert ist.*

Stellen/Transitionen-Systeme

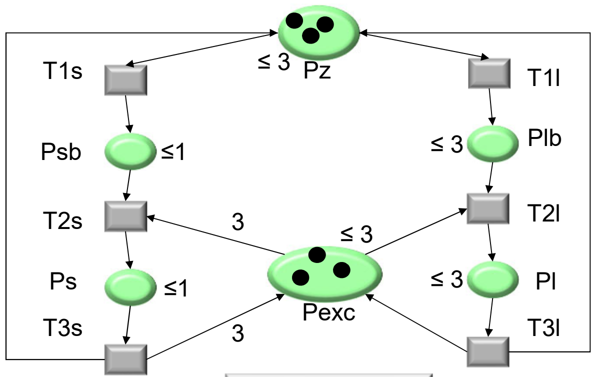
*Bei **Stellen/Transitionen-Systemen** bezeichnet man die **S-Ecken als Stellen** und die **T-Ecken als Transitionen**. Jeder Ecke s_i wird eine Kapazität von Marken, die zwischen 0 und $k_i \in N$ liegt, zugewiesen. Als zusätzliche Erweiterung können Kanten s_it_j [bzw. t_is_j] ein Gewicht w_{ij} erhalten, das angibt, um wie viele Marken sich die Markierung der Stelle ändert, wenn die Transition schaltet.*

*Ein Petri-Netz heißt **sicher** bezüglich einer bestimmten Anfangsmarkierung, wenn für jede S-Ecke die Anzahl der Marken nach beliebig häufigem Schalten der T-Ecken (ohne Beachtung irgendwelcher Kapazitäten) beschränkt bleibt.*

*Ein **Deadlock** eines Petri-Netzes ist eine Menge von Stellen, die, wenn sie zu einem gewissen Zeitpunkt keine Marken mehr tragen, auch in der Folgezeit nie wieder markiert werden.*

*Ein **Trap** eines Petri-Netzes ist eine Menge von Stellen, die, wenn mindestens eine von ihnen mindestens eine Marke enthält, auch in der Folgezeit nicht alle gleichzeitig unmarkiert sein können.*

Gegenseitiger Ausschluss



schreiben
schreib**b**erechtigt
lesen
lese**b**erechtigt
exclusive

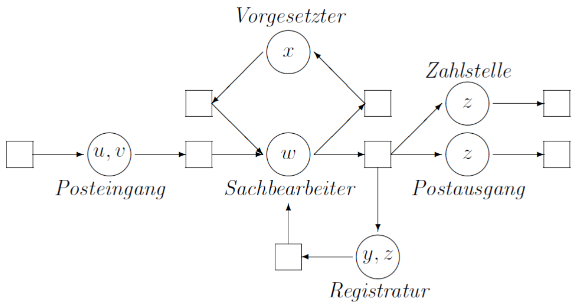
- Maximal **3 Prozesse gleichzeitig lesend**
  - $M(Pl) \leq 3$
  - $M(Pl) > 0 \Rightarrow M(Ps) = 0$
- Maximal **1 Prozess gleichzeitig schreibend**
  - $M(Ps) \leq 1$
  - $M(Ps) > 0 \Rightarrow M(Pl) = 0$
- Kein Prozess geht verloren, d.h. immer genau 3 Prozesse im Zustand Pz, Plb, Psb, Pl oder Ps:
  $M(Pz) + M(Plb) + M(Psb) + M(Pl) + M(Ps) = 3$

Sicherheits-Anforderungen

- Maximal **3 Prozesse gleichzeitig lesend**
  - $M(Pl) \leq 3$
  - $M(Pl) > 0 \Rightarrow M(Ps) = 0$
- Maximal **1 Prozess gleichzeitig schreibend**
  - $M(Ps) \leq 1$
  - $M(Ps) > 0 \Rightarrow M(Pl) = 0$
- Kein Prozess geht verloren, d.h. immer genau 3 Prozesse im Zustand Pz, Plb, Psb, Pl oder Ps:
  - $M(Pz) + M(Plb) + M(Psb) + M(Pl) + M(Ps) = 3$

Prädikat/Ereignis-Systeme

*Bei **Prädikat/Ereignis-Systemen** bezeichnet man die **S-Ecken als Prädikate** und die **T-Ecken als Ereignisse**. Die Marken sind unterscheidbar und repräsentieren die unterschiedlichen Objekte, die das zugehörige Prädikat erfüllen. Ähnlich den Stellen/Transitionen-Systemen können Ecken unterschiedliche Kapazitäten der jeweiligen Marken zugewiesen werden und den Kanten entsprechend unterschiedliche Gewichte.*



Aufgaben

1. Modellieren Sie die Vorbereitung eines Ausflugs als Bedingungs/Ereignis-System. Beachten Sie dabei folgende Notwendigkeiten:
   a) Zunächst ist das Reiseziel zu bestimmen.
   b) Danach ist das Verkehrsmittel, Pkw oder Bahn, auszuwählen.
   c) Parallel zu b) darf der benötigte Proviant eingekauft werden.
   d) Nach b) und c) wird entschieden, welches Gepäck sonst noch mitgenommen werden soll.
   e) m Fall einer Autofahrt ist das Auto aufzutanken; im Fall einer Bahnfahrt sind Fahrkarten zu kaufen.
   f) Wenn alles dies erledigt ist, kann es losgehen.

Aufgaben

2. Im nachfolgenden werden Definitionen von Petri-Netzklassen aufgeführt. Ordnen Sie die Netzwerke aus den folgenden Abbildungen diesen Definitionen zu. Erläutern Sie warum ggf. dass Netz zu der Klasse dazu gehört bzw. warum es nicht dazu gehört.
   - **a)** **Zustandsmaschinen** (ZM) haben ausschließlich unverzweigte Transitionen, die außerdem nicht am absoluten Rand liegen: Eine Zustandsmaschine ist ein Netz, bei dem jede Transition der Vorbereich und der Nachbereich nur genau eine Stelle beinhalten.
   - **b)** **Synchronisationsgraphen** (SG) haben ausschließlich unverzweigte Stellen, die außerdem nicht am absoluten Rand liegen: Ein Synchronisationsgraph ist ein Netz, bei dem jede Stelle der Vorbereich und der Nachbereich nur genau eine Transition beinhalten.
   - c) Bei verallgemeinerten Zustandsmaschinen (VZM) bzw. Synchronisationsgraphen (VSG) sind absolute Randknoten beider Arten zugelassen: Eine **verallgemeinerte Zustandsmaschine** ist ein Netz, bei dem jede Transition der Vorbereich und der Nachbereich maximal eine Stelle beinhalten.
   - d) Ein **verallgemeinerter Synchronisationsgraph** ist ein Netz, bei dem jede Stelle der Vorbereich und der Nachbereich maximal eine Transition beinhalten.

Informatik
Hochschule für angewandte Wissenschaften Hamburg

Aufgaben

2.