

Script generated by TTT

Title: Grundlagen_Betriebssysteme (02.11.2015)

Date: Mon Nov 02 13:48:38 CET 2015

Duration: 86:34 min

Pages: 30

C:\www\lgbs-ws1516\whiteboard\lgbs_course3.3.html

Favoriten | Vorgeschlagene Sites | Einf | GBS

Modellierung paralleler Systeme

In diesem Abschnitt behandeln wir grundlegende Mechanismen zur Modellierung und Beschreibung parallel ablaufender Systeme.

[Modellierungsziele](#)

[Verhaltensbeschreibung](#)

[Ereignisse und Aktionsstrukturen](#)

[Aktionen als Zustandsübergänge](#)

[Petri-Netze](#)

Generated by Targeteam



Ereignisse und Aktionsstrukturen



In diesem Abschnitt wird der Begriff **Prozess** anhand von Aktionsstrukturen mathematisch gefasst. Dies ermöglicht die mathematische Modellierung beliebiger, nichtkontinuierlicher ("diskreter") Abläufe verteilter Systeme, insbesondere von Prozessen in Rechenanlagen.

[Prozess](#)

[Kausale Abhängigkeiten](#)

[Sequentialisierung](#)

Generated by Targeteam



Prozess



Gegeben seien eine Menge (das "Universum") E^* von **Ereignissen** (engl. events) und eine Menge A von **Aktionen** (engl. actions).

Definition

Ein Triple $p = (E, \leq, \alpha)$ nennen wir einen **Prozess** oder auch eine **Aktionsstruktur**, falls folgende Aussagen gelten:

$E \subseteq E^*$, E heißt die Ereignismenge.

\leq ist eine partielle Ordnung über E , \leq ist die Kausalitätsrelation.

$\alpha: E \rightarrow A$ ist die Aktionsmarkierung des Prozesses p .

Die Abbildung α ordnet jedem Ereignis eine Aktion zu. Leerer Prozess = Aktionsstruktur mit leerer Ereignismenge.

[Beispiel: Fußgängerübergang](#)

[Charakterisierung von Prozessen](#)

Generated by Targeteam



Wir betrachten einen Prozess mit 14 Ereignissen. Jedem Ereignis werden Aktionen zugeordnet.

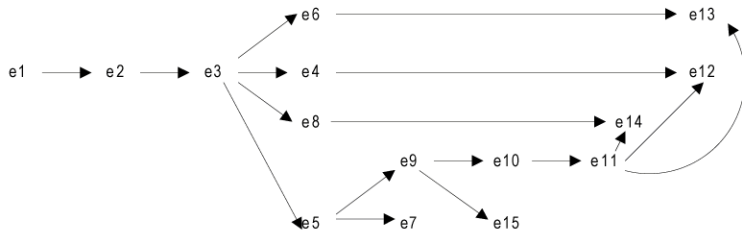
Ereignis	Aktion
e1	Knopf wird gedrückt
e2	Ampel für Autofahrer schaltet auf Gelb
e3	Ampel für Autofahrer schaltet auf Rot
e4	Auto hält auf Spur 1
e5	Ampel für Fußgänger schaltet auf Grün
e6	Auto hält auf Spur 2
e7	Fußgänger benutzt Fußgängerübergang
e8	Auto hält auf Spur 3
e9	Ampel für Fußgänger schaltet auf Rot
e10	Ampel für Autofahrer schaltet auf Rot und Gelb
e11	Ampel für Autofahrer schaltet auf Grün
e12	Auto fährt an auf Spur 1
e13	Auto fährt an auf Spur 2
e14	Auto fährt an auf Spur 3
e15	Knopf wird gedrückt

Die Relation der kausalen Abhängigkeiten sei gegeben durch:

- $e1 \leq e2, e2 \leq e3, e3 \leq e4, e3 \leq e5, e3 \leq e6, e5 \leq e7,$
- $e3 \leq e8, e5 \leq e9, e9 \leq e10, e10 \leq e11, e11 \leq e12,$
- $e11 \leq e13, e11 \leq e14, e4 \leq e12, e6 \leq e13, e8 \leq e14, e9 \leq e15.$



Endliche Prozesse lassen sich anschaulich durch knotenmarkierte, gerichtete, zyklensfreie Graphen darstellen. Die Knoten repräsentieren die Ereignisse und sind durch die zugeordneten Aktionen markiert.



Generated by Targeteam



e1	Knopf wird gedrückt
e2	Ampel für Autofahrer schaltet auf Gelb
e3	Ampel für Autofahrer schaltet auf Rot
e4	Auto hält auf Spur 1
e5	Ampel für Fußgänger schaltet auf Grün
e6	Auto hält auf Spur 2
e7	Fußgänger benutzt Fußgängerübergang
e8	Auto hält auf Spur 3
e9	Ampel für Fußgänger schaltet auf Rot
e10	Ampel für Autofahrer schaltet auf Rot und Gelb
e11	Ampel für Autofahrer schaltet auf Grün
e12	Auto fährt an auf Spur 1
e13	Auto fährt an auf Spur 2
e14	Auto fährt an auf Spur 3
e15	Knopf wird gedrückt

*gleiche
Aktion
mehrfach
ausgeführt
=> mehrere
Ereignisse*

Die Relation der kausalen Abhängigkeiten sei gegeben durch:

- $e1 \leq e2, e2 \leq e3, e3 \leq e4, e3 \leq e5, e3 \leq e6, e5 \leq e7,$
- $e3 \leq e8, e5 \leq e9, e9 \leq e10, e10 \leq e11, e11 \leq e12,$
- $e11 \leq e13, e11 \leq e14, e4 \leq e12, e6 \leq e13, e8 \leq e14, e9 \leq e15.$

partielle Ordnung beschreibt Kausalität zwischen Ereignissen; e12 findet nur nach e4 und e11 statt.



Ereignisse haben eine eindeutige Identität.

Parallel, nebenläufig

Für einen Prozess $p = (E, \leq, \alpha)$ nennen wir zwei Ereignisse $e1, e2 \in E$ **parallel** oder **nebenläufig** (engl. concurrent), falls sie im Prozess p nicht in einer kausalen Relation stehen, d.h. wenn gilt:

$$\neg(e1 \leq e2 \text{ oder } e2 \leq e1)$$

Parallele Ereignisse sind **kausal unabhängig**, sie können zeitlich nebeneinander oder in beliebiger Reihenfolge stattfinden.

Sequentiell

Ein Prozess $p = (E, \leq, \alpha)$ heißt **sequentiell**, wenn in ihm kein Paar von parallelen Ereignissen auftritt, d.h. die Kausalitätsrelation \leq eine lineare Ordnung ist.

Der Ablauf eines Programmes kann auch als Prozess im Sinne der Aktionsstrukturen gedeutet werden. Sequentielle Programme erzeugen sequentielle Prozesse.

Endlich

Ein Prozess $p = (E, \leq, \alpha)$ heißt **endlich**, falls seine Ereignismenge eine endliche Menge ist.

Generated by Targeteam



Unterschiedliche Ursachen für die kausale Beziehung $e \leq d$.

Echt kausale Beziehungen

das Ereignis e ist kausal für das Ereignis d in dem Sinn, dass d ohne e niemals auftreten kann.

Beispiel

e = Geldeinwurf, d = Kartenausgabe; $e \leq d$, d.h. der Fahrkartenautomat gibt erst dann eine Fahrkarte aus, wenn der passende Geldbetrag eingeworfen wurde.

zeitliche Beziehungen

das Ereignis e endet, bevor das Ereignis d beginnt.

Beispiel: Nachricht muss gesendet sein, bevor sie empfangen werden kann.

Happend-before

Systembeschränkungen

Systembeschränkungen, z.B. wechselseitiger Ausschluss. Das Ereignis e darf nicht parallel zum Ereignis d auftreten.

Beispiel

e = Fußgänger überquert die Fahrbahn beim Übergang, und

d = Auto überfährt den Übergang.

e und d dürfen nicht parallel ausgeführt werden, aber es besteht keine echte kausale Abhängigkeit zwischen e und d .

Generated by Targeteam



Die kausale Beziehung impliziert also eine **zeitliche Relation**; das ist die bekannte **happened before**-Beziehung von L. Lamport. Die Umkehrung gilt nicht. Die zeitliche Relation zwischen Ereignissen sagt nichts über die Kausalität aus.

Wichtig: Ereignisse wurden als atomare Ereignisse modelliert. D.h. Beginn und Ende fallen auf einen Zeitpunkt zusammen.

Die **Happend-before Relation** spielt im Kontext von Verteilten Anwendungen eine wichtige Rolle (siehe Vorlesung **Verteilte Anwendungen**).

Generated by Targeteam



Unterschiedliche Ursachen für die kausale Beziehung $e \leq d$.

Echt kausale Beziehungen

das Ereignis e ist kausal für das Ereignis d in dem Sinn, dass d ohne e niemals auftreten kann.

Beispiel

e = Geldeinwurf, d = Kartenausgabe; $e \leq d$, d.h. der Fahrkartenautomat gibt erst dann eine Fahrkarte aus, wenn der passende Geldbetrag eingeworfen wurde.

zeitliche Beziehungen

das Ereignis e endet, bevor das Ereignis d beginnt.

Beispiel: Nachricht muss gesendet sein, bevor sie empfangen werden kann.

Happend-before

Systembeschränkungen

Systembeschränkungen, z.B. wechselseitiger Ausschluss. Das Ereignis e darf nicht parallel zum Ereignis d auftreten.

Beispiel

e = Fußgänger überquert die Fahrbahn beim Übergang, und

d = Auto überfährt den Übergang.

e und d dürfen nicht parallel ausgeführt werden, aber es besteht keine echte kausale Abhängigkeit zwischen e und d .

Generated by Targeteam



In diesem Abschnitt wird der Begriff **Prozess** anhand von Aktionsstrukturen mathematisch gefasst. Dies ermöglicht die mathematische Modellierung beliebiger, nichtkontinuierlicher ("diskreter") Abläufe verteilter Systeme, insbesondere von Prozessen in Rechenanlagen.

Prozess

Kausale Abhängigkeiten

Sequentialisierung

Generated by Targeteam



Sequentialisierung



Idee: vereinfachte Darstellung paralleler Abläufe, aus der Sicht eines Beobachters. Vollständige Sequentialisierung: partielle Kausalitätsordnung zu linearer Ordnung ergänzen.

Definition

Sequentieller Beobachter hält Ereignisse bzw. Aktionen in einem sequentiellen Ablaufprotokoll fest \Rightarrow Sequentialisierung eines Prozesses mit parallelen Aktionen.

Jeder endliche Prozess p ist durch die Menge seiner vollständigen Sequentialisierungen eindeutig bestimmt. Darstellung einer Programmausführung als Prozess.

Beispiel

Für das Beispiel des Fußgängerübergangs ist folgendes eine vollständige Sequentialisierung.

$e_1 \rightarrow e_2 \rightarrow e_3 \rightarrow e_4 \rightarrow e_6 \rightarrow e_5 \rightarrow e_7 \rightarrow e_8 \rightarrow e_9$

$e_{10} \rightarrow e_{11} \rightarrow e_{14} \rightarrow e_{12} \rightarrow e_{13} \rightarrow e_{15}$

Spuren

Generated by Targteam



Definition



Ein Prozess $p_1 = (E_1, \leq_1, \alpha_1)$ heißt eine Sequentialisierung eines Prozesses $p_2 = (E_2, \leq_2, \alpha_2)$, falls gilt:

$$E_1 = E_2$$

$$\forall e, d \in E_1 : e \leq_2 d \Rightarrow e \leq_1 d$$

$$\alpha_1 = \alpha_2$$

Ist p_1 sequentiell, so heißt die Sequentialisierung **vollständig**.

Generated by Targteam



Sequentialisierung



Idee: vereinfachte Darstellung paralleler Abläufe, aus der Sicht eines Beobachters. Vollständige Sequentialisierung: partielle Kausalitätsordnung zu linearer Ordnung ergänzen.

Definition

Sequentieller Beobachter hält Ereignisse bzw. Aktionen in einem sequentiellen Ablaufprotokoll fest \Rightarrow Sequentialisierung eines Prozesses mit parallelen Aktionen.

Jeder endliche Prozess p ist durch die Menge seiner vollständigen Sequentialisierungen eindeutig bestimmt. Darstellung einer Programmausführung als Prozess.

Beispiel

Für das Beispiel des Fußgängerübergangs ist folgendes eine vollständige Sequentialisierung.

$e_1 \rightarrow e_2 \rightarrow e_3 \rightarrow e_4 \rightarrow e_6 \rightarrow e_5 \rightarrow e_7 \rightarrow e_8 \rightarrow e_9$

$e_{10} \rightarrow e_{11} \rightarrow e_{14} \rightarrow e_{12} \rightarrow e_{13} \rightarrow e_{15}$

Spuren

Generated by Targteam



Spuren



Darstellung sequentieller Prozesse mittels Spuren.

Sei A eine Menge von Aktionen, dann bezeichnen wir mit A^+ die Menge der endlichen Folgen von Aktionen aus A und mit A^∞ die Menge der unendlichen Folgen von Aktionen.

Jedem sequentiellen Prozess können wir eindeutig eine Folge von Aktionen zuordnen, wie sprechen von der **Spur** (engl. trace). Sei $p = (E, \leq, \alpha)$ ein sequentieller Prozess. Wir definieren:

$$\text{spur} : \{p \mid p \text{ ist sequentieller Prozess}\} \rightarrow A^+ \cup A^\infty$$

$$\text{spur}(p) = \text{empty, falls } E = \emptyset$$

$$\text{spur}(p) = \langle a \rangle \cdot \text{spur}(p \upharpoonright E \setminus \{e\}), \text{ falls } E \neq \emptyset, \text{ wobei } e \text{ das gemäß der Kausalitätsordnung kleinste Ereignis in } p \text{ ist und } \alpha(e) = a \text{ gilt.}$$

$p \upharpoonright E \setminus \{e\}$ bezeichnet den Teilprozess, der durch die Einschränkung auf die Ereignismenge $E \setminus \{e\}$ entsteht.

Für einen nichtsequentiellen Prozess p gilt:

$$\text{Spuren}(p) = \{\text{spur}(q) \mid \text{Prozess } q \text{ ist eine vollständige Sequentialisierung von } p\}$$



Generated by Targteam



In diesem Abschnitt wird der Begriff **Prozess** anhand von Aktionsstrukturen mathematisch gefasst. Dies ermöglicht die mathematische Modellierung beliebiger, nichtkontinuierlicher ("diskreter") Abläufe verteilter Systeme, insbesondere von Prozessen in Rechenanlagen.

Prozess

Kausale Abhängigkeiten

Sequentialisierung

Generated by Targeteam



Zustände und Zustandsänderungen zur Modellierung des Systemverhaltens; Zustandsraum = Menge aller Systemzustände

Interpretierte Aktionen

Aktionen werden Zustandsänderungen als Bedeutung zugeordnet.

Modell

Zu diesem Zweck definieren wir **nichtdeterministische Zustandsautomaten mit Transitionsaktionen**.

Zustandsautomat

Beispiel Fahrkartenautomat

Aktionsspur

Konflikte

Generated by Targeteam



Ein nichtdeterministischer Zustandsautomat ist gegeben durch:

S, eine Menge von Zuständen, genannt **Zustandsraum**,

A, eine Menge von **Transitionsaktionen**,

$R \subseteq S \times A \times S$ eine **Zustandsübergangsrelation**,

Seien $s_0, s_1 \in S$ und $a \in A$ gegeben. $(s_0, a, s_1) \in R$ bedeutet, dass im Zustand s_0 die Aktion a ausgeführt werden kann und dies zum Nachfolgezustand $s_1 \in S$ führen kann.

Diese Art von Automaten heißt **nichtdeterministisch**, da in einem Zustand mehrere Transitionsaktionen möglich sein können und eine Transitionsaktion zu unterschiedlichen Nachfolgezuständen führen kann.

Wir schreiben (für gegebene Relation R) $s_0 \rightarrow_a s_1$, um auszudrücken, dass $(s_0, a, s_1) \in R$ gilt.

S_0 eine Menge von möglichen Anfangszuständen.

Generated by Targeteam



Akzeptiert werden 1- und 2 DMark Münzen. Mittels zweier Knöpfe kann man zwischen einer Kurzstrecke für 1 DM oder einer normalen Fahrt für 2 DM wählen. Der Automat gibt Wechselgeld zurück.

Menge der Zustände

$S \subseteq S_1 \times S_2$ mit

$S_1 = \{\text{Wahl, kurz, normal}\}$ und $S_2 = \{-1, 0, 1, 2\}$

mit der folgenden Interpretation eines Zustandes $s = (x, y)$:

$x = \text{Wahl}$: Automat wartet auf die Wahl Taste

$x = \text{kurz}$: eine Kurzstrecken Karte wurde gewählt und ist noch auszugeben

$x = \text{normal}$: eine normale Fahrkarte wurde gewählt und ist noch auszugeben

$y = -1$: es ist noch 1 DM zurückzugeben

$y = 0$: es ist kein Geld mehr einzuwerfen oder zurückzugeben

$y = 1$: es muss noch 1 DM eingeworfen werden

$y = 2$: es muss noch 2 DM eingeworfen werden

Menge der Transitionsaktionen

$A = \{\text{Wk, Wn, E1, E2, R1, Ak, An}\}$, mit

Wk: Wahl einer Kurzstrecken Karte

Wn: Wahl einer normalen Fahrkarte

E1: Einwurf eines 1DM Stückes

E2: Einwurf eines 2DM Stückes

R1: Rückgabe eines 1DM Stückes

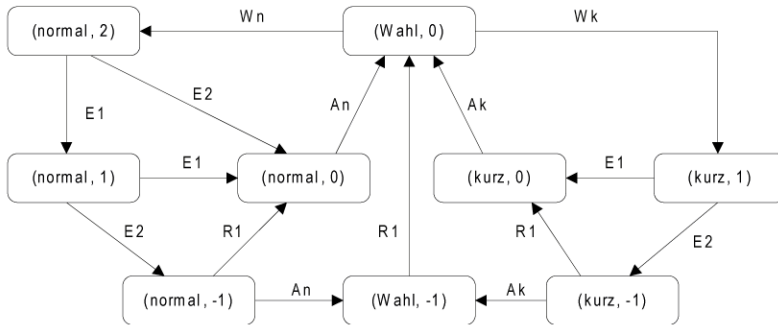
Ak: Ausgabe einer Fahrkarte für Kurzstrecke

An: Ausgabe einer Fahrkarte für normale Fahrt





Zustandsübergänge



Generated by Targeteam



Aktionsspur



Jedem Zustandsautomaten lassen sich ausgehend von der gegebenen Menge von Anfangszuständen Spuren zuordnen.

Definition

Gegeben sei ein Zustandsautomat $Z = (S, A, R, S_0)$. Eine Folge a_i , wobei $1 \leq i \leq k$ mit $k \in \mathbb{N} \cup \{\infty\}$, ist eine endliche oder unendliche **Aktionsspur** des Zustandsautomaten, falls eine Folge von Zuständen s_i existiert, mit $s_i \in S$, $s_1 \in S_0$ und

$$s_{i-1} \xrightarrow{a_i} s_i \text{ für alle } i \text{ mit } 1 < i \leq k$$

Eine Aktion a heißt **deterministisch**, wenn für jeden Zustand $s \in S$ höchstens ein Nachfolgezustand $s_i \in S$ existiert mit

$$s \xrightarrow{a} s_i$$

Beispiel

Generated by Targeteam



Konflikte



Aktionen und deren Zustandsänderungen können in Konflikt zueinander stehen.

Beispiel

Mögliche Konflikte bei parallel durchzuführenden Zuweisungen. Wir betrachten die Aktionen, die den folgenden beiden Zuweisungen entsprechen:

$$(1) \ x = x + 1$$

$$(2) \ x = 2 * x$$

Die Zuweisungen stehen im Konflikt zueinander.

Werden sie parallel ausgeführt, so kann dem entsprechenden Prozess keine Zustandsänderung eindeutig zugeordnet werden; wir sagen auch, die Aktionen **stören** sich wechselseitig.

Störungsfreiheit

Generated by Targeteam



Definition Störungsfreiheit



Gegeben sei ein Prozess $p = (E, \leq, \alpha)$. Der Prozess p heißt **störungsfrei**, genau dann, wenn für jedes Paar von Ereignissen $e_1, e_2 \in E$ gilt:

1.) $e_1 \leq e_2$ oder $e_2 \leq e_1$ oder

$$2.) \ \forall (\alpha(e_1)) \cap N(\alpha(e_2)) = N(\alpha(e_1)) \cap \forall (\alpha(e_2)) = N(\alpha(e_1)) \cap N(\alpha(e_2)) = \emptyset.$$

Die Bedingung (2) nennt man auch **Bernstein-Bedingungen**.

Generated by Targeteam



Zustände und Zustandsänderungen zur Modellierung des Systemverhaltens; Zustandsraum = Menge aller Systemzustände

Interpretierte Aktionen

Aktionen werden Zustandsänderungen als Bedeutung zugeordnet.

Modell

Zu diesem Zweck definieren wir *nichtdeterministische Zustandsautomaten mit Transitionsaktionen*.

Zustandsautomat

Beispiel Fahrkartenautomat

Aktionsspur

Konflikte

Generated by Targeteam



Formalismus von C.A. Petri 1962 entwickelt. Ansatz führte zu einer Theorie zur Analyse und Entwicklung nebenläufiger Systeme.

informelle Charakterisierung

Ein **Petri-Netz** ist gerichteter Graph mit Kanten und zweierlei Knoten.

Knoten: Stellen (graphisch Kreise) und Transitionen (graphisch Rechtecke)

Kanten: von Stellen zu Transitionen oder von Transitionen zu Stellen

Belegung der Stellen mit Marken/Werten (Token);

In einem booleschen Netz sind als Werte nur 0 oder 1 zugelassen.

In einem Stellen/Transitionenetz sind für die Belegung der Stellen natürlichzahlige Werte zugelassen; die maximale Belegung definiert die Stellenkapazität.

Zustand: definiert durch Belegung der Stellen;

Zustandsübergang durch sogenannte Schaltregeln (engl. firing rule); Belegung ändert sich. Man spricht in diesem Zusammenhang vom Schalten einer Transition.

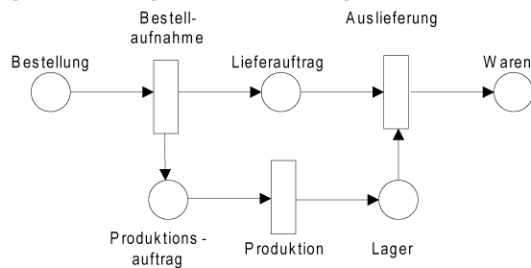
Markierung der Kanten, Kantengewichte: Gewichtung gibt an, wieviele Marken beim Schalten einer Transition von den Eingangsknoten (Stellen) der Transition abgezogen und den Ausgangsknoten (Stellen) der Transition hinzugefügt werden.

Beispiel eines Petri-Netzes

Generated by Targeteam



grobe Modellierung einer Materialverwaltung.



Generated by Targeteam



Ein Petri-Netz ist ein Tripel (S, T, F) mit

S ist eine endliche Menge von **Stellen** (engl. place)

T ist eine endliche Menge von **Transitionen** (engl. transition) und es gilt: $S \cap T = \emptyset$ d.h. Stellen und Transitionen sind disjunkt.

F ist die **Flussrelation** mit $F \subseteq (S \times T) \cup (T \times S)$

Für einen Knoten $x \in (S \cup T)$ gilt:

$\cdot x = \{y \mid y F x\}$ den Vorbereich von x

$x \cdot = \{y \mid x F y\}$ den Nachbereich von x

Mit obiger Definition ist die statische Struktur eines Netzes formal erfasst.

Für das Beispiel Materialverwaltung gilt beispielsweise:

$\cdot \text{Bestellaufnahme} = \{\text{Bestellung}\}$

$\text{Bestellaufnahme} \cdot = \{\text{Produktionsauftrag, Lieferauftrag}\}$

Verfeinerung

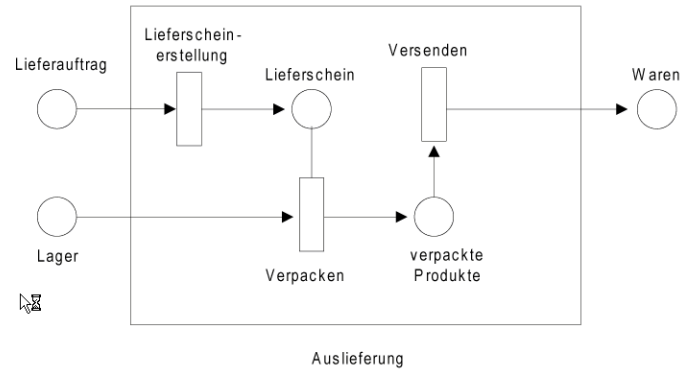
Generated by Targeteam



Netzstrukturen können schrittweise verfeinert, konkretisiert werden.

Beispiel

Verfeinerung der Materialverwaltung: z.B. der Komponente Auslieferung. Anhand des Lieferauftrages wird der Lieferschein geschrieben, der zusammen mit dem Produkt verpackt und versandt wird.



Generated by Targeteam

Ein Petri-Netz ist ein Tripel (S, T, F) mit

S ist eine endliche Menge von **Stellen** (engl. place)

T ist eine endliche Menge von **Transitionen** (engl. transition) und es gilt: $S \cap T = \emptyset$ d.h. Stellen und Transitionen sind disjunkt.

F ist die **Flussrelation** mit $F \subseteq (S \times T) \cup (T \times S)$

Für einen Knoten $x \in (S \cup T)$ gilt:

$\cdot x = \{y \mid y F x\}$ den Vorbereich von x

$x \cdot = \{y \mid x F y\}$ den Nachbereich von x

Mit obiger Definition ist die statische Struktur eines Netzes formal erfasst.

Für das Beispiel Materialverwaltung gilt beispielsweise:

$\cdot \text{Bestellaufnahme} = \{\text{Bestellung}\}$

$\text{Bestellaufnahme} \cdot = \{\text{Produktionsauftrag, Lieferauftrag}\}$

Verfeinerung

Generated by Targeteam



Gegeben sei ein Petri-Netz (S, T, F) .

Eine Abbildung $c: S \rightarrow \mathbb{N} \cup \{\infty\}$ gibt die **Kapazität** einer Stelle an.

Eine Abbildung $w: F \rightarrow \mathbb{N} \cup \{0\}$ gibt die **Gewichtung** einer Kante an. Gewichtung 1, falls keine explizite Markierung.

Eine Abbildung $M: S \rightarrow \mathbb{N}$ heißt natürlichzahlige **Markierung** der Stellen. Die Markierung beschreibt einen **Zustand** des Netzes.

Es muss gelten: $\forall s \in S: M(s) \leq c(s)$

Ein solches Netz heißt **Stellen-Transitionsnetz**

Falls gilt $M: S \rightarrow \mathbb{B}$, mit $\mathbb{B} = \{0, 1\}$, dann heißt das Netz: **Bedingungs/Ereignisnetz** oder Boolesches Netz.

Generated by Targeteam