Logische Zeit in verteilten Systemen

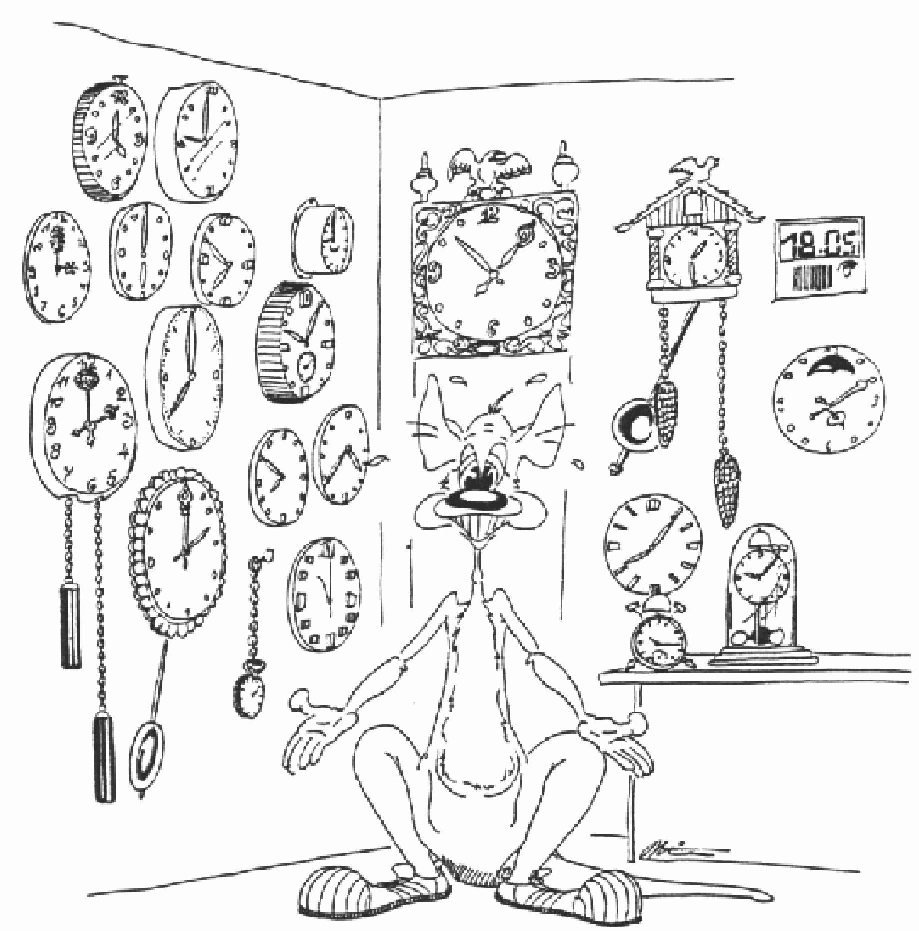


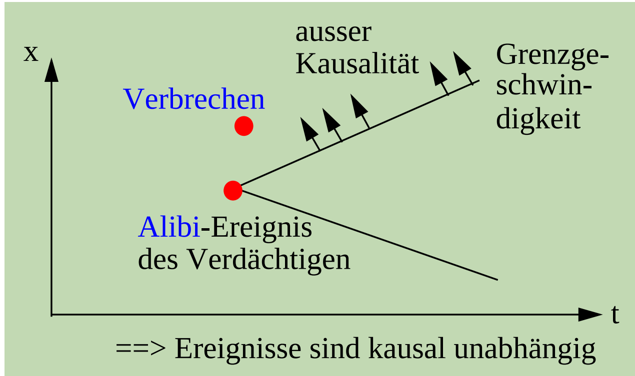
Bild: R. G. Herrtwich, G. Hommel

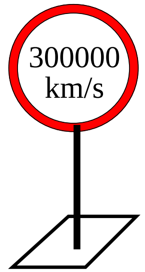
Kommt Zeit, kommt Rat

1. Volkszählung: Stichzeitpunkt in der Zukunft
   - liefert eine gleichzeitige, daher kausaltreue "Beobachtung"

2. Kausalitätsbeziehung zwischen Ereignissen ("Alibi-Prinzip")
   - wurde Y später als X geboren, dann kann Y unmöglich Vater von X sein

   --> Testen verteilter Systeme: Fehlersuche/ -ursache





"speed limit of causality"
(P. Langevin)

3. Wechselseitiger Ausschluss
   - bedient wird, wer am längsten wartet

4. Viele weitere nützliche Anwendungen in unserer "verteilten realen Welt"
   - z.B. kausaltreue Beobachtung durch "Zeitstempel" der Ereignisse

> *Le temps est un grand maître, il règle bien des choses.*
> Corneille, Sertorius

Logische Zeitstempel von Ereignissen

- Verteilte Berechnung abstrakt: n Prozesse, halbgeordnete Ereignismenge E, Nachrichten (Sende- / Empfangsereignis)
- Zweck: Ereignissen eine Zeit geben ("dazwischen" egal)
- Gesucht: Abbildung C: E --> H

Clock

"Zeitbereich":
Halbgeordnete Menge
--> "füher", "später"

- Für e ∈ E heisst C(e) *Zeitstempel* von e
- C(e) bzw. e *früher* als C(e') bzw. e', wenn C(e) < C(e')
- Wie soll H aussehen?

z.B.:

- **N** (lineare Ordnung)
- **R** (bzw. REAL-Datentyp)
- Potenzmenge von E
- $\mathbf{N}^n$ (d.h. n-dim. Vektoren)

- Sinnvolle Forderung:

Kausalrelation ("Pfad im Diagramm")

Uhrenbedingung: e < e' ==> C(e) < C(e')

Ordnungshomomorphismus

Zeitrelation "früher"

Interpretation ("Zeit ist kausaltreu"):

Wenn ein Ereignis e ein anderes Ereignis e' beeinflussen kann, dann muss e einen kleineren Zeitstempel als e' haben

Logische Uhren von Lamport

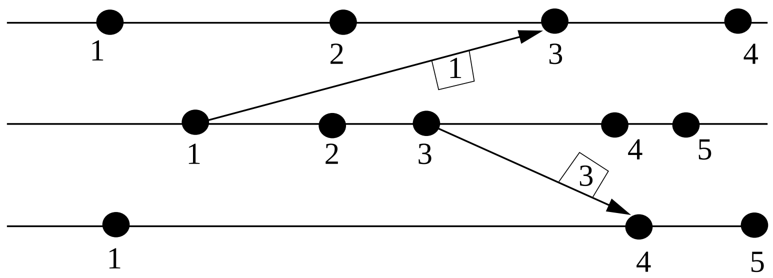
Commun. ACM 1978:
Time, Clocks, and the Ordering of Events in a Distributed System

C: (E,<) --> (**R**,<)   Zuordnung von Zeitstempeln

Kausal-
relation

(oder **N**)

e < e' ==> C(e) < C(e')   Uhrenbedingung



Protokoll zur Implementierung der Uhrenbedingung:

- Lokale Uhr (= Zähler) *tickt* "bei" *jedem* Ereignis
- Sendeereignis: Uhrwert mitsenden (*Zeitstempel*)
- Empfangsereignis: *max*(lokale Uhr, Zeitstempel)

zuerst! danach "ticken"

Behauptung:
Protokoll respektiert Uhrenbedingung

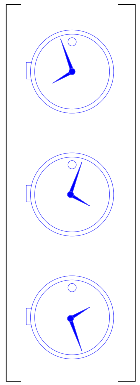
Beweis: Kausalitätspfade sind monoton...

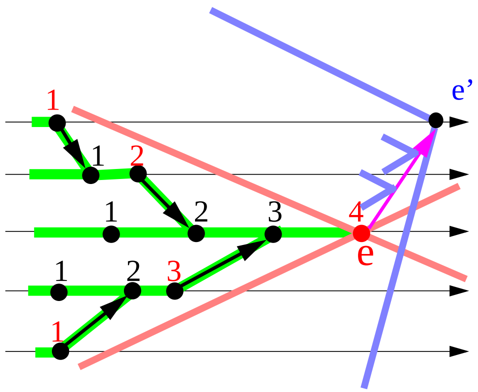
Vektorzeit

Andere Zeiten, andere Sitten

Quot tempora tot astra
G. Bruno (1548-1600)

"relativistische" Weltsicht, vgl. auch
- Nicolaus Kopernicus
- Galileo Galilei





Vektorzeit: Motivation

Umkehrung der Uhrenbedingung gilt nicht für Lamport-Zeit
- C(e) < C(e’) ==> e < e’ gilt nicht!
- es gilt nur: C(e) < C(e’) ==> e < e’ oder e||e’

Zeit := vergangene Zeit ← viele Uhren messen die Zeit, indem sie *vergangene* Sekunden zählen
:= Vergangenheit
:= Menge vergangener Ereignisse ← vgl. dies mit der *Lamport-Zeit* ("lokal vergangen")

Kausalrelation

Zeit(e) := {e’| e’ ≤ e} = *Kegel* von e

Genauer: Zeitstempel eines Ereignisses

Kann durch lokal späteste Ereignisse repräsentiert werden (linksabgeschlossen)

Hiervon gibt es n Stück (n = Anzahl der Prozesse)

!

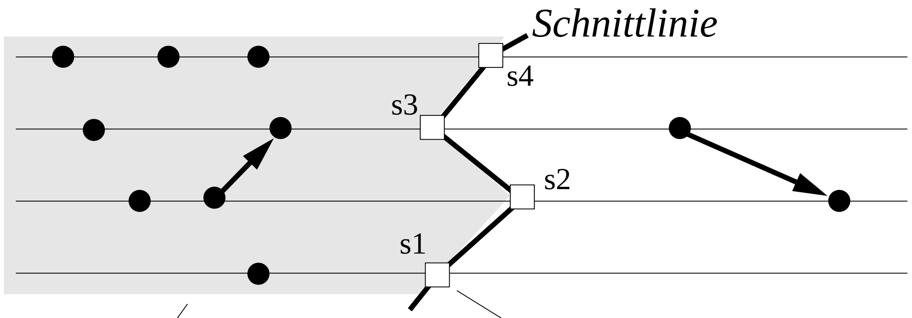
--> *Zeitstempel* ist n-dimensionaler Vektor

--> *Zeit* ist Menge n-dimensionaler Vektoren

--> *Uhr* ist ein array C[1:n]

zum Anzeigen von Zeitvektoren

Schnitt und Schnittlinie



Schnittereignis
(zusätzlich je ein Ereignis pro Prozess "festlegen")

Vergangenheit V
$\{e \mid \exists s_i: e <_l s_i\}$

- Schnittlinie trennt Zeitdiagramm / Ereignismenge in zwei disjunkte Mengen "Vergangenheit" / "Zukunft"

- Bemerkung: $e \in V \wedge e' <_l e \Longrightarrow e' \in V$
  (linksabgeschlossen bzgl. lokaler Kausalrelation)

Denkübung: Man vergleiche den Begriff des *Schnittes* (insbes. des *konsistenten Schnittes*, vgl. nachfolgende Folie) mit dem früher erwähnten Begriff der *Präfixberechnung*! Man beachte auch die Halbordnungsstruktur bzw. Verbandsstruktur dieser Begriffe.

Konsistente Schnitte

Def. **Schnitt**:

$S \subseteq E$ heisst *Schnitt* von E, falls $e \in S \wedge e' <_l e \Longrightarrow e' \in S$
(d.h. Schnitt wird mit seiner Vergangenheit identifiziert)

Def. **konsistenter Schnitt**:

$S \subseteq E$ heisst *konsistent*, falls $e \in S \wedge e' < e \Longrightarrow e' \in S$

Def.: Schnitt S *später* als S' :<==> $S' \subseteq S$

bzw. $\subset$ bei "strikt später"

Beh.: Jeder konsistente Schnitt ist ein Schnitt

Bew.: $<_l \subseteq <$

Bem.: Schnitt(linie) inkonsistent <==>
$\exists$ "Nachricht aus der Zukunft"

Bem.: Schnitt(linie) konsistent <==>
Schnittereignisse paarweise kausal unabhängig

Bew. als Übung

Bem.: Schnitt(linie) konsistent <==>
lässt sich senkrecht darstellen (Gummibandtransf.)

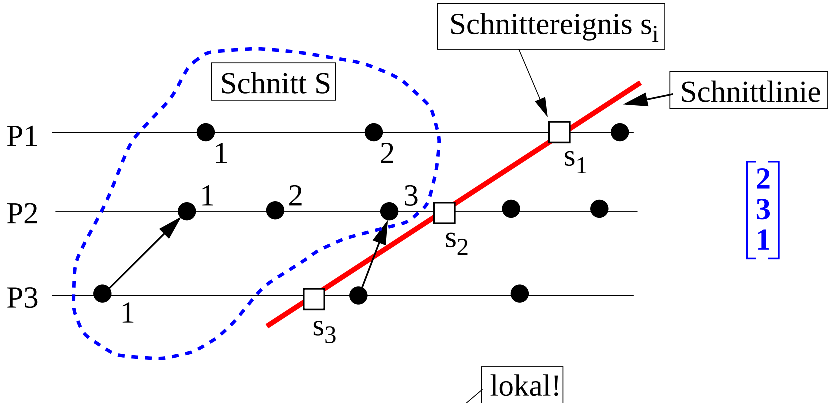
Bew. bereits bekannt: Diagramm auseinanderschneiden und versetzen

Vektorzeitstempel von Schnitten

Zeitstempel $\tau(S)$ eines Schnittes S ist ein Vektor aus $\mathbf{N}^n$

alle Ereignisse, die links von einer Schnittlinie liegen

Anzahl der Prozesse



Def. $\tau(S)[i] := |\{e \in E_i \mid e <_l s_i\}| := |S \cap E_i|$

(für jede Komponente i mit $1 \leq i \leq n$)

Interpretation:

$\tau(S)[i]$ ist die "Stelle", wo die Prozessachse von P_i durch die Schnittlinie geschnitten wird

Beachte: man kann zu *konsistenten* und *inkonsistenten* Schnitten den zugehörigen Zeitvektor definieren!

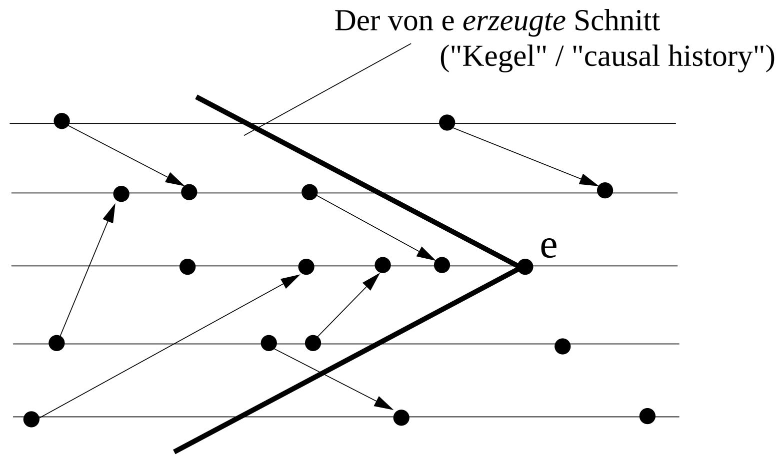
Kausale Vergangenheit eines Ereignisses

Def. *kausale Vergangenheit* $\downarrow(e)$ eines Ereignisses e:

$$\downarrow(e) = \{e' \mid e' \leq e\}$$

Beh.: $\downarrow(e)$ ist ein konsistenter Schnitt

Bew. als Übung



Beh.: $e' \leq e$ <==> $e' \in \downarrow(e)$

Beh.: $e \parallel e'$ <==> $\neg(e \in \downarrow(e')) \wedge \neg(e' \in \downarrow(e))$

(Bew. klar)

Vektorzeitstempel von Ereignissen

Kausale Vergangenheit

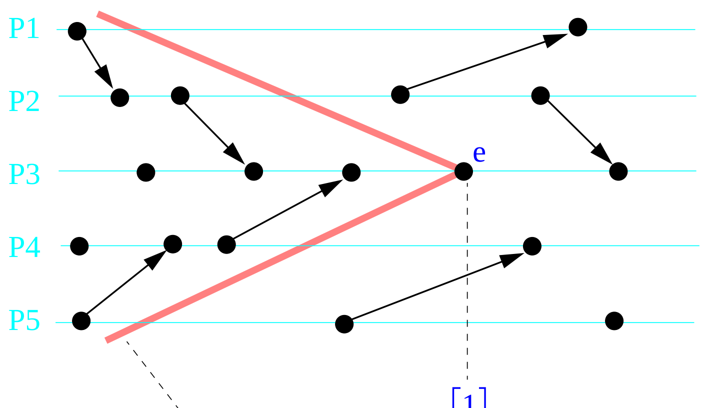
- Bezeichne $\downarrow$e den *Kegel* $\{e' \in E \mid e' \leq e\}$ von e
  - jeder Kegel ist ein konsistenter Schnitt (da linksabgeschlossen) ("Kegelmantel" könnte also als senkrechte Linie gezeichnet werden!)
  - Repräsentation durch die n lokal am weitesten rechts liegenden Ereignisse

Menge der Ereignisse von P_i

- Dann definiere $\tau(e) := \tau(\downarrow e)$

Also: $\tau(e)[i] := |\{e' \in E \mid e' \leq e\} \cap E_i| := |\{e' \in E_i \mid e' \leq e\}|$

  - das heisst: Zeitstempel eines Ereignisses = Zeitstempel seines Kegels

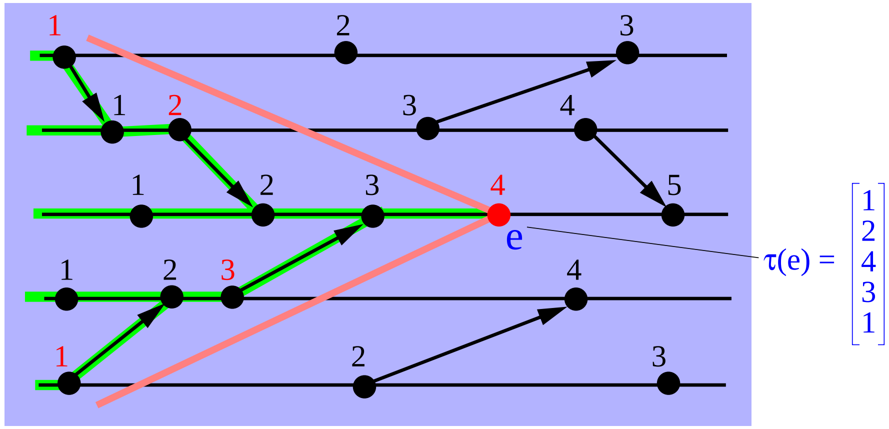


Schnittlinie in der Form eines Kegels (vgl. "Lichtkegel" der relativistischen Physik): $\downarrow$e ist ein konsistenter Schnitt

$\begin{bmatrix}1\\2\\4\\3\\1\end{bmatrix}$ $\tau(e)$: Vektorzeitstempel von e

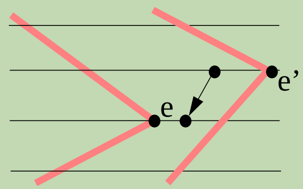
$\tau(e) < \tau(e')$

- Jeder Prozess numeriert seine Ereignisse lokal durch

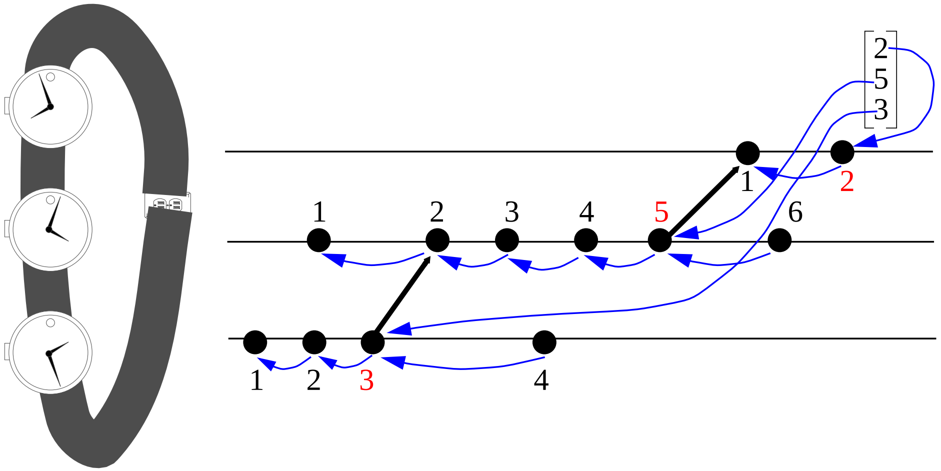


- Vektor $\tau(e)$ repräsentiert gesamte kausale Vergangenheit des Ereignisses e
  - Veranschaulichung durch einen "Kegel" $\{e' \mid e' \leq e\}$
  - Nummer des jeweils lokal letzten kausal vorangehenden Ereignisses steht in der jeweiligen Komponente des Vektors

- Interpretation von $\tau(e) < \tau(e')$:
  - e liegt in der kausalen Vergangenheit von e'
  - Kegel von e ist ganz im Kegel von e' enthalten



Vektorzeitstempel: Interpretation



- Zeigt auf jeweils jüngstes kausal vergangenes lok. Ereignis
- Damit implizit auch auf alle vorangehenden
  (wegen lokaler totaler Ordnung)
- Vektor repräsentiert *gesamte kausale Vergangenheit*
- Kodiert "*Wissen*" über (jedes einzelne) vergangene Ereignis
  Genauer: Vektorzeit repräsentiert die Kausalrelation in isomorpher Weise!

Denkübungen:

- wie stellt man fest, ob e' im Kegel von e mit τ(e) liegt?
- gibt es eine noch kompaktere Kodierung?

"Zeitstempelarithmetik"

$$\begin{bmatrix}1\\3\\4\\3\\2\end{bmatrix} \leq \begin{bmatrix}1\\7\\4\\6\\2\end{bmatrix} \qquad\qquad \begin{bmatrix}1\\3\\4\\3\\7\end{bmatrix} \parallel \begin{bmatrix}5\\3\\8\\3\\2\end{bmatrix}$$

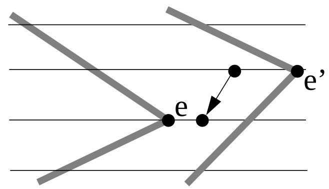
vergleichbar
(komponentenweise ≤)

"konkurrent"

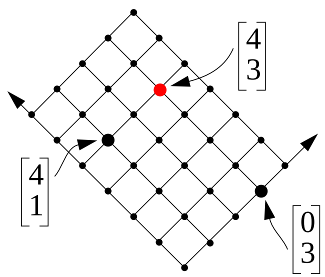
‘<‘ definiert als “≤ aber ≠”

Interpretation von τ(e) < τ(e):

- e liegt in der kausalen Vergangenheit von e’
- Kegel von e ist im Kegel von e’ enthalten



$$\sup\left(\begin{bmatrix}1\\4\\2\\3\\7\end{bmatrix}, \begin{bmatrix}8\\3\\4\\3\\2\end{bmatrix}\right) = \begin{bmatrix}8\\4\\4\\3\\7\end{bmatrix}$$

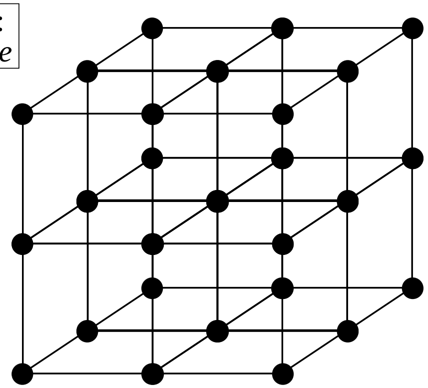


sup = komponentweises Maximum

Der Zeitverband

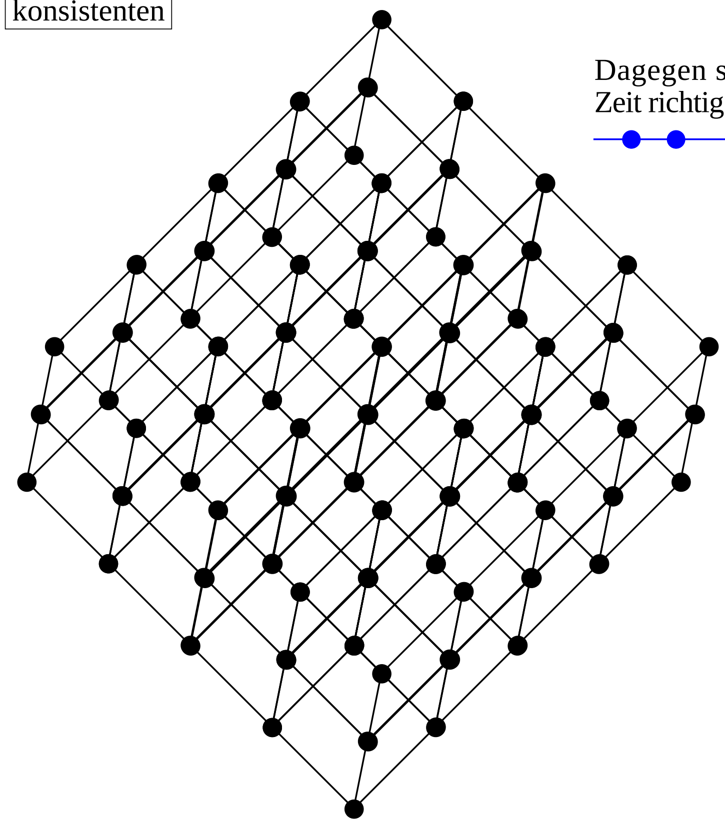
Zeitvektoren bilden bzgl. sup bzw. $\leq$ einen *Verband,* der sich in kanonischer Weise als *n-dimensionales Gitter* darstellen lässt

engl.: *lattice*



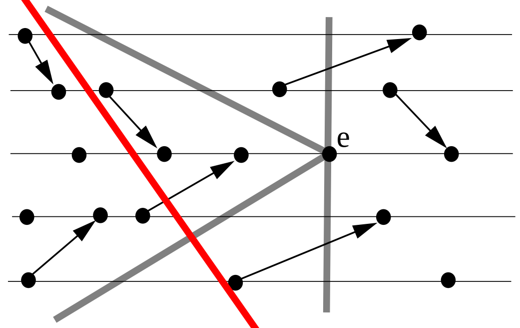
Er entspricht dem *Verband aller Schnitte* der Berechnung

nicht nur der konsistenten



Dagegen sieht lineare Zeit richtig langweilig aus!

Vektorzeit und ideale Beobachter



$$\begin{bmatrix}0\\0\\0\\0\\0\end{bmatrix} \quad \ldots \quad \begin{bmatrix}2\\4\\5\\4\\3\end{bmatrix}$$

Wahrnehmungen des idealen Beobachters

$$\tau(e) = \begin{bmatrix}1\\2\\4\\3\\1\end{bmatrix} \qquad id(e) = \begin{bmatrix}1\\3\\4\\3\\2\end{bmatrix}$$

- Numeriere Ereignisse lokal
- Idealisierter Beobachter nimmt "Ticken" sofort wahr
- Geeignete Datenstruktur hierfür: Vektor / array
- Für *jeden* Beobachter gilt stets: $\tau(e) \leq id(e)$ $(\forall e)$
  (komponentenweise '$\leq$' ; idealer Beobachter sieht stets die gesamte kausale Vergangenheit und evtl. einige weitere Ereignisse)
- $\tau(e)$ = Infimum aller idealen Sichten $id(e)$
- Beachte: $id(e)$ hängt vom Zeitdiagramm ab!
- Aber $\tau(e)$ ist invariant bzgl. Gummibandtransformation!

Daher nur zur Motivation

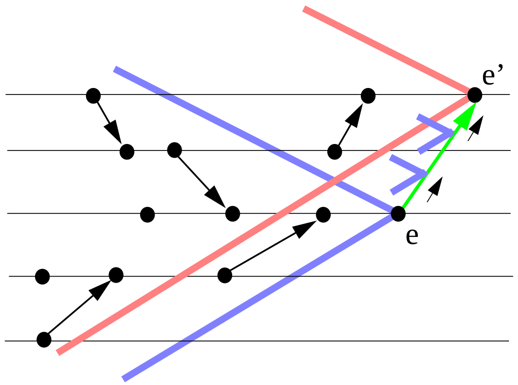
Implementierung der Vektorzeit

- Idee: Analog zur Lamport-Zeit
  (hier allerdings stets vektoriell!)

- Nachrichten enthalten die gesamte kausale Vergangenheit des Senders ==> Zeitvektor des Sendeereignisses

- Bei Empfang einer Nachricht:
  - Vereinigung der Kegel
  - ==> Supremum der Vektoren

Wissen über vergangene Ereignisse vereinigen



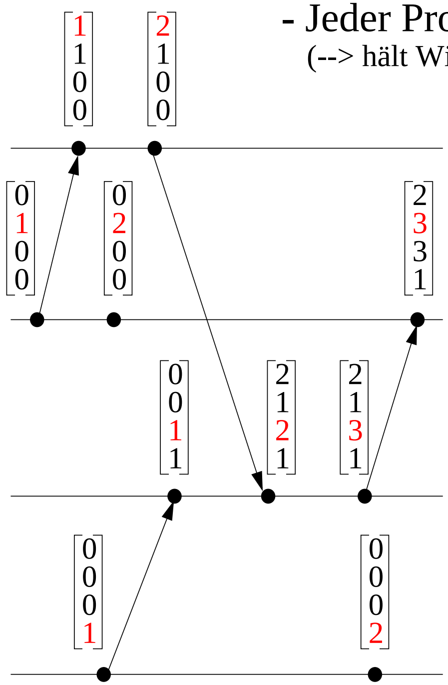
Mitschleppen des Kegels des Sendeereignisses und Vereinigung mit dem Kegel des Empfangsereignisses

--> "induktiv": ein Ereignis hat ein "vollständiges Wissen" über alle seine vergangenen Ereignisse

Propagieren des Zeitwissens

(--> Implementation der Vektorzeit)

Andere Zeiten, andere Sitten



- Jeder Prozess besitzt eine *Vektoruhr*
  (--> hält Wissen über vergangene Ereignisse)

- *bei jedem Ereignis:*
  eigene Komponente erhöhen
- *beim Senden:*
  neuen Vektor mitsenden
- *beim Empfangen:*
  komponentenweises Maximum der beiden Vektoren

Vereinigung der beiden Kegel

Kausalrelation

bzgl. Zeitvektor

- *Behauptung*: $e < e' \Leftrightarrow \tau(e) < \tau(e')$

- *Anschauliche Interpretation*:
  - $\tau(e) \leq \tau(e') \Leftrightarrow$ es gibt eine Kausalkette von e zu e'

monoton bzgl. Zeitvektoren!

- *Korollar*: $e \parallel e' \Leftrightarrow \tau(e) \parallel \tau(e')$

Interpretation: Genau die "gleichzeitigen" Ereignisse beeinflussen sich nicht ggs.

Kausal- und Zeitstruktur: Isomorphie

"Hauptsatz": $e < e' \Leftrightarrow \tau(e) < \tau(e')$

- Umkehrung der Uhrenbedingung
- gilt nicht für Lamport-Zeit

Beweis:

(1) $e \leq e' \;\Leftrightarrow\; e \in \downarrow e'$
wegen Def. von Kegel

(2) $e \in \downarrow e' \;\Leftrightarrow\; \downarrow e \subseteq \downarrow e'$
klar nach Def. von Kegel (= kons. Schnitt)

(3) $\downarrow e \subseteq \downarrow e' \;\Leftrightarrow\; \tau(\downarrow e) \leq \tau(\downarrow e')$
weil "später" sich jeweils überträgt

(4) $\tau(\downarrow e) \leq \tau(\downarrow e') \;\Leftrightarrow\; \tau(e) \leq \tau(e')$
nach Def. von $\tau(e)$

Irreflexivität folgt aus Injektivität von τ

Anschauliche Interpretation: Wissen über lokale Zeit über Kausalkette propagiert

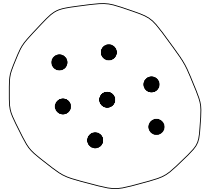
---

Verschärfung: Falls $e \in E_i$ (und $e \neq e'$):
$e < e' \Leftrightarrow \tau(e)[i] < \tau(e')[i]$

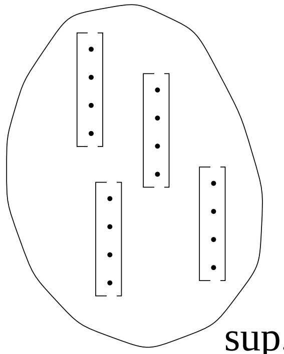
Effizienter: Nur eine Komponente testen

Rechnen mit Ereignismengen

Mengen von Ereignissen (Kegel) $\Leftrightarrow$ *Zeitvektoren*



$\cap \cup \subseteq$ — Kausalität

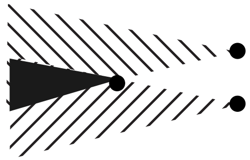


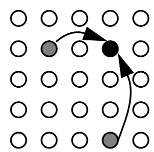
Zeit

sup, inf, $\leq$

Mengentheoretische Operationen $\Leftrightarrow$ Algebraische Operationen (--> "rechnen")

Verbandsstruktur auf 2^E (Ideale) $\Leftrightarrow$ Produduktverband auf $\mathbf{N}^n$





Ordnungstheoretische Eigenschaften $\Leftrightarrow$ Algebraische Eigenschaften

---

Vektorzeit und Vektoruhren ermöglichen eine "operationale Manipulation" der Kausalrelation

Eigenschaften der Vektorzeit

$\tau'(e) := \Sigma_i\, \tau(e)[i]$ hat Eigenschaften der Lamport-Zeit:

1) $e < e'$ ==> $\tau'(e) < \tau'(e')$ (Uhrenbedingung)
2) Umkehrung gilt nicht
3) τ' ist nicht injektiv
4) $\tau'(e)$ ist ein Skalar

Beachte: $\tau'(e)$ ist das "Volumen" des Kegels, die Lamportzeit die längste vorangehende Kette von e.

Realzeitpunkte

Beh.:
$\tau(e) < \tau(e')$ ==> $t(e) < t(e')$

Bew.: $\tau(e) < \tau(e')$ ==> $e < e'$ ==> $t(e) < t(e')$

Bem.: Gilt nicht für die Lamport-Zeit!

Frage: Gibt es kompaktere Zeitstempel als Vektoren der Länge n? (Für die auch die Umkehrung der Uhrenbedingung gelten.)

nicht leicht!

Wäre für die Praxis sehr wichtig (Zeitstempel in Nachrichten können unangenehm lang werden...)

Woraus stammt dieses Zitat?

Und überall hingen, lagen und standen Uhren.
Da gab es auch *Weltzeituhren in Kugelform,*
welche die Zeit für jeden Punkt der Erde anzeigten...
...
...
M. schüttelte lächelnd den Kopf.
"Die Uhr allein würde niemand nützen.
Man muss sie auch lesen können."

Anwendungen der Vektorzeit

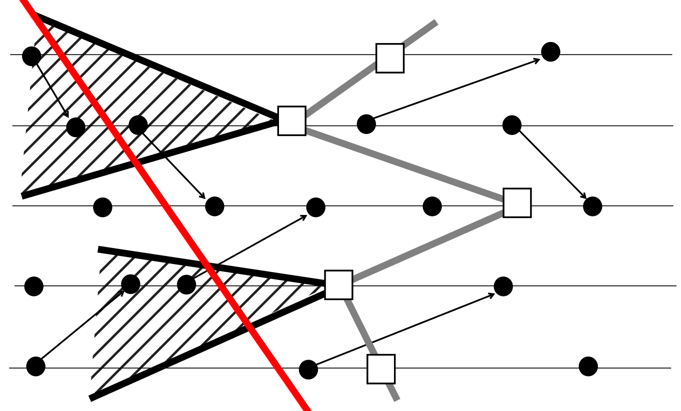
<Momo trifft Professor Hora>:

Und überall hingen, lagen und standen Uhren.
Da gab es auch *Weltzeituhren in Kugelform,*
welche die Zeit für jeden Punkt der Erde anzeigten...
"Vielleicht", meinte Momo,
braucht man dazu eben so eine Uhr."
Meister Hora schüttelte lächelnd den Kopf.
"Die Uhr allein würde niemand nützen.
Man muss sie auch lesen können."

Michael Ende, Momo

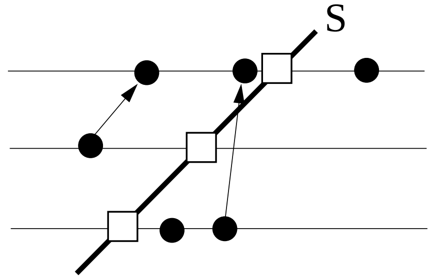
- Debugging !
    - Lokalisierung von Fehlern (“kann [nicht] Ursache sein...)
    - Race conditions; Synchronisationsfehler (kausale Unabhängigkeit)
    - Effizientes Replay
- Leistungsanalyse
    - “Flaschenhals” im Zeitverband; Synchronisationsgrad
    - Kausal unabhängige Ereignisse können parallel ausgeführt werden
- Implementierung konsistenter Schnappschüsse
    - Menge lokaler Schnappschüsse mit paarweise konkurrenten Ereignissen
- Realisierung von kausaltreuen Beobachtern
    - Causal broadcast
    - Causal order

Zeitvektoren der Schnittereignisse



- Betrachte nun die Zeitvektoren der Schnittereignisse s_i eines Schnittes S.
- Liegen die Kegel $\downarrow s_i$ ganz im Schnitt? (Gilt $\downarrow s_i \subseteq S$?)

Nicht immer:



Konsistente Hülle eines Schnittes

Für einen Schnitt S definiere $S^* := \downarrow s_1 \cup \downarrow s_2 \cup_{...} \cup \downarrow s_n$
(Vereinigung aller Schnittereigniskegel)

Beh.: Für jeden Schnitt S ist S*

a) eindeutig ← klar nach Konstruktion

b) ein Schnitt

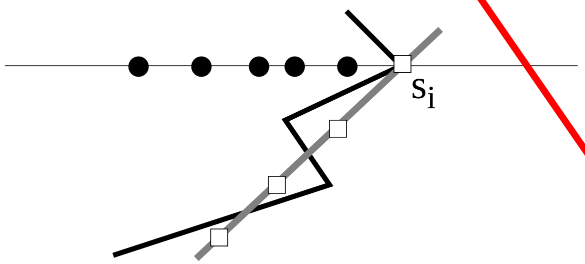
c) konsistent

← - jeder Kegel ist ein kons. Schnitt (Übung)
- kons. Schnitte sind bzgl. Vereinigung abgeschlossen (Verband --> Übung)

Beh.: Es gilt $S \subseteq S^*$ (d.h. S* ist später/gleich S)

Bew.: $S \cap E_i \subseteq \downarrow s_i$

(Alle Ereignisse des Schnittes auf dem i-ten Prozess werden durch $\downarrow s_i$ abgedeckt)



Fragen:

a) Gibt es u.U. einen Schnitt S’ mit $S \subset S' \subset S^*$?

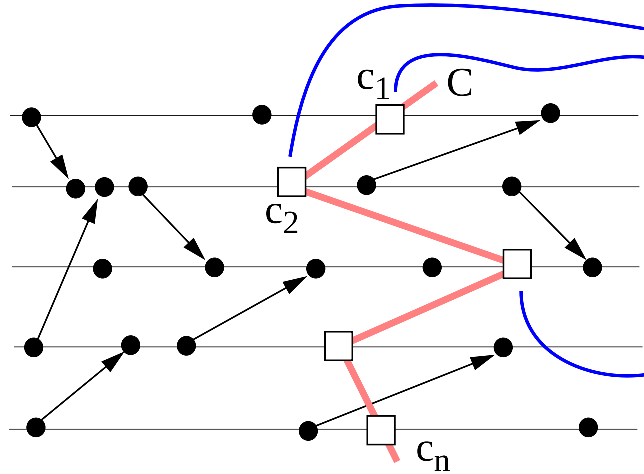
b) ... konsistenten Schnitt...?

Schnittmatrix

- Schnittmatrix \$ eines Schnittes C (mit Schnittereign. c_i):

$\$(C) := (\tau(c_1), \tau(c_2),...,\tau(c_n))$

d.h. Schnittereignisvektoren c_i als Spaltenvektoren



dia → sup

$$\begin{array}{ccccc|c} 3 & 1 & 1 & 0 & 0 & 3 \\ 0 & 4 & 3 & 0 & 0 & 4 \\ 0 & 0 & 5 & 0 & 0 & 5 \\ 0 & 1 & 3 & 4 & 0 & 4 \\ 0 & 0 & 1 & 1 & 3 & 3 \end{array}$$

Frage: Kann man an den Schnittmatrizen etwas über die Schnitte erkennen? (z.B.: ob später, früher; ob konsistent...)

$$\boxed{\text{C konsistent} \Leftrightarrow \text{dia(\$)} = \text{sup(\$)}}$$

Diagonalvektor

Zeilenmaximum

(d.h. Maximum einer Zeile ist das Diagonalelement)

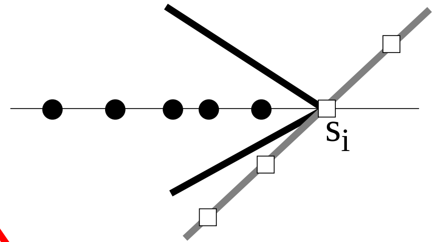
Diagonalvektor und Zeilensupremum

- Def.: dia(\$) ist der Vektor mit $\text{dia}(\$)[i] = s_i[i]$

- Beh.: Diagonalvektor von \$ ist der Schnittvektor von S,
  d.h. $\text{dia}(\$) = \tau(S)$

- Bew.:

$\text{dia}(\$)[i] = \tau(s_i)[i]$
$= |\{e' \in E_i |\ e' \leq s_i\}|$
wieso? $= |\{e' \in E_i |\ e' \leq_l s_i\}|$
$= \tau(S)[i]$



- Def.: sup(\$) ist der Vektor $\sup(\tau(s_1),\ldots,\tau(s_n))$
  (zeilenweise Maximum)

Beispiel:

$$\$ = \begin{matrix} 3 & 1 & 1 & 5 & 0 \\ 0 & 4 & 3 & 0 & 0 \\ 0 & 0 & 5 & 0 & 0 \\ 0 & 1 & 3 & 4 & 0 \\ 0 & 0 & 1 & 1 & 3 \end{matrix} \qquad \text{dia}(\$) = \begin{bmatrix} 3 \\ 4 \\ 5 \\ 4 \\ 3 \end{bmatrix} \qquad \sup(\$) = \begin{bmatrix} 5 \\ 4 \\ 5 \\ 4 \\ 3 \end{bmatrix}$$

Verträglichkeit von τ mit $\cup$ und $\cap$

Beh.: Für zwei Schnitte S, S' einer Berechnung gilt:

$$\boxed{\tau(S \cup S') = \sup(\tau(S), \tau(S'))}$$

Bew.:

$\tau(S \cup S')[i] = |\{e \in E_i |\ e \leq_l s_i \vee e \leq_l s_i'\}|$

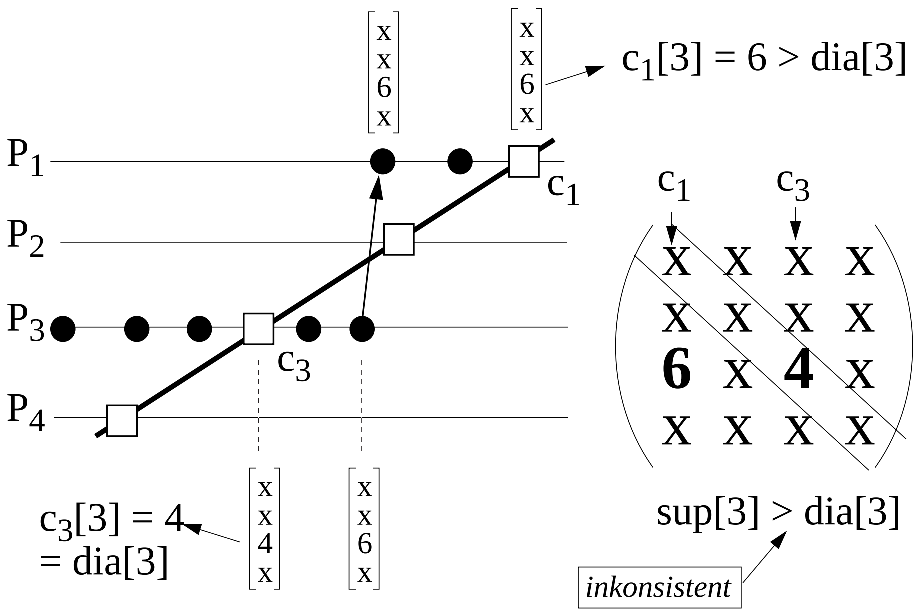
$= \max(|\{e \in E_i |\ e \leq_l s_i\}|, |\{e \in E_i |\ e \leq_l s_i'\}|)$

$= \max(\tau(S)[i], \tau(S')[i])$

Bem.: $\tau(S \cap S') = \inf(\tau(S), \tau(S'))$ analog

Bem.: $\cup$, $\cap$, inf, sup sind assoziativ
--> $\tau(S_1 \cup S_2 \cup \ldots \cup S_k)$ ist sinnvoll.

Das “sup = dia”-Konsistenzkriterium



Ein Prozess (P_1) verschieden von P_3 weiss (bei c_1) etwas über lokale Ereignisse auf P_3, von denen P_3 selbst noch nichts weiss (d.h. die *nach* c_3 geschehen)

$\Longleftrightarrow$

Es gibt einen Pfad von einem Ereignis auf P_3 *nach* c_3 zu einem Ereignis *vor* c_1

$\Longleftrightarrow$

[Generalisierung über alle Indizes i≠j]

Der Schnitt ist inkonsistent

Konsistenzkriterium

Beweis hier ohne "anschauliche" Zeitdiagramme

- Beh.: S inkonsistent ==> $S \neq S^*$
- Bew.: S^* ist stets konsistent (vgl. oben)

- Korollar: S inkonsistent ==> $\tau(S) \neq \tau(S^*)$
  (Versch. Schnitte haben versch. Zeitstempel)

- Beh.: S inkonsistent ==> dia(\$) ≠ sup(\$)
- Bew.: dia(\$) = $\tau(S)$ und

  Verträglichkeit

  $\text{sup}(\$) \overset{\text{Def.}}{=} \text{sup}(\tau(s_1),\ldots,\tau(s_n)) = \tau((\downarrow s_1) \cup \ldots (\downarrow s_n)) \overset{\text{Def.}}{=} \tau(S^*)$.

  Wende nun obiges Korollar an.

- Beh.: S konsistent ==> dia(\$) = sup(\$)
- Bew.: $\downarrow s_i$ liegt ganz in S, d.h. $\downarrow s_i \subseteq S$

  Denn: 1) $x \in \downarrow s_i$ ==> $x \leq s_i$
  2) $y \in S \wedge x \leq y$ ==> $x \in S$
  Wegen $s_i \in S$: $x \in \downarrow s_i$ ==> $x \in S$
  Also $\downarrow s_i \subseteq S$

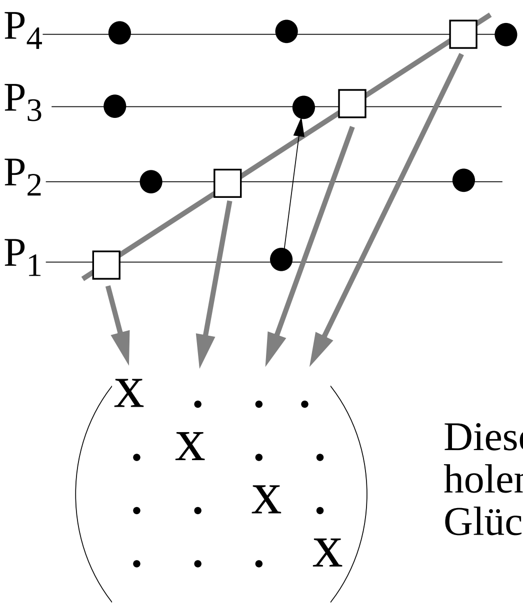
  ==> $S^* \subseteq S$. Umkehrung gilt sowieso:
  ==> $S^* = S$ vgl. vorh. Lemma
  ==> $\tau(S^*) = \tau(S)$ ==> sup(\$) = dia(\$).

Daraus folgt das Konsistenzkriterium:

S konsistent ⇔ dia(\$) = sup(\$)

Implementierung konsistenter Schnappschüsse mit Vektorzeit?

1) Schlechter Ansatz: Alle Prozesse auffordern, ihren lokalen Zustand zu senden und testen, ob konsistent:



Dieses solange wiederholen, bis man einmal Glück hat...

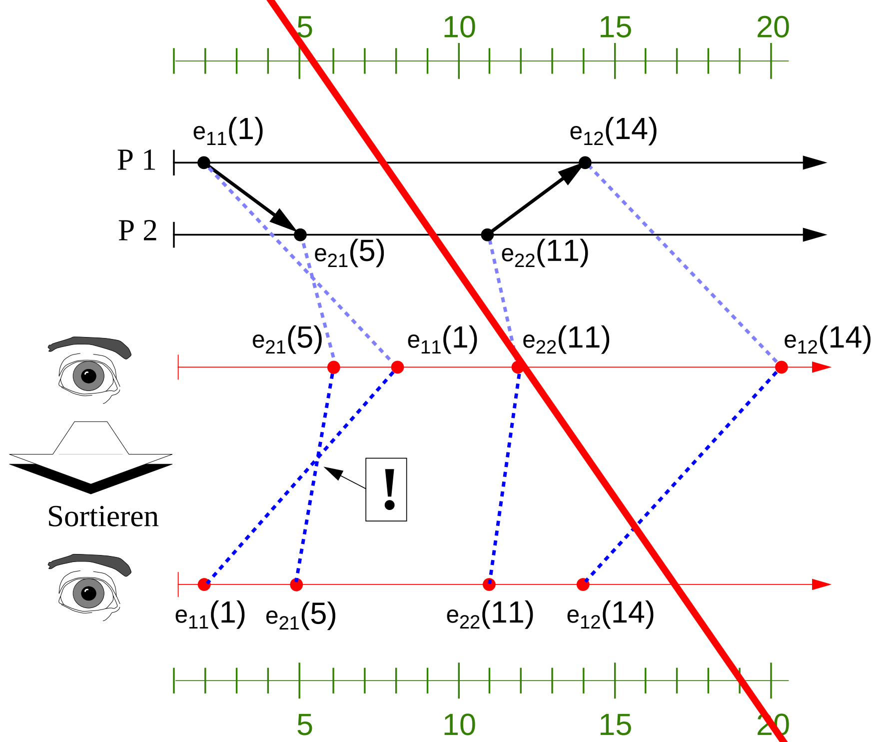
2) Besser: Vektoriellen *Stichzeitpunkt* in der Zukunft festlegen, bei dessen Erreichen / Überschreiten jeder Prozess seinen lokalen Zustand übermittelt

- z.Z.: so definierter Schnitt ist konsistent
- wieso ist der Stichzeitpunkt garantiert noch nicht vorbei?
- wieso wird der Stichzeitpunkt garantiert erreicht / überschritten?
  --> “konzeptioneller Trick”: Erfinde n+1-ten Prozess, dessen Uhr man voll unter Kontrolle hat... --> Optimierung...

Realisierung kausaltreuer Beobachter mit Realzeit

- Grundidee: *Zeit respektiert Kausalität*

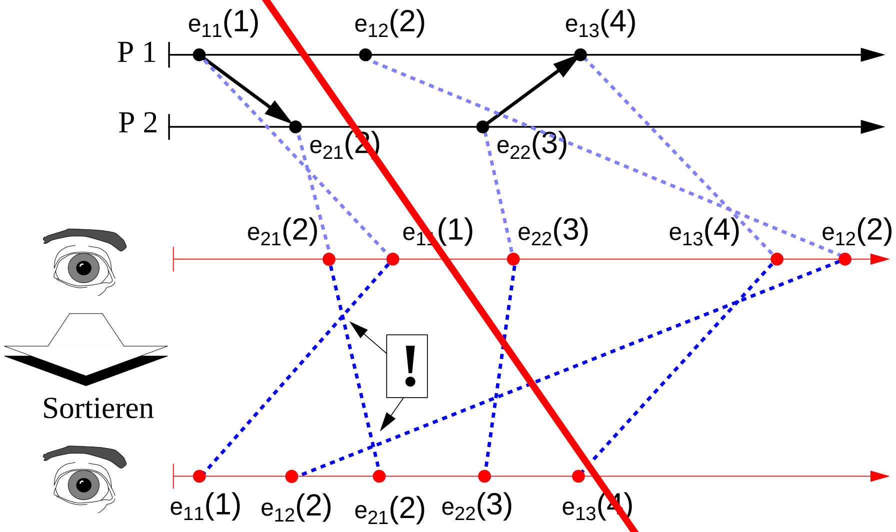
==> *Sortieren* nach globaler Zeit
= “Sortieren" nach Kausalität (--> topologisches Sortieren)



- Beobachter stellt "wahre" Berechnung wieder her
- Problem: (Globale) Realzeit wird benötigt

Realisierung kausaltreuer Beobachter mit Lamport-Zeit

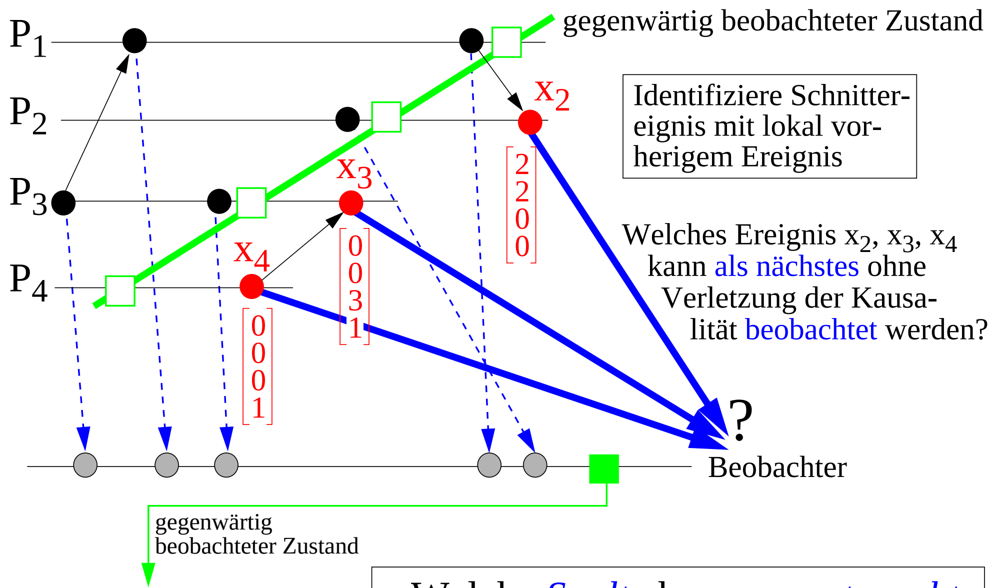
- Grundidee: *Lamport-Zeit respektiert Kausalität* ==> Sortieren liefert eine *lineare Erweiterung* der Kausalrelation



- Problem: Schlecht für *Online-Monitoring* geeignet
  - bevor man ein Ereignis "annimmt", muss man sicher sein, dass kein Ereignis mit einem kleineren Zeitstempel mehr kommt (vgl. e_{13} und e_{12})!
  - FIFO-Kanäle helfen nur beschränkt (--> lange Verzögerungen)
  - auch problematisch, wenn nur eine *Teilmenge* von Ereignissen betrachtet wird

==> Bessere Lösung?

Realisierung kausaltreuer Beobachter



$$\begin{pmatrix} \mathbf{2} & 0 & 0 & 0 \\ 0 & \mathbf{1} & 0 & 0 \\ 1 & 0 & \mathbf{2} & 0 \\ 0 & 0 & 0 & \mathbf{0} \end{pmatrix}$$

- Welche *Spalte* kann *ausgetauscht* werden? (x_2, x_4, aber nicht x_3)
- Beobachter merkt sich dia($); es muss *Zeitstempel ≤ dia($)* sein (ausgenommen die Diagonalkomponente)

- Strategie: dia($) = sup($) --> stets konsistent halten!
- Beobachter benötigt nur den Diagonalvektor, keine Matrix, um momentanen Zustand zu identifizieren

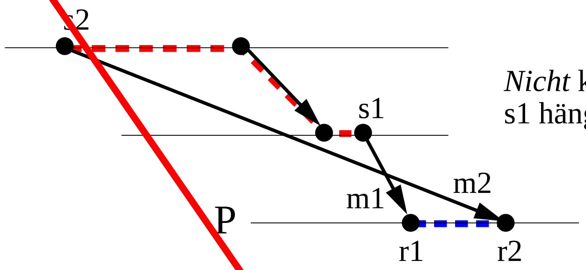
Prinzip: Verwende Vektorzeit um (indirektes) *Wissen* über "kausal frühere" Sendeereignisse zu kodieren:

"Dieses Ereignis hängt von einem anderen ab, das ich eigentlich erhalten haben müsste; also warte ich das andere erst ab..."

Kausal geordneter Nachrichtenempfang

- Empfangene Nachrichten respektieren die Kausalrelation
- Problem ähnlich zur Realisierung kausaltreuer Beobachter

causal order property



Nicht kausal geordnet:
s1 hängt von s2 ab

(Beobachter)

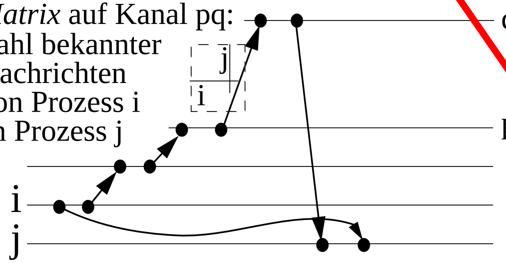
- Eine Nachricht wird nur dann an einen Prozess P ausgeliefert, wenn alle (bzgl. send-Ereignisse) kausal früheren Nachrichten an den gleichen Prozess schon ausgeliefert wurden
  - Formal: r1, r2 auf gleichem Prozess und r1 < r2 ==> s1 < s2 (wobei ri Empfangsereignis zu si ist)

!

- Kein Überholen einer einzelnen Nachricht durch eine Kette von Nachrichten ==> "*Globale FIFO-Eigenschaft*"

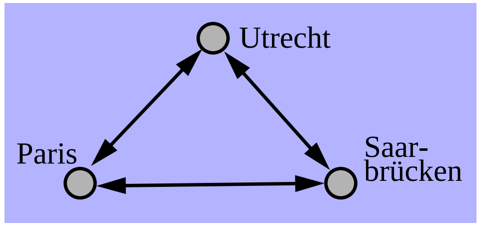
- Realisierung: *Vektor von Vektoren* ("Matrixuhr")

Matrix auf Kanal pq:
Zahl bekannter
Nachrichten
von Prozess i
an Prozess j

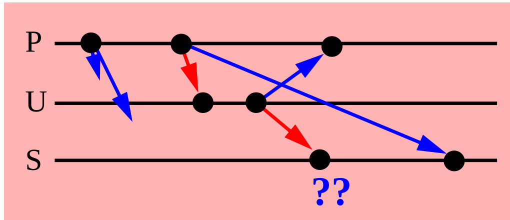


- Jeder Prozess ist ein kausaltreuer Beobachter bzgl. der Nachrichten, die er empfängt
- Schema kausaltreuer Beobachter mit n Vektoren der Länge n verwenden

Kausaltreue Nachrichtenordnung



- Diskussion von drei Personen per "broadcast"
- Verwirrungen, da indirekte Kommunikation gelegentlich schneller als die direkte ist

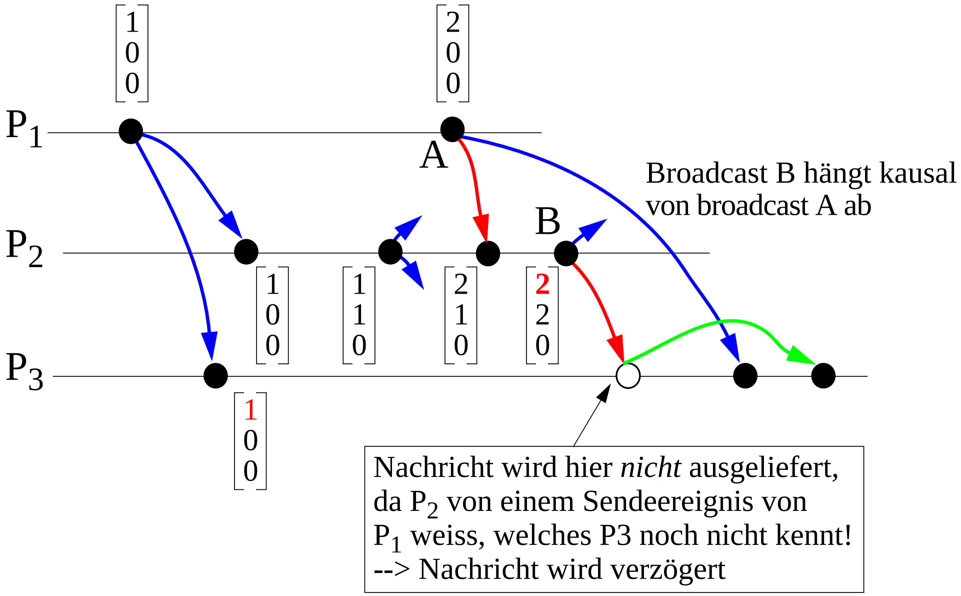


- Lösung: Jeder Teilnehmer soll alle relevanten Ereignisse kausaltreu wahrnehmen
- Dazu Vektorzeit verwenden (Vektor genügt, da jede Spalte der Matrix identische Elemente hat!)

```
Date: Fri, 3 Nov 89 16:46:55 +0100
From: Bernadette Charron <charron@...fr>
To: mattern

DATE : (101,5,5)
Bonjour a tous,
Me revoila...

Au fait, avec vos estampilles
vectorielles, les processus "lents"
sont tout de suite detectes...On ne
peut plus dormir en silence, sans etre
repere, a moins d'accuser le reseau.

Comme j'ai BEAUCOUP reflechi, je
rajoute 100 actions internes pour ma
composante.
```

!

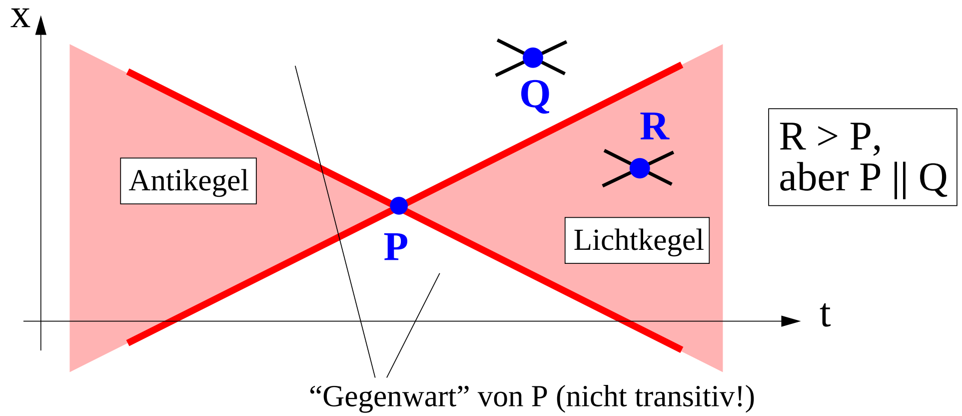
Implementierung von kausalem Broadcast



- *Prinzip*: Verwende die Vektorzeit, um (indirektes) Wissen über "kausal frühere" Ereignisse zu kodieren
  - *"dieses Ereignis hängt von einem anderen ab, das ich eigentlich erhalten haben müsste; also warte ich das andere erst ab..."*
- Nur Broadcast-Ereignisse sind relevante Ereignisse
- Vektoren sind Spezialfälle der Matrixuhren
  - alle Elemente einer Spalte identisch --> Reduktion zu einem Vektor
- Verallgemeinerung des Sequenznummerverfahrens zur Implementierung von FIFO bei Nicht-FIFO-Kanälen

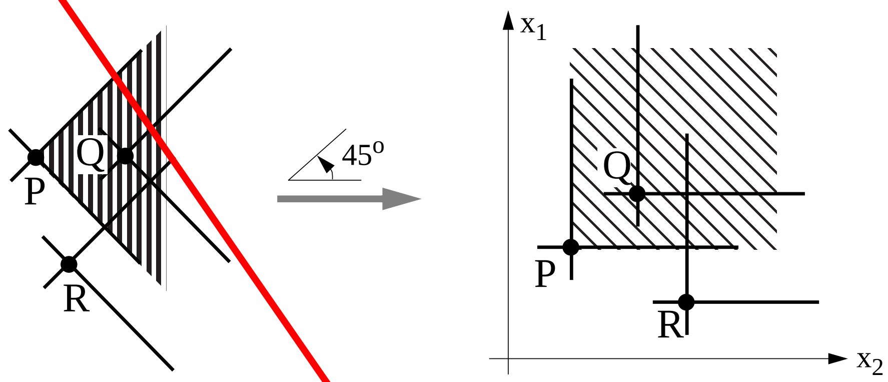
Denkübung:
Vektoren sind relativ aufwendig: Geht kausaler Broadcast, causal order, kausaltreue Beobachtung auch mit weniger aufwendigen Datenstrukturen?

Vektorzeit und Minkowski-Raumzeit



Raumzeit	*Vektorzeit*
Halbordnung	Halbordnung
2-dimensionale Kegel bilden Verband (bzgl. Schnitt)	Zeitvektoren bilden Verband (sup)
Lorentz-Transformation lässt Lichtkegel invariant	Gummiband-Transformation lässt Kausalrelation invariant
Raumzeitkoordinaten ermöglichen Test, ob potentiell kausal abhängig: Mit $u = (x_1, t_1)$, $v = (x_2, t_2)$ prüfe $c^2(t_2-t_1)^2 - (x_2-x_1)^2 >= 0$	Zeitvektoren ermöglichen einfachen Test, ob potentiell kausal abhängig: (prüfe, ob in allen Komponenten kleiner)

Lichtkegel- und Vektorzeit-Ordnung



90^o Lichtkegel (maximale Geschwindigkeit normiert zu “1 Raumeinheit per Zeiteinheit”, d.h.“Lichtjahr / Jahr”)

$X=(x_1,x_2),\ Y=(y_1,y_2)$

Vektoren = Koordinaten der Punkte

- Lichtkegel von X ganz im Lichtkegel von Y enthalten (linkes Bild) $\Leftrightarrow\ x_1 < y_1 \wedge x_2 < y_2$ (rechtes Bild) $\Leftrightarrow (x_1,x_2) < (y_1,y_2) \Leftrightarrow X < Y$.

==> 2-dimensionale Kegel ≈ 2-dimensionale Würfel

==> Zumindest bei 2 Dimensionen haben Raumzeit und Vektorzeit i.w. die gleiche Struktur!

- “später”
- potentielle Kausalität
- Verbandsstruktur

Resümee: Konzepte der Vorlesung

- Prinzipielle Phänomene und Begriffe herausarbeiten
  - Kausalität, Konsistenz, verteilte Berechnung, safety und liveness,...
- Geeignete Modelle und Abstraktionen entwickeln
  - z.B. Zeitdiagramme, Atommodell, Zustandsgitter, Gummibandtransform.
- Problemlösungs-, Analyse- und Verifikationstechniken
  - z.B. Beweis über Invarianten
- Techniken, Einsichten, Zusammenhänge,...
  - Komplexitätsanalyse
  - Transformationen zwischen Problemklassen
  - Problemverständnis von einem höheren Standpunkt