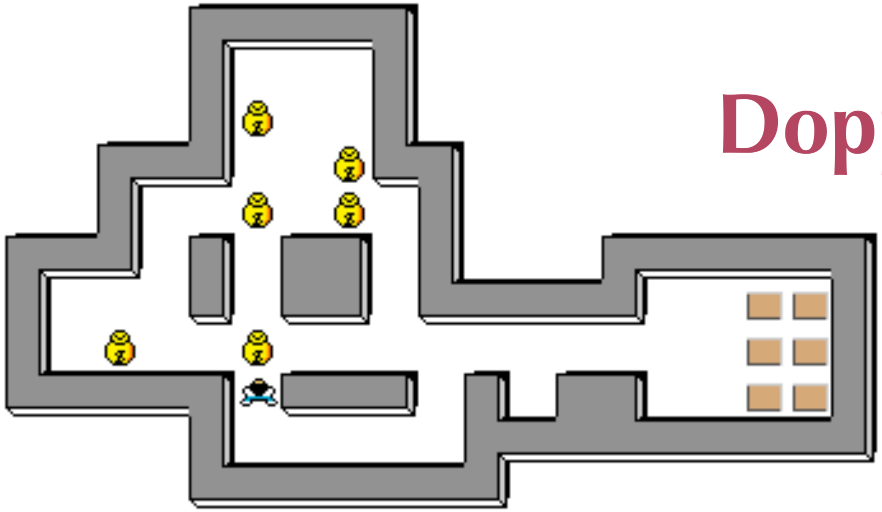
Die Folie zum Tag: Niko-Ban

http://freechristmasgames.com/sokoban/sokoban.htm



Doppeltes Säckeschieben

In Situation S (Bild) führe zweimal nacheinander *SchiebN* aus.
(Strategisch dumm, sollte aber gehen.)

$$Ban(F(5,8),S) \wedge Sack(F(5,7),S) \wedge Frei(F(5,6),S) \quad ✔$$

$$\Rightarrow Poss(SchiebN,S)$$

$$\Rightarrow Ban(F(5,7),Res(SchiebN,S)) \wedge \neg Ban(F(5,8),Res(SchiebN,S)) \wedge Sack(F(5,6), Res(SchiebN,S)) \wedge \neg Sack(F(5,7), Res(SchiebN,S)) \wedge Frei(F(5,7), Res(SchiebN,S)) \wedge \neg Frei(F(5,6), Res(SchiebN,S))$$

Jetzt wende *SchiebeN* das zweite Mal an, also in *Res*(*SchiebN*,*S*)!

$$Ban(F(5,7),Res(SchiebN,S)) \wedge Sack(F(5,6),Res(SchiebN,S)) \quad ✔$$

$$Frei(F(5,5),Res(SchiebN,S)) \quad ?$$

Nein! Über $Frei(F(5,5),Res(SchiebN,S))$ ist nichts bekannt!!

Das *Frame Problem*

Das Frame Problem ist das Problem, bei der Modellierung von Wandel auch all das modellieren zu müssen, das sich *nicht* ändert.

Frame
=
„Rahmen“

- Man könnte Frame-Axiome angeben, die explizit listen, was sich alles nicht ändert, z.B.:
  $Ban(F(h,v),s) \wedge Frei(F(g,w),s) \wedge g \neq h-2 \Rightarrow Frei(F(g,w),Res(SchiebN,s))$
  - ↪ wären aber *viel* zu viele Frame-Axiome!
    (nämlich $O(AF)$ Axiome bei A Aktionen und F Fluenten)
  - ↪ Repräsentationales Frame-Problem
- Wie leite ich die Wandel über eine Folge von Aktionen ab in Abhängigkeit der veränderten, *nicht* der fixen Fakten?
  - ↪ Inferenzielles Frame-Problem

Exkurs: Frame-Probleme und -Unprobleme

- In den 1970er bis 1980er Jahren wurde das Frame-Problem und seine vermeintlichen Auswirkungen heiß diskutiert
- Es diente gar philosophisch denkenden Kollegen als ein Argument dafür „*What Computers Can't Do*"
- Dabei war schon damals klar:
  - Es ist ein Phänomen, das auftaucht bei einer (rückschauend: naiven) Formulierung von Wandel in PL 1. Stufe
  - In anderen Formulierungen (z.B. über Mengen von Merkmalen) taucht es nicht auf
  - Solche anderen Formulierungen wurden auch bereits in den 1970er und 1980er Jahren verwendet (z.B. im Planen, s.Kap.5)

Eine Lösung des repräsentationalen Frame-P.

Modelliere nicht aktionszentriert, sondern fluent-zentriert!
Statt …

- Nachbedingungs-Axiome der Form
  $Poss(a,s) \Rightarrow$ Effekte der Aktion a

… besser:

- Nachfolgezustands-Axiome der Form:
  $Poss(a,s) \Rightarrow$
  [Fluent ist wahr im Zustand $Res(a,s)$
  $\Leftrightarrow$ a hat es wahr gemacht
  v es war schon vorher wahr
  und a hat es nicht geändert]

Beispiel: Nachfolgezustands-Axiome in Soko-Ban

- **Nachfolgezustands-Axiom für *Ban*:**

$$
\begin{aligned}
&Poss(a,s) \Rightarrow \\
&\quad [Ban(F(h,v),Res(a,s)) \\
&\quad \Leftrightarrow \quad [(a = GehN \vee a = SchiebN) \wedge Ban(F(h,v+1),s)] \\
&\quad\quad \vee [(a = GehS \vee a = SchiebS) \wedge Ban(F(h,v-1),s)] \\
&\quad\quad \vee \ldots \text{ analog für GehO, SchiebO, GehW, SchiebW } \ldots \\
&\quad\quad \vee [Ban(F(h,v),s) \wedge a \neq GehN \wedge a \neq SchiebN \wedge a \neq GehS \wedge a \neq \ldots]]
\end{aligned}
$$

- **Nachfolgezustands-Axiom für *Frei*:**

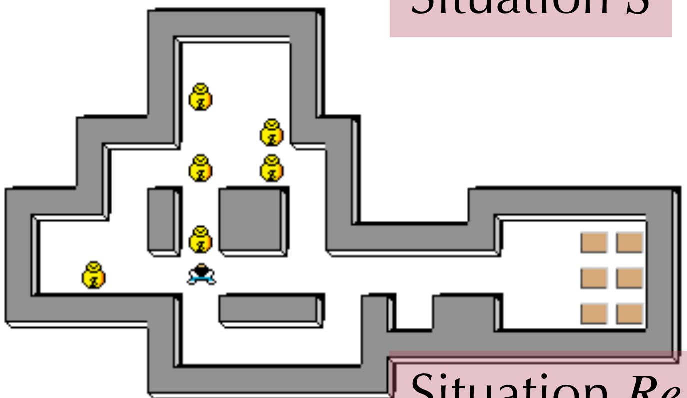
$$
\begin{aligned}
&Poss(a,s) \Rightarrow \\
&\quad [Frei(F(h,v),Res(a,s)) \\
&\quad \Leftrightarrow \quad [a = SchiebN \wedge Ban(F(h,v+1),s)] \\
&\quad\quad \vee [a = SchiebS \wedge Ban(F(h,v-1),s)] \\
&\quad\quad \vee \ldots \text{ analog für SchiebO, SchiebW } \ldots \\
&\quad\quad \vee [Frei(F(h,v),s) \wedge (a \neq SchiebN \vee \neg Ban(F(h,v+1),s)) \\
&\quad\quad\quad\quad \wedge (a \neq SchiebS \vee \neg Ban(F(h,v-1),s)) \wedge \ldots]]
\end{aligned}
$$

- **Nachfolgezustands-Axiom für *Sack*:** analog

Eigenschaften der Nachfolgezustands-Axiome

- 1 Axiom je Fluent, also F Stück (gegenüber $O(AF)$ Frame-Axiomen)
- $O(AE)$ Literale (∅ AE/F pro Axiom) für A Aktionen und max. E Effekte pro Aktion (wobei normalerweise $E<<F$)

Situation S



Situation $Res(SchiebN,S)$

Beispiel: Doppeltes Säckeschieben

Wie vorher gilt

- $Ban(F(5,8),S) \wedge Sack(F(5,7),S) \wedge Frei(F(5,6),S)$

$\Rightarrow$ $Poss(SchiebN,S)$

- $Ban(F(5,7),Res(SchiebN,S))$ ableitbar aus NFZ-Axiom Ban: $[Ban(F(h,v+1),\ldots) \rightarrow Ban(F(h,v),\ldots)]$
- $Sack(F(5,6),Res(SchiebN,S))$ ableitbar aus NFZ-Axiom $Sack$ [analog]

! $Frei(F(5,5),Res(SchiebN,S))$ **ableitbar** aus NFZ-Axiom $Frei$ **wegen**
$Frei(F(5,5),Res(SchiebN,S)) \wedge$
$\neg Ban(F(5,6), Res(SchiebN,S)))$

Zur Lösung des inferenziellen Frame-P.

Es gilt $Sack(F(2,7), S_0)$. Ist der Sack in bislang 123 Aktionen unberührt und soll nun verschoben werden, braucht es eine aufwändige Ableitung des Fakts

$$Sack\big(F(2,7), Res(a_{123}, Res(a_{122}, Res(\ldots Res(a_1,S_0)\ldots)\big)$$

Keine gute Lösung dafür im „reinen" Situationskalkül!

Möglichkeiten (hier nicht vertieft):

- „Buchführung", welche (Grund-)Fakten sich wann ändern (Inkrementelle Fortschreibung der Start-Situationsbeschreibung)
- Übergang in andere Kalküle
  - Event-Kalkül
  - Fluent-Kalkül

„Dominoeffekte“: N-*Power*-Soko-Ban

Betrachte folgende Variation von Soko-Ban:

- Der Ban kann nicht nur einen Sack schieben, sondern N, sofern sie in Feldern in Schieberichtung lückenlos voreinander stehen.

Wie würde sich die Modellierung der Aktionen ändern?

➥Außer dem „direkten“ Effekt des Sackschiebens gäbe es „kontextabhängige“ „Dominoeffekte“ bezüglich 1…N-1 Säcken, die von dem tatsächlich geschobenen stehen.

➥Modellierbar mit je N-1 Fallunterscheidungen (Implikationen) in den NF-Zustands-Axiomen von *Frei* und *Sack*

➥Aber: Wie zuvor Repräsentations- und Inferenzproblem

Das *ramification* Problem

Das *ramification* Problem ist das Problem, bei der Modellierung von Wandel auch Folgeeffekte von „primären“ Veränderungen effizient zu behandeln (repräsentational, inferenziell).

Ramification
=
„Verzweigung“,
„Dominoeffekt“

- Wiederum keine gute Lösung im Situationskalkül
- Grundidee einer Lösung:
  - Spezifiziere wie gehabt „primäre“ Veränderungen
  - Leite Folgeeffekte innerhalb einer Situation ab, indem all die Fakten revidiert werden, die mit den primären Veränderungen inkonsistent sind

Das *qualification*-Problem (verbreitetste Interpretation)

- **Wie können wir sicherstellen, dass die Vorbedingungs-Axiome alle Vorbedingungen (*qualifications*) aufzählen?**
- Klassisches Beispiel: Die Kartoffel im Auspuff:
  Jede Modellierung des Autofahrens, die *nicht* erwähnt, dass *keine* Kartoffel im Auspuff steckt (und keine Möhre, kein Kohlrabi, kein …), ist unvollständig
- Stimmt — aber kann/soll man das immer mit modellieren?

So what?

↪ Alle endlichen Modelle von irgendwas sind unvollständig. Das ist eine Tatsache, aber kein (technisches) Problem!

↪ Die interessante Frage ist: Wie kann ein wissensbasiertes System mit solchen Ausnahmen clever umgehen?

Normalfallschließen

Technisches Problem klassischer Logik beim Frame-Problem, beim qualification-Problem, bei vielen „natürlichen“ Schlüssen:

- Im Normalfall sollen gewisse Konsequenzen gelten;
- unter bekannten Ausnahmen sollen andere Konsequenzen gelten;
- solange nicht bekannt ist, dass ein Ausnahmefall vorliegt, soll der Normalfall angenommen werden (was zurückgenommen werden kann, wenn später Ausnahme-Information nachkommt)

Beispiele

- Mangels Information beim Soko-Ban, dass eine Aktion einen Sack verschoben hat, geh davon aus, er steht am selben Ort
- Mangels Information beim Auto-Anlassen, dass der Auspuff verstopft ist, geh davon aus, er ist okay.

Normalfälle in klassischer Logik?

Beispiel: „Normalerweise können Vögel fliegen"

- $\forall x.\ Vogel(x) \Rightarrow Kann\_fliegen(x)$
- modelliert nicht das „normalerweise", insbesondere folgt aus
  $\forall x.\ Pinguin(x) \Rightarrow Vogel(x)$ dass
  $\forall x.\ Pinguin(x) \Rightarrow Kann\_fliegen(x)$!
- Zweiter Versuch:
  $\forall x.\ Vogel(x) \wedge \neg Unnormal(x) \Rightarrow Kann\_fliegen(x)$ und
  $\forall x.\ Pinguin(x) \vee Strauß(x) \vee Brathuhn(x) \Rightarrow Unnormal(x)$.
- … funktioniert, aber erfordert ausdrückliche Aufzählung aller Ausnahmen (vgl. Frame-Axiome im Situationskalkül!)
- … und bei Ableitungen Nachweis erforderlich, dass jeweils keine Unnormalität vorliegt!

Vogelflug in PROLOG

```
kannfliegen(X) :- vogel(X), not(unnormal(X)).
vogel(tweety).
vogel(hansi).
vogel(X) :- pinguin(X).
unnormal(X) :- pinguin(X).
pinguin(ping).

?- kannfliegen(X).
X = tweety ;
X = hansi ;
No

pinguin(tweety).

?- kannfliegen(X).
X = hansi ;
No
```

Wie in Kapitel 2 besprochen:
Prolog verwendet *negation by finite failure*, was hier das Gesuchte leistet! Aber:

- die logische Semantik davon ist nicht ohne Weiteres klar
- das funktioniert mit Literalen in Hornklausellogik – was ist mit allgemeineren Sätzen?

Arten Nichtmonotonen Schließens

- Default-Schließen

  Normalerweise können Vögel Fliegen
  Tweety ist ein Vogel
  ------------------ „solange keine gegenteilige Information vorliegt“
  Tweety kann fliegen

- Autoepistemisches Schließen

  Ich kenne alle meine Geschwister
  Ich weiß (derzeit) nichts darüber, dass Peter mein Bruder ist
  ----
  Peter ist nicht mein Bruder

  Schließen auf Grund des eigenen Wissens und Nichtwissens

- Schließen auf Basis von Kommunikations-Konventionen

  Wenn im Fahrplan kein Zug um 17:14 steht, dann fährt auch keiner

- Inkonsistenztolerantes Schließen

Reiters Default-Logik

Erziele Normalfall-Schlüsse über nichtklassische Inferenzregeln, die Teil der Modellierung, nicht Teil des Kalküls sind!

Die Regeln („Defaults“) sind von der Form $\alpha : \beta_1,\ldots,\beta_n / \gamma$ (für Formeln α, β_i, γ) mit der Interpretation:

Wenn α gilt, dann leite γ ab, sofern nichts gegen ein β_i spricht

α ist die Prämisse, γ die Konklusion und β_i die Rechtfertigungen

Beispiel:

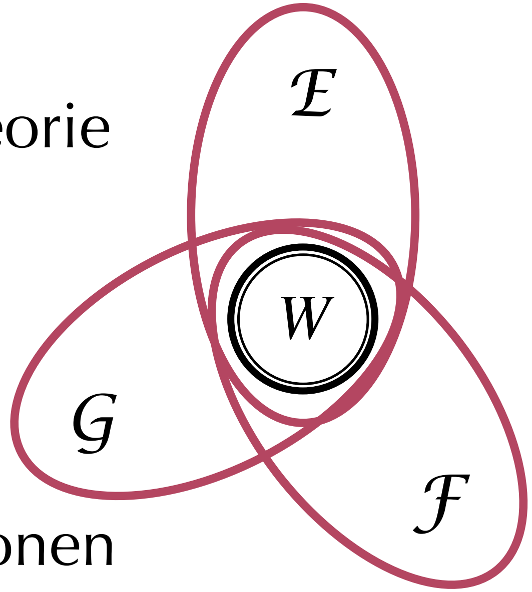
$$Vogel(x) : Kann\_fliegen(x) \,/\, Kann\_fliegen(x)$$

Eine Default-Theorie (D,W) besteht aus einer Menge von Formeln (PL 1.Stufe) W und aus einer Menge von Defaults D

Extensionen einer Default-Theorie

Mit den Defaults als neuen (anwendungsspezifisch formulierten) Ableitungsregeln muss ein neuer Ableitbarkeitsbegriff her!

Eine Extension $\mathcal{E}$ (Menge aller ableitbaren *prädikatenlogischen* Formeln einer Default-Theorie (D,W) soll:



- die Ausgangs-Formelmenge W enthalten
- abgeschlossen sein bezüglich klassischer Ableitbarkeit
- eine maximale Menge von Default-Konklusionen enthalten, d.h. wenn $\alpha : \beta_1,\ldots,\beta_n / \gamma$ ein Default ist, $\alpha\in\mathcal{E}$ und $\neg\beta_i\notin\mathcal{E}$, dann soll auch $\gamma\in\mathcal{E}$ sein
- genau nur die genannten Formeln enthalten.

Definition Extension

Sei (D,W) eine Default-Theorie, $\mathcal{E}$ Menge prädikatenlogischer Formeln. Th(.) bezeichnet klassische Ableitbarkeit aus einer Formelmenge in *PL 1. Stufe*. Seien $\mathcal{E}_0$, $\mathcal{E}_1$ definiert wie folgt:

$$\mathcal{E}_0 = W$$

$$\mathcal{E}_{i+1} = \mathrm{Th}(\mathcal{E}_i) \cup \{\gamma \mid \alpha:\beta_1,\ldots,\beta_n/\gamma \in D,\ \alpha \in \mathcal{E}_i,\ \neg\beta_i \notin \mathcal{E}\}$$

$\mathcal{E}$ ist Extension von (D,W) gdw. $\mathcal{E} = \cup \mathcal{E}_i$

Beispiele:

$D_1 = \{Vogel(x) : Kann\_fliegen(x) \,/\, Kann\_fliegen(x)\}$,
$W_1 = \{Vogel(Tweety)\}$
$\mathcal{E}_1 = \mathrm{Th}(W_1 \cup \{Kann\_fliegen(Tweety)\})$

$D_2 = \{Vogel(x) : Kann\_fliegen(x) \,/\, Kann\_fliegen(x)\}$,
$W_2 = \{Vogel(Tweety), Pinguin(Tweety), \forall x.\ Pinguin(x) \Rightarrow \neg Kann\_fliegen(x)\}$
$\mathcal{E}_2 = \mathrm{Th}(W_2)$

Mehr zu Extensionen

- Extensionen haben die Form Th($W \cup$ Konklusionen(D')) für $D' \subseteq D$
- Extensionen sind konsistent
  (außer W ist inkonsistent oder Default-Konklusionen sind es)
- Extensionen konstruieren: Finde alle maximalen $D' \subseteq D$,
  bei denen [$W \cup$ Konklusionen(D')] konsistent ist
  und die Prämissen enthalten sind
  (↪ Teilmengen von Extensionen sind nie Extensionen, da nicht maximal!)

Beispiel:

$D = \{A{:}C \,/\, C,\ {:}B \,/\, B,\ {:}A \,/\, A\}$, $\qquad W = \{A \Rightarrow \neg B\}$

$\{A,B\}$ inkonsistent unter W, also entfällt auch $\{A,B,C\}$, bleiben als Kandidaten

- Th($W \cup$ Konklusionen($\{A,C\}$)): ✓
- Th($W \cup$ Konklusionen($\{B,C\}$)): – (keine C-Prämisse enthalten!)
- Th($W \cup$ Konklusionen($\{B\}$)): ✓
- alle anderen potenziellen Extensionen ($\emptyset$, $\{A\}$, $\{C\}$) nicht maximal

Widersprechende Defaults

Beispiel:

$D = \{ \quad \mathit{Republican}(x) : \neg \mathit{Pacifist}(x) \,/\, \neg \mathit{Pacifist}(x)\ ,$
$\qquad\quad \mathit{Quaker}(x) : \mathit{Pacifist}(x) \,/\, \mathit{Pacifist}(x)\ \}$

$W = \{ \quad \mathit{Republican}(\mathit{Nixon}), \mathit{Quaker}(\mathit{Nixon})\ \}$

erzeugt zwei Extensionen: $\mathcal{E} = \mathrm{Th}(W \cup \{\mathit{Pacifist}(\mathit{Nixon})\})$
$\mathcal{F} = \mathrm{Th}(W \cup \{\neg \mathit{Pacifist}(\mathit{Nixon})\})$

Demnach stellen sich zwei Fragen:

- Was sollen wir nun glauben bzgl. Nixons Pazifismus?
  - „skeptische" Variante: α ableitbar aus (D,W) gdw. α ist in allen Extensionen enthalten
  - „mutige" Variante: α ableitbar aus (D,W) gdw. es gibt eine Extension, in der α enthalten ist
- Haben Default-Theorien Extensionen? Wie charakterisiert?