Kapitel 6: Logikorientierte Anfragesprachen

6.1 Überblick: Dualismus der Logik

Syntax (beweistheoretische Sicht):

Formeln F1, ..., Fn
mit Aussagenvariablen,
Prädikat-/Funktionssymbolen

Ableitungsregeln
(Beweis-/Deduktionsregeln):
F1, ..., Fn $\vdash$ G

Semantik (modelltheoretische Sicht):

Interpretation ψ in
(mathematischer) Struktur S

Gültigkeit in S unter ψ
(G ist in S wahr):
$\psi \models G$

idealerweise:

wenn G ableitbar, dann G wahr (Korrektheit)
wenn G wahr, dann G ableitbar (Vollständigkeit)

Aussagenlogik: Syntax

Definition:

Eine atomare ***Formel*** der Aussagenlogik ist eine Aussagenkonstante True oder False oder eine Aussagevariable der Form Ai (i=1, 2, ...).

Die Menge der Formeln der Aussagenlogik ist:

(i) Jede atomare Formel ist eine Formel.

(ii) Wenn F, G Formeln sind, dann auch (F), $\neg F$, $F \wedge G$, $F \vee G$ sowie $F \Rightarrow G$ (kurz für $\neg F \vee G$) und $F \Leftrightarrow G$ (kurz für $(F \wedge G) \vee (\neg F \wedge \neg G)$).

Notation und Präzedenzen:

Statt $\neg F$ schreibt man auch $\overline{F}$.

$\neg$ bindet stärker als $\wedge$ und $\vee$, und $\wedge$ und $\vee$ binden stärker als $\Rightarrow$, $\Leftrightarrow$.

Aussagenlogik: Semantik

Definition:

Sei F eine Formel der Aussagenlogik mit atomarer Formelmenge D.

Eine ***passende Struktur*** S zu F ist eine Menge P von Aussagen, so daß jede Variable in D einer dieser Aussagen zugeordnet werden kann.

Eine ***Interpretation*** von F ist eine Abbildung ψ: D $\rightarrow$ P, für die gilt ψ(True)=1, ψ(False)=0, ψ(Ai)=1 (0), falls ψ(Ai) in S wahr (falsch) ist.

ψ wird auf beliebige Formeln F, G, usw. über D fortgesetzt:

(i) $\psi((F)) = \psi(F)$

(ii) $\psi(F \wedge G)= 1$ falls $\psi(F)= \psi(G)=1$, 0 sonst

(iii) $\psi(F \vee G)= 1$ falls $\psi(F)= 1$ oder $\psi(G)=1$, 0 sonst

(iv) $\psi(\neg F)=1$ falls $\psi(F)=0$, 0 sonst.

Aussagenlogik: Modelle und Folgerungen

Definition:

Sei F eine Formel und ψ eine Interpretation von F.
Wenn $\psi(F)=1$ ist, heißt ψ ***Modell*** von F ($\psi \models F$ oder $\models_\psi F$).
F heißt ***erfüllbar***, falls F mindestens ein Modell hat, sonst ***unerfüllbar***.
F heißt ***(allgemein-)gültig*** oder ***Tautologie***, falls jede Interpretation in einer zu F passenden Struktur ein Modell von F ist.

Satz: F ist Tautologie genau dann, wenn $\neg F$ unerfüllbar ist

Definition:

G heißt ***Folgerung*** von F1, ..., Fk (F1,...,Fk $\models$ G), wenn für jedes ψ gilt: falls ψ Modell von F1, ..., Fk ist, dann ist ψ auch Modell von G.
F und G heißen ***äquivalent*** ($F \equiv G$), falls für jedes ψ gilt: $\psi(F)= \psi(G)$.

Satz: G ist Folgerung von F1, ..., Fk g.d.w. $(F1 \wedge F2 \wedge \ldots \wedge Fk) \Rightarrow G$ eine Tautologie ist. F und G sind äquivalent g.d.w. F Folgerung von G ist und umgekehrt.

Aussagenlogik: Beispiele

1) F1: $T \Rightarrow \neg P$, F2: $P \wedge \neg T$ bzw. F: $(T \Rightarrow \neg P) \wedge (P \wedge \neg T)$
$\psi(T)$ = „7 ist durch 2 teilbar“, $\psi(P)$ = „7 ist eine Primzahl“

2) F: $(S \Rightarrow \neg R) \wedge (\neg R \Rightarrow S) \wedge (A \Rightarrow \neg S) \wedge (A)$
$\psi(S)$ = „die Sonne scheint“, $\psi(R)$ = „es regnet“ (wahr),
$\psi(A)$ = „es ist April“

3) a) $\neg(\neg F) \Leftrightarrow F$
b) $(X \wedge Y) \vee (\neg X \wedge Z)$
c) $F \wedge \neg F$

Aussagenlogik: Deduktionskalkül

Ein einfacher Deduktionskalkül:

(i)		$\vdash F \Rightarrow (F \Rightarrow F)$
(ii)		$\vdash (F \Rightarrow (G \Rightarrow H)) \Rightarrow ((F \Rightarrow G) \Rightarrow (F \Rightarrow H))$
(iii)		$\vdash (\neg F \Rightarrow \neg G) \Rightarrow (G \Rightarrow F)$
(iv)	$F \Leftrightarrow G, H$	$\vdash H[F/G]$
(v)	(v) $F \Rightarrow G, F$	$\vdash G$

Definition:

Sei H eine endl. Formelmenge. F heißt aus H ***ableitbar*** ($H \vdash F$), wenn es eine endl. Sequenz F0, F1, ..., Fn von Formeln gibt mit Fn=F, so daß für alle Fi gilt: Fi ist Element von H, Fi ist eine der Formeln (i) bis (iii) oder Fi ist aus F0, ..., F(i-1) mittels (iv) oder (v) abgeleitet. Für $H=\emptyset$ heißt F ***Theorem*** des Deduktionskalküls.

Satz: Der einfache Deduktionskalkül ist korrekt und vollständig, d.h.:

$H \vdash F$ genau dann, wenn $H \models F$

(und $\vdash F$ genau dann, wenn $\models F$)

Aussagenlogik: alternative Kalküle bzw. Äquivalenzregeln

$\neg\neg F \Leftrightarrow F$	(Doppelnegation)
$F \wedge F \Leftrightarrow F$	(Idempotenz)
$F \vee F \Leftrightarrow F$	
$F \wedge G \Leftrightarrow G \wedge F$	(Kommutativität)
$F \vee G \Leftrightarrow G \vee F$	
$F \wedge (G \wedge H) \Leftrightarrow (F \wedge G) \wedge H$	(Assoziativität)
$F \vee (G \vee H) \Leftrightarrow (F \vee G) \vee H$	
$F \wedge (F \vee G) \Leftrightarrow F$	(Absorption)
$F \vee (F \wedge G) \Leftrightarrow F$	
$F \wedge (G \vee H) \Leftrightarrow (F \wedge G) \vee (F \wedge H)$	(Distributivität)
$F \vee (G \wedge H) \Leftrightarrow (F \vee G) \wedge (F \vee H)$	
$\neg (F \wedge G) \Leftrightarrow (\neg F \vee \neg G)$	(Gesetz von de Morgan)
$\neg (F \vee G) \Leftrightarrow (\neg F \wedge \neg G)$	
$F \vee 1 \Leftrightarrow 1$	(Tautologieregeln)
$F \wedge 1 \Leftrightarrow F$	
$F \vee 0 \Leftrightarrow F$	(Unerfüllbarkeitsregeln)
$F \wedge 0 \Leftrightarrow 0$	
$F \vee (\neg F) \Leftrightarrow 1$	(Tertium non datur)
$F \wedge (\neg F) \Leftrightarrow 0$	(Kontradiktion)
$((F \Leftrightarrow G) \wedge H) \Leftrightarrow ((F \Leftrightarrow G) \wedge H[F/G])$	(Substitution)

Beispiel für aussagenlogische Deduktion

„Worin besteht das Geheimnis Ihres Lebens?“:
Wenn ich kein Bier zu einer Mahlzeit trinke, dann habe ich immer Fisch.
Wenn ich Fisch und Bier zur selben Mahlzeit habe, verzichte ich auf Eiscreme.
Wenn ich Eiscreme habe oder Bier meide, dann rühre ich Fisch nicht an.

Modellierung:
Formel G: $(\neg B \Rightarrow F) \wedge ((F \wedge B) \Rightarrow \neg E) \wedge ((E \vee \neg B) \Rightarrow \neg F)$

Vereinfachung, sprich Deduktion (mit Unterstreichung als Negation):

$$
\begin{array}{lll}
G \Leftrightarrow & (B \vee F) \wedge (\underline{F \wedge B} \vee \underline{E}) \wedge (\underline{(E \vee \neg B)} \vee \underline{F}) & \text{Def. Implikation} \\
\Leftrightarrow & (B \vee F) \wedge (\underline{F} \vee \underline{B} \vee \underline{E}) \wedge ((\underline{E} \wedge B) \vee \underline{F}) & \text{de Morgan} \\
\Leftrightarrow & (B \vee F) \wedge (\underline{F} \vee \underline{B} \vee \underline{E}) \wedge (\underline{E} \vee \underline{F}) \wedge (B \vee \underline{F}) & \text{Distributivität} \\
\Leftrightarrow & (\, (B \vee F) \wedge (B \vee \underline{F}) \,) \quad \wedge \quad (\, (\underline{F} \vee \underline{B} \vee \underline{E}) \wedge (\underline{E} \vee \underline{F}) \,) & \text{Komm., Assoz.} \\
\Leftrightarrow & (\, (B \vee (F \wedge \underline{F}) \,) \quad \wedge \quad (\, (\underline{F} \vee \underline{E}) \vee (\underline{B} \wedge 0) \,) & \text{Distrib., Tautologie} \\
\Leftrightarrow & (\, (B \vee 0 \,) \quad \wedge \quad (\, (\underline{F} \vee \underline{E}) \vee 0 \,) & \text{Kontradikt., Unerf.} \\
\Leftrightarrow & B \wedge \; (\underline{F} \vee \underline{E}) & \text{Unerfüllbarkeit}
\end{array}
$$

Fazit: Trinke zu jeder Mahlzeit Bier, und iss niemals Fisch mit Eiscreme!

Prädikatenlogik: Syntax

Definition:

Gegeben seien Variablen xi, Prädikatsymbole Pi der Stelligkeit ki und Funktionssymbole fi der Stelligkeit li. ***Terme*** sind:

(i) Jede Variable ist ein Term

(ii) Wenn t1, ..., tk Terme sind und fi ein k-stelliges Funktionssymbol ist, dann ist auch fi(t1, ..., tk) ein Term.

Eine ***atomare Formel*** hat die Form Pi(t1, ..., tk) mit einem k-stelligen Prädikatsymbol Pi und Termen t1, ..., tk.

Formeln der Prädikatenlogik 1. Ordnung sind:

(i) Jede atomare Formel ist eine Formel.

(ii) Für Formeln F und G sind auch $\neg$F, F$\wedge$G, F$\vee$G und (F) Formeln.

(iii) Für eine Variable xi und eine Formel F sind auch $\forall$xi (F) und $\exists$xi (F) Formeln (mit ***gebundener*** Variable xi).

Notation und Präzedenzen:
Statt $\forall$x $\forall$y $\forall$z (F) schreibt man auch $\forall$x,y,z (F).
Junktoren binden stärker als Quantoren.
Alle gebundenen Variablen sind paarweise verschieden.

Prädikatenlogik: Semantik (1)

Definition:

Geg. sei eine Menge von Funktions- und Prädikatsymbolen mit entspr. Stelligkeiten. Eine passende ***Struktur*** ist ein Tripel S = (U, fun, pred) mit

- einem ***Universum*** (einer ***Trägermenge***) U von Individuen,
- einer Menge fun von ***Funktionen*** fi: $U \times ... \times U \rightarrow U$ der Stelligkeit li, so daß für jedes li-stellige Symbol eine entspr. Funktion existiert
- einer Menge pred von ***Prädikaten*** Pi: $U \times ... \times U \rightarrow \{0,1\}$ der Stelligkeit ki, so daß für jedes ki-stellige Symbol ein entspr. Prädikat existiert.

Eine ***Interpretation*** ψ einer Formel F in einer passenden Struktur S = (U, fun, pred) ist eine Abbildung, die U festlegt und jedem Funktions- bzw. Prädikatsymbol in F eine Funktion aus fun bzw. ein Prädikat aus pred zuordnet.

Prädikatenlogik: Semantik (2)

ψ wird auf beliebige Formeln F, G, usw. und Terme fortgesetzt:

(i) $\psi((F)) = \psi(F)$

(ii) $\psi(F \wedge G) = 1$ falls $\psi(F) = \psi(G) = 1$, 0 sonst

(iii) $\psi(F \vee G) = 1$ falls $\psi(F) = 1$ oder $\psi(G) = 1$, 0 sonst

(iv) $\psi(\neg F) = 1$ falls $\psi(F) = 0$, 0 sonst

(v) $\psi(f(t1, ..., tk)) = \psi(f)(\psi(t1), ..., \psi(tk))$
für ein Funktionssymbol f und Terme t1, ..., tk,

(vi) $\psi(x) = a$ mit einem beliebigen $a \in U$ für eine (freie) Variable x

(vii) $\psi(P(t1, ..., tk)) = \psi(P)(\psi(t1), ..., \psi(tk))$
für ein Prädikatsymbol P und Terme t1, ..., tk,

(viii) $\psi(\forall x\ (F)) = 1$ falls $\psi(F[x/a]) = 1$ für alle $a \in U$

(ix) $\psi(\exists x\ (F)) = 1$ falls $\psi(F[x/a]) = 1$ für ein beliebiges $a \in U$.

Prädikatenlogik: Beispiele (1)

1) F: $\forall q\ \exists p\ \forall x,y\ (\text{greater}(p,q) \wedge$
$((\text{greater}(x,1) \wedge \text{greater}(y,1)) \Rightarrow (\neg\ \text{equal}(\text{mult}(x,y),p)))\ )$

Struktur $S = (N_0, \{+, \cdot\}, \{>, =\})$.

Interpretation ψ:

add $\mapsto$ +, mult $\mapsto$ ·, greater $\mapsto$ >, equal $\mapsto$ =,

$q \mapsto 1$, $p \mapsto 2$, $x \mapsto 3$, $y \mapsto 4$

2) F: K (1, Lauer, Merzig) $\wedge$ K(2, Schneider, Homburg) $\wedge$
P(1, Papier) $\wedge$ B(7, 16, 1, 1, 100)

Struktur: Datenbank mit dem Schema

Kunden (KNr, Name, Stadt),
Produkte (PNr, Bez),
Bestellungen (Monat, Tag, KNr, PNr, Menge).

über dem Universum $N_0 \cup \Sigma^*$

Interpretation ψ (F) = 1, falls es die vier Tupel in der Datenbank gibt,
0 sonst.

Erweiterungen der Prädikatenlogik

Mehrsortige Logik mit mehreren Trägermengen (Sorten) und einer Signatur für die Prädikate und Funktionen.

Beispiel: U= {U1, U2} mit $U1=R^n$, U2=R und

Addition: $U1 \times U1 \rightarrow U1$, Multiplikation: $U1 \times U2 \rightarrow U1$,

Skalarprodukt: $U1 \times U1 \rightarrow U2$,

Vektorprodukt: $U1 \times U1 \rightarrow U1$, Nullvektor: $\rightarrow U1$,

Prädikaten Orthogonal: $U1 \times U1 \rightarrow \{0,1\}$ usw.

Prädikate als Argumente von Prädikaten oder Funktionen,
Quantifizierung von Prädikaten oder Funktionen

→ Prädikatenlogik höherer Ordnung

Beispiel:

$$\forall x,y \; (H(x,y) \Leftrightarrow \forall P \; ((R(x,y) \Rightarrow P(x,y)) \wedge \forall u,w,z \; (P(u,w) \wedge P(w,z) \Rightarrow P(u,z))) \Rightarrow (H(x,y) \Rightarrow P(x,y)) \;)$$

Prädikatenlogik: Modelle und Folgerungen

Definition:

Sei F eine Formel und ψ eine Interpretation von F.

Wenn $\psi(F)=1$ ist, heißt ψ ***Modell*** von F ($\psi \models F$ oder $\models_\psi F$).

F heißt ***erfüllbar***, falls F mindestens ein Modell hat, sonst ***unerfüllbar***.

F heißt ***(allgemein-)gültig*** oder ***Tautologie***, falls jede Interpretation in einer zu F passenden Struktur ein Modell von F ist.

Satz: F ist Tautologie genau dann, wenn $\neg$F unerfüllbar ist

Definition:

G heißt ***Folgerung*** von F1, ..., Fk ($F1,\ldots,Fk \models G$), wenn für jedes ψ gilt: falls ψ Modell von F1, ..., Fk ist, dann ist ψ auch Modell von G.

F und G heißen ***äquivalent*** ($F \equiv G$), falls für jedes ψ gilt: $\psi(F)= \psi(G)$.

Satz: G ist Folgerung von F1, ..., Fk g.d.w. $(F1 \wedge F2 \wedge \ldots \wedge Fk) \Rightarrow G$ eine Tautologie ist. F und G sind äquivalent g.d.w. F Folgerung von G ist und umgekehrt.

Prädikatenlogik: Beispiele (2)

Formeln F1, ..., F5 bzw. Formel F= F1 $\wedge$ ... $\wedge$ F5:

$F1 = \forall x,y,z\ (\ G(f(f(x,y),z), f(x,f(y,z)))\ )$

$F2 = \forall x\ (\ G(f(x,e),x)\ )$

$F3 = \forall x,y\ (\ G(x, y) \Leftrightarrow G(y, x)\ )$

$F4 = \forall x\ \forall y\ \forall z\ (\ G(x,y) \wedge G(y,z) \Rightarrow G(x,z)\ )$

$F5 = \forall x\ \exists y\ (\ G(f(x,y), e)\ )$

Prädikatenlogik: Deduktionskalküle

alle aussagenlogischen Äquivalenzen

Wenn $F \Leftrightarrow G$ und F als Teilformel in H vorkommt, dann ist $H \Leftrightarrow H[F/G]$

$\neg \forall(F) \Leftrightarrow \exists x\, (\neg F)$

$\neg \exists\, (F) \Leftrightarrow \forall x\, (\neg F)$

$(\forall x\, (F)) \wedge G \Leftrightarrow \forall x\, (F \wedge G)$ falls x in G nicht frei vorkommt

$(\forall x\, (F)) \vee G \Leftrightarrow \forall x\, (F \vee G)$ falls x in G nicht frei vorkommt

$(\exists x\, (F)) \wedge G \Leftrightarrow \exists x\, (F \wedge G)$ falls x in G nicht frei vorkommt

$(\exists x\, (F)) \vee G \Leftrightarrow \exists x\, (F \vee G)$ falls x in G nicht frei vorkommt

$(\forall x\, (F)) \wedge (\forall y\, (G)) \Leftrightarrow \forall x\, (F \wedge G[y/x])$, falls x in G nicht frei vorkommt

$(\exists x\, (F)) \vee (\exists y\, (G)) \Leftrightarrow \exists x\, (F \vee G[y/x])$, falls x in G nicht frei vorkommt

$\forall x\, (\forall y\, (F)) \Leftrightarrow \forall y\, (\forall x\, (F))$

$\exists x\, (\exists y\, (F)) \Leftrightarrow \exists y\, (\exists x\, (F))$

$\forall x\, (F) \Rightarrow F[x/a]$ für alle Konstanten a

$(\, (\forall x\, (F)) \wedge (F \Rightarrow G)\,) \quad \Rightarrow \quad (\forall x\, (G))$

Vollständige und korrekte Kalküle: Resolutionskalkül, Tableaukalkül

Beispiel für prädikatenlogische Deduktion (1)

1. *i) Informatikstudenten können programmieren.*
2. *ii) Studenten mit guten Mathematiknoten studieren Informatik.*
3. *iii) Studenten, die mit einem Informatiker verwandt sind,*
4. *haben gute Mathematiknoten.*
5. *iv) Alle saarländischen Studenten sind mit Heinz Becker verwandt.*
6. *v) Die Verwandtschaftsbeziehung ist symmetrisch.*
7. *vi) Heinz Becker hat Informatik studiert.*

Universum: Menge aller Studenten

Prädikate:

KannProgrammieren P(x)

StudiertInformatik I(x)

HatGuteMathenoten M(x)

SindVerwandt V(x,y)

IstSaarländer S(x)

Funktionen und Konstanten:

HeinzBecker hb()

Beispiel für prädikatenlogische Deduktion (2)

Formeln:

1. i) $\forall x\ (\ I(x) \Rightarrow P(x)\ ) \wedge$
2. ii) $\forall x\ (\ M(x) \Rightarrow I(x)\ ) \wedge$
3. iii) $\forall x\ \ \forall y\ (\ (V(x,y) \wedge I(y)) \Rightarrow M(x)\ ) \wedge$
4. iv) $\forall x\ (\ S(x) \Rightarrow V(x,hb)\ ) \wedge$
5. v) $\forall x\ \ \forall y\ (\ (V(x,y) \Leftrightarrow V(y,x)\ ) \wedge$
6. vi) $I(hb)$

Vereinfachung, sprich Deduktion:

(iii) $\wedge$ (iv) $\wedge$ (vi):

$$\forall x\ \ \forall y\ (\ I(hb) \wedge (S(x) \Rightarrow V(x,hb)) \wedge (V(x,y) \wedge I(y)) \Rightarrow M(x))\ )$$

$$\vdash\ \ \forall x\ \ (\ I(hb) \wedge (S(x) \Rightarrow V(x,hb)) \wedge (V(x,hb) \wedge I(hb)) \Rightarrow M(x))\ \ )$$

$$\vdash\ \ \forall x\ \ (\ S(x) \Rightarrow M(x)\ )\ \ (*)$$

(*) $\wedge$ (ii) $\wedge$ (i):

$$\forall x\ \ (\ (S(x) \Rightarrow M(x))\ \ \wedge\ \ (M(x) \Rightarrow I(x))\ \ \wedge\ \ (I(x) \Rightarrow P(x))\ )$$

$$\vdash\ \ \forall x\ \ (\ S(x) \Rightarrow P(x)\ )$$

Fazit: alle Saarländer können programmieren!

6.2 Domain-Relationenkalkül (DRK)

Anfragen sind prädikatenlogische Mengenspezifikationen der Form

$\{<x1, ..., xn> \mid F(x1, ..., xn)\}$

wobei F ein prädikatenlogischer Ausdruck über einer Menge von ***Domain-Variablen*** ist mit x1, ..., xn als einzigen freien Variablen.

Die Menge der zulässigen prädikatenlogischen Ausdrücke F ist:

1) Für Variablen x1, ..., xn und eine Relation R mit n Attributen ist $<x1,...,xn> \in R$ (auch R(x1, ..., xn)) ein zulässiger Ausdruck.
2) Für Variablen x, y, Konstanten c und Vergleichsoperationen $\theta \in \{=, \neq, <, >, \leq, \geq\}$ sind $x\ \theta\ y$ und $x\ \theta\ c$ zulässige Ausdrücke.
3) Falls F1 und F2 zulässige Ausdrücke sind, dann sind auch $F1 \wedge F2$, $F1 \vee F2$, $\neg F1$ und (F1) zulässig.
4) Falls F ein zulässiger Ausdruck mit einer freien Variable x ist, dann sind auch $\exists\ x: F(x)$ und $\forall\ x: F(x)$ zulässige Ausdrücke.
5) Falls F ein zulässiger Ausdruck ist, dann ist auch (F) zulässig.
6) Nur die durch 1) bis 5) erzeugten Ausdrücke sind zulässig.

Semantik des DRK

Eine ***Datenbank*** über einem Schema mit Relationen $R1(A11, ..., A1n_1), ..., Rm(Am1, ..., Amn_m)$ ist eine Formelmenge H, die für jede Relation Ri ein Prädikatsymbol Ri verwendet und für jedes Tupel $\langle a1, ..., an_i\rangle \in val(Ri)$ eine atomare Formel $Ri(a1, ..., an_i)$ enthält, bzw. ein Modell von H.

Das Ergebnis einer ***Anfrage*** $\{\langle x1, ..., xn\rangle \mid F(x1, ..., xn)\}$ über der Datenbank H ist:
$\{\langle a1, a2, ... an\rangle \mid a1, a2, ..., an \in U \text{ und } H \models F[x1/a1, x2/a2, ..., xn/an]\}$.

Sonderfall:
wenn F keine freien Variablen enthält, ist das Ergebnis 1 wenn $H \models F$ und 0 sonst.

Beispiele für DRK-Anfragen (1)

1) Finden Sie (die Namen) alle(r) Kunden mit negativem Saldo.

$\rightarrow \{<k,n,st,sa,r> \mid <k,n,st,sa,r> \in \text{Kunden} \wedge sa < 0.0\}$ bzw.

$\rightarrow \{<n> \mid \exists\, k,st,sa,r\colon <k,n,st,sa,r> \in \text{Kunden} \wedge sa < 0.0\}$

2) Finden Sie die Namen aller Kunden, die eine unbezahlte Bestellung haben, die vor Anfang Oktober erfolgte.

$$\rightarrow \{<n> \mid \exists\, k,st,sa,r\colon <\underline{k},n,st,sa,r> \in \text{Kunden} \wedge$$
$$\exists\, b,m,t,p,me,su,stat\colon <b,m,t,\underline{k},p,me,su,stat> \in \text{Best.} \wedge$$
$$m < 10 \;\wedge\; stat \neq \text{'bezahlt'}\}$$

3) Namen der Homburger Kunden, die seit Sept. ein Prod. aus Homburg geliefert bekommen haben, jeweils mit der Bez. des Produkts.

$$\rightarrow \{<n,bez> \mid \exists\, k,st,sa,r\colon <\underline{k},n,st,sa,r> \in \text{Kunden} \wedge$$
$$\exists\, p,g,pr,l,v\colon <\underline{p},bez,g,pr,l,v> \in \text{Produkte} \wedge$$
$$\exists\, b,m,t,me,su,stat\colon <b,m,t,\underline{k},\underline{p},me,su,stat> \in \text{Best.} \wedge$$
$$st=\text{'Homburg'} \wedge mw9 \wedge l=\text{'Homburg'}\}$$

Beispiele für DRK-Anfragen (2)

4) Kunden, von denen mindestens eine Bestellung registriert ist.

$$\rightarrow \{<k,n,st,sa,r> \mid <\underline{k},n,st,sa,r> \in \text{Kunden} \land \exists\ b,m,t,p,me,su,stat: <b,m,t,\underline{k},p,me,su,stat> \in \text{Best.}\}$$

5) Kunden, von denen keine Bestellung registriert ist.

$$\rightarrow \{<k,n,st,sa,r> \mid <k,n,st,sa,r> \in \text{Kunden} \land \forall\ b,m,t,k',p,me,su,stat: <b,m,t,k',p,me,su,stat> \in \text{Best.} \Rightarrow k \neq k'\}$$

6) Kunden, die alle überhaupt lieferbaren Produkte bestellt haben.

$$\rightarrow \{<k,n,st,sa,r> \mid <\underline{k},n,st,sa,r> \in \text{Kunden} \land \forall\ p: (\exists\ bez,g,pr,l,v: <\underline{p},bez,g,pr,l,v> \in \text{Produkte}) \Rightarrow (\exists\ b,m,t,me,su,stat: <b,m,t,\underline{k},\underline{p},me,su,stat> \in \text{Bestellungen})\ \}$$

Anfrage-GUI Query-by-Example

Finden Sie die Namen der Homburger Kunden, die seit Anfang Sept. ein Produkt aus Homburg oder Saarbrücken geliefert bekommen haben, dessen Preis über 100 DM liegt, jeweils mit der Bez. des Produkts.

Kunden

KNr	Name	Stadt	Saldo	Rabatt
_55	.print	Homburg		

Bestellungen

BestNr	Monat	Tag	KNr	PNr	Menge	Summe	Status
	>= 9		_55	_33			

Produkte

PNr	Bez	Gewicht	Preis	Lagerort	Vorrat
_33	.print		> 100	Homburg Saarbrücken	

6.3 Tupel-Relationenkalkül (TRK)

Anfragen sind prädikatenlogische Mengenspezifikation der Form

$\{t \mid F(t)\}$ bzw. $\{<t1.A1, ..., tn.An> \mid F(t1, ..., tn)\}$,

wobei F eine prädikatenlogische Formel über einer Menge von ***Tupelvariablen*** ist mit t bzw. t1, ..., tn als einzigen freien Variablen und A1, ..., An Attributnamen sind.

Die Menge der zulässigen prädikatenlogischen Ausdrücke F ist:

1) Für eine Variable r und eine Relation R ist $r \in R$ (auch R(r)) ein zulässiger Ausdruck.
2) Für Var. r, s, Attr. A, B mit dom(A)=dom(B), Konst. $c \in$ dom(A) und Vergleichsoperationen $\theta \in \{=, \neq, <, >, \leq, \geq\}$ sind r.A θ s.B und r.A θ c zulässige Ausdrücke.
3) Falls F1 und F2 zulässige Ausdrücke sind, dann sind auch $F1 \wedge F2$, $F1 \vee F2$, $\neg F1$ und (F1) zulässig.
4) Falls F ein zulässiger Ausdruck mit einer freien Tupelvariable r ist, dann sind auch $\exists$ r: F(r) und $\forall$ r: F(r) zulässige Ausdrücke.
5) Falls F ein zulässiger Ausdruck ist, dann ist auch (F) zulässig.
6) Nur die aufgrund von 1) bis 5) erzeugten Ausdrücke sind zulässig.

Semantik des TRK

Übersetzung in den DRK:

In einer Anfrage der Form {t | F(t)} oder {<t.B1, ..., t.Bn> | F(t)} wird

- jeder Term der Form r.Ai mit einer Tupelvariablen r durch eine Domain-Variable ra_i ersetzt und
- jeder Term der Form $r \in R$ über einer Relation mit Schema R(A1, ..., Am) durch einen Term $R(ra_1, ..., ra_m)$ mit Domain-Variablen $ra_1, ..., ra_m$.

Beispiele für TRK-Anfragen (1)

1) Finden Sie (die Namen) alle(r) Kunden mit negativem Saldo.

   $\rightarrow \{t \mid t \in \text{Kunden} \wedge t.\text{Saldo} < 0.0\}$ bzw.

   $\{t.\text{Name} \mid t \in \text{Kunden} \wedge t.\text{Saldo} < 0.0\}$

2) Finden Sie die Namen aller Kunden, die eine unbezahlte Bestellung haben, die vor Anfang Oktober erfolgte.

   $\rightarrow \{t.\text{Name} \mid t \in \text{Kunden} \wedge \exists\, b: b \in \text{Bestellungen} \wedge$
   $t.\text{KNr}=b.\text{KNr} \wedge b.\text{Monat} < 10 \wedge b.\text{Status} \neq \text{'bezahlt'}\}$

3) Namen der Homburger Kunden, die seit Sept. ein Prod. aus Homburg geliefert bekommen haben, jeweils mit der Bez. des Produkts.

   $\rightarrow \{k.\text{Name}, p.\text{Bez} \mid k \in \text{Kunden} \wedge p \in \text{Produkte} \wedge$
   $\exists\, b: b \in \text{Bestellungen} \wedge$
   $k.\text{Stadt}=\text{'Homburg'} \wedge b.\text{Monat} \geq 9 \wedge p.\text{Lagerort'}=\text{'Homburg'} \wedge$
   $k.\text{KNr}=b.\text{KNr} \wedge b.\text{PNr}=p.\text{PNr}\}$

Beispiele für TRK-Anfragen (2)

4) Kunden, von denen mindestens eine Bestellung registriert ist.

$\rightarrow \{t \mid t \in \text{Kunden} \wedge \exists\, b: b \in \text{Bestellungen} \wedge b.\text{KNr}=t.\text{KNr}\}$

5) Kunden, von denen keine Bestellung registriert ist.

$\rightarrow \{t \mid t \in \text{Kunden} \wedge \neg(\exists\, b: b \in \text{Bestellungen} \wedge b.\text{KNr}=t.\text{KNr})\}$

oder

$\{t \mid t \in \text{Kunden} \wedge \forall\, b: b \in \text{Bestellungen} \Rightarrow b.\text{KNr} \neq t.\text{KNr}\}$

6) Kunden, die alle überhaupt lieferbaren Produkte bestellt haben.

$$\rightarrow \{t \mid t \in \text{Kunden} \wedge \forall\, p: p \in \text{Produkte} \Rightarrow (\exists\, b: b \in \text{Bestellungen} \wedge b.\text{PNr}=p.\text{PNr} \wedge b.\text{KNr}=t.\text{KNr}) \}$$

Domain-unabhängige Anfragen

Problem: Anfragen liefern u.U. unendliche Resultatmengen.

Beispiel: $\{t \mid \neg(t \in R)\}$ für eine endliche Relation R.

Lösungsidee:
Eine Anfrage des TRK heißt ***domain-unabhängig*** g.d.w. für jede mögliche Ausprägung der Datenbank die Resultatmenge von der Datenbank, nicht aber von den zugrundeliegenden Domains abhängt.

Wahl des Universums U bei der Interpretation einer TRK-Formel F:

- ***unbeschränkte Interpretation*:**
  U besteht aus der Vereinigung aller Domains der Datenbank
- ***beschränkte Interpretation*:**
  U besteht aus dem ***aktiven Domains*** der Datenbank und der Formel F

Definition:
F heißt ***domain-unabhängig*** g.d.w. ihre unbeschränkte Interpretation für jeden Domain Dom, der den aktiven Domain ADom umfasst, mit ihrer beschränkten Interpretation übereinstimmt.

Domain-unabhängige Anfragen: Beispiele

Sei R eine Relation mit sch(R)={A}, dom(A)={1,2}, val(R)={<1>}.

Der Ausdruck $\{t \mid \neg (t \in R)\}$ hat

als unbeschränkte Interpretation den Wert {<2>} und

als beschränkte Interpretation den Wert Ø.

Der Ausdruck $\{t \mid \exists\ x : \neg (x=t) \wedge x.A=1\}$ hat

als unbeschränkte Interpretation den Wert {<2>} und

als beschränkte Interpretation den Wert Ø.

Der Ausdruck $\{t \mid t \in R \wedge \forall\ x : x=t\}$ hat

als unbeschränkte Interpretation den Wert Ø und

als beschränkte Interpretation den Wert {<1>}.

Sichere Anfragen: Idee

F(x) mit Var. x heißt ***beschränkt***, wenn für jede Interpretation von F gilt:
- wenn F(x) wahr ist,
  dann liegt die Interpretation von x in ADom von F.

F(x) mit Var. x heißt ***unbeschränkt***, wenn für jede Interpret. von F gilt:
- wenn die Interpretation von x nicht in ADom von F liegt,
  muß F(x) wahr sein.

Durch syntaktische Einschränkungen von Formeln können
- **freie Variablen** **beschränkt** werden
  (so daß in {t | F(t)} F(t) nur für t aus ADom erfüllbar ist),
- **durch Existenzquantoren gebundene Variablen** **beschränkt** werden
  (so daß in $\exists$ x : G(x) G(x) nur für x aus ADom erfüllbar ist) und
- **durch Allquantoren gebundene Variablen** **unbeschränkt** werden
  (so daß in $\forall$ x : G(x) G(x) für Interpret. von x, die nicht im ADom liegen, immer wahr ist).

Sichere Anfragen: Syntaktische Einschränkung

Definition:

Sei F(t) eine Formel des TRK mit freier Variable t. Zu jeder Var. in F läßt sich ein Schema aus F und dem Datenbankschema ableiten.

Die Beschränktheit und die Unbeschränktheit der Attr. von Var. in F sind:

1) In $t \in R$ sind alle Attribute von t beschränkt.
2) In t.A=c mit einer Konstanten c ist t.A beschränkt.
3) In $r.A=s.B \wedge F$ ist r.A beschränkt, wenn s.B in F beschränkt ist, und s.B beschränkt, wenn r.A in F beschränkt ist..

4a) In $F \wedge G$ ist t.A beschränkt g.d.w. t.A in F oder in G beschränkt ist.
4b) In $F \wedge G$ ist t.A unbeschränkt g.d.w. t.A in F und in G unbeschr. ist.
5a) In $F \vee G$ ist t.A beschränkt g.d.w. t.A in F und in G beschränkt ist.
5b) In $F \vee G$ ist t.A unbeschränkt g.d.w. t.A in F oder in G unbeschr. ist.
6a) In $\neg F$ ist t.A beschränkt g.d.w. t.A in F unbeschränkt ist.
6b) In $\neg F$ ist t.A unbeschränkt g.d.w. t.A in F beschränkt ist.
7a) In $\exists$ x: F(x) ist t.A. beschränkt g.d.w. t.A in F beschränkt ist.
7b) In $\exists$ x: F(x) ist t.A. unbeschränkt g.d.w. t.A in F unbeschränkt ist.
8a) In $\forall$ x: F(x) ist t.A beschränkt g.d.w. t.A in F beschränkt ist.
8b) In $\forall$ x: F(x) ist t.A unbeschränkt g.d.w. t.A in F unbeschränkt ist.

Sichere und unsichere Anfragen: Beispiele

Sei R eine Relation mit sch(R)={A,B}, val(R)={<2,2>}.

$\{t \mid \neg(t \in R) \vee t.A=2 \vee t.B=2\}$

$\{t \mid \neg(t \wedge R) \wedge t.A=2\}$

$\{t \mid \neg(t \wedge R) \wedge t.A=2 \wedge t.B=2\}$

$\{t \mid \exists\, s: (s \in R \wedge s.A=t.A) \}$

$\{t \mid \exists\, s: (s \in R \vee s=t) \}$

$\{t \mid \exists\, s: (s \in R \wedge s=t) \}$

$\{t \mid \neg(\exists\, s: (s \in R \wedge s=t)) \}$

$\{t \mid \forall\, s: (s \in R \wedge s=t) \}$

$\{t \mid \forall\, s: (s \in R \Rightarrow s=t) \}$

$\{t \mid \forall\, s: (s \in R \Rightarrow (s=t \wedge t.A=2 \wedge t.B=2)) \}$

$\{t \mid t \in R \wedge \forall\, s: (s \in R \Rightarrow (s=t \wedge t.A=2 \wedge t.B=2)) \}$

$\{t \mid \forall\, s: (s \in R \wedge (s=t \wedge t.A=2 \wedge t.B=2)) \}$

$\{t \mid \forall\, s: (s \in R \Rightarrow \neg(s=t)) \}$

$\{t \mid t \in K \wedge \forall\, p: p \in P \Rightarrow (\exists\, b: b \in B \wedge b.PNr=p.PNr \wedge b.KNr=t.KNr)\}$

Rezept für sichere Anfragen

Satz:

Eine Anfrage des TRK ist sicher
und damit garantiert domain-unabhängig, wenn

- die Anfrage die Form $\{t \mid t \in R \wedge F(t)\}$ hat und.
- jede existenzquantifizierte Teilformel
  die Form $\exists\, x:\ x \in R \wedge F(t)$ hat und
- jede allquantifizierte Teilformel
  die Form $\forall\, x: x \in R \Rightarrow F(x)$ hat.

6.4 Mächtigkeit relationaler Anfragesprachen

Satz:

- Alle Anfragen, die sich mit der RA ausdrücken lassen, lassen sich auch mit dem sicheren TRK oder dem sicheren DRK ausdrücken.

- Alle Anfragen, die sich mit dem sicheren TRK ausdrücken lassen, lassen sich auch mit der RA oder dem sicheren DRK ausdrücken.

- Alle Anfragen, die sich mit dem sicheren DRK ausdrücken lassen, lassen sich auch mit der RA oder dem sicheren TRK ausdrücken.

6.5 Datalog: Deduktionsregeln als Anfragesprache

Extensionale Repräsentation von Information: Fakten, Tabellen
Intensionale Repräsentation von Information: Regeln, Ableitungen

Beispiel:
intensionale Repräsentation von Flugverbindungen (V)
auf der Grundlage einer extensionalen Repräsentation von Flügen (F)

als Formeln des DRK:

$\forall$ a, z: F (a,z) $\Rightarrow$ V (a,z)

$\forall$ a, z, o: V (a,o) $\wedge$ F (o,z) $\Rightarrow$ V (a,z)

als Fixpunktgleichung in der RA:

V = F $\cup$ (V |x| [V.Zielort=F.Abflugort] F)

Datalog: Definitionen (1)

Eine ***Hornklausel*** über einer endlichen Menge von Prädikatsymbolen ist eine logische Formel der Form

$$\forall\ X1, \ldots, Xn:\ P1\ (Z11, Z12, \ldots, Z1k_1) \wedge \ldots \wedge Pm\ (Zm1, Zm2, \ldots, Zmk_m) \Rightarrow P0\ (Z01, Z02, \ldots, Z0k_0)$$

mit ki-stelligen Prädikaten Pi, Variablen X1, ..., Xn und Konstanten oder Variablen Zij.

Die Formel

$$P0\ (Z01, Z02, \ldots, Z0k_0)$$

wird als ***"Kopf"*** (engl.: head) der Hornklausel bezeichnet, die Formel

$$P1\ (Z11, Z12, \ldots, Z1k_1) \wedge \ldots \wedge Pm\ (Zm1, Zm2, \ldots, Zmk_m)$$

als ***"Rumpf"*** (engl.: body).

Datalog: Definitionen (2)

Ein ***Datalog-Programm*** besteht aus

- einer Menge von Fakten der Form Pi (V1, V2, ..., Vk) mit einem k-stelligen Prädikat Pi und Konstanten V1, ..., Vk und
- einer Menge von Regeln in Form von Hornklauseln.

Ein ***Datalog-Programm mit Negation*** besteht aus

- einer Menge von Fakten und einer Menge von
- Regeln in Form von Klauseln, bei denen Rumpfprädikate negiert sein können.

Eine ***Anfrage*** (Ziel; engl: goal) ist ein Ausdruck der Form P(Z1, ..., Zk) mit einem k-stelligen Prädikat P und Konstanten oder Variablen Z1, ..., Zk, so daß mindestens eines der Zj eine (freie) Variable ist.

Eine Regel r heißt ***rekursiv***,
wenn es Regeln r1, r2, ..., rn gibt mit r1 = rn = r, so daß für alle i ein Rumpfprädikat von ri im Kopf von r(i+1) steht.

Mächtigkeit von Datalog

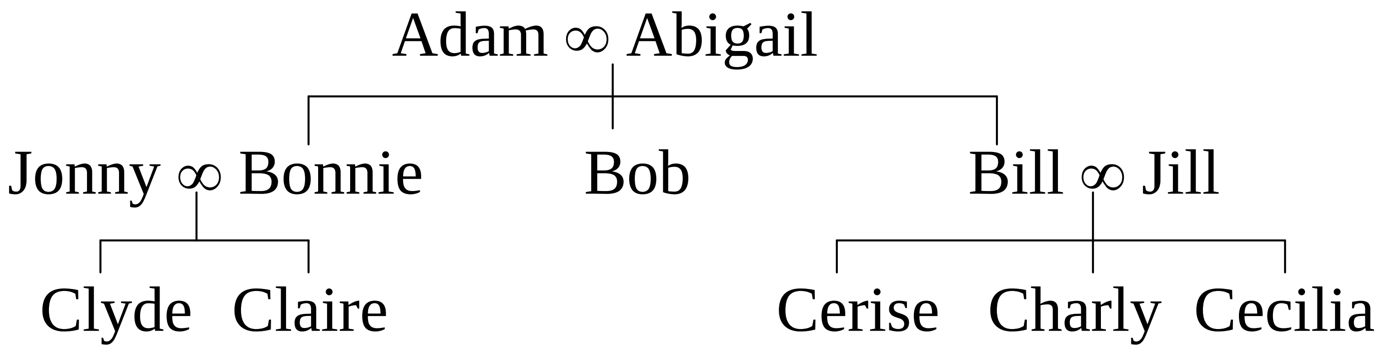
Satz:

1) Die Menge der Anfragen, die sich mit
Datalog ***ohne Rekursion und ohne Negation*** ausdrücken lassen,
ist eine echte Untermenge der Anfragen,
die sich mit dem DRK ausdrücken lassen.

2) Die Menge der Anfragen, die sich mit
Datalog ***mit Negation, aber ohne Rekursion*** ausdrücken lassen,
ist identisch mit der Menge der Anfragen,
die sich mit dem DRK ausdrücken lassen.

3) Die Menge der Anfragen, die sich mit
Datalog ***mit Rekursion und mit Negation*** ausdrücken lassen,
ist eine echte Obermenge der Menge von Anfragen,
die sich mit dem DRK ausdrücken lassen.

Datalog-Beispiel: Fakten



Mann (Adam), Mann (Jonny), Mann (Bob), Mann (Bill), Mann (Clyde) ,
Mann (Charly), Frau (Abigail), Frau (Bonnie), Frau (Jill), Frau (Claire),
Frau (Cerise), Frau (Cecilia)
Ehepaar (Adam, Abigail), Ehepaar (Jonny, Bonnie), Ehepaar (Bill, Jill)
Elternteil (Adam, Bonnie), Elternteil (Adam, Bob), Elternteil (Adam, Bill),
Elternteil (Abigail, Bonnie), Elternteil (Abigail, Bob), Elternteil (Abigail, Bill),
Elternteil (Jonny, Clyde), Elternteil (Jonny, Claire),
Elternteil (Bonnie, Clyde), Elternteil (Bonnie, Claire),
Elternteil (Bill, Cerise), Elternteil (Bill, Charly), Elternteil (Bill, Cecilia),
Elternteil (Jill, Cerise), Elternteil (Jill, Charly), Elternteil (Jill, Cecilia)
SelbeGeneration (Adam, Abigail),
SelbeGeneration (Adam, Adam), SelbeGeneration (Abigail, Abigail)
SpieltMit (Clyde, Charly)

Datalog-Beispiel: Regeln

Elternteil (X, Y)	$\Rightarrow$ Vorfahren (X, Y)
Elternteil (X, Y) $\wedge$ Vorfahren (Y, Z)	$\Rightarrow$ Vorfahren (X, Z)
Elternteil (X, Y) $\wedge$ Mann (X)	$\Rightarrow$ Vater (X, Y)
Elternteil (X, Y) $\wedge$ Frau (X)	$\Rightarrow$ Mutter (X, Y)
Elternteil (X, Y) $\wedge$ Elternteil (X, Z) $\wedge$ $Y \neq Z$	$\Rightarrow$ Geschwister (Y, Z)
Elternteil (X, Y) $\wedge$ Elternteil (U, W) $\wedge$ Geschwister (X, U) $\wedge$ Frau (W)	$\Rightarrow$ Cousine (Y, W)
Elternteil (X, Y) $\wedge$ Elternteil (U, W) $\wedge$ SelbeGeneration (X, U)	$\Rightarrow$ SelbeGeneration (Y, W)
Geschwister (X, Y)	$\Rightarrow$ SpieltMit (X, Y)
SpieltMit (Clyde, Y)	$\Rightarrow$ SpieltMit (Claire, Y)
TRUE	$\Rightarrow$ SpieltMit (Cecilia, Y)
Elternteil (X, Y)	$\Rightarrow$ Verwandt (X, Y)
Geschwister (X, Y)	$\Rightarrow$ Verwandt (X, Y)
Verwandt (X, Y)	$\Rightarrow$ Verwandt (Y, X)
Verwandt (X, Y) $\wedge$ Verwandt (Y, Z)	$\Rightarrow$ Verwandt (X, Z)
Mann (X) $\wedge$ $\neg$Ehepaar (X, Y)	$\Rightarrow$ Ledig (X)
Frau (Y) $\wedge$ $\neg$Ehepaar (X, Y)	$\Rightarrow$ Ledig (Y)