Definition: Prädikatenlogische Sprache erster Stufe (PL1- Sprache)

(ohne Funktionssymbole)
gegeben durch Paar (A, F) mit

- $A =$ Alphabet mit
  - unendlich viele Variable (aus einer Variablenmenge, genannt Var_A)
  - keine oder beliebig viele Konstantensymbole (aus einer Konstantenmenge Konst, genannt $Konst_A$)
  - mindestens ein Prädikatensymbol, möglicherweise unendlich viele Prädikatensymbole (aus einer Prädikatensymbolmenge, genannt Präd_A)
- $F =$ Menge aller wohlgeformten Formeln (Ausdrücke) mit Symbolen aus A

Definition: Term

Ein Konstantensymbol aus $Konst_A$ oder eine Variable aus Var_A.

Definition: Atomare Formel (Atom)

$p(t_1, \cdots, t_n)$ mit p ist n-stelliges Prädikatensymbol aus A und $t_1, \cdots, t_n$ Terme

Satz: Konstruktionsregel für F

kleinste Menge mit

- Jedes Atom aus A ist in F
- Falls f und g in F sind, dann sind auch $\neg f, (f \wedge g), (f \vee g), (f \Rightarrow g), (f \Leftrightarrow g)$ in F
- Falls X eine Variable ist und f in F, dann sind auch $(\forall X)(f)$ und $(\exists X)(f)$ in F

Definition: Interpretation

Eine Interpretation I für eine PL1-Sprache (A, F) ist ein Tripel (Dom, k, ext) mit:

- *dom* ist eine nicht leere Menge, genannt Wertebereich (domain) von I
- k ist eine Abbildung von Konstantensymbolen aus A in *dom*
- *ext* (extension) ist eine Abbildung von Prädikatensymbolen aus A in Mengen von Tupeln, gebildet aus Werten von Dom unter Beachtung der Stelligkeit der Prädikatensymbole. Diese Tupelmengen heißen **Prädikate**, $ext(p)$ wird auch **Ausprägung** von p genannt.

Definition: Wahrheitswerte für PL1

Konstanten und Variablen

$\| t_1 \|_{I,\rho}$: $Var_A \cup Konst_A \rightarrow Dom$ definiert durch:

- $k(c)$ für Konstantensymbole
- $p(X)$ für Variablen

Relationen

$\models_{I,\rho} p(t_1, \cdots, t_n)$ (Unter der Interpretation I und Belegung ρ wahr) genau dann wenn $(\| t_1 \|_{I,\rho}, \cdots \| t_n \|_{I,\rho}) \in ext(\rho))$ für jedes Atom $p(t_1, \cdots, t_n)$ aus F

Definition: Folgerung ($F' \models f$)

Formel $f \in F$ mit $F' \subseteq F$, falls jedes Modell von F' auch ein Modell von f ist.

Definition: unter Folgerbarkeit abgeschlossen

Transitive Hülle von Folgerung

Definition: Theorie

Besteht aus PL1-Sprache (A, F) und Formelmenge $F' \subseteq F$, die unter Folgerbarkeit abgeschlossen ist. (F': Sätze der Theorie)

Definition: Axiomatisierbar

Theorie T, falls es eine entscheidbare Formelmenge $X \subseteq T$ gibt, derart dass alle Sätze von T Folgerungen von X (dann Axiomensystem von T mit Elementen Axiome) sind

Satz: Modell einer Theorie

Eine Interpretation I ist ein Modell einer Theorie T, falls I ein Modell der Menge aller Sätze von T ist.

Definition: Konsistente Theorie

Falls Theorie wenigstens ein Modell hat, sonst inkonsistent.

Definition: formale Ableitung

Eine formale Ableitung einer Formel f von der Menge $\{f_1, ..., f_m\}$ in der PL1- Logik ist eine Folge $b_1, \cdots, b_l$ von Formeln der Sprache (A, F) mit

- b_l ist gleich f
- jede Formel $b_j, j \in \{1, \cdots l\}$ ist eine der Formeln $f_i, i \in \{1, \cdots, m\}$ oder eine allgemeingültige Formel, die aus b_j in der Liste vorangehender Formeln durch Anwendung einer Ableitungsregel der PL1-Logik erhalten werden kann.

Definition: ableitbar ($\vdash$)

Falls es eine formale Ableitung gibt (aus $\emptyset$ oder Formelmenge)

Satz: Vollständigkeitssatz von Gödel

$$\emptyset \models f \Longrightarrow \vdash f$$

Definition: Relationale Sprache

$R = (A, F)$ mit

- Es gibt in A
  - $1 \leq |Konst_A| < \infty$
  - $|\text{Präd}_A| < \infty$
  - ausgezeichnetes, zweistelliges Prädikatensymbol = (Gleichheit)
  - ausgezeichnete Teilmenge einstelliger Prädikatensymbole, die **einfachen Typen**
- Typen, kleinste Menge mit
  - jeder einfache Typ von A ist ein Typ von R
  - falls τ_1, τ_2 Typen von R, dann auch $(\tau_1 \wedge \tau_2), (\tau_1 \vee \tau_2), \neg\tau_1$ Typen von R

Definition: Relationale Interpretation

Sei $R = (A, F)$ relationale Sprache, dann ist eine Interpretation $I = (Dom, k, ext)$ für R eine relationale Interpretation für R, wenn

- k ist eine Bijektion (Dom ist endlich)
- $ext(=) = \{(d, d) | d \in Dom\}$

Definition: Relationale Datenbank

Tripel (R, I, IB) mit

- R ist eine relationale Sprache

- I ist eine relationale Interpretation
- IB ist eine Menge von Formeln von R, so dass insbesondere für jedes n-stellige Prädikatensymbol P, das verschieden ist von "=" und von den einfachen Typen, IB eine Formel der folgenden Gestalt enthalten muss ($\tau_1, \cdots \tau_n$ einfache Typen):

$$(\forall X_1) \cdots (\forall X_n)(p(X_1, \cdots X_n) \Rightarrow \tau_1(X_1) \wedge \cdots \wedge \tau_n(X_n))$$

Definition: Erlaubte relationale Datenbank

Falls I ein Modell von IB ist

Definition: Intentionale, extentionale Prädikatensymbole

- Intentional: durch ein Programm definiert
- Extentional: als Relationen in einer Datenbank gespeichert

Definition: Fixpunkttheorem (Knaster / Tarski)

Sei τ eine monotone Transformation auf einem vollständigen Verband $(V, \leq)$. Dann hat τ einen kleinsten Fixpunkt

$$lfp(\tau) = inf(\{x \in V | \tau(x) lex\})$$

Definition: Datalog Programm

Ein Datalog-Programm P (ohne IBen(Integritätsbedingungen)) ist eine endliche Menge von Horn-Klauseln mit Jedes $d \in P$ ist entweder

- ein Fakt $q(...)$. ohne Variable
- eine sichere Regel $q(...) : -p_1(...), ..., p_n(...)$. mit $q \in iPraedikat$

Eine Regel heißt sicher, wenn alle in ihr vorkommenden Variablen beschränkt sind.

Definition: Bedeutung eines Datalog Programms

Menge der Grundatome, die logisch aus P gefolgert werden können.

Satz von Gödel / Skolem

Eine Klauselmenge P hat ein Modell genau dann wenn P hat ein Herbrand-Modell. Daraus folgt, dass ein Verfahren analog zu Wahrheitstabellen in der Aussagenlogik möglich ist.

Skolemisierung

Jeder Formel der PL1 Logik, kann in eine erfüllbarkeitsäquivalte Formel in Skolem-Form gebracht werden. Dies bedeutet Pränexnormalform und alle Existenzquantoren durch Funktionen ersetzen.

Definition: Herbrand-Interpretation

Eine Teilmenge der Herbrand- Basis

Grundatom

Ein Grundatom f ist eine logische Folgerung einer Menge D von Datalog Klauseln (z.B. $D \vDash f$) $\diamondsuit_{Def}$. Jedes Herbrand Modell von D ist auch ein Modell von f.
Da f ein Grundatom ist gilt $D \vDash f \implies f$ ist in jedem Herbrand-Modell von D enthalten. Das heißt $f \in \bigcap\{I|IHerbrand - ModellvonD\}$.
Sei $f \in \bigcap\{I|IHerbrand - ModellvonD\}$, dann ist f ein Grundatom und jedes Modell von D auch in Modell von f.

Definition: Mege aller Konsequenzen

$cons(D) =_{def} \{f \in HB_D | D \vDash f\}$

Definition: Substitution

Eine Substitution ist eine endliche Menge der Form

$$\{X_1/t_1, \cdots, X_n/t_n\}, X_1, ..., X_n \text{ unterschiedliche Variablen}, t_1, ..., t_n Terme, X_i \neq t_i \tag{1}$$

Sei θ eine Substitution, t ein Term (Variable oder Konstante), so gilt

$$t\theta =_{def} \begin{cases} t_i, & \text{falls } t/t_i \in \theta \\ t, & \text{sonst} \end{cases} \tag{2}$$

Definition: Grundsubstitution

Substitution bei der alle t_i Konstanten sind.

Definition: Unifizierbar

Seien L_1 und L_2 heißen **unifizierbar**, wenn $(\exists$ Substitution $\Theta)(L_1\Theta = L_2\Theta)$. Θ heißt dann **Unifikator**.

Definition: Komposition

Sei $\Theta = \{X_1/t_1, \cdots, X_n/t_n\}, \sigma = \{Y_1/n_1, \cdots, Y_m/t_m\}$ Substitutionen.
Die Komposition $\Theta\sigma$ von Θ und σ erhält man aus

$$X_1/t_1\sigma, \cdots, X_m/t_m\sigma, Y_1/n_q, \cdots, Y_m/n_m \tag{3}$$

Durch Streichen von Elementen der Form Z/Z sowie Y_i/n_i mit $Y_i = X_j$ für ein j$j \in \{1, ..., n\}$

Definition: allgemeinere Substitution

Sei $\Theta = \{X_1/t_1, \cdots, X_n/t_n\}, \sigma = \{Y_1/n_1, \cdots, Y_m/t_m\}$ Substitutionen.

Die Komposition $\Theta\sigma$ von Θ und σ erhält man aus $X_1/t_1\sigma, \cdots, X_m/t_m\sigma, Y_1/n_q, \cdots, Y_m/n_m$

Definition: Beweisbaum

B entsteht aus S durch Anwendung von Θ auf alle Benennungen von Zielknoten. B repräsentiert einen Beweis für $g\Theta$, g benennung der Wurzel von S.

Definition: Tiefe eines Baums

maximale Anzahl von Zielknoten auf einem Pfad von einem Blattknoten zur Wurzel. Entsprechend Knoten der Tiefe i, Ebene i eines Baumes. Zusätzlich: Spezielle Suchbäume (Tiefe 0) für Fakten aus P.

Suchbaum zu cons

Sei P ein Datalog-Programm. Die Suchbaum / Beweisbaum Methode, angewand auf alle Ziele $q(X_1, \cdots, X_{Stelligkeit(q)})$, q intentionales Prädikatesymbol von P, liefert cons(P) als Ergebnis

Suchbaum, Vollständigkeit

Die Suchbaum / Beweisbaum Methode bleibt vollständig für ein Programm P, wenn nur Bäume mit max. Tiefe max_fakt(P) betrachtet werden.

Resolutionsmethode

Für allgemeine Klauselformen entwickelte Methode zum automatischen Beweisen.

Definition: Vollständiger Verband

Partiell geordnete Mengt $(V, \leq)$ bei der zu jeder Teilmenge ein Infinum($\perp_V$) & Suprenum ($\top_V$)besteht. Jeder endliche Verband (und jeder Teilmengenverband) ist vollständig.

Definition: Monotone Transformation

Abbildung τ mit $(\forall a, b \in V)(a \leq b \Rightarrow \tau(a) \leq \tau(b))$.

Definition: Fixpunkt

$a \in V : \tau(a) = a$

Satz: Fixpunkttheorem (Knaster / Tarski)

Sei τ eine monotone Transformation auf einem vollständigen Verband $(V, \leq)$. Dann hat τ einen kleinsten Fixpunkt

$$lfp(\tau) = inf(\{x \in V | \tau(x) lex\})$$

Magic Set Methode

Transformiere ein Programm in eine Version, die für ein gegebenes Ziel die gleiche Ausgabe hat aber das Ziel bei bottom-up Auswertung berücksichtigt wird. Algorithmus, Beispiel hier.

Vorgehen

1.Schritt

Füge für das Ziel $g = q(\cdots)$ die Regel $query^{f\cdots}$[1]$(X_1, \cdots, X_k) : -q^{\alpha}(\cdots)$. ein, wobei $X_1, \cdots, X_k$ Variablen aus $q^{\alpha}(\cdots)$ sind. Erzeuge für jede Regel $r \in P$ und jedes mögliche Bindungsmuster β des Prädikates im Kopf von r eine Regel mit Bindungsmuster für jedes ihrer itensionalen Prädikate. Bestimme dabei unter Beachtung von β für jedes Argument im Rumpf ob es ausgezeichnet ist oder nicht. Falls ein IDB-Pr¨adikat im Rumpf mehrfach auftritt, sollte man es durchnummerieren.

[1] Hochgestellte Zeichen sind Bindungsmuster (wie in Coral)

2.Schritt

Forme P_g^{B2} zu P_g^{magic3}.
Sei $P_g^{magic} := P_g^B$. Mach dann für jedes $r \in P_g^B$ und draus folgend für jedes Vorkommen $p^\beta\_i(t_1, \ldots t_l)$ eines IDB-Prädikates im Rumpf von r folgendes:

- Streiche alle anderen Vorkommen von IDB-Prädikaten im Rumpf von r
- Ersetze p_g^β durch $magic\_r\_p^\beta\_i$
- Streiche alle Variablen aus $(t_1, \ldots t_l)$, die nicht ausgezeichnet sind. [4]
- Streiche alle nicht ausgezeichneten EDB-Prädikate aus r.
- Sei $z^\alpha(s_1, \cdots, s_k)$ das Prädikat im Kopf von r. Streiche alle Variablen aus $(s_1, \cdots, s_k)$, die nicht ausgezeichnet sind; α wird nicht verändert.
  Ersetze $magic\_r\_p^\beta\_i(t_1, \ldots t_l)$ durch $magic\_z^\alpha(s'_1, \cdots s'_{\bar{l}})$[5]
- Füge P_g^{magic} die neuen Regeln hinzu

3. + 4.Schritt

[2] Menge aller erreichbaren Regeln aus Schritt 1
[3] Bezüglich g äquivalent
[4] Bei "Prädikaten" ohne Argumente die entsthen können: Fall entsprechende Relation $\neq \emptyset$ wahr, sonst falsch
[5] Änderungen aus letztem Schritt

for each $r \in P_g^\beta$ **do**
begin
for each $p^\beta\_i(t_1, ..., t_l), p \in iPräd$ im Rumpf von r **do**
begin erzeuge Prädikat $m = magic\_r\_p^\beta\_i(t'_1, ..., t'_{\tilde{l}})$,
wobei die $t'_1, ..., t'_{\tilde{l}}$ die ausgezeichneten Argumente von
$t_1, ..., t_l$ sind;
if p Prädikatensymbol im Kopf von r
then füge m am Beginn des Rumpfes von r ein
else füge m unmittelbar vor $p^\beta\_i(t_1, ..., t_l)$ ein
end; /* Einfügeposition für Semantik ohne Bedeutung */
ersetze Rumpf von r in P_g^{magic} durch den geänderten Rumpf
end;

Figure 1:

for each $r \in P_g^{magic}$ **do**
for each $p^\beta\_i(t_1, ..., t_l)$ im Rumpf von r **do**
$P_g^{magic} := P_g^{magic} \cup \{magic\_p^\beta(t'_1, ..., t'_{\tilde{l}}) :- \quad magic\_r\_p^\beta\_i(t'_1, ..., t'_{\tilde{l}})\}$;

Figure 2:

Definition: Ausgezeichnet

Argument eines Teilziels

Konstantensymbol, gemäß α gebunden, es in einem EDB-Prädikat auftritt, das ein ausgezeichnetes Argument hat.

EDB-Prädikat

Alle seine Argumente sind ausgezeichnet

Definition: Abhängigkeitsgraph

Gerichteter Graph $DG(P) = (V, E)$ eines Programms P, falls V die Menge aller Prädikatensymbole von P und $e = (p, q) \in E \Leftrightarrow q$ kommt im Rumpf einer Regel von P vor, deren Kopfprädikat p ist. e ist mit "$\neg$" benannt, wenn q dabei wenigstens einmal negiert vorkommt.

Definition: Schichtung eines Programms

Eine Folge $\Pi_1, \cdots, \Pi_n$ von Mengen von Prädikatensymbolen mit

- $\{\Pi_1, \cdots, \Pi_n\}$ ist eine Partition der Prädikatensymbole in P
- $q \in \Pi_i, r \in \Pi_j, (q, r)$ Kante in $DG(P) \Rightarrow i \geq j$, für alle Prädikatensymbole q,r aus P, $i, j \in \{1, \cdots, n\}$
- $q \in \Pi_i, r \in \Pi_j, (q, r)$ mit "$\neg$" markierte Kante in $DG(P) \Rightarrow i > j$

Definition: Geschichtetes Programm

Ein Programm P heißt geschichtet, wenn P eine Schichtung hat.

Satz: Schichtung / Zyklus

Ein Programm P ist geschichtet, gdw. $DG(P)$ enthält keinen Zyklus mit einer Kante, die mit "$\neg$" markiert ist.

Eigenschaften: Perfektes Modell

"Kleinstes" der minimalen Modelle, wenn das stärkste Gewicht darauf gelegt wird, dass die Prädikate niedriger Schichten klein bleiben

Satz: sicher geschichtet / perfektes Modell

Sei P ein sicheres, geschichtetes Programm, dann ist das Perfekte Modell von P unabhängig von der Wahl der Schichtungen

Definition: Anfrage

Sei $\sigma = \{(RT_1, \alpha_1), \cdots, (RT_n m\alpha_n)\}$ ein relationales Datenbankschema über $\alpha = \bigcup_{i=1}^{n} \alpha_i$ mit Wertebereichsfunktion dom. Sei α in eine genügend große Attributmenge α_0 eingebettet[6].
Eine Anfrage q auf σ ist eine partielle Funktion

$$q|Z\sigma \Rightarrow R_\beta^\infty$$

Definition: Anfragesprache

Eine Anfragesprache zu σ ist eine Menge L_0 von Ausdrücken zusammen mit einer Bedeutungsfunktion (in Zeichen: $({L_\sigma}^7, \mu^8)$), so dass für jeden Ausdruck $e \in L_\sigma$ gilt: $\mu(e)$ ist eine Anfrage von σ

Definition: Ausdruckskraft

Die Ausdruckskraft einer Anfragesprache (L_σ, μ) zu einer DB-Schema σ ist definiert als $\mu(L_\sigma) =_{Def.} \{\mu(e)|e \in L_\sigma\}$. Eine Sprache (L_σ, μ') ist **ausdrucksstärker** als eine Sprache (L_σ, μ), wenn gilt: $\mu(L_\sigma) \subseteq \mu'(L'_\sigma)$ Im Fall $\mu(L_\sigma) = \mu'(L'_\sigma)$ werden die Sprachen **äquivalent** genannt

[6] Enthalten darin
[7] Sprache
[8] Bedeutungsfunktion