Relationenkalkül

Relationenkalkül

- Zwei Ausprägungen: *Tupelrelationenkalkül* (TRK) und *Domänenrelationenkalkül* (DRK).
- Kalkül hat Variablen*, Konstanten, Vergleichsoperatoren, logische Verknüpfungen* und *Quantoren*.
  - *TRK*: Variablen bezeichnen Tupel (d.h., werden daran gebunden).
  - *DRK*: Variables bezeichnen *Domänenelemente* (= Wertebereiche von Attributen).
  - TRK and DRK sind einfache Teilmengen von First-Order-Logik.
- Ausdrücke im Kalkül werden *Formeln* genannt. Ein Antwort-Tupel ist im wesentlichen eine Zuweisung von Konstanten zu Variablen, so daß die Formel *true* lautet.

Tupelrelationenkalkül

- *Query* hat die Form: {T | *p*(T) }

  Mit T = Tupelvariable

  p(T) = Formel, die T beschreibt
- *Resultat* umfaßt die Menge der Tupel t, für die die Formel *p*(T) = TRUE ist
- *Formel* ist rekursiv definiert, beginnt mit *atomaren Formeln*

  (Auswahl von Tupeln aus Relationen oder Wertvergleiche) und Konstruktion größerer und besserer Formeln durch Verwendung von *logischen Verknüfungen*

TRK-Formeln

- *Atomare Formel:*
  - R $\in$ Rname ,
  - oder R.a *op* S.b, or R.a *op* constant (R, S Tupelvariablen)
  - *op* ist aus <, >, =, $\leq$, $\geq$, $\neq$
- *Formel:*
  - Eine atomare Formel, oder
  - $\neg p$, $p \vee q$, $p \wedge q$, wobei *p* und *q* Formeln sind, oder
  - $\exists$R (*p*(R)) , wobei Variable X *frei* in *p*(X), oder
  - $\forall$R (*p*(R)) , wobei Variable X *frei* in *p*(X)
- Die Verwendung von Quantoren $\exists$X und $\forall$X geschieht, um X zu *binden*.
  - Eine Variable, die ungebunden ist, ist frei.
- In einer Query { T | *p*(T) } ist T die einzige freie Variable in der Formel *p*.

TRK: Beispiele

Finde die Namen und Alter aller Segler mit einem Rating größer 7.

$$\{P \mid \exists S \in Sailors (S.rating > 7 \land P.sname = S.sname \land P.age = s.age\}$$

Finde die Namen der Segler, die ein rotes Boot reserviert haben.

$$\{P \mid \exists S \in Sailors \exists R \in Reserves \exists B \in Boats (R.sid = S.sid \land B.bid = R.bid \land B.color = 'red' \land P.sname = S.sname)\}$$

Finde die Namen der Segler, die mindestens zwei Boote reserviert haben.

$$\{P \mid \exists S \in Sailors \exists R1 \in Reserves \exists R2 \in Reserves (S.sid = R1.sid \land R1.sid = R2.sid \land R1.bid \neq R2.bid \land P.sname = S.sname)\}$$

TRK: Beispiele (2)

Finde die Namen der Segler, die alle Boote reserviert haben.

$$\{P \mid \exists S \in Sailors \, \forall B \in Boats \, (\exists R \in Reserves$$

$$(S.sid = R.sid \wedge R.bid = B.bid \wedge P.sname = S.sname))\}$$

Finde die Segler, die alle roten Boote reserviert haben.

$$\{P \mid \exists S \in Sailors \; \forall B \in Boats$$

$$(B.color = 'red' \Rightarrow (\exists R \in Reserves \;\; (S.sid = R.sid \wedge R.bid = B.bid)))\}$$

Andere Schreibweise:

p$\Rightarrow$ q ist logisch äquivalent to $\neg$p$\vee$q

$$\{P \mid \exists S \in Sailors \, \forall B \in Boats$$

$$(B.color \neq 'red' \vee (\exists R \in Reserves (S.sid = R.sid \wedge R.bid = B.bid)))\}$$

Domänenrelationenkalkül

- *Query* hat die Form: $\{\langle x_1,x_2, ..., x_n\rangle \mid p(\langle x_1,x_2, ..., x_n\rangle) \}$
  Mit x_i = Domänenvariable oder Konstante
  $p(\langle x_1,x_2, ..., x_n\rangle)$ = Formel im DRK, dessen freie Variable x_i sind mit $1 \leq i \leq n$
- *Resultat* umfaßt die Menge der Tupel $\langle x_1,x_2, ..., x_n\rangle$, für die die Formel = TRUE ist
- *Formel* ist rekursiv definiert, beginnt mit *atomaren Formeln* (Auswahl von Tupeln aus Relationen oder Wertvergleiche) und Konstruktion größerer und besserer Formeln durch Verwendung von *logischen Verknüpfungen*
- Konstruktion der Formeln analog zum TRK, wobei gilt:
  Tupelvariable R = $\langle x_1,x_2, ..., x_n\rangle$

DRK: Beispiele

Finde die Namen und Alter aller Segler mit einem Rating größer 7.

$$\left\{ \langle I,N,T,A\rangle \mid \langle I,N,T,A\rangle \in Sailors \wedge T > 7 \right\}$$

- Die Bedingung $\langle I,N,T,A\rangle \in Sailors$ sichert, daß die Domain-Variablen *I, N, T* und *A* an die Felder des Tupels der Relation Sailors gebunden werden.
- Der Term $\langle I,N,T,A\rangle$ links vom ` | ’ (lies “so daß“) besagt, daß jedes Tupel $\langle I,N,T,A\rangle$, das die Bedingung $T>7$erfüllt, zur Ergebnis-relation gehört.

Finde die Namen der Segler mit einem Rating > 7, die das Boot #103 reserviert haben.

$$\{ \langle I,N,T,A\rangle \mid \langle I,N,T,A\rangle \in Sailors \wedge T > 7 \wedge$$

$$\exists Ir,Br,D \left(\langle Ir,Br,D\rangle \in \mathrm{Re}serves \wedge Ir = I \wedge Br = 103 \right) \}$$

Unsichere Queries, Ausdrucksmächtigkeit

- Es ist möglich, syntaktisch korrekte Anfragen im Kalkül zu formulieren, die eine unendliche Anzahl von Ergebnissen produzieren! Solche Anfragen heißen *unsicher*.
  - z.B. $\left\{ S \mid \neg \left(S \in Sailors \right) \right\}$
- Es ist bekannt, daß jede Query, die in der Relationenalgebra ausgedrückt werden kann, als eine sichere Query im TRK/DRK ausgedrückt werden kann; die Umkehrung gilt ebenso.
- *Relationale Vollständigkeit:* Eine Query Language (z.B. SQL) kann jede Anfrage ausdrücken, die sich in Relationenalgebra / Relationenkalkül ausdrücken läßt.
- Relationenkalkül ist nicht-prozedural, Nutzer formulieren Anfragen, indem sie das Ergebnis beschreiben (WHAT - not HOW), d.h. deklarativ
- Algebra und sicheres Kalkül haben dieselbe Ausdruckskraft (führt zum Begriff relationale Vollständigkeit).