

12. Übung zur Vorlesung “Objektorientierte und Deduktive Datenbanken” im Wintersemester 2004 – mit Musterlösungen –

Prof. Dr. Gerd Stumme, Dipl.-Inform. Christoph Schmitz

31. Januar 2005

Aufgabe 1 – Modellierung in F-Logic

1. Modellieren Sie passend zu dem Beispiel aus der Vorlesung folgende Sachverhalte:

- Gebäude haben eine Adresse und beinhalten eine Menge von Räumen. Für jedes Gebäude ist ein Angestellter der Hausmeister.
- Räume haben eine Größe und gehören jeweils einer Abteilung; ein Raum kann frei, vorgemerkt oder belegt sein.
- Ein Gebäude soll eine Methode haben, um freie Räume zu erfragen:
 $building[freeRooms \Rightarrow room]$.
- Ein Hörsaal ist eine spezielle Art von Raum. Ein Hörsaal hat eine Anzahl von Plätzen und kann eines oder mehrere der Ausstattungsmerkmale Beamer, Audio und Video haben.

$building[address \Rightarrow addressType; janitor \Rightarrow person; rooms \Rightarrow room; freeRooms \Rightarrow room]$

$wa73 : addressType$

$hopla : addressType$

$free : roomState$

$reserved : roomState$

$occupied : roomState$

$room[state \Rightarrow roomState; size \Rightarrow integer; owner \Rightarrow dept]$

$video : facility$

$audio : facility$

$beamer : facility$

$lectureHall :: room$

$lectureHall[seats \Rightarrow integer; facilities \Rightarrow facility]$

$X[freeRooms \Rightarrow Y] \leftarrow X : building[rooms \Rightarrow Y] \wedge Y[state \rightarrow free]$

2. Stellen Sie folgende Anfragen:

(i) Welche Räume sind am Standort *wa73* frei?

$? - X : building[rooms \rightarrow Y; address \rightarrow wa73] \wedge Y[state \rightarrow free]$

(ii) Wer ist der Hausmeister des Standorts *wa73*?

$? - X : building[address \rightarrow wa73; janitor \rightarrow Z]$

(iii) An welchen Standorten gibt es freie Hörsäle mit Beamer?

$? - X : building[rooms \rightarrow Y; address \rightarrow Z] \wedge Y : lectureHall[state \rightarrow free; facilities \rightarrow beamer]$

Aufgabe 2 – F-Logic-Regeln

Geben Sie F-Logic-Regeln an, um folgende Anfragen auf dem Beispielmmodell zu unterstützen:

(i) Zu Personen sollen die Co-Autoren anfragbar sein, also $? - mary[coauthor \rightarrow X]$.

$X[coauthor \rightarrow Y] \leftarrow X : Faculty[papers \rightarrow Z] \wedge Y : Faculty[papers \rightarrow Z] \wedge \neg(X = Y)$

(ii) Man möchte zu Departments die Veröffentlichungen anfragen: $? - cs_1[publications \rightarrow X]$.

$X[publications \rightarrow Y] \leftarrow Z : faculty[affiliation \rightarrow X; papers \rightarrow Y]$

(iii) Implementieren Sie die Methode $empl[superior \Rightarrow empl]$, die transitiv für jeden Angestellten den Chef, den Chef-des-Chefs, den Chef-des-Chefs-des-Chefs usw. zurückgibt.

$X[superior \rightarrow Y] \leftarrow X : empl[boss \rightarrow Y : empl] \vee (X[boss \rightarrow Z] \wedge Z[superior \rightarrow Y])$

Aufgabe 3 – Schema-Browsing in F-Logic

Können Sie sich vorstellen, wie man folgende Anfragen in F-Logic formulieren kann:

(i) In welcher Beziehung stehen *bob* und *mary*?

$? - bob[X \rightarrow mary]$

(ii) Haben *bob* und *mary* gemeinsame Verbindungen zu *cs₁*?

$? - bob[X \rightarrow cs_1] \wedge mary[X \rightarrow cs_1]$

Aufgabe 4 – F-Logic

In den F-Logic-Beispielen aus der Vorlesung wird unter anderem ausgedrückt, daß die Klasse *faculty* (nicht ihre Objekte!) einen Durchschnittslohn von 50000 hat:

faculty[*avgSalary* \rightarrow 50000]

Allerdings hat *avgSalary* keinen Typ, was so nicht sein darf. Können Sie das Problem beheben?

(*Tip: Klassen sind Objekte!*)

Wir können eine Klasse einführen, die *faculty* als Instanz beinhaltet, und dort einen Typ für *avgSalary* festlegen:

faculty : *employmentGroup*

employmentGroup[*avgSalary* \Rightarrow *integer*]

Damit ist festgelegt, daß *avgSalary* vom Typ *integer* ist.