

**Übung zur Vorlesung *Grundlagen: Datenbanken* im WS21/22**  
Christoph Anneser, Josef Schmeißer, Moritz Sichert, Lukas Vogel (gdb@in.tum.de)  
<https://db.in.tum.de/teaching/ws2122/grundlagen/>

**Blatt Nr. 03**

Tool zum Üben der relationalen Algebra: <https://dbis-uibk.github.io/relax/calc>.  
Für das Unischema im Dropdown links „Kemper Datenbanksysteme“ auswählen.

**Hausaufgabe 1**

Formulieren Sie folgende Anfragen auf dem bekannten Universitätsschema in der Relationalen Algebra:

- Finden Sie die *Assistenten* von *Professoren*, die den Studenten Fichte unterrichtet haben – z.B. als potentielle Betreuer seiner Diplomarbeit.
- Finden Sie die *Studenten*, die *Vorlesungen* hören (bzw. gehört haben), für die ihnen die direkten Voraussetzungen fehlen.

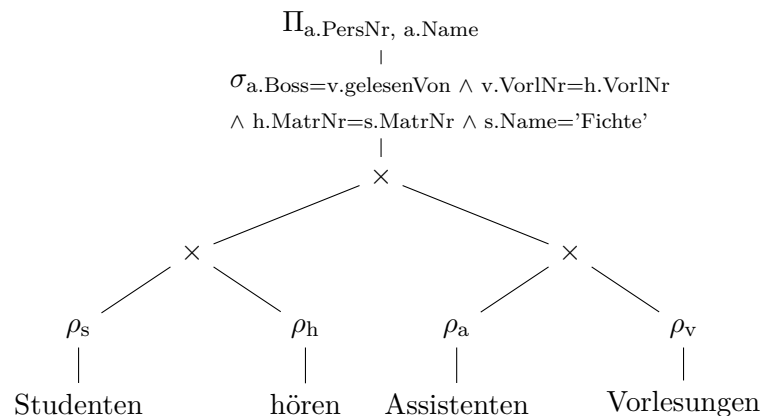
**Lösung:**

Die Anfragen sehen in relationaler Algebra wie folgt aus:

- Folgende Abfrage bildet zuerst das Kreuzprodukt über alle beteiligten Relationen, d.h. *Studenten*, *Vorlesungen*, *Assistenten* und *hören*. Anschließend erfolgt eine umfangreiche Selektion, die die auf Fichte zugeschnittenen Tupel extrahiert.

$$\Pi_{a.PersNr, a.Name}(\sigma_{a.Boss=v.gelesenVon \wedge v.VorlNr=h.VorlNr \wedge h.MatrNr=s.MatrNr \wedge s.Name='Fichte'}(\rho_a(\text{Assistenten}) \times \rho_s(\text{Studenten}) \times \rho_v(\text{Vorlesungen}) \times \rho_h(\text{hören})))$$

In Operatorbaumdarstellung:

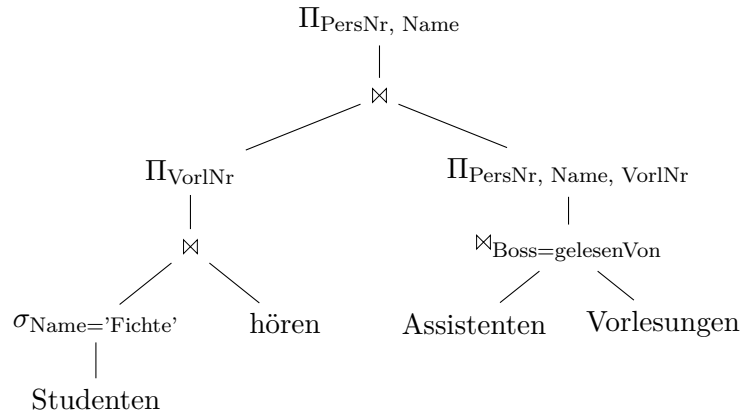


Die Bildung des Kreuzprodukts gilt es nach M\u00f6glichkeit zu vermeiden, da dadurch mitunter sehr gro\u00dfe Zwischenergebnisse entstehen. Dies kann zu sp\u00fcrbaren Leistungseinbu\u00dfen w\u00e4hrend der Anfragebearbeitung f\u00fchren. Folgende Anfrage berechnet dieselbe Ergebnismenge, setzt jedoch bereits Optimierungstechniken, wie fr\u00fche Selektion

und den (natürlichen) Verbundoperator ein.

$$\Pi_{\text{PersNr, Name}}((\Pi_{\text{PersNr, Name, VorlNr}}(\text{Assistenten} \bowtie_{\text{Boss=gelesenVon}} \text{Vorlesungen})) \bowtie (\Pi_{\text{VorlNr}}(\sigma_{\text{Name='Fichte'}}(\text{Studenten}) \bowtie \text{hören})))$$

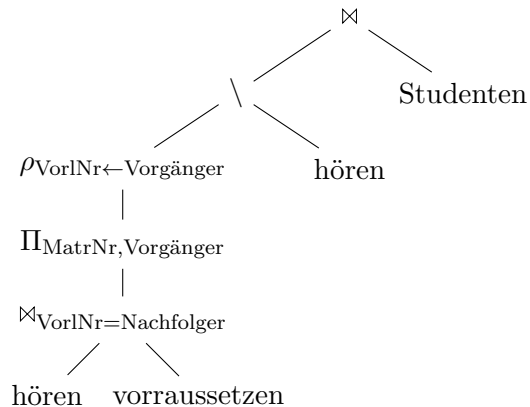
In Operatorbaumdarstellung:



- b) Wir konstruieren eine hypothetische Ausprägung der Relation *hören*, die gelten müsste, wenn alle Studenten alle benötigten Vorgängervorlesungen hören. Von dieser Menge ziehen wir die tatsächliche Ausprägung von *hören* ab, so dass diejenigen Einträge übrig bleiben, bei denen ein Student die Vorgängervorlesung nicht hört (bzw. gehört hat).

$$R := (\rho_{\text{VorlNr} \leftarrow \text{Vorgänger}}(\Pi_{\text{MatrNr, Vorgänger}}(\text{hören} \bowtie_{\text{VorlNr=Nachfolger}} \text{voraussetzen})) - \text{hören}) \bowtie \text{Studenten}$$

In Operatorbaumdarstellung:



## Hausaufgabe 2

Formulieren Sie folgende Anfragen auf dem bekannten Universitätsschema im Tupel- und Domänenkalkül:

- Finden Sie die *Vorlesungen*, die keine Hörer haben.
- Finden Sie die *Studenten*, die alle *Vorlesungen* hören.

**Lösung:**

- Finden Sie die *Vorlesungen*, die keine Hörer haben.

### Formulierung im Tupelkalkül

$$\{v \mid v \in \text{Vorlesungen} \wedge \exists h \in \text{hören}(v.\text{VorlNr} = h.\text{VorlNr})\}$$

oder

$$\{v \mid v \in \text{Vorlesungen} \wedge \forall h \in \text{hören}(v.\text{VorlNr} \neq h.\text{VorlNr})\}$$

### Formulierung im Domänenkalkül

$$\{[v,t,s,g] \mid [v,t,s,g] \in \text{Vorlesungen} \wedge \exists m([m,v] \in \text{hören})\}$$

b) Finden Sie die *Studenten*, die alle *Vorlesungen* hören.

### Formulierung im Tupelkalkül

$$\{s \mid s \in \text{Studenten} \wedge \forall v \in \text{Vorlesungen}(\exists h \in \text{hoeren}(v.\text{VorlNr} = h.\text{VorlNr} \wedge s.\text{MatrNr} = h.\text{MatrNr}))\}$$

### Formulierung im Domänenkalkül

$$\{[m,n,s] \mid [m,n,s] \in \text{Studenten} \wedge \forall v,t,sws,g([v,t,sws,g] \in \text{Vorlesungen} \Rightarrow [m,v] \in \text{hören})\}$$

## Hausaufgabe 3

Finden Sie alle *Studenten*, die alle *Vorlesungen* gehört haben, die von Sokrates gelesen wurden. Formulieren Sie die Anfrage

- in der Relationenalgebra,
- im relationalen Tupelkalkül und
- im relationalen Domänenkalkül.

**Lösung:**

### Formulierung in relationaler Algebra

1. Wir ermitteln die von Sokrates gelesenen *Vorlesungen* (nur die Vorlesungsnummern in diesem Fall):

$$V := \Pi_{\text{VorlNr}}(\text{Vorlesungen} \bowtie_{\text{gelesenVon} = \text{PersNr}(\sigma_{\text{Name}='Sokrates'}(\text{Professoren}))})$$

2. Durch Anwendung des Divisionsoperators bekommen wir die Matrikelnummern der *Studenten*, die alle von Sokrates gehaltenen *Vorlesungen* gehört haben:

$$M := \text{hören} \div V$$

**Wichtig:** Für die relationale Division muss sichergestellt sein, dass das Schema von  $V$  eine Teilmenge des Schemas von  $\text{hören}$  ist:  $\text{sch}(V) \subset \text{sch}(\text{hören})$

3. Aus *Studenten* ermitteln wir zusätzlich die Namen der Studenten:

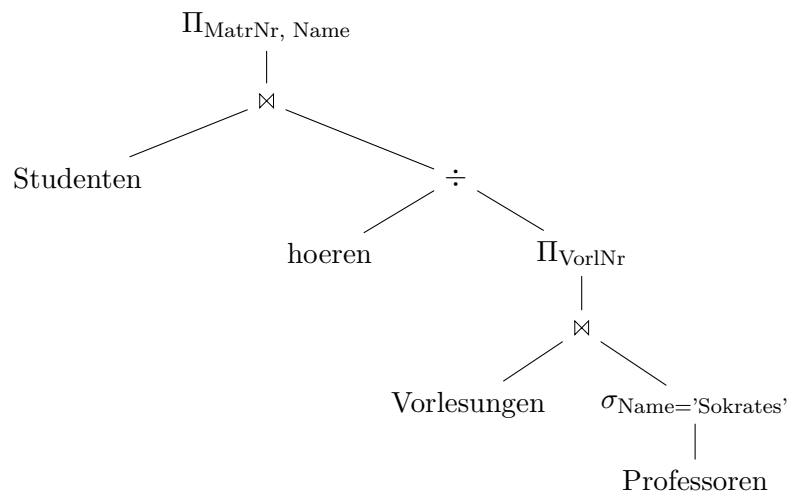
$$S := \Pi_{\text{MatrNr}, \text{Name}}(\text{Studenten} \bowtie M)$$

Zusammengefasst ergibt sich dann folgender Ausdruck:

$$\Pi_{\text{MatrNr}, \text{Name}}(\text{Studenten} \bowtie (\text{hören} \div \Pi_{\text{VorlNr}}(\text{Vorlesungen} \bowtie_{\text{gelesenVon} = \text{PersNr}}(\sigma_{\text{Name}='Sokrates'}(\text{Professoren}))))))$$

**Hinweis:** Die Zusammenfassung ist optional. Relationale Ausdrücke können wie oben gezeigt *modularisiert* werden indem Teilausdrücke (und die dadurch definierten Relationen) an Variablen zugewiesen werden.

In Operatorbaumdarstellung:



### Formulierung im Tupelkalkül

1. Wir ermitteln mittels des Allquantors alle Vorlesungen, die von Sokrates gelesen werden.
2. Wir fordern, dass ein Student alle diese Vorlesungen gehört hat (Existenzquantor).

$$\{s \mid s \in \text{Studenten} \wedge \forall v \in \text{Vorlesungen} \\ (\exists p \in \text{Professoren}(v.\text{gelesenVon} = p.\text{PersNr} \wedge p.\text{Name} = \text{'Sokrates'}) \\ \Rightarrow \exists h \in \text{hören}(s.\text{MatrNr} = h.\text{MatrNr} \wedge v.\text{VorlNr} = h.\text{VorlNr}))\}$$

### Formulierung im Domänenkalkül

Das Vorgehen ist analog zu dem beim relationalen Tupelkalkül.

$$\begin{aligned} & \{ [m,n] \mid \exists s([m,n,s] \in \text{Studenten}) \\ & \quad \wedge \forall v,t,\text{sWS},g( \\ & \quad \quad ([v,t,\text{sWS},g] \in \text{Vorlesungen} \wedge \exists rg,ra([g, \text{'Sokrates'},rg,ra] \in \text{Professoren})) \\ & \quad \quad \Rightarrow \\ & \quad \quad ([m,v] \in \text{hören}) \\ & \quad ) \\ & \} \end{aligned}$$

### Hausaufgabe 4

Gegeben seien die beiden Relationen  $R : \{[a_1, \dots, a_n]\}$  und  $S : \{[b_1, \dots, b_m]\}$ . Geben Sie die folgenden Ausdrücke im Tupel- und Domänenkalkül an:

- a)  $Q_1 := R \bowtie_{a_1=b_1} S$
- b)  $Q_2 := R \bowtie_{a_1=b_1} S$
- c)  $Q_3 := R \times_{a_1=b_1} S$
- d)  $Q_4 := R \triangleleft_{a_1=b_1} S$

### Lösung:

Bitte beachten Sie, dass in dieser Aufgabe ausschließlich allgemeine *Theta*-Joins ( $\bowtie_{\Theta}, \bowtie_{\Theta}, \dots$ ) verwendet werden. Gemäß Definition werden somit alle Attribute der beiden Eingaberelationen in die Ausgabere Relation projiziert, einschließlich der Attribute, welche in der Joinbedingung enthalten sind. Lediglich bei *natürlichen* Joins, wo implizit eine Gleichheitsbedingung für alle gleichnamigen Attribute erfüllt sein muss, werden gleichnamige Attribute nicht doppelt in die Ausgabere Relation projiziert. Siehe hierzu auch Folie Kapitel 3, "Andere Join-Arten".

- a)  $Q_1 := R \bowtie_{a_1=b_1} S$

### Formulierung im Tupelkalkül

$$Q_1 := \{ [r.a_1, \dots, r.a_n, s.b_1, \dots, s.b_m] \mid r \in R \wedge s \in S \wedge r.a_1 = s.b_1 \}$$

Da der Joinoperator Tupel aus verschiedenen Relationen verbindet, müssen für die Ergebnismenge neue Tupel mithilfe des Tupelkonstruktors konstruiert werden:  $[\text{attribut}_1 : \text{wert}_1, \dots, \text{attribut}_n : \text{wert}_n]$ . Die oben verwendete Tupelkonstruktion  $[r.a_1, \dots]$  ist eine verkürzte Schreibweise für  $[a_1 : r.a_1, \dots]$  und kann verwendet werden, wenn der Attributname im neuen Tupel unverändert bleibt.

Im Falle des Thetajoins kann auch die Tupelkonkatenation  $t_1 \circ t_2$  verwendet werden:

$$Q_1 := \{ r \circ s \mid r \in R \wedge s \in S \wedge r.a_1 = s.b_1 \}$$

### Formulierung im Domänenkalkül

$$Q_1 := \{[a_1, \dots, a_n, b_1, \dots, b_m] \mid [a_1, \dots, a_n] \in R \wedge [b_1, \dots, b_m] \in S \wedge a_1 = b_1\}$$

oder

$$Q_1 := \{[a_1, \dots, a_n, b_1 : a_1, b_2, \dots, b_m] \mid [a_1, \dots, a_n] \in R \wedge [a_1, b_2, \dots, b_m] \in S\}$$

b)  $Q_2 := R \bowtie_{a_1=b_1} S$

### Formulierung im Tupelkalkül

$$Q_2 := Q_1 \cup \{[r.a_1, \dots, r.a_n, b_1 : \text{null}, \dots, b_m : \text{null}] \mid r \in R \wedge \exists s \in S (r.a_1 = s.b_1)\}$$

### Formulierung im Domänenkalkül

$$Q_2 := Q_1 \cup \{[a_1, \dots, a_n, b_1 : \text{null}, \dots, b_m : \text{null}] \mid [a_1, \dots, a_n] \in R \wedge \exists c_2, \dots, c_m ([a_1, c_2, \dots, c_m] \in S)\}$$

c)  $Q_3 := R \bowtie_{a_1=b_1} S$

### Formulierung im Tupelkalkül

$$Q_3 := \{s \mid s \in S \wedge \exists r \in R (r.a_1 = s.b_1)\}$$

### Formulierung im Domänenkalkül

$$Q_3 := \{[b_1, \dots, b_m] \mid [b_1, \dots, b_m] \in S \wedge \exists a_2, \dots, a_n ([b_1, a_2, \dots, a_n] \in R)\}$$

d)  $Q_4 := R \triangleleft_{a_1=b_1} S$

### Formulierung im Tupelkalkül

$$Q_4 := \{s \mid s \in S \wedge \nexists r \in R (r.a_1 = s.b_1)\}$$

### Formulierung im Domänenkalkül

$$Q_4 := \{[b_1, \dots, b_m] \mid [b_1, \dots, b_m] \in S \wedge \nexists a_2, \dots, a_n ([b_1, a_2, \dots, a_n] \in R)\}$$