Query Optimization: Exercise
Session 1

Bernhard Radke

October 22, 2018
Exercise sessions are here to illustrate the material of the course with examples, special cases, etc.

Homework every week: programming assignment and a few problems

Do 75% or better to get a bonus of 0.3 on your exam grade

Written exam at the end of the semester (presumably on Feb. 26)

Slides on the website (db.in.tum.de/teaching/ws1819/queryopt)
Algebra Revised
uni schema:

- Studenten : {[MatrNr: integer, Name: string, Semester: integer]}
- Professoren : {[PersNr: integer, Name: string, Rang: string, Raum: integer]}
- Assistenten : {[PersNr: integer, Name: string, Fachgebiet: string, Boss: integer]}
- hoeren : {[MatrNr: integer, VorlNr: integer]}
- voraussetzen : {[Vorgaenger: integer, Nachfolger: integer]}
- pruefen : {[MatrNr: integer, VorlNr: integer, PersNr: integer, Note: decimal]}
Relational Calculus
what the result looks like (declarative)

tuple calculus: \( \{ t \mid P(t) \} \)
- \( \{ p \mid p \in \text{Professoren} \land p.\text{Rang} = 'C4' \} \)
- \( \{ s \mid s \in \text{Studenten} \land \exists h \in \text{hoeren}(s.\text{MatrNr} = h.\text{MatrNr}) \land \exists v \in \text{Vorlesungen}(h.\text{VorlNr} = v.\text{VorlNr}) \land \exists p \in \text{Professoren}(p.\text{PersNr} = v.\text{gelesenVon} \land p.\text{Name} = 'Curie')) \}

domain calculus: \( \{ [v_1, \ldots, v_n] \mid P(v_1, \ldots, v_n) \} \)
- \( \{ [p, n, r, o] \mid [p, n, r, o] \in \text{Professoren} \land r = 'C4' \} \)
- \( \{ [m, n, s] \mid \exists m([m, n, s] \in \text{Studenten} \land \exists v([m, v] \in \text{hoeren} \land \exists p([v, t, d, p] \in \text{Vorlesungen} \land \exists a([p, a, r, o] \in \text{Professoren} \land a = 'Curie')))) \} \)
compare that to SQL

- SELECT * FROM Professoren p WHERE p.Rang='C4'
- SELECT s.MatrNr, s.Name, s.Semester
  FROM Studenten s, hoeren h, Vorlesungen v, Professoren p
  WHERE s.MatrNr=h.MatrNr AND h.VorlNr=v.VorlNr AND
  v.gelesenVon=p.PersNr AND p.Name='Curie'

- what the result looks like (declarative)
Relational Algebra
how the result is built (procedural)

- \( \sigma_{\text{Rang} = 'C4'}(\text{Professoren}) \)
- \( \sigma_{\text{S.MatrNr} = \text{H.MatrNr}}(S \times \sigma_{\text{H.VorlNr} = \text{V.VorlNr}}( \text{H} \times \sigma_{\text{V.gelesenVon} = \text{P.PersNr}}( \text{V} \times \sigma_{\text{P.Name} = 'Curie'}(P)))) \)
- \( S \Join (H \Join (V \Join \sigma_{\text{V.gelesenVon} = \text{P.PersNr}} \sigma_{\text{P.Name} = 'Curie'}(P)))) \)
Textbook Optimization
- Translate SQL into an executable plan
- Many equivalent plans
- Large differences in resource consumption
- Minimize cost function

\[
C_{\text{out}}(T) = \begin{cases} 
0 & \text{if } T \text{ is a leaf } R_i \\
|T| + C_{\text{out}}(T_1) + C_{\text{out}}(T_2) & \text{if } T = T_1 \Join T_2
\end{cases}
\]
Find all Students that attend the course 'Ethik'

- SQL query
- canonical translation
- break up conjunctive selections
- push down selections
- introduce joins
- determine join order
- introduce and push down projections
Programming Assignments

TinyDB

Query Optimization: Exercise

Session 1

October 22, 2018
very simple database system
storage layer and runtime system already implemented
you will build a compile time system step by step
Code: fork gitlab.db.in.tum.de/qo18/tasks
General

- You can work in groups of up to two students
- Include a file containing your names, email addresses and matriculation numbers into your fork.
- Handwritten (and/or scanned) solutions will not be accepted! Use LaTeX (preferable) or Word.
- Push as PDF into your gitlab fork

Programming

- Target: GNU/Linux
- Language: c++ (great opportunity to learn it)
- Submissions:
  - You can work within the TinyDB directory, changing its structure if needed
  - (Briefly) comment the code: every class, field, method, design choice
Slides: db.in.tum.de/teaching/ws1819/queryopt
Exercise task: gitlab
Questions, Comments, etc:
  mattermost @ mattermost.db.in.tum.de/qo18
Exercise due: 9 AM next monday