

Aufgabe 24 (Wiederholungsfragen):

Arbeiten Sie noch einmal das Skript zum Kapitel 6 bis einschließlich Abschnitt 6.2.1 durch, damit Sie hinterher die folgenden Fragen beantworten können:

a) Welche Phasen der Anfrageoptimierung gibt es?

Übersetzung und Sichtexpansion

Logische Optimierung

Physische Optimierung

Relationenalgebra, Operatorbaum

Zugriffspläne, Operatorbäume m. konkreten Algorithmen und Zugriffspfaden

Kostenbasierte Auswahl

Codeerzeugung

Ausführung

b) Wie ist der Zusammenhang zwischen einem Operatorbaum und einem Ausdruck der relationalen Algebra?

Die Wurzel stellt die Projektion dar, die Selektionen (Bedingungen) enthält. Diese wiederum enthalten die Verknüpfungen der Relationen.

c) Was ist ein Anfrageausführungsplan? Wie erhält man ihn?

Man erhält ihn durch Umformung; statt Kreuzprodukte werden Verbunde verwendet, Anfragen werden hierarchisch geordnet.

d) Betrachten Sie die Regeln zur logischen Optimierung und machen Sie sich klar, in welcher Reihenfolge sie angewendet werden sollten. Wie wirkt sich jede der Regeln auf den Anfrageausführungsplan aus?

1. \bowtie ist kommutativ
sagt aus, daß bei einem Join die Reihenfolge der Relationen keine Rolle spielt
2. \bowtie ist assoziativ
bedeutet, daß die Reihenfolge der Abarbeitung von Joins keine Rolle spielt
3. $\pi_x(\pi_y(R_1)) \Leftrightarrow \pi_x(R_1)$ für $X \subseteq Y$
fasst zwei Projektionen zusammen
4. P_1, P_2, \dots Selektionsprädikate $\sigma_{P_1}(\sigma_{P_2}(R)) \Leftrightarrow \sigma_{P_1 \wedge P_2}(R) \Leftrightarrow \sigma_{P_2}(\sigma_{P_1}(R))$
Reihenfolge von Selektionen kann vertauscht werden
5. $\sigma_{P_1}(\pi_X(R)) \Leftrightarrow \pi_X(\sigma_{P_1}(R))$ falls $\text{attr}(P_1) \subseteq X$
Vertauschen von σ mit π : Wenn die Auswahlbedingung X nur Attribute P_1 in der Projektionsliste enthält, können die beiden Operationen vertauscht werden
6. $\sigma_P(R_1 \bowtie R_2) \Leftrightarrow \sigma_P(R_1) \bowtie R_2$, falls $\text{attr}(P) \subseteq R_1$
Selektion und Verbund sind austauschbar, falls die Selektionsattribute alle aus einer der beiden beteiligten Relationen stammen. Falls das Selektionsprädikat so aufgesplittet werden kann, dass in $P = P_1 \wedge P_2$ die beiden Teile P_1 und P_2 passende Attribute haben, so gilt
 $\sigma_P(R_1 \bowtie R_2) \Leftrightarrow \sigma_{P_1}(R_1) \bowtie \sigma_{P_2}(R_2)$ falls $\text{attr}(P_1) \subseteq R_1$ und $\text{attr}(P_2) \subseteq R_2$
7. $\sigma_P(R_1 \cup R_2) \Leftrightarrow \sigma_P(R_1) \cup \sigma_P(R_2)$
Selektion und \cup sind vertauschbar
8. $\sigma_P(R_1 \setminus R_2) \Leftrightarrow \sigma_P(R_1) \setminus \sigma_P(R_2) \Leftrightarrow \sigma_P(R_1) \setminus (R_2)$
Selektion und \setminus sind vertauschbar
9. ...

Die Reihenfolge ist:

- Komplexe Selektionsprädikate werden aufgelöst (Regel 4)
- mittels der Regeln 5-8 werden Selektionen möglichst weit in Richtung der Blätter verschoben. Gegebenenfalls müssen Selektionen gemäß Regel 4 vertauscht werden

- mit Regel 11 werden kartesische Produkte mit anschließender Selektion in Joins verwandelt
- die Regeln 3, 5, 9 und 10 ermöglichen es, die Projektionen ebenfalls in Richtung Blätter zu verschieben
- die Einzelschritte werden in dieser Reihenfolge so lange ausgeführt, bis keine Ersetzungen mehr möglich sind

e) Wie viele verschiedene JOIN-Ordnungen gibt es für eine Anfrage, die 10 Relationen vereinigt?

Immer $n-1$, daher 9.

f) Warum spricht man bei der logischen Optimierung von einem heuristischen Verfahren?

Weil Operationen verschoben werden, um kleinere Zwischenergebnisse zu erhalten und Redundanzen zu erkennen.

(*) Aufgabe 25 (Logische Anfrageoptimierung):

Für diese Aufgabe (und eine weitere Aufgabe auf dem kommenden Übungszettel) ist das Datenbankschema in der unten stehenden Abbildung₁ gegeben. Zu diesem Schema betrachten wir im Folgenden 3 Anfragen:

1. SELECT VNAME, NNAME, ADRESSE
FROM ANGESTELLTER, ABTEILUNG
WHERE ANAME = 'Research' AND ABTNUMMER = ANR;
2. SELECT A.VNAME, A.NNAME, S.VNAME, S.NNAME
FROM ANGESTELLTER AS A, ANGESTELLTER AS S
WHERE A.SUPERSSN = S.SSN;
3. SELECT VNAME, NNAME, 1.1*GEHALT
FROM ANGESTELLTER, ARBEITET_AN, PROJEKT
WHERE SSN = ESSN AND PNR = PNUMMER AND PNAME = 'ProduktX';

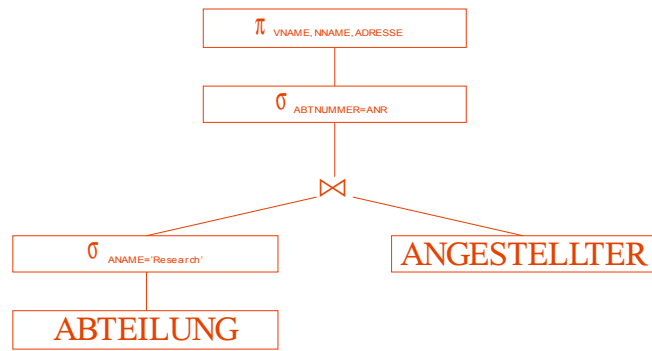
a) Beschreiben Sie zu jeder der drei Anfragen in ganzen Sätzen, was mit dieser Anfrage berechnet werden soll. Es geht weniger um eine wörtliche Übersetzung der Anfrage, sondern mehr um eine sinnvolle Interpretation!

1. Es sollen Namen und Adressen aller Angestellten aus der Abteilung „Research“ ausgegeben werden
2. Hier wird eine Gegenüberstellung zwischen Mitarbeitern und ihren Abteilungsleitern vorgenommen
3. Name und Gehalt * 1.1 aller Angestellten, die am Projekt „ProduktX“ arbeiten, sollen ausgegeben werden

b) Geben Sie zu jeder der Anfragen einen äquivalenten Ausdruck in der Relationenalgebra an.

1. $\pi_{VNAME, NNAME, ADRESSE}(\sigma_{ABTNUMMER=ANR}(ANGESTELLTER \bowtie (\sigma_{ANAME='Research'}(ABTEILUNG))))$
2. $\pi_{VNAME, NNAME}(\sigma_{SSN=SUPERSSN}(\pi_{VNAME, NNAME}(ANGESTELLTER \bowtie ANGESTELLTER)))$
3. $\pi_{VNAME, NNAME, 1.1*GEHALT}(\sigma_{SSN=ESSN}(ANGESTELLTER \bowtie (\sigma_{PNUMMER=PNR}(ARBEITET_AN \bowtie \sigma_{PNAME='ProduktX'}(PROJEKT))))$

c) Verwandeln Sie die Anfragen in je einen initialen Anfragebaum, indem Sie die kanonische (naive) Übersetzung in einen Operatorbaum verwenden.



d) Wenden Sie die Optimierungsregeln aus der Vorlesung auf das Ergebnis von Teil c) an, um für jede der drei Anfragen einen äquivalenten, logisch optimierten Anfragebaum zu erhalten.

e) Vergleichen Sie die Ergebnisse von Teil c) und d).

Abbildung 8.1(a): Beispiel einer SQL2-Datendefinition; SQL2-Anweisungen definieren das FIRMA-Schema aus Abbildung 7.7.

CREATE TABLE ANGESTELLTER

```
( VNAME          VARCHAR(15)          NOT NULL ,
  INITIAL        CHAR ,
  NNAME         VARCHAR(15)          NOT NULL ,
  SSN           CHAR(9)             NOT NULL ,
  GDATUM        DATE ,
  ADRESSE       VARCHAR(30) ,
  GESCHLECHT    CHAR ,
  GEHALT        DECIMAL(10,2) ,
  SUPERSSN      CHAR(9) ,
  ANR           INT                 NOT NULL ,
  PRIMARY KEY (SSN) ,
  FOREIGN KEY (SUPERSSN) REFERENCES ANGESTELLTER(SSN) ,
  FOREIGN KEY (ANR) REFERENCES ABTEILUNG(ABTNUMMER) );
```

CREATE TABLE ABTEILUNG

```
( ABTNUMMER      INT                 NOT NULL ,
  NAME          VARCHAR(30)          NOT NULL ,
  MGRSSN        CHAR(9)             NOT NULL ,
  MGRANFANGSDATUM DATE ,
  PRIMARY KEY (ABTNUMMER) ,
  UNIQUE (NAME) ,
  FOREIGN KEY (MGRSSN) REFERENCES ANGESTELLTER(SSN) );
```

CREATE TABLE ABT_STANDORTE

```
( ABTNUMMER      INT                 NOT NULL ,
  STANDORT      VARCHAR(30)          NOT NULL ,
  PRIMARY KEY (ABTNUMMER, STANDORT) ,
  FOREIGN KEY (ABTNUMMER) REFERENCES ABTEILUNG(ABTNUMMER) );
```

CREATE TABLE PROJEKT

```
( PNAME          VARCHAR(30)          NOT NULL ,
  PNUMMER       INT                 NOT NULL ,
  PSTARTDATUM   DATE ,
  ABTNR         INT                 NOT NULL ,
  PRIMARY KEY (PNUMMER) ,
  UNIQUE (PNAME) ,
  FOREIGN KEY (ABTNR) REFERENCES ABTEILUNG(ABTNUMMER) );
```

CREATE TABLE ARBEITET_AN

```
( ESSN          CHAR(9)             NOT NULL ,
  PNR           INT                 NOT NULL ,
  STUNDEN       DECIMAL(3,1)        NOT NULL ,
  PRIMARY KEY (ESSN, PNR) ,
  FOREIGN KEY (ESSN) REFERENCES ANGESTELLTER(SSN) ,
  FOREIGN KEY (PNR) REFERENCES PROJEKT(PNUMMER) );
```

CREATE TABLE ANGEHÖRIGER

```
( ESSN          CHAR(9)             NOT NULL ,
  ANGEHÖRIGER_NAME VARCHAR(15)        NOT NULL ,
  GESCHLECHT    CHAR ,
  GDATUM        DATE ,
  GRAD          VARCHAR(8) ,
  PRIMARY KEY (ESSN, ANGEHÖRIGER_NAME) ,
```