Tutorium 6
Anfrageoptimierung
Big Data Engineering

Prof. Dr. Jens Dittrich

bigdata.uni-saarland.de

13./14. Juni 2022

Wiederholung - Frage 1

Frage

Was versteht man unter kostenbasierter Optimierung?

Wiederholung - Frage 1

Frage

Was versteht man unter kostenbasierter Optimierung?

Lösung

Ein Plan wird nicht nur durch strikte Regeln optimiert, sondern es werden mehrere Pläne aufgezählt und zusätzlich die Kosten anhand verschiedener Kostenfunktionen (je nach Modell und Implementierung) abgeschätzt. Anschließend wird der Plan mit den geringsten Kosten ausgeführt.

Wiederholung - Frage 2

Frage

Was versteht man unter Joinselektivität?

Wiederholung - Frage 2

Frage

Was versteht man unter Joinselektivität?

Lösung

Die Joinselektivität ist das Verhältnis der Größe des Join-Ergebnisses zur Größe des kartesischen Produktes der Eingaberelationen:

$$sel_{R \bowtie S} = \frac{|R \bowtie S|}{|R \times S|} \leq 1.$$

In anderen Worten: Die Joinselektivität bezeichnet die Selektivität des Joinprädikats.

Wiederholung - Frage 3

Frage

Was versteht man unter einem Joingraphen?

Wiederholung - Frage 3

Frage

Was versteht man unter einem Joingraphen?

Lösung

Mit einem Joingraphen können wir die Joins zwischen verschiedenen Relationen in einer Anfrage visualisieren. Ein Joingraph hat dabei einen Knoten für jede Eingaberelation und eine Kante für jedes Joinpädikat. Zusätzlich werden Knoten, auf denen ein Filterprädikat existiert, mit diesem annotiert und alle Kanten mit dem entsprechenden Joinprädikat annotiert.

Wiederholung - Frage 4

Frage

Inwiefern helfen Joingraphen bei der Bestimmung der optimalen Joinreihenfolge?

Wiederholung - Frage 4

Frage

Inwiefern helfen Joingraphen bei der Bestimmung der optimalen Joinreihenfolge?

Lösung

Betrachten wir alle möglichen Joinreihenfolgen einer Anfrage, so kommt es oft vor, dass zwischen zwei oder mehr Relationen gar kein Joinprädikat existiert. Effektiv wird dann das kartesische Produkt berechnet, was aufgrund der Ergebnisgröße in der Praxis sehr teuer ist. Mithilfe des Joingraphen können wir nun nur Joinreihenfolgen ohne kartesische Produkt aufzählen.

Aufgabe 1.1

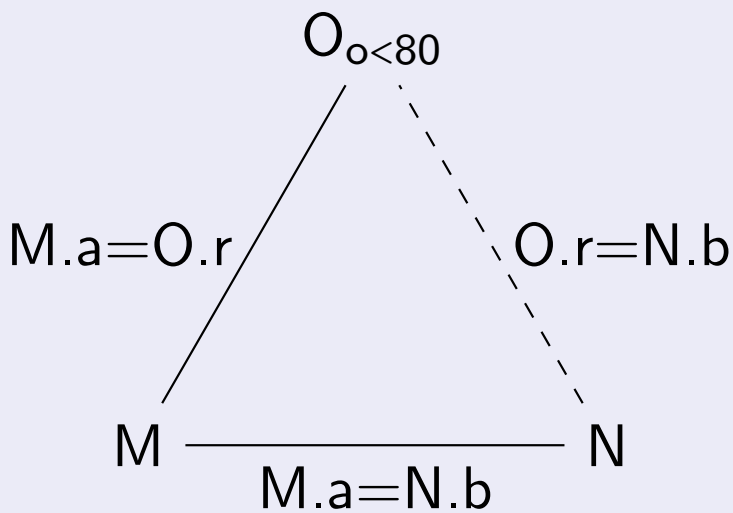
Frage

Zeichnen Sie für die folgende SQL Anfrage den zugehörigen Joingraphen. Können Sie zusätzliche Joinprädikate ableiten, so zeichnen Sie auch diese mit gestrichelten Kanten ein. Beschriften Sie außerdem alle Kanten mit dem jeweiligen Joinprädikat.

```
SELECT M.b, N.r, O.t
FROM   M, N, O
WHERE  M.a = N.b AND O.o < 80 AND O.r = M.a;
```

Aufgabe 1.1

Lösung



Aufgabe 1.2

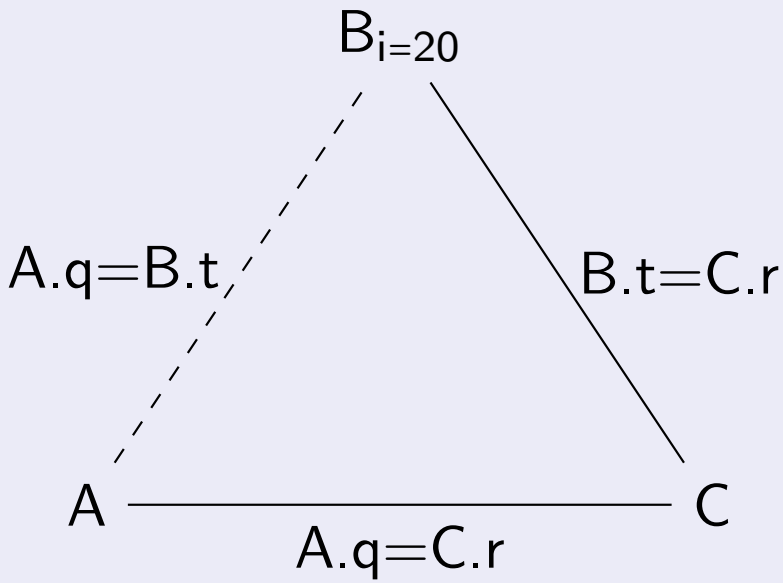
Frage

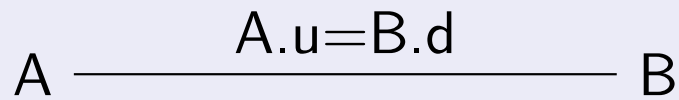
Zeichnen Sie für die folgende SQL Anfrage den zugehörigen Joingraphen. Können Sie zusätzliche Joinprädikate ableiten, so zeichnen Sie auch diese mit gestrichelten Kanten ein. Beschriften Sie außerdem alle Kanten mit dem jeweiligen Joinprädikat.

```
SELECT A.b, B.c
FROM   A, B, C
WHERE  A.q = C.r AND C.r = B.t AND B.i = 20
    UNION
SELECT A.b, B.c
FROM   A JOIN B ON B.d = A.u
```

Aufgabe 1.2

Lösung





Die beiden Teilgraphen sind nicht verbunden, da die beiden Anfragen unabhängig sind und lediglich durch das Schlüsselwort `UNION` vereinigt werden.

Aufgabe 1.3

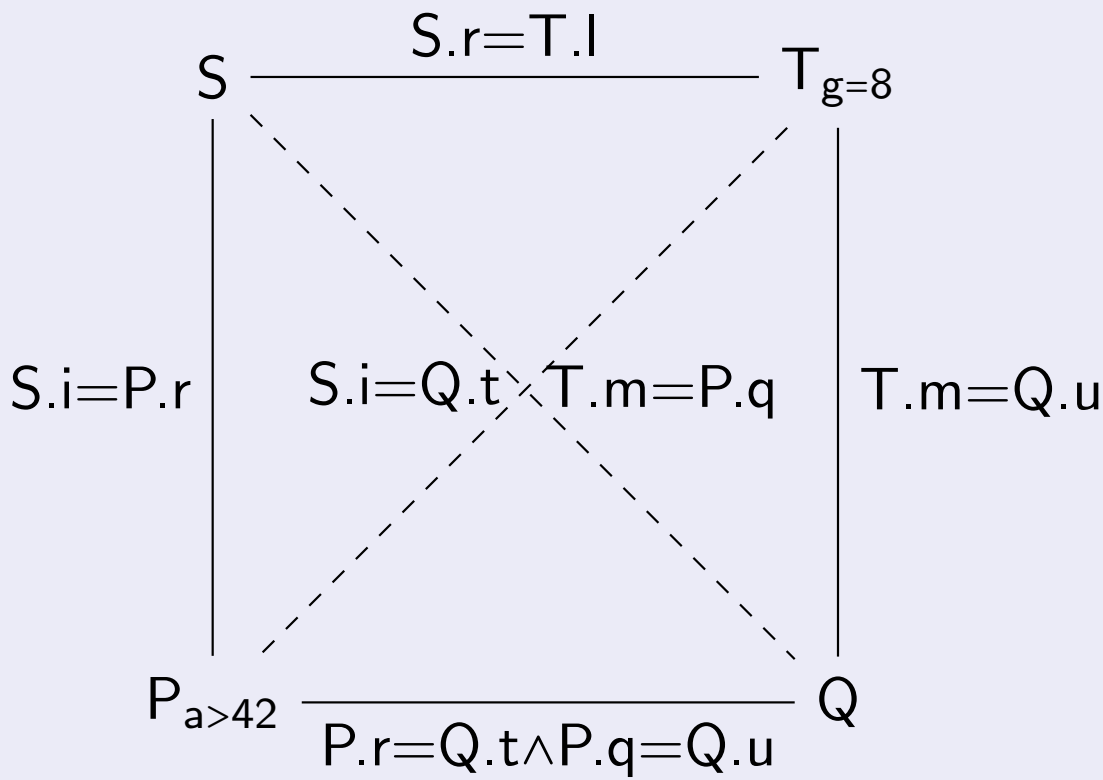
Frage

Zeichnen Sie für die folgende SQL Anfrage den zugehörigen Joingraphen. Können Sie zusätzliche Joinprädikate ableiten, so zeichnen Sie auch diese mit gestrichelten Kanten ein. Beschriften Sie außerdem alle Kanten mit dem jeweiligen Joinprädikat.

```
SELECT *
FROM   (SELECT P.r AS t, Q.u AS w
        FROM   P, Q
        WHERE  P.r = Q.t AND Q.u = P.q
               AND P.a > 42
       ) AS R, S, T
WHERE  S.i = R.t AND T.m = R.w AND S.r = T.l
       AND T.g = 8;
```

Aufgabe 1.3

Lösung



Aufgabe 2

Frage

Betrachten Sie folgende SQL Anfrage.

```
SELECT *
FROM M, N, O
WHERE M.p = O.a AND M.q = N.c;
```

Gegeben seien die folgenden Ergebnisgrößen für die genannten Teilprobleme.

$$|\{M\}| = 200 \quad |\{N\}| = 100 \quad |\{O\}| = 50$$
$$|\{M,N\}| = 200 \quad |\{M,O\}| = 200 \quad |\{N,O\}| = 5.000$$
$$|\{M,N,O\}| = 200$$

Bestimmen sie die optimale Joinreihenfolge. Ihnen steht ausschließlich ein einfacher hash-basierter Join mit folgender Kostenfunktion zur Verfügung:

$$C_{\text{HashJoin}}(R \bowtie S) = |R| + |S|$$

Für kartesische Produkte müssen Sie hingegen folgende Kostenfunktion verwenden:

$$C(R \times S) = |R| \cdot |S|$$

Aufgabe 2

Lösung

Teilplan	Kosten	Ergebnisgröße
M	0	200
N	0	100
O	0	50
$M \bowtie N$	$0 + 0 + 200 + 100 = 300$	200
$M \bowtie O$	$0 + 0 + 200 + 50 = 250$	200
$N \times O$	$0 + 0 + 100 \cdot 50 = 5.000$	5.000
$(M \bowtie N) \bowtie O$	$300 + 0 + 200 + 50 = 550$	200
$(M \bowtie O) \bowtie N$	$250 + 0 + 200 + 100 = 550$	200
$(N \times O) \bowtie M$	$5.000 + 0 + 5.000 + 200 = 10.200$	200

Die optimale Joinreihenfolge ist $(M \bowtie N) \bowtie O$ oder $(M \bowtie O) \bowtie N$.
Anmerkung: Die Pläne, die ein kartesisches Produkt enthalten, hätte man gar nicht betrachten müssen, da diese zu teuer sind.

Aufgabe 3

Frage

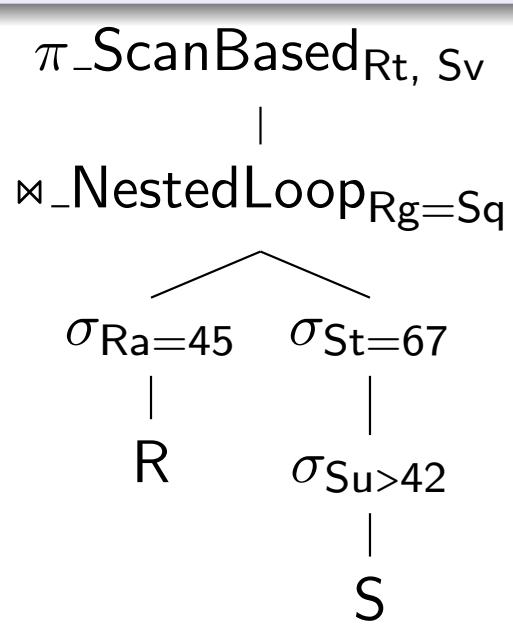
Optimieren Sie die verbliebenen logischen Operatoren in folgendem "hybriden" Anfragebaum kostenbasiert und übersetzen Sie ihn in einen physischen Plan. Es gelten folgende Kosten für einen Scan bzw. Indexzugriff, Tabellengrößen und Selektivitäten:

$$scan(T) = |\ T\ |$$

$$index(T, p) = log_2(|\ T\ |) + 25 \cdot sel(p) \cdot |\ T\ |$$

$$|\ \mathrm{R}\ | = 100.000 \quad sel(\mathrm{Ra{=}45}) = 0,03$$

$$|\ \mathrm{S}\ | = 200.000 \quad sel(\mathrm{St{=}67}) = 0,3 \quad sel(\mathrm{Su{>}42}) = 0,05$$



Aufgabe 3

Lösung

$$
\begin{aligned}
\mathit{scan}(\mathsf{R}) &= 100.000 \\
\mathit{index}(\mathsf{R}, \mathsf{Ra}{=}45) &= \mathit{log}_2(100.000) + 25 \cdot 0,03 \cdot 100.000 \approx 75.017 \\
\mathit{scan}(\mathsf{S}) &= 200.000 \\
\mathit{index}(\mathsf{S}, \mathsf{St}{=}67) &= \mathit{log}_2(200.000) + 25 \cdot 0,3 \cdot 200.000 \approx 1,5 \cdot 10^6 \\
\mathit{index}(\mathsf{S}, \mathsf{Su}{>}42) &= \mathit{log}_2(200.000) + 25 \cdot 0,05 \cdot 200.000 \approx 250018
\end{aligned}
$$