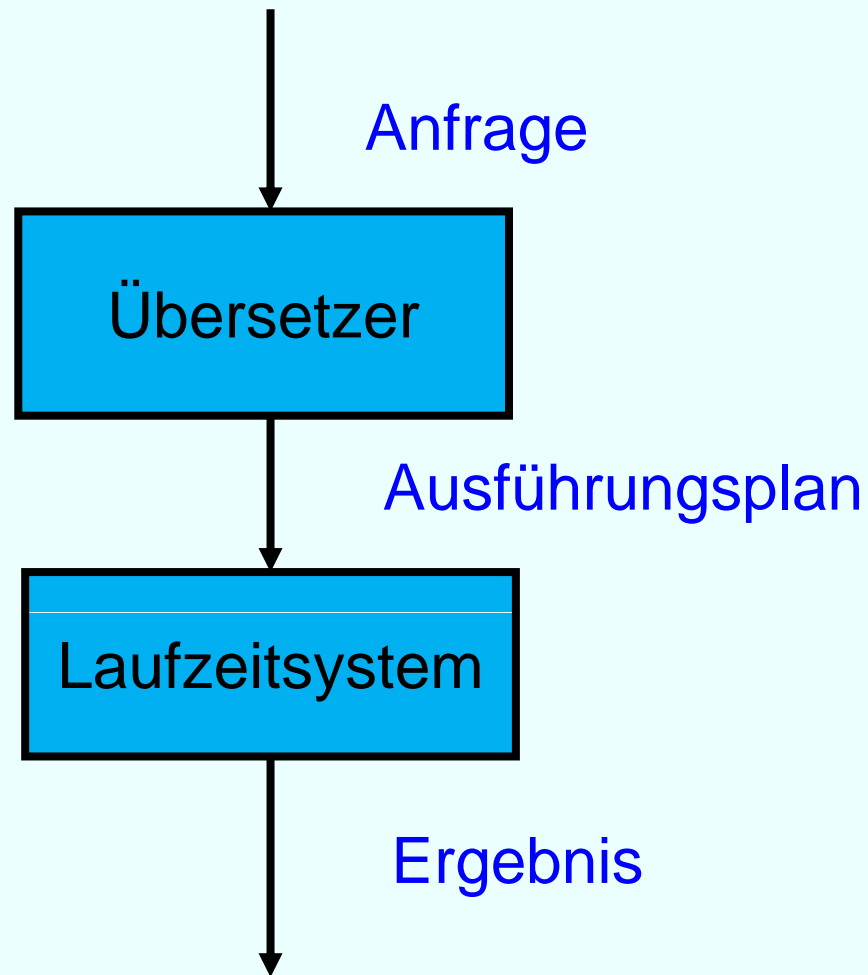


Anfragebearbeitung



Übersetzung

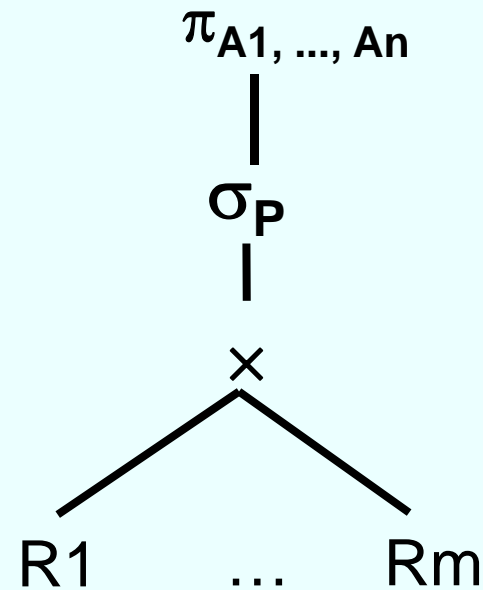
- SQL ist deklarativ, Übersetzung für Laufzeitsystem in etwas prozedurales
- DBMS übersetzt SQL in eine interne Darstellung
- weit verbreiteter Ansatz ist Übersetzung in eine relationale Algebra

Kanonische Übersetzung

- Standardübersetzung von SQL in relationale Algebra
- Algebra-Ausdrücke werden oft graphisch repräsentiert

- Beispiel

select A_1, \dots, A_n
from R_1, \dots, R_m
where p



Optimierung

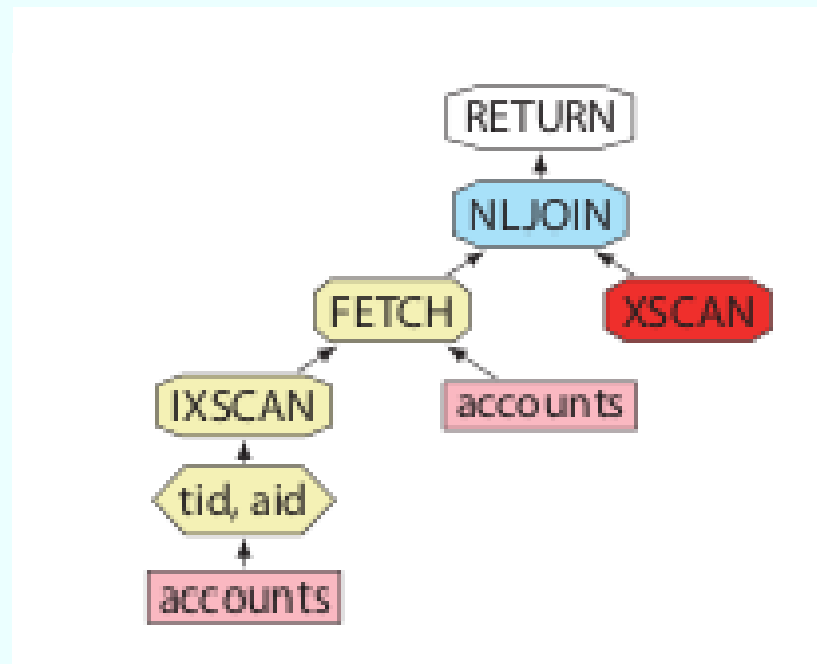
- Kanonischer Plan nicht effizient, z.B. Kreuzprodukt
- DBMS besitzt Optimierer zur Überführung des Plans in eine effiziente Form
- Finden eines optimalen Plans sehr schwieriges Problem:
immer noch Gegenstand aktueller Forschung

Optimierung(2)

- Benutzer ↔ Anfrageoptimierung?
- Benutzer können
 - generierten Plan einsehen
 - generierten Plan analysieren
 - (gegebenenfalls Anfrage umbauen oder DBMS Hinweise zur Ausführung geben)

Visualisierung von Plänen

PostgreSQL:



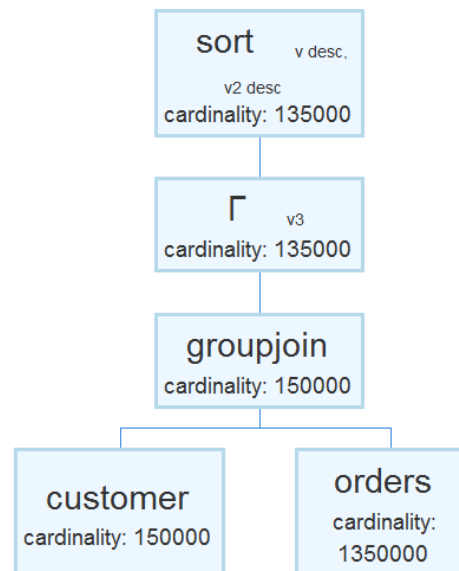
Visualisierung von Plänen

Hyper:

Query Plan

Show Information: [All](#) / [None](#)

Attributes Cardinalities Criteria Predicates Restrictions Residuals Outputs



Legend

Anfrageoptimierung 1x1

- Anfrageoptimierung ist kostenbasiert
- Abschätzung der Kosten mit Hilfe von Kostenmodellen und Statistiken
- Anwendung von Heuristiken
alle möglichen Pläne anzuschauen viel zu teuer
- Zwei Ebenen der Optimierung:
 - Logische Ebene
 - Physische Ebene

Logische Ebene

- Ausgangspunkt: relationaler Algebra-Ausdruck entstanden nach kanonischer Übersetzung
- Optimierung: Transformation in äquivalente Ausdrücke (mit schnellerer Ausführung)
- Ziel der Umformungen: Ausgaben (Ergebnisse) der einzelnen Operatoren möglichst klein

Logische Ebene(2)

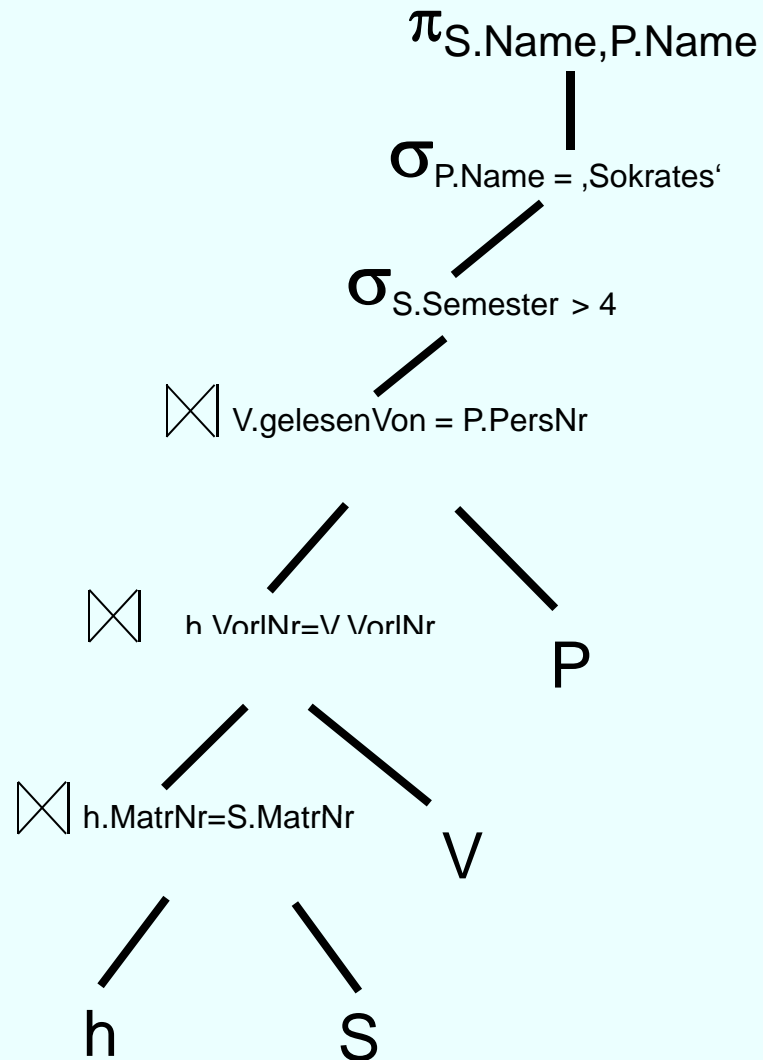
Grundlegende Techniken - Regeln:

- Aufbrechen von Selektionen
- Verschieben von Selektionen nach "unten" im Plan
- Zusammenfassen von Selektionen und Kreuzprodukten zu Joins
- Bestimmung der Joinreihenfolge
- Einfügen von Projektionen
- Verschieben von Projektionen nach "unten" im Plan

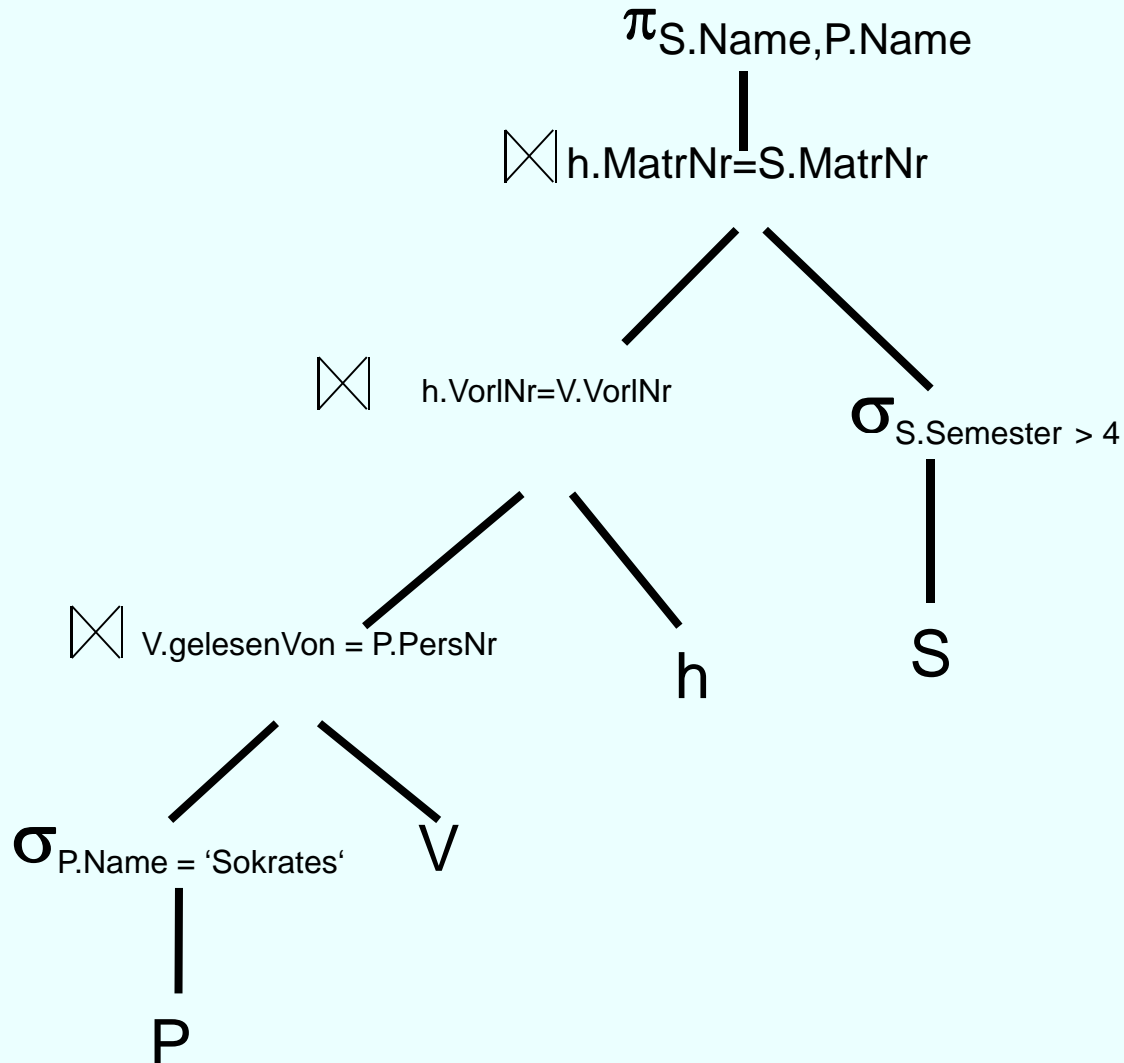
Beispielanfrage

```
select S.Name, P.Name
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 S.Semester > 4
and P.Name = 'Sokrates';
```

Anfrageplan



Optimierter Anfrageplan



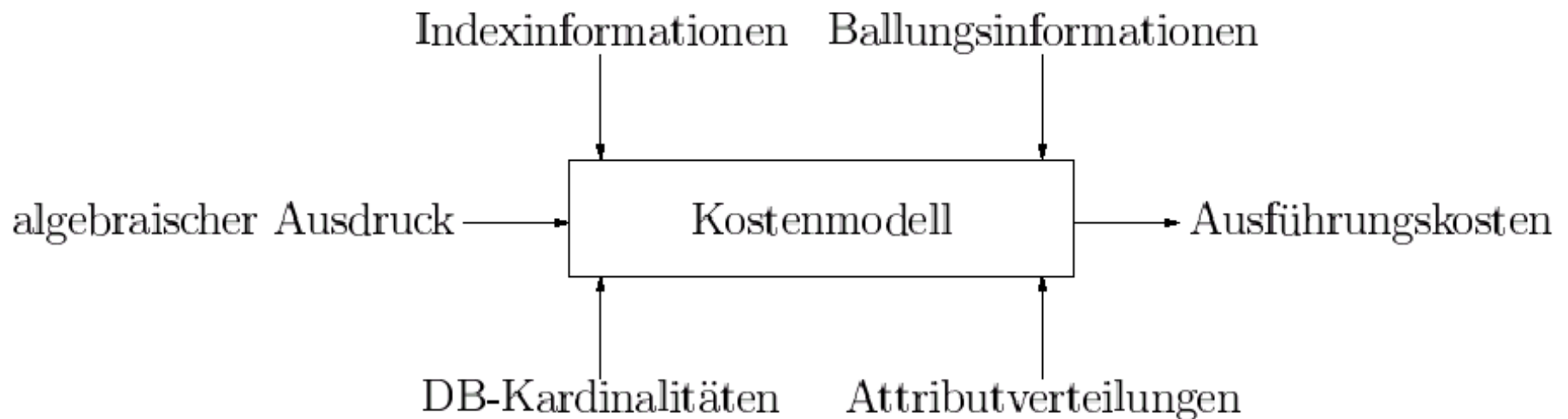
Physische Optimierung

- Unterscheidung logische und physische Algebra-Operatoren
- physische Algebra-Operatoren stellen Realisierung der logischen dar
- mehrere physische Operatoren für einen logischen Operator möglich
- Optimierung auf der physischen Ebene bedeutet:
 - einen dieser Operatoren auszuwählen
 - zu entscheiden, ob Indexe benutzt werden sollen
 - Zwischenergebnisse zu materialisieren, etc.

Implementierung Operatoren

- Selektion:
 - Scan
 - Indexscan
- Join:
 - Nested-Loop-Join
 - (Sort-)Merge-Join
 - Index-Join
 - Hash-Join

Kostenmodelle



Kostenabschätzungen

- **Selektivitäten**
Anteil der qualifizierenden Tupel einer Operation
- **Abschätzung der Selektivität durch**
 - Formeln
 - Stichprobenverfahren
- **Selektionskosten**
- **Joinkosten**
- **Joinreihenfolge**

→ hierfür braucht man Statistiken

Wann Statistiken sammeln?

Wann Statistiken fortschreiben? (Auszug aus [IBM DB2 Manual](#))
unter anderem:

- Nachdem Daten in eine Tabelle geladen und geeignete Indizes erstellt wurden.
- Nachdem ein neuer Index für eine Tabelle erstellt wurde.
- Nachdem eine Tabelle mit dem Dienstprogramm REORG reorganisiert wurde.
- Nachdem eine Tabelle und die zugehörigen Indizes durch UPDATE-, INSERT- oder DELETE-Operationen in erheblichem Umfang geändert wurden.
- Vor dem Binden von Anwendungsprogrammen, deren Leistung von kritischer Bedeutung ist.
- Nachdem der Befehl REDISTRIBUTE DATABASE PARTITION GROUP ausgeführt wurde.

Welche Statistiken gibt es?

(Auszug aus [IBM DB2 Manual](#))

SYSCAT: read-only view – Systemkatalog

SYSSTAT: änderbarer view - Statistikdaten

- Tabellenstatistiken (SYSCAT/SYSSTAT.**TABLES**)
- Spaltenstatistiken (SYSCAT/SYSSTAT.**COLUMNS**)
- Statistiken für Spaltengruppen (SYSCAT/SYSSTAT.**COLGROUPS**)
- Verteilungsstatistiken für Spaltengruppen (SYSCAT/SYSSTAT.**COLGROUPDIST**)
- Verteilungsstatistiken für Spaltengruppen (SYSCAT/SYSSTAT.**COLGROUPDISTCOUNTS**)
- Indexstatistiken (SYSCAT/SYSSTAT.**INDEXES**)
- Spaltenverteilungsstatistiken (SYSCAT/SYSSTAT.**COLDIST**)

Zusammenfassung

- Anfragebearbeitung und -optimierung sind wichtige Aufgaben eines DBMS
- auch Benutzer sollten Ahnung davon haben, da Entwurfsentscheidungen und Anfrageformulierung einen Einfluss auf die Performanz eines DBMS haben