

Lernzettel

Relationale Algebra: Operatoren, Ausdrücke von Abfragen und Optimierung

Universität: Technische Universität Berlin
Kurs/Modul: Informationssysteme und Datenanalyse
Erstellungsdatum: September 19, 2025



Zielorientierte Lerninhalte, kostenlos!
Entdecke zugeschnittene Materialien für deine Kurse:

<https://study.AllWeCanLearn.com>

Informationssysteme und Datenanalyse

Lernzettel: Relationale Algebra: Operatoren, Ausdrücke von Abfragen und Optimierung

(1) Grundlagen der relationalen Algebra Die relationale Algebra ist eine formale Sprache zur Beschreibung von Abfragen über Relationen. Wichtige Konzepte:

- Relationen sind Tabellen aus Tupeln, deren Spalten Attribute tragen.
- Operatoren kombinieren oder transformieren Relationen zu neuen Relationen.
- Ausdrücke werden von innen nach außen ausgewertet; der äußere Ausdruck liefert das Abfrageergebnis.

(2) Operatoren der relationalen Algebra Wichtige Grundoperatoren (Notation: R, S bezeichnen Relationen):

- Selektion (Auswahl): $\sigma_{\text{Bedingung}}(R)$
- Projektion (Spaltenauswahl): $\pi_{\text{Attribute}}(R)$
- Umbenennung: $\rho_{\text{NeueNamen}}(R)$
- Kartesisches Produkt: $R \times S$
- Vereinigung: $R \cup S$
- Schnittmenge: $R \cap S$
- Differenz: $R - S$
- Natural Join: $R \bowtie S$
- Theta-Join: $R \bowtie_{\theta} S$
- Division: $R \div S$

(3) Typische Operatoren im Detail

- Selektion: $\sigma_{\text{Bedingung}}(R)$ filtert Tupel von R nach der Bedingung.
- Projektion: $\pi_{\text{Attr1}, \text{Attr2}}(R)$ reduziert die Spalten von R auf die angegebenen Attribute.
- Umbenennung: $\rho_{A/B}(R)$ benennt Attribute oder ganze Relationen um.
- Join-Arten:
 - Natürlicher Join: $R \bowtie S$ verbindet auf gemeinsamen Attributen.
 - Theta-Join: $R \bowtie_{\theta} S$ verbindet nach einer Bedingung θ .
- Division: $R \div S$ liefert Tupel x , für die in jeder Tupelbildung von S ein passendes Tupel existiert.

(4) Ausdrücke von Abfragen – Beispiele Beispiel 1: Selektion gefolgt von Projektion

$$\pi_{\text{Name,Abteilung}}(\sigma_{\text{Gehalt} > 50000}(M))$$

Beispiel 2: Join zweier Relationen mit Selektion Sei M (Mitarbeiter) mit Attributen Name, Gehalt, AbteilID und A (Abteilungen) mit ID, Abteilung.

$$\pi_{\text{Name,Abteilung}}(\sigma_{\text{Gehalt} > 50000}(M \bowtie_{M.=A.} A))$$

Beispiel 3: Übersetzung eines SQL-Statements in RA SQL: SELECT M.Name, A.Abteilung FROM Mitarbeiter M JOIN Abteilungen A ON M.AbteilID = A.ID WHERE M.Gehalt > 50000; Relationale Algebra:

$$\pi_{\text{Name,Abteilung}}(\sigma_{\text{Gehalt} > 50000}(M \bowtie_{M.=A.} A))$$

(5) Optimierung von Abfragen (theoretisch) Grundideen der Optimierung:

- Pushdown von Selektionen: $\sigma_{\text{cond}}(R \bowtie S)$ möglichst früh anwenden, um weniger Tupel zu verarbeiten.
- Projektionen nah an der Datennutzung anwenden, um unnötige Spalten zu vermeiden.
- Join-Reihenfolge optimieren: Radius der Join-Operatoren so wählen, dass große Relationen später verbunden werden.
- Umformen mit Äquivalenzregeln:

$$(R \bowtie S) \bowtie T \equiv R \bowtie (S \bowtie T)$$

sowie Verschieben von Selektionen über Joins, sofern sinnvoll.

- Wahl geeigneter Join-Algorithmen (in Umsetzungsebene): Nested-Loop, Hash-Join, Sort-Mmerge, je nach Größe und Verteilung der Daten.
- Nutzung von Indizes (in der Implementierung): Indizes beschleunigen Selektionen und Join-Bedingungen.

(6) Hinweise zur Umsetzung in SQL und Zuordnung zur Optimierung - RA-Ausdrücke dienen als Grundlage; SQL-Abfragen werden oft in RA-Graphen transformiert, um Optimierer zu unterstützen. - Typische Umsetzungen: Selektion vor Join, Projektion nur der benötigten Spalten, Vermeidung unnötiger Duplikate, Nutzung geeigneter Joins (z. B. Hash-Join bei großen Tabellen).

(7) Zusammenfassung - Relationale Algebra liefert formale Operatoren zur Beschreibung von Abfragen. - Operatoren wie $\cup$, $\cap$, $\times$, σ , π , $\bowtie$ ermöglichen kompakte Abfragenformeln. - Optimierung zielt darauf ab, die Größe der Zwischenergebnisse zu minimieren und die Ausführung zu beschleunigen, z. B. durch Pushdown-Regeln und geschickte Join-Reihenfolgen.