



Informations- und Wissensmanagement

Kapitel 5: **Anfrageoptimierung und relationale Algebra**



Vorgehensmodell der Anfrageevaluierung

- Parsen der Anfrage (SQL)
- Transformation in eine Standardform
(**Relationenalgebra**)
→ **Logische Optimierung**
- Transformation in alternative Zugriffspläne,
→ **Physische Optimierung**
- Ausführung des gewählten Zugriffsplans

[Einleitung](#)

[Projektion](#)

[Selektion](#)

[Join](#)

[Mengen-
op.](#)

[Vollst.keit](#)

Maßnahmen zur log. Optimierung:

- Ändern der Operator-Reihenfolge
- Zusammenfassen von Operatoren
- Unnötigen Operatoren entfernen



Algebra – Motivation

- deklarative Anfragesprachen (z.B. SQL):
 - Datenunabhängigkeit
 - vom DBMS optimierbar (keine Aussagen, wie Anfrage bearbeitet werden soll)
- Wir betrachten in diesem Kapitel relationale Algebra:
 - einfacher,
 - erlaubt Aussagen zur Ausführungsreihenfolge.
→ ***Basis für die Anfrageoptimierung***

[Einleitung](#)

[Projektion](#)

[Selektion](#)

[Join](#)

[Mengen-
op.](#)

[Vollst.keit](#)



Anfragealgebren

- Mathematik
 - Algebra definiert durch *Wertebereich* und auf diesem definierte *Operatoren*.
- Für Datenbankabfragen
 - Inhalte der Datenbank sind Werte, und
 - Operatoren definieren Funktionen zum Berechnen von Anfrageergebnissen.

[Einleitung](#)

[Projektion](#)

[Selektion](#)

[Join](#)

[Mengen-
op.](#)

[Vollst.keit](#)

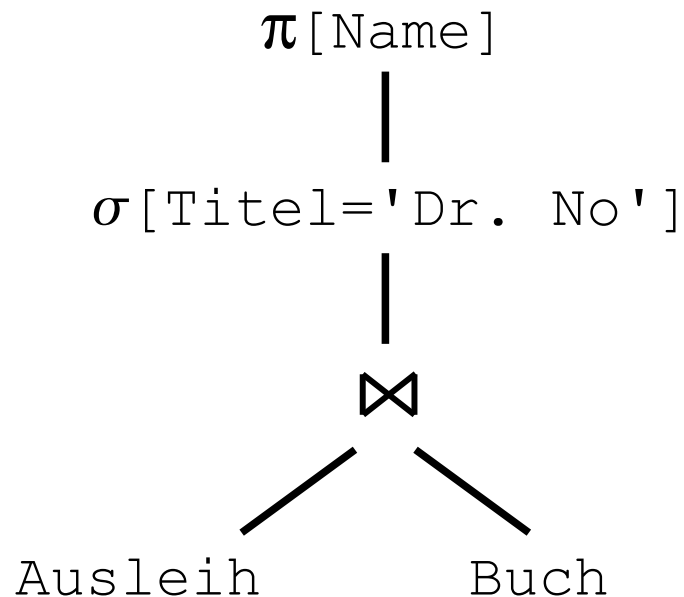


Darstellung

- Als Term

$\pi[\text{Name}] (\sigma[\text{Titel} = \text{'Dr. No'}] (\text{Ausleih} \bowtie \text{Buch}))$

- Als Operatorbaum (oft anschaulicher)



Laufendes Beispiel

Ausleih

| INV.NR | NAME |
|--------|--------|
| 4711 | Meyer |
| 1201 | Schulz |
| 0007 | Müller |
| 4712 | Meyer |

Buch

| INV.NR | TITEL | ISBN | AUTOR |
|--------|--------------|-------|---------|
| 0007 | Dr. No | 3-324 | Fleming |
| 1201 | Objektbanken | 3-111 | Heuer |
| 4711 | Datenbanken | 3-345 | Vossen |
| 4712 | Datenbanken | 3-345 | Ullman |
| 4717 | PASCAL | 3-989 | Wirth |

[Einleitung](#)

[Projektion](#)

[Selektion](#)

[Join](#)

[Mengen-
op.](#)

[Vollst.keit](#)



Projektion (1)

- Beispiel 1: Projektion auf ein Attribut

$\pi[\text{Name}] (\text{Ausleih})$

ergibt als Ergebnisrelation
eine **Menge**:

| NAME |
|--------|
| Meyer |
| Schulz |
| Müller |

- zum Vergleich:
select Name **from** Ausleih
ergibt als Ergebnisrelation
eine Multimenge:

| NAME |
|--------|
| Meyer |
| Schulz |
| Müller |
| Meyer |

Einleitung
Projektion
Selektion
Join
Mengen-
op.
Vollst.keit

Projektion (2)

- Beispiel 2: Projektion auf Attributmenge

$\pi[\text{Invnr}, \text{ISBN}] (\text{Buch})$

ergibt

| INV.NR | ISBN |
|--------|-------|
| 0007 | 3-324 |
| 1201 | 3-111 |
| 4711 | 3-345 |
| 4712 | 3-345 |
| 4717 | 3-989 |

[Einleitung](#)

[Projektion](#)

[Selektion](#)

[Join](#)

[Mengen-
op.](#)

[Vollst.keit](#)



Projektion (3)

- Optimierungsregel:
Bei vielen Projektionen hintereinander reicht die **zuletzt ausgeführte** auch allein

$$\pi[\text{Invnr}] (\pi[\text{Invnr}, \text{ISBN}] (\text{Buch}))$$

ergibt optimiert

$$\pi[\text{Invnr}] (\text{Buch})$$

- Grafische Darstellung oft hilfreich.
- Wieso sind derartige Optimierungen wichtig?
 - Für Anfrageausführung – Alternativen sind zwar äquivalent, aber unterschiedlich teuer in der Ausführung.
 - Optimierung nutzt Äquivalenz aus, um günstige Ausführung zu finden.



Projektion (4)

- Beispiel:
 - **create view** bk **as**
 select Titel, Name **from**
 Buch **left outer join** Ausleih
 using (INV.NR)
 - **select** Titel **from** bk
- Wie kann uns hier die Optimierungsregel zur Projektion helfen?

Einleitung

Projektion

Selektion

Join

Mengen-
op.

Vollst.keit



Projektion (5)

- *Anmerkung:* Ersetzung von

$\pi[\text{Invnr}] (\pi[\text{Invnr}, \text{ISBN}] (\text{Buch}))$

durch

$\pi[\text{Invnr}] (\text{Buch})$

ist immer vorteilhaft.

- Bei anderen Transformationen hängt Vorteilhaftigkeit vom Datenbankzustand ab, z. B. Vertauschung von Selektionen.

Einleitung

Projektion

Selektion

Join

Mengen-
op.

Vollst.keit



Selektion (1)

- Beispiel

$\sigma[\text{Name} \leq \text{'N'}]$ (Ausleih)

ergibt

| INV.NR | NAME |
|--------|--------|
| 4711 | Meyer |
| 0007 | Müller |
| 4712 | Meyer |

Einleitung

Projektion

Selektion

Join

Mengen-
op.

Vollst.keit



Selektion (2)

- Selektionsbedingungen:
 - F *Konstanten-Selektion*

Attribut θ Konstante

boolesches Prädikat θ ist = oder \neq ,
bei linear geordneten Wertebereichen
auch \leq , $<$, \geq oder $>$
 - F *Attribut-Selektion*

Attribut1 θ Attribut2
 - F logische Verknüpfung mehrerer Konstanten-
oder Attribut-Selektionen mit \wedge , \vee oder \neg .
(Wird gleich diskutiert.)

Einleitung

Projektion

Selektion

Join

Mengen-
op.

Vollst.keit

Selektion (3)

- Einfache Optimierungsregeln:
 - Selektionen lassen sich in der Reihenfolge beliebig vertauschen,
Beispiel: $\sigma_{\text{Invnr}=4711}(\sigma_{\text{Name}\leq\text{'N'}}(\text{Ausleih}))$
 $= \sigma_{\text{Name}\leq\text{'N'}}(\sigma_{\text{Invnr}=4711}(\text{Ausleih}))$
 - Manchmal lassen sich Projektion und Selektion vertauschen;
 - Ist $\pi_{\text{Invnr}}(\sigma_{\text{Name}\leq\text{'N'}}(\text{Ausleih}))$
 $= \sigma_{\text{Name}\leq\text{'N'}}(\pi_{\text{Invnr}}(\text{Ausleih}))$?

Einleitung

Projektion

Selektion

Join

Mengen-
op.

Vollst.keit



Selektion (4)

- Voraussetzung für Vertauschbarkeit:
 - Selektionsattribute kommen in der Projektionsliste vor.
 - Beispiel: $\pi_{\text{Name}}(\sigma_{\text{Name} \leq 'N'}(\text{Ausleih}))$
 - Überprüfung der Vertauschbarkeit setzt Analyse der Selektionsbedingung voraus.

Einleitung
Projektion
Selektion
Join
Mengen-
op.
Vollst.keit

Selektion (5)

- Selektionsbedingungen:
 - *F Konstanten-Selektion*
Attribut θ Konstante
 - *F Attribut-Selektion*
Attribut1 θ Attribut2
 - *F logische Verknüpfung mehrerer Konstanten- oder Attribut-Selektionen mit \wedge , \vee oder \neg .*
Jedoch:
 - Analyse aufwendig.
 - z.T. redundant.
 - Negation problembehaftet
 - Sicherheit der Anfrage?!?

Einleitung
Projektion
Selektion
Join
Mengen-
op.
Vollst.keit



Verbund

- Syntax des (natürlichen) Verbundes (englisch: natural join)

`Relation1 ⋈ Relation2`

- Verbund verknüpft Tabellen über gleichbenannten Spalten bei gleichen Attributwerten.
- Natural Join:
 - Verbund nur über gleich benannte Attribute
 - Join-Attribut taucht in der Ergebnisrelation nur als eine Spalte auf

Einleitung

Projektion

Selektion

Join

Mengen-
op.

Vollst.keit

Laufendes Beispiel

Ausleih

| INV.NR | NAME |
|--------|--------|
| 4711 | Meyer |
| 1201 | Schulz |
| 0007 | Müller |
| 4712 | Meyer |

Buch

| INV.NR | TITEL | ISBN | AUTOR |
|--------|--------------|-------|---------|
| 0007 | Dr. No | 3-324 | Fleming |
| 1201 | Objektbanken | 3-111 | Heuer |
| 4711 | Datenbanken | 3-345 | Vossen |
| 4712 | Datenbanken | 3-345 | Ullman |
| 4717 | PASCAL | 3-989 | Wirth |

Einleitung
Projektion
Selektion
[Join](#)
Mengen-
op.
Vollst.keit

Verbund: Beispiel (1)

- Ausleih \bowtie Buch
ergibt

| NAME | INV.NR | TITEL | ISBN | AUTOR |
|--------|--------|--------------|-------|---------|
| Müller | 0007 | Dr. No | 3-324 | Fleming |
| Schulz | 1201 | Objektbanken | 3-111 | Heuer |
| Meyer | 4711 | Datenbanken | 3-345 | Vossen |
| Meyer | 4712 | Datenbanken | 3-345 | Ullman |

- Nicht ausgeliehenes Pascal-Buch verschwindet: Tupel, die keinen Partner finden (dangling tuples), werden eliminiert.
- In SQL: *outer join* \bowtie der „dangling tuples“ übernimmt.

Verbund: Beispiel (2)

- $\pi[\text{Autor}] (\text{Buch}) \bowtie \pi[\text{Invnr}] (\text{Ausleih})$
 - keine gemeinsamen Attribute
 - Verbund entartet zu *kartesischem Produkt*.
- $\pi[\text{Autor}] (\text{Buch}) \times \pi[\text{Invnr}] (\text{Ausleih})$

| AUTOR | INV.NR |
|---------|--------|
| Fleming | 4711 |
| Fleming | 1201 |
| Fleming | 0007 |
| Fleming | 4712 |
| Heuer | 4711 |
| Heuer | 1201 |
| Heuer | 0007 |
| Heuer | 4712 |
| ... | ... |

Eigenschaften vom Verbund

- Verbund ist kommutativ: $r_1 \bowtie r_2 = r_2 \bowtie r_1$
- Verbund ist assoziativ: $(r_1 \bowtie r_2) \bowtie r_3 = r_1 \bowtie (r_2 \bowtie r_3)$
- Daher erlaubt: $\bigotimes_{i=1}^p r_i$
- Beispiel dafür, daß Join-Reihenfolge wichtig ist:

| r_1 | r_2 | r_3 | | | | | | | | | | | | | | | | | | |
|--|---------------|-------|--|---|---|---|-------|-----|-----|---|---------------|--|---|---|---|-------|-----|-----|---|---------------|
| <table border="1"><thead><tr><th>B</th></tr></thead><tbody><tr><td>b</td></tr></tbody></table> | B | b | <table border="1"><thead><tr><th>B</th><th>C</th></tr></thead><tbody><tr><td>a</td><td>c_1</td></tr><tr><td>...</td><td>...</td></tr><tr><td>a</td><td>$c_{1000000}$</td></tr></tbody></table> | B | C | a | c_1 | ... | ... | a | $c_{1000000}$ | <table border="1"><thead><tr><th>B</th><th>D</th></tr></thead><tbody><tr><td>a</td><td>d_1</td></tr><tr><td>...</td><td>...</td></tr><tr><td>a</td><td>$d_{1000000}$</td></tr></tbody></table> | B | D | a | d_1 | ... | ... | a | $d_{1000000}$ |
| B | | | | | | | | | | | | | | | | | | | | |
| b | | | | | | | | | | | | | | | | | | | | |
| B | C | | | | | | | | | | | | | | | | | | | |
| a | c_1 | | | | | | | | | | | | | | | | | | | |
| ... | ... | | | | | | | | | | | | | | | | | | | |
| a | $c_{1000000}$ | | | | | | | | | | | | | | | | | | | |
| B | D | | | | | | | | | | | | | | | | | | | |
| a | d_1 | | | | | | | | | | | | | | | | | | | |
| ... | ... | | | | | | | | | | | | | | | | | | | |
| a | $d_{1000000}$ | | | | | | | | | | | | | | | | | | | |

Anmerkung: wenn r_1 und r_3 keine gemeinsamen Attribute haben, ist das Zwischenergebnis das Kreuzprodukt \rightarrow teurer geht nicht!



Kommentare zu vorangegangener Folie

- gute Join-Reihenfolge ist wichtig!
 - abhängig von den Daten in den Relationen.
 - schwierig zu erkennen
 - Problem: physische Datenunabhängigkeit
 - Keine Festlegung der Join-Reihenfolge in SQL möglich
- guter Optimierer im DBMS ist wichtig!

Einleitung
Projektion
Selektion
Join
Mengen-
op.
Vollst.keit



Mengenoperationen und Umbenennung (1)

- Mengenoperationen der relationalen Algebra setzen gleiches Schema voraus.

Einleitung

Projektion

Selektion

Join

Mengen-
op.

Vollst.keit



Mengenoperationen und Umbenennung (2)

Buch1

AUTOR1

Fleming

Heuer

Vossen

Ullman

Wirth

Buch2

AUTOR2

Witt

Vossen

Silberschatz

Meier

Wirth

Einleitung

Projektion

Selektion

Join

Mengen-
op.

Vollst.keit

- *Umbenennung*

$\beta[\text{neu} \leftarrow \text{alt}] (\text{relation})$

(bzw. $\beta_{\text{neu} \leftarrow \text{alt}} (\text{relation})$)

ändert Attributnamen von *alt* in *neu*.

- Beispiel: $\beta[\text{Autor1} \leftarrow \text{Autor2}] (\text{Buch2})$
- Durch Umbenennung nun Vereinigung, Differenz und Durchschnitt möglich.



Mengenoperationen: Vereinigung

relation1 \cup relation2

Beispiel:

Buch1 \cup β [Autor1 \leftarrow Autor2] (Buch2)

AUTOR1

Fleming

Heuer

Meier

Silberschatz

Ullman

Vossen

Wirth

Witt

Einleitung

Projektion

Selektion

Join

Mengen-
op.

Vollst.keit

Mengenoperationen: Differenz

relation1 - relation2

Beispiel:

Buch1 - β [Autor1 \leftarrow Autor2] (Buch2)

AUTOR1

Fleming

Heuer

Ullman

Einleitung
Projektion
Selektion
Join
Mengen-
op.
Vollst.keit

Mengenoperationen: Durchschnitt

$\text{relation1} \cap \text{relation2}$

Beispiel:

$\text{Buch1} \cap \beta[\text{Autor1} \leftarrow \text{Autor2}] \quad (\text{Buch2})$

AUTOR1

Vossen

Wirth

Einleitung
Projektion
Selektion
Join
Mengen-
op.
Vollst.keit



Mengenoperationen, Umbenennung (1)

Umbenennung ermöglicht

- Verbunde, wo bisher kartesische Produkte ausgeführt wurden (unterschiedliche Attribute werden gleich benannt),
- kartesische Produkte, wo bisher Verbunde ausgeführt wurden (gleiche Attribute werden unterschiedlich genannt),
- Mengenoperationen.

Einleitung

Projektion

Selektion

Join

Mengen-
op.

Vollst.keit

Laufendes Beispiel, leicht modifiziert

- Illustration des ersten Bullets der vorangegangenen Folie:

Ausleih

| INV.NR | NAME |
|--------|--------|
| 4711 | Meyer |
| 1201 | Schulz |
| 0007 | Müller |
| 4712 | Meyer |

Attribut umbenannt

Buch

| <u>L-NR</u> | TITEL | ISBN | AUTOR |
|-------------|--------------|-------|---------|
| 0007 | Dr. No | 3-324 | Fleming |
| 1201 | Objektbanken | 3-111 | Heuer |
| 4711 | Datenbanken | 3-345 | Vossen |
| 4712 | Datenbanken | 3-345 | Ullman |
| 4717 | PASCAL | 3-989 | Wirth |

Mengenoperationen, Umbenennung (2)

- Was ist der Natural Join dieser Relationen ohne Umbenennung?

Buch1

AUTOR

Fleming

Heuer

Vossen

Ullman

Wirth

Buch2

AUTOR

Witt

Vossen

Silberschatz

Meier

Wirth

Einleitung
Projektion
Selektion
Join
Mengen-
op.
Vollst.keit

Mengenoperationen, Umbenennung (3)

Beispiel:

Buch1

AUTOR

Fleming

Heuer

Vossen

Ullman

Wirth

Buch2

AUTOR

Witt

Vossen

Silberschatz

Meier

Wirth

- Wir wollen Paare bilden (Fleming, Witt), (Fleming, Vossen), ..., (Heuer, Witt), ...
- Vorgehen: Umbenennung und Natural Join.

Einleitung
Projektion
Selektion
Join
Mengen-
op.
Vollst.keit



Eigenschaften von Mengenoperationen

- Mengenoperationen sind kommutativ:

$$r_1 \cup r_2 = r_2 \cup r_1$$

- Mengenoperationen sind assoziativ:

$$(r_1 \cup r_2) \cup r_3 = r_1 \cup (r_2 \cup r_3)$$

- das gilt auch für $-$, \cap , \cup (und \times , \bowtie)

Einleitung

Projektion

Selektion

Join

Mengen-
op.

Vollst.keit

Weitere Optimierungsregeln

- Projektion kann mit Vereinigung vertauscht werden

$$\pi_x(r_1 \cup r_2) = \pi_x(r_2) \cup \pi_x(r_1)$$

warum nicht auch mit $-$, \cap ?

- Selektion und Kreuzprodukt können zum Verbund zusammengefasst werden, wenn Selektionsbedingung eine Join-Bedingung ist

– Beispiel Equi-Join

$$\sigma_{R.A = S.B} (R \times S) = R \bowtie_{R.A = S.B} S$$

- Umformungsbedingungen aus der Logik gelten
- Operatoren ohne Auswirkungen aufs Ergebnisse entfernen (Operationen auf leerer Menge, Konjunktionen von Selektionsbedingungen mit Überschneidungen etc.)

Einleitung

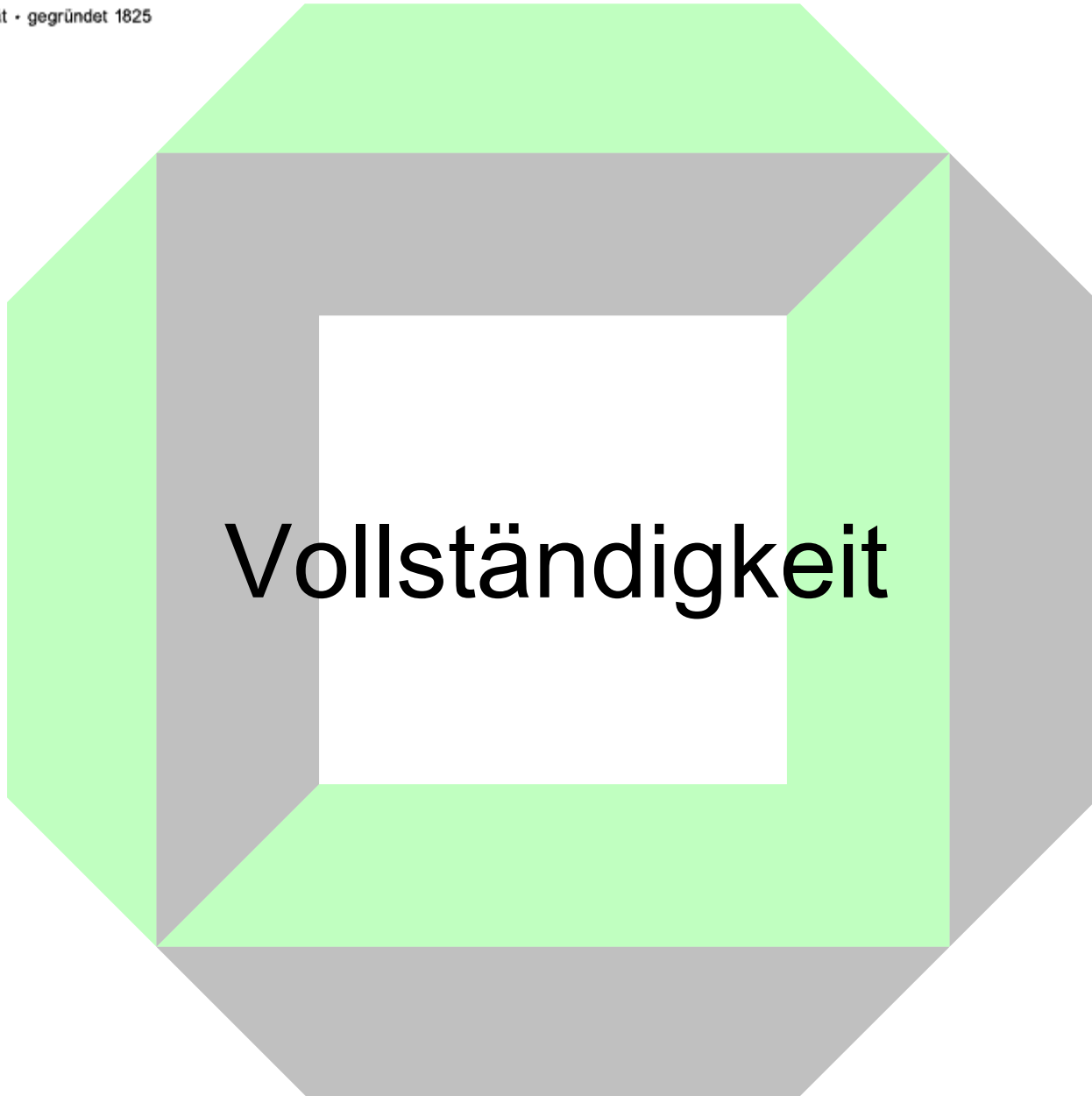
Projektion

Selektion

Join

Mengen-
op.

Vollst.keit





Anforderungen

- Anfragen (in Anfragesprache) werden i. Allg. abgebildet auf Folge von Algebra-Operatoren.
- Anforderungen an diese Algebra:
 - **Optimierbarkeit:**
Bestehend aus wenigen Operationen, für die es Optimierungsregeln gibt.
 - **Effizienz:**
Jede Operation ist effizient ausführbar.
Im Relationenmodell hat jede Operation eine Komplexität $\leq O(n^2)$
 - **Mengenorientiertheit:** Jede Operation soll auf Mengen von Daten gleichzeitig arbeiten, nicht navigierend nur auf einzelnen Elementen (one-tuple-at-a-time).

Einleitung

Projektion

Selektion

Join

Mengen-
op.

Vollst.keit





Unabhängigkeit und Vollständigkeit (1)

- Minimale Relationenalgebra:
 $\Omega = \pi, \sigma, \bowtie, \beta, \cup$ und $-$
- *Relationale Vollständigkeit*: Jede andere Menge von Operationen, genauso mächtig wie Ω .
- *Strenge relationale Vollständigkeit*:
Zu jedem Ausdruck mit Operatoren aus Ω gibt es einen Ausdruck auch mit der anderen Menge von Operationen, also ohne Sprachkonstrukte wie z.B. „:“, „while“.

Einleitung
Projektion
Selektion
Join
Mengen-
op.
[Vollst.keit](#)



Unabhängigkeit und Vollständigkeit (2)

- Ω ist unabhängig:
Kein Operator kann weggelassen werden, ohne Vollständigkeit zu verlieren.
- Andere unabhängige Menge:
 \bowtie durch \times ersetzen.
- Warum wichtig?
 - Redundanzfreiheit
für formale Überlegungen vorteilhaft.
 - Minimalität bequemer, wenn es darum geht, Vollständigkeit nachzuweisen.

Einleitung
Projektion
Selektion
Join
Mengen-
op.
[Vollst.keit](#)

Beispiel für Vollständigkeit

- Verbund kann über kartesisches Produkt hergeleitet werden:

$$R := \{ a_1, \dots, a_i, r_1, \dots, r_n \}$$

$$S := \{ a_1, \dots, a_i, s_1, \dots, s_m \}$$

$$R \bowtie S =$$

$$\pi [r_1, \dots, r_n, a_1, \dots, a_i, s_1, \dots, s_m] \\ (\sigma [R.a_1 = S.a_1 \wedge \dots \wedge R.a_i = S.a_i] (R \times S))$$

Einleitung

Projektion

Selektion

Join

Mengen-
op.

Vollst.keit



Schlußbemerkungen

Was war Gegenstand dieses Kapitels?

- Grundlage: Relationale Algebra,
- Optimierungs-/Transformationsregeln für Algebraausdrücke – Anfrageoptimierung,
- erhebliche Unterschiede zwischen äquivalenten Ausdrücken,
- Zusammenhang zu physischer Datenunabhängigkeit.