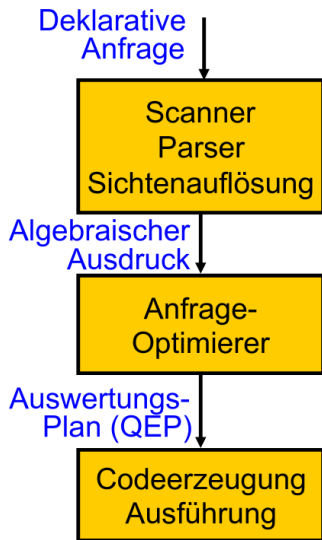


# *Datenbanken:* Tutorium 13

Marvin Jahn

29.01.2020

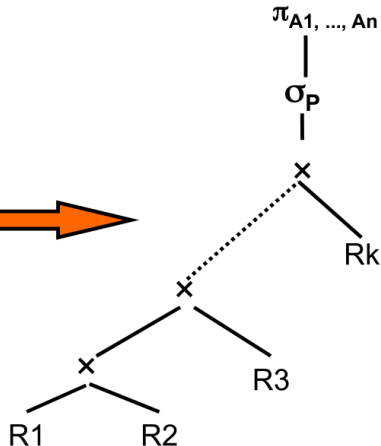
# Wdh.: Ablauf der Anfragebearbeitung



# Wdh.: Kanonische Übersetzung

**Kanonische Übersetzung:** Standardübersetzung von SQL in relationale Algebra

**select** A1, ..., An  
**from** R1, ..., Rk  
**where** P



# Optimierung einer Anfrage

- es gibt verschiedene Möglichkeiten, eine Anfrage in relationale Algebra zu übersetzen

# Optimierung einer Anfrage

- es gibt verschiedene Möglichkeiten, eine Anfrage in relationale Algebra zu übersetzen
- die kanonische Übersetzung ist eine “direkte” Übersetzung des SQL-Codes in relationale Algebra, aber ineffizient (Kreuzprodukte sind sehr teuer)

# Optimierung einer Anfrage

- es gibt verschiedene Möglichkeiten, eine Anfrage in relationale Algebra zu übersetzen
- die kanonische Übersetzung ist eine “direkte” Übersetzung des SQL-Codes in relationale Algebra, aber ineffizient (Kreuzprodukte sind sehr teuer)
- das Problem, einen optimalen Anfrageplan zu erstellen, ist NP-hart; man begnügt sich also mit einer “möglichst guten” Lösung

# Optimierung einer Anfrage

Vorgehen:

- Bestimmen der kanonischen Übersetzung

# Optimierung einer Anfrage

Vorgehen:

- Bestimmen der kanonischen Übersetzung
- Aufbrechen von Selektionen



# Optimierung einer Anfrage

Vorgehen:

- Bestimmen der kanonischen Übersetzung
- Aufbrechen von Selektionen
- Verschieben von Selektionen nach “unten” im Plan

# Optimierung einer Anfrage

Vorgehen:

- Bestimmen der kanonischen Übersetzung
- Aufbrechen von Selektionen
- Verschieben von Selektionen nach “unten” im Plan
- Zusammenfassen von Selektionen und Kreuzprodukten zu Joins

# Optimierung einer Anfrage

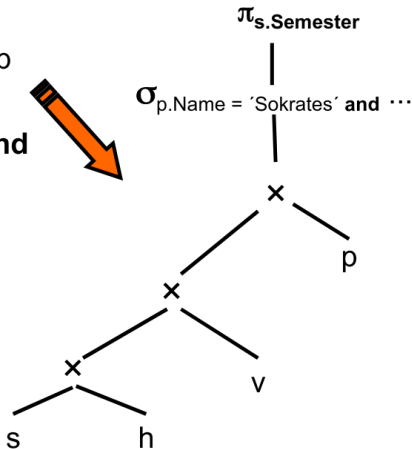
Vorgehen:

- Bestimmen der kanonischen Übersetzung
- Aufbrechen von Selektionen
- Verschieben von Selektionen nach “unten” im Plan
- Zusammenfassen von Selektionen und Kreuzprodukten zu Joins
- Bestimmen der Joinreihenfolge

# Optimierung einer Anfrage: Beispiel

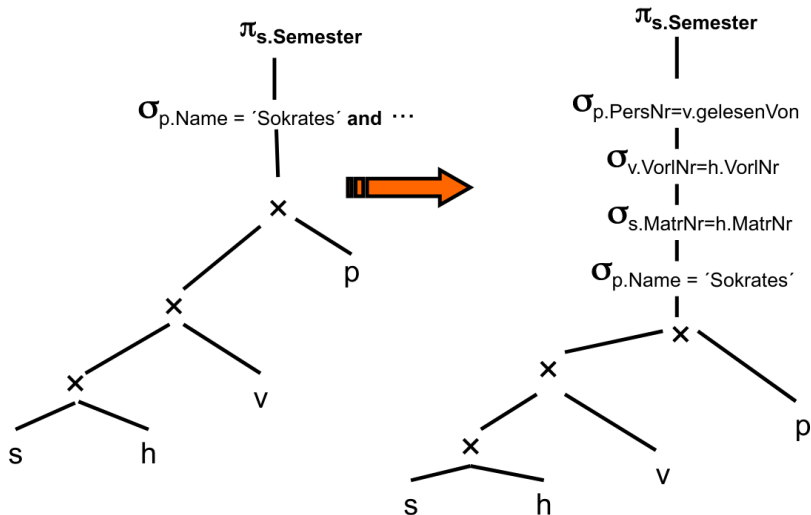
## Kanonische Übersetzung

**select** distinct s.Semester  
**from** Studenten s, hören h  
Vorlesungen v, Professoren p  
**where** p.Name = 'Sokrates' **and**  
v.gelesenVon = p.PersNr **and**  
v.VorlNr = h.VorlNr **and**  
h.MatrNr = s.MatrNr



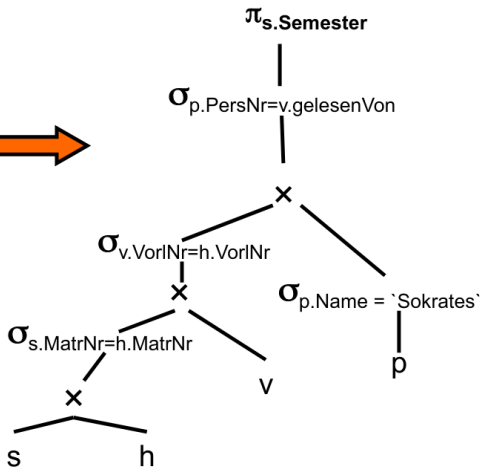
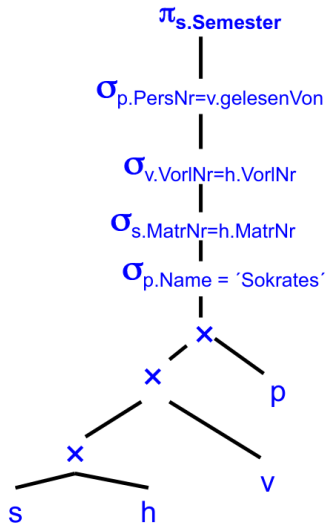
# Optimierung einer Anfrage: Beispiel

## Aufbrechen von Selektionen



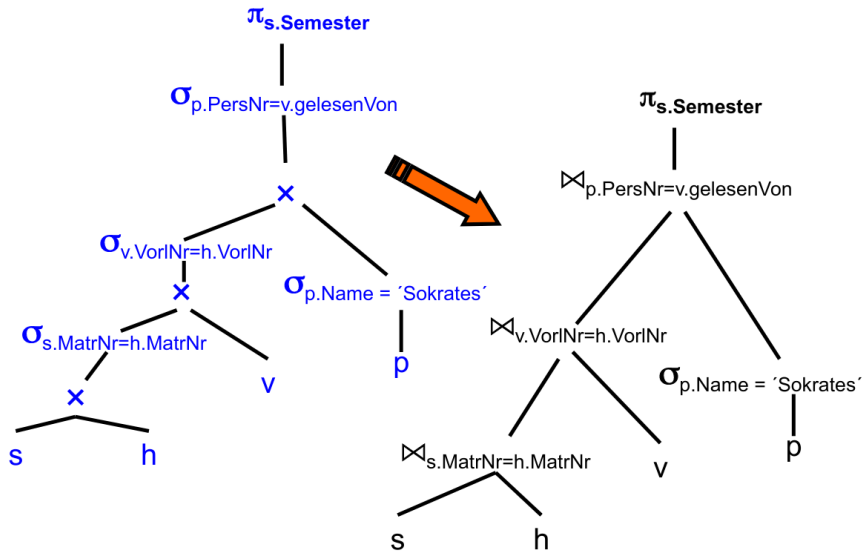
# Optimierung einer Anfrage: Beispiel

*Verschieben von Selektionen nach "unten" im Plan*



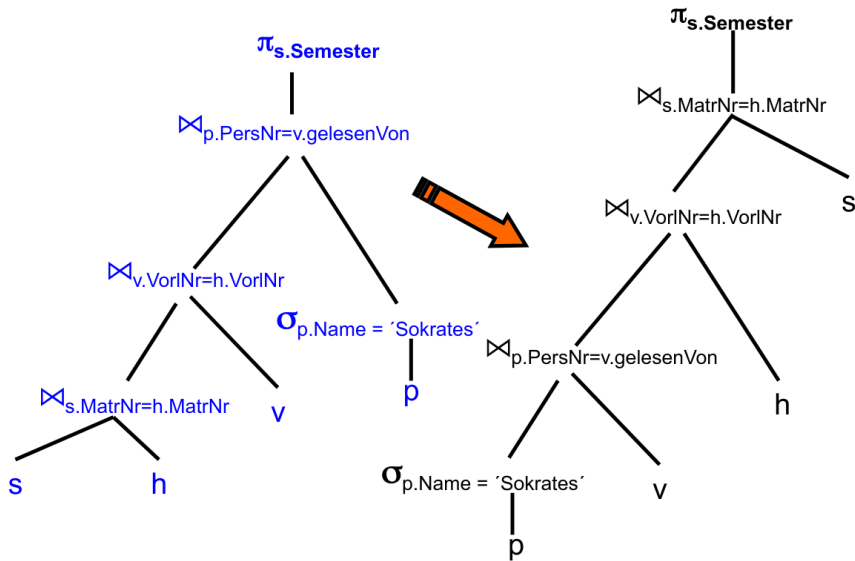
# Optimierung einer Anfrage: Beispiel

*Zusammenfassen von Selektionen und Kreuzprodukten zu Joins*



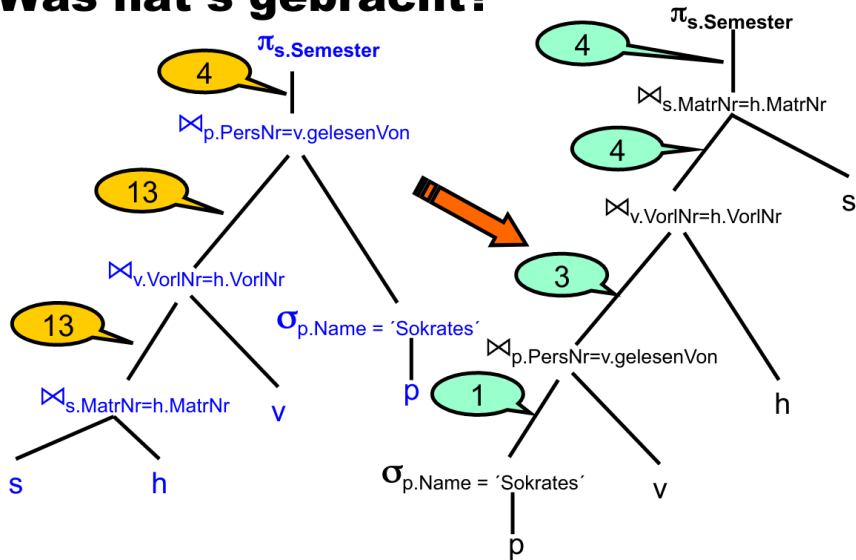
# Optimierung einer Anfrage: Beispiel

Bestimmen der Joinreihenfolge





## Was hat's gebracht?



- **Selektivität:** Anteil der qualifizierenden Tupel einer Operation

# Selektivität

- **Selektivität:** Anteil der qualifizierenden Tupel einer Operation
- Selektion mit Bedingung  $p$ :

$$sel_p := \frac{|\sigma_p(R)|}{|R|}$$

# Selektivität

- **Selektivität:** Anteil der qualifizierenden Tupel einer Operation
- Selektion mit Bedingung  $p$ :

$$sel_p := \frac{|\sigma_p(R)|}{|R|}$$

- Join von  $R$  mit  $S$ :

$$sel_{RS} := \frac{|R \bowtie S|}{|R \times S|} = \frac{|R \bowtie S|}{|R| \cdot |S|}$$