



## 4. Grundlagen des Relationenmodells

Inhalt  
Übersicht  
Grundkonzepte  
Abbildung von ER-Diagrammen  
Relationenalgebra  
Algebraische Optimierung

## Übersicht (1)

- **Datenstruktur**
  - Relation (Tabelle)
    - einzige Datenstruktur (neben atomaren Werten)
    - alle Informationen ausschließlich durch Werte dargestellt
    - zeitinvariante Typinformation: Relationenschema
    - Integritätsbedingungen auf/zwischen Relationen: relationale Invarianten
- **Operatoren auf (mehreren) Relationen**
  - Vereinigung, Differenz
  - Kartesisches Produkt
  - Projektion
  - Selektion
  - zusätzlich: Grundoperationen (Einfügen, Löschen, Ändern)



## Übersicht (2)

- **Beziehungen**
  - sind stets explizit, binär und symmetrisch
  - werden durch Werte dargestellt: Rolle von Primär-/Fremdschlüssel (Gewährleistung von referentieller Integrität)
  - können in SQL automatisch gewartet werden (referentielle Aktionen)
- **Entwurfstheorie**
  - Normalformenlehre (wünschenswerte und zweckmäßige Relationen)
  - Synthese von Relationen



## Grundkonzepte (1)

<u>ERM</u>	<u>RM</u>
Einwertiges Attribut Definitionsbereich Primärschlüssel	} wie im ERM
Zusammengesetztes Attribut Mehrwertiges Attribut	} nur als unabhängige, einwertige Attribute
Entity-Menge Relationship-Menge	} <b>Relation</b>



## Grundkonzepte (2)

### Mathematische Notation:

$D_1, D_2, \dots, D_n$

Definitionsbereiche (Domänen)

$R \subseteq D_1 \times D_2 \times \dots \times D_n$

Relation (Beziehung)

$t \in R$

Tupel / Record

### Notation für Datenbank-Relationen:

$A_1, A_2, \dots, A_n$

Attribute

$D_1, D_2, \dots, D_n$

primitive Datentypen

$R \in \text{Rel} (A_1:D_1, \dots, A_n:D_n)$

Relation über den Attributen  $A_1, \dots, A_n$   
mit den Domänen  $D_1, \dots, D_n$

- **Darstellungsmöglichkeit für R:** n-spaltige Tabelle
  - Jede **Relation** kann als Tabelle dargestellt werden
- **Relation ist eine Menge:** Garantie der Eindeutigkeit der Zeilen/Tupel
  - **Primärschlüssel** (und ggf. mehrere Schlüsselkandidaten)



## Grundkonzepte (3)

### ■ **Grundregeln:**

1. Jede Zeile (Tupel) ist eindeutig und beschreibt ein Objekt der Miniwelt
2. Die Ordnung der Zeilen ist ohne Bedeutung; durch ihre Reihenfolge wird keine für den Benutzer relevante Information ausgedrückt
3. Die Ordnung der Spalten ist ohne Bedeutung, da sie einen eindeutigen Namen (Attributnamen) tragen
4. Jeder Datenwert innerhalb einer Relation ist ein atomares Datenelement
5. Alle für den Benutzer bedeutungsvollen Informationen sind ausschließlich durch Datenwerte ausgedrückt
6. Es existieren ein Primärschlüssel und ggf. weitere Schlüsselkandidaten



## Grundkonzepte (4)

- **Informationsdarstellung im RM**
  - ausschließlich durch Werte, eingeschränkt durch Wertebereiche
  - Reihenfolge von Zeilen und Spalten enthält keine Information
- **Wie wird „relationenübergreifende“ Information dargestellt?**
  - Fremdschlüssel
    - Bezug auf den Primärschlüssel einer anderen (oder derselben) Relation definiert (gleicher Definitionsbereich)
    - trägt inter- oder intra-relationale Informationen
  - Beziehungen werden durch Fremdschlüssel und zugehörigen Primärschlüssel oder Schlüsselkandidaten dargestellt!



## Grundkonzepte (5)

- **Definition Fremdschlüssel:**
  - Ein Fremdschlüssel bzgl. einer Relation R1 ist ein Attribut oder eine Attributkombination FS einer Relation R2, für das/die zu jedem Zeitpunkt gilt: zu jedem Wert (ungleich Null) von FS muss ein gleicher Wert des Primärschlüssels PS oder eines Schlüsselkandidaten SK in irgendeinem Tupel von Relation R1 vorhanden sein.
- **Bemerkungen zu Fremdschlüssel:**
  - Fremdschlüssel und zugehöriger Primärschlüssel (Schlüsselkandidat) tragen wichtige inter-relationale (manchmal auch intra-relationale) Informationen. Sie sind auf dem gleichen Wertebereich definiert (vergleichbar und vereinigungsverträglich). Sie gestatten die Verknüpfung von Relationen mit Hilfe von Relationenoperationen.
  - Fremdschlüssel können Nullwerte aufweisen, wenn sie nicht Teil eines Primärschlüssels sind oder wenn nicht explizit NOT NULL spezifiziert ist.



## Grundkonzepte (6)

- **Bemerkungen zu Fremdschlüssel (Forts.)**
  - Schlüsselkandidaten können Nullwerte aufweisen, wenn nicht explizit NOT NULL spezifiziert ist.
  - Ein Fremdschlüssel ist zusammengesetzt, wenn der zugehörige Primärschlüssel (Schlüsselkandidat) zusammengesetzt ist. (In SQL2 beschreibt die „exotische“ MATCH-Option die Semantik partiell definierter Fremdschlüssel).
  - Eine Relation kann mehrere Fremdschlüssel besitzen, welche die gleiche oder verschiedene Relationen referenzieren.
  - Referenzierte und referenzierende Relation sind nicht notwendigerweise verschieden („self-referencing table“).
  - Zyklen sind möglich (geschlossener referentieller Pfad).



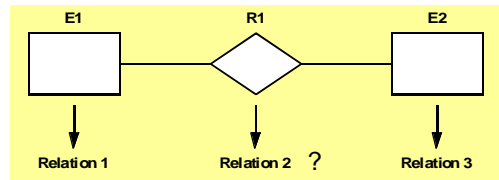
## Grundkonzepte (7)

- **Modellinhärente Integritätsbedingungen:**  
Welche Zusicherungen werden vom Datenmodell garantiert?
  - Mengeneigenschaft von Relationen (zur Abbildung von Entities/Relationships)
  - Beziehungstypen (1:1, ..., n:m) → mit Einschränkungen als (1:n)
  - referentielle Integrität (wertbasiert)
  - Kardinalitätsrestriktionen? → wünschenswert
  - Semantik der benutzerdefinierten Beziehung? → es ist keine Systemunterstützung vorgesehen

## Abbildung ERM-RM (1)

- Kriterien

- Informationserhaltung
  - Abbildung der ER-Konzepte auf RM-Konzepte
  - Möglichst genaue Übereinstimmung der Semantik (Übernahme aller spezifizierten Eigenschaften)
- Minimierung der Redundanz
- Minimierung des Verknüpfungsaufwandes
- Aber auch
  - Natürlichkeit der Abbildung
  - Keine Vermischung von Objekten
  - Verständlichkeit



Die RM-Konzepte erreichen nicht das semantische Ausdrucksvermögen der ER-Konzepte. Deshalb gehen gewisse Aspekte der Semantik verloren. Es kann jedoch versucht werden, diese durch weitergehende Datenmodellkonzepte (siehe SQL) oder durch Anwendungsfunktionen nachzubilden.

N. Ritter, DIS, SS 2005, Kapitel 4

11

## Abbildung ERM-RM (2)

- 2 Entity-Mengen mit (n:m)-Verknüpfung

- Regel:
  - Die Relationship-Menge wird auf eine Relation abgebildet, wobei der Primärschlüssel sich aus den Primärschlüsseln der beteiligten Entity-Mengen zusammensetzt. Alle Namen können übernommen werden; es ist jedoch auch eine Umbenennung möglich. Die Attributnamen in einer Relation müssen eindeutig sein.



Verwendung von drei Relationen erforderlich:

**PROJEKT** (JNR, BEZEICH, ...)

**PERS** (PNR, PNAME, ...)

**MITARBEIT** (JNR, PNR)

N. Ritter, DIS, SS 2005, Kapitel 4

12

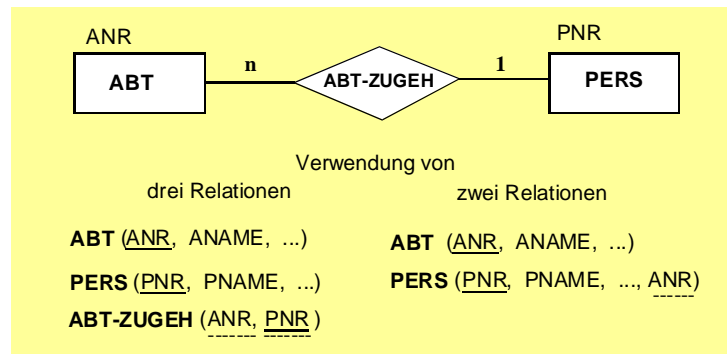


## Abbildung ERM-RM (3)

### 2 Entity-Mengen mit (1:n)-Verknüpfung

#### Regel:

- (1:n)-Beziehungen lassen sich, wenn sie keine eigenen Attribute besitzen, ohne eigene Relation darstellen. Hierzu wird in der abhängigen Relation (mit Beziehungskardinalität 1) der Primärschlüssel der referenzierten Relation als Fremdschlüssel verwendet.



N. Ritter, DIS

13

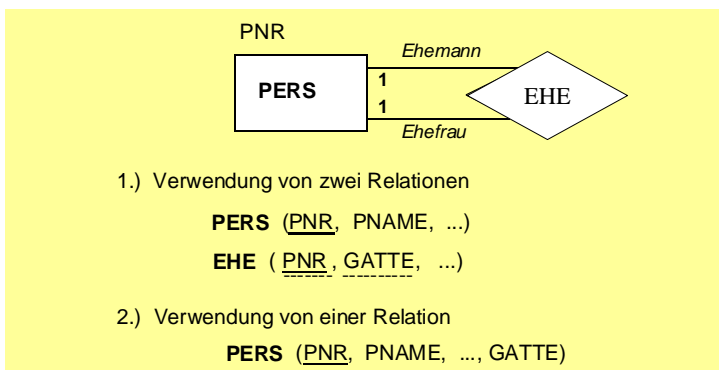


## Abbildung ERM-RM (4)

### 1 Entity-Menge mit (1:1)-Verknüpfung

#### Regel:

- Der Primärschlüssel der zugehörigen Entity-Menge wird in zwei Rollen verwendet. Deshalb ist eine Umbenennung erforderlich.



N. Ritter, DIS

14

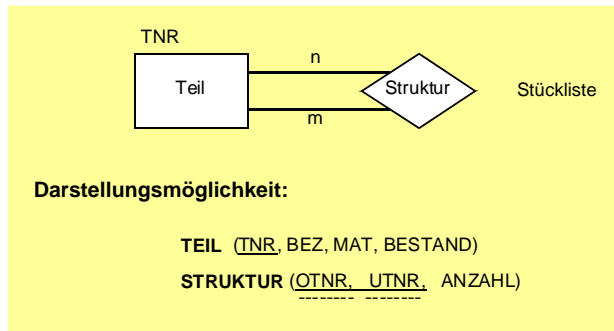


## Abbildung ERM-RM (5)

### 1 Entity-Menge mit (n:m)-Verknüpfung

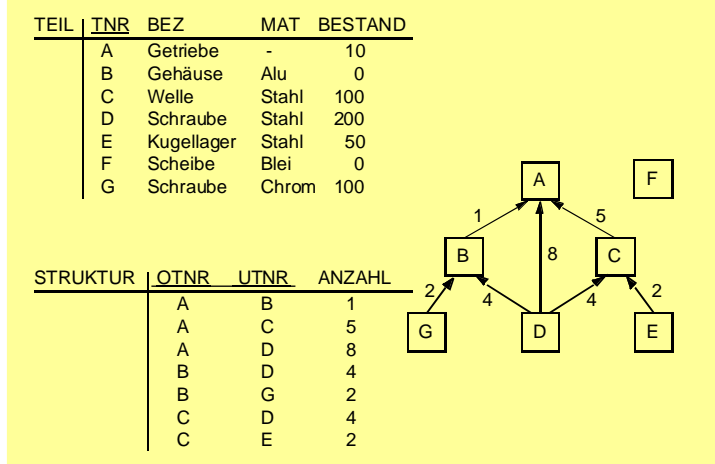
#### Regel:

- Eine (n:m)-Relationship-Menge muss durch eine eigene Relation dargestellt werden. Der Primärschlüssel der zugehörigen Entity-Menge wird in zwei Rollen verwendet. Deshalb ist eine Umbenennung erforderlich.



## Abbildung ERM-RM (6)

### 1 Entity-Menge mit (n:m)-Verknüpfung (Forts.)

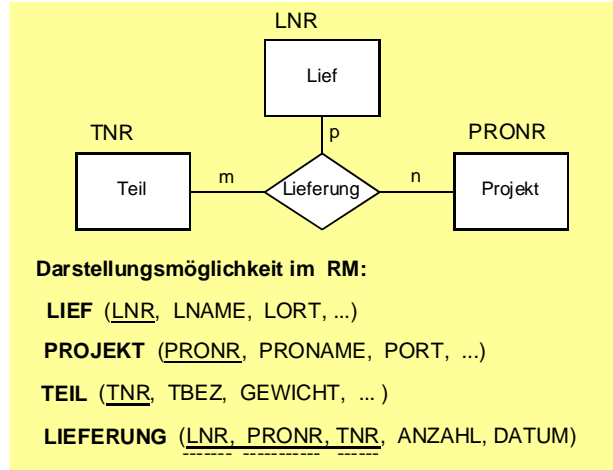






## Abbildung ERM-RM (7)

- mehrere Entity-Mengen mit (n:m)-Verknüpfung



N. Ritter, DIS, SS 2005, Kapitel 4

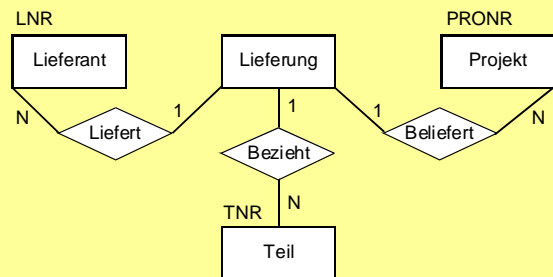
17



## Abbildung ERM-RM (8)

- mehrere Entity-Mengen mit (n:m)-Verknüpfung (Forts.)

Erinnerung: Manche Systeme unterstützen nur binäre Relationship-Mengen:



N. Ritter, DIS, SS 2005, Kapitel 4

18



## Abbildung ERM-RM (9)

- Abbildungstypen innerhalb einer Entity-Menge
  - **Entity-Menge kann viele Attribute besitzen**
    - **PERS** (PNR, NAME, ADRESSE, ..., GEHALT, VWL)
    - Ist weitere Zerlegung der Relation im DB-Schema sinnvoll?
  - **Horizontale Partitionierung**
    - Klassenbildung anhand von Selektionsbedingungen
    - PERS-VIP** (PNR, ..., VWL) GEHALT > 100K
    - PERS** (PNR, ..., VWL) GEHALT ≤ 100K
    - ist eigentlich Aufgabe eines Sichtkonzeptes
  - **Vertikale Partitionierung**
    - zur leichteren Einhaltung von Leistungs- und Schutzaspekten:  
**PERS-ÖFF** (PNR, PNAME, ADRESSE, ...)  
**PERS-PRIV** (PNR, GEHALT, VWL, ...)
    - ist Aufgabe des internen Schemas und des Sichtkonzeptes

N. Ritter, DIS, SS 2005, Kapitel 4

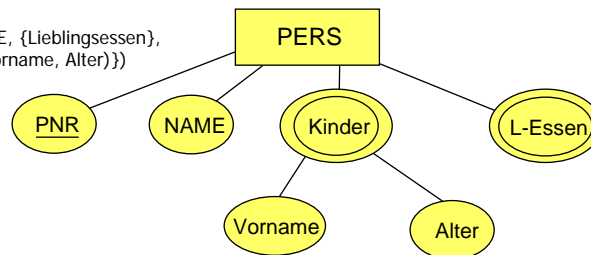
19



## Abbildung ERM-RM (10)

- Abbildungstypen innerhalb einer Entity-Menge (Forts.)

**PERS** (PNR, NAME, {Lieblingsessen},  
{Kinder (Vorname, Alter)})



### Darstellungsmöglichkeit:

**PERS** (PNR, NAME ...)  
**L-ESSEN** (PNR, GERICHT)  
**KINDER** (PNR, VORNAME, ALTER)

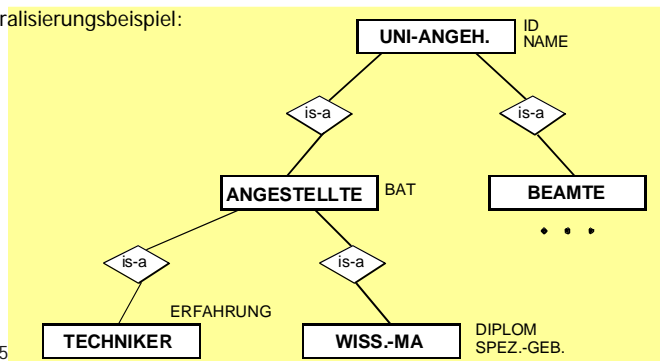
N. Ritter, DIS, SS 2005, Kapitel 4

20



## Abbildung ERM-RM (11)

- Generalisierung
  - RM sieht keine Unterstützung der Abstraktionskonzepte vor
  - keine Maßnahmen zur Vererbung (von Struktur, Integritätsbedingungen, Operationen)
  - „Simulation“ der Generalisierung eingeschränkt möglich
  - Generalisierungsbeispiel:



N. Ritter, DIS, SS 2005

21



## Abbildung ERM-RM (12)

- Generalisierung: Hausklassenmodell
  - Jede Instanz ist genau einmal und vollständig in ihrer Hausklasse gespeichert
  - Es wird eine horizontale Partitionierung der DB-Instanzen erreicht
  - Einsatz in ORION und O<sub>2</sub>

UNI-ANGEH.		ID	NAME		
		111	Ernie		

ANGESTELLTE		ID	NAME	BAT
		007	Garfield	Ia

TECHNIKER		ID	ERFAHRUNG	NAME	BAT
		123	SUN	Donald	IVa

WISS.-MA.		ID	DIPLOM	SEPZ.-GEB.	NAME	BAT
		333	Informatik	RECOVERY	Daisy	Ila
		765	Mathematik	ERM	Grouch	Ila

N. Ritter, DIS, SS 2005

22



## Abbildung ERM-RM (13)

- Generalisierung: Hausklassenmodell (Forts.)
  - **Eigenschaften:**
    - niedrige Speicherkosten und keine Änderungsanomalien
    - Retrieval kann rekursives Suchen in Unterklassen erfordern
    - explizite Rekonstruktion durch Relationenoperationen ( $\pi$ ,  $\cup$ )
      - Beispiel: Finde alle ANGESTELLTE:  
 $\pi_{ID, NAME, BAT}(TECHNIKER) \cup \pi_{ID, NAME, BAT}(WISS.-MA) \cup ANGESTELLTE$

Projektion: unärer Operator der Relationalalgebra zur Auswahl von Spalten/Attributen; siehe unten.



## Abbildung ERM-RM (14)

- Generalisierung: Partitionierungs-Modell
  - Jede Instanz wird entsprechend der Klassenattribute in der Is-a-Hierarchie zerlegt und in Teilen in den zugehörigen Klassen gespeichert
  - Es wird nur das ID-Attribut dupliziert
  - Es wird eine vertikale Partitionierung in der DB erzielt
  - IRIS wendet dieses Speicherungsmodell an

UNI-ANGEH.	ID	NAME	ANGESTELLTE	ID	BAT
	007	Garfield		007	Ia
	111	Ernie		123	IVa
	123	Donald		333	IIa
	333	Daisy		765	IIa
	765	Grouch			

TECHNIKER	ID	ERFAHRUNG
	123	SUN

WISS.-MA	ID	DIPLOM	SPEZ.-GEB
	333	Informatik	ERM
	765	Mathematik	MAD



## Abbildung ERM-RM (15)

- Generalisierung: Partitionierungs-Modell (Forts.)
  - **Eigenschaften**
    - geringfügig erhöhte Speicherkosten, aber hohe Aufsuch- und Aktualisierungskosten
    - Integritätsbedingungen:  $TECHNIKER.ID \subseteq ANGESTELLTE.ID$ , usw.
    - Instanzenzugriff erfordert implizite oder explizite **Verbundoperationen** ( $\bowtie$ )
      - Beispiel: Finde alle TECHNIKER-Daten

$TECHNIKER \bowtie_{ID} ANGESTELLTE \bowtie_{ID} UNI-ANGEH.$

Join/Verbund: binärer Operator der Relationenalgebra zur Verbindung der Tupeln der Argumentrelationen über gleiche Attributwerte; siehe unten.



## Abbildung ERM-RM (16)

- Generalisierung: Volle Redundanz
  - **Volle Redundanz**
    - Eine Instanz wird wiederholt in jeder Klasse, zu der sie gehört, gespeichert
    - Sie besitzt dabei die Werte der Attribute, die sie geerbt hat, zusammen mit den Werten der Attribute der Klasse

UNI-ANGEH.	ID	NAME	ANGESTELLTE	ID	NAME	BAT
	007	Garfield		007	Garfield	Ia
	111	Ernie		123	Donald	IVa
	123	Donald		333	Daisy	IIa
	333	Daisy		765	Grouch	IIa
	765	Grouch				

TECHNIKER	ID	NAME	BAT	ERFAHRUNG
	123	Donald	IVa	SUN

WISS.-MA	ID	NAME	BAT	DIPLOM	SPEZ.-GEB.
	333	Daisy	IIa	Informatik	RECOVERY
	765	Grouch	IIa	Mathematik	ERM



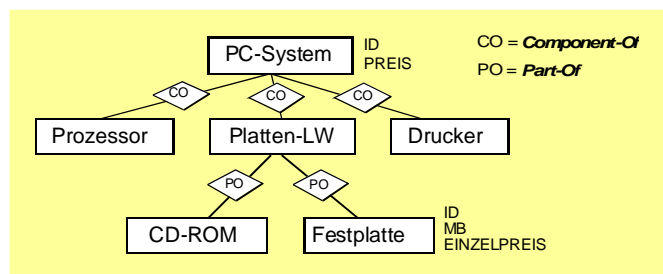
## Abbildung ERM-RM (17)

- Generalisierung: Volle Redundanz (Forts.)
  - **Eigenschaften**
    - sehr hoher Speicherplatzbedarf und Auftreten von Änderungsanomalien
    - sehr einfaches Retrieval, da nur die Zielklasse (z. B. ANGESTELLTE) aufgesucht werden muss



## Abbildung ERM-RM (18)

- Aggregation
  - Eigenschaften der Aggregation werden durch relationale Operationen nicht unterstützt
  - Beispiel:





## Abbildung ERM-RM (19)

- Aggregation (Forts.)

PC-System	PREIS	PROZESSOR	PLATTENK.	DRUCKER
PC-1	2000,-	I-Pentium III 866 MHz	PK1	Tintenstrahlr.
PC-2	2600,-	I-Pentium III 1000 MHz	PK2	Laserdrucker
PC-3	3700,-	I-Pentium IV 1,7 GHz	PK2	Laserdrucker

Plattenk.	ID	CD-ROM	Festplatte
	PK1		F1
	PK2	C1	F2
	PK3	C2	F3

CD-ROM	ID	Geschwindigk.	Preis	Festplatte	ID	Kapaz.	Preis
	C1	48-fach Speed	82.-		F1	10 GB	209.-
	C2	50-fach Speed	119.-		F2	30 GB	259.-
					F3	40 GB	449.-

Prozessor	ID	SPEC int 2000	Preis
	I-Pentium III 866 MHz	394	489.-
	I-Pentium III 1000 MHz	407	689.-
	I-Pentium IV 1,7 GHz	524	1079.-

Drucker	ID	Hersteller	Typ	Preis
	Tintenstrahlr.	HP	Desk Jet 640C	179.-
	Laserdrucker	HP	Laser 1200	949.-

N. Ritter, DIS, SS 2005



## Abbildung ERM-RM (20)

- Zusammenfassung der Abbildungskonzepte

- Datenstruktur:** Relation (Tabelle)
    - einzigste Datenstruktur (neben atomaren Werten)
    - alle Informationen ausschließlich durch Werte dargestellt
    - Integritätsbedingungen auf/zwischen Relationen: relationale Invarianten
  - Abbildung von Beziehungen durch PS/SK – FS**
    - alle Beziehungen sind explizit, binär und symmetrisch
    - alle Beziehungstypen müssen im Prinzip durch (n:1)-Beziehungen dargestellt werden
    - (n:m)-Beziehungstypen sind durch eine eigene Relation darzustellen
    - ein (n:1)-Beziehungstyp wird in der Regel nur dann auf eine eigene Relation abgebildet, wenn er beschreibende Attribute besitzt
  - Bewertung hinsichtlich der Abstraktionskonzepte**
    - keine direkte Bereitstellung der Abstraktionskonzepte
    - begrenzte Möglichkeiten zur Abbildung der Abstraktionskonzepte

N. Ritter, DIS, SS 2005, Kapitel 4

30



## Relationenalgebra (1)

- Datenmodell = Datenobjekte + Operatoren
- Unterstützung verschiedener Benutzerklassen
- Im RM wird vereinheitlichte Sprache angestrebt für
  - alle Aufgaben der Datenverwaltung: Datendefinition, Anfragen (*Queries*), Datenmanipulation, Zugriffs-, Integritäts- und Transaktionskontrolle
  - zur Nutzung
    - im 'Stand-Alone'-Modus (Ad-hoc-Anweisungen) und
    - in einer Wirtssprache (eingebettete DB-Anweisungen)
- Vier verschiedene Grundtypen:
  - Relationenalgebra (z. B. ISBL)
  - Relationenkalkül (z. B. Alpha)
  - Abbildungsorientierte Sprachen (z. B. SQL)
  - Graphikorientierte Sprachen (z. B. Query-by-Example)



## Relationenalgebra (2)

- **Algebra: nicht leere Menge von Objekten + Familie von Operationen**
- **Operationen**
  - **Klassische Mengenoperationen:**
    - Vereinigung, Differenz, kartesisches Produkt
    - ableitbar: Durchschnitt
  - **Relationenoperationen:**
    - Projektion, Restriktion (Selektion)
    - ableitbar: Verbund (Join), Division
- Auswahlvermögen entspricht Prädikatenkalkül erster Ordnung („relational vollständig“)





## Relationenalgebra (3)

### ■ Selektion (Restriktion): $\sigma_P$

- Auswahl von Zeilen einer Relation über ein Prädikat
- $P$  = log. Formel (ohne Quantoren!) bestehend aus Attributnamen, Konstanten, Vergleichsoperatoren ( $<$ ,  $=$ ,  $>$ ,  $\leq$ ,  $\neq$ ,  $\geq$ ) und logischen Verknüpfungen ( $\vee$ ,  $\wedge$ ,  $\neg$ )
- $\sigma_P(R) = \{ t \mid t \in R \wedge P(t) \}$
- Beispiel:

**ERG :=  $\sigma_{\text{ANR}='K55' \wedge \text{GEHALT} > 50\,000}$  (PERS)**

PERS	<u>PNR</u>	NAME	ALTER	GEHALT	ANR	MNR
	406	COY	47	50 700	K55	123
	123	MÜLLER	32	43 500	K51	-
	829	SCHMID	36	45 200	K53	777
	574	ABEL	28	36 000	K55	123



## Relationenalgebra (4)

### ■ Projektion: $\pi$

- Auswahl von Spalten (Attribute)  $A_1, A_2, \dots, A_k$  aus einer Relation  $R$  (Grad  $n \geq k$ )
- $\pi_{A_1, A_2, \dots, A_k}(R) = \{ p \mid \exists t \in R : p = \langle t[A_1], \dots, t[A_k] \rangle \}$   
(Alternative: Benutzung von Spaltennummern)
- Duplikateliminierung
- Beispiel:

**$\pi_{\text{ANR}, \text{MNR}}$  (PERS)**

PERS	<u>PNR</u>	NAME	ALTER	GEHALT	ANR	MNR
	406	COY	47	50 700	K55	123
	123	MÜLLER	32	43 500	K51	-
	829	SCHMID	36	45 200	K53	777
	574	ABEL	28	36 000	K55	123



## Relationenalgebra (5)

- Klassische Mengenoperationen
  - **Voraussetzung: Gleicher Grad und Vereinigungsverträglichkeit der beteiligten Relationen**
  - **Basisoperatoren**
    - Vereinigung:**  $R \cup S = \{t \mid t \in R \vee t \in S\}$
    - Differenz:**  $R - S = \{t \mid t \in R \wedge t \notin S\}$
  - **Redundante Operatoren**
    - Durchschnitt:**  $R \cap S = R - (R - S) = \{t \mid t \in R \wedge t \in S\}$
    - Symmetrische Differenz:**  $R \triangleright S = (R \cup S) - (R \cap S)$



## Relationenalgebra (6)

- Erweitertes Kartesisches Produkt
  - $K = R \times S = \{k \mid \exists x \in R, y \in S: (k = x | y)\}$   
mit  $x | y = \langle x_1, \dots, x_r, y_1, \dots, y_s \rangle$ ,

nicht  $\langle \langle x_1, \dots, x_r \rangle, \langle y_1, \dots, y_s \rangle \rangle$   
wie 'übliches' kartesisches Produkt!

PERS	PNR	ALTER	ANR
	406	47	K55
	123	32	K51
	829	36	K53

ABT	ANR	ANAME	ORT	ABT x PERS	ANR	ANAME	ORT	PNR	ALTER	ANR'
	K51	PLAN.	KL		K51	PLAN.	KL	406	47	K55
	K53	EINK.	F		K51	PLAN.	KL	123	32	K51
					K51	PLAN.	KL	829	36	K53
					K53	EINK.	F	406	47	K55
					K53	EINK.	F	123	32	K51
					K53	EINK.	F	829	36	K53



## Relationenalgebra (7)

- Verbund, Join,  $\Theta$ -Join
  - Seien R und S Relationen,  $\Theta \in \{<, =, >, \leq, \neq, \geq\}$  (arithm. Vergleichsoperator), A Attribut von R und B Attribut von S.  $\Theta$ -Verbund zwischen R und S:
 
$$V = (R \bowtie_{A \Theta B} S) = \sigma_{A \Theta B} (R \times S)$$
  - Alternative Definition anhand Spaltennummern  
Annahme: R hat Grad r und S hat Grad s,  $1 \leq i \leq r, 1 \leq j \leq s$ ,
 
$$V = (R \bowtie_{i \Theta j} S) = \sigma_{i \Theta_{r+j}} (R \times S)$$
  - Gleichverbund ( $\Theta = „=”$ )
    - Ein Gleichverbund zwischen R und S heißt *verlustfrei*, wenn alle Tupel von R und S am Verbund teilnehmen (sonst *verlustbehaftet*). Die inverse Operation Projektion erzeugt dann wieder R und S (*lossless join*).

N. Ritter, DIS, SS 2005, Kapitel 4

37



## Relationenalgebra (8)

- Verbund, Join,  $\Theta$ -Join (Forts.)
  - Definition ‚fortsetzbar‘ auf mehrere Join-Attribute
  - Natürlicher Verbund  $R \bowtie S$ : Gleichverbund über alle übereinstimmenden Attribute und anschließende Projektion, so dass keine Attribute doppelt
  - Verlustfreier Verbund:

$$\pi_{ANR, ANAME, ORT} (ABT \bowtie PERS) = ABT$$

$$\pi_{PNR, ANR, ALTER} (ABT \bowtie PERS) = PERS;$$

ABT	ANR	ANAME	ORT
	K51	PLAN	KL
	K53	EINK.	F
	K55	VERTR.	F

PERS	PNR	ALTER	ANR
	406	47	K55
	123	32	K51
	829	36	K53
	574	28	K55

ABT $\bowtie$ PERS	ANR	ANAME	ORT	PNR	ALTER
	K51	PLAN	KL	123	32
	K53	EINK.	F	829	36
	K55	VERTR.	F	406	47
	K55	VERTR.	F	574	28

N. Ritter, DIS, SS 2005, Kapitel 4



## Relationenalgebra (9)

- Definition *Natürlicher Verbund*

- gegeben:  $R(A_1, A_2, \dots, A_{r-j+1}, \dots, A_r), S(B_1, B_2, \dots, B_j, \dots, B_s)$
- o.B.d.A. (sonst. Umsortierung):  $B_1 = A_{r-j+1}, B_2 = A_{r-j+2}, \dots, B_j = A_r$
- Natürlicher Verbund zwischen R und S:

$$N = R \bowtie S =$$

$$\pi_{A_1, \dots, A_r, B_{j+1}, \dots, B_s} (\sigma_{(R.A_{r-j+1} = S.B_1) \wedge \dots \wedge (R.A_r = S.B_j)} (R \times S))$$



## Relationenalgebra (10)

- Natürlicher Verbund – Beispiel

- Finde alle Angestellten (PNR, ALTER, ANAME), die in einer Abteilung in Frankfurt arbeiten und zwischen 30 und 34 Jahre alt sind.

ABT	<u>ANR</u>	ANAME	AORT
	K51	Planung	Kaiserslautern
	K53	Einkauf	Frankfurt
	K55	Vertrieb	Frankfurt

PERS	<u>PNR</u>	NAME	ALTER	GEHALT	ANR	MNR
406		Coy	47	50 700	K55	123
123		Müller	32	43 500	K51	-
829		Schmid	36	45 200	K53	777
574		Abel	28	36 000	K55	123



## Relationenalgebra (11)

- Natürlicher Verbund – Beispiel (Forts.)
  - **Annahmen:**
    - ABT: N/10 Tupel
    - PERS: N Tupel
    - Gleichverteilung der Attributwerte  
AORT: 20 Werte  
ALTER: 50 Werte
    - Stochastische Unabhängigkeit der Werte verschiedener Attribute
    - Verlustfreie Verbunde von R1 und R2 über Primär-/Fremdschlüssel, mit  $\text{Card}(R1) < \text{Card}(R2)$ :  $\text{Card}(R1 \bowtie R2) = \text{Card}(R2)$



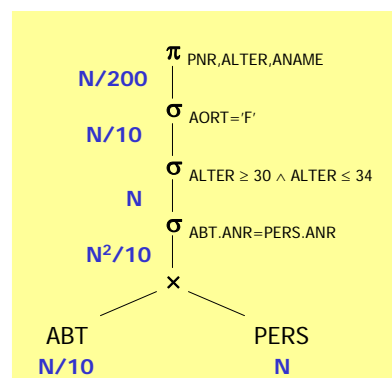
## Relationenalgebra (12)

- Natürlicher Verbund – Beispiel (Forts.)

- **Lösung 1:**

$\pi_{\text{PNR,ALTER,ANAME}}(\sigma_{\text{AORT}='F'}(\sigma_{\text{ALTER} \geq 30 \wedge \text{ALTER} \leq 34}(\sigma_{\text{ABT.ANR}=\text{PERS.ANR}}(\text{ABT} \times \text{PERS}}))))$

zugehöriger Operatorbaum:





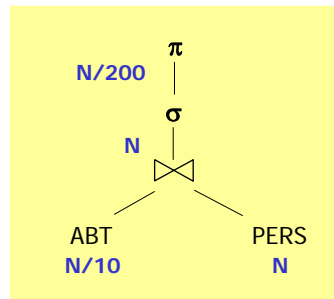
## Relationenalgebra (13)

- Natürlicher Verbund – Beispiel (Forts.)

- Lösung 2:

$\pi_{\text{PNR,ALTER,ANAME}} (\sigma_{\text{ALTER} \geq 30 \wedge \text{ALTER} \leq 34 \wedge \text{AORT}='F'} (\text{ABT} \bowtie \text{PERS}))$

zugehöriger Operatorbaum:



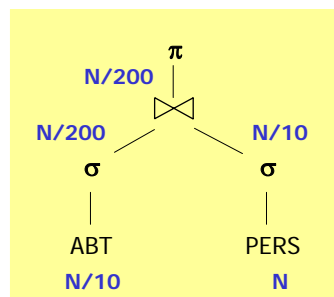
## Relationenalgebra (14)

- Natürlicher Verbund – Beispiel (Forts.)

- Lösung 3:

$\pi_{\text{PNR,ALTER,ANAME}} ((\sigma_{\text{AORT}='F'} (\text{ABT})) \bowtie (\sigma_{\text{ALTER} \geq 30 \wedge \text{ALTER} \leq 34} (\text{PERS})))$

zugehöriger Operatorbaum:



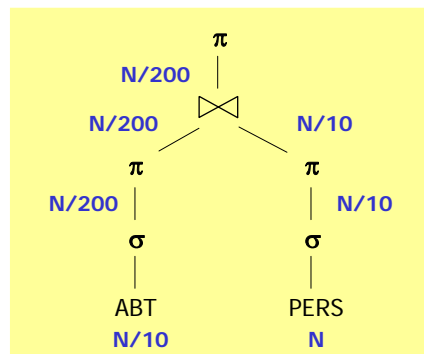
## Relationenalgebra (15)

- Natürlicher Verbund – Beispiel (Forts.)

- Lösung 4:**

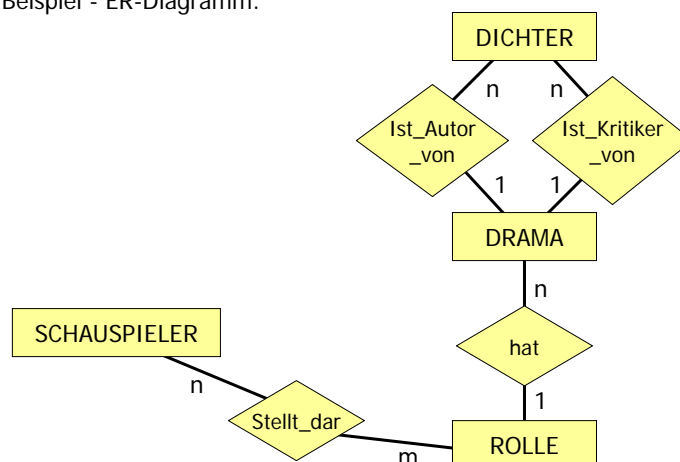
$\pi_{PNR,ALTER,ANAME} ((\pi_{ANR,ANAME} (\sigma_{AORT='F'} (ABT))) \bowtie (\pi_{PNR,ALTER,ANR} (\sigma_{ALTER \geq 30 \wedge ALTER \leq 34} (PERS))))$

zugehöriger Operatorbaum:



## Relationenalgebra (16)

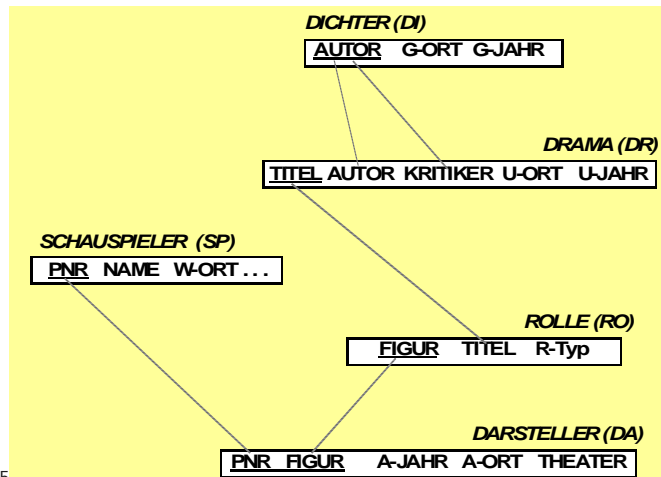
- Beispiel - ER-Diagramm:





## Relationenalgebra (17)

- Beispiel – relationales DB-Schema:



N. Ritter, DIS, SS 2005

47



## Relationenalgebra (18)

- Beispiel – Anfragen:

- Finde alle Schauspieler (NAME), die die Figur „Faust“ gespielt haben.

$$\pi_{\text{NAME}} (\sigma_{\text{FIGUR} = \text{„FAUST“}} (\text{SP} \bowtie_{\text{PNR}} \text{DA}))$$

- Finde alle Schauspieler (NAME), die im Drama „Faust“ mitgespielt haben.

$$\pi_{\text{NAME}} (\sigma_{\text{TITEL} = \text{„Faust“}} (\text{SP} \bowtie_{\text{PNR}} \text{DA} \bowtie_{\text{FIGUR}} \text{RO}))$$

- Finde alle Schauspieler (NAME), die in Dramen von Schiller mitgespielt haben.

$$\pi_{\text{NAME}} (\sigma_{\text{AUTOR} = \text{„Schiller“}} (\text{SP} \bowtie_{\text{PNR}} \text{DA} \bowtie_{\text{FIGUR}} \text{RO} \bowtie_{\text{TITEL}} \text{DR}))$$

N. Ritter, DIS, SS 2005, Kapitel 4

48





## Relationenalgebra (19)

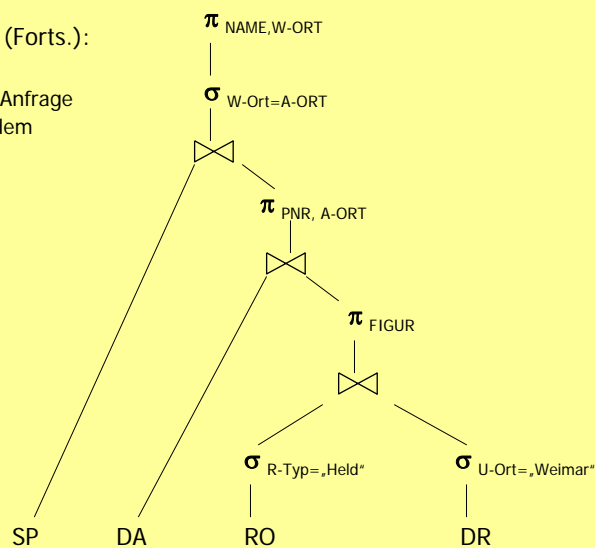
- Beispiel – Anfragen (Forts.):
  - Finde alle Schauspieler (NAME, W-ORT), die bei in Weimar uraufgeführten Dramen an ihrem Wohnort als 'Held' mitgespielt haben.

$$\pi_{\text{NAME, W-ORT}} (\sigma_{\text{W-ORT} = \text{A-ORT}} (\text{SP} \bowtie (\pi_{\text{PNR, A-ORT}} (\text{DA} \bowtie (\pi_{\text{FIGUR}} (\sigma_{\text{R-TYP} = \text{'HELD'}} (\text{RO}) \bowtie (\sigma_{\text{U-ORT} = \text{'WEIMAR'}} (\text{DR}))))))))$$


## Relationenalgebra (20)

- Beispiel – Anfragen (Forts.):

- Vorangegangene Anfrage entspricht folgendem Operatorbaum:





## Relationenalgebra (21)

### ■ Beispiel – Anfragen (Forts.):

- Liste alle Dramen mit ihren Autoren (TITEL, AUTOR, G-JAHR) auf, die nach 1800 uraufgeführt wurden.

$$\pi_{\text{TITEL, AUTOR, G-JAHR}} (\sigma_{\text{U-JAHR} > 1800} (\text{DI} \bowtie \text{DR}))$$

- Finde alle Schauspieler (NAME, W-ORT), die in Dramen von Schiller, die von in Weimar geborenen Dichtern kritisiert wurden, mitgespielt haben.

$$\pi_{\text{NAME, W-ORT}} (\text{SP} \bowtie \text{DA} \bowtie \text{RO} \bowtie \text{PNR} \bowtie \text{FIGUR} \bowtie \text{TITEL})$$
$$(\sigma_{\text{AUTOR} = \text{„Schiller“}} (\text{DR})) \bowtie (\sigma_{\text{G-ORT} = \text{„Weimar“}} (\text{DI}))$$
$$\text{KRITIKER} = \text{AUTOR}$$


## Relationenalgebra (22)

### ■ Beispiel – Anfragen (Forts.):

- Finde die Schauspieler, die **nie** gespielt haben.

$$\pi_{\text{PNR}} (\text{SP}) - \pi_{\text{PNR}} (\text{DA})$$

- Finde die Schauspieler, die **nur** Faust oder Wallenstein gespielt haben.

$$\pi_{\text{PNR}} (\text{DA}) - \pi_{\text{PNR}} (\sigma_{\text{FIGUR} \neq \text{„FAUST“} \wedge \text{FIGUR} \neq \text{„Wallenstein“}} (\text{DA}))$$

- Anfragen wie „Welcher Dichter ist Schauspieler?“ oder „Welcher Dichter hat in einem seiner Stücke gespielt?“ können „eigentlich“ nicht beantwortet werden, da es keine systemkontrollierte Beziehung zwischen **Dichter** und **Schauspieler** gibt.



## Relationenalgebra (23)

- ACHTUNG: Connection Trap!
  - Verbund kann im Allg. nicht als Umkehroperation zur Projektion angesehen werden
  - Beispiel: DA1 und DA2 als Projektionen auf DA; DA3 als Verbund von DA1 und DA2

DA1	PNR	A-ORT
	P1	MA
	P1	KL
	P2	MA

DA	PNR	FIGUR	A-ORT
	P1	Faust	MA
	P1	Mephisto	KL
	P2	Wallenstein	MA

DA2	FIGUR	A-ORT
	Faust	MA
	Mephisto	KL
	Wallenstein	MA

DA3	PNR	FIGUR	A-ORT
	P1	Faust	MA
	P1	Wallenstein	MA
	P1	Mephisto	KL
	P2	Faust	MA
	P2	Wallenstein	MA

N. Ritter, DIS, SS 2005, Kapitel 4

53



## Algebraische Optimierung (1)

- Relationenalgebraische Formulierungen spezifizieren Ausführungsreihenfolge (prozedurale Elemente), äquivalente Umformungen möglich
- **Optimierungsproblem**
  - gegeben: Ausdruck der Relationenalgebra (RA)
  - gesucht: äquivalenter, möglichst effizient auszuführender RA-Ausdruck
  - Bestimmung einer möglichst guten Ausführungsreihenfolge (Einsatz von Heuristiken)
- **Statistische Kenngrößen** werden dem DB-Katalog entnommen
  - $N_i = \text{Card}(R_i)$
  - $j_i = \text{Anzahl der verschiedenen Werte eines Attributs } A_i$

N. Ritter, DIS, SS 2005, Kapitel 4

54



## Algebraische Optimierung (2)

### ■ Rewrite-Regeln

#### ■ Kommutativgesetz für Produkte und Verbunde

- $R1 \times R2 \equiv R2 \times R1$
- $R1 \bowtie R2 \equiv R2 \bowtie R1$

#### ■ Assoziativgesetz für Produkte und Verbunde

- $(R1 \times R2) \times R3 \equiv R1 \times (R2 \times R3)$
- $(R1 \bowtie R2) \bowtie R3 \equiv R1 \bowtie (R2 \bowtie R3)$

#### ■ Zusammenfassung von Folgen von Projektionen

- $\pi_{A,B,C} (\pi_{A,B,C,\dots,Z} (SP)) \equiv \pi_{A,B,C} (SP)$

#### ■ Zusammenfassung von Folgen von Selektionen

- $\sigma_{F1} (\sigma_{F2} (R)) \equiv \sigma_{F1 \wedge F2} (R) \equiv \sigma_{F2 \wedge F1} (R) \equiv \sigma_{F2} (\sigma_{F1} (R))$

$\bowtie$   
steht hier für  
beliebige  
 $\theta$ -Verbunde



## Algebraische Optimierung (3)

### ■ Rewrite-Regeln (Forts.)

#### ■ Vertauschung von Selektionen und Projektionen

- F enthält nur Attribute aus A, ..., Z:

$$\sigma_F (\pi_{A, \dots, Z} (R)) \equiv \pi_{A, \dots, Z} (\sigma_F (R))$$

- F enthält Attribute aus A, ..., Z, B1, ..., Bm:

$$\pi_{A, \dots, Z} (\sigma_F (R)) \equiv \pi_{A, \dots, Z} (\sigma_F (\pi_{A, \dots, Z, B1, \dots, Bm} (R)))$$

#### ■ Vertauschung von Selektion und Kartesischem Produkt

- F enthält nur Attribute aus R1:

$$\sigma_F (R1 \times R2) \equiv \sigma_F (R1) \times R2$$

- allgemeiner:  $F = F1 \wedge F2 \wedge F3$

F1 nur auf R1, F2 nur auf R2, F3 auf beiden

$$\sigma_F (R1 \times R2) \equiv \sigma_{F1} (R1) \bowtie_{F3} \sigma_{F2} (R2)$$



## Algebraische Optimierung (4)

- Annahmen
  - Gleichverteilung der Attributwerte eines Attributes
  - Stochastische Unabhängigkeit der Werte verschiedener Attribute
- Selektivitätsfaktor (SF)
  - basiert auf statistischen Werten
  - beschreibt hinsichtlich eines Qualifikationsprädikats den erwarteten Anteil an Tupeln, die das Prädikat erfüllen
  - $0 \leq SF \leq 1$
  - $\text{Card}(\sigma_p(R)) = SF(p) \cdot \text{Card}(R)$



## Algebraische Optimierung (5)

### ■ SF-Berechnung

$j_i$ : Anzahl der Werte des Attributs  $A_i$

■ $A_i = a_i$	$SF = \begin{cases} 1/j_i & \text{falls } j_i \text{ bekannt, z.B. Index auf } A_i \\ 1/10 & \text{sonst} \end{cases}$
■ $A_i = A_k$	$SF = \begin{cases} 1/\max(j_i, j_k) & \text{falls } j_i \text{ und } j_k \text{ bekannt} \\ 1/j_i & \text{falls nur } j_i \text{ bekannt} \\ 1/10 & \text{sonst} \end{cases}$
■ $A_i \geq a_i$	$SF = \begin{cases} (\text{high-key} - a_i) / (\text{high-key} - \text{low-key}) & \text{falls bekannt} \\ 1/3 & \text{sonst} \end{cases}$
■ $A_i \geq a_i \wedge A_i \leq a_k$	$SF = \begin{cases} (a_k - a_i) / (\text{high-key} - \text{low-key}) & \text{falls bekannt} \\ 1/4 & \text{sonst} \end{cases}$



## Algebraische Optimierung (6)

- **SF-Berechnung bei Ausdrücken**
  - $SF(p(A) \wedge p(B)) = SF(p(A)) \cdot SF(p(B))$
  - $SF(p(A) \vee p(B)) = SF(p(A)) + SF(p(B)) - SF(p(A)) \cdot SF(p(B))$
  - $SF(\neg p(A)) = 1 - SF(p(A))$
- **Join-Selektivitätsfaktor (JSF)**
  - $Card(RS) = JSF * Card(R) * Card(S)$
  - bei (N:1)-Joins (verlustfrei):  $Card(RS) = \text{Max}(Card(R), Card(S))$



## Algebraische Optimierung (7)

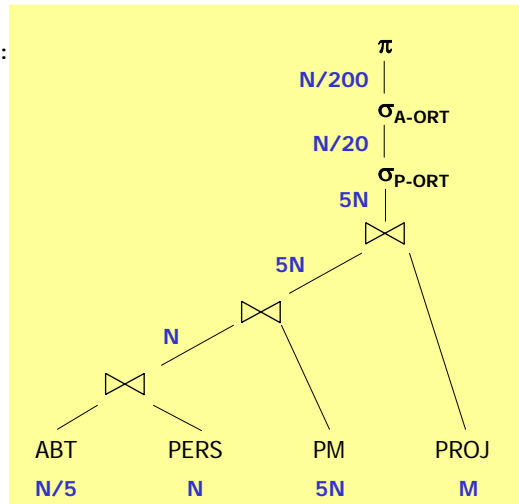
- **Beispiel**
  - **DB-Schema**
    - ABT (ANR, BUDGET, A-ORT)
    - PERS (PNR, NAME, BERUF, GEHALT, ALTER, ANR)
    - PM (PNR, JNR, DAUER, ANTEIL)
    - PROJ (JNR, BEZEICHNUNG, SUMME, P-ORT)
  - **Anfrage:** *Finde Name und Beruf von Angestellten, deren Abteilung in KL ist und die in KL Projekte durchführen.*
  - **Annahmen:**
    - ABT: N/5 Tupel
    - PERS: N Tupel
    - PM: 5N Tupel
    - PROJ: M Tupel
    - Anzahl der Attributwerte von A-ORT: 10, P-ORT: 100
    - Verlustfreie Verbunde von R1 und R2 über Primär-/Fremdschlüssel



## Algebraische Optimierung (8)

- Beispiel (Forts.)

- Ausgangslösung:



N. Ritter, DIS, SS 2005, Kapitel 4

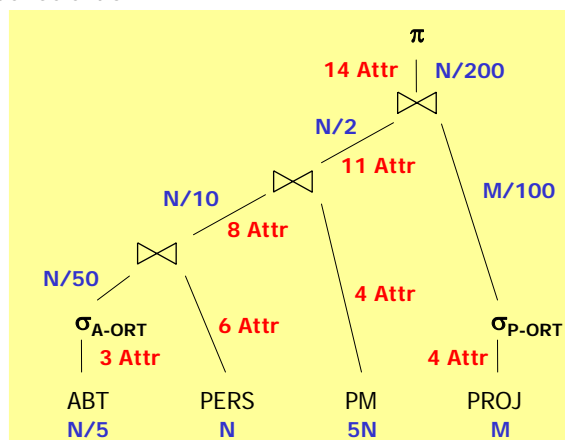
61



## Algebraische Optimierung (9)

- Beispiel (Forts.)

- Verschieben der Selektion:



N. Ritter, DIS, SS 2005, Kapitel 4

62



## Algebraische Optimierung (9)

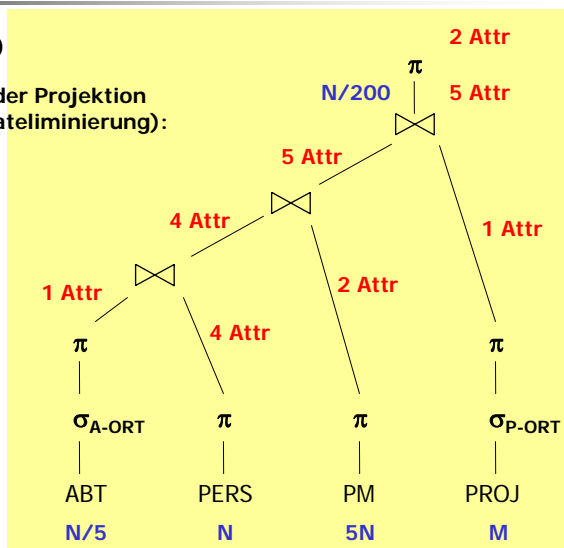
- Beispiel (Forts.)
  - Verschieben der Selektion:

*I.*  
**Führe Selektion so früh  
wie möglich aus!**



## Algebraische Optimierung (10)

- Beispiel (Forts.)
  - Verschieben der Projektion  
(ohne Duplikateliminierung):







## Algebraische Optimierung (10)

- Beispiel (Forts.)
  - Verschieben der Projektion (ohne Duplikateliminierung):

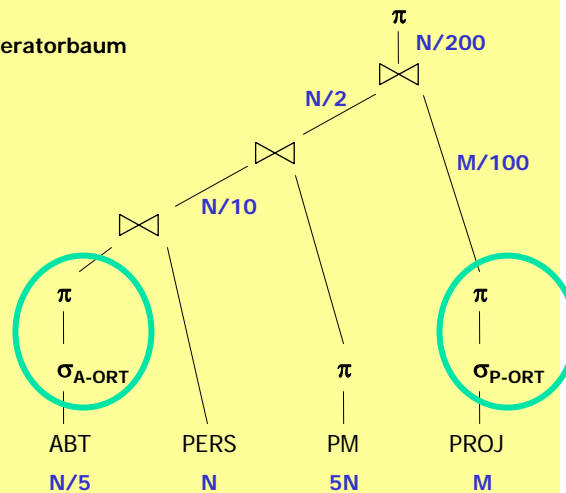
*II.  
Führe Projektion  
(ohne Duplikateliminierung)  
so früh wie möglich aus!*

Bem.: Der Nutzen einer frühzeitigen Projektionsausführung hängt von mehreren Faktoren ab.



## Algebraische Optimierung (11)

- Beispiel (Forts.)
  - Optimierter Operatorbaum (Vorschlag):





## Algebraische Optimierung (11)

- **Beispiel (Forts.)**
  - **Optimierter Operatorbaum (Vorschlag):**

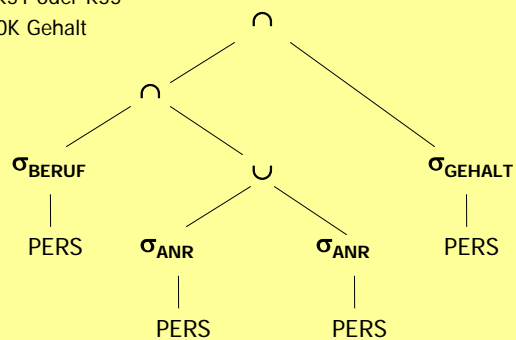
*III.  
Verknüpfe Folgen von unären  
Operatoren wie Selektion und  
Projektion (wenn diese tupelweise  
abgewickelt werden können)!*



## Algebraische Optimierung (12)

- **Weitere Optimierungsmaßnahmen**
  - **Ausdrucksauswertung**

- Beispiel: Finde alle Programmierer aus Abteilung K51 oder K55 mit mehr als 50K Gehalt





## Algebraische Optimierung (12)

- Weitere Optimierungsmaßnahmen (Forts.)

- **Ausdrucksauswertung**

- Beispiel: Finde alle Programmierer aus Abteilung K51 oder K55 mit mehr als 50K Gehalt

$$\sigma_{\text{BERUF}=\text{„P“} \wedge \text{GEHALT}>50\text{K} \wedge (\text{ANR}=\text{„K51“} \vee \text{ANR}=\text{„K55“})}$$

PERS



## Algebraische Optimierung (12)

- Weitere Optimierungsmaßnahmen (Forts.)

- **Ausdrucksauswertung**

- Beispiel: Finde alle Programmierer aus Abteilung K51 oder K55 mit mehr als 50K Gehalt

*IV.*

*Fasse einfache Selektionen auf einer Relation zusammen!*



## Algebraische Optimierung (13)

- **Weitere Optimierungsmaßnahmen (Forts.)**

*V.*

*Verknüpfe bestimmte Selektionen  
mit einem vorausgehenden  
Kartesischen Produkt  
zu einem Verbund!*

*VI.*

*Berechne gemeinsame Teilbäume  
nur einmal (wenn die Zwischen-  
speicherung der Ergebnisse nicht  
zu teuer ist)!*



## Algebraische Optimierung (14)

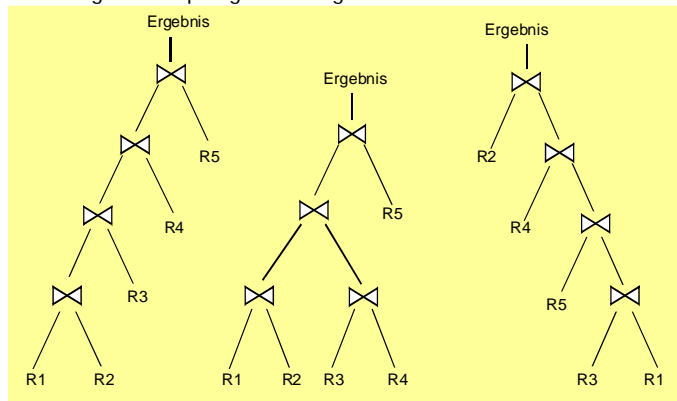
- **Weitere Optimierungsmaßnahmen (Forts.)**

- Kombination von Verbundoperationen
  - Assoziativität und Kommutativität von Verbundoperationen (gilt auch für Vereinigung und Durchschnitt)
  - **Allgemeines Problem bei binären Relationenoperationen**
    - Was ist die beste Verknüpfungsreihenfolge?
    - Im allgemeinen Fall sind  $n!$  Reihenfolgen möglich
    - Die genaue Größe einer Zwischenrelation ergibt sich erst nach Ende der erzeugenden Operation
      - Dynamische Entscheidung aufwendiger, aber genauer als Abschätzung
      - Bei jedem Auswertungsschritt werden die momentan kleinsten (Zwischen-)Relationen ausgewählt



## Algebraische Optimierung (15)

- Weitere Optimierungsmaßnahmen (Forts.)
  - Kombination von Verbundoperationen (Forts.)
    - Allgemeines Problem bei binären Relationenoperationen (Forts.)
      - Einige Verknüpfungsreihenfolgen für den Verbund mit  $n=5$



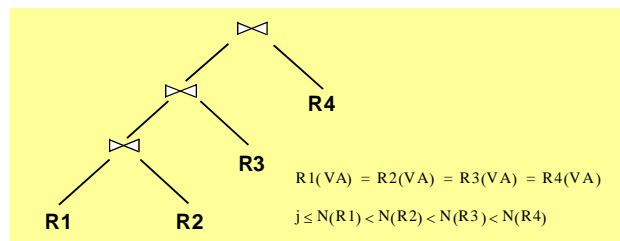
N. Ritter, DIS

73



## Algebraische Optimierung (16)

- Weitere Optimierungsmaßnahmen (Forts.)
  - Kombination von Verbundoperationen (Forts.)
    - Abschätzung
      - beim (1:n)-Verbund:  $N(R1)=j \Rightarrow N(T1)=N(R2)$
    - Bestimmung der Verbundreihenfolge (Heuristik):



**VII. Bestimme die Verbundreihenfolge so, dass die**

N. Ritter

**Anzahl und Größe der Zwischenobjekte minimiert wird!**

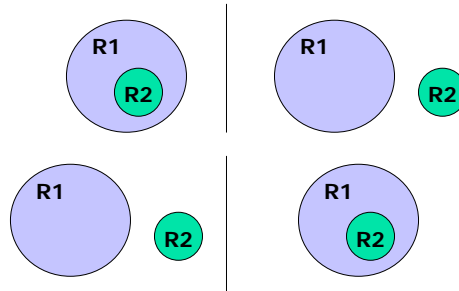
'4



## Algebraische Optimierung (17)

### ■ Weitere Optimierungsmaßnahmen (Forts.)

- Reihenfolge von Mengenoperationen
- Kardinalität der Vereinigung:  
 $\max(N(R1), N(R2))$   
 $\leq N(R1 \cup R2)$   
 $\leq N(R1) + N(R2)$
- Kardinalität des Durchschnitts:  
 $0$   
 $\leq N(R1 \cap R2)$   
 $\leq \min(N(R1), N(R2))$



*VIII. Verknüpfte bei Mengenoperationen immer  
zuerst die kleinsten Relationen!*

N. Ritter

'5



## Algebraische Optimierung (18)

### ■ Heuristische Regeln:

- Führe Selektion so früh wie möglich aus
- Führe Projektion (ohne Duplikateliminierung) frühzeitig aus
- Verknüpfe Folgen von unären Operationen wie Selektion und Projektion
- Fasse einfache Selektionen auf einer Relation zusammen
- Verknüpfe bestimmte Selektionen mit einem vorausgehenden Kartesischen Produkt zu einem Verbund
- Berechne gemeinsame Teilbäume nur einmal
- Bestimme Verbundreihenfolge so, dass die Anzahl und Größe der Zwischenobjekte minimiert wird
- Verknüpfe bei Mengenoperationen immer zuerst die kleinsten Relationen

N. Ritter, DIS, SS 2005, Kapitel 4

76



## Weitere Operatoren (1)

- Division
  - Ziel
    - Beantwortung von Fragen, bei denen eine „ganze Relation“ zur Qualifikation herangezogen wird
    - Simulation des Allquantors  $\Rightarrow$  ein Tupel aus R steht mit allen Tupeln aus S in einer bestimmten Beziehung
  - Definition
    - Sei R vom Grad r und S vom Grad s,  $r > s$  und  $s \neq 0$ ;  
t sei (r-s)-Tupel, u sei s-Tupel;  
S-Attribute  $\subset$  R-Attribute;
    - Dann gilt:  $R \div S = \{t \mid \forall u \in S: (t \cup u \in R)\}$



## Weitere Operatoren (2)

- Division (Forts.)

- Beispiel:

DA	PNR	FIGUR	A-Jahr ...
	P1	Faust	1999
	P1	Nathan	1998
	P2	Werther	1997
	P3	Faust	1998
	P3	Nathan	1999
	P3	Werther	1998

RO	FIGUR	TITEL	R-Typ
	Faust	Faust	
	Nathan	Nathan der Weise ...	
	Werther	Die Leiden ...	

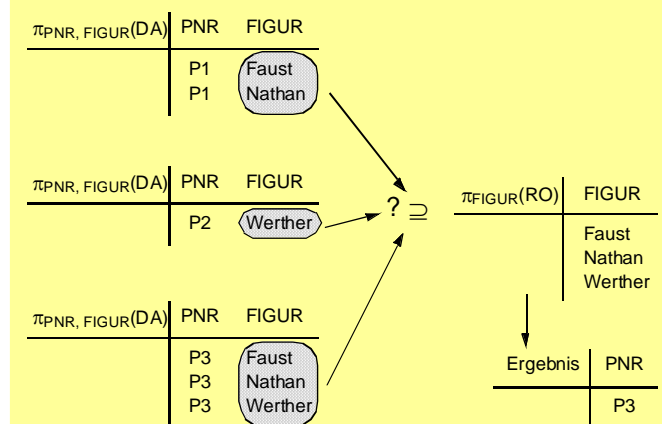
- Welche Schauspieler haben alle Rollen gespielt:

$$(\pi_{\text{PNR, FIGUR}}(\text{DA})) \div (\pi_{\text{FIGUR}}(\text{RO}))$$



## Weitere Operatoren (3)

- Division (Forts.)
  - Beispiel (Forts.)
    - Welche Schauspieler haben alle Rollen gespielt:  $(\pi_{\text{PNR, FIGUR}}(\text{DA})) \div (\pi_{\text{FIGUR}}(\text{RO}))$



N. Ritter, DIS

79



## Weitere Operatoren (4)

- Division (Forts.)
  - Beschreibung der Division mit den Grundoperatoren
    - $T = \pi_{1, 2, \dots, r-s}(R)$
    - $W = (T \times S) - R$
    - $V = \pi_{1, 2, \dots, r-s}(W)$
    - $R \div S = T - V$
    - $= \pi_{1, 2, \dots, r-s}(R) - \pi_{1, 2, \dots, r-s}((\pi_{1, 2, \dots, r-s}(R) \times S) - R)$
  - Es gilt:  $(R \times S) \div S = R$

N. Ritter, DIS, SS 2005, Kapitel 4

80





## Weitere Operatoren (5)

- Division (Forts.)

- Weitere Beispiele

- Finde alle Schauspieler (NAME), die **alle** Rollen in Dramen von Goethe gespielt haben.

$$\pi_{\text{Name}}( (SP \bowtie (\pi_{\text{PNR,FIGUR}}(DA))) \div (\pi_{\text{FIGUR}}(RO \bowtie (\sigma_{\text{AUTOR}=\text{Goethe}}(DR)))) )$$

- Finde alle Schauspieler (NAME), die **alle** Narrenrollen am Pfalztheater gespielt haben.

$$\pi_{\text{Name}}( (SP \bowtie (\pi_{\text{PNR,FIGUR}}(\sigma_{\text{THEATER}=\dots}(DA))) \div (\pi_{\text{FIGUR}}(\sigma_{\text{R-Typ}=\dots}(RO))) )$$



## Weitere Operatoren (6)

- Intervallverbund (Band Join)

- Anstatt des arithmetischen Vergleichsoperators  $\Theta$  des  $\Theta$ -Joins wird hier eine Intervall-Bedingung überprüft.
  - Grob: Kartesisches Produkt zwischen zwei Relationen R (Grad r) und S (Grad s) eingeschränkt durch eine Intervall-Bedingung zwischen i-Spalte von R und j-Spalte von S.
  - Intervall  $I = [c_1, c_2]$  mit  $c_1, c_2$  positive Konstanten, wobei eine größer Null sein muss.
  - Intervall-Verbund zwischen R und S:

$$V = R \bowtie_{ij} S = \sigma_{ij} (R \times S) = \sigma_{R.i-c_1 \leq S.j \leq R.i+c_2} (R \times S)$$



## Weitere Operatoren (7)

- Intervallverbund (*Band Join*)
  - Bemerkung
    - Ein Tupel  $s$  aus  $S$  'kombiniert' mit einem Tupel  $r$  aus  $R$  nur, wenn der Wert der  $j$ -Spalte von  $S$  im Intervall der Größe  $c_1+c_2$  um den Wert der  $i$ -Spalte von  $R$  liegt.
    - Beispiel:  $G = \sigma_{\text{PNR} \neq \text{PNR}' \wedge \text{ALTER} \in [\text{ALTER}' - 2, \text{ALTER}' + 2]} (\text{PERS} \bowtie \text{PERS}')$

PERS	PNR	ALTER
	P1	25
	P2	23
	P3	28

G	PNR	ALTER	PNR'	ALTER'
	P1	25	P2	23
	P2	23	P1	25



## Weitere Operatoren (8)

- Äußerer Verbund (*Outer Join*)
  - Ziel: ‚Verlustfreiheit‘ soll erzwungen werden!
  - Trick: Einfügen spezieller Leerzeilen zur künstlichen Erzeugung von Verbundpartnern
  - Beispiel

SP	PNR	NAME
	P1	x
	P2	y

DA	PNR	FIGUR
	P1	F
	P1	W
	P3	M

$$SP' = SP \cup ((\pi_{\text{PNR}}(\text{DA}) - \pi_{\text{PNR}}(\text{SP})) \times \equiv)$$

$$DA' = DA \cup ((\pi_{\text{PNR}}(\text{SP}) - \pi_{\text{PNR}}(\text{DA})) \times \equiv)$$



## Weitere Operatoren (9)

### ■ Äußerer Verbund (Forts.)

#### ■ Definition

- Seien A die Verbundattribute, „≡“ der undefinierte Wert und

$$R' = R \cup ((\pi_A(S) - \pi_A(R)) \times \equiv \times \dots)$$

$$S' = S \cup ((\pi_A(R) - \pi_A(S)) \times \equiv \times \dots)$$

Äußerer Gleichverbund:

$$R \begin{array}{c} \sqcap \\ \times \\ \sqcap \end{array} S := R' \begin{array}{c} \times \\ \times \\ \times \end{array} S' \\ R.A=S.A \quad R'.A=S'.A$$

Äußerer natürlicher Verbund:

$$R \begin{array}{c} \sqcap \\ \times \\ \sqcap \end{array} S := R' \begin{array}{c} \times \\ \times \\ \times \end{array} S'$$



## Weitere Operatoren (10)

### ■ Äußerer Verbund (Forts.)

#### ■ Linker äußerer Gleichverbund

- Bei dieser Operation bleibt die linke Argumentrelation verlustfrei, d. h., bei Bedarf wird ein Tupel durch „NULL“-Werte „nach rechts“ aufgefüllt

$$R \begin{array}{c} \sqcap \\ \times \\ \sqcap \end{array} S := R \begin{array}{c} \times \\ \times \\ \times \end{array} S' \\ R.A=S.A \quad R.A=S'.A$$

#### ■ Rechter äußerer Gleichverbund

- Dabei bleibt analog die rechte Argumentrelation verlustfrei; fehlende Partnertupel werden durch Auffüllen mit „NULL“-Werten „nach links“ ergänzt

$$R \begin{array}{c} \times \\ \times \\ \times \end{array} S := R' \begin{array}{c} \sqcap \\ \times \\ \sqcap \end{array} S \\ R.A=S.A \quad R'.A=S.A$$

## Weitere Operatoren (11)

- Äußerer Verbund (Forts.)

- Beispiele:

R	A	B	C
a <sub>1</sub>	b <sub>1</sub>	c <sub>1</sub>	
a <sub>2</sub>	b <sub>2</sub>	c <sub>2</sub>	

S	C	D	E
	c <sub>1</sub>	d <sub>1</sub>	e <sub>1</sub>
	c <sub>3</sub>	d <sub>2</sub>	e <sub>2</sub>

ERG	A	B	C	D	E
	a <sub>1</sub>	b <sub>1</sub>	c <sub>1</sub>	d <sub>1</sub>	e <sub>1</sub>

ERG	A	B	C	D	E
	a <sub>1</sub>	b <sub>1</sub>	c <sub>1</sub>	d <sub>1</sub>	e <sub>1</sub>
	--	--	c <sub>3</sub>	d <sub>2</sub>	e <sub>2</sub>

ERG	A	B	C	D	E
	a <sub>1</sub>	b <sub>1</sub>	c <sub>1</sub>	d <sub>1</sub>	e <sub>1</sub>
	a <sub>2</sub>	b <sub>2</sub>	c <sub>2</sub>	--	--

ERG	A	B	C	D	E
	a <sub>1</sub>	b <sub>1</sub>	c <sub>1</sub>	d <sub>1</sub>	e <sub>1</sub>
	a <sub>2</sub>	b <sub>2</sub>	c <sub>2</sub>	--	--
	--	--	c <sub>3</sub>	d <sub>2</sub>	e <sub>2</sub>

N. Ritter, DIS, SS 2005, Kapitel 4

87

## Weitere Operatoren (12)

- Äußere Vereinigung (*Outer Union*)

- Diese Operation erlaubt die Vereinigung zweier Relationen, die nicht vereinigungsverträglich sind. Wenn zwei Relationen partiell verträglich sind, d.h., einige ihrer Attribute sind vereinigungsverträglich, dann kann OUTER UNION angewendet werden.

- Beispiel:

STUDENT	MATRNR	FBNR	SEM
	123	FB5	5
	789	FB9	9

HIWI	MATRNR	FBNR	JOB
	465	FB5	TUTOR
	987	FB9	PROGR

STUD-HIWI	MATRNR	FBNR	SEM	JOB
	123	FB5	5	-
	789	FB9	9	-
	465	FB5	-	TUTOR
	987	FB9	-	PROGR

Achtung: Interpretationsprobleme beim Ergebnis!

In ähnlicher Weise lassen sich weitere Operationen einführen (Outer Intersection/Difference), die jedoch nicht besonders nützlich sind!

N. Ritter, DIS, SS 2005, Kapitel 4

88



# Relationenalgebra

## ■ Zusammenfassung

- Algebra mit Auswahlvermögen der Prädikatenlogik 1. Stufe
- Abgeschlossenheit bzgl. der Algebraoperationen
- Klassische Mengenoperationen
- Relationenoperationen

