



## 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)



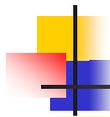
## Übersicht (2)

- **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)

N. Ritter, HMS, 13.09.2007

5



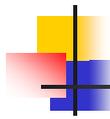
## Grundkonzepte (3)

### ■ 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

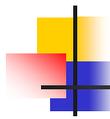
N. Ritter, HMS, 13.09.2007

6



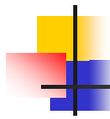
## Grundkonzepte (4)

- **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.



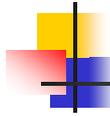
## Grundkonzepte (6)

- **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.
  - 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).



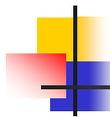
## Grundkonzepte (7)

- **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



## 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 = \text{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

N. Ritter, HMS, 13.09.2007

13

## Relationenalgebra (4)

- **Projektion:  $\pi$**

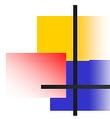
- **Auswahl von Spalten (Attribute)  $A_1, A_2, \dots, A_k$  aus einer Relation R (Grad  $n \geq k$ )**
- $\pi_{A_1, A_2, \dots, A_k}(R) = \{ p \mid \exists t \in R : p = \langle t[A_1], \dots, t[A_k] \rangle \}$   
(Alternative: Benutzung von Spaltennummern)
- Duplikateliminierung
- Beispiel:

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

N. Ritter, HMS, 13.09.2007

14

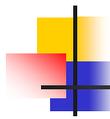


## Relationenalgebra (5)

- 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:  $R \triangleright S = (R \cup S) - (R \cap S)$

N. Ritter, HMS, 13.09.2007

15



## Relationenalgebra (6)

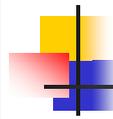
- Erweitertes Kartesisches Produkt
  - $K = R \times S = \{k \mid \exists x \in R, y \in S: (k = x | y)\}$   
mit  $x | y = \langle x_1, \dots, x_r, y_1, \dots, y_s \rangle$ ,

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

N. Ritter, HMS, 13.09.2007



## 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, HMS, 13.09.2007

17



## 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:

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

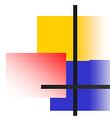
$$\pi_{PNR, ANR, ALTER} (ABT \bowtie PERS) = PERS;$$

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, HMS, 13.09.2007

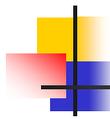


## Relationenalgebra (9)

- 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:

$$N = R \bowtie S =$$

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

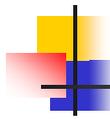


## Relationenalgebra (10)

- 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

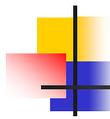


## Relationenalgebra (11)

- 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  $\text{Card}(R1) < \text{Card}(R2)$ :  $\text{Card}(R1 \bowtie R2) = \text{Card}(R2)$

N. Ritter, HMS, 13.09.2007

21



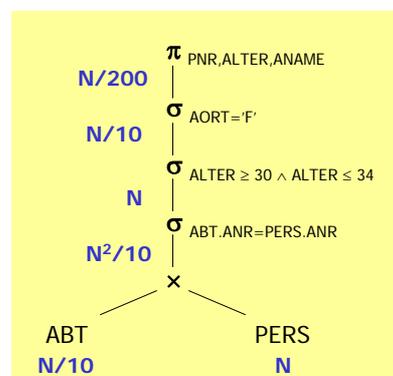
## Relationenalgebra (12)

- Natürlicher Verbund – Beispiel (Forts.)

- **Lösung 1:**

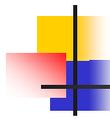
$\pi_{\text{PNR,ALTER,ANAME}}(\sigma_{\text{AORT}='F'}(\sigma_{\text{ALTER} \geq 30 \wedge \text{ALTER} \leq 34}(\sigma_{\text{ABT.ANR}=\text{PERS.ANR}}(\text{ABT} \times \text{PERS}}))))$

zugehöriger Operatorbaum:



N. Ritter, HMS, 13.09.2007

22



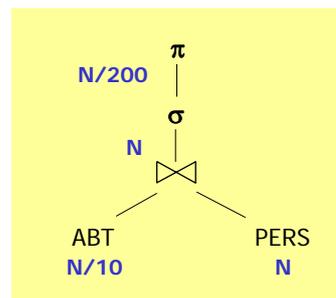
## Relationenalgebra (13)

- 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:



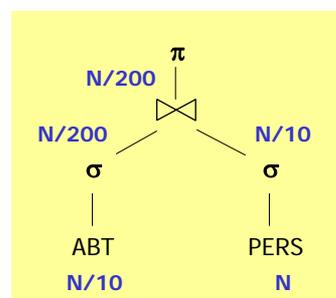
## Relationenalgebra (14)

- 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:



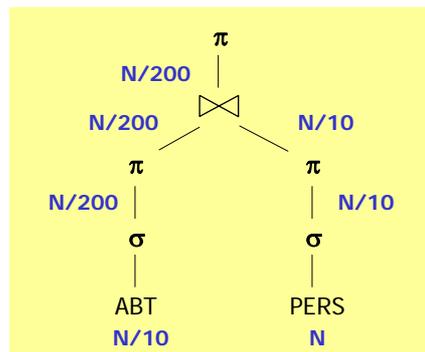
## Relationenalgebra (15)

- 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:

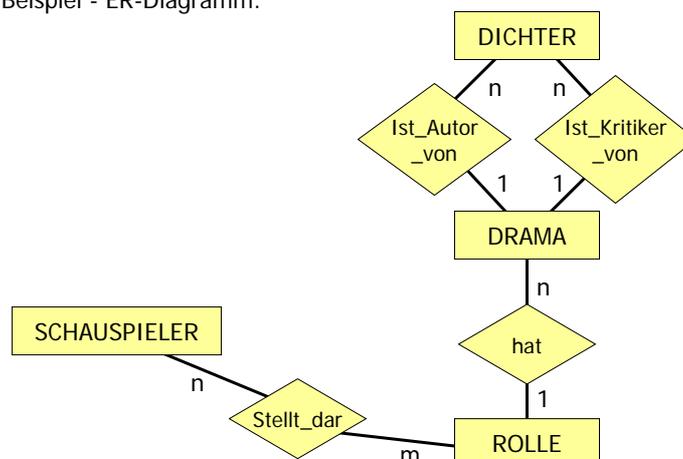


N. Ritter, HMS, 13.09.2007

25

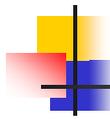
## Relationenalgebra (16)

- Beispiel - ER-Diagramm:



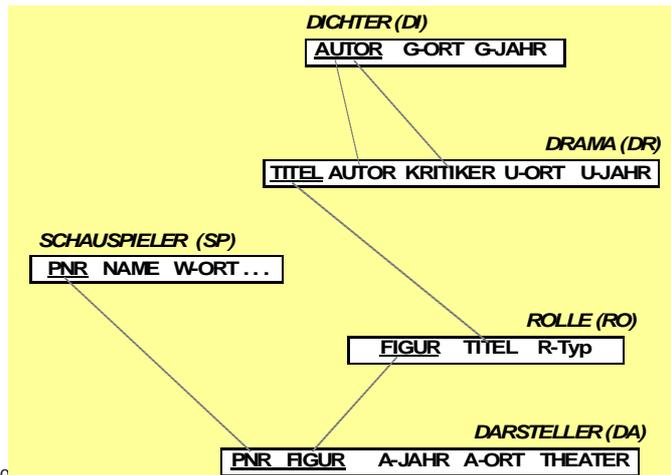
N. Ritter, HMS, 13.09.2007

26



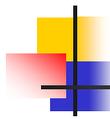
## Relationenalgebra (17)

- Beispiel – relationales DB-Schema:



N. Ritter, HMS, 13.0

27



## 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.

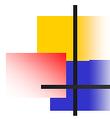
$$\pi_{\text{NAME}} (\sigma_{\text{TITEL} = \text{„Faust“}} (\text{SP} \bowtie_{\text{PNR}} \text{DA} \bowtie_{\text{FIGUR}} \text{RO}))$$

- 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, HMS, 13.09.2007

28



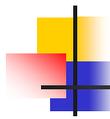
## Relationenalgebra (19)

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

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

N. Ritter, HMS, 13.09.2007

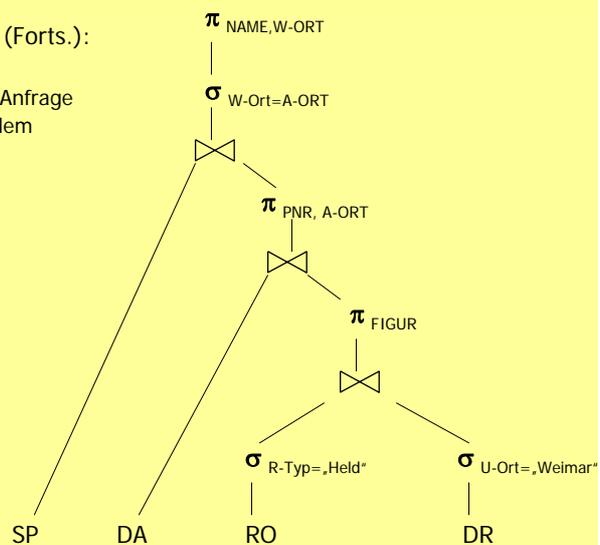
29



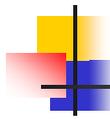
## Relationenalgebra (20)

- Beispiel – Anfragen (Forts.):

- Vorangegangene Anfrage entspricht folgendem Operatorbaum:



N. Ritter, HMS, 13.09.2007



## 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.):

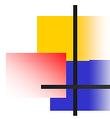
- Finde die Schauspieler, die **nie** gespielt haben.

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

- 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, HMS, 13.09.2007

33

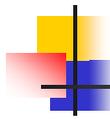


## Algebraische Optimierung (1)

- 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, HMS, 13.09.2007

34



## 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

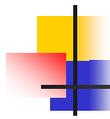
- F enthält nur Attribute aus R1:

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

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

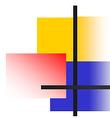
F1 nur auf R1, F2 nur auf R2, F3 auf beiden

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



## 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
  - $0 \leq SF \leq 1$
  - $\text{Card}(\sigma_p(R)) = SF(p) \cdot \text{Card}(R)$

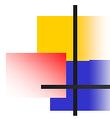


## Algebraische Optimierung (5)

### ■ SF-Berechnung

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

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



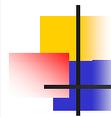
## Algebraische Optimierung (6)

- **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)

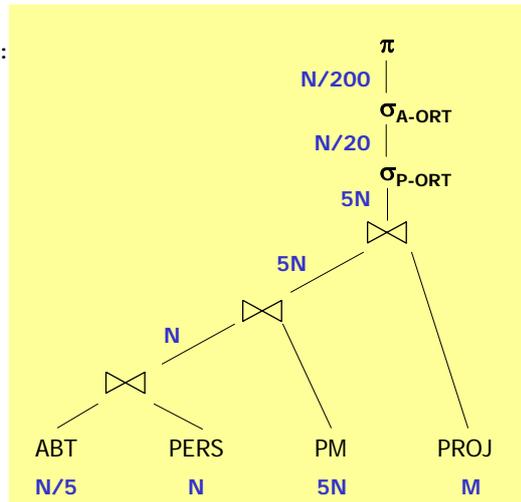
- **Beispiel**
  - **DB-Schema**
    - ABT (ANR, BUDGET, A-ORT)
    - PERS (PNR, NAME, BERUF, GEHALT, ALTER, ANR)
    - PM (PNR, JNR, DAUER, ANTEIL)
    - PROJ (JNR, BEZEICHNUNG, SUMME, P-ORT)
  - **Anfrage:** *Finde Name und Beruf von Angestellten, deren Abteilung in KL ist und die in KL Projekte durchführen.*
  - **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)

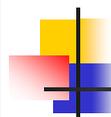
- Beispiel (Forts.)

- Ausgangslösung:



N. Ritter, HMS, 13.09.2007

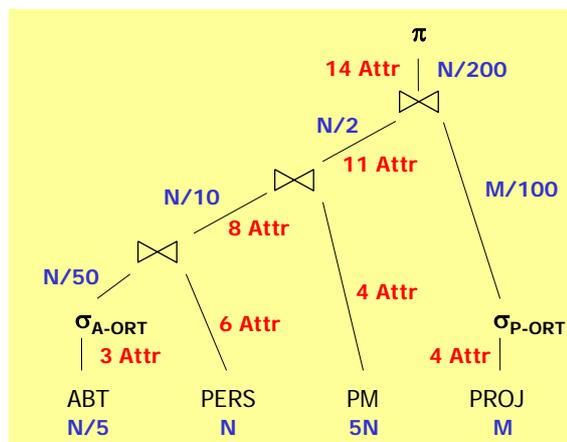
41



## Algebraische Optimierung (9)

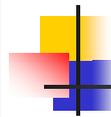
- Beispiel (Forts.)

- Verschieben der Selektion:



N. Ritter, HMS, 13.09.2007

42



## Algebraische Optimierung (9)

- Beispiel (Forts.)
  - Verschieben der Selektion:

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

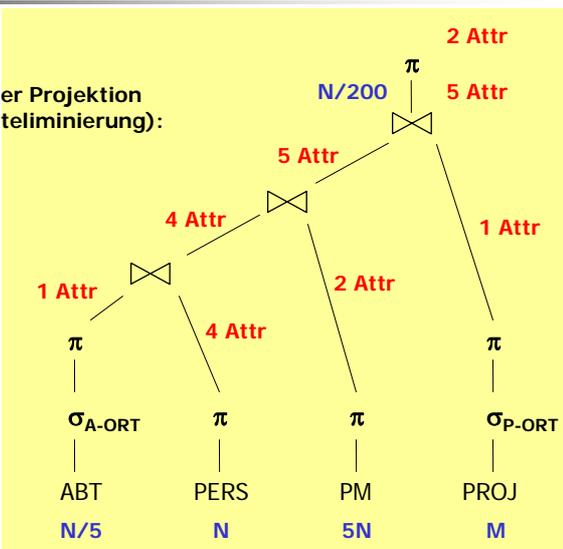
N. Ritter, HMS, 13.09.2007

43



## Algebraische Optimierung (10)

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



N. Ritter, HMS, 13.09.2007

44



## 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.

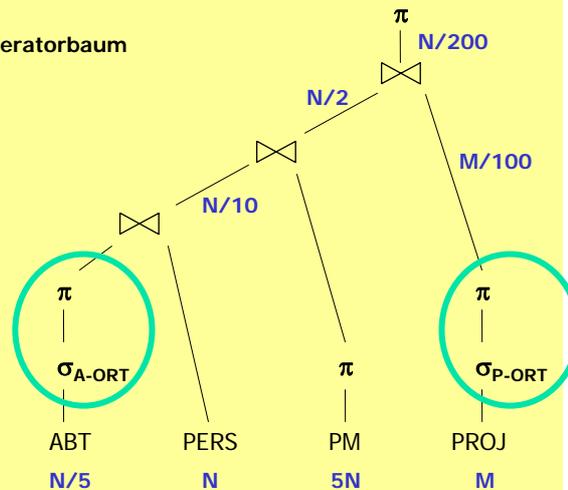
N. Ritter, HMS, 13.09.2007

45



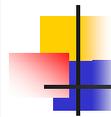
## Algebraische Optimierung (11)

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



N. Ritter, HMS, 13.09.2007

46



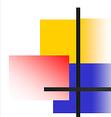
## Algebraische Optimierung (11)

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

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

N. Ritter, HMS, 13.09.2007

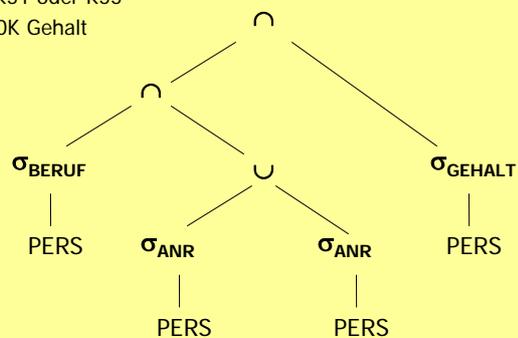
47



## Algebraische Optimierung (12)

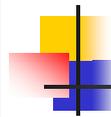
- **Weitere Optimierungsmaßnahmen**
  - **Ausdrucksauswertung**

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



N. Ritter, HMS, 13.09.2007

48



## Algebraische Optimierung (12)

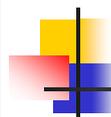
- Weitere Optimierungsmaßnahmen (Forts.)

- **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)

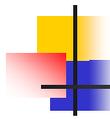
- Weitere Optimierungsmaßnahmen (Forts.)

- **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)!*

N. Ritter, HMS, 13.09.2007

51



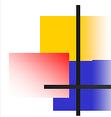
## Algebraische Optimierung (14)

- **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

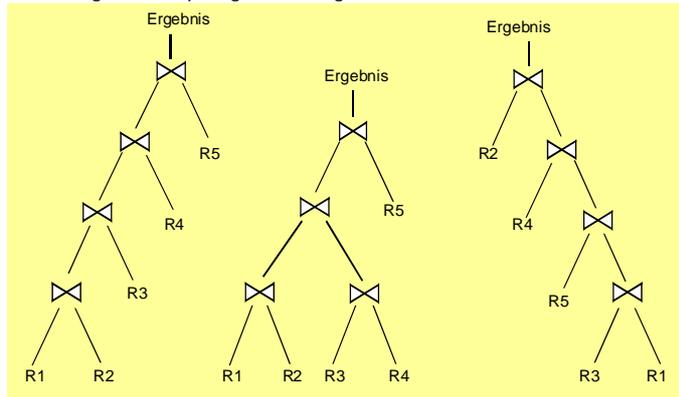
N. Ritter, HMS, 13.09.2007

52



## Algebraische Optimierung (15)

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



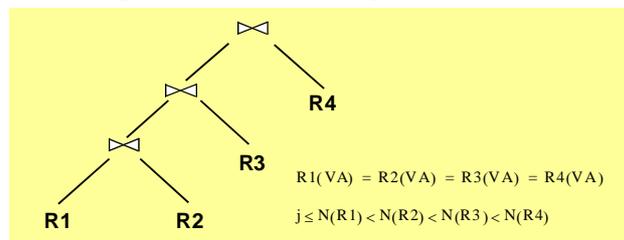
N. Ritter, I

53



## 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):

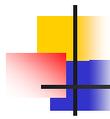


**VII. Bestimme die Verbundreihenfolge so, dass die**

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

N. Rit

54



## Algebraische Optimierung (17)

### ■ Weitere Optimierungsmaßnahmen (Forts.)

- Reihenfolge von Mengenoperationen

- Kardinalität der Vereinigung:

$$\max(N(R1), N(R2))$$

$$\leq N(R1 \cup R2)$$

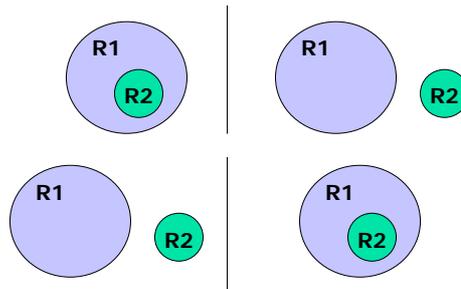
$$\leq N(R1) + N(R2)$$

- Kardinalität des Durchschnitts:

$$0$$

$$\leq N(R1 \cap R2)$$

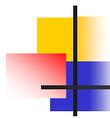
$$\leq \min(N(R1), N(R2))$$



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

N. Ril

55



## 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, HMS, 13.09.2007

56



## 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, HMS, 13.09.2007

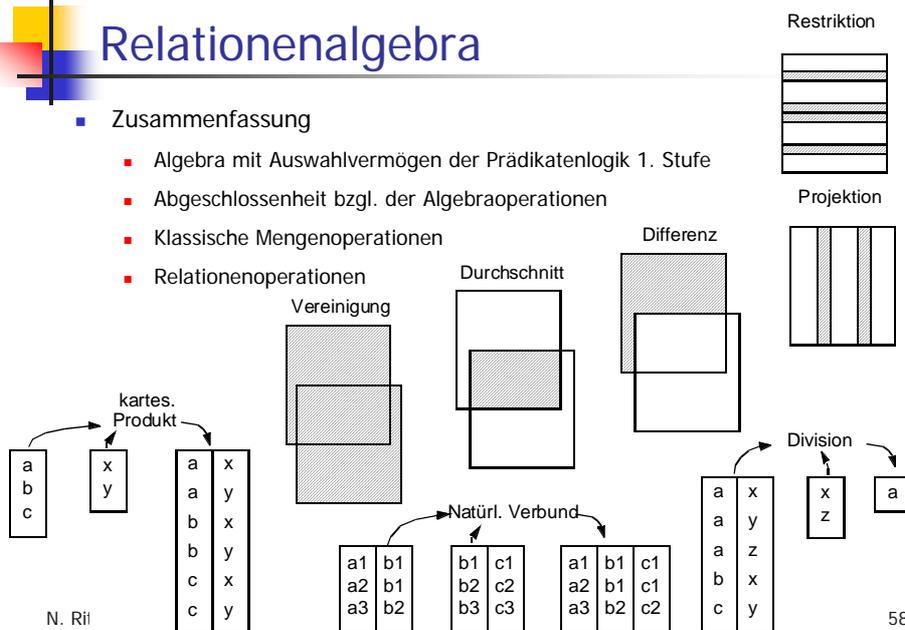
57



## Relationenalgebra

- Zusammenfassung

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



N. Rit

58