

Script generated by TTT

Title: groh: profile1 (08.05.2015)

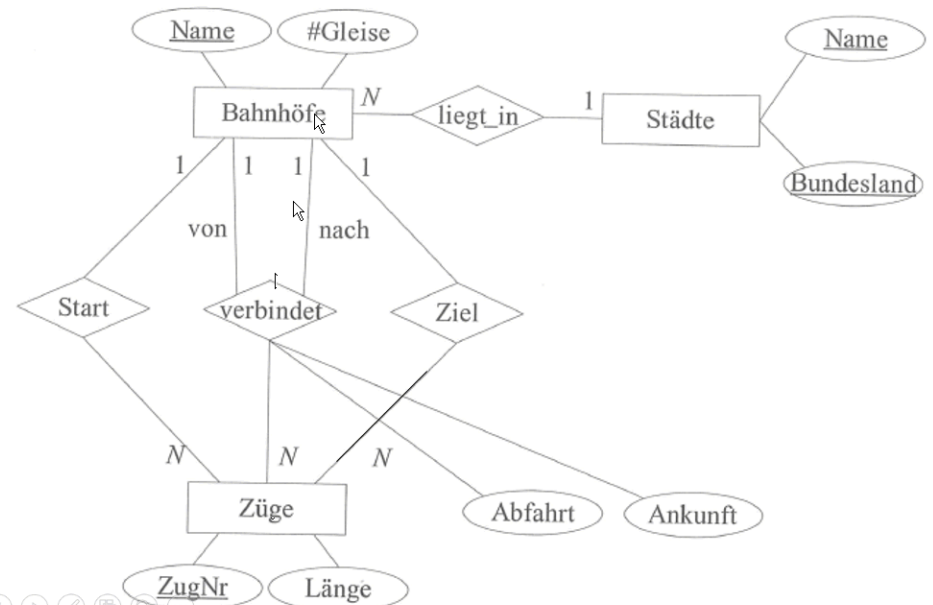
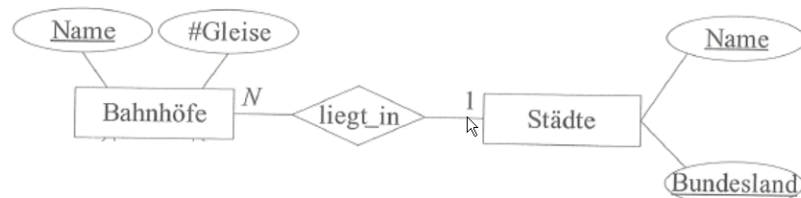
Date: Fri May 08 09:14:03 CEST 2015

Duration: 90:41 min

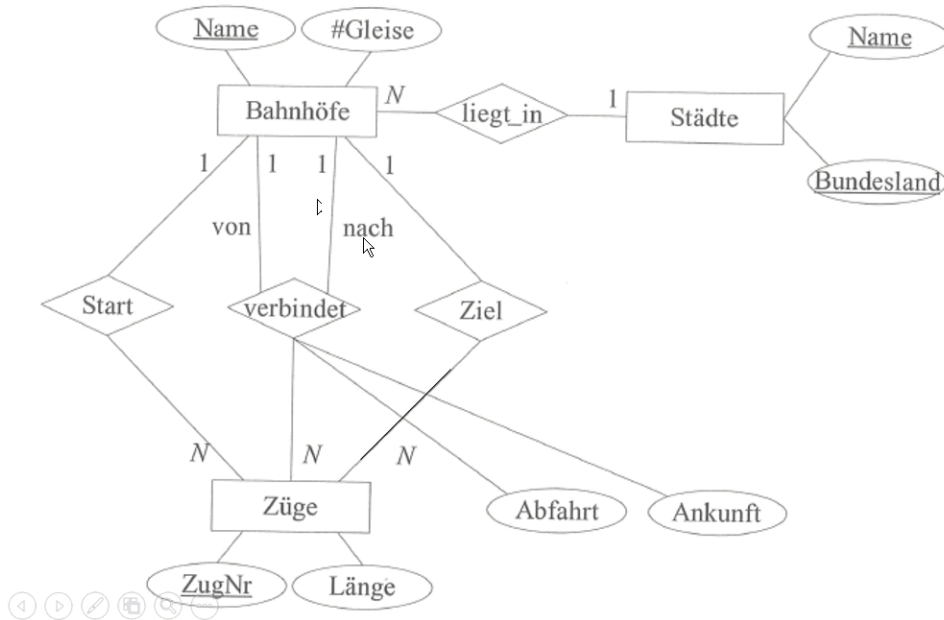
Pages: 72

Aufgabe 2.6 (aus [2])

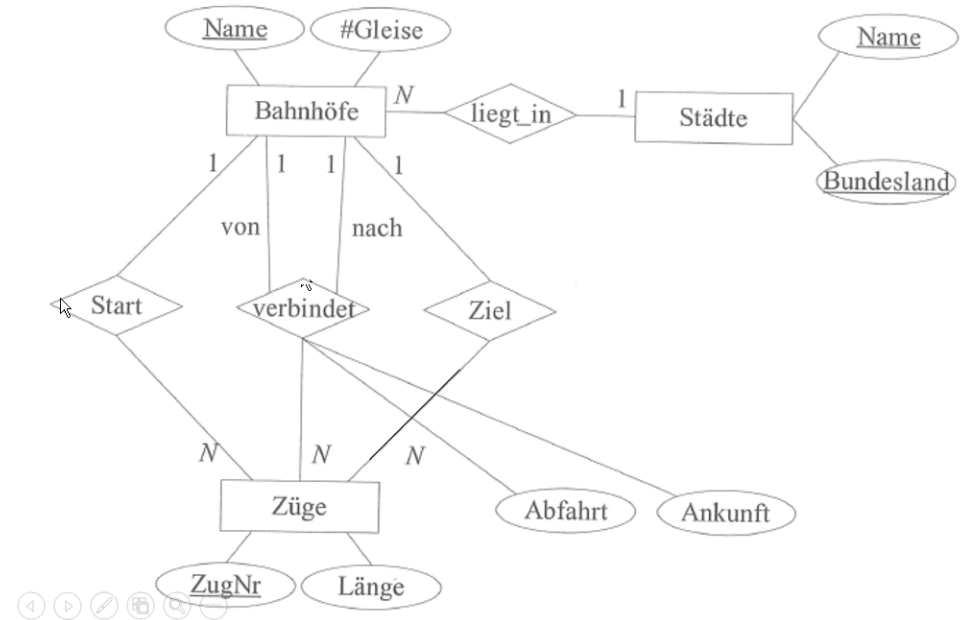
Modellieren Sie ein Zugauskunftssystem, in dem die wichtigsten Züge (z.B. die Intercity- und Eurocity-Züge) repräsentiert werden. Aus dem System sollen die Start- und Zielbahnhöfe und die durch den Zug verbundenen Bahnhöfe einschließlich Ankunfts- und Abfahrtszeiten ersichtlich sein. Geben Sie die Funktionalitäten der Beziehungstypen an.



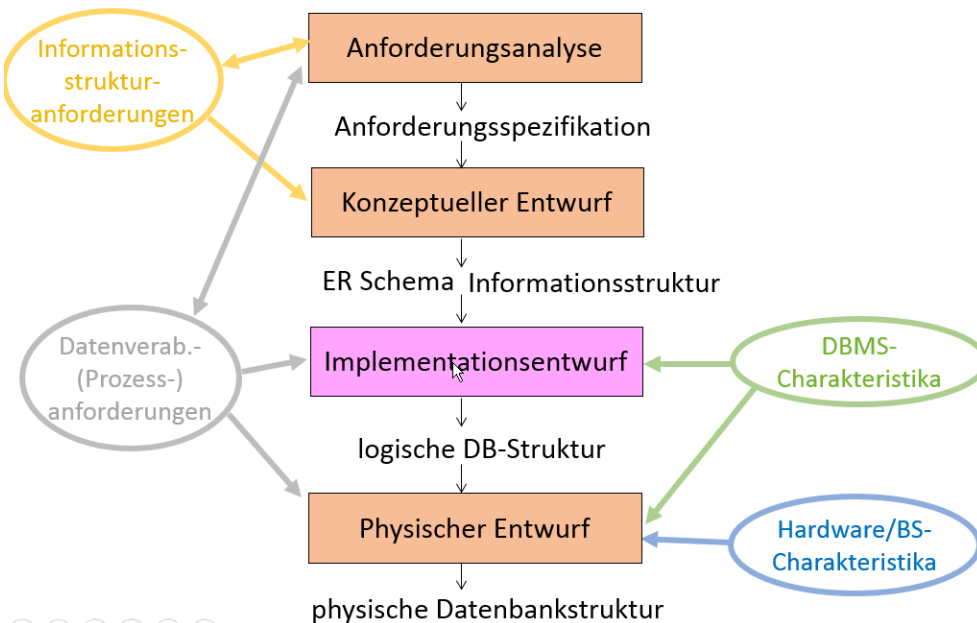
Beispiel: Zugauskunftssystem



Beispiel: Zugauskunftssystem



Phasen des Datenbankentwurfs



Grundlagen des relationalen Modells

Seien A_1, A_2, \dots, A_n **Attribute** und D_1, D_2, \dots, D_n ihre **Domänen** (Wertebereiche)

- **Relation:** $R \subseteq D_1 \times \dots \times D_n$
Bsp.: Telefonbuch \subseteq string x string x integer
- **Tupel:** $t \in R$
Bsp.: $t = („Mickey Mouse“, „Main Street“, 8124711)$
- **Schema:** legt die Struktur der gespeicherten Daten fest
Bsp.: Telefonbuch: $\{[Name: string, Straße: string, \underline{Telefon\#}: integer]\}$
Notation: sch(R)

Seien A_1, A_2, \dots, A_n **Attribute** und
 D_1, D_2, \dots, D_n ihre **Domänen** (Wertebereiche)

- **Relation:** $R \subseteq D_1 \times \dots \times D_n$
Bsp.: Telefonbuch \subseteq string x string x integer
- **Tupel:** $t \in R$
Bsp.: $t = („Mickey Mouse“, „Main Street“, 8124711)$
- **Schema:** legt die Struktur der gespeicherten Daten fest
Bsp.: Telefonbuch: {[Name: string, Straße: string, Telefon#:integer]}
Notation: sch(R)

Seien A_1, A_2, \dots, A_n **Attribute** und
 D_1, D_2, \dots, D_n ihre **Domänen** (Wertebereiche)

- **Relation:** $R \subseteq D_1 \times \dots \times D_n$
Bsp.: Telefonbuch \subseteq string x string x integer
- **Tupel:** $t \in R$
Bsp.: $t = („Mickey Mouse“, „Main Street“, 8124711)$
- **Schema:** legt die Struktur der gespeicherten Daten fest
Bsp.: Telefonbuch: {[Name: string, Straße: string, Telefon#:integer]}
Notation: sch(R)

Seien A_1, A_2, \dots, A_n **Attribute** und
 D_1, D_2, \dots, D_n ihre **Domänen** (Wertebereiche)

- **Relation:** $R \subseteq D_1 \times \dots \times D_n$
Bsp.: Telefonbuch \subseteq string x string x integer
- **Tupel:** $t \in R$
Bsp.: $t = („Mickey Mouse“, „Main Street“, 8124711)$
- **Schema:** legt die Struktur der gespeicherten Daten fest
Bsp.: Telefonbuch: {[Name: string, Straße: string, Telefon#:integer]}
Notation: sch(R)

Seien A_1, A_2, \dots, A_n **Attribute** und
 D_1, D_2, \dots, D_n ihre **Domänen** (Wertebereiche)

- **Relation:** $R \subseteq D_1 \times \dots \times D_n$
Bsp.: Telefonbuch \subseteq string x string x integer
- **Tupel:** $t \in R$
Bsp.: $t = („Mickey Mouse“, „Main Street“, 8124711)$
- **Schema:** legt die Struktur der gespeicherten Daten fest
Bsp.: Telefonbuch: {[Name: string, Straße: string, Telefon#:integer]}
Notation: sch(R)

Seien A_1, A_2, \dots, A_n **Attribute** und
 D_1, D_2, \dots, D_n ihre **Domänen** (Wertebereiche)

- **Relation:** $R \subseteq D_1 \times \dots \times D_n$
Bsp.: Telefonbuch \subseteq string x string x integer
- **Tupel:** $t \in R$
Bsp.: $t = (\text{„Mickey Mouse“}, \text{„Main Street“}, 8124711)$
- **Schema:** legt die Struktur der gespeicherten Daten fest
Bsp.: Telefonbuch: $\{\{\text{Name: string, Straße: string, Telefon#:integer}\}$
Notation: sch(R)

Seien A_1, A_2, \dots, A_n **Attribute** und
 D_1, D_2, \dots, D_n ihre **Domänen** (Wertebereiche)

- **Relation:** $R \subseteq D_1 \times \dots \times D_n$
Bsp.: Telefonbuch \subseteq string x string x integer
- **Tupel:** $t \in R$
Bsp.: $t = (\text{„Mickey Mouse“}, \text{„Main Street“}, 8124711)$
- **Schema:** legt die Struktur der gespeicherten Daten fest
Bsp.: Telefonbuch: $\{\{\text{Name: string, Straße: string, Telefon#:integer}\}$
Notation: sch(R)

Seien A_1, A_2, \dots, A_n **Attribute** und
 D_1, D_2, \dots, D_n ihre **Domänen** (Wertebereiche)

- **Relation:** $R \subseteq D_1 \times \dots \times D_n$
Bsp.: Telefonbuch \subseteq string x string x integer
- **Tupel:** $t \in R$
Bsp.: $t = (\text{„Mickey Mouse“}, \text{„Main Street“}, 8124711)$
- **Schema:** legt die Struktur der gespeicherten Daten fest
Bsp.: Telefonbuch: $\{\{\text{Name: string, Straße: string, Telefon#:integer}\}$
Notation: sch(R)

Seien A_1, A_2, \dots, A_n **Attribute** und
 D_1, D_2, \dots, D_n ihre **Domänen** (Wertebereiche)

- **Relation:** $R \subseteq D_1 \times \dots \times D_n$
Bsp.: Telefonbuch \subseteq string x string x integer
- **Tupel:** $t \in R$
Bsp.: $t = (\text{„Mickey Mouse“}, \text{„Main Street“}, 8124711)$
- **Schema:** legt die Struktur der gespeicherten Daten fest
Bsp.: Telefonbuch: $\{\{\text{Name: string, Straße: string, Telefon#:integer}\}$
Notation: sch(R)

Seien A_1, A_2, \dots, A_n **Attribute** und
 D_1, D_2, \dots, D_n ihre **Domänen** (Wertebereiche)

- **Relation:** $R \subseteq D_1 \times \dots \times D_n$
 Bsp.: Telefonbuch \subseteq string x string x integer
- **Tupel:** $t \in R$
 Bsp.: $t = („Mickey Mouse“, „Main Street“, 8124711)$
- **Schema:** legt die Struktur der gespeicherten Daten fest
 Bsp.: Telefonbuch: $\{[Name: string, Straße: string, \underline{Telefon#:integer}] \}$
 Notation: $sch(R)$

Seien A_1, A_2, \dots, A_n **Attribute** und
 D_1, D_2, \dots, D_n ihre **Domänen** (Wertebereiche)

- **Relation:** $R \subseteq D_1 \times \dots \times D_n$
 Bsp.: Telefonbuch \subseteq string x string x integer
- **Tupel:** $t \in R$
 Bsp.: $t = („Mickey Mouse“, „Main Street“, 8124711)$
- **Schema:** legt die Struktur der gespeicherten Daten fest
 Bsp.: Telefonbuch: $\{[Name: string, Straße: string, \underline{Telefon#:integer}] \}$
 Notation: $sch(R)$

Seien A_1, A_2, \dots, A_n **Attribute** und
 D_1, D_2, \dots, D_n ihre **Domänen** (Wertebereiche)

- **Relation:** $R \subseteq D_1 \times \dots \times D_n$
 Bsp.: Telefonbuch \subseteq string x string x integer
- **Tupel:** $t \in R$
 Bsp.: $t = („Mickey Mouse“, „Main Street“, 8124711)$
- **Schema:** legt die Struktur der gespeicherten Daten fest
 Bsp.: Telefonbuch: $\{[Name: string, Straße: string, \underline{Telefon#:integer}] \}$
 Notation: $sch(R)$

Telefonbuch		
Name:String	Straße:String	Telefon#:integer
Mickey Mouse	Main Street	4711
Donald Duck	Broadway	95672
...

- **Ausprägung:** der aktuelle Zustand der Datenbasis
- **Schlüssel:** minimale Menge von Attributen, deren Werte ein Tupel eindeutig identifizieren
- **Primärschlüssel:** wird unterstrichen
 - Einer der Schlüsselkandidaten wird als Primärschlüssel ausgewählt
 - Hat eine besondere Bedeutung bei der Referenzierung von Tupeln

Telefonbuch		
Name:String	Straße:String	Telefon#:integer
Mickey Mouse	Main Street	4711
Donald Duck	Broadway	95672
...

- **Ausprägung:** der aktuelle Zustand der Datenbasis
- **Schlüssel:** minimale Menge von Attributen, deren Werte ein Tupel eindeutig identifizieren
- **Primärschlüssel:** wird unterstrichen
 - Einer der Schlüsselkandidaten wird als Primärschlüssel ausgewählt
 - Hat eine besondere Bedeutung bei der Referenzierung von Tupeln

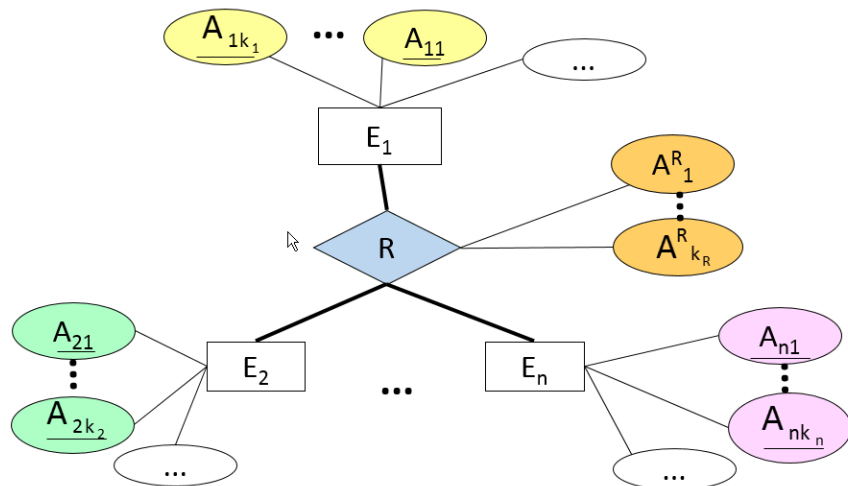


Telefonbuch		
Name:String	Straße:String	Telefon#:integer
Mickey Mouse	Main Street	4711
Donald Duck	Broadway	95672
...

- **Ausprägung:** der aktuelle Zustand der Datenbasis
- **Schlüssel:** minimale Menge von Attributen, deren Werte ein Tupel eindeutig identifizieren
- **Primärschlüssel:** wird unterstrichen
 - Einer der Schlüsselkandidaten wird als Primärschlüssel ausgewählt
 - Hat eine besondere Bedeutung bei der Referenzierung von Tupeln



Relationale Darstellung von Beziehungen

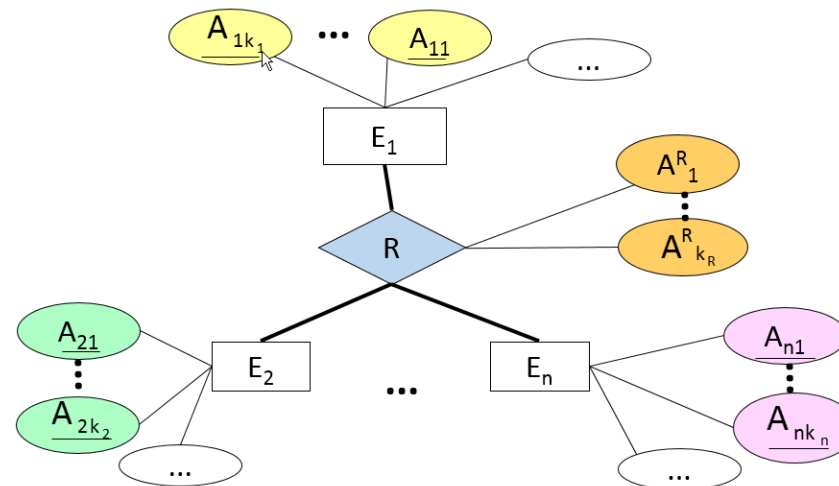


$$R: \{ \{ \underline{A_{11}}, \dots, A_{1k_1} \}, \{ A_{21}, \dots, A_{2k_2} \}, \dots, \{ A_{n1}, \dots, A_{nk_n} \}, \{ A_{11}^R, \dots, A_{k_R}^R \} \}$$

Schlüssel von E₁
Schlüssel von E₂
Schlüssel von E_n
Attribute von R



Relationale Darstellung von Beziehungen

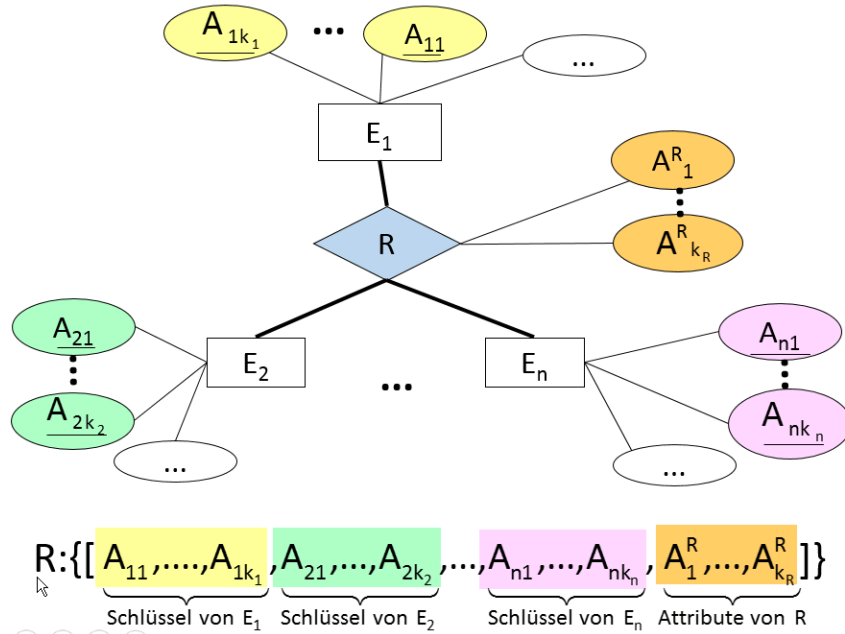


$$R: \{ \{ \underline{A_{11}}, \dots, A_{1k_1} \}, \{ A_{21}, \dots, A_{2k_2} \}, \dots, \{ A_{n1}, \dots, A_{nk_n} \}, \{ A_{11}^R, \dots, A_{k_R}^R \} \}$$

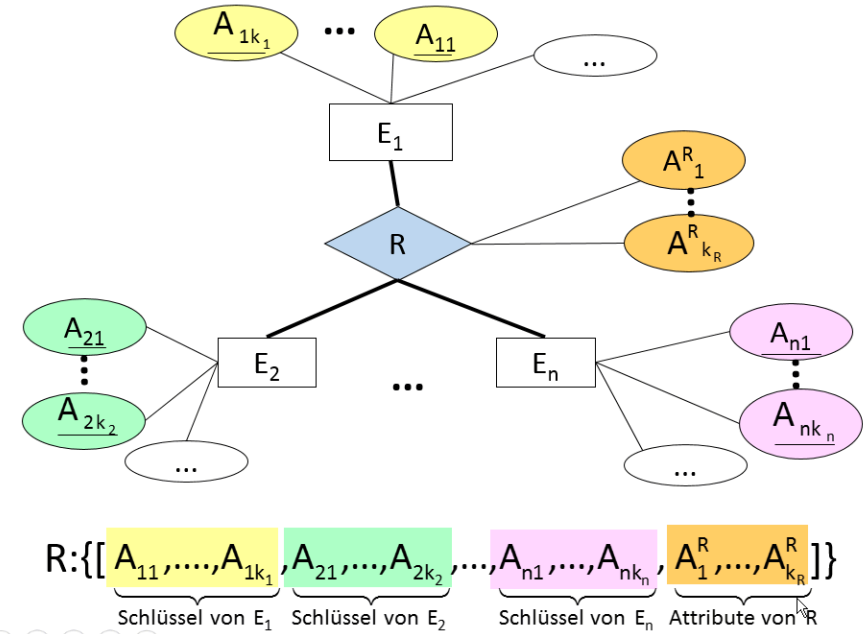
Schlüssel von E₁
Schlüssel von E₂
Schlüssel von E_n
Attribute von R



Relationale Darstellung von Beziehungen



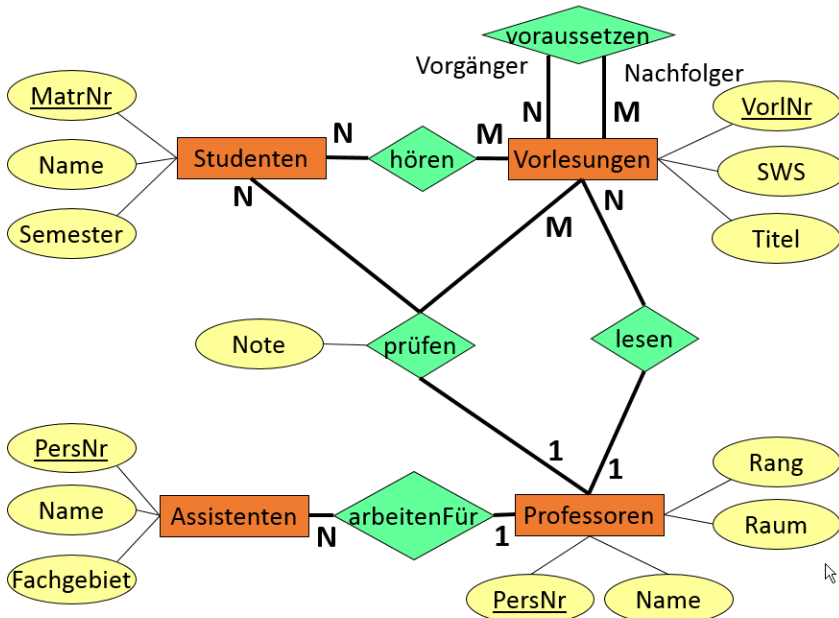
Relationale Darstellung von Beziehungen



71

71

Uml-Schema



Relationale Darstellung von Entities

Studenten: $\{ \{ \text{MatrNr:integer}, \text{Name: string}, \text{Semester: integer} \} \}$

Vorlesungen: $\{ \{ \text{VorlNr:integer}, \text{Titel: string}, \text{SWS: integer} \} \}$

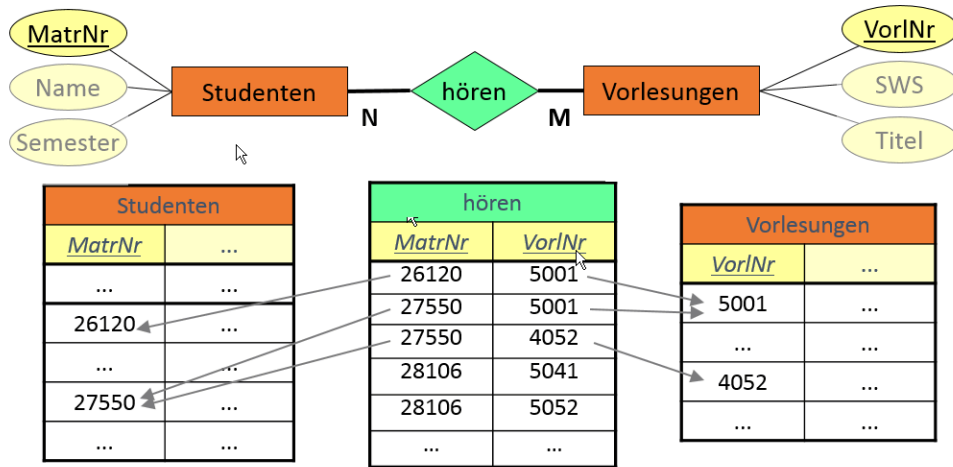
Professoren: $\{ \{ \text{PersNr:integer}, \text{Name: string}, \text{Rang: string}, \text{Raum: integer} \} \}$

Assistenten: $\{ \{ \text{PersNr:integer}, \text{Name: string}, \text{Fachgebiet: string} \} \}$

69

70

beziehungen unseres Beispiel-Schemas



Primärschlüssel von hören

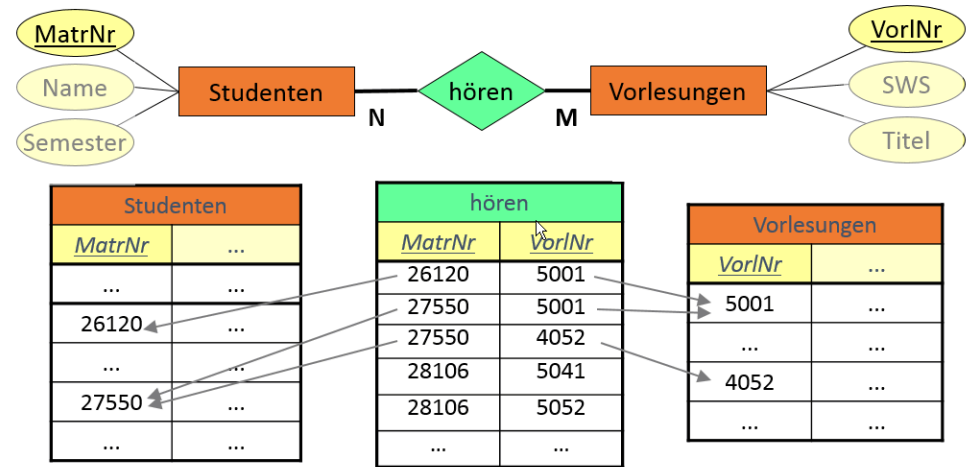
hören : {[MatrNr: integer, VorlNr: integer]}

Fremdschlüssel



72

beziehungen unseres Beispiel-Schemas



Primärschlüssel von hören

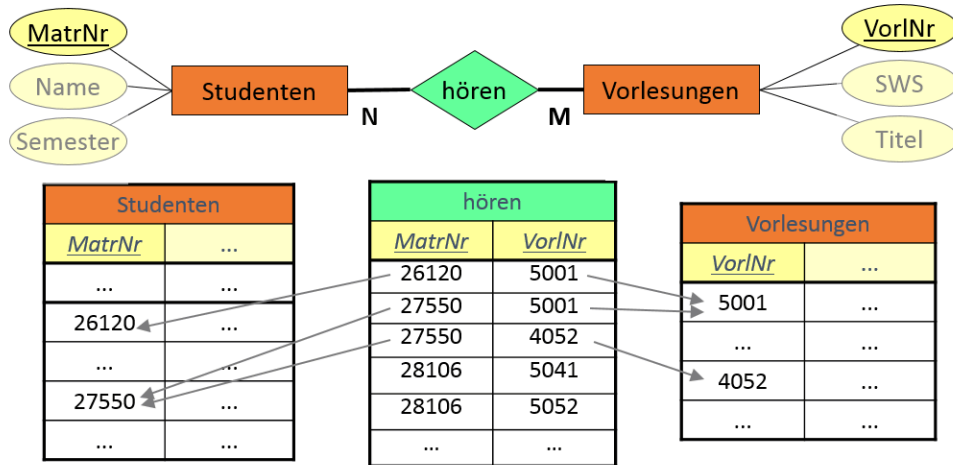
hören : {[MatrNr: integer, VorlNr: integer]}

Fremdschlüssel



72

beziehungen unseres Beispiel-Schemas



Primärschlüssel von hören

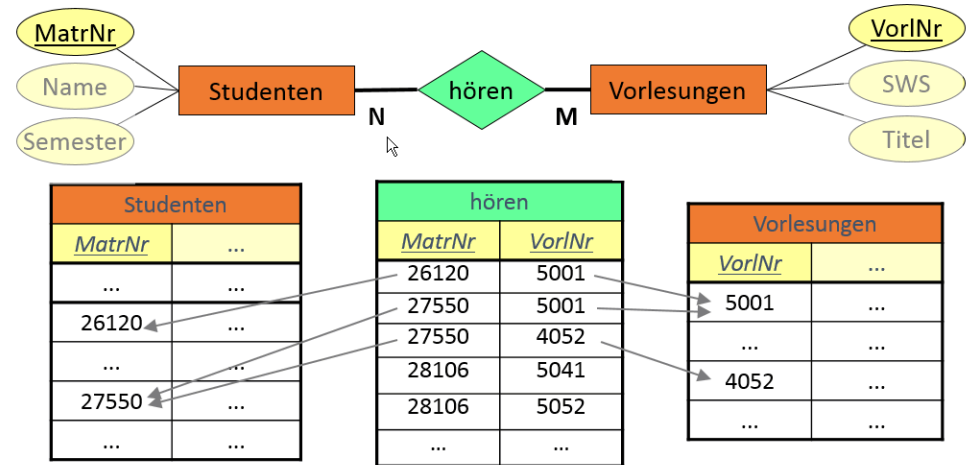
hören : {[MatrNr: integer, VorlNr: integer]}

Fremdschlüssel



72

beziehungen unseres Beispiel-Schemas



Primärschlüssel von hören

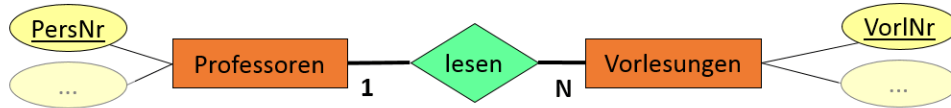
hören : {[MatrNr: integer, VorlNr: integer]}

Fremdschlüssel



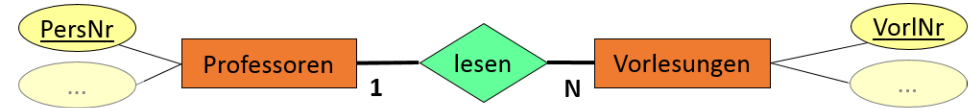
72

beziehungen unseres Beispiel-Schemas



Professoren		lesen		Vorlesungen	
PersNr	...	PersNr	VorlNr	VorlNr	...
2125	...	2137	5001	5001	...
...	...	2125	5041	5041	...
...
...	...	2125	5049	5049	...
...
2137

beziehungen unseres Beispiel-Schemas



Professoren		lesen		Vorlesungen	
PersNr	...	PersNr	VorlNr	VorlNr	...
2125	...	2137	5001	5001	...
...	...	2125	5041	5041	...
...
...	...	2125	5049	5049	...
...
2137

Primärschlüssel von lesen

lesen: {[PersNr: integer, **VorlNr: integer**]}

Fremdschlüssel

Nur das ist der Primärschlüssel von lesen!!!
Der zugehörige Prof ist durch 1:N bereits eindeutig bestimmt

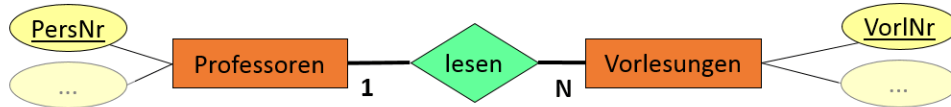
Primärschlüssel von lesen

lesen: {[PersNr: integer, **VorlNr: integer**]}

Fremdschlüssel

Nur das ist der Primärschlüssel von lesen!!!
Der zugehörige Prof ist durch 1:N bereits eindeutig bestimmt

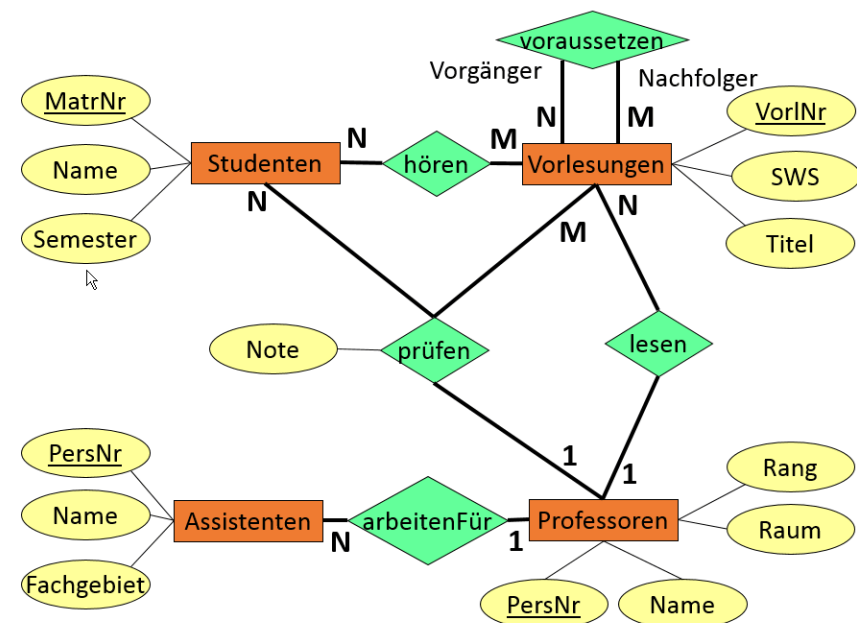
verfeinerung des relationalen Schemas



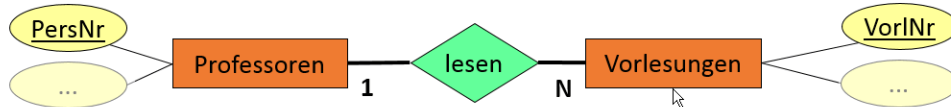
⇒ 1:N-Beziehungen (auch 1:1-Bez.) sind verfeinerbar:

- Initial-Entwurf:
 - Vorlesungen : {[VorlNr, Titel, SWS]}
 - Professoren : {[PersNr, Name, Rang, Raum]}
 - lesen: {[PersNr, VorlNr]} ≡ lesen: {[VorlNr, PersNr]}
- Verfeinerung durch Zusammenfassung
 - Vorlesungen : {[VorlNr, Titel, SWS, **gelesenVon**]}
 - Professoren : {[PersNr, Name, Rang, Raum]}

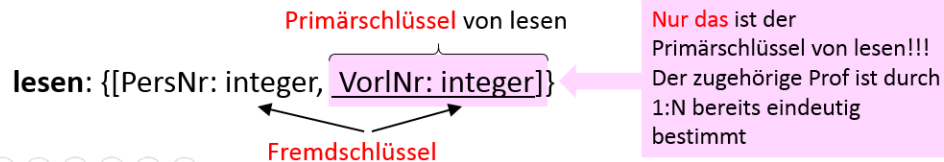
Uni-Schema



beziehungen unseres Beispiel-Schemas

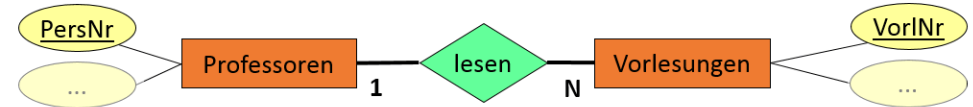


Professoren		lesen		Vorlesungen	
PersNr	...	PersNr	VorlNr	VorlNr	...
2125	...	2137	5001	5001	...
...	...	2125	5041	5041	...
...
...	...	2125	5049	5049	...
...
2137



73

verfeinerung des relationalen Schemas



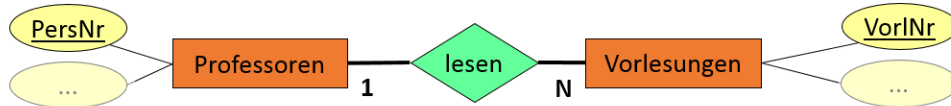
⇒ 1:N-Beziehungen (auch 1:1-Bez.) sind verfeinerbar:

- Initial-Entwurf:
 - Vorlesungen** : {[VorlNr, Titel, SWS]}
 - Professoren** : {[PersNr, Name, Rang, Raum]}
 - lesen** : {[PersNr, VorlNr]} ≡ **lesen** : {[VorlNr, PersNr]}
- Verfeinerung** durch Zusammenfassung
 - Vorlesungen** : {[VorlNr, Titel, SWS, *gelesenVon*]}
 - Professoren** : {[PersNr, Name, Rang, Raum]}



74

verfeinerung des relationalen Schemas



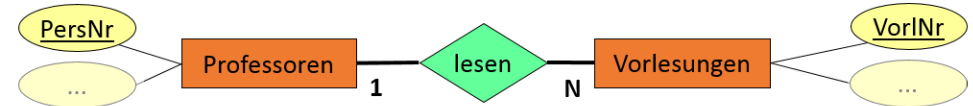
⇒ 1:N-Beziehungen (auch 1:1-Bez.) sind verfeinerbar:

- Initial-Entwurf:
 - Vorlesungen** : {[VorlNr, Titel, SWS]}
 - Professoren** : {[PersNr, Name, Rang, Raum]}
 - lesen** : {[PersNr, VorlNr]} ≡ **lesen** : {[VorlNr, PersNr]}
- Verfeinerung** durch Zusammenfassung
 - Vorlesungen** : {[VorlNr, Titel, SWS, *gelesenVon*]}
 - Professoren** : {[PersNr, Name, Rang, Raum]}



74

verfeinerung des relationalen Schemas



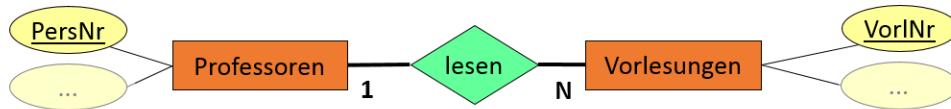
⇒ 1:N-Beziehungen (auch 1:1-Bez.) sind verfeinerbar:

- Initial-Entwurf:
 - Vorlesungen** : {[VorlNr, Titel, SWS]}
 - Professoren** : {[PersNr, Name, Rang, Raum]}
 - lesen** : {[PersNr, VorlNr]} ≡ **lesen** : {[VorlNr, PersNr]}
- Verfeinerung** durch Zusammenfassung
 - Vorlesungen** : {[VorlNr, Titel, SWS, *gelesenVon*]}
 - Professoren** : {[PersNr, Name, Rang, Raum]}

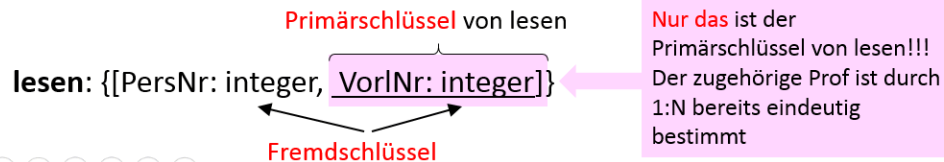


74

beziehungen unseres Beispiel-Schemas

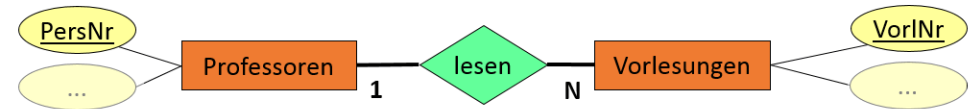


Professoren		lesen		Vorlesungen	
PersNr	...	PersNr	VorlNr	VorlNr	...
2125	...	2137	5001	5001	...
...	...	2125	5041	5041	...
...
...	...	2125	5049	5049	...
2137



73

verfeinerung des relationalen Schemas

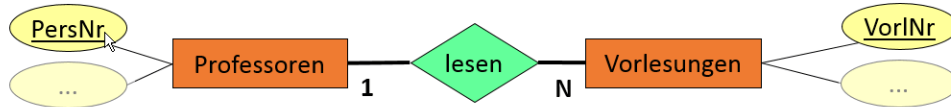


⇒ 1:N-Beziehungen (auch 1:1-Bez.) sind verfeinerbar:

- Initial-Entwurf:
 - Vorlesungen : {[VorlNr, Titel, SWS]}
 - Professoren : {[PersNr, Name, Rang, Raum]}
 - lesen: {[PersNr, VorlNr]} ≡ lesen: {[VorlNr, PersNr]}
- Verfeinerung durch Zusammenfassung
 - Vorlesungen : {[VorlNr, Titel, SWS, **gelesenVon**]}
 - Professoren : {[PersNr, Name, Rang, Raum]}

74

verfeinerung des relationalen Schemas



⇒ 1:N-Beziehungen (auch 1:1-Bez.) sind verfeinerbar:

- Initial-Entwurf:
 - Vorlesungen : {[VorlNr, Titel, SWS]}
 - Professoren : {[PersNr, Name, Rang, Raum]}
 - lesen: {[PersNr, VorlNr]} ≡ lesen: {[VorlNr, PersNr]}
- Verfeinerung durch Zusammenfassung
 - Vorlesungen : {[VorlNr, Titel, SWS, **gelesenVon**]}
 - Professoren : {[PersNr, Name, Rang, Raum]}

verfeinerung des relationalen Schemas

Vorsicht: So geht es NICHT !! --> Anomalien

Professoren				
PersNr	Name	Rang	Raum	liest
2125	Sokrates	C4	226	5041
2125	Sokrates	C4	226	5049
2125	Sokrates	C4	226	4052
...
2134	Augustinus	C3	309	5022
2136	Curie	C4	36	???
...

Vorlesungen		
VorlNr	Titel	SWS
5001	Grundzüge	4
5041	Ethik	4
5043	Erkenntnistheorie	3
5049	Mäeutik	2
4052	Logik	4
5052	Wissenschaftstheorie	3
5022	Glaube und Wissen	2
...

- Update-Anomalie: Was passiert wenn Sokrates umzieht?
- Lösch-Anomalie: Was passiert wenn „Glaube und Wissen“ wegfällt?
- Einfüge-Anomalie: Curie forscht nur und liest keine Vorlesung.

74

77

verfeinerung des relationalen Schemas

Ausprägung von Professoren und Vorlesungen nach Verfeinerung

PersNr	Name	Rang	Raum	liest	VorlNr	Titel	SWS	Colspan
2125	Sokrates	C4	226	5041	5001	Grundzüge	4	2127
2125	Sokrates	C4	226	5049	5041	Ethik	4	2129
2125	Sokrates	C4	226	4052	5043	Erkenntnistheorie	3	2128
...	5049	Mäeutik	2	2125
2134	Augustinus	C3	309	5022	4052	Logik	4	2125
2136	Curie	C4	36	???	5052	Wissenschaftstheorie	3	2129
2137	Kant	C4	7	...	5022	Glaube und Wissen	2	...

Vorsicht: So geht es NICHT !! --> Anomalien

Professoren				
PersNr	Name	Rang	Raum	liest
2125	Sokrates	C4	226	5041
2125	Sokrates	C4	226	5049
2125	Sokrates	C4	226	4052
...
2134	Augustinus	C3	309	5022
2136	Curie	C4	36	???
...

Vorlesungen		
VorlNr	Titel	SWS
5001	Grundzüge	4
5041	Ethik	4
5043	Erkenntnistheorie	3
5049	Mäeutik	2
4052	Logik	4
5052	Wissenschaftstheorie	3
5022	Glaube und Wissen	2
...

- Update-Anomalie: Was passiert wenn Sokrates umzieht?
- Lösch-Anomalie: Was passiert wenn „Glaube und Wissen“ wegfällt?
- Einfüge-Anomalie: Curie forscht nur und liest keine Vorlesung.

verfeinerung des relationalen Schemas

Ausprägung von Professoren und Vorlesungen nach Verfeinerung

PersNr	Name	Rang	Raum	liest	VorlNr	Titel	SWS	Colspan
2125	Sokrates	C4	226	5001	5001	Grundzüge	4	2127
2125	Sokrates	C4	226	5041	5041	Ethik	4	2129
2125	Sokrates	C4	226	4052	5043	Erkenntnistheorie	3	2128
...	5049	Mäeutik	2	2125
2134	Augustinus	C3	309	5022	4052	Logik	4	2125
2136	Curie	C4	36	???	5052	Wissenschaftstheorie	3	2129
2137	Kant	C4	7	...	5022	Glaube und Wissen	2	...

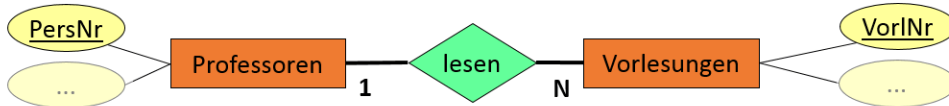
Vorsicht: So geht es NICHT !! --> Anomalien

Professoren				
PersNr	Name	Rang	Raum	liest
2125	Sokrates	C4	226	5041
2125	Sokrates	C4	226	5049
2125	Sokrates	C4	226	4052
...
2134	Augustinus	C3	309	5022
2136	Curie	C4	36	???
...

Vorlesungen		
VorlNr	Titel	SWS
5001	Grundzüge	4
5041	Ethik	4
5043	Erkenntnistheorie	3
5049	Mäeutik	2
4052	Logik	4
5052	Wissenschaftstheorie	3
5022	Glaube und Wissen	2
...

- Update-Anomalie: Was passiert wenn Sokrates umzieht?
- Lösch-Anomalie: Was passiert wenn „Glaube und Wissen“ wegfällt?
- Einfüge-Anomalie: Curie forscht nur und liest keine Vorlesung.

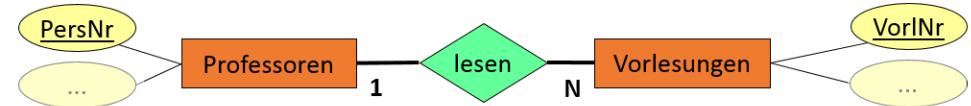
beziehungen unseres Beispiel-Schemas



Professoren		lesen		Vorlesungen	
PersNr	...	PersNr	VorlNr	VorlNr	...
2125	...	2137	5001	5001	...
...	...	2125	5041	5041	...
...
...	...	2125	5049	5049	...
...
2137

Primärschlüssel von lesen
lesen: {[PersNr: integer, VorlNr: integer]}
 Fremdschlüssel
 Nur das ist der Primärschlüssel von lesen!!!
 Der zugehörige Prof ist durch 1:N bereits eindeutig bestimmt

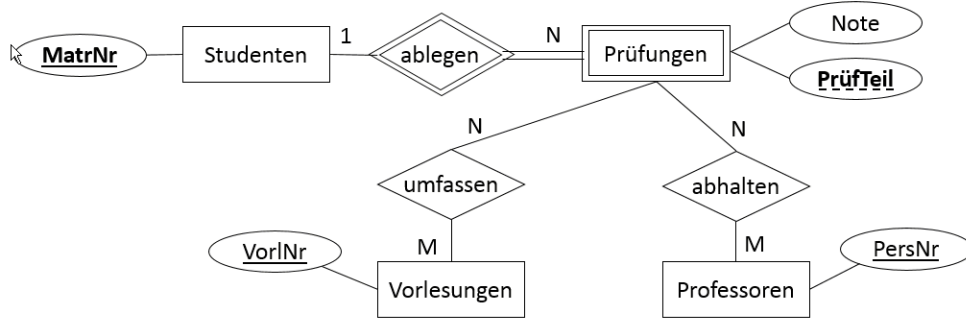
beziehungen unseres Beispiel-Schemas



Professoren		lesen		Vorlesungen	
PersNr	...	PersNr	VorlNr	VorlNr	...
2125	...	2137	5001	5001	...
...	...	2125	5041	5041	...
...
...	...	2125	5049	5049	...
...
2137

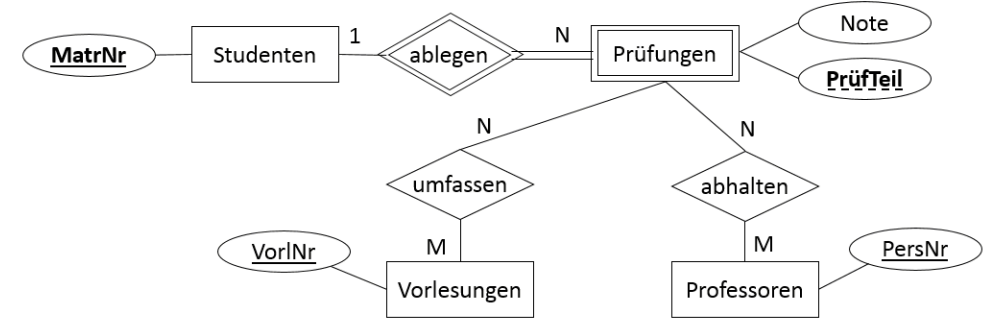
Primärschlüssel von lesen
lesen: {[PersNr: integer, VorlNr: integer]}
 Fremdschlüssel
 Nur das ist der Primärschlüssel von lesen!!!
 Der zugehörige Prof ist durch 1:N bereits eindeutig bestimmt

Relationale Modellierung schwacher Entitytypen



Prüfungen: {[**MatrNr: integer**, **PrüfTeil: string**, Note: integer]}

Relationale Modellierung schwacher Entitytypen



Prüfungen: {[**MatrNr: integer**, **PrüfTeil: string**, Note: integer]}

umfassen: {[**MatrNr: integer**, **PrüfTeil: string**, VorlNr: integer]}

abhalten: {[**MatrNr: integer**, **PrüfTeil: string**, PersNr: integer]}



78

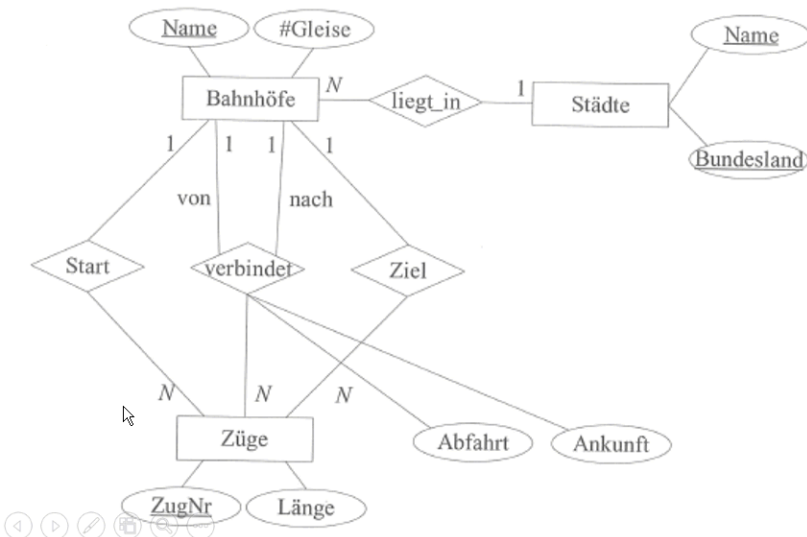


79

Aufgabe 3.1

Gegeben sei die ER-Modellierung von Zugverbindungen in Abbildung 3.1.

a) Fügen Sie bei den Beziehungen Kardinalitäten in der (min, max) -Notation hinzu.

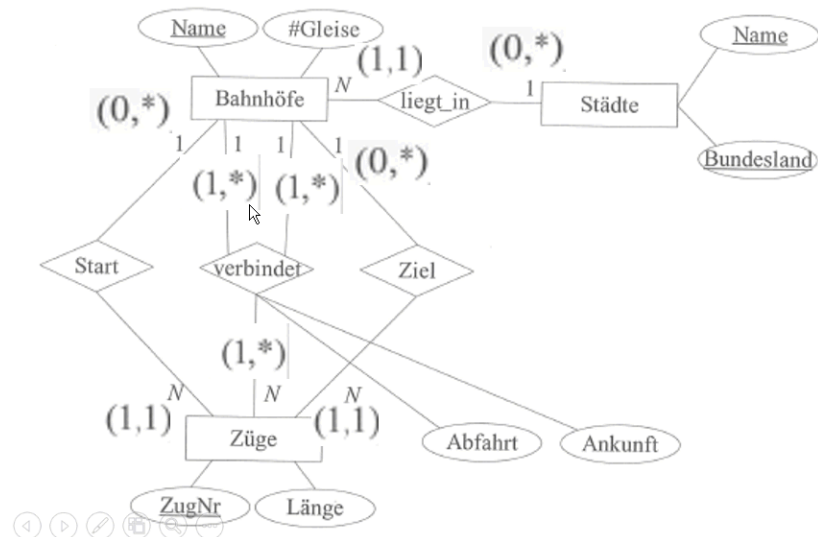


83

Aufgabe 3.1

Gegeben sei die ER-Modellierung von Zugverbindungen in Abbildung 3.1.

a) Fügen Sie bei den Beziehungen Kardinalitäten in der (min, max) -Notation hinzu.



87



b) Übertragen Sie das ER-Modell in ein relationales Schema.

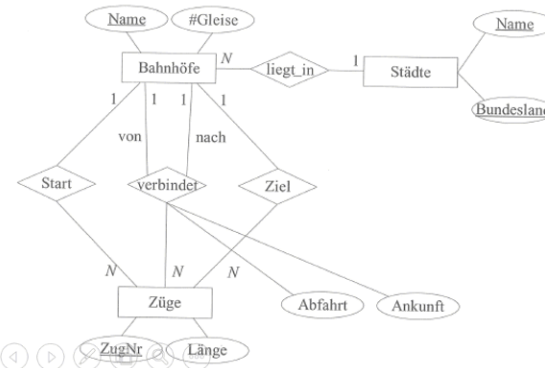
Die initiale Überführung ergibt folgende Relationen für die Entitytypen:

Städte : {[Name : string, Bundesland : string]}



c) Verfeinern Sie das relationale Schema soweit möglich durch Eliminierung von Relationen.

- Städte : {[Name : string, Bundesland : string]} (1)
- Bahnhöfe : {[Name : string, #Gleise : integer]} (2)
- Züge : {[ZugNr : integer, Länge : integer]} (3)
- liegt_in : {[BName : string, SName : string, Bundesland : string]} (4)
- Start : {[ZugNr : integer, BName : string]} (5)
- Ziel : {[ZugNr : integer, BName : string]} (6)
- verbindet : {[VonBahnhof : string, NachBahnhof : string, ZugNr : integer, Abfahrt : date, Ankunft : date]} (7)



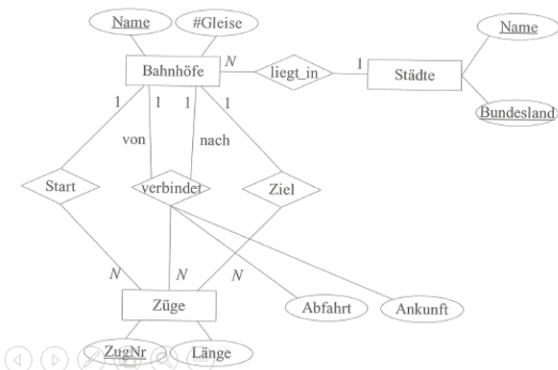
c) Verfeinern Sie das relationale Schema soweit möglich durch Eliminierung von Relationen.

- Städte : {[Name : string, Bundesland : string]} (1)
- Bahnhöfe : {[Name : string, #Gleise : integer]} (2)
- Züge : {[ZugNr : integer, Länge : integer]} (3)
- liegt_in : {[BName : string, SName : string, Bundesland : string]} (4)
- Start : {[ZugNr : integer, BName : string]} (5)
- Ziel : {[ZugNr : integer, BName : string]} (6)
- verbindet : {[VonBahnhof : string, NachBahnhof : string, ZugNr : integer, Abfahrt : date, Ankunft : date]} (7)

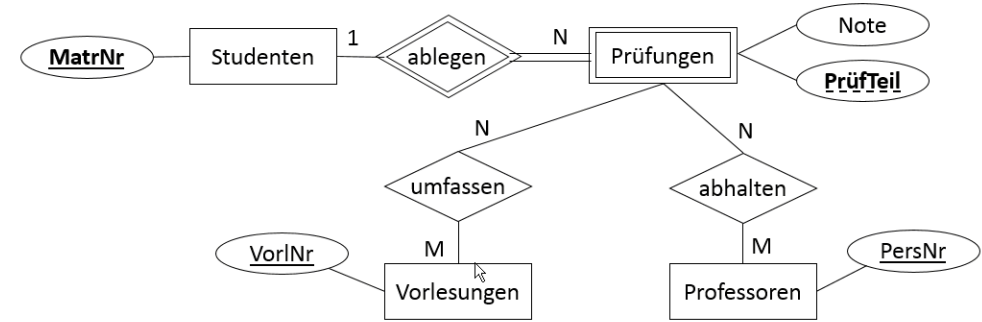


c) Verfeinern Sie das relationale Schema soweit möglich durch Eliminierung von Relationen.

- Städte : {[Name : string, Bundesland : string]} (1)
- Bahnhöfe : {[Name : string, #Gleise : integer]} (2)
- Züge : {[ZugNr : integer, Länge : integer]} (3)
- liegt_in : {[BName : string, SName : string, Bundesland : string]} (4)
- Start : {[ZugNr : integer, BName : string]} (5)
- Ziel : {[ZugNr : integer, BName : string]} (6)
- verbindet : {[VonBahnhof : string, NachBahnhof : string, ZugNr : integer, Abfahrt : date, Ankunft : date]} (7)



Relationale Modellierung schwacher Entitytypen



Prüfungen: {[**MatrNr: integer**, **PrüfTeil: string**, Note: integer]}
 umfassen: {[**MatrNr: integer**, **PrüfTeil: string**, **VorlNr: integer**]}
 abhalten: {[**MatrNr: integer**, **PrüfTeil: string**, **PersNr: integer**]}

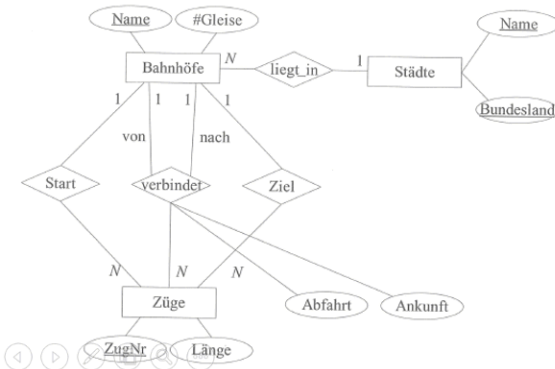
- (1)
- (2)
- (3)
- (4)
- (5)
- (6)
- (7)

94

79

c) Verfeinern Sie das relationale Schema soweit möglich durch Eliminierung von Relationen.

Städte : {[Name : string, Bundesland : string]}
 Bahnhöfe : {[Name : string, #Gleise : integer]}
 Züge : {[ZugNr : integer, Länge : integer]}
 liegt_in : {[BName : string, SName : string, Bundesland : string]}
 Start : {[ZugNr : integer, BName : string]}
 Ziel : {[ZugNr : integer, BName : string]}
 verbindet : {[VonBahnhof : string, NachBahnhof : string, ZugNr : integer, Abfahrt : date, Ankunft : date]}



94

Die relationale Algebra

minimale Algebra

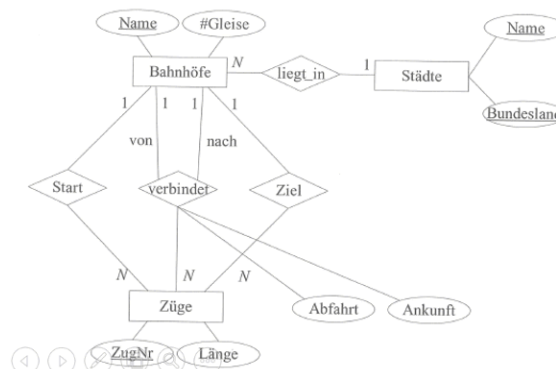
- σ Selektion
- π Projektion
- \times Kreuzprodukt
- ρ Umbenennung
- \cup Vereinigung
- $-$ Mengendifferenz

syntaktischer Zucker

- \div Division
- \cap Mengendurchschnitt
- \bowtie Join (verbund)
- \bowtie äußerer Join
- \ltimes Semi-Join (rechter)
- \ltimes Semi-Join (linker)
- \ltimes linker äußerer Join
- \ltimes rechter äußerer Join
- ...

c) Verfeinern Sie das relationale Schema soweit möglich durch Eliminierung von Relationen.

Städte : {[Name : string, Bundesland : string]} (1)
 Bahnhöfe : {[Name : string, #Gleise : integer]} (2)
 Züge : {[ZugNr : integer, Länge : integer]} (3)
 liegt_in : {[BName : string, SName : string, Bundesland : string]} (4)
 Start : {[ZugNr : integer, BName : string]} (5)
 Ziel : {[ZugNr : integer, BName : string]} (6)
 verbindet : {[VonBahnhof : string, NachBahnhof : string, ZugNr : integer, Abfahrt : date, Ankunft : date]} (7)



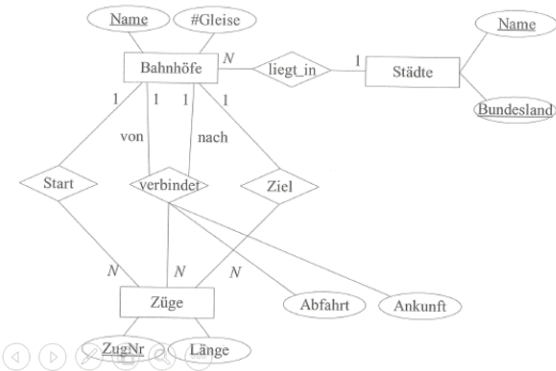
97

94



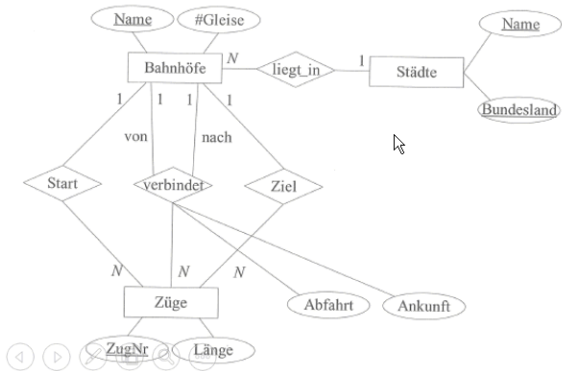
c) Verfeinern Sie das relationale Schema soweit möglich durch Eliminierung von Relationen.

- (1) Städte : {[Name : string, Bundesland : string]}
- (2) Bahnhöfe : {[Name : string, #Gleise : integer]}
- (3) Züge : {[ZugNr : integer, Länge : integer]}
- (4) liegt_in : {[BName : string, SName : string, Bundesland : string]}
- (5) Start : {[ZugNr : integer, BName : string]}
- (6) Ziel : {[ZugNr : integer, BName : string]}
- (7) verbindet : {[VonBahnhof : string, NachBahnhof : string, ZugNr : integer, Abfahrt : date, Ankunft : date]}



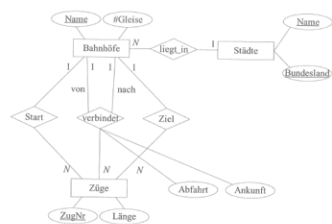
c) Verfeinern Sie das relationale Schema soweit möglich durch Eliminierung von Relationen.

- (1) Städte : {[Name : string, Bundesland : string]}
- (2) Bahnhöfe : {[Name : string, #Gleise : integer]}
- (3) Züge : {[ZugNr : integer, Länge : integer]}
- (4) liegt_in : {[BName : string, SName : string, Bundesland : string]}
- (5) Start : {[ZugNr : integer, BName : string]}
- (6) Ziel : {[ZugNr : integer, BName : string]}
- (7) verbindet : {[VonBahnhof : string, NachBahnhof : string, ZugNr : integer, Abfahrt : date, Ankunft : date]}



c) Verfeinern Sie das relationale Schema soweit möglich durch Eliminierung von Relationen.

- (1) Städte : {[Name : string, Bundesland : string]}
- (2) Bahnhöfe : {[Name : string, #Gleise : integer]}
- (3) Züge : {[ZugNr : integer, Länge : integer]}
- (4) liegt_in : {[BName : string, SName : string, Bundesland : string]}
- (5) Start : {[ZugNr : integer, BName : string]}
- (6) Ziel : {[ZugNr : integer, BName : string]}
- (7) verbindet : {[VonBahnhof : string, NachBahnhof : string, ZugNr : integer, Abfahrt : date, Ankunft : date]}



(4) \mapsto (2), (5) \mapsto (3), (6) \mapsto (3)

Damit ergibt sich folgendes Schema:

- Städte : {[Name : string, Bundesland : string]}
- Bahnhöfe : {[Name : string, #Gleise : integer, SName : string, Bundesland : string]}
- Züge : {[ZugNr : integer, Länge : integer, StartBahnhof : string, ZielBahnhof : string]}
- verbindet : {[VonBahnhof : string, NachBahnhof : string, ZugNr : integer, Abfahrt : date, Ankunft : date]}



Selektion

$\sigma_P : E \rightarrow E'$ mit P: Prädikat, E: Relation (Tabelle), E': Relation (Tabelle):

$\sigma_P(E)$ wählt **alle Tupel** (Zeilen) aus E aus, **die P erfüllen**.
Ergebnis: Relation (Tabelle) E'

Beispiel: $\sigma_{Semester > 10}$ (Studenten)

Studenten		
MatrNr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12
26120	Fichte	10
26830	Aristoxenos	8
27550	Schopenhauer	6
28106	Carnap	3
29120	Theophrastos	2
29555	Feuerbach	2

$\sigma_{Semester > 10}$

$\sigma_{Semester > 10}$ (Studenten)		
MatrNr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12

$\sigma_P : E \rightarrow E'$ mit P : Prädikat, E : Relation (Tabelle),
 E' : Relation (Tabelle):

$\sigma_P (E)$ wählt **alle Tupel** (Zeilen) aus E aus, **die P erfüllen**.
 Ergebnis: Relation (Tabelle) E'

Beispiel: $\sigma_{Semester > 10}$ (Studenten)

Studenten		
MatrNr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12
26120	Fichte	10
26830	Aristoxenos	8
27550	Schopenhauer	6
28106	Carnap	3
29120	Theophrastos	2
29555	Feuerbach	2

$\xrightarrow{\sigma_{Semester > 10}}$

$\sigma_{Semester > 10}$ (Studenten)		
MatrNr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12

$\sigma_P : E \rightarrow E'$ mit P : Prädikat, E : Relation (Tabelle),
 E' : Relation (Tabelle):

$\sigma_P (E)$ wählt **alle Tupel** (Zeilen) aus E aus, **die P erfüllen**.
 Ergebnis: Relation (Tabelle) E'

Beispiel: $\sigma_{Semester > 10}$ (Studenten)

Studenten		
MatrNr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12
26120	Fichte	10
26830	Aristoxenos	8
27550	Schopenhauer	6
28106	Carnap	3
29120	Theophrastos	2
29555	Feuerbach	2

$\xrightarrow{\sigma_{Semester > 10}}$

$\sigma_{Semester > 10}$ (Studenten)		
MatrNr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12

$\sigma_P : E \rightarrow E'$ mit P : Prädikat, E : Relation (Tabelle),
 E' : Relation (Tabelle):

$\sigma_P (E)$ wählt **alle Tupel** (Zeilen) aus E aus, **die P erfüllen**.
 Ergebnis: Relation (Tabelle) E'

Beispiel: $\sigma_{Semester > 10}$ (Studenten)

Studenten		
MatrNr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12
26120	Fichte	10
26830	Aristoxenos	8
27550	Schopenhauer	6
28106	Carnap	3
29120	Theophrastos	2
29555	Feuerbach	2

$\xrightarrow{\sigma_{Semester > 10}}$

$\sigma_{Semester > 10}$ (Studenten)		
MatrNr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12

$\sigma_P : E \rightarrow E'$ mit P : Prädikat, E : Relation (Tabelle),
 E' : Relation (Tabelle):

$\sigma_P (E)$ wählt **alle Tupel** (Zeilen) aus E aus, **die P erfüllen**.
 Ergebnis: Relation (Tabelle) E'

Beispiel: $\sigma_{Semester > 10}$ (Studenten)

Studenten		
MatrNr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12
26120	Fichte	10
26830	Aristoxenos	8
27550	Schopenhauer	6
28106	Carnap	3
29120	Theophrastos	2
29555	Feuerbach	2

$\xrightarrow{\sigma_{Semester > 10}}$

$\sigma_{Semester > 10}$ (Studenten)		
MatrNr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12

$\sigma_P: E \rightarrow E'$ mit P : Prädikat, E : Relation (Tabelle),
 E' : Relation (Tabelle):

$\sigma_P(E)$ wählt **alle Tupel** (Zeilen) aus E aus, **die P erfüllen**.
 Ergebnis: Relation (Tabelle) E'

Beispiel: $\sigma_{Semester > 10}$ (Studenten)

Studenten		
MatrNr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12
26120	Fichte	10
26830	Aristoxenos	8
27550	Schopenhauer	6
28106	Carnap	3
29120	Theophrastos	2
29555	Feuerbach	2

$\xrightarrow{\sigma_{Semester > 10}}$

$\sigma_{Semester > 10}$ (Studenten)		
MatrNr	Name	Semester
24002	Xenokrates	18
25403	Jonas	12

Das **Selektionsprädikat** P in $\sigma_P(E)$ ist eine Formel, die zu **wahr** oder **falsch** ausgewertet werden kann und die aufgebaut ist aus:

- **Attributnamen** der Argumentrelation E
- **Konstanten**
- den arithmetischen **Vergleichsoperatoren** $=, <, \leq, >, \geq, \neq$
- den **logischen Operatoren** \wedge (und), \vee (oder) und \neg (nicht)

P wird auf **allen** $e \in E$ ausgewertet. Wenn $P(e) = \text{wahr}$ dann wird e in die Ergebnisrelation E' **aufgenommen**.

Beispiel: $E' = \text{Professoren} \times \text{hören}$

Professoren			
PersNr	Name	Rang	Raum
2125	Sokrates	C4	226
2126	Russel	C4	232
...
2137	Kant	C4	7

\times

hören	
MatrNr	VorlNr
26120	5001
27550	5001
...	...
25403	5022

$=$

$E' = \text{Professoren} \times \text{hören}$					
PersNr	Name	Rang	Raum	MatrNr	VorlNr
2125	Sokrates	C4	226	26120	5001
2125	Sokrates	C4	226	27550	5001
...
2125	Sokrates	C4	226	25403	5022
2126	Russel	C4	232	26120	5001
...
2126	Russel	C4	232	25403	5022
...
2137	Kant	C4	7	25403	5022

- Problem: riesige Ergebnisse (im Bsp. $|E'| = |\text{Prof.}| * |\text{hören}| = 7 * 12$)
- "bessere" Operation: Join (siehe weiter unten)

- $\text{sch}(E)$: die **Menge aller Attribute** von E

Bsp.: $\text{sch}(\text{Professoren}) = \{\text{PersNr}, \text{Name}, \text{Rang}, \text{Raum}\}$

- $E.A$: **Qualifizierter Name** des Attributs A in Relation (Tabelle) E

Bsp.: $\text{Professoren.PersNr}, \text{Assistenten.PersNr}$

Professoren				Assistenten			
PersNr	Name	Rang	Raum	PersNr	Name	Fachgebiet	Boss
...

\times

Professoren x Assistenten							
Professoren.PersNr	Professoren.Name	Rang	Raum	Assistenten.PersNr	Assistenten.Name	Fachgebiet	Boss
...

- $sch(E)$: die Menge aller Attribute von E

Bsp.: $sch(\text{Professoren}) = \{\text{PersNr, Name, Rang, Raum}\}$

- E.A : Qualifizierter Name des Attributs A in Relation (Tabelle) E

Bsp.: Professoren.PersNr, Assistenten.PersNr

Professoren				X	Assistenten				=
PersNr	Name	Rang	Raum		PersNr	Name	Fachgebiet	Boss	
...	

Professoren x Assistenten							
Professoren.PersNr	Professoren.Name	Rang	Raum	Assistenten.PersNr	Assistenten.Name	Fachgebiet	Boss
...