

Aufgabe 3 (1)

- **A1:** **select * from Kategorie**
where Bezeichnung not in
(select Bezeichnung from Kunde)
- **A2:** **select * from Kategorie kat**
where not exists
(select * from Kunde k
where k.Bezeichnung=kat.Bezeichnung)

Aufgabe 3 (2)

Kategorie	
Bezeichnung	Klass.Kriterium
U18	Unter 18 Jahre
privat	Privatkunde
gk	Grosskunde

Kunde		
Name	Vorname	Bezeichnung
Meier	Florian	NULL
Müller	Erich	privat
Huber	Max	gk

A1: **where** Bezeichnung not in (NULL, privat, gk)

U18 not in (NULL, privat, gk) -> not (U18 in (NULL, privat, gk)) ->

not (U18=NULL or U18=privat or U18=gk) -> not (Unknown or false or false) -> not Unknown -> Unknown

U18 nicht im Ergebnis. gk, privat ebenfalls nicht!

Aufgabe 3 (3)

Kategorie	
Bezeichnung	Klass.Kriterium
U18	Unter 18 Jahre
privat	Privatkunde
gk	Grosskunde

Kunde		
Name	Vorname	Bezeichnung
Meier	Florian	NULL
Müller	Erich	privat
Huber	Max	gk

A2: where der Unteranfrage:

(U18=NULL or U18=privat or U18=gk) -> unknown

Florian Meier kein Ergebnistupel -> U18 im Gesamtergebnis

gk, privat nicht im Ergebnis, weil not exists **false** liefert

Aufgabe 3 (4)

Vorsicht vor NULL-Werten:

- Dreiwertige Logik in SQL true und false, unknown bei Vergleichen mit NULL-Werten.
- Nur true-Fall in **where**-Klausel liefert Tupel
- Arithmetik: $1 + \text{NULL} = \text{NULL}$; $0 * \text{NULL} = \text{NULL}$
- Bei Gruppierung zählt NULL als eigene Gruppe!

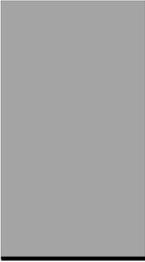
Aufgabe 3 (5)

and	True	Unknown	False
True	True	Unknown	False
Unknown	Unknown	Unknown	False
False	False	False	False

not	
True	False
Unknown	Unknown
False	true

Aufgabe 3 (6)

or	True	Unknown	False
True	True	True	True
Unknown	True	Unknown	Unknown
False	True	Unknown	False



Übung Datenbanksysteme
Besprechung Midtermklausur

15.12.2002



Informationen vorweg

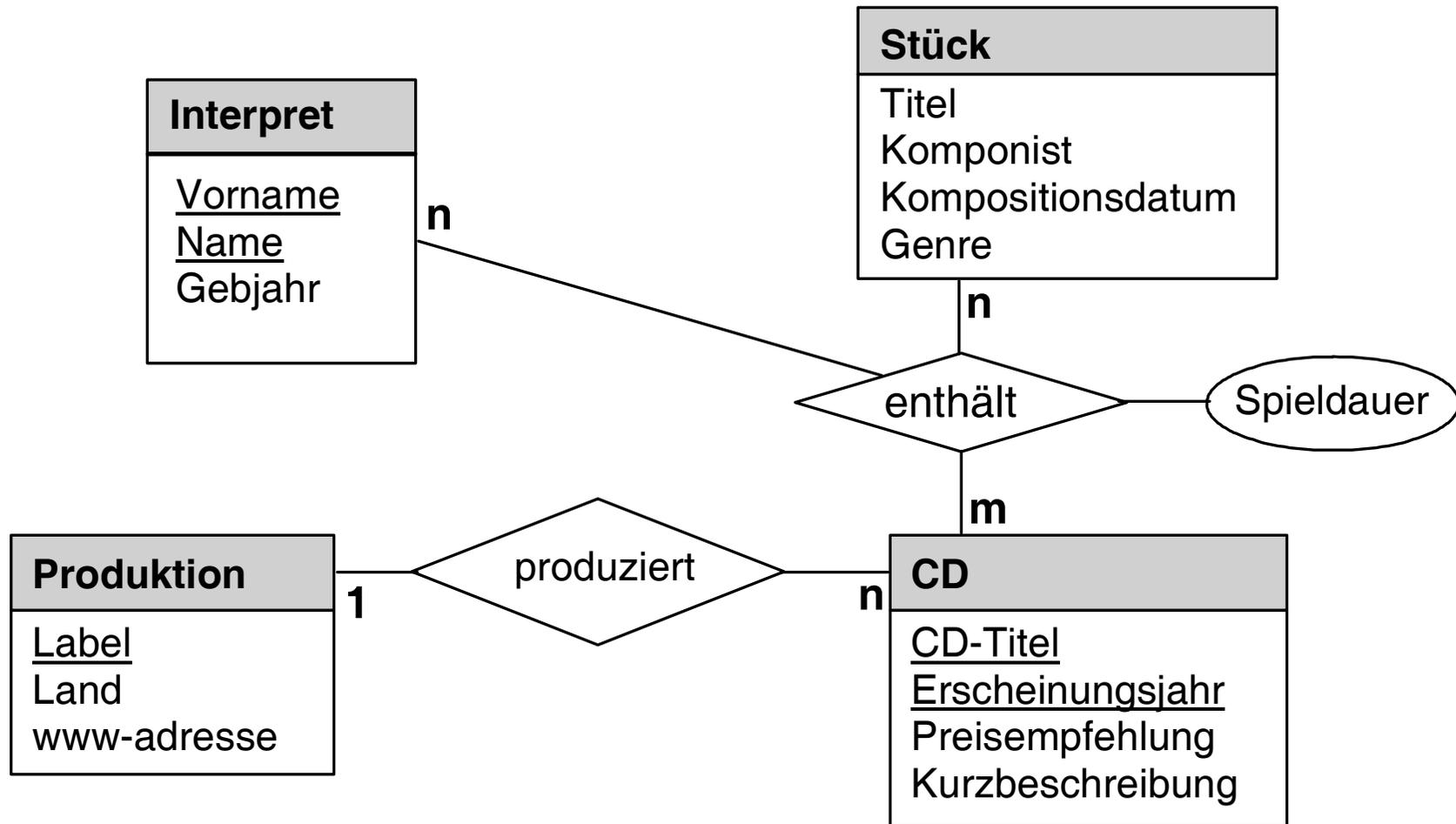
Wichtig:

- Noten gibt es erst bei der Final
- Punkte von Midterm (40) und Final (60) werden aufaddiert
- Punkteliste am Lehrstuhl oder unter <http://www3.in.tum.de/lehre/WS2003/DBS-uebung/Midterm-DB-Klausur-net.pdf>

Etwas Statistik:

- Teilgenommen haben 104 Studenten/Studentinnen
- Der Punkteschnitt ist 28,4
- Die Maximalpunktzahl ist 38
- Die Minimalpunktzahl ist 0

Aufgabe 1: Umsetzung ER zu Relationenschema



Aufgabe 1.1

Ermitteln Sie den minimalen Schlüssel.

- Ursprüngliche Idee:
Titel, Komponist, Kompositionsdatum
- Aber auch: Titel, Komponist
unter der Annahme, daß jeder Komponist
jeden Titel nur einmal vergibt.

Aufgabe 1.2

Umsetzung des E/R-Diagramms unter Berücksichtigung der Funktionalitäten (einschließlich der Datentypen und Schlüssel)

- **Stück:** {[Titel: string, Komponist: string, Kompositionsdatum:date, Genre:string]}
- **Interpret:** {[Vorname: string, Name: string, Gebjahr:date/integer]}

Aufgabe 1.2 (2)

- **CD:** {[CD-Titel:string, Erscheinungsjahr:date/integer, Preisempf:numeric, Kurzbeschreibung:string]}
- **Produktion:** {[Label: string, Land: string, www-adresse: string]}
- **Produziert:** {[Label: string, CD-Titel:string, Erscheinungsjahr:date/integer]}

Aufgabe 1.2 (3)

- **Enthält:** {[Vorname: string, Name: string, Titel: string, Komponist: string, CD-Titel:string, Erscheinungsjahr:date/integer, Spieldauer:time/integer(in sec)]}

Aufgabe 1.2 (4)

- Vereinfachen von CD und produziert (gleicher Schlüssel, 1:N Relationship)
- **CD**: {[CD-Titel:string, Erscheinungsjahr:date/integer, Preisempf:numeric, Kurzbeschreibung:string, Label:string]}

Aufgabe 1.3

Identifikatoren anstatt wertbasierter Schlüssel.

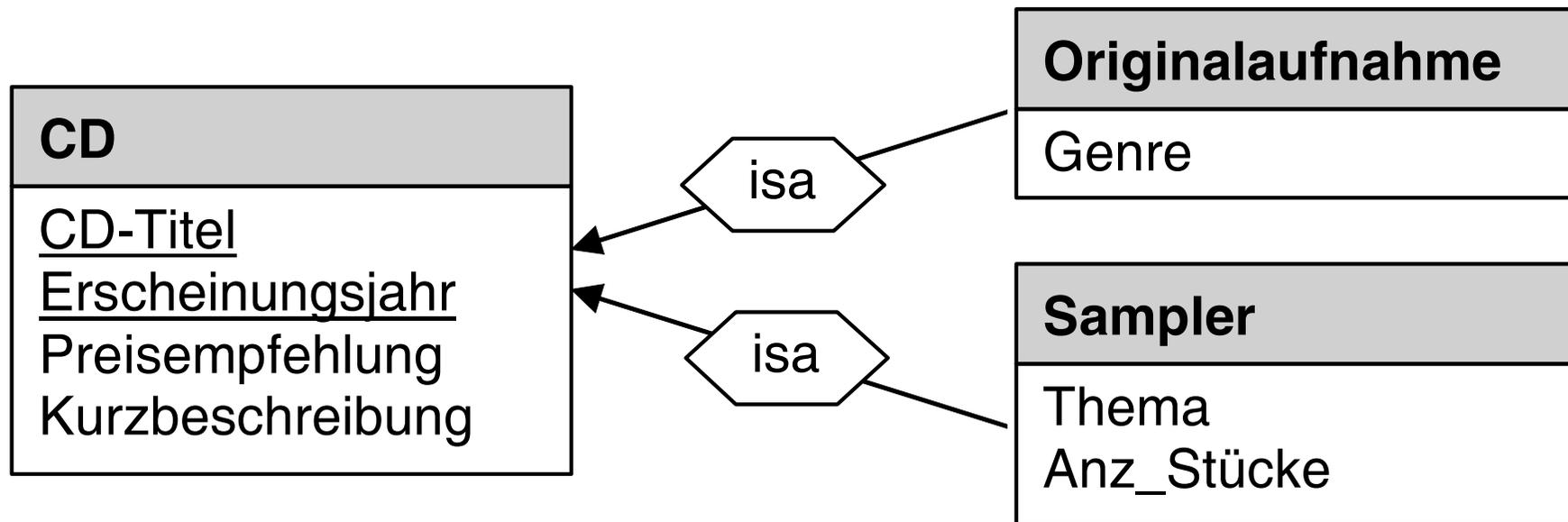
- Bei CD: sinnvoll, da
 - zusammengesetzter Schlüssel
 - dieser Schlüssel in zwei weiteren „Beziehungsrelationen“ eingebettet ist (*produziert* und *enthält*), was dort zu Speicherplatzersparnis führt
- Bei Interpret, Stück: sinnvoll, da zusammengesetzter Schlüssel aus string, der auch in *enthält* verwendet wird.

Aufgabe 1.3 (2)

- Falsche Begründungen:
 - Identifikatoren in allen Tabellen einfügen **(Nein!)**
 - Identifikatoren ersetzen Attribute **(Nein!)**
 - Identifikatoren stellen Schlüsseleigenschaft sicher **(Nein!)**

Beispiel ist so entworfen, daß z.B. Vorname, Name in Interpret Schlüssel ist!

Aufgabe 1.4



Aufgabe 1.4

- Zwei der folgenden Alternativen:

- **Tabelle pro Entität:**

CD: {[CD-Titel, Erscheinungsjahr,
Preiseempfehlung, Kurzbeschreibung]}

Originalaufnahme: {[CD-Titel, Erscheinungsjahr,
Genre]}

Sampler: {[CD-Titel, Erscheinungsjahr, Thema,
Anz_Stücke]}

Wichtig: Schlüssel übernehmen!

Aufgabe 1.4 (2)

- **Pushdown:**

Originalaufnahme: {[CD-Titel, Erscheinungsjahr,
Preisempfehlung, Kurzbeschreibung, Genre]}

Sampler: {[CD-Titel, Erscheinungsjahr,
Preisempfehlung, Kurzbeschreibung, Thema,
Anz_Stücke]}

**Extra Tabelle für weder
Sampler/Originalaufnahme:**

CD: {[CD-Titel, Erscheinungsjahr,
Preisempfehlung, Kurzbeschreibung]}

Aufgabe 1.4 (3)

- **Pullup:**

CD: {[CD-Titel, Erscheinungsjahr, *Typ*,
Preisempfehlung, Kurzbeschreibung, Genre,
Thema, Anz_Stücke]}

Bei fehlendem *Typ* Begründung notwendig:
Unterscheidung durch Testen auf Nullwerte

Aufgabe 2: Anfragen

Relationenschema

- **Besucher:** {[Name]}
- **Platz:** {[Platznr, Konzerthaus, Kategorie]}
- **reserviert:** {[Konzerthaus, Datum, Platznr, Besucher, Preis]}
- **Konzert:** {[Ktitel, Konzerthaus, Datum, Dirigent]}
- **Konzertstück:** {[Titel, Komponist]}
- **spielt:** {[Konzerthaus, Datum, Stücktitel]}

Aufgabe 2.1

- Tupelkalkül: *Konzerttitel und Dirigenten von Konzerten, die im Prinzregententheater aufgeführt werden/wurden.*

Selektion und Projektion auf einer Tabelle

$\{[k.Ktitel, k.Dirigent] \mid k \in \text{Konzert} \wedge$
 $k.Konzerthaus = \text{'Prinzregententheater'} \}$

**Umständlichere Formulierung über spielt
möglich aber unschön.**

Aufgabe 2.2

- *Domänenkalkül*: Platznr und Kategorie der Plätze, die für das Konzert am 7.12.2003 in der Philharmonie reserviert waren.

Implizite Joinformulierung:

- $\{[pn, kat] \mid \exists kh ([pn, kh, kat] \in \text{Platz}) \wedge \exists d, b, p ([kh, d, pn, b, p] \in \text{reserviert} \wedge d=7.12.2003 \wedge kh='Philharmonie'))\}$

Aufgabe 2.2 (2)

- Explizite Joinformulierung:
- $\{[pn, kat] \mid \exists k ([pn, k, kat] \in \text{Platz}) \wedge \exists kh, d, pno, b, p ([kh, d, pno, b, p] \in \text{reserviert} \wedge pn = pno \wedge k=kh \wedge d=7.12.2003 \wedge kh='Philharmonie')\}$

Generell:

Vorsicht bei Stelligkeit und Bezeichnung der Domänen ($[kat, pn, k]$ und dann falscher Join über k)

Aufgabe 2.3

- *Relationale Algebra:* Komponisten, die der Konzertbesucher Hans Meier in den von ihm besuchten Konzerten gehört hat.

$\pi_{\text{Komponist}} (\text{Konzertstück} \bowtie_{\text{Titel=Stückeritel}} (\text{spielt} \bowtie (\sigma_{\text{Besucher}='Hans Meier'} (\text{reserviert}))))$

$\pi_{\text{Komponist}} (\text{Konzertstück} \bowtie_{\text{Titel=Stückeritel}} (\text{spielt} \bowtie (\text{Konzert} \bowtie_{\text{Konzerthaus=Konzerthaus} \wedge \text{Datum=Datum}} (\sigma_{\text{Besucher}='Hans Meier'} (\text{reserviert}))))))$

Aufgabe 2.4

- **select distinct** k.Ktitel, k.Dirigent
from Konzert k, spielt s,
Konzertstück ks
where ks.Komponist = Mozart **and**
ks.Titel = s.Stücktitel **and**
s.Konzerthaus = k.Konzerthaus **and**
s.Datum = k.Datum;
distinct wegen mehreren Stücken von Mozart
in einem Konzert.