

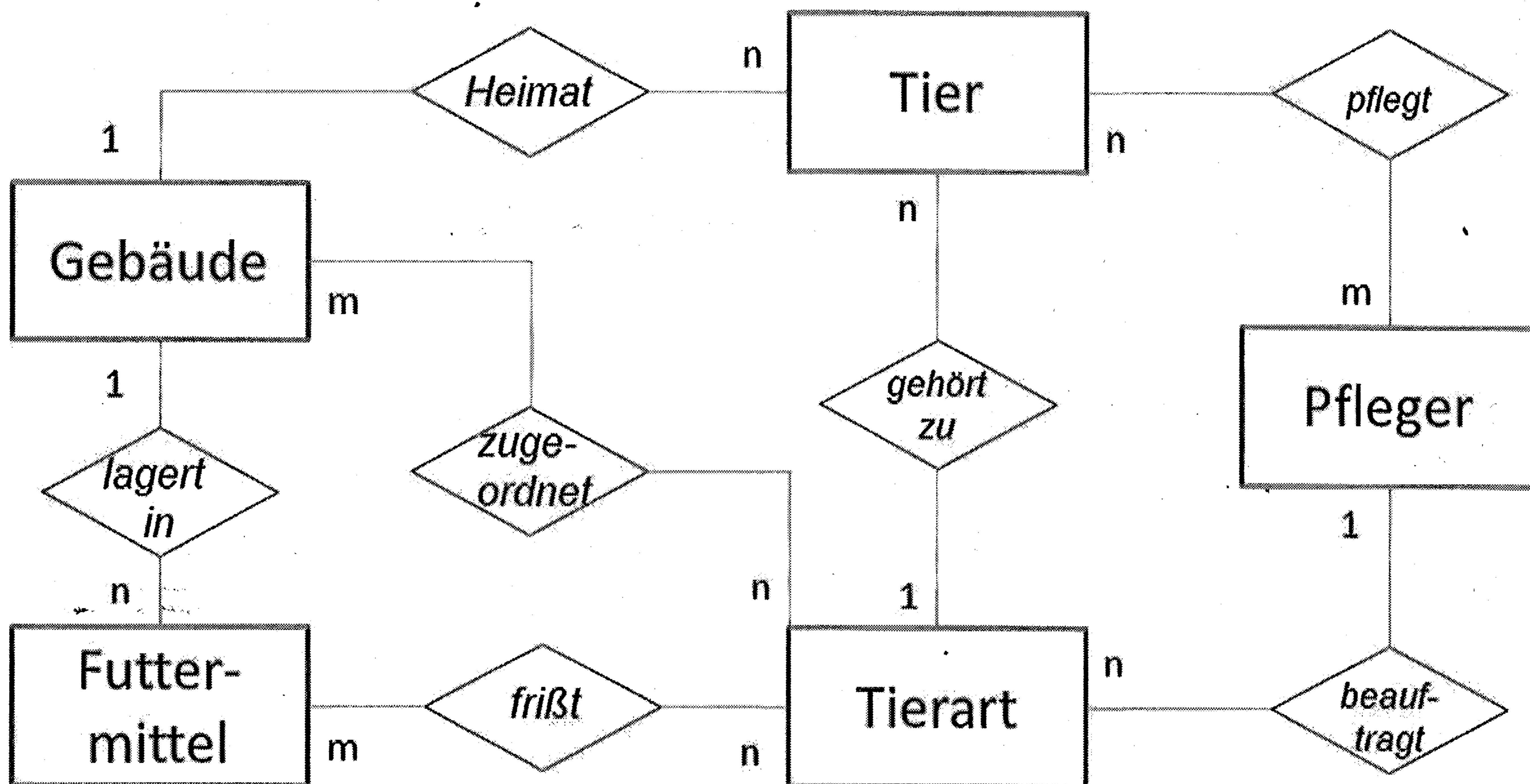
Lösungsvorschlag

Nach-Klausur Datenbanken 1 (1671)

SS 2015

Aufgabe 1: ER Modell

(21 Punkte)



Es ergeben sich aus der Aufgabenstellung die 5 Entities *Gebäude*, *Futtermittel*, *Tierart*, *Tier* und *Pfleger*.

Beziehungen:

- *Gehört zu*: Jedes Tier gehört zu einer Tierart, es können aber mehrere Tiere der gleichen Art im Zoo leben.
- *Frißt*: Tiere sollen eine abwechslungsreiche Nahrung bekommen, also muss eine Tierart mehrere Futtermittel haben, die ihrerseits von mehreren Tierarten gefressen werden.
- *Lagert in*: Pro Futtermittel gibt es ein Lager in einem Gebäude, es ist nicht ausgeschlossen, dass mehrere Läger in einem Gebäude sind.
- *Zugeordnet*: Tierarten können in mehreren Gebäuden leben, in einem Gebäude können auch mehrere Tierarten multikulturell zusammenleben.
- *Heimat*: Jedes Tier hat ein Heimatgebäude.
- *Pflegt*: Ein Pfleger pflegt mehrere Tiere, und die Tiere haben mehrere Pfleger.
- *Beauftragt*: Jede Tierart hat einen beauftragten Pfleger, es ist nicht ausgeschlossen, dass ein Pfleger für mehrere Tierarten beauftragt sein kann.

Nicht dargestellt werden können die Constraints (zumindest nicht auf einfache Weise), also die Aussage, dass die Tiere in einem Gebäude beheimatet sein müssen, das für die Tierart ausgelegt ist.

Aufgabe 2: SQL

(21 Punkte)

- a) Eine Tupelvariable bezeichnet ein Tupel einer Relation in einer Query. Die Verwendung von Tupelvariablen ist erforderlich, wenn innerhalb einer Abfrage verschiedene Tupel derselben Relation betrachtet werden.
- b) Eine View ist eine dynamisch erstellte Tabelle und wird über eine Query definiert.
- c) Eine im System physikalisch vorhandene Tabelle wird als Basistabelle bezeichnet (im Gegensatz zu einer View, die auch als abgeleitete Tabelle bezeichnet wird).
- d)

B	A
50	30
71	66

- e) 160
- f) 145,25
- g) Im Unterschied zu dem Vergleich in d)
... where c in ...

wird beim Join eine Verdoppelung des 1. Tupels vorgenommen, da der Wert mit 2 Tupeln von r2 gejoined wird. Das führt zu einer zusätzlichen Zeile im Ergebnis:

B	A
50	30
50	30
71	66

Um diese doppelte Zeile zu eliminieren, muss ein DISTINCT eingefügt werden.

```
select distinct r1.b, r1.a  
from r1 join r2 on r1.c = r2.c  
where r2.d < (select max(d) from r2);
```

Aufgabe 3: Normalformen

(8 Punkte)

- a) Wann ist ein Attribut Y von einem anderen Attribut X funktional abhängig?

Ein Attribut Y ist funktional abhängig von einem anderen Attribut X wenn es zu jedem Wert x_1 von X höchstens einen Wert y_1 von Y gibt.

- b) Definieren Sie die 2. und 3. Normalform.

Eine Relation ist in der 2. Normalform, wenn jedes Nichtschlüsselattribut voll funktional abhängig von jedem Schlüsselkandidaten ist.

Eine Relation ist in der 3. Normalform, wenn es keine transitiven Abhängigkeiten von Nichtschlüsselattributen gibt.

Formal:

Ein Relationenschema R mit Fd-Menge F ist in 3. Normalform, wenn für alle $X \rightarrow A \in F^+$ mit $A \notin X$ gilt: X enthält einen Schlüssel für R oder A ist Schlüsselattribut.

Anders formuliert, die 3. Normalform ist verletzt, wenn es eine funktionale Abhängigkeit $X \rightarrow A$ gibt, so dass X Nichtschlüssel und A Nichtschlüsselattribut ist.

- c) Kann eine Relation mit einem Schlüssel, der aus einem einzelnen Schlüsselattribut besteht, die 2. Normalform verletzen? Begründen Sie Ihre Antwort. (3 Punkte)

Eine Relation mit nur einem Schlüsselattribut ist immer in 2. Normalform.

Da es nur ein Schlüsselattribut gibt, ist jedes Attribut voll funktional vom Schlüssel, also ist die Relation in 2NF.

Da es aber nicht ausgeschlossen ist, dass es weitere Abhängigkeiten zwischen Nicht-Schlüsselattributen gibt, ist sie nicht unbedingt in 3NF.

- d) Gegeben sei die folgende Relation. Ermitteln Sie, welche einzelnen Attribute von anderen einzelnen Attributen abhängig sein können. Falls keine funktionale Abhängigkeit vorliegen kann, geben Sie an, warum das nicht sein kann.

A	B	C
a1	b1	c1
a2	b5	c2
a3	b2	c3
a2	b4	c2
a4	b3	c2

Mögliche FD	FD möglich (ja, nein)	Begründung
$A \rightarrow B$	Nein	Der Wert a2 hat zwei unterschiedliche B-Werte: b4 und b5
$A \rightarrow C$	Ja	Der Wert a2 wird in beiden Tupeln auf den gleichen C-Wert abgebildet.
$B \rightarrow A$	Ja	B hat nur unterschiedliche Werte, damit kann es nicht sein, dass ein B-Wert auf unterschiedliche A-Werte abgebildet wird.
$B \rightarrow C$	Ja	Siehe oben
$C \rightarrow A$	Nein	c2 wird auf unterschiedliche Werte in A abgebildet.
$C \rightarrow B$	Nein	c2 wird auch auf unterschiedliche Werte in B abgebildet.

Aufgabe 4: SQL

(26 Punkte)

- a) Suche alle Personen, die sowohl Autoren von Artikeln sind als auch als Herausgeber einer Zeitschrift fungieren.

(6 Punkte)

```
SELECT    p.Vorname, p.Name
FROM      Person p
WHERE     p.p_id in
          ( SELECT    p_ref
            FROM      Ist_Herausgeber)
AND       p.p_ID in
          ( SELECT    p_ref
            FROM      Ist_Autor)
```

Alternative Lösung:

```
SELECT    p.Vorname, p.Name
FROM      Person p
          JOIN  Ist_Herausgeber h on p.p_ID = h.p_ref
          JOIN  Ist_Autor a on p.p_ID = a.p_ref
```

- b) Ähnlich wie die Anfrage in a): Suche alle Personen, die Herausgeber einer Zeitschrift sind, und, falls sie auch Artikel in einer der Zeitschriften veröffentlicht haben, gib die Artikel-Nr. an.

(4 Punkte)

```
SELECT    p.Vorname, p.Name, a.a_ref
FROM      Person p
          JOIN  Ist_Herausgeber h on p.p_ID = h.p_ref
          LEFT OUTER JOIN  Ist_Autor a on p.p_ID = a.p_ref
```

- c) Suche alle Herausgeber, die auch in der von Ihnen herausgegeben Zeitschrift bereits als Autoren aufgetreten sind.

(6 Punkte)

```
SELECT    p.Vorname, p.Name
FROM      Person p
          JOIN Ist_Herausgeber h    ON p.p_ID = h.p_ref
          JOIN Zeitschrift z        ON h.z_ref = z.z_ID
/* Damit haben wir die Herausgeber und ihre Zeitschriften*/

          JOIN Ist_Autor a          ON p.p_ID = a.p_ref
          JOIN Artikel art          ON a.a_ref = art.art_ID
/* Damit haben wir die Herausgeber, die Autoren sind zusammen mit ihren Artikeln */

          JOIN Ausgabe aus         ON art.aus_ref = aus.aus_ID
          JOIN Zeitschrift z2       ON aus.z_ref = z2.z_ID
/* Damit haben wir die Zeitschriften zu den Artikeln der Autoren */

WHERE z1.z_ID = z2.z_ID
/*      Damit wird überprüft, ob die Autoren in der Zeitschrift veröffentlicht haben, die sie
auch herausgeben
*/
```

- d) Gibt es unterschiedliche Artikel mit gleichem Namen?

(5 Punkte)

```
SELECT    Titel
FROM      Artikel
GROUP BY  Titel
HAVING    count(Titel) > 1
```

- e) Gib die Namen der Personen aus, die an mehr als 20 Artikeln mitgewirkt haben.

(5 Punkte)

```
SELECT    p.Vorname, p.Name
FROM      Person p
WHERE     p.p_ID in
          ( SELECT    p_ref
            FROM      Ist_Autor
            GROUP BY  p_ref
            HAVING    count (p_ref) > 20 )
```

Aufgabe 5: Optimierung

(21 Punkte)

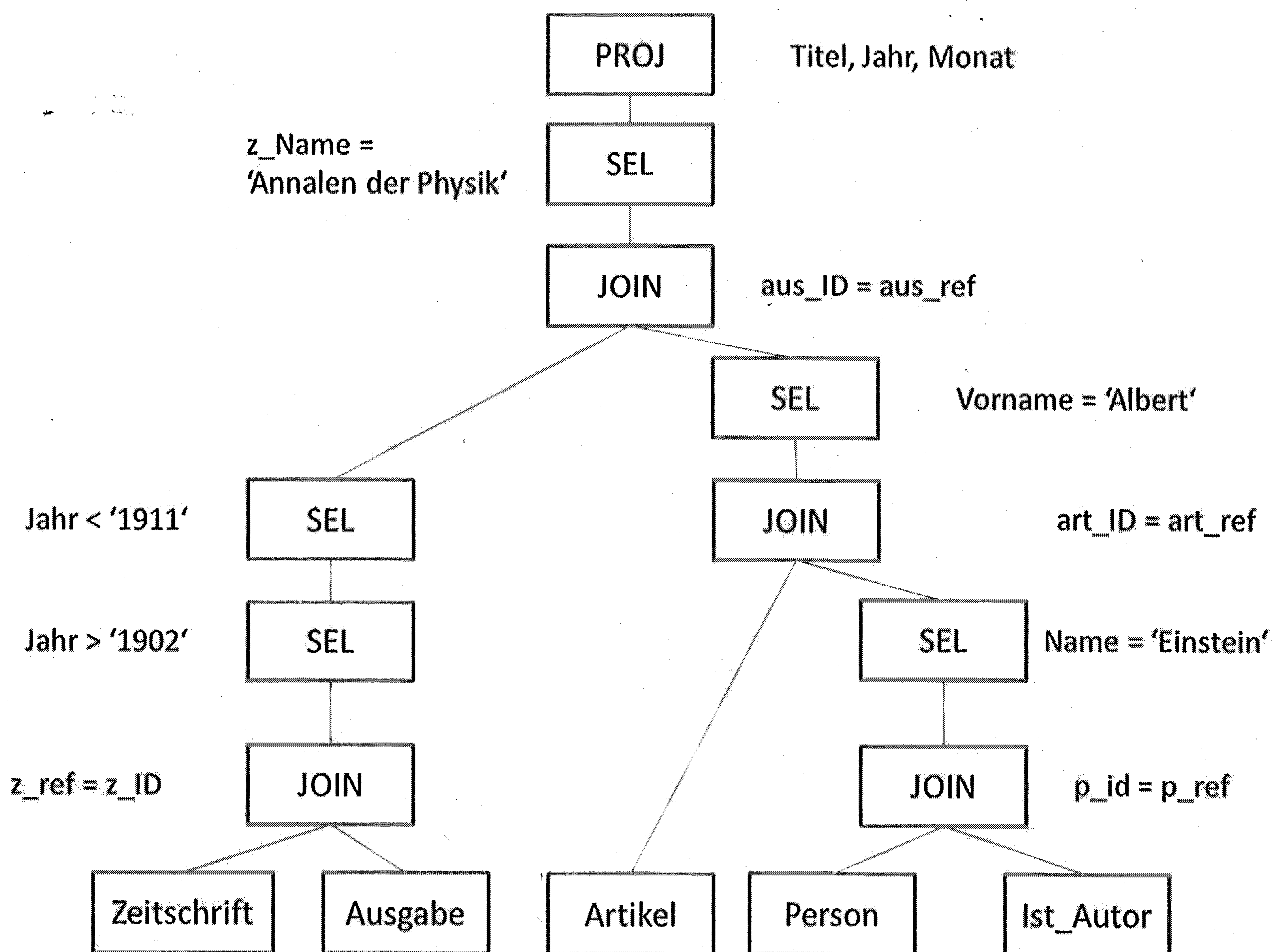
a)

(4 Punkte)

Die Query liefert die Titel aller Artikel, die von Albert Einstein (mit-)geschrieben wurden und in den Jahren 1903 bis 1910 in der Zeitschrift ‚Annalen der Physik‘ erschienen sind, zusammen mit dem Erscheinungsjahr und dem Erscheinungsmonat der Zeitschrift.

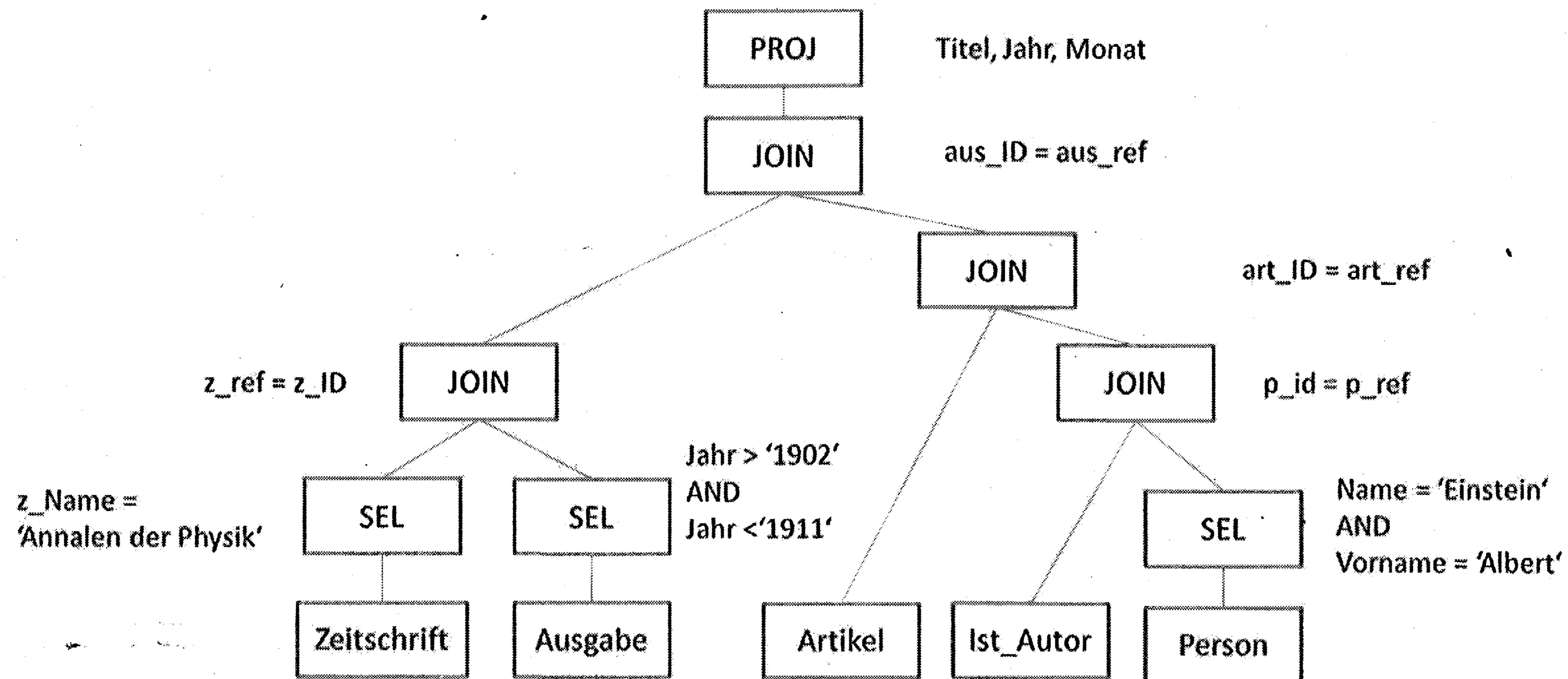
b)

(5 Punkte)



c)

(4 Punkte)



Erklärung: Selektionen werden möglichst weit zu Blättern geschoben, insbesondere vor die Joins, mehrere Selektionen auf einer Relation werden zusammengefasst zu einer Selektion mit mehreren Selektionsprädikaten, die durch AND verknüpft sind.

d)

(4 Punkte)

$\pi_{\text{Titel, Jahr, Monat, Jahrgang}}$

(

(

($\sigma_{z_Name = 'Annalen der Physik'}$
(Zeitschrift)

)

$\bowtie_{z_ID = z_ref}$

($\sigma_{\text{Jahr} < '1911' \text{ AND } \text{Jahr} > '1902'}$
(Ausgabe)

)

)

$\bowtie_{aus_ID = aus_ref}$

(Artikel $\bowtie_{art_ID = art_ref}$

($\sigma_{\text{Vorname} = 'Albert' \text{ AND } \text{Name} = 'Einstein'}$ (Person)

$\bowtie_{p_ID = p_ref}$

Ist_Autor

)

)

)