

Übung zur Vorlesung *Grundlagen: Datenbanken* im WS20/21
Christoph Anneser, Josef Schmeißer, Moritzichert, Lukas Vogel (gdb@in.tum.de)
<https://db.in.tum.de/teaching/ws2021/grundlagen/>

Blatt Nr. 02

Tool zum Üben der relationalen Algebra: <https://dbis-uibk.github.io/relax/calc>.
Für das Unischema im Dropdown links „Kemper Datenbanksysteme“ auswählen.

Hausaufgabe 1

Modellieren Sie ein Wahlsystem für die 18. Landtagswahl in Bayern. Das System soll Wahlberechtigte und Wahlkandidaten in ihren jeweiligen Wahl- und Stimmkreisen verwalten. Da Wahlberechtigte einzelne Kandidaten einer Liste mit ihrer Zweitstimme wählen können, soll es auch die Listen der jeweiligen Parteien modellieren. Es muss außerdem sicherstellen, dass das Wahlrecht eingehalten wird. Dazu gehört insbesondere, dass jede Person nur höchstens eine Stimme abgeben kann und die Wahl geheim ist (also auch das System selbst nicht wissen kann, wer wie abgestimmt hat). Betrachten Sie für die Modellierung die folgenden Beschreibungen:

Partei Es gibt mehrere Parteien mit unterschiedlichen Namen. Jede Partei kann in jedem Stimmkreis eine Kandidatin für die Erststimme und in jedem Wahlkreis eine Liste für die Zweitstimme aufstellen.

Wahlkreis Bayern hat zum Zeitpunkt der 18. Landtagswahl genau folgende Wahlkreise: Oberbayern, Niederbayern, Oberpfalz, Oberfranken, Mittelfranken, Unterfranken und Schwaben. Ein Wahlkreis enthält mehrere Stimmkreise. Pro Wahlkreis kann jede Partei mit einer Liste für die Zweitstimme antreten.

Stimmkreis Jeder Stimmkreis (z.B. „117 Freising“) ist genau einem Wahlkreis zugeordnet. Eine Partei kann hier eine Kandidatin für die Erststimme aufstellen. Allerdings kann es auch parteilose Kandidaten geben, die nur für sich selbst antreten. Die Wahlberechtigten und deren Stimmen werden in den Stimmkreisen verwaltet.

Liste Eine Liste besteht aus mehreren Kandidaten einer Partei. Sie ist spezifisch für einen Wahlkreis. Wahlberechtigte können mit ihrer Zweitstimme eine ganze Liste oder einen bestimmten Kandidaten einer Liste wählen.

Kandidatin Eine Kandidatin kann in ihrem Stimmkreis antreten und dann mit der Erststimme gewählt werden, unabhängig davon, welcher Partei sie angehört (oder ob sie überhaupt in einer ist). Zusätzlich kann sie auf der Liste einer Partei stehen. Das bedeutet, dass sie von Wahlberechtigten mit der Zweitstimme direkt gewählt werden kann.

Wahlberechtigte Wahlberechtigte leben in einem Stimmkreis, in dem sie eine Erst- und eine Zweitstimme abgeben können. Mit der Erststimme können sie genau eine Kandidatin, die in ihrem Stimmkreis antritt, wählen. Mit der Zweitstimme können sie entweder eine Kandidatin ihrer zugehörigen Wahlkreisliste oder die Liste selbst ohne die Auswahl einer bestimmten Kandidatin wählen. Sie können auch jede Stimme individuell ungültig abgeben.

Weiter Informationen zum bayrischen Wahlrecht finden Sie hier: <https://www.wahlrecht.de/landtage/bayern.htm>.

Entwerfen Sie ein ER-Modell für dieses System. Sie müssen dabei nur die in der Aufgabe erwähnten Sachverhalte berücksichtigen.

- a) Identifizieren Sie alle relevanten Entity-Typen und ihre Attribute.
- b) Beschreiben Sie alle notwendigen Relationships zwischen den Entity-Typen. Achten Sie darauf, dass das Wahlrecht eingehalten wird.
- c) Ergänzen Sie die Relationships mit Funktionalitätsangaben. Fügen Sie auch (min,max)-Angaben hinzu.
- d) Bildet Ihr Modell die Realität komplett ab? Welche Gegebenheiten kann Ihr Modell nicht ausdrücken? Warum nicht?

Lösung: Abbildung 1 zeigt ein mögliches ER-Diagramm. Abbildung 2 enthält zusätzlich Funktionalitätsangaben, Abbildung 3 zusätzlich (min,max)-Angaben.

- a) Aus der obigen Liste ergeben sich die zwingend notwendigen Entity-Typen. Die Beispiellösung modelliert zusätzlich die Erst- und Zweitstimme als getrennte Entity-Typen, da sie mit jeweils anderen Entity-Typen in Beziehung stehen. Der Listenplatz ist als getrennter Entity-Typ modelliert um über sein Attribut „Rang“ den Rang der Kandidatin auf einer Liste ausdrücken zu können.

Die Beschreibungen enthalten nicht alle Informationen über die erforderlichen Attribute der Entity-Typen, also müssen diese nach eigenem Ermessen hinzugefügt werden. Die meisten Entity-Typen haben z.B. einen Namen. Stimmkreise haben eine Nummer, Wahlberechtigte eine Anschrift (benötigt um die Wahlunterlagen versenden zu können) und Kandidaten einen Beruf (der auf dem Stimmzettel stehen muss). In der Beispiellösung haben Stimmen zusätzlich eine „ist gültig“-Attribut, da das Wahlrecht zwischen einer ungültigen und einer nicht abgegebenen Stimme unterscheidet.

- b) Besonders zu beachten ist hier, dass es keine Beziehung zwischen einer Wahlberechtigten und ihrer Stimme geben darf. Die Systemadministratoren, die Programmierer oder der Staat selbst könnten sonst herausfinden wer wen gewählt hat: Die Wahl wäre, anders als vom Wahlrecht gefordert, nicht mehr geheim.

Von besonderem Interesse sind die Relationen zwischen Zweitstimme, Liste und Listenplatz. Da man mit seiner Zweitstimme *entweder* einen Kandidaten auf einer Liste *oder* eine ganze Liste wählt, muss es eine Beziehung zu beiden Entitätstypen geben. Die Anforderung, dass nur eine der beiden Beziehungen bestehen darf, lässt sich mit einem ER-Diagramm nicht ausdrücken.

Die beiden ternären Beziehungen lesen sich wie folgt: „Eine Partei stellt pro Wahlkreis bis zu eine Liste auf“ und „Eine Kandidatin kann auf bis zu einer Liste einen Listenplatz haben. Eine Liste hat beliebig viele Listenplätze“. Beide Beziehungen könnte man jeweils auch äquivalent durch zwei binäre Beziehungen abbilden.

- c) Abbildung 2 zeigt das ER-Diagramm mit Funktionalitätsangaben, Abbildung 3 zeigt es mit (min,max)-Angaben.

Dabei sind (min, max)-Angaben ausdrucksstärker als Funktionalitätsangaben. Mit ihnen lässt sich z.B. ausdrücken, dass

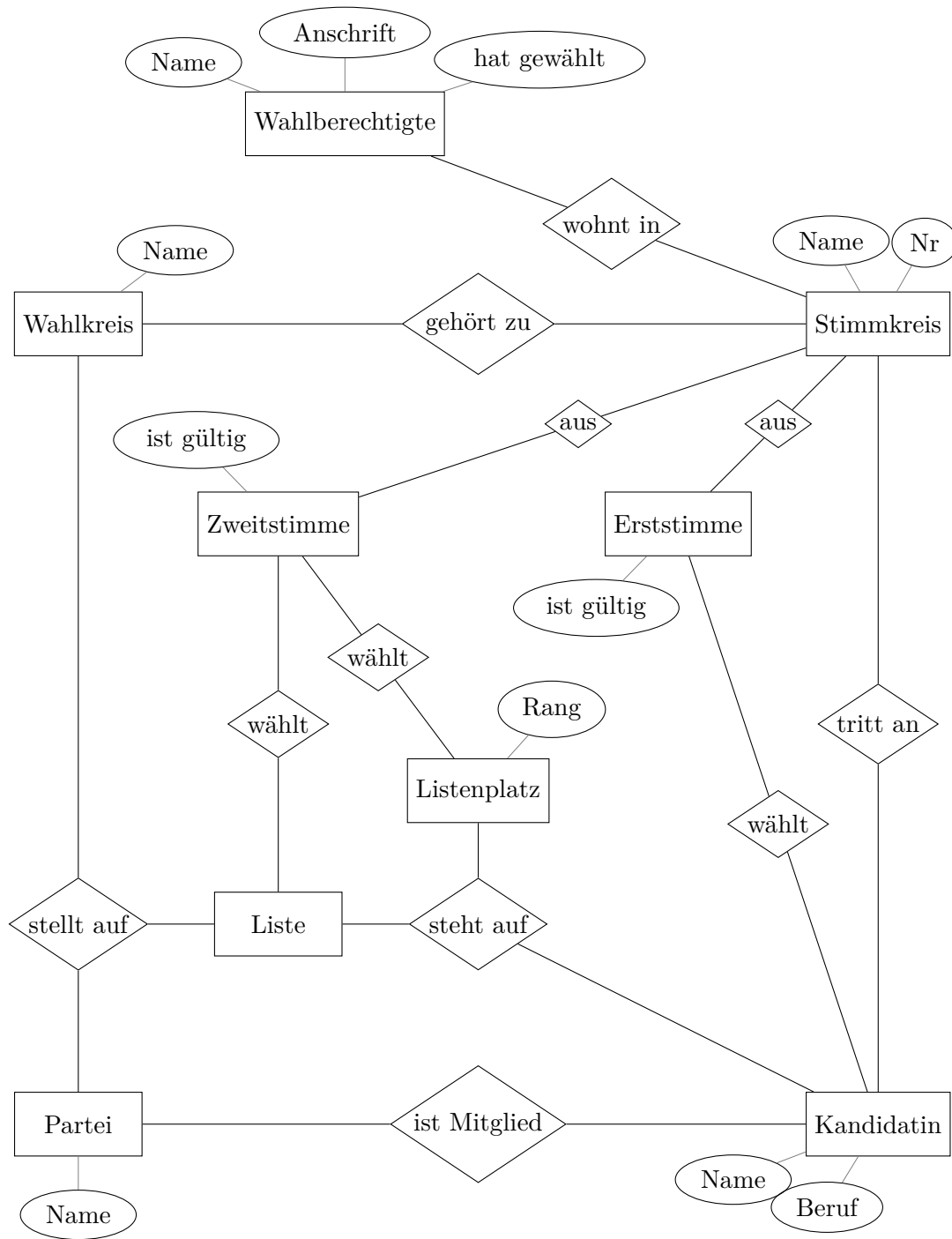


Abbildung 1: ER-Diagramm zur Landtagswahl

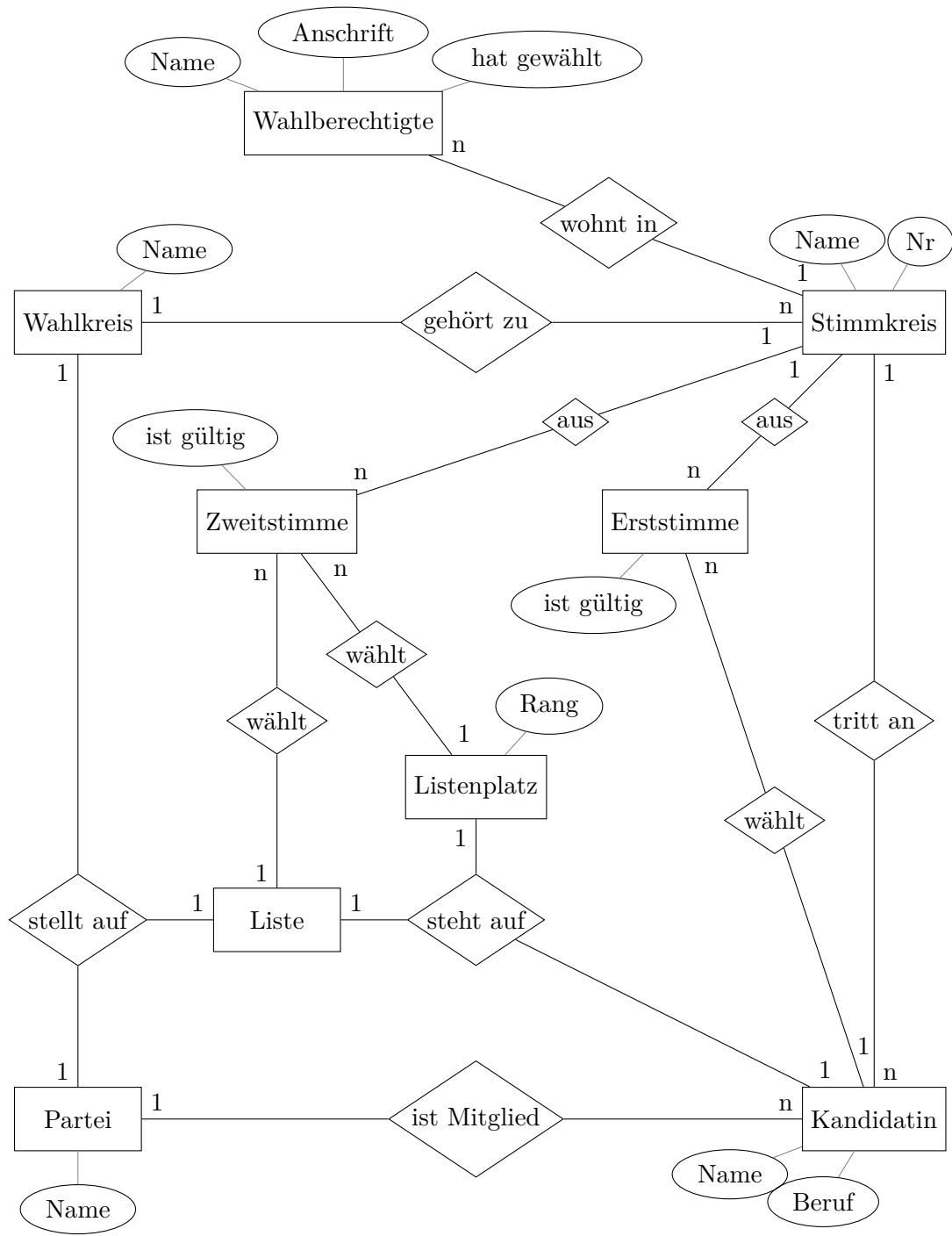


Abbildung 2: ER-Diagramm zur Landtagswahl mit Funktionalitätsangaben

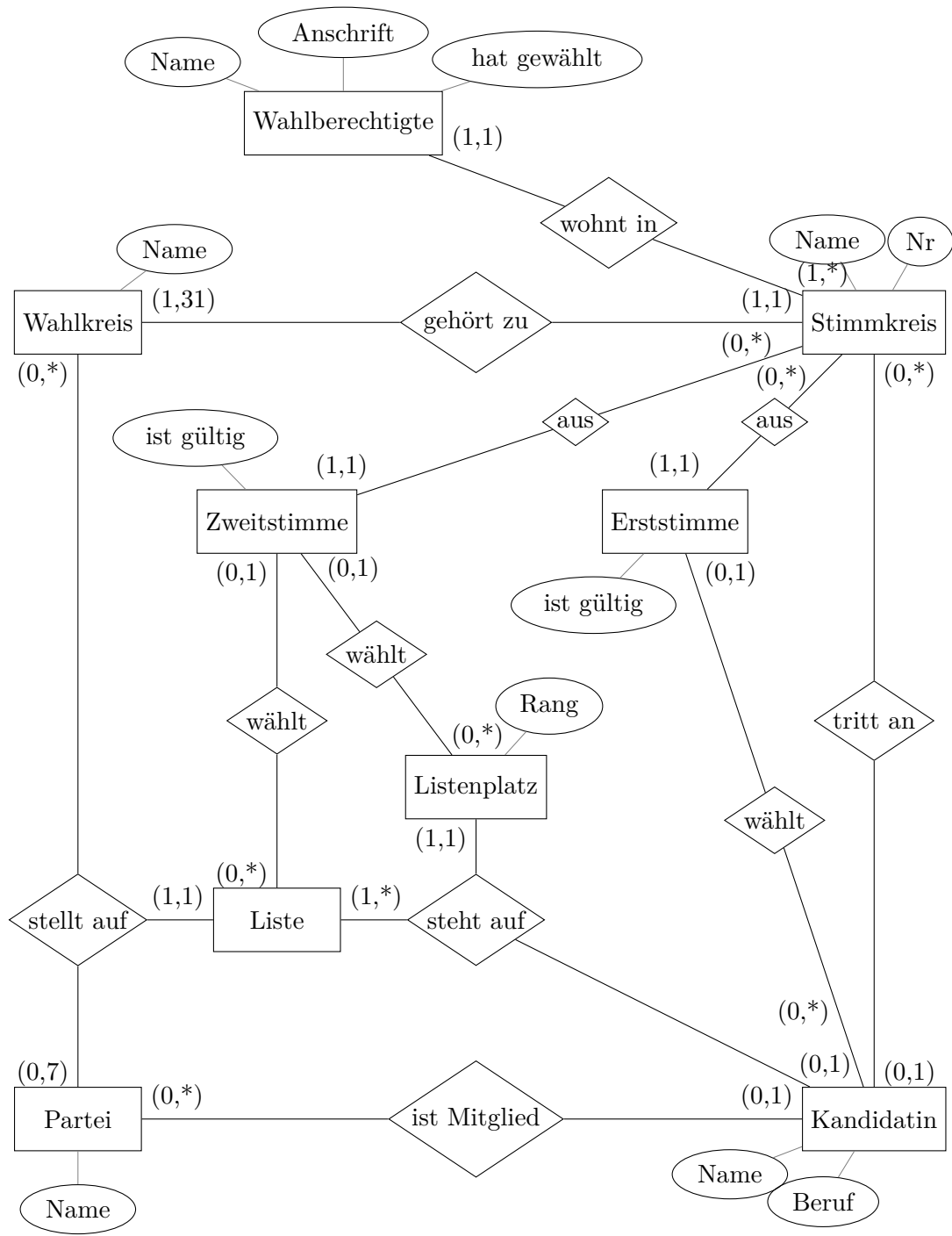
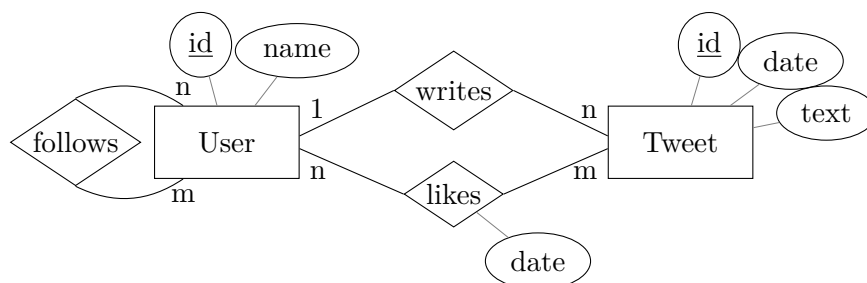


Abbildung 3: ER-Diagramm zur Landtagswahl mit (min,max)-Angaben

- es in der aktuellen Landtagswahl pro Wahlkreis maximal 31 Stimmkreise (in Oberbayern) gab,
 - eine Partei höchstens genau einmal in jedem der sieben Wahlkreise antreten kann ((0, 7) an „Partei“)
 - eine Partei keine Listen aufstellen ((0, 7) an „Partei“) - und ein Wahlkreis keine Liste haben ((0, *) an „Wahlkreis“) muss (jeweils in der ternären Beziehung „stellt auf“),
 - eine Kandidatin nicht Mitglied einer Partei sein muss, und
 - eine Kandidatin nicht auf einer Liste stehen muss (jeweils (0, 1) an „Kandidatin“).
- d) Dieses Modell bildet natürlich nicht die Realität in allen Einzelheiten ab. Hier einige Begebenheiten, die wir entweder nicht modelliert haben, oder die ein ER-Diagramm alleine nicht abbilden kann:
- Kandidaten selbst sind wahlberechtigt
 - Eine Kandidatin, die sich in ihrem eigenen Stimmkreis um das Direktmandat bewirbt darf in diesem nicht auf der Wahlkreisliste stehen (in anderen Stimmkreisen des Wahlkreises aber schon)
 - Man kann mit der Erststimme nur eine Kandidatin aus dem eigenen Stimmkreis wählen
 - Man kann mit der Erststimme nur eine Kandidatin wählen, die sich auf das Direktmandat bewirbt
 - Eine Kandidatin kann sich nur in ihrem Stimmkreis um das Direktmandat bewerben
 - Mit der Zweitstimme wählt man *entweder* einen Kandidaten auf einer Liste *oder* eine ganze Liste, aber nicht beides
 - Briefwahl
 - Hochrechnungen und Auszählungen dürfen erst nach Schließung der Wahllokale veröffentlicht werden. So wird sichergestellt, dass kein Wähler beeinflusst wird. Das System sollte daher garantieren, dass niemand vorher Zwischenergebnisse einsehen kann (auch keine Systemadministratoren oder Entwickler, da diese selbst wahlberechtigt sind).

Hausaufgabe 2

Gegeben sei folgendes ER-Diagramm, das User, deren Tweets, Likes und Follows modelliert:



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

- b) Verfeinern Sie das relationale Schema durch Elimination von Relationen.
- c) Kann das Attribut *date* des Entity-Typs *Tweet* stattdessen der Relationship *writes* zugeordnet werden? Kann das Attribut *date* der Relationship *likes* dem Entity-Typen *Tweet* zugeordnet werden?

Lösung:

a)

User : { [id,name] }
 Tweet : { [id,date, text] }
 follows : { [follower_id, follows_id] }
 writes : { [tweet_id,user_id] }
 likes : { [user_id, tweet_id, date] }

- b) Die Relationen *Tweet* und *writes* können zusammengefasst werden, da sie den gleichen Schlüssel haben und die Relationship eine 1-zu-n-Beziehung ist. Eine weitere Verfeinerung ist nicht möglich.

User : { [id,name] }
 Tweet : { [id,user_id, date, text] }
 follows : { [follower_id, follows_id] }
 likes : { [user_id, tweet_id, date] }

- c) Das Attribut *date* kann auch *writes* zugeordnet werden. Insbesondere nach der Verfeinerung sind alle Attribute von *writes* und von *Tweet* in einer Relation zusammengefasst. Für das relationale Schema ist es also egal, ob das Attribut ursprünglich zu *writes* oder zu *Tweet* gehört hat.

Das Attribut *date* der Relation *likes* kann keinem anderen Entity-Typen zugeordnet werden, da weder *User* noch *Tweet* wegen ihrer n-zu-m-Beziehung ein eindeutiges „like_date“-Attribut haben können.

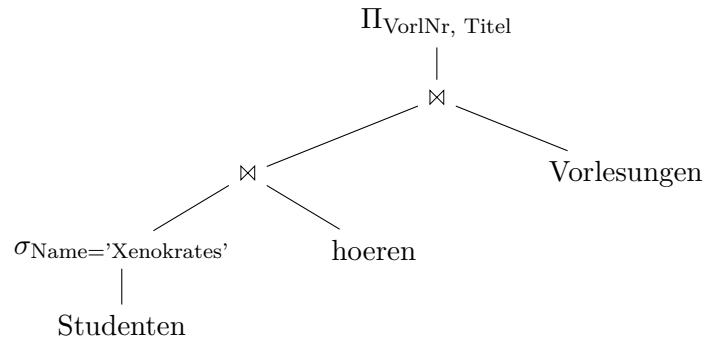
Hausaufgabe 3

Formulieren Sie die folgenden Anfragen auf dem bekannten Universitätsschema in Relationalalgebra. Geben Sie die Lösungen in der Operatorbaum-Darstellung an.

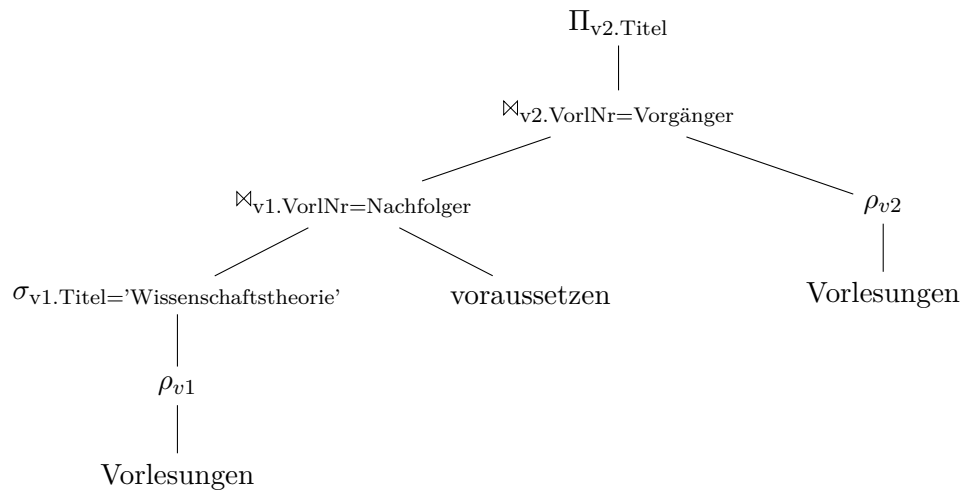
- a) Geben Sie alle *Vorlesungen* an, die der *Student* Xenokrates gehört hat.
- b) Geben Sie die Titel der direkten Voraussetzungen für die *Vorlesung* Wissenschaftstheorie an.
- c) Geben Sie Paare von *Studenten*(-Namen) an, die sich aus der *Vorlesung* Grundzüge kennen.

Lösung:

- a) Geben Sie alle *Vorlesungen* an, die der *Student* Xenokrates gehört hat.



b) Geben Sie die Titel der direkten Voraussetzungen für die *Vorlesung* Wissenschaftstheorie an.



c) Geben Sie Paare von *Studenten*(-Namen) an, die sich aus der *Vorlesung* Grundzüge kennen.

