

# 1. Übungsblatt

## 3.0 VU Datenmodellierung

27. April 2017

### Allgemeines

In diesem Übungsteil werden Sie den Entwurf einer kleinen Datenbank üben, deren Überführung in das Relationenschema, sowie die relationale Algebra und den Relationenkalkül kennenlernen.

Lösen Sie die Beispiele **eigenständig**, denn bei der Prüfung und in der Praxis sind Sie auch auf sich alleine gestellt. Wir weisen Sie darauf hin, dass abgeschriebene Lösungen mit 0 Punkten beurteilt werden.

Geben Sie ein einziges PDF Dokument ab (max. 5MB). Erstellen Sie Ihr Abgabedokument computerunterstützt. Wir akzeptieren keine gescannten handschriftlichen PDF-Dateien.

### Deadlines

<b>bis 11.05.</b>	<b>06:55 Uhr</b>	Upload der Abgabe über TUWEL
ab 18.05.	13:00 Uhr	Feedback in TUWEL verfügbar
bis 21.05.	23:59 Uhr	Reservierung eines Termins für das Tutorengespräch
bis 21.05.	23:59 Uhr	Reservierung eines Termins für das Abgabegespräch

(Terminreservierung für Tutoren- und Abgabegespräch über **TUWEL**)

### Tutorengespräch

1. Sie müssen sich über TUWEL zu einem Tutorengespräch anmelden. Bitte machen Sie das rechtzeitig, je später Sie sich anmelden, umso eingeschränkter ist das Terminangebot.
2. Wenn Sie kein Blatt abgegeben haben, sind Sie nicht zum Tutorengespräch zugelassen.
3. Sie kommen mit Ihrem Studierendenausweis zu der von Ihnen reservierten Zeit vorbei, und absolvieren das Tutorengespräch. Sie erhalten dabei wertvolles Feedback zu Ihrem Übungsblatt. Das Gespräch ist verpflichtend und dauert ca. 30 Minuten.

4. Nützen Sie das Tutorengespräch um hilfreiches Feedback zu Ihrem Übungsblatt zu erhalten: Gehen Sie vor dem Gespräch Ihre korrigierte Lösungen durch, identifizieren Sie Unklarheiten und stellen Sie den TutorInnen entsprechende **Fragen**. Trauen Sie sich zuzugeben, wenn Sie etwas nicht verstanden haben. Ihre Fragen haben keinerlei negativen Einfluss auf Ihre Note!

## Abgabegespräch

1. Sie müssen sich über TUWEL zu einem Abgabegespräch anmelden. Bitte machen Sie das rechtzeitig, je später Sie sich anmelden, umso eingeschränkter ist das Terminangebot.
2. Sie kommen mit Ihrem Studierendenausweis zu der von Ihnen reservierten Zeit vorbei, und absolvieren das Abgabegespräch. Stoffgebiet des Abgabegesprächs sind die mit dem Übungsblatt abgedeckten Themengebiete. Wir setzen voraus, dass Sie sich mit Ihrer korrigierten Abgabe auseinandergesetzt haben. Das Abgabegespräch hat Prüfungscharakter.
3. Sie absolvieren Ihr Abgabegespräch gemeinsam mit anderen KollegInnen. Das Gespräch dauert ca. 60 Minuten.
4. Sie können auf die Abgabe maximal 15 Punkte erreichen. Diese setzen sich wie folgt zusammen:  
**5 Punkte** auf das Übungsblatt  
**10 Punkte** auf das Abgabegespräch
5. Die Assistenten tragen die Punkte des Abgabegesprächs in TUWEL ein und Sie sehen dort, wieviele Punkte Sie bekommen haben.
6. Falls Sie nicht zu Ihrem Gesprächstermin erscheinen, bekommen Sie automatisch 0 Punkte auf das Gespräch.

## EER

### Aufgabe 1 (EER-Diagramm erstellen) [0.5 Punkte]

Zeichnen Sie folgenden Sachverhalt in einem EER-Diagramm. Verwenden Sie dazu die (min, max)-Notation. Nullwerte sind nicht erlaubt. Manchmal kann es notwendig sein, zusätzliche künstliche Schlüssel einzuführen. Eine Unterstützung bei der Erstellung von EER-Diagrammen bietet das Tool dia (<http://wiki.gnome.org/Apps/Dia>, binaries unter <http://dia-installer.de>). Achtung: im Diagramm Editor ER auswählen!

Als Teil ihrer Strategie zur Verbesserung des Dokumentations-Workflows bietet die Kreditkartenfirma “Schuldenfroh” um ein redesign ihrer Datenbank.

Nachdem Sie sich vergewissert haben, dass die Beschreibung wenigstens ein paar Anglizismen enthält, machen Sie sich an die Arbeit.

Jede Kreditkarte ist eindeutig identifiziert durch ihre Kartennummer (KNR). Zusätzlich muss das Ablaufdatum (ABLAUFDATUM) und der Sicherheitscode (CVC) vermerkt sein. Ersetzt eine Karte eine ältere Karte, so wird gespeichert welche Karte ersetzt wird (jede Karte kann maximal eine andere Karte ersetzen und durch maximal eine andere Karte ersetzt werden), und ab wann die neue Karte gültig ist (AB).

KundInnen haben eine eindeutige Kundennummer (KNR). Zusätzlich ist ihr Vor- und Nachname (VNAME, NNAME) und ihr Geburtsdatum (GEBDAT) bekannt, wobei die Kombination aus Vorname, Nachname und Geburtsdatum ebenfalls eindeutig ist. Außerdem muss für jede Kundin/jeden Kunden eine Kontonummer (IBAN) vorhanden sein.

Bei Kreditkarten wird unterschieden auf welche Kundin/welchen Kunden die Kreditkartet lautet auf der einen, und welche KundInnen für eine Karte zeichnungsberechtigt sind auf der anderen Seite. Jede Kreditkarte lautet auf genau eine Kundin/einen Kunden. Jede Kundin/Jeder Kunde kann jedoch für beliebig viele Karten zeichnungsberechtigt sein, wobei für jede Karte die Anzahl der Zeichnungsberechtigten auf 5 begrenzt ist.

Partner der Kreditkartenfirma sind sowohl durch ihren Markennamen (MNAME) eindeutig identifiziert, als auch durch ihre Partner-ID (PID). Bei Partnern wird zwischen Vertriebspartnern und Bezahlpartnern unterschieden. Vertriebspartner vergeben Kreditkarten, wobei jede Karte von genau einem Vertriebspartner vergeben werden muss. Für Vertriebspartner wird neben dem Preis (PREIS) einer Karte auch der Prozentsatz (ANTEIL) gespeichert, den der Vertriebspartner von jedem mit der Karte getätigten Umsatz erhält. Bezahlpartner sind Unternehmen welche Kreditkarten von "Schuldenfroh" akzeptieren. Für jeden Bezahlpartner wird der Anteil des Umsatzes gespeichert der als Gebühr abgeführt werden muss (GEBUEHR). Manche Vertriebspartner gewähren einen Rabatt bei Zahlungen mit bestimmten Bezahlpartnern. Es soll vermerkt werden, welcher Vertriebspartner bei welchem Bezahlpartner welchen Rabatt (PROZENT) gewährt.

Jeder Bezahlpartner erhält mindestens einen Account. Ein Account ist eindeutig identifiziert durch den Bezahlpartner sowie durch den Benutzernamen (BENUTZER). Zusätzlich soll für jeden Account das Passwort (PW), sowie sämtliche Loginversuche gespeichert werden. Ein Login(versuch) wird eindeutig identifiziert durch den Account, sowie durch das Datum (DATUM) des Logins, gemeinsam mit einer id (ID). Darüber hinaus wird die IP-Adresse des Logins gespeichert (IP), und ob der Login erfolgreich war (SUCCESS).

MitarbeiterInnen von Bezahlpartnern werden durch ihre Sozialversicherungsnummer (SNVR) eindeutig identifiziert. Darüber hinaus wird ihr Vorname (VNAME), Nachname (NNAME), ihr Geburtsdatum (GEBDAT), sowie ihr

Gehalt (GEHALT) gespeichert. Es wird nicht gespeichert, welche MitarbeiterInnen bei welchem Partner arbeiten, sondern nur, welche MitarbeiterInnen Zugriff auf welchen Account haben, wobei es für jeden Account mindestens einen MitarbeiterIn mit Zugriff geben muss.

Eine Transaktion ist eindeutig identifiziert durch die Kreditkarte auf welche sie gebucht wurde, den Account mit dem die Buchung durchgeführt wurde, sowie durch eine Transaktionsid (TID). Zu jeder Transaktion wird außerdem der Betrag (TOTAL) sowie das Datum (DATUM) gespeichert.

Jedes Security-Level ist eindeutig bezeichnet durch seinen Namen (NAME) in Kombination mit seiner Nummer (NR). Für jede Transaktion wird genau ein Security-Level verwendet.

Die Kreditkartenfirma beschäftigt Security-Agents. Zu jedem Security-Agent wird ein Name (NAME), sein Trust-Level (TLEVEL), sowie eine eindeutige Agent-ID (AID) verwaltet.

Zu Transaktionen, bei denen der Verdacht auf einen Betrugsfall besteht wird zusätzlich noch eine Bemerkung (BEM), die Art des Betrugs (TYP), das Datum an dem der Betrugsverdacht erkannt wurde (DATUM) gespeichert, sowie welcher Security-Agent den Verdacht gemeldet hat. Sobald ein KundeIn auf einen Verdachtsfall reagiert wird das Datum (DATUM) der Reaktion gespeichert, sowie ob es sich in der Tat um einen Betrug handelt oder nicht (OK).

Es soll weiter gespeichert werden, welcher Security-Agent welcheN MitarbeiterIn zu welchem Security-Level zugelassen hat, und wann das passiert ist (AM). Es gilt zu beachten, dass jeder MitarbeiterIn zu mindestens einem Security-Level zugelassen sein muss.

**Lösung:** Siehe Abbildung 1.

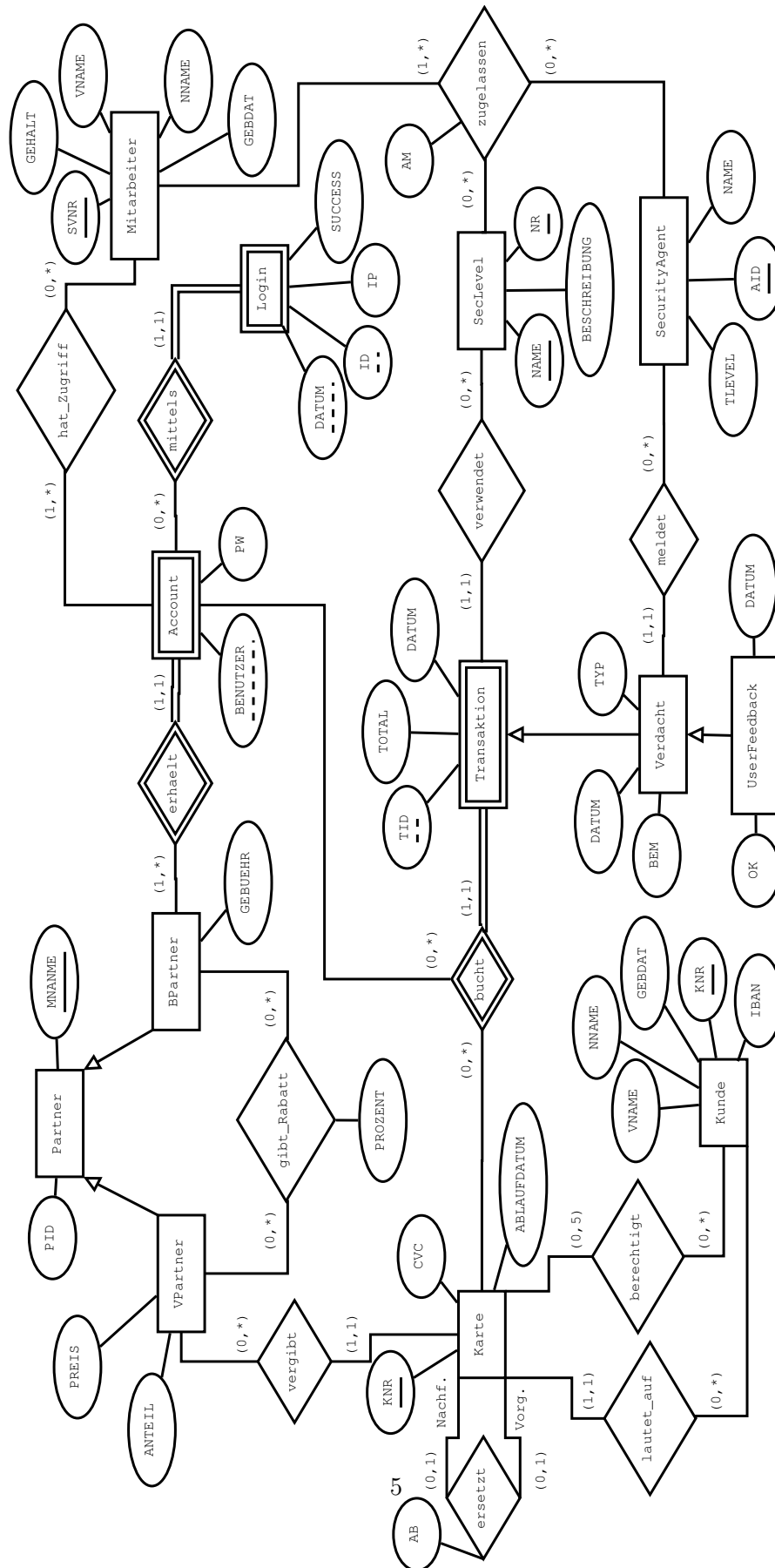


Abbildung 1: Lösung von Aufgabe 1

Anmerkungen:

- Alternative Schlüssel:
  - Kunde: VNAME + NAME + GEBDAT
  - Partner: PID

**Aufgabe 2 (EER-Beispiel selbst erstellen)** *[0.5 Punkte]*

Ihr Kollege Herr Mustermann hat Sie gebeten, ihm ein wenig Nachhilfe zu geben. Erstellen Sie für ihn ein kleines Übungsbeispiel zum Thema EER-Modellierung. Das Beispiel sollte mindestens 5 Entitytypen, 5 Beziehungstypen, eine Generalisierung und einen schwachen Entitytyp enthalten. Vielleicht bringen Sie in dem Beispiel auch noch eine rekursive oder eine ternäre Beziehung unter. Sie erhalten die Punkte für die textuell ausformulierte Angabe sowie für die Musterlösung. (Trauen Sie sich durchaus kreativ bei der Wahl Ihres Beispiels zu sein.)

**Aufgabe 3 (Überführung ins Relationenschema)** *[0.5 Punkte]*

Überführen Sie das EER-Diagramm aus Abbildung 2 in ein Relationenschema. Nullwerte sind nicht erlaubt. Verwenden Sie möglichst wenig Relationen. Unterstreichen Sie sämtliche Primärschlüssel, schreiben Sie die Fremdschlüssel kursiv und stellen Sie sicher, dass ein Fremdschlüssel eindeutig der passenden Relation zugeordnet werden kann.

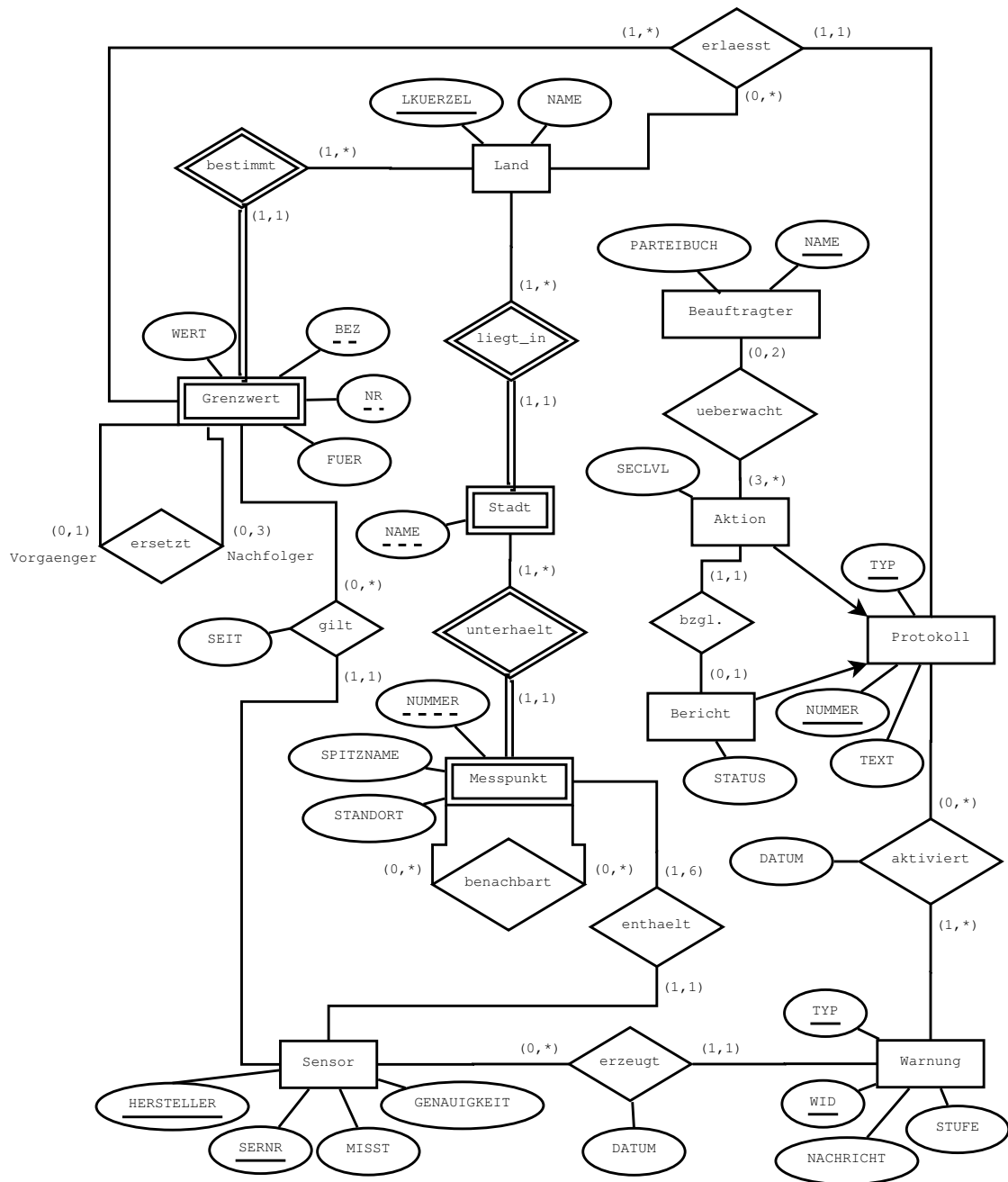


Abbildung 2: EER-Diagramm zu Aufgabe 3

### Lösung:

Land (LKUERZEL, NAME)

Grenzwert (Land.LKUERZEL, BEZ, NR, FUER, WERT)

ersetzt (Vorgaenger:LKUERZEL, Vorgaenger:BEZ, Vorgaenger:NR,

	<i>Nachfolger:LKUERZEL, Nachfolger:BEZ, Nachfolger:NR)</i>
Stadt	( <u>Land.LKUERZEL</u> ,NAME)
Messpunkt	( <u>Stadt.LKUERZEL, Stadt.NAME, NUMMER, SPITZNAME, STANDORT</u> )
benachbart	( <u>Messpunkt1.LKUERZEL, Messpunkt1.NAME, Messpunkt1.NUMMER,</u> <u>Messpunkt2.LKUERZEL, Messpunkt2.NAME, Messpunkt2.NUMMER</u> )
Sensor	( <u>HERSTELLER, SERNR, MISST, GENAUGKEIT,</u> <u>Messpunkt.LKUERZEL, Messpunkt.NAME, Messpunkt.NUMMER,</u> <u>Grenzwert.LKUERZEL, Grenzwert.BEZ, Grenzwert.NR, SEIT</u> )
Warnung	( <u>TYP, WID, STUFE, NACHRICHT, Sensor.HERSTELLER, Sensor.SERNR</u> DATUM)
Protokoll	(TYP, NUMMER, TEXT, <u>Land.LKUERZEL,</u> <u>Grenzwert.LKUERZEL, Grenzwert.BEZ, Grenzwert.NR</u> )
aktiviert	( <u>Warnung.TYP, Warnung.WID, Protokoll.TYP, Protokoll.NUMMER,</u> DATUM)
Aktion	( <u>Protokoll.TYP, Protokoll.NUMMER, SECLVL</u> <u>Bericht.TYP, Bericht.NUMMER</u> )
Bericht	( <u>Protokoll.TYP, Protokoll.NUMMER, STATUS</u> )
Beauftragter	( <u>NAME, PARTEIBUCH</u> )
ueberwacht	( <u>Beauftragter.NAME, Aktion.TYP, Aktion.NUMMER</u> )

#### Aufgabe 4 (Semantik von EER Diagrammen) [0.6 Punkte]

Ein Kollege von Ihnen betreibt zur Finanzierung seines Studiums ein Restaurant. Obwohl er dies - um den zeitlichen Aufwand einzugrenzen - als Franchisenehmer tut, ist sich der Besuch der LVA "Datenmodellierung" bislang nicht ausgegangen. Als der Franchise-Geber nun neue Richtlinien zur Menüerstellung erlässt, muss sich Ihr Kollege an Sie um Hilfe wenden: Als Beschreibung der neuen Regeln bezüglich Mittags- und Abendmenüs wurde Ihm ausschließlich das in Abbildung 3 dargestellte EER-Diagramm übermittelt, welches er leider nicht lesen kann.



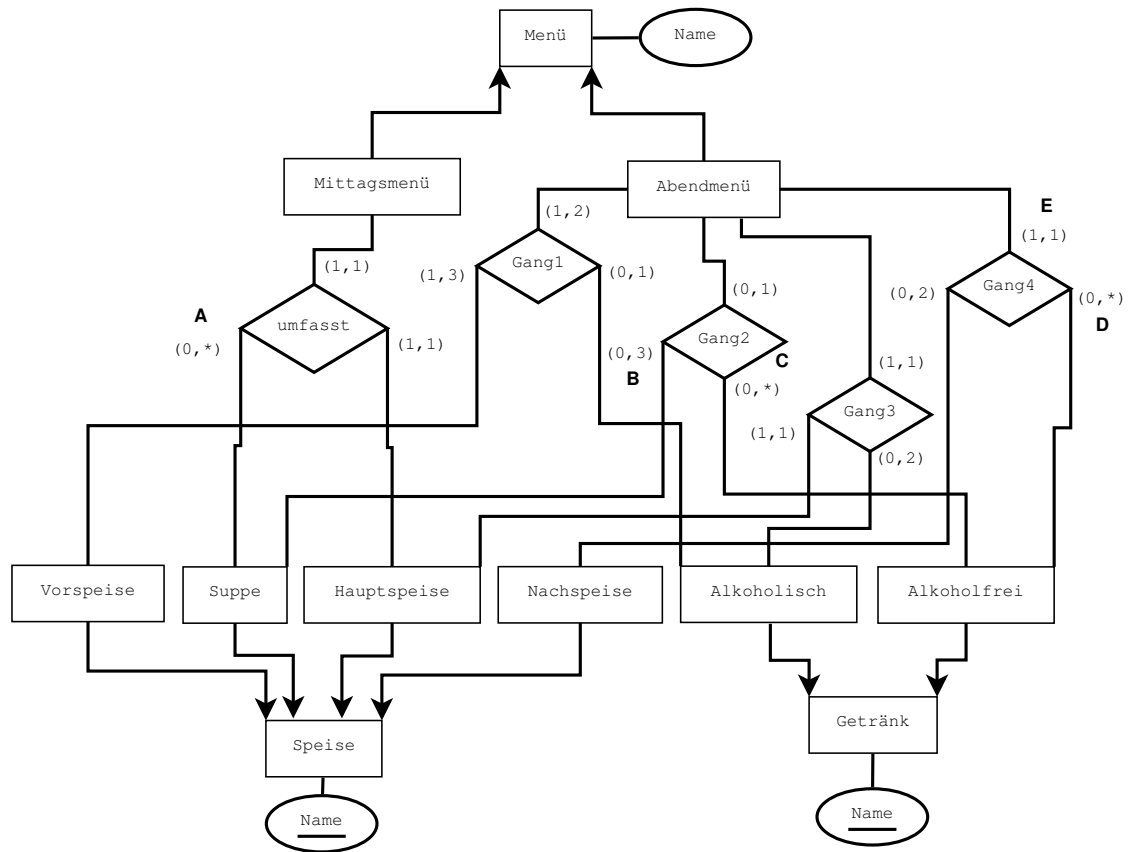


Abbildung 3: EER-Diagramm zu Aufgabe 4

1. Erklären Sie Ihrem Kollegen in natürlicher Sprache woraus sich ein *Mittagsmenü* zusammensetzt, und ob es im Bezug auf die Speisen Einschränkungen gibt.

**Lösung:**

Jedes Mittagsmenü besteht aus genau einer Suppe und einer Hauptspeise. Jede Suppe darf in beliebig vielen Mittagsmenüs vorkommen. Für jede Hauptspeise muss es genau ein Mittagsmenü geben.

2. Ihr Kollege zeigt Ihnen seine Speisekarte mit den bisherigen Mittags- und Abendmenüs (siehe unten). Entspricht diese Speisekarte den neuen Bestimmungen, oder gibt es Verletzungen der Vorschriften? Beschreiben Sie mindestens 8 Verletzungen des EER-Diagramms durch die aktuelle Speisekarte.

Vorspeisen ----- Antipasti-Teller Lachsbrötchen Rotkraut-Vogerlsalat mit Maroni, Speck und Orangen Feta mit Tomate in Knoblauch Spargelragout Gemischter Salat	Suppen ----- Leberknödelsuppe Kürbiscremesuppe Miso-Suppe Spargelcremesuppe Gemüsesuppe
Hauptspeisen ----- Wiener Schnitzel mit Pommes Frites Spaghetti Bolognese Pizza Hawaii Schweinsbraten mit Knödel 8 Schätze Gagh Lachs vom Grill Burger	Nachspeisen ----- Schokoladeneis Tiramisu Topfenknödel mit Preiselbeerschaum Marillen-Topfen-Creme
Alkoholfreie Getränke ----- Mineralwasser Apfelsaft Cola Himbeerkracherl Pflaumensaft Mango Lassi alkoholfreies Bier Raktajino Espresso Earl Grey	Alkoholische Getränke ----- Bier Weißwein Rotwein Radler Uzo Likör Wodka Martini (gerührt) Romulanisches Ale Acamarianischer Brandy
Mittagsmenüs ----- M1: Miso-Suppe, 8 Schätze M2: Leberknödelsuppe, Lachs vom Grill M3: Leberknödelsuppe, Kürbiscremesuppe, Schweinsbraten mit Knödel M4: Wiener Schnitzel mit Pommes Frites	

M5: Gemüsesuppe, Pizza Hawaii  
M6: Spargelcremesuppe, Spagehetti Bolognese  
M7: Gemüsesuppe, Burger  
M8: Kürbiscremesuppe, Gagh

#### Abendmenüs

-----

##### A1:

1. Gang: Rotkraut-Vogerlsalat mit Maroni, Speck und Orangen; Likör
2. Gang: Miso-Suppe; Mango Lassi
3. Gang: Schweinsbraten mit Knödel; Bier
4. Gang: Schokoladeneis; Raktajino

##### A2:

1. Gang: Feta mit Tomate in Knoblauch; Wodka Martini (gerührt)
1. Gang: Lachsbrötchen; Romulanisches Ale
2. Gang: Spargelcremesuppe
3. Gang: Wiener Schnitzel mit Pommes Frites; Radler
4. Gang: Marillen-Topfen-Creme; Himbeerkracherl

##### A3:

1. Gang: Antipasti-Teller; Weißwein
3. Gang: Lachs vom Grill; Romulanisches Ale
4. Gang: Tiramisu; Espresso

##### A4:

1. Gang: Feta mit Tomate in Knoblauch; Uzo
2. Gang: Leberknödelsuppe; Mineralwasser
4. Gang: Topfenknödel mit Preiselbeerschaum; Espresso

##### A5:

1. Gang: Lachsbrötchen; Wodka Martini (gerührt)
1. Gang: Gemischter Salat; Acamarianischer Brandy
2. Gang: Miso-Suppe; Himbeerkracherl
3. Gang: Pizza Hawaii; Rotwein

##### A6:

1. Gang: Lachsbrötchen; Weißwein
3. Gang: Gagh; Romulanisches Ale
4. Gang: Schokoladeneis; Cola

##### A7:

- |   |
|---|
| 1. Gang: Lachsbrötchen; Wodka Martini (gerührt)<br>1. Gang: Antipasti-Teller; Weißwein<br>3. Gang: Burger; Bier<br>3. Gang: Spagehetti Bolognese; Rotwein<br>4. Gang: Marillen-Topfen-Creme; Earl Grey<br><br>A8:<br>1. Gang: Feta mit Tomate in Knoblauch; Acamarianischer Brandy<br>2. Gang: Miso-Suppe; Earl Grey<br>3. Gang: Gagh; Rotwein<br>4. Gang: Schokoladeneis; Pflaumensaft |
|---|

### Lösung:

Folgende Verletzungen finden sich in der Speisekarte:

- Das Mittagsmenü **M3** enthält eine Suppe zu viel.
- Das Mittagsmenü **M4** enthält keine Suppe (= 1 Suppe zu wenig).
- Im Abendmenü **A2** fehlt das Getränk zum 2. Gang.
- Das Abendmenü **A4** enthält keinen 3. Gang (Gang3)
- Das Abendmenü **A5** enthält keinen 4. Gang (Gang4)
- Das Abendmenü **A7** enthält zwei 3. Gänge (Gang3), das ist einer zu viel.
- Die Vorspeise **Lachsbrötchen** steht mit 4 verschiedenen Paaren aus Abendmenü und Alkoholischem Getränk in einer "Gang 1" Beziehung (kommt in 4 verschiedenen 1. Gängen vor). Es sind jedoch maximal 3 erlaubt.
- Die Vorspeise **Spargelragout** kommt in keiner "Gang 1" Beziehung vor (ist Teil keines 1. Ganges). Jede Vorspeise muss jedoch in mindestens einer Beziehung vorkommen.
- Die Hauptspeise **8 Schätze** kommt in keiner "Gang 3" Beziehung vor (ist niemals Hauptgang eines Menüs). Jede Hauptspeise muss jedoch in mindestens einer solchen Beziehung vorkommen.
- Die Hauptspeise **Gagh** steht mit 2 verschiedenen Paaren aus Abendmenü und Alkoholischem Getränk in einer "Gang 3" Beziehung (ist Teil des Hauptganges in 2 verschiedenen Menüs). Jede Hauptspeise darf jedoch in maximal einer solchen Beziehung stehen.
- Die Nachspeise **Schokoladeneis** steht mit 3 verschiedenen Paaren aus Abendmenü und Alkoholfreiem Getränk in einer "Gang 4" Beziehung (ist Teil des letzten Ganges von 3 Abendmenüs). Jede Nachspeise darf jedoch in maximal zwei solchen Beziehungen stehen.
- Das alkoholische Getränk **Rotwein** steht mit 3 verschiedenen Paaren aus Abendmenü und Hauptspeise in einer "Gang 3" Beziehung (ist Teil des Hauptganges

von 3 Abendmenüs). Jedes alkoholische Getränk darf jedoch in maximal zwei solchen Beziehung stehen.

- Das alkoholische Getränk **Wodka Martini (gerührt)** steht mit 2 verschiedenen Paaren aus Abendmenü und Vorspeise in einer “Gang 1” Beziehung (ist Teil der Vorspeise von 2 Abendmenüs). Jedes alkoholische Getränk darf jedoch in maximal einer solchen Beziehung stehen.
- Das alkoholische Getränk **Weißwein** steht mit 2 verschiedenen Paaren aus Abendmenü und Vorspeise in einer “Gang 1” Beziehung (ist Teil der Vorspeise von 2 Abendmenüs). Jedes alkoholische Getränk darf jedoch in maximal einer solchen Beziehung stehen.

3. Ihr Kollege zeigt auf die im EER-Diagramm mit  $(A)$ ,  $(B)$ ,  $(C)$ ,  $(D)$  markierten Kardinalitäten und fragt, was die genauesten Werte in der  $(\min, \max)$  Notation wären, welche von seiner aktuellen Speisekarte erfüllt werden (beziehen Sie sämtliche auf der Speisekarte erwähnten Beziehungen in die Antwort mit ein, auch wenn es sich dabei um ungültige (z.B. unvollständige; laut EER-Diagramm) Beziehungs-Instanzen handelt).

(Beispiel: für  $(E)$  wäre die Antwort  $(0, 1)$ , da alle Abendmenü-Entitäten genau in einer “Gang 4”-Beziehung vorkommen, bis auf **A5**, welches in keiner solchen Beziehung steht. Daher: Minimum = 0 und Maximum = 1.)

#### **Lösung:**

Die Speisekarte erfüllt folgende Werte:

- $(A)$ :  $(1, 2)$
- $(B)$ :  $(0, 3)$
- $(C)$ :  $(0, 1)$
- $(D)$ :  $(0, 2)$

## **Relationale Algebra - Relationenkalkül**

Um Ihnen die Erstellung Ihrer Abgabe zu den folgenden Aufgaben zu erleichtern, haben wir unter <http://dbai.tuwien.ac.at/education/dm/resources/symbols.html> eine Liste mit den wichtigsten Symbolen der relationalen Algebra zusammengestellt. Sie können diese per copy/paste in Ihr Word/LibreOffice/OpenOffice/... Dokument einfügen. Zusätzlich sind die entsprechenden L<sup>A</sup>T<sub>E</sub>X Befehle vermerkt.

### **Aufgabe 5 (Primitive Operatoren) [0.5 Punkte]**

- (a) Der Anti-Semi-Join  $R \triangleright S$  liefert alle Tupel  $r$  aus  $R$ , für welche es kein Tupel  $s$  in  $S$  gibt, so dass  $r$  und  $s$  die selben Werte auf den gemeinsamen Attributen von  $R$  und  $S$  haben.

Nehmen Sie an, dass zwei Relationen  $R$  und  $S$  gegeben sind. Hierbei umfasst das Schema von  $R$   $g + r$  viele Attribute  $(R_1, \dots, R_r, G_1, \dots, G_g)$  und das Schema von  $S$   $g + s$  viele Attribute  $(S_1, \dots, S_s, G_1, \dots, G_g)$ .

Drücken Sie  $R \triangleright S$  mittels des Linken Semi-Joins ( $\ltimes$ ) und der primitiven Operatoren  $(\sigma, \pi, \cup, -, \times, \rho)$  aus.

- (b) Die Operatoren  $\sigma, \pi, \cup, -, \times, \rho$  werden *primitive Operatoren* genannt, weil sich durch sie, gemeinsam mit den Basisausdrücken (Relationen der Datenbank und konstante Relationen), alle anderen Operatoren der relationalen Algebra ausdrücken lassen. Es ist dies jedoch nicht die einzige (minimale) Menge an Operatoren, aus welcher sich alle anderen Operatoren ableiten lassen. In der Vorlesung wurden unter anderem die folgenden Operatoren besprochen: Natürlicher Join ( $\bowtie$ ), Allgemeiner Join ( $\bowtie_\theta$ ), (linker, rechter, voller) äußere Join ( $\ltimes, \ltimes, \ltimes$ ), (linker, rechter) Semi-Join ( $\ltimes, \ltimes$ ), Mengendurchschnitt ( $\cap$ ), Division ( $\div$ ).

Betrachten Sie nun die Operatoren  $\pi, \cup, -, \rho$ . Welche/Welchen oben erwähnten Operator (außer  $\sigma$  und  $\times$ ) müsste man zu diesen vier Operatoren dazugeben, um wiederum eine Menge primitiver Operatoren zu erhalten (also eine Menge an Operatoren mit denen alle anderen Operatoren ausgedrückt werden können)? Zeigen Sie, dass es sich wirklich um eine Menge primitiver Operatoren handelt, indem Sie die Operatoren  $\sigma$  und  $\times$  mit Hilfer der von Ihnen gewählten Operatoren sowie  $\pi, \cup, -, \rho$  ausdrücken (verwenden Sie dazu wiederum das in a) beschriebene Schema, d.h., stellen Sie die Ausdrücke  $\sigma_\theta(S)$  und  $S \times T$  dar).

- (c) Beschreiben Sie weiters einen neuen, zusammengesetzten relationalen Operator in natürlicher Sprache und veranschaulichen Sie dessen Anwendung an einem Beispiel. Drücken Sie diesen Operator anschließend mit Hilfe der primitiven Operatoren aus. Der neue Operator soll aus zumindest drei primitiven Operatoren zusammengesetzt werden.

### Lösung:

- (a)  $R \triangleright S \equiv R - (R \ltimes S)$
- (b) Der Allgemeine Join  $\bowtie_\theta$  ist ausreichend, damit es sich wieder um eine Menge primitiver Operatoren handelt.

- $\sigma_\theta(S) \equiv \pi_{S.S_1, \dots, S.S_s, S.G_1, \dots, S.G_g}(S \bowtie_\theta \rho_R(S))$

Anmerkung: In  $\theta$  wäre allen Attributen noch das  $S$  voranzustellen ist. Um das  $\theta$  unverändert zu lassen könnte man auch die Attribute in einer der beiden Relationen umbenennen.

Nimmt man, wie in der Vorlesung angedeutet, an dass man sich nicht um die Attributnamen kümmern muss, weil beim Kreuzprodukt (welches dem Allgemeinen Join zugrunde liegt) Namenskonflikte bei gleichnamigen Attributen durch das Voranstellen des Relationennamens gelöst werden, könnte man auch

$\pi_{S1.S_1, \dots, S1.S_s, S1.G_1, \dots, S1.G_g}(S \bowtie_{\theta} S)$  schreiben. In diesem Fall wäre jedoch wiederum  $\theta$  entsprechend anzupassen.

- $S \times T \equiv S \bowtie_{1=1} \rho_{U_1 \leftarrow G_1, \dots, U_g \leftarrow G_g}(T)$

### Aufgabe 6 (Äquivalenzen) [0.5 Punkte]

Gegeben sind folgende relationale Ausdrücke über  $R(\underline{ABC})$ ,  $S(\underline{DE})$ ,  $T(\underline{BD})$  und  $U(\underline{AB})$ . Überprüfen Sie, ob die folgenden Ausdrücke in relationaler Algebra äquivalent sind. Falls ja, so erklären Sie warum. Falls nein, geben Sie eine Begründung (und wo möglich ein Gegenbeispiel) an.

- (a)  $r_1: \pi_D(R \bowtie S) \cup \pi_D(S)$  und  $r_2: \pi_D(S)$
- (b)  $r_3: \pi_A(R) \bowtie U$  und  $r_4: \pi_A(R \bowtie U)$
- (c)  $r_5: \sigma_{A=1}(\sigma_{D=4}(R \bowtie T))$  und  $r_6: (\sigma_{D=4}(T) \bowtie R) \cap (\sigma_{A=1}(R) \bowtie T)$
- (d)  $r_7: \rho_{A \leftarrow C}(R - U)$  und  $r_8: \pi_A(U)$
- (e)  $r_9: \sigma_{A>5}(\pi_A(R)) \cap \sigma_{A<5}(\pi_A(U))$  und  $r_{10}: \sigma_{A>5}(\pi_A(R)) - \sigma_{A>4}(\pi_A(R))$

### Lösung:

- (a) **Ja:**  $R \bowtie S$  wird zum kartesischen Produkt, da  $R$  und  $S$  keine Attribute gemeinsam haben. Die Projektion auf  $D$  liefert alle  $D$ -Tupel aus  $S$ . Die Vereinigung mit  $\pi_D(S)$  ändert an der Ergebnismenge nichts.
- (b) **Nein:** Die gemeinsamen Attribute von  $R$  und  $U$  sind  $A$  und  $B$ . Wenn nun ein Tupel in  $R$  mit einem Tupel in  $U$  auf dem Wert von  $A$  aber nicht  $B$  übereinstimmt, dann stimmen diese beiden Tupel in  $r_4$  nicht auf allen gemeinsamen Attributen überein. In  $r_3$  wird das Tupel aus  $R$  nur auf das Attribut  $A$  projiziert. Das dadurch entstehende Tupel ist nun mit dem Tupel aus  $B$  kompatibel. Ein konkretes Gegenbeispiel ist die Ausprägung  $\{(1, 2, 3)\}$  für  $R$  und  $\{(1, 3)\}$  für  $U$ . In  $r_4$  können diese beiden Tupel keinen Join eingehen, das Ergebnis von  $r_4$  ist also die leere Menge  $\emptyset$ . In  $r_3$  wird jedoch  $(1, 2, 3)$  zuerst auf das Tupel  $(1)$  projiziert. Dieses kann nun den Join mit  $(1, 3)$  eingehen, und das Ergebnis von  $r_3$  ist daher  $\{(1)\}$ .
- (c) **Ja:** Hintereinanderausführung der Selektion kann durch Mengendurchschnitt ausgedrückt werden. Dabei ist es in diesem Fall egal, ob die Selektion vor oder nach dem Join verwendet wird.
- (d) **Nein:** Dem Ausdruck  $r_7$  kann keine Ergebnismenge zugeordnet werden, da  $R$  und  $U$  nicht das gleiche Schema haben.
- (e) **Ja:** Beide Ausdrücke liefern die leere Menge. Bei  $r_8$  wird dies durch den Durchschnitt zweier disjunkter Mengen erzielt. Bei  $r_{10}$  durch Mengendifferenz einer Menge und einer Übermenge.

### Aufgabe 7 (Größenabschätzung) [0.5 Punkte]

Gegeben sind die Relationen  $R(\underline{A}BCD)$  mit  $x$  Tupeln,  $S(\underline{B}DE)$  mit  $y$  Tupeln und  $T(\underline{A}CD)$  mit  $z$  Tupeln. Geben Sie die minimale bzw. maximale Größe (= Anzahl der Tupel) folgender Ausdrücke in relationaler Algebra an und begründen Sie Ihre Antwort.

- (a)  $r_1 : \pi_A(\sigma_{A=1 \vee A=4}(R \bowtie T))$  mit  $x = 5$  und  $z = 9$  **Lösung:** (min 0; max 2)
- (b)  $r_2 : \pi_{AB}(R) \cup \pi_{AB}(\rho_{A \leftarrow D}(S))$  mit  $x = 7$  und  $y = 4$  **Lösung:** (min 7; max 11)
- (c)  $r_3 : (S \bowtie T) \bowtie \pi_A(R)$  mit  $x = 6$ ,  $y = 8$  und  $z = 12$  **Lösung:** (min 0; max 48)
- (d)  $r_4 : (\pi_B(S) \cup \pi_B(R)) \bowtie \rho_{B \leftarrow A}(T)$  mit  $x > y > z$  **Lösung:** (min 0; max  $z$ )

#### Lösung:

Aufgabe (a): Falls kein Tupel die Selektionsbedingung erfüllt, ergibt sich 0 als untere Schranke.  $R \bowtie T$  liefert maximal 5 Tupel, von denen maximal 2 selektiert werden können. Die Projektion liefert dann genau 2 Tupel.

Aufgabe (b):  $\pi_{AB}(R)$  hat genau 7 Tupel,  $\pi_{AB}(\rho_{A \leftarrow D}(S))$  hat genau 4 Tupel, da auf die Schlüssel projiziert wird. Die Vereinigung hat daher maximal 11 und minimal 7 Tupel.

Aufgabe (c):  $S \bowtie T$  hat minimal 0 Tupel. Damit hat auch  $r_3$  minimal 0 Tupel. Maximal hat  $S \bowtie T$   $8 \times 12 = 96$  Tupel (wenn alle Tupel auf  $D$  gleich sind). Diese 96 Tupel enthalten 12 verschiedene Attributwerte für  $A$  ( $A$  ist ein Schlüssel in  $T$ ). Jedoch besitzt  $R$  maximal 6 verschiedene Werte für  $A$ . D.h. bei der Berechnung des linken Semi-Joins können nur jene Tupel übrig bleiben, welche einen dieser 6 Werte enthalten. Anstatt  $8 \times 12$  erhalten wir daher nur  $8 \times 6 = 48$  Tupel.

Aufgabe (d): Die Projektion auf auf  $R$ , bez.  $S$ , liefert minimal jeweils 1 Tupel und maximal  $x$ , bez.  $y$ , Tupel. Der Join mit  $T$  (wobei Attribut  $A$  auf  $B$  unbenannt wurde) liefert dann minimal 0 Tupel. Da die maximal  $x + y$  Tupel aus der Vereinigung von  $\pi_B(S) \cup \pi_B(R)$  sicher größer sind als  $z$ , liefert der Join mit  $T$  maximal  $z$  Tupel.

### Aufgabe 8 (Formalisieren von Anfragen) [0.5 Punkte]

Gegeben ist eine Datenbank über Pokémon und Arenen in Form des folgenden Relationenschemas.

- P = Pokémon ( PID, Bezeichnung, Wettkampfpunkte, Typ )
- A = Arena ( Name, Ort, Level, *Pokémon.PID* )
- Z = zugeordnet ( *Pokémon.PID*, *Arena.Name* )

Die in Relation A gespeicherte PID bezeichnet den Leiter der Arena. Drücken Sie folgende Anfragen in relationaler Algebra aus.



- (a) Gesucht sind die Bezeichnungen aller Arenaleiter, zu deren Arenen ein Pokémon mit über 3000 Wettkampfpunkten zugeordnet ist.
- (b) Gesucht sind Bezeichnung und Typ aller Pokémon, die weniger als 2000 Wettkampfpunkte haben und keiner Arena zugeordnet sind, deren Level kleiner als 5 ist oder der ein Pokémon vom Typ “Wasser” zugeordnet ist.
- (c) Betrachten Sie nun folgenden Ausdruck in relationaler Algebra. Beschreiben Sie in eigenen Worten, welches Ergebnis dieser Ausdruck liefert.

$$\pi_{\text{Ort}}(A \bowtie \pi_{\text{Name}}(\pi_{\text{PID,Name}}(Z) \div \pi_{\text{PID}}(\sigma_{\text{Typ}=\text{'Feuer'} \wedge \text{Wettkampfpunkte} > 3200}(P))))$$

**Lösung:**

- (a)  $\pi_{\text{Bezeichnung}}(P \bowtie (A \bowtie \pi_{\text{Name}}(Z \bowtie \sigma_{\text{Wettkampfpunkte} > 3000}(P))))$
- (b)  $\pi_{\text{Bezeichnung, Typ}}(\sigma_{\text{Wettkampfpunkte} < 2000}(P) \bowtie (\pi_{\text{PID}}(P) - (\pi_{\text{PID}}(Z \bowtie (\pi_{\text{Name}}(\sigma_{\text{level} < 5}(A)) \cup \pi_{\text{Name}}(Z \bowtie \sigma_{\text{Typ}=\text{'Wasser'}}(P)))))))$

**Erklärung:**

Es wird nach Pokémon gesucht, welche zwei Eigenschaften erfüllen: Zum Einen sollen sie weniger als 2000 Wettkampfpunkte haben, und zum Anderen sollen sie nicht gewissen Arenen zugeordnet sein. Während wir die erste Eigenschaft mittels einer einfachen Selektion überprüfen können (mittels  $\sigma_{\text{Wettkampfpunkte} < 2000}(P)$ ), ist dies im zweiten Fall nicht so einfach möglich: Wir suchen alle Pokémon, welche *nicht* einer bestimmten Menge an Arenen zugeordnet sind. Anders also zuvor, wo wir nur auf Eigenschaften eines einzelnen Tupels gesehen haben (im konkreten Fall auf den Wert des Attributs “Wettkampfpunkte”), erfordert dies nun, dass wir Mengen von Tupeln miteinander vergleichen (die Menge der Pokémon mit jenen der Arenen).

In der relationalen Algebra haben wir jedoch keine direkte Möglichkeit für einzelne Tupel (in unserem Fall jeden Eintrag in einer Ausprägung von  $P$ ) komplizierte Eigenschaften zu überprüfen, welche Informationen von anderen Tabellen benötigen (die Selektion erlaubt nur Tests auf Attributwerte einer Relation).

Eine typische Vorgehensweise um dieses Problem mit den Operatoren der relationalen Algebra (welche jeweils auf Mengen von Tupel arbeiten) zu lösen ist wie folgt: In einem ersten Schritt suchen wir zuerst einmal nach all jenen Pokémons, welche die gesuchte Eigenschaft *nicht* erfüllen. Nennen wir diese Anfrage *BadPoks*. Im zweiten Schritt ziehen wir diese dann (mittels des Differenzoperators  $-$ ) von der Menge aller Pokémon ab. Dadurch erhalten wir die gesuchten Pokémon. Benennen wir die Abfrage welche uns alle Pokémon liefert mit *AllPoks*, so erhalten wir das Ergebnis *GoodPoks* als  $\text{GoodPoks} = \text{AllPoks} - \text{BadPoks}$ .

Wie erhalten wir nun die Menge *BadPoks* der “schlechten” Pokémon? Dies ist die Menge all jener Pokémon, welche einer Arena zugeordnet sind die entweder ein Level kleiner als fünf hat, oder der ein “Wasser”-Pokémon zugeordnet ist. Einmal angenommen wir hätten eine Menge *BadArenas*, welche genau diese “schlechten” Arenen

enthält wäre das Problem leicht zu lösen:  $BadPoks = \pi_{PID}(Z \bowtie BadArenas)$ . Betrachten wir zuerst einmal nur  $Z \bowtie BadArenas$ : Die Tabelle  $Z$  enthält die Informationen welche Pokémon welcher Arena zugeordnet sind. Wir wollen aber nur jene Pokémon, welcher einer "schlechten" Arena zugeordnet sind. D.h. wir wollen nur jene Einträge in  $Z$  erhalten, bei denen der zweite Wert in  $BadArenas$  aufscheint. Dies ist genau was der Linke Semijoin  $\bowtie$  tut. Das Ergebnis von  $Z \bowtie BadArenas$  sind Paare  $(PID, Name)$  von Pokémon und Arenen. Wir sind jedoch nur an den Pokémon interessiert, weshalb wir mit  $\pi_{PID}$  auf diese projizieren.

Es bleibt also nur noch, die Menge  $BadArenas$  zu berechnen. Die Arenen mit einem Level kleiner als fünf zu erhalten ist wiederum leicht:  $LowLevelAr = \sigma_{level < 5}(A)$ . Um jene Arenen zu erhalten, welchen ein Wasserpokémon zugeordnet ist ( $WaPokAr$ ), benötigen wir zum Einen eine Liste aller Wasserpokémon ( $WaPok = \sigma_{Typ='Wasser'}(P)$ ), und müssen uns zum Anderen aus der Menge an Zuordnungen jene Paare heraussuchen, in denen ein Wasserpokémon vorkommt ( $Z \bowtie WaPok$ ). Von diesen Paaren sind wir jeweils nur an der Information über die Arena interessiert (wir projizieren mittel  $\pi_{Name}$  auf die Arena), und erhalten so  $WaPokAr = \pi_{Name}(Z \bowtie WaPok)$ .

Die gesuchte Menge  $BadArenas$  enthält nun alle Arenen welche entweder in  $LowLevelAr$  oder in  $WaPokAr$  vorkommen. D.h. wir erhalten  $BadArenas = \pi_{Name}(LowLevelAr) \cup WaPokAr$ . Nachdem die Vereinigung voraussetzt, dass beide Relationen das selbe Schema haben, muss  $LowLevelAr$  noch auf den Namen projiziert werden.

Damit erhalten wir nun die Menge  $BadPoks$ . Um diese von  $AllPoks = P$  abziehen zu können, müssen beide Relationen das selbe Schema haben ( $BadPoks$  enthält nur das Attribut  $PID$ ), d.h. erhalten  $GoodPoks = \pi_{PID}(P) - BadPoks$ .

D.h. wir können nun die Relation  $\sigma_{Wettkampfpunkte < 2000}(P)$  betrachten, welche alle Pokémon mit weniger als 2000 Wettkampfpunkten enthält. Von diesen Pokémon wollen wir nur jene erhalten, welche auch in der Relation  $GoodPoks$  (mit dem Schema  $(PID)$ ) vorkommen. Eine Möglichkeit dies zu tun ist wiederum mittels dem Semijoin  $\sigma_{Wettkampfpunkte < 2000}(P) \bowtie GoodPoks$ . Nachdem wir nur an der Bezeichnung und dem Typ der Pokémon interessiert sind projizieren wir das Ergebnis noch entsprechend, und erhalten somit die oben angeführte Abfrage.

- (c) Gesucht ist der Ort der Arena, der alle Pokémon vom Typ "Feuer" mit über 3200 Wettkampfpunkten zugeordnet sind.

### Aufgabe 9 (Relationenkalkül) [0.5 Punkte]

Übersetzen Sie die Anfragen, die Sie in Aufgabe 8 (a) und Aufgabe 8 (b) in relationaler Algebra formuliert haben, in sichere Ausdrücke des Tupel- und Domänenkalküls. Geben Sie bei jeder der Anfragen an, in welchem Kalkül sie formuliert ist.

**Lösung:**

(a) Tupelkalkül:  $\left\{ [p.\text{Bezeichnung}] \mid p \in P \wedge \exists a \in A (p.\text{PID} = a.\text{PID} \wedge \exists z \in Z (z.\text{Name} = a.\text{Name} \wedge \exists p' \in P (p'.\text{PID} = z.\text{PID} \wedge p'.\text{Wettkampfpunkte} > 3000))) \right\}$

Domänenkalkül:  $\left\{ [b] \mid \exists p, w, t ([p, b, w, t] \in P \wedge \exists n, o, l ([n, o, l, p] \in A \wedge \exists p' ([p', n] \in Z \wedge \exists b', w', t' ([p', b', w', t'] \in P \wedge w' > 3000))) \right\}$

(b) Tupelkalkül:  $\left\{ [p.\text{Bezeichnung}, p.\text{Typ}] \mid p \in P \wedge p.\text{Wettkampfpunkte} < 2000 \wedge \neg \exists z \in Z (z.\text{PID} = p.\text{PID} \wedge \exists a \in A (a.\text{Name} = z.\text{Name} \wedge ((a.\text{Level} < 5) \vee \exists z' \in Z (z'.\text{Name} = z.\text{Name} \wedge \exists p' \in P (p'.\text{PID} = z.\text{PID} \wedge p'.\text{Typ} = \text{'Wasser'})))))) \right\}$

Domänenkalkül:  $\left\{ [b, t] \mid \exists p, w ([p, b, w, t] \in P \wedge w < 2000 \wedge \neg \exists n ([p, n] \in Z \wedge \exists o, l, p' ([n, o, l, p'] \in A \wedge ((l < 5) \vee \exists p'' ([p'', n] \in Z \wedge \exists b'', w'', t'' ([p'', b'', w'', t''] \in P \wedge t'' = \text{'Wasser'})))))) \right\}$

## Gesamter Stoff

### Aufgabe 10 (Vermischtes) [0.4 Punkte]

Ihr Kollege Karl Konfus erklärt einige wichtige Zusammenhänge, die er sich für die Prüfung merken will. Leider hat er manche Dinge falsch verstanden, und ist sich bei anderen Dingen selber nicht ganz sicher. Finden Sie die Fehler in seinen Aussagen, begründen Sie warum die Erklärungen richtig oder falsch sind, und stellen Sie die falschen Aussagen richtig. Beantworten Sie außerdem seine Fragen und erklären beziehungsweise begründen Sie Ihre Antworten kurz.

- (a) Wenn zwei Relationenschemata  $R(\alpha)$  und  $S(\beta)$  keine gemeinsamen Attribute besitzen (d.h. kein Attribut kommt sowohl in  $\alpha$  als auch in  $\beta$  vor), dann ergibt der natürliche Verbund ( $R \bowtie S$ ) folgendes Ergebnis: Die Attributmenge des Ergebnisschemas ist leer, und die dazugehörige Ausprägung enthält genau das leere Tupel “()”.
- (b) Angenommen in einem Relationenschema  $R(ABCDE)$  reichen die folgenden Kombinationen von Attributen aus, um einen Eintrag eindeutig identifizieren zu können:  $CE$ ,  $ABC$  und  $D$ . Dann können sowohl  $S_1 = D$ ,  $S_2 = CE$  und  $S_3 = ABC$  als Schlüssel gewählt werden, obwohl  $S_1$  weniger Attribute enthält als  $S_2$ , was wiederum weniger Attribute enthält als  $S_3$  — und das obwohl  $S_2$  und  $S_3$  ein gemeinsames Attribut besitzen.
- (c) Handelt es sich bei den folgendem Ausdrücken jeweils um gültige Anfragen – im Tupel- oder Domänenkalkül – über dem Schema  $R(A)$ ,  $S(ABC)$ , und falls ja, in was für einer Anfragesprache wurde sie formuliert?

(1)  $\{ [t] \mid [t] \in R \wedge \exists s, u ([t, s, u] \in S \wedge t = s) \}$

$$(2) \{[t] \mid t \in R \wedge \exists[s.A, t.A, s.C] \in S(s.C > 4) \vee t.A = s.A\}$$

### Lösung:

- (a) **Falsch:** Wenn zwei Relationenschema kein gemeinsames Attribut haben, so degeneriert der natürliche Verbund zum Kreuzprodukt. D.h. das Ergebnis von  $R \bowtie S = R \times S$  sieht wie folgt aus: Die Attributmenge des Ergebnisschemas ist  $\alpha \cup \beta$ , und die Ausprägung enthält alle Kombinationen von Tupeln in der Ausprägung von  $R$  und Tupeln der Ausprägung von  $S$ .
- (b) **Richtig:** Schlüssel müssen *Teilmengenminimal* sein, nicht *Kardinalitätsminimal*. Das heißt, die Minimalität bezieht sich nicht auf die absolute Anzahl an Attributen, sondern nur darauf, dass keine Teilmenge der Attribute ebenfalls noch die Tupel (bzw. Entitäten im Fall von EER-Diagrammen) eindeutig identifiziert. Nachdem weder  $S_1 \subseteq S_2$ ,  $S_2 \subseteq S_3$ , noch  $S_1 \subseteq S_3$  gilt sind alle drei Attributmengen gültige Schlüssel.
- (c) Es gilt folgendes:
- (1) **Ja**, es handelt sich um eine gültige Anfrage im Domänenkalkül: Die Variablen werden an einzelne Attribute gebunden. Tupel werden mittels dem Tupelkonstruktor  $[ ]$  erstellt.
  - (2) **Nein**, es handelt sich hier nicht um eine gültige Anfrage, weder im Tupel- noch im Domänenkalkül:  $[t]$  ist die Syntax des Domänenkalküls (im Tupelkalkül kann der Tupelkonstruktor  $[ ]$  nicht direkt auf eine Variable angewendet werden, da diese ja bereits auf ein gesamtes Tupel gebunden wird), wohingegen  $t \in R$  die Syntax des Tupelkalküls ist (die Tupelvariable  $t$  wird an Tupel in  $R$  gebunden).  $[s.A, t.A, s.C] \in S$  ist eine unerlaubte Mischung aus der Syntax beider Sprachen:  $s.A$  erlaubt im Tupelkalkül auf einzelne Attribute einer Tupelvariable zuzugreifen. Jedoch ist dort die Verwendung des Tupelkonstruktors ausschließlich zur Definition des Ergebnisses erlaubt, jedoch keine Ausdrücke der Form  $[...] \in R$ , da Variablen im Tupelkalkül immer an ganze Tupel gebunden werden. Außerdem ist die Variable  $s$  nicht definiert. Die restlichen Ausdrücke stellen wiederum Ausdrücke in der Syntax des Tupelkalküls dar.

## Zusatzaufgabe

### Aufgabe 11 (Organisatorisches) [0.0 Punkte]

- a) Bis wann müssen Sie sich zu einem Tutorengespräch anmelden wenn Sie an einem teilnehmen wollen?
- b) Bis wann müssen Sie sich zu einem Abgabegespräch anmelden wenn Sie an einem teilnehmen wollen?

- c) Gilt für die Anmeldungen zu den Tutoren- bzw. Abgabegesprächen die selbe Deadline?
- d) Wo bzw. wie können Sie sich zu einem Tutoren- oder Abgabegespräch anmelden?

Die richtigen Antworten finden Sie auf der ersten Seite des Übungsblattes.