

# Informationssysteme - Zettel 8

Christian Schulz, Lukas Schäfer, Lukas Schwitzgebel, Philipp Mirolid

July 17, 2017

## 1 Normalformen

- a) **Kandidatenschlüssel:**  $\{A, B\}$  richtig  
**Nicht 3NF ( $\rightarrow$  1NF):**  
 $A \rightarrow C$ :  $A$  ist kein Superschlüssel und  $C$  ist nicht prim, da  $C \notin \{A, B\}$ .
- b) **Kandidatenschlüssel:**  $\{A\}, \{B\}$  richtig  
**Nicht 3NF ( $\rightarrow$  1NF):**  
 $CD \rightarrow E$ :  $CD$  ist nicht Superschlüssel und  $E \notin \{A\} \wedge E \notin \{B\} \Rightarrow E$  nicht prim.
- c) **Kandidatenschlüssel:**  $\{A, B\}, \{C, F\}$  {C,E}  
**3NF:**
- $AB \rightarrow CDEF$ :  $AB$  ist Kandidatenschlüssel  $\Rightarrow AB$  ist Superschlüssel.
  - $CF \rightarrow AB$ :  $CF$  ist Kandidatenschlüssel  $\Rightarrow CF$  ist Superschlüssel.
  - $E \rightarrow F$ :  $F \in \{C, F\} \Rightarrow F$  ist prim.
- d) **Kandidatenschlüssel:**  $\{MatrikelNr, Arbeitgeber\}, \{SozialversNr, Arbeitgeber\}$   
**Nicht 3NF ( $\rightarrow$  1NF):** richtig  
 $MatrikelNr \rightarrow Studiengang$ :  $MatrikelNr$  ist nicht Superschlüssel und  $Studiengang \notin \{MatrikelNr, Arbeitgeber\} \wedge Studiengang \notin \{SozialversNr, Arbeitgeber\} \Rightarrow Studiengang$  nicht prim.
- e) **Kandidatenschlüssel:**  $\{ArtNr\}$  richtig  
**Nicht 3NF ( $\rightarrow$  1NF):**  
 $Logistikfirma \rightarrow Versandkosten$ :  $Logistikfirma$  ist nicht Superschlüssel und  $Versandkosten \notin \{ArtNr\} \Rightarrow Versandkosten$  nicht prim.
- f) **Kandidatenschlüssel:**  $\{Nutzer, Bilderalbum\}, \{Nutzer, Albumtitel\}$   
**3NF:** richtig
- $Nutzer, Bilderalbum \rightarrow Albumtitel, Öffentlich$ :  $Nutzer, Bilderalbum$  ist Kandidatenschlüssel  $\Rightarrow Nutzer, Bilderalbum$  ist Superschlüssel.
  - $Nutzer, Albumtitel \rightarrow Bilderalbum$ :  $Nutzer, Albumtitel$  ist Kandidatenschlüssel  $\Rightarrow Nutzer, Albumtitel$  ist Superschlüssel.

Wenn ihr eine neue Relation (wie unten) einfügt, kann eine Rechnung mehrere Details und damit auch Produkte haben, was euer Problem in 2.1 löst  
[Details]:{[Rechnungsnummer(Schlüssel) -> Rechnung,...]}  
bei .. Weitere Attribute von Details einfügen

## 2 Normalisierung

Nach Videos von Dittrich bzgl. der Normalformen, darf die 1. Normalform keinerlei Datenstrukturen als Typen für Attribute beinhalten. Da hier *Details* eine Liste von Tupeln sein soll, ist dies nicht zulässig für bereits die 1. NF. Daher "falten" wir diese Liste erst einmal auf. Dadurch verändert sich das Schema und vorherige Einträge werden nun zu mehreren Einträgen (ein neuer Eintrag für jedes Tupel in der Liste *Details*).

**Neues Schema:** *Rechnung* : (*Rechnungsnummer*, *Datum*, *KundenID*, *KundenName*, *KundenAdresse*, *ProduktID*, *ProduktName*, *Einzelpreis*, *Anzahl*)

### 2.1 geltende funktionale Abhängigkeiten FDs

- $ProduktID \rightarrow Produktname, Einzelpreis$

Von hier an sind wir nicht sicher wie wir fortfahren sollen.

Bezüglich des weiteren Satzes "Es gibt keine zwei Tupel auf einer Rechnung mit der gleichen *ProduktID*" in der Beschreibung ist uns nicht klar, wie wir diesen in einer FD darstellen sollen. Weder darf eine Rechnung nur ein Produkt enthalten, noch darf ein Produkt nur in einer Rechnung vorkommen.

Darüber hinaus wird in den Videos keine Schema oder dergleichen präsentiert um eine solche Beschreibung in eine 3NF zu übertragen. Sollte man hierzu wie in Aufgabe 3 den Synthesealgorithmus nutzen?

## 3 Synthesealgorithmus

### 3.1 funktionale Abhängigkeiten

$MatrNr, SName, Semester, SAdresse \rightarrow PersNr, Name, Fachgebiet$

$MatrNr \rightarrow SName, Semester, SAdresse$

$BossPersNr \rightarrow BossName$

$PersNr \rightarrow Name, Fachgebiet$

, also auch  
 $PersNr, Name, Fachgebiet$   
 $BossNr, BossName$

### 3.2 Kandidatenschlüssel

$\{MatrNr, BossPersNr\}$

es reicht die MatrNr aus,  
da jeder Boss eindeutig  
bestimmt ist

Mit *MatrNr* bekommen wir jegliche Information zu einem Studenten und von diesem aus auch über seinen Tutor. Lediglich die Informationen über den Professor fehlen noch und diese erhalten wir mit *BossPersNr*.

### 3.3 kanonische Überdeckung der funktionalen Abhängigkeiten

Bei Links- und Rechtsreduktion bzw. der kanonischen Überdeckung handelt es sich eigentlich um die ersten 2 Schritte des Synthesealgo-

rithmus.

richtig, mit eurem  
Schema :)

1. **Linksreduktion:**

$MatrNr, SName, Semester, SAdresse \rightarrow PersNr, Name, Fachgebiet$   
wird zu  $MatrNr \rightarrow PersNr, Name, Fachgebiet$ , da bereits mit  
 $MatrNr \rightarrow SName, Semester, SAdresse$  die unterstrichenen Attribute  
von  $MatrNr$  erreicht werden. Somit stellen diese keinen Mehrwert in der  
Prämisse dar.

2. **Rechtsreduktion:**

$MatrNr \rightarrow PersNr, Name, Fachgebiet$   
wird zu  $MatrNr \rightarrow PersNr$ , da bereits mit  
 $PersNr \rightarrow Name, Fachgebiet$  die unterstrichenen Attribute von  $PersNr$   
erreicht werden. Somit stellen diese keinen Mehrwert in der Konklusion  
dar.

### 3.4 Synthesalgorithmus

3. **Leere Klauseln:**

Es gibt keine leeren Klauseln/ funktionale Abhängigkeiten bei denen die  
rechte Seite  $\emptyset$  entspricht.

4. **Zusammenfassen:**

$MatrNr \rightarrow PersNr$  und  $MatrNr \rightarrow SName, Semester, SAdresse$   
wird zu  
 $MatrNr \rightarrow PersNr, SName, Semester, SAdresse$ .  
Zusätzlich bleiben vorherige Abhängigkeiten bestehen. Dies führt ab-  
schließend zu folgenden Abhängigkeiten:

$$\begin{aligned} &MatrNr \rightarrow PersNr, SName, Semester, SAdresse \\ &BossPersNr \rightarrow BossName \\ &PersNr \rightarrow Name, Fachgebiet \end{aligned}$$

**Diese Menge von funktionalen Abhängigkeiten ist allerdings keine 3NF!**

So ist beispielsweise weder  $BossPersNr$  ein Superschlüssel, noch ist  $BossName$   
in dem Kandidatenschlüssel enthalten und somit prim. Woran liegt das?

Da ihr nicht beachtet habt, dass kein  
Student genau ein Boss haben kann und  
somit keine Relation zwischen den  
Studenten und den Bossen habt