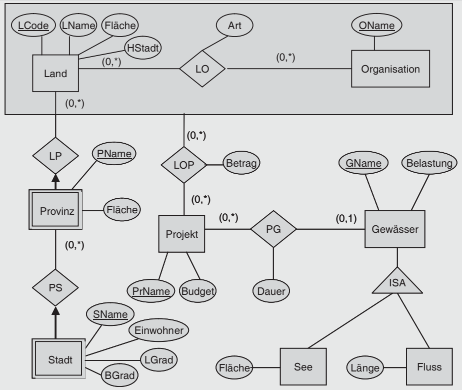
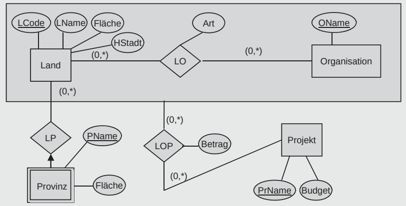
Rückblick: ER-Modellierung

ER-Schema



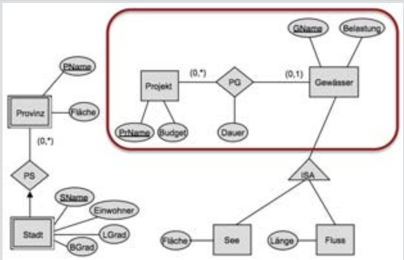
Transformation A



Land(<u>LCode</u>, HStadt, LName, Fläche)
Provinz(<u>PName</u>, <u>LCode</u>, Fläche)
Organisation(<u>OName</u>)
Projekt(<u>PrName</u>, Budget)

LO(<u>LCode</u>, <u>OName</u>, Art)
LOP(<u>LCode</u>, <u>OName</u>, <u>PrName</u>, Betrag)

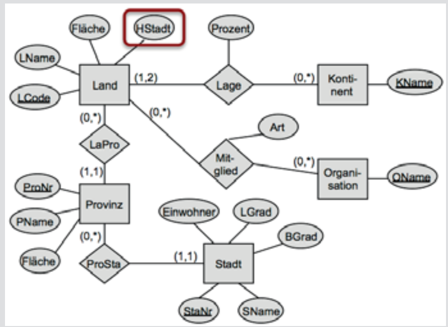
Transformation B



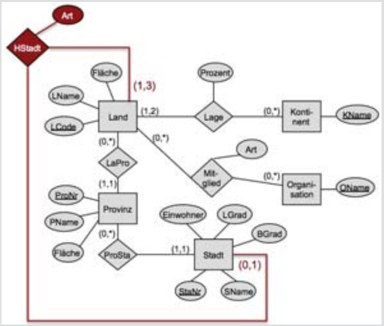
Provinz(<u>PName</u>, <u>LCode</u>, Fläche)
Stadt(<u>SName</u>, <u>LCode</u>, <u>PName</u>, Einwohner, LGrad, BGrad)
Projekt(<u>PrName</u>, Budget)
Gewässer(<u>GName</u>, Belastung)
See(<u>GName</u>, Fläche)
Fluss(<u>GName</u>, Länge)

PG(<u>GName</u>, PrName, Dauer)

Wie könnte man die Modellierung anpassen, um mehrere Hauptstädte zu unterstützen?



Wie könnte man die Modellierung anpassen, um mehrere Hauptstädte zu unterstützen?



Ausblick

Januar/Februar

- Kapitel 6: Formaler Datenbankentwurf
- Kapitel 7: Physischer Datenbankentwurf
- Kapitel 8: Auswertung von Anfrageoperatoren
- Kapitel 9: Transaktionsverwaltung

Kapitel 6: Formaler Datenbankentwurf

- Die Schwierigkeiten der konzeptuellen Modellierung sind zu einem großen Teil dadurch begründet, dass sich die relevanten Strukturen einer Miniwelt erst in Diskussionen mit den Anwendern oder durch Analyse von Dokumenten erfassen lassen.
- Mit der Transformation eines konzeptuellen Schemas in ein relationales Schema, dem logischen Entwurf, ändert sich diese Situation.
- Die Konzepte des relationalen Datenmodells werden dahingehend erweitert, dass sich die Güte eines logischen Entwurfs formal überprüfen läßt.
- Ziel ist die Vermeidung von sogenannten Anomalien.

6.1 Motivation

Relationen mit Anomalien

Stadt

SNr	SName	LCode	LFläche
7	Freiburg	D	357
9	Berlin	D	357
40	Moscow	RU	17075
43	St.Petersburg	RU	17075

Kontinent

KName	LCode	KFläche	Prozent
Europe	D	3234	100
Europe	RU	3234	20
Asia	RU	44400	80

Anomalien bzgl. Relation `Stadt`

Stadt

SNr	SName	LCode	LFläche
7	Freiburg	D	357
9	Berlin	D	357
40	Moscow	RU	17075
43	St.Petersburg	RU	17075

- Anomalie beim Einfügen: Es können nur Länder aufgenommen werden, zu denen auch Städte existieren.
- Anomalie beim Löschen: Werden Städte gelöscht, können u.U. alle Informationen über gewisse Länder verloren gehen.
- Anomalie beim Ändern: Änderungen der Fläche eines Landes müssen bei mehreren Zeilen vorgenommen werden.

Relationen ohne Anomalien

Stadt'

SNr	SName
7	Freiburg
9	Berlin
40	Moscow
43	St.Petersburg

Land'

LCode	LFläche
D	357
RU	17075

StadtLand'

SNr	LCode
7	D
9	D
40	RU
43	RU

Lage'

LCode	KName	Prozent
D	Europe	100
RU	Europe	20
RU	Asia	80

Kontinent'

KName	KFläche
Europe	3234
Asia	44400

6.2 Funktionale Abhängigkeiten

6.2.1 Definition

- Sei ein Relationsschema gegeben durch sein Format V und seien $X, Y \subseteq V$.
- Sei $r \in \text{Rel}(V)$. r erfüllt eine *funktionale Abhängigkeit (FA)* $X \to Y$, wenn für alle $\mu, \nu \in r$ gilt:

$$\mu[X] = \nu[X] \Rightarrow \mu[Y] = \nu[Y].$$

  Wir sagen auch die FA $X \to Y$ gilt in r.
- Sei $\mathcal{F}$ eine Menge funktionaler Abhängigkeiten über V und $X, Y \subseteq V$. Die Menge aller Relationen r in $\text{Rel}(V)$, die alle funktionalen Abhängigkeiten in $\mathcal{F}$ erfüllen, bezeichnen wir mit $\text{Sat}(V, \mathcal{F})$.

$$r = \begin{array}{ccc} A & B & C \\ \hline a_1 & b_1 & c_1 \\ a_2 & b_1 & c_2 \end{array}$$

Welche FA werden durch r erfüllt?

FA	gilt in r
$A \to B$	1
$A \to C$	1
$B \to A$	0
$B \to C$	0
$C \to A$	1
$C \to B$	1
$AB \to C$	1
$AC \to B$	1
$BC \to A$	1
$A \to \emptyset$	1
$A \to A$	1
$AB \to A$	1
$ABC \to A$	1
⋮	⋮
$\emptyset \to A$	0
$\emptyset \to B$	1
$\emptyset \to C$	0
$\emptyset \to AB$	0
⋮	⋮

6.2.2 Membership-Test

- $\mathcal{F}$ *impliziert* die funktionale Abhängigkeit $X \to Y$, $\mathcal{F} \models X \to Y$, wenn jede Relation $r \in \mathrm{Sat}(V, \mathcal{F})$ auch $X \to Y$ erfüllt.
- Die Menge $\mathcal{F}^+ = \{X \to Y \mid \mathcal{F} \models X \to Y\}$ nennen wir die *Hülle* von $\mathcal{F}$.
- Der Test $X \to Y \in \mathcal{F}^+$ ist der *Membership-Test*.

Armstrong-Axiome

Sei $r \in \mathrm{Sat}(V, \mathcal{F})$.

(A1) Reflexivität: Wenn $Y \subseteq X \subseteq V$, dann erfüllt r die FA $X \to Y$.

(A2) Augmentation: Wenn $X \to Y \in \mathcal{F}, Z \subseteq V$, dann erfüllt r auch die FA $XZ \to YZ$.

(A3) Transitivität: Wenn $X \to Y, Y \to Z \in \mathcal{F}$, dann erfüllt r auch die FA $X \to Z$.

(A1) erlaubt die Herleitung funktionaler Abhängigkeiten, ohne Bezug auf $\mathcal{F}$: *triviale funktionale Abhängigkeiten*.

Korrektheit und Vollständigkeit

- Die Armstrong-Axiome sind *korrekt* in dem Sinn, dass die mit ihnen herleitbaren funktionalen Abhängigkeiten in der Tat Elemente der Hülle $\mathcal{F}^+$ sind.
- Die Armstrong-Axiome sind auch *vollständig*, d.h., jede funktionale Abhängigkeit in $\mathcal{F}^+$ kann auch mit ihnen hergeleitet werden.

Membership-Test Variante 1:

Starte mit $\mathcal{F}$ und wende solange die Regeln (A1)–(A3) an, bis entweder $X \to Y$ hergeleitet, oder $\mathcal{F}^+$ hergeleitet und $X \to Y \notin \mathcal{F}^+$.

Ein solcher Algorithmus ist im Allgemeinen mindestens exponentiell in der Anzahl der Attribute in V, da für (A2) alle Teilmengen von V betrachtet werden müssen.

Beispiel Membership-Test

Sei $V = \{A, B, C, D, E, F, G, H, I\}$ und
$\mathcal{F} = \{AB \to E, BE \to I, E \to G, GI \to H\}$.
Es soll getestet werden, ob $AB \to GH \in \mathcal{F}^+$.

1. $AB \to E \overset{A2}{\Rightarrow} AB \to ABE$
2. $BE \to I \overset{A2}{\Rightarrow} ABE \to ABEI$
3. $\overset{A3}{\Rightarrow} AB \to ABEI$
4. $E \to G \overset{A2}{\Rightarrow} ABEI \to ABEGI$
5. $\overset{A3}{\Rightarrow} AB \to ABEGI$
6. $GI \to H \overset{A2}{\Rightarrow} ABEGI \to ABEGHI$
7. $\overset{A3}{\Rightarrow} AB \to ABEGHI$
8. $\overset{A1}{\Rightarrow} ABEGHI \to GH$
9. $\overset{A3}{\Rightarrow} AB \to GH$

weitere Axiome

Seien $X, Y, Z, W \subseteq V$ und $A \in V$.

(A4) Vereinigung: Wenn $X \to Y, X \to Z \in \mathcal{F}$, dann erfüllt r auch die FA $X \to YZ$.

(A5) Pseudotransitivität: Wenn $X \to Y, WY \to Z \in \mathcal{F}$, dann erfüllt r auch die FA $XW \to Z$.

(A6) Dekomposition: Wenn $X \to Y \in \mathcal{F}, Z \subseteq Y$, dann erfüllt r auch die FA $X \to Z$.

(A7) Reflexivität: Wenn $X \subseteq V$, dann erfüllt r auch die FA $X \to X$.

(A8) Akkumulation: Wenn $X \to YZ, Z \to AW \in \mathcal{F}$, dann erfüllt r auch die FA $X \to YZA$.

Die Axiomensysteme $\{$(A1), (A2), (A3)$\}$ und $\{$(A6), (A7), (A8)$\}$ sind zueinander äquivalent.

Hierzu müssen wir zeigen, dass jedes Axiom der einen Menge durch die Axiome der anderen Menge simuliert werden kann.

Beispiel: (A6) kann durch (A1) und (A3) simuliert werden.

Seien $X, Y, Z \subseteq Y$.
Zu zeigen: $X \to Y, Z \subseteq Y \Rightarrow X \to Z$.

$$Z \subseteq Y \quad \overset{A1}{\Rightarrow} \quad Y \to Z$$
$$X \to Y, Y \to Z \quad \overset{A3}{\Rightarrow} \quad X \to Z$$

Beispiel Membership-Test – mit Axiomen (A1-A8)

Sei $V = \{A, B, C, D, E, F, G, H, I\}$ und
$\mathcal{F} = \{AB \to E, BE \to I, E \to G, GI \to H\}$.
Es soll getestet werden, ob $AB \to GH \in \mathcal{F}^+$.

1. $AB \to E, E \to G$
2. $\overset{A3}{\Rightarrow} AB \to G$
3. $\overset{A1}{\Rightarrow} AB \to B$
4. $\overset{A4}{\Rightarrow} AB \to BE, BE \to I$
5. $\overset{A3}{\Rightarrow} AB \to I$
6. $\overset{A4}{\Rightarrow} AB \to GI, GI \to H$
7. $\overset{A3}{\Rightarrow} AB \to H$
8. $\overset{A4}{\Rightarrow} AB \to GH$

(Attribut-)Hülle X^+ von X (bzgl. $\mathcal{F}$)

$$X^+ = \{A \mid A \in V \text{ und } X \to A \in \mathcal{F}^+\}.$$

Membership-Test Variante 2:

Berechne zunächst X^+ mittels (A6) - (A8) und teste anschließend, ob $Y \subseteq X^+$.

XPlus-Algorithmus

```
XPlus(X,Y,F) boolean {
  result := X;
  WHILE (changes to result) DO
    FOR each X' → Y' ∈ F DO
       IF (X' ⊆ result) THEN result := result ∪ Y';
    end.
  IF (Y ⊆ result) RETURN true ELSE false;
}
```


Der XPlus-Algorithmus hat eine Laufzeit, die polynomiell in der Darstellung von $\mathcal{F}$ ist.

Beispiel XPlus-Algorithmus

Sei $V = \{A, B, C, D, E, F, G, H, I\}$ und
$\mathcal{F} = \{AB \to E, BE \to I, E \to G, GI \to H\}$.

Es soll getestet werden, ob $AB \to GH \in \mathcal{F}^+$.

Axiom	Anwendung	`result`
(A7)	$AB \to AB$	$\{A, B\}$
(A8)	$AB \to ABE$	$\{A, B, E\}$
(A8)	$AB \to ABEI$	$\{A, B, E, I\}$
(A8)	$AB \to ABEIG$	$\{A, B, E, I, G\}$
(A8)	$AB \to ABEIGH$	$\{A, B, E, I, G, H\}$
(A6)	$AB \to GH$	

Schlüssel

Sei $V = \{A_1, \dots, A_n\}$. $X \subseteq V$ heißt *Schlüssel* für V (bzgl. $\mathcal{F}$), wenn

- $X \to A_1 \dots A_n \in \mathcal{F}^+$,
- $Y \subset X \Rightarrow Y \to A_1 \dots A_n \notin \mathcal{F}^+$.

Basierend auf dem XPlus-Algorithmus können wir zu gegebenen $V, \mathcal{F}$ einen Schlüssel berechnen wie folgt:

(1) Beginne mit $X := V$.
(2) Für jedes $A \in V$: falls $(X \setminus \{A\})^+ = V$, dann $X := X \setminus \{A\}$.
(3) X ist ein Schlüssel.

Bemerkung

- Jedes $A \in V$ wird nur einmal betrachtet; der XPlus-Algorithmus wird n-mal aufgerufen.
- Sofern mehrere Schlüssel existieren (Beispiel: $V = \{$`Stadt, Adresse, PLZ`$\}$, $\mathcal{F} = \{$`Stadt Adresse` $\to$ `PLZ`, `PLZ` $\to$ `Stadt`$\}$), wird nur einer dieser berechnet.
- Der Test, ob eine Attributemenge Schlüssel ist, ist exponentiell (NP-vollständig).

6.2.3 Ausblick

Wie sieht der Schlüssel für die Menge der FAs $\mathcal{F}$ aus?

Sei $V = \{A, B, C, D, E, F, G, H, I\}$ und
$\mathcal{F} = \{AB \to E, BE \to I, E \to G, GI \to H\}$.