

DAB1 – Datenbanken 1

Dr. Daniel Aebi (aebd@zhaw.ch)

Lektion 14: Rückblick, Ausblick

Wo stehen wir?

Grundlagen Mengen/BA	Einführung
	Relationale Algebra Relationale Bags
	Entity-Relationship Design
	SQL



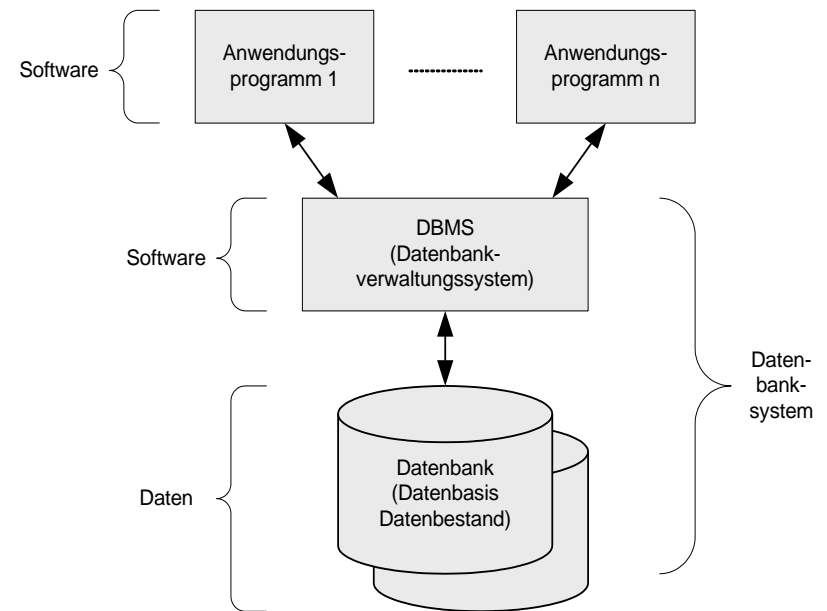
← "You are here"

Ziele Lektion 14

- Rückblick++
- Einzelne Aspekte ggf. noch ergänzen
- Begriff «physischer Entwurf» kennen
- Ausblick auf DAB2

Rückblick++

- Wie alles begann...
- Aufgaben eines DBS (Begriffe: DB, DBMS, DBS, IS)
 - Datenbanksystem (DBS)
 - DBMS (Datenbankverwaltungssystem)
 - Datenbank (Datenbasis, Datenbestand)
 - Anwendungsprogramme (AP)
 - DBS + AP = Informationssystem (IS)



Rückblick++

- Relationenmodell – wichtige Begriffe:
 - Attribut
 - Domäne (Datentyp, Wertebereich)
 - Tupel
 - Relationenformat
 - Relation
 - Bag

Rückblick++

- Relationenmodell – Operationen der relationalen Algebra:
 - Entfernende Operatoren:
 - Selektion, Projektion
 - Mengenoperatoren:
 - Vereinigung, Schnittmenge, Differenz
 - Kombinerende Operatoren:
 - Kartesisches Produkt, Join, Joinvarianten
 - Umbenennung:
 - Verändert nicht Tupel, sondern Format

Rückblick++

- Relationenmodell – Operatoren:
 - σ Selektion
 - π Projektion
 - \times Kreuzprodukt
 - \bowtie Join (Verbund)
 - \setminus Mengendifferenz
 - \cup Mengenvereinigung
 - \cap Mengendurchschnitt
 - \div Division
 - ρ Umbenennung
 - \Join Linker äusserer Verbund
 - \Join Rechter äusserer Verbund
 - \Join Voller äusserer Verbund

→ Hilfsmittel für die Modulprüfung

Rückblick++

- Relationenmodell – Verbundvarianten:
 - Natural Join
 - Theta-Join
 - Cross-Join
 - Self-Join
 - Outer-Joins

Rückblick++

- **Natürlicher Verbund** (natural join):
- Gegeben seien: $R(A_1, \dots, A_m, B_1, \dots, B_k)$ und $S(B_1, \dots, B_k, C_1, \dots, C_n)$
- $R \bowtie S = \pi_{A_1, \dots, A_m, R.B_1, \dots, R.B_k, C_1, \dots, C_n}(\sigma_{R.B_1 = S.B_1 \wedge \dots \wedge R.B_k = S.B_k}(R \times S))$

$R \bowtie S$											
$R - S$				$R \cap S$				$S - R$			
A_1	A_2	...	A_m	B_1	B_2	...	B_k	C_1	C_2	...	C_n

→ Hilfsmittel für die Modulprüfung

Gelb: bezieht sich auf das Format

Rückblick++

- Theta join: **Verallgemeinerung** des natürlichen Joins.
- Verknüpfungsbedingung kann selbst gestaltet werden.
- Konstruktion des Ergebnisses:
 - Bilde Kreuzprodukt
 - Selektiere mittels der Joinbedingung
 - Also: $R \bowtie_{\theta} S = \sigma_{\theta}(R \times S)$
 - $\theta \in \{=, <, >, \leq, \geq, \neq\}$
- Schema: Wie beim Kreuzprodukt.
- Natural Join ist ein Spezialfall des Theta-Joins.

Rückblick++

- **Allgemeiner Verbund** (theta join):
- Gegeben seien: $R(A_1, \dots, A_n)$ und $S(B_1, \dots, B_m)$
- $R \bowtie_{\theta} S = \sigma_P(R \times S)$ (kartesisches Produkt von R und S mit Selektion)

$R \bowtie_{\theta} S$							
R				S			
A_1	A_2	...	A_n	B_1	B_2	...	B_m

→ Hilfsmittel für die Modulprüfung

Rückblick++

- Relationenmodell – bag-Operationen:
 - Selektion, Projektion: Wie gehabt, aber mit Duplikaten
 - Joins: Wie gehabt, aber mit Duplikaten
 - Mengenoperationen:
 - $r \cup s = \{ \langle t, k \rangle \in \text{dom}(R) \times \mathbb{N} \mid k = \max(r(t), s(t)) \}$ "bag union"
 - $r \sqcup s = \{ \langle t, k \rangle \in \text{dom}(R) \times \mathbb{N} \mid k = r(t) + s(t) \}$ "bag concatenation"
 - $r \cap s = \{ \langle t, k \rangle \in \text{dom}(R) \times \mathbb{N} \mid k = \min\{r(t), s(t)\} \}$
 - $r \setminus s = \{ \langle t, k \rangle \in \text{dom}(R) \times \mathbb{N} \mid k = \max\{0, r(t) - s(t)\} \}$

→ Hilfsmittel für die Modulprüfung

Rückblick++

- Hörsaalübung 1:
 - Gegeben: 3 Relationen
(alle domains integer):

r	A	B	C
	0	0	0
	0	0	1
	1	0	0
	1	0	1
	1	1	0

s	B	C	D
	0	0	0
	0	0	1
	0	1	0
	0	1	1
	1	0	0
	1	0	1
	1	1	0
	1	1	1

u	D	E
	0	0
	0	1
	1	1

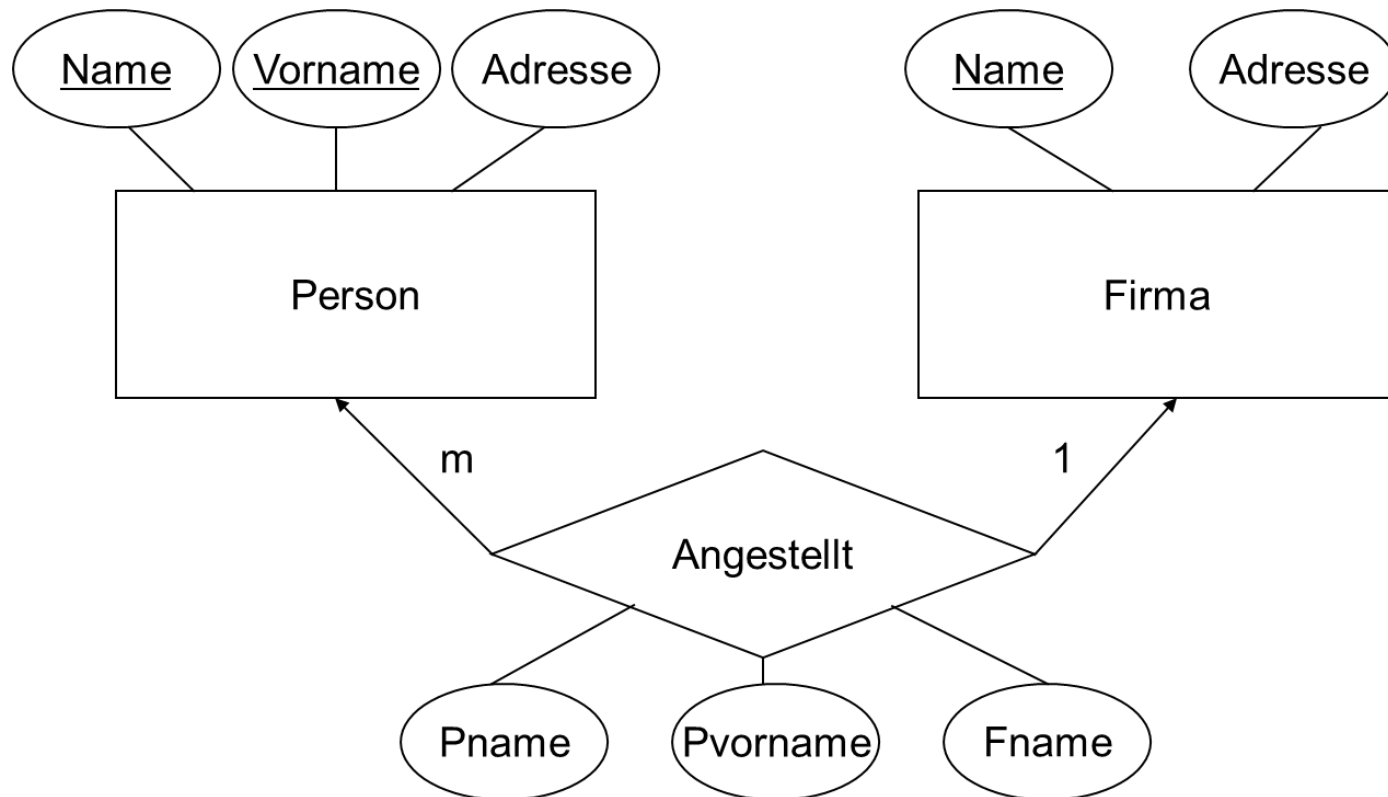
- Gesucht: $(r \bowtie \sigma_{B=1}(s) \bowtie u) \cap (r \bowtie \sigma_{D=1}(s) \bowtie u)$

Rückblick++

- Entwurfsmethode – wichtige Begriffe:
 - Entitätstypen (unabhängige, abhängige)
 - Beziehungstypen
 - Attribute
 - Kardinalitäten

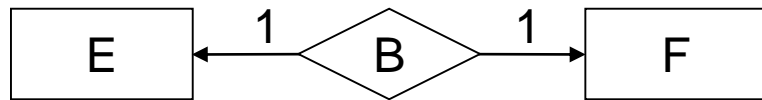
Rückblick++

- Entwurfsmethode – Beispiel:
 - Kardinalitäten: 1, m, ID, ISA

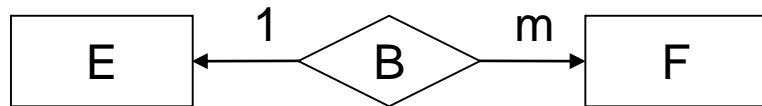


Rückblick++

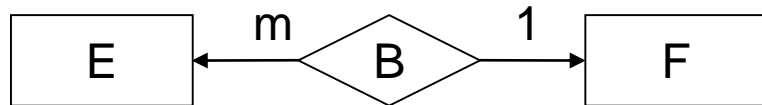
- Entwurfsmethode – Kardinalitäten:
 - Werden mit Hilfe von Schlüsseln durchgesetzt:



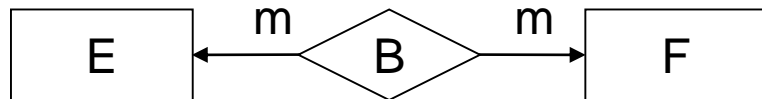
Schlüssel von E **und** F



Schlüssel von F



Schlüssel von E

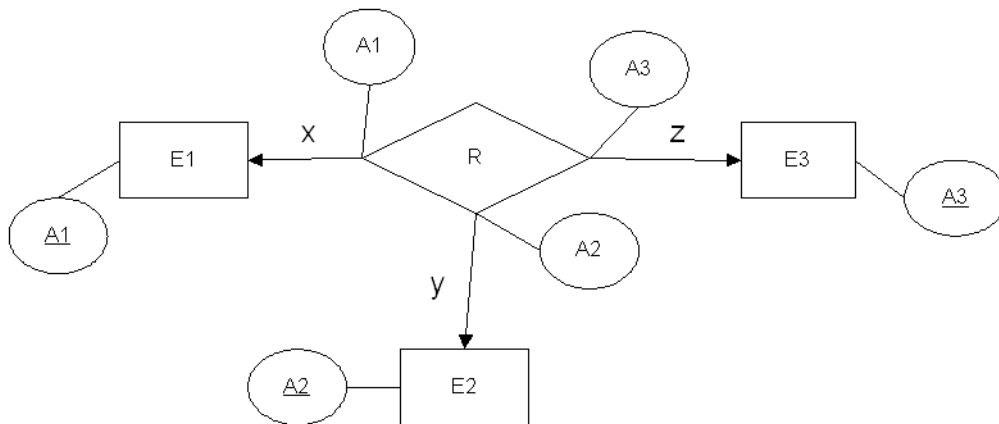


Schlüssel von E **komb.** F

→ Hilfsmittel für die Modulprüfung

Rückblick++

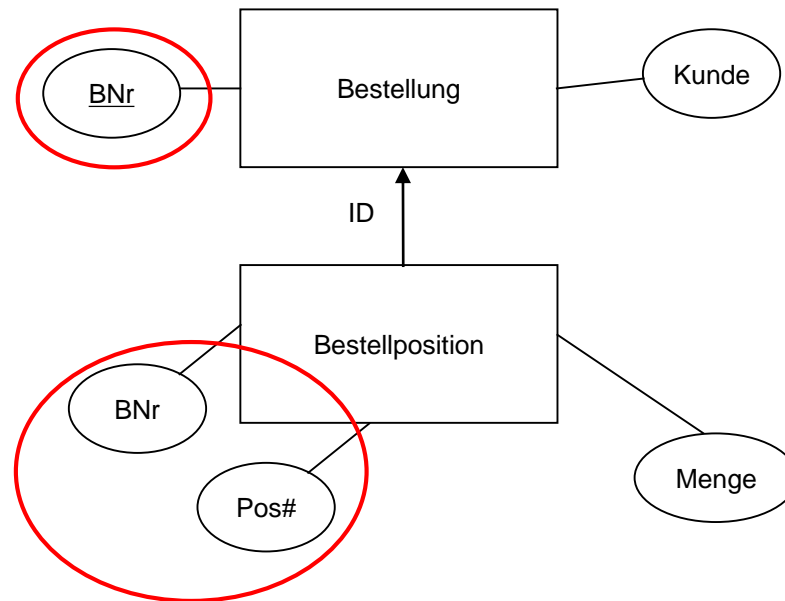
- Entwurfsmethode – Kardinalitäten:
 - Werden mit Hilfe von Schlüsseln durchgesetzt:



x	y	z	Schlüssel
m	m	m	{A1, A2, A3}
m	m	1	{A1, A2}
m	1	m	{A1, A3}
m	1	1	{A1, A2} und {A1, A3}
1	m	m	{A2, A3}
1	m	1	{A2, A3} und {A1, A2}
1	1	m	{A2, A3} und {A1, A3}
1	1	1	{A2, A3} und {A1, A3} und {A1, A2}

→ Hilfsmittel für die Modulprüfung

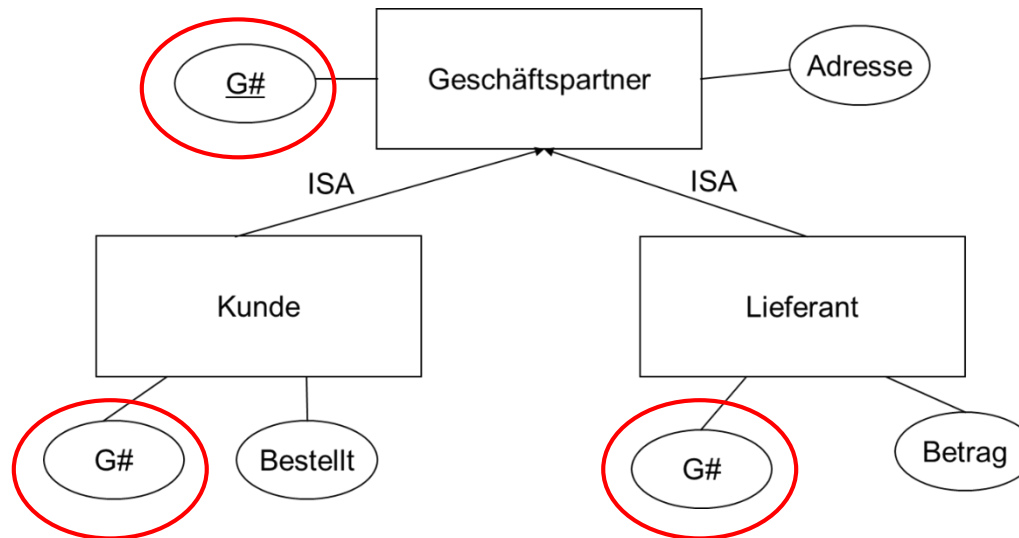
- Entwurfsmethode – ID-Abhängigkeit:



- Bei einer Bestellposition brauchen wir ein weiteres Attribut, z.B. «Pos#», um die einzelnen Bestellpositionen zu unterscheiden.

→ Hilfsmittel für die Modulprüfung

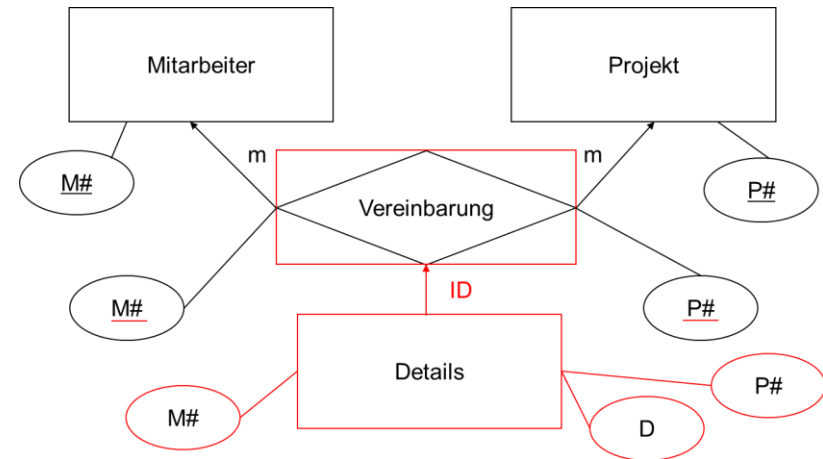
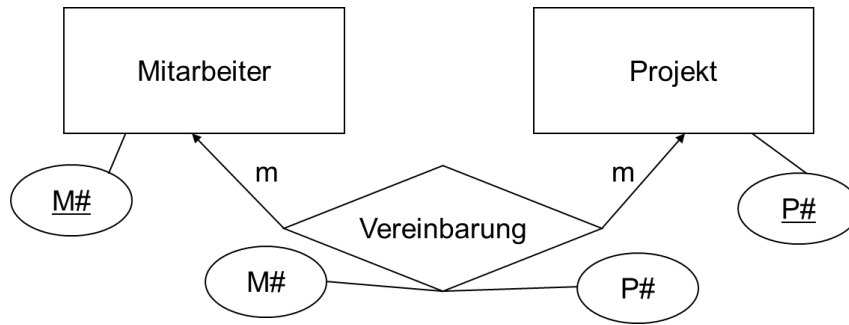
- Entwurfsmethode – ISA-Abhängigkeit:



- Jeder Kunde resp. jeder Lieferant ist auch Geschäftspartner.

→ Hilfsmittel für die Modulprüfung

- Entwurfsmethode – zusammengesetzte Entitätstypen:
 - Wir wollen unterschiedlich viele (auch keine) Details zu jeder Vereinbarung festhalten können



- Achtung: An einem ZET kann auch wieder ein BT angehängt sein!

→ Hilfsmittel für die Modulprüfung

Rückblick++

- Entwurfsmethode – korrektes Diagramm (6 Regeln):
 1. Definiere unabhängigen Entitätstyp
 2. Definiere Beziehungstyp
 3. Definiere Attribut
 4. Wandle Beziehungstyp in zusammengesetzten Entitätstyp um
 5. Definiere ID-abhängigen Entitätstyp
 6. Definiere ISA-abhängigen Entitätstyp

→ Hilfsmittel für die Modulprüfung

Rückblick++

- Hörsaalübung 2:
 - Beispiel: Projektleiter (ein Attribut: PL#) sind zuständig für Projekte (ein Attribut: P#) und zwar höchstens ein Projektleiter für ein Projekt. Ein Projektleiter kann aber für mehrere Projekte zuständig sein. Es können jedoch mehrere Projektleiter – in der Rolle als Stellvertreter – zuständig sein für ein Projekt. Zeichnen Sie das dazu passende ER-Schema.

Rückblick++

- SQL: DDL
 - CREATE, ALTER, DROP
 - Datentypen
 - Constraints (Schlüssel – PRIMARY/FOREIGN, CHECK, DEFAULT, ...)
 - ...
- SQL: DML
 - INSERT, UPDATE, DELETE
 - Direkt, mit Subqueries
 - ...

Rückblick++

- SQL: DQL
 - SELECT – FROM – WHERE – GROUP BY – HAVING – ORDER BY
 - Verbundvarianten
 - Aliase
 - Korrelierte / unkorrelierte Subqueries
 - [NOT] EXISTS
 - [NOT] IN
 - ANY, ALL, SOME
 - LIKE
 - Aggregatfunktionen (SUM, AVG, MIN, MAX, COUNT)
 - Gruppierung
 - Sortierung
 - NULL's
 - Sichten
 - ...

Rückblick++

- Hörsaalübung 3:

- Ausgangslage: 2 Tabellen
(Domänen: int bzw. char(1))

A	x	y
	1	A
	2	B
	3	C
	4	D

B	x	y
	1	A
	3	C

- Erstellen Sie mit MySQL Workbench eine neue Datenbank
- Formulieren Sie in MySQL-Workbench die notwendigen CREATE TABLE-Anweisungen für die Tabellen A und B
- Füllen Sie die beiden Tabellen mit Hilfe von INSERT-Anweisungen
- Formulieren Sie in SQL folgende Abfragen:
 - $A \cup B$ (jeweils mit und ohne Duplikate)
 - $A \setminus B$
 - $A \cap B$

Begriff: Physischer Entwurf

Bisher:

- Konzeptioneller Entwurf (→ ER-Sprache)
- Logischer Entwurf (→ SQL)

Was noch fehlt:

- Physischer Entwurf (→ SQL, andere Massnahmen)

Ziele:

- Performanz
- Physische Datenunabhängigkeit
- Sicherheit

Was kommt später noch auf Sie zu...

Themen, die in DAB2 behandelt werden:

- Stored Procedures und Trigger («Datenbankprogrammierung»)
- Speicherorganisationsformen
- Optimierung (Indexe, Statistiken, Abfragebäume, ...)
- Transaktionsverwaltung, Recovery
- Kurze Einführung in Non-standard-Datenbanksysteme (data warehouses, NoSQL)

Aber...

Es gibt noch sehr viele andere, wichtige und interessante Themen im Datenbankbereich:

- Nicht-relationale Datenbanken (XML-/OO-/OR-/...-DB)
- Datenbank-Anwendungsentwicklung
- Business Intelligence
- Information Retrieval
- "big data", NoSQL
- ...
- → Wahlfachmodule