

8. Datenbankdesign: Relationenmodell



Der Begriff der „Relation“

Seien A_1, A_2, \dots, A_n **Namen von Mengen**
und $W(A_1), W(A_2), \dots, W(A_n)$ die zugehörigen
Wertebereiche,

so ist eine **Relation** R definiert durch die **Untermenge**
des kartesischen Produkts der Wertebereiche

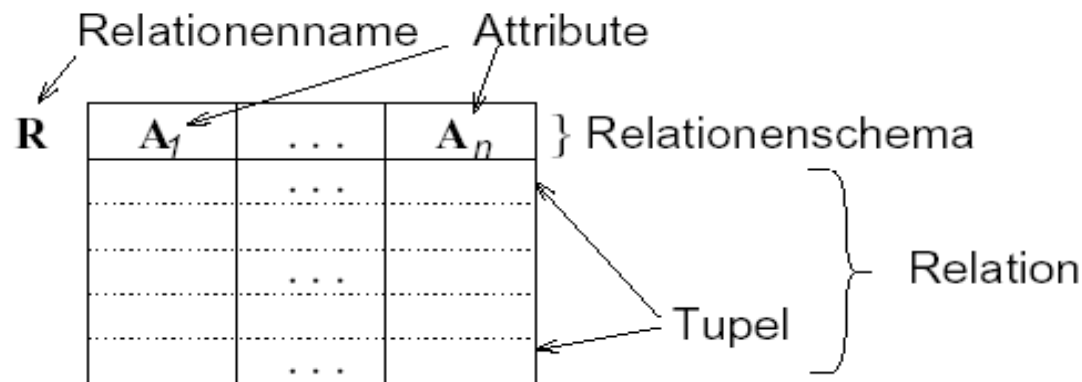
$$R(A_1, A_2, \dots, A_n) \subseteq W(A_1) \times W(A_2) \times \dots \times W(A_n)$$



Anschaulich kann man eine Ausprägung einer Relation als **Tabelle** darstellen!

Relationenmodell

Codd im Jahre 1970



Veranschaulichung eines Relationenschemas und einer Relation



Relationen können als Tabellen dargestellt werden

Zwei Relationen zur Darstellung von Personen

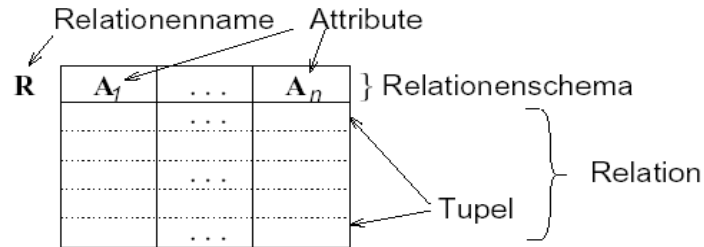
Personen	PANr	Vorname	Nachname	PLZ	Ort	Straße	HNr	Geb.datum
	4711	Andreas	Heuer	18209	DBR	BHS	15	31.10.1958
	5588	Gunter	Saake	39106	MD	STS	55	05.10.1960
	6834	Michael	Korn	39104	MD	BS	41	24.09.1974
	7754	Andreas	Möller	18209	DBR	RS	31	25.02.1976
	8832	Tamara	Jagellovsk	38106	BS	GS	12	11.11.1973
	9912	Antje	Hellhof	18059	HRO	AES	21	04.04.1970
	9999	Christa	Loeser	69121	HD	TS	38	10.05.1969

Pers_Telefon	PANr	Telefon
	4711	038203-12230
	4711	0381-498-3401
	4711	0381-498-3427
	5588	0391-345677
	5588	0391-5592-3800
	9999	06221-400177



Relationenmodell

Codd im Jahre 1970



Veranschaulichung eines Relationenschemas und einer Relation

- Eine Zeile nennt man **Tupel**
- Die Anzahl der Spalten n nennt man den **Grad** der Relation
- Die Anzahl der Zeilen nennt man die **Kardinalität** der Relation



Wichtige Merkmale: Eine Relation....

- hat einen eindeutigen Namen
- hat 0 - n **Tupel**
- die **Ordnung** der Tupel ist bedeutungslos für den Zugriff
- hat 1 - m Spalten_ (Attribute)
- die Ordnung der Attribute ist bedeutungslos für den Zugriff
- alle Attributwerte eines Attributs_ werden ein und derselben Menge entnommen
- hat mindestens einen **Schlüssel**, dessen Werte für die eindeutige Identifizierung der Tupel verwendet werden können



Schlüssel

- Primärschlüssel
- Fremdschlüssel
- Zugriffsschlüssel
- Sortierschlüssel
- Verschlüsselungsschlüssel
- ...



Primärschlüssel

Ein Primärschlüssel ist ein Attribut oder eine Kombination von Attributen, dessen Werte jeweils ein Tupel einer Relation eindeutig identifizieren.

Der Primärschlüssel einer Relation wird durch Unterstreichung gekennzeichnet.



Primärschlüssel als spezielle Integritätsbedingung

Integritätsbedingungen

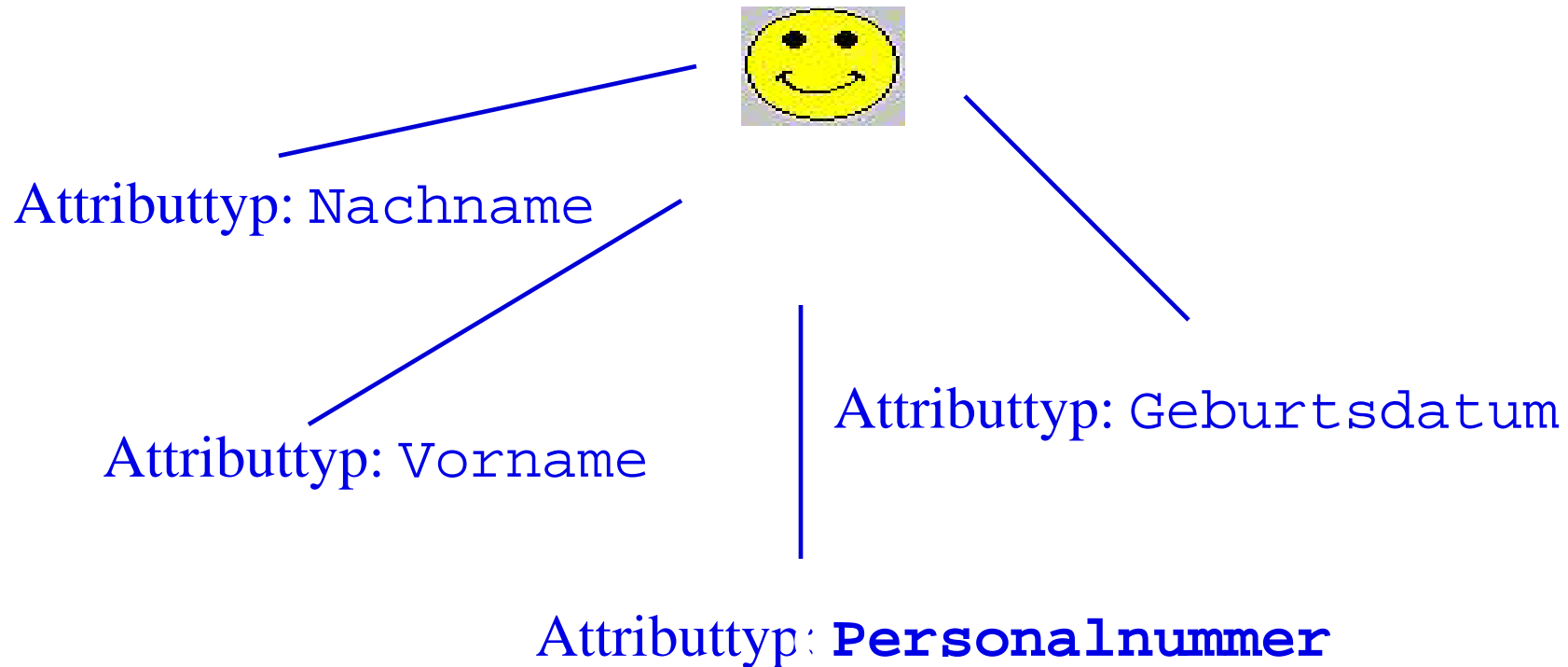
Identifizierende Attributmenge $K := \{B_1, \dots, B_k\} \subseteq R$:

$$\forall t_1, t_2 \in r [t_1 \neq t_2 \implies \exists B \in K : t_1(B) \neq t_2(B)].$$

- **Schlüssel**: ist minimale identifizierende Attributmenge
 {Vorname, Nachname, PLZ, Geburtsdatum} und
 {PANr} für Personen
 {PANr, Telefon} für Pers_Telefon
- **Primattribut**: Element eines Schlüssels
- **Primärschlüssel**: ausgezeichnete Schlüssel
- **Fremdschlüssel**: $\leadsto \square \square$



Primärschlüssel: Beispiel



Zusammenfassung Grundbegriffe

Begriffe des Relationenmodells

Begriff	Informale Bedeutung
Attribut	Spalte einer Tabelle
Wertebereich	mögliche Werte eines Attributs (auch Domäne)
Attributwert	Element eines Wertebereichs
Relationenschema	Menge von Attributen
Relation	Menge von Zeilen einer Tabelle
Tupel	Zeile einer Tabelle
Datenbankschema	Menge von Relationenschemata
Datenbank	Menge von Relationen (Basisrelationen)



Zusammenfassung Grundbegriffe

Begriffe des Relationenmodells II

Begriff	Informale Bedeutung
Schlüssel	minimale Menge von Attributen, deren Werte ein Tupel einer Tabelle eindeutig identifizieren
Primärschlüssel	ein beim Datenbankentwurf ausgewählter Schlüssel
Fremdschlüssel	Attributmenge, die in einer anderen Relation Schlüssel ist
Fremdschlüsselbedingung	alle Attributwerte des Fremdschlüssels tauchen in der anderen Relation als Werte des Schlüssels auf



Normalisierung von Relationen



Normalisierung von Relationen

Ziele:

- einer Relation nur solche Attribute zuordnen, „die wirklich zu ihr gehören“,
- Anomalien vermeiden,
- Integrität durch geeignete Strukturen sichern
 - ◆ → Unnötige Redundanz vermeiden



Normalisierung von Relationen

Ziele: Vermeidung von “Anomalien“

- ◆ Lösch-Anomalie,
- ◆ Änderungs-Anomalie,
- ◆ Einfüge-Anomalie



Exkurs: Datenintegrität

- Was ist Integrität?
- Wie kann man die Datenintegrität sicherstellen?
 - ◆ Konstruktive Maßnahmen
 - ◆ Operative Maßnahmen



Exkurs: Redundanz

- Was ist Redundanz?
- Ist Redundanz schlecht?
- Wie kann man Redundanz vermeiden?



Normalisierung von Relationen

- Wir werden 3 Normalformen behandeln
(Es gibt mehr)
- ENF, ZNF, DNF
- „Funktionale Abhängigkeit“



Normalisierung

Unnormalisierte Relationen besitzen Attributwerte, die sich aus mehreren Elementen zusammensetzen.

Matr. Nr.	Name	Geb.-Datum	SR	SR-Name	Klausur	Klausur-Name	Note
4711	Klein	11.11.68	V	VWL	1, 4	BWL-A, VWL-B	2,3; 4,0
0815	Schulze	12.12.60	V	VWL	3	VWL-A	5,0
3307	Bauer	10.10.65	B	BWL	1, 3, 5	BWL-A, VWL-A, Jura PR	3,3; 5,0; 1,7
8801	Werner	30.11.67	W	Wipäd	2, 6	BWL-B, Jura ÖR	1,0; 1,7



Funktionale Abhängigkeit

x, y seien beliebige Wörter.

Es soll gelten:

$x\sigma y$

d.h. x und y stehen in einer zweistelligen Relation zueinander,

für jedes x existiert ein y

d.h. Nullwerte für y sollen ausgeschlossen sein

Dann ist der Begriff der Funktionalen Abhängigkeit gegeben durch die Implikation

$$(x\sigma y_1 \wedge x\sigma y_2) \rightarrow (y_1 = y_2)$$

d.h. wenn x bestimmt ist, dann ist auch y bestimmt.



volle Funktionale Abhängigkeit

C sei ein Attribut einer Relation R.

S_1 sei ein Schlüssel der Relation R,
der sich aus mehreren Attributen zusammensetzt.

Man sagt:

C ist voll funktional abhängig von S_1 ,
wenn es keine echte Untermenge S_0 von S_1 gibt,
so dass bereits
 $S_0 \rightarrow C$ gilt.



Transitive (funktionale) Abhängigkeit

wenn es Nichtschlüsselattribute gibt,
die von anderen Nichtschlüsselattributen
funktional abhängig sind

spricht man von einer

transitiven Abhängigkeit



ENF

Eine in der 1. Normalform befindliche Relation (auch 1NF oder ENF genannt) ist dadurch gekennzeichnet, dass sie keine Attribute mit Attributwerten aufweist, die sich aus mehreren Elementen zusammensetzen.

Matr. Nr.	Name	Geb.-Datum	SR	SR-Name	Klausur	Klausur-Name	Note
4711	Klein	11.11.68	V	VWL	1	BWL-A	2,3
4711	Klein	11.11.68	V	VWL	4	VWL-B	4,0
0815	Schulze	12.12.60	V	VWL	3	VWL-A	5,0
3307	Bauer	10.10.65	B	BWL	1	BWL-A	3,3
3307	Bauer	10.10.65	B	BWL	3	VWL-A	5,0
3307	Bauer	10.10.65	B	BWL	5	Jura PR	1,7
8801	Werner	30.11.67	W	Wipäd	2	BWL-B	1,0
8801	Werner	30.11.67	W	Wipäd	6	Jura ÖR	1,7



ZNF

Eine Relation ist in der 2. *Normalform* (auch 2NF oder ZNF genannt), wenn sie in der 1. Normalform vorliegt und jedes nicht dem Schlüssel angehörende Attribut voll funktional abhängig ist vom Schlüssel der Relation.

Matr. Nr.	Name	Geb.-Datum	SR	SR-Name
4711	Klein	11.11.68	V	VWL
0815	Schulze	12.12.60	V	VWL
3307	Bauer	10.10.65	B	BWL
8801	Werner	30.11.67	W	Wipäd

Klausur	Klausur-Name
1	BWL-A
2	BWL-B
3	VWL-A
4	VWL-B
5	Jura PR
6	Jura ÖR

Matr. Nr.	Klausur	Note
4711	1	2,3
4711	4	4,0
0815	3	5,0
3307	1	3,3
3307	3	5,0
3307	5	1,7
8801	2	1,0
8801	6	1,7



DNF

Eine Relation ist in der 3. Normalform (auch 3NF oder DNF genannt), wenn sie in der 2. Normalform vorliegt und keine transitiven Abhängigkeiten aufweist.

Matr. Nr.	Name	Geb.-Datum	SR
4711	Klein	11.11.68	V
0815	Schulze	12.12.60	V
3307	Bauer	10.10.65	B
8801	Werner	30.11.67	W

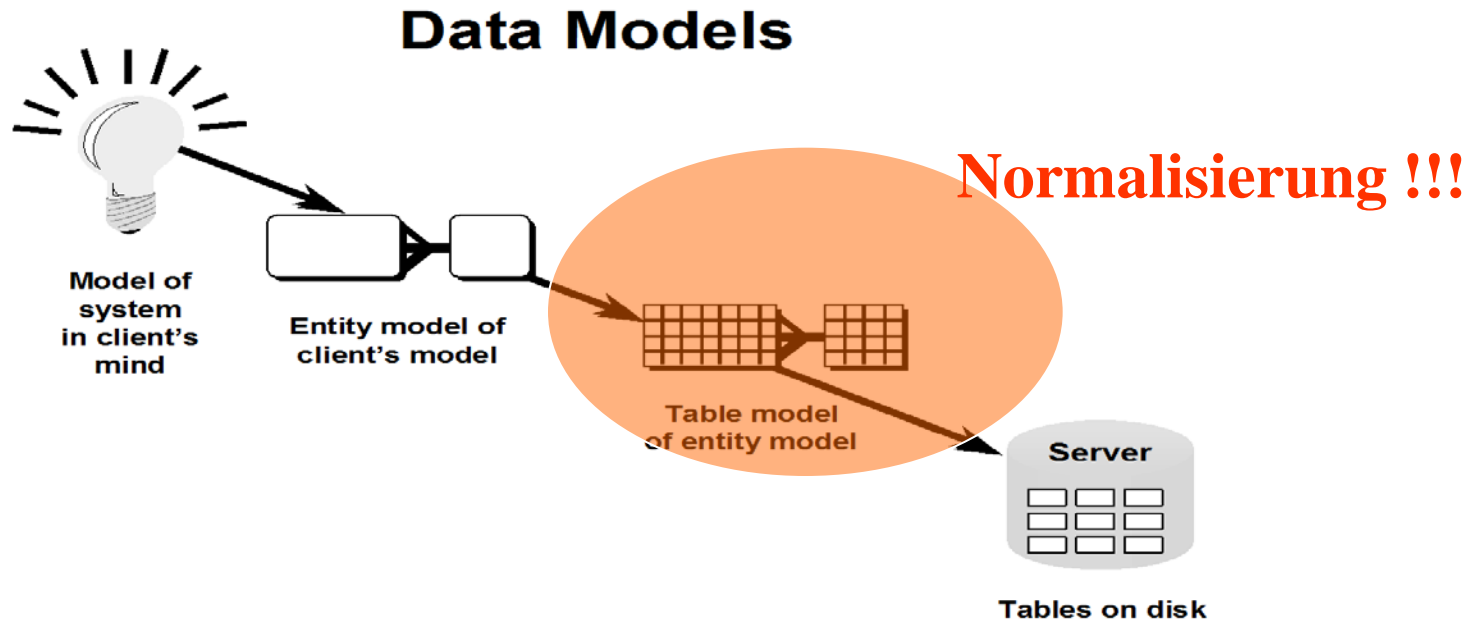
SR	SR-Name
B	BWL
V	VWL
W	Wipäd

Klausur	Klausur-Name
1	BWL-A
2	BWL-B
3	VWL-A
4	VWL-B
5	Jura PR
6	Jura ÖR

Matr. Nr.	Klausur	Note
4711	1	2,3
4711	4	4,0
0815	3	5,0
3307	1	3,3
3307	3	5,0
3307	5	1,7
8801	2	1,0
8801	6	1,7



An welcher Stelle im Entwurfsprozess wird die Normalisierung eingesetzt?



Zur Erinnerung!

Datenbanksysteme werden typischerweise

- im **Mehrbenutzerbetrieb**
- in **N-Tier-Architekturen** betrieben!

Daraus folgt für das **Datenbankdesign**:

*Es gilt, ein Datenmodell
für eine globale Sicht der Daten
herzustellen!!*

~~„Softwareentwicklung“ Design für EINE Anwendung!!!~~



Normalisierung: Verfahren

- Ziele
- Verfahrensschritte (s.u.)
- Transformation ERD \leftrightarrow Rel. Schema (s.u.)
- Anforderungen an das Ergebnis-Schema

Semantische Korrektheit

Vollständigkeit

Genügsamkeit

Redundanzfreiheit



Normalisierung: Verfahrensschritte

- (1) Dokumentiere die Attribute in Form eines Datenwörterbuchs.
- (2) Dokumentiere die Anforderungen aus Sicht der Anwender.
- (3) Setze die Anforderungen in Relationen um (Primärschl. festlegen!)
- (4) Fasse Relationen mit gleichen Primärschlüsselattributen zusammen.
- (5) Analysiere die funktionalen Abhängigkeiten.
- (6) Eliminiere nicht voll funktionale Abhängigkeiten.
- (7) Eliminiere ggf. auftretende transitive Abhängigkeiten.
- (8) Ggf. erneute Zusammenfassung von Relationen mit ident. Primärschlüsseln.



Normalisierungs-Beispiel

„Projektinformationssystem“



Normalisierungs-Beispiel

- L1: PNR, NACHNAME, GEB_DATUM, **TEL**
- L2: PNR, PROJNR, ANTEIL
- L3: PROJNR, PROJBEZ, PROJORT
- L4: PNR, ABTNR, ABTORT
- L5: ABTNR, ABTANZ
- L6: PNR, KIVNAME, KIALTER
- L7: ABTNR, MGRNR
- L8: LNR, PROJNR, LNAME, LORT
- L9: PROJNR, LNR, TNR, MENGE, LIBED, TBEZ, TEINHEIT, TFARBE, DATUM
- L10: PROJNR, TNR, LAGORT, PROJORT
- L11: TNR, TBEZ
- L12: TNR, TNR_UNTERTEIL, ANZAHL
- L13: PROJNR, PNR



Versuche, die erste Liste (L1) zu normalisieren!

Mitarbeiter(PNR, NACHNAME, ~~ALTER~~, GEB_DATUM, **TEL?**)

Exemplar-Ebene:

(4711, Müller, 13.03.1871, (2345, 2346,))

(4711, Festnetz-1, 2345)

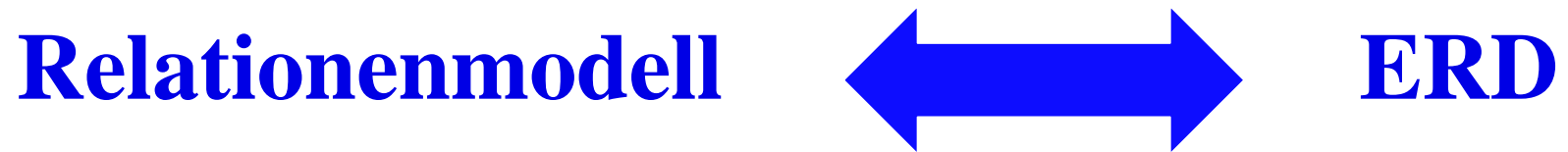
(4711, Festnetz-2, 2346)

....

Mitarbeiter(PNR, NACHNAME, GEB_DATUM, TEL-2, TEL-2, ...)

Mitarbeiter(PNR, NACHNAME, GEB_DATUM, TELEFONANSCHLÜSSE *string*)

Transformationen



Transformationen



In der Praxis wird hiervon gerne Gebrauch gemacht, da das ERD besonders anschaulich und damit geeignet ist, um die Datenstrukturen mit Anwendern zu diskutieren, die keine Datenbankexperten sind.

Andererseits ist die Relationendarstellung wesentlich kompakter und sehr nahe an der Tabellendarstellung, die von relationalen DBMS zur Umsetzung angeboten wird.



Transformationen Relationenmodell ERD

Regeln siehe Skript !

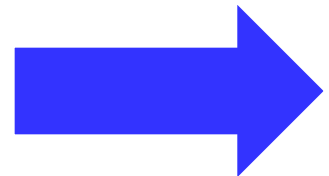
Beispiel: **STUDENT**(Matr.Nr., Name, Geb.-Datum, SR)
KLAUSUR(Klausur, Klausur-Name)
KLAUSURERGEBNIS(Matr.Nr., Klausur, Note)
STUDIENRICHTUNG(SR, SR-Name)

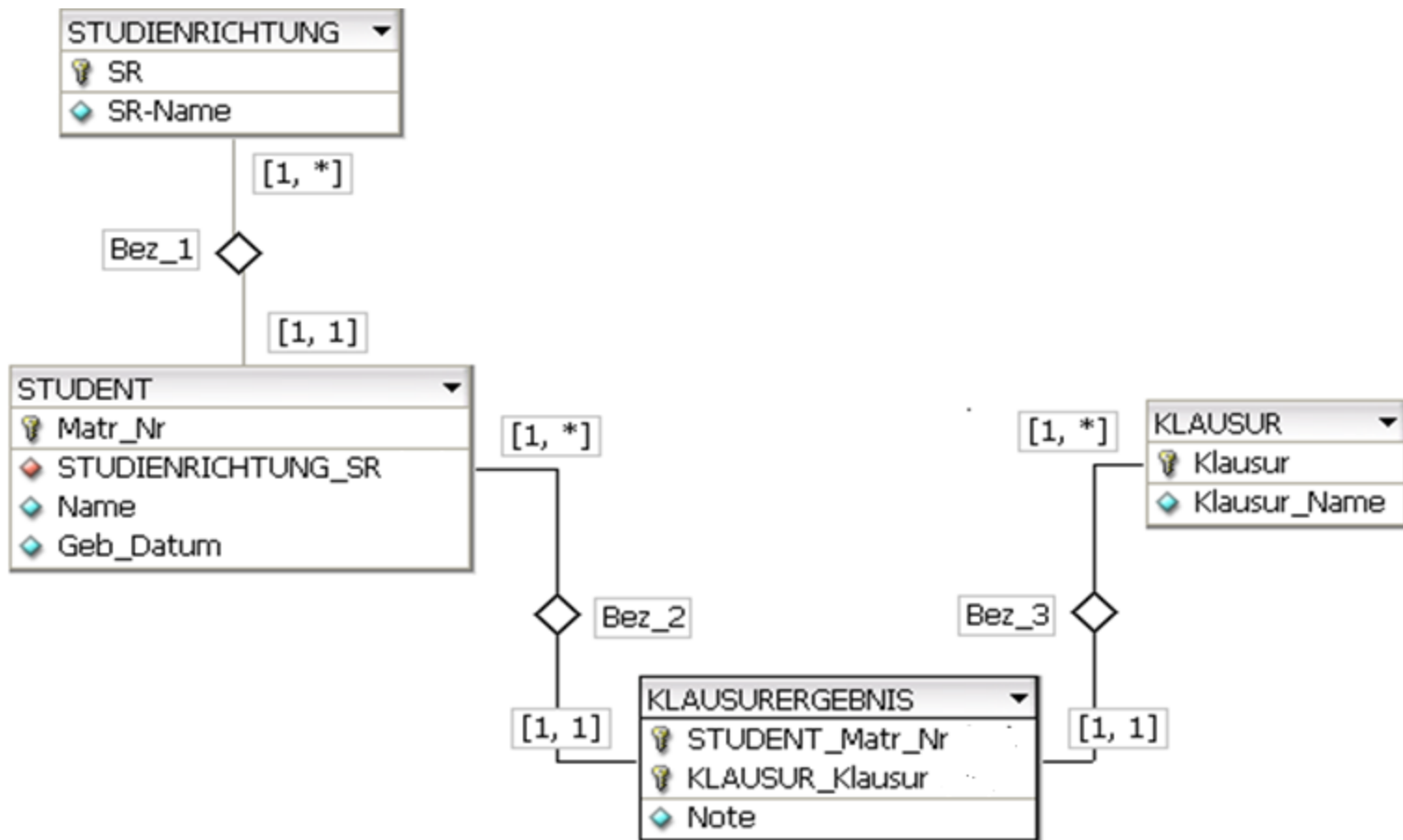
Matr. Nr.	Name	Geb.-Datum	SR
4711	Klein	11.11.68	V
0815	Schulze	12.12.60	V
3307	Bauer	10.10.65	B
8801	Werner	30.11.67	W

SR	SR-Name
B	BWL
V	VWL
W	Wipäd

Klausur	Klausur-Name
1	BWL-A
2	BWL-B
3	VWL-A
4	VWL-B
5	Jura PR
6	Jura ÖR

Matr. Nr.	Klausur	Note
4711	1	2,3
4711	4	4,0
0815	3	5,0
3307	1	3,3
3307	3	5,0
3307	5	1,7
8801	2	1,0
8801	6	1,7





Nicht-normalisierte Datenbanken



Motivation für die Einrichtung nicht-normalisierter Datenbanken (Beispiele)

- **Datenbestand wird einmalig erstellt und ändert sich nicht mehr (z.B. Archiv)**
- **Verbesserung des Antwortzeitverhaltens bei Abfragen (z.B. Data Warehouse, s.u.)**
- **Erhöhung der Sicherheit (z.B. Replikation)**
- **Gesetzliche oder andere Vorschriften (z.B. doppelte Buchführung)**
- **Organisatorische Notwendigkeiten (z.B. Druckdateien)**



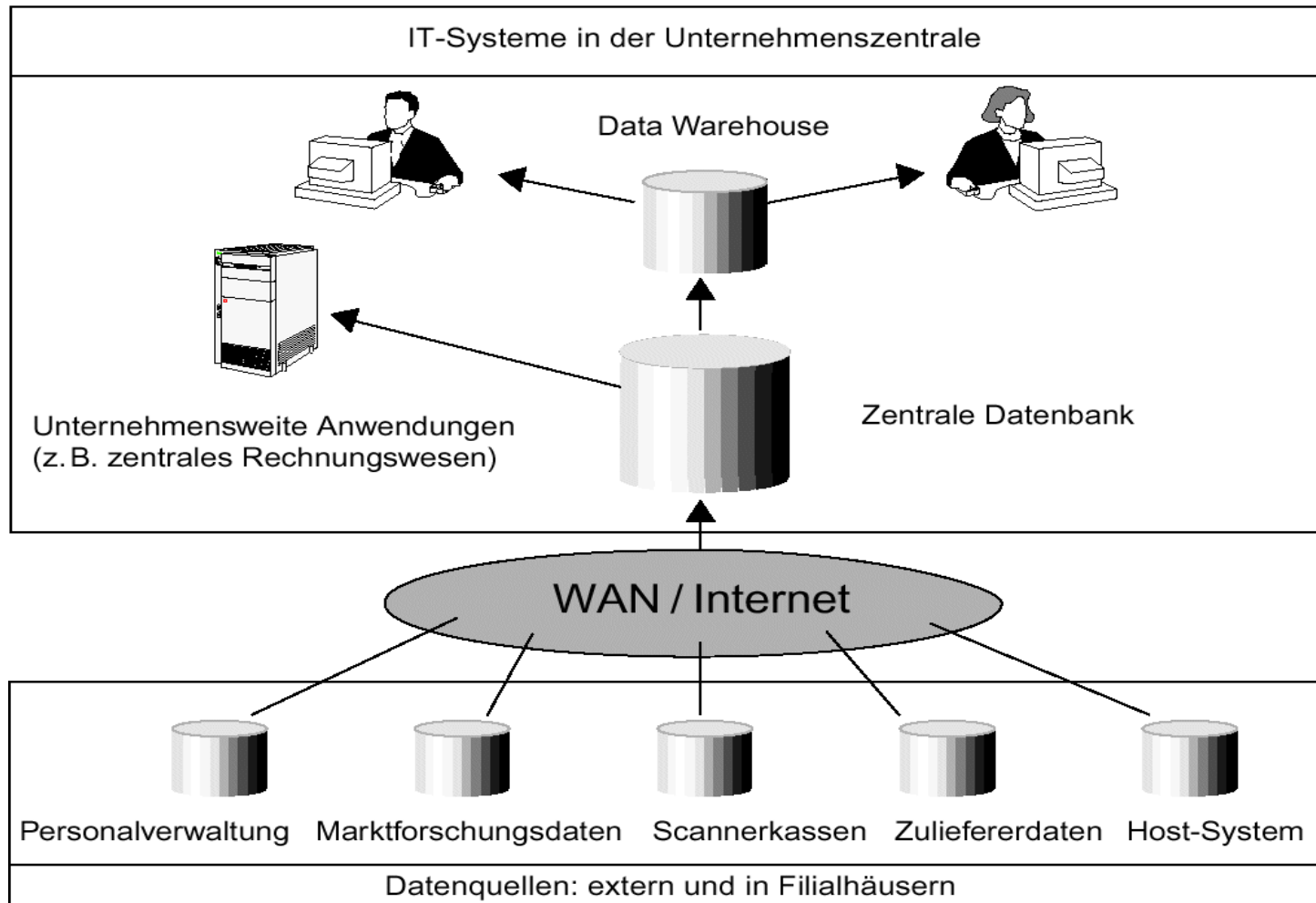
~~Warehouse = Warenhaus !~~



Die wesentlichen Merkmale eines Data Warehouse:

- **Fachorientierung**
- **Integrierte Datenbasis**
- **Stabile Datenbasis**
- **Historische Datenbasis**
- **Sehr große Datenmengen**



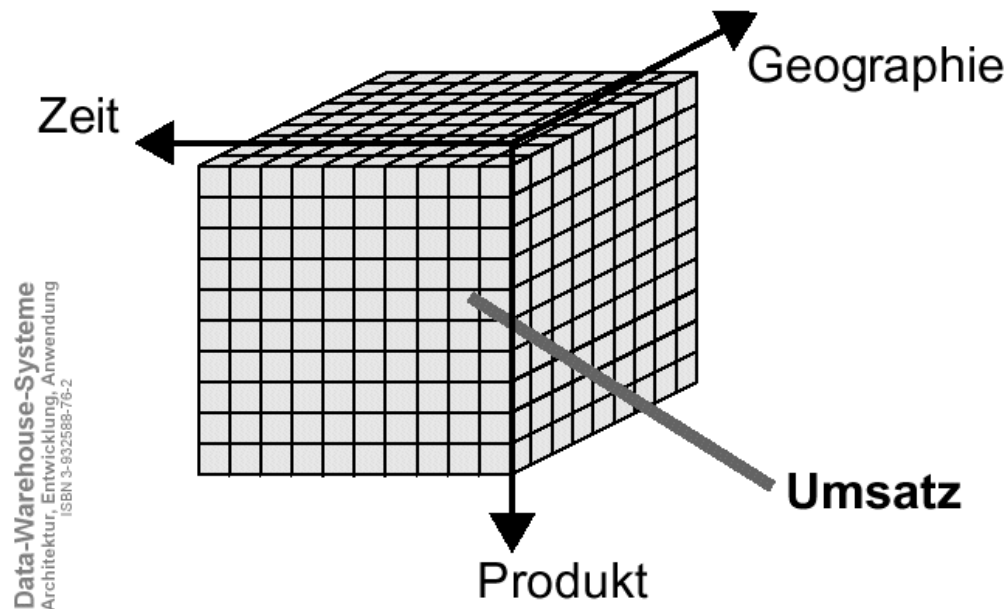


Data-Warehouse-Systeme; Architektur, Entwicklung, Anwendung; ISBN 3-932588-76-2



Dimension

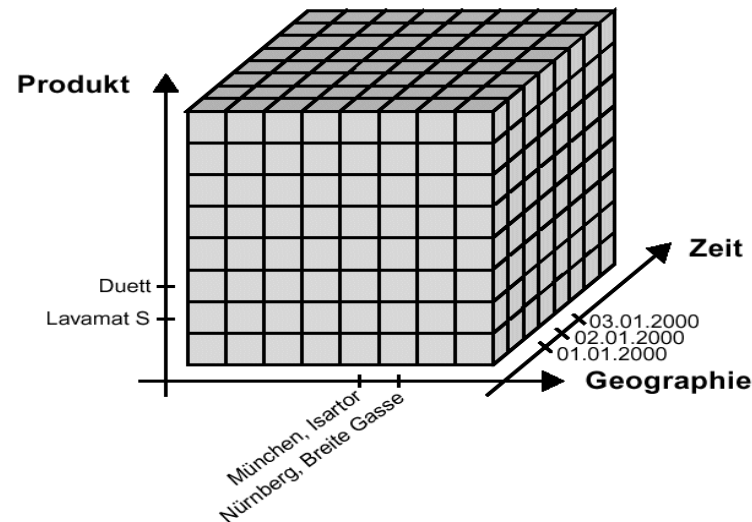
- Eine Dimension ist eine Menge von mindestens zwei Dimensionselementen (Wertausprägungen), nach denen die Daten beliebig angeordnet werden können.
- Sie dient der eindeutigen, orthogonalen Strukturierung des Datenraums und kann zur Klassifikation der betrachteten Kenngrößen (hier: des Umsatzes) verwendet werden.



Data Warehouse: ROLAP → Relationales OLAP

Dualismus von Würfel und Tabelle

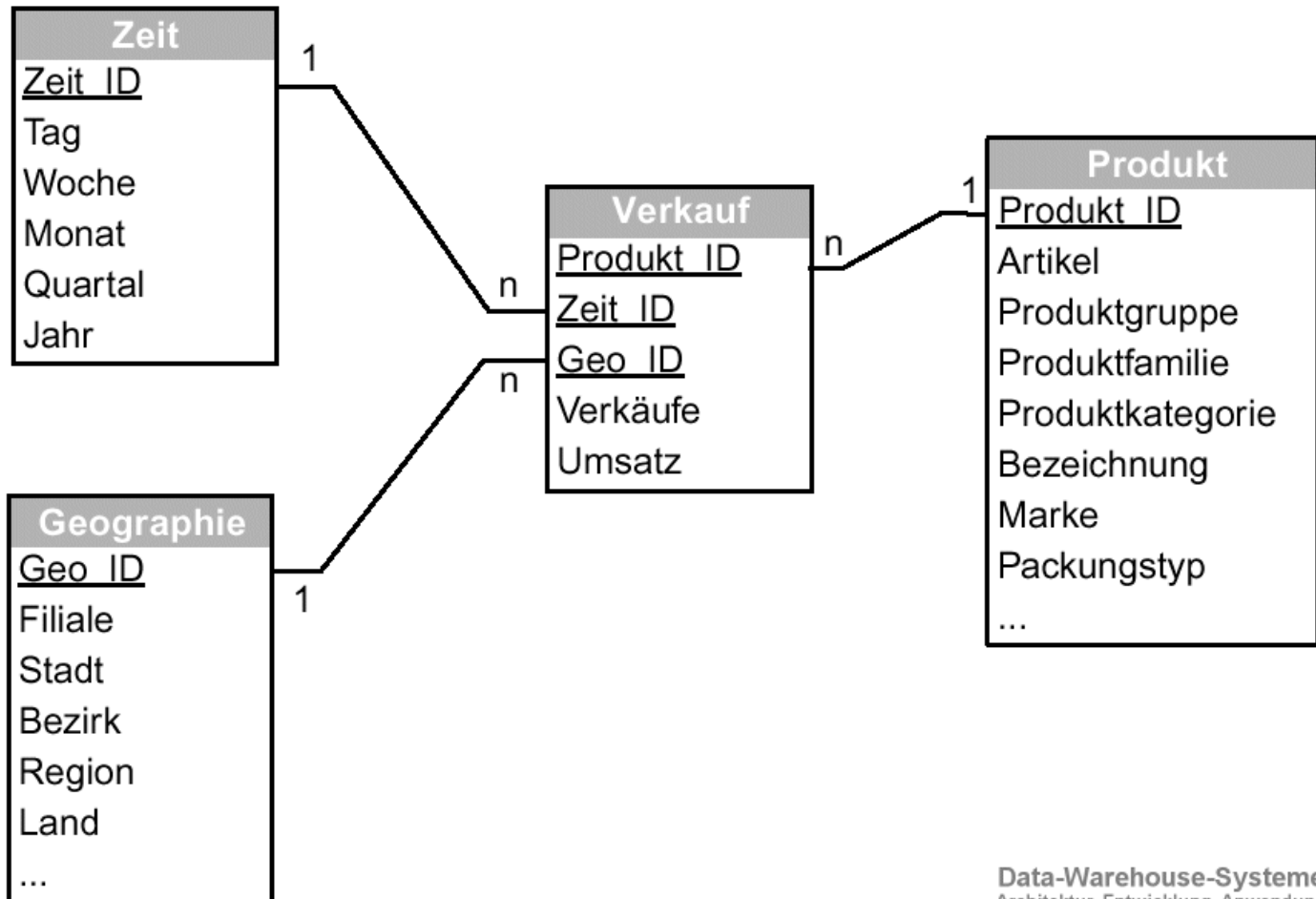
Data-Warehouse-Systeme; Architektur, Entwicklung, Anwendung; ISBN 3-932988-76-2



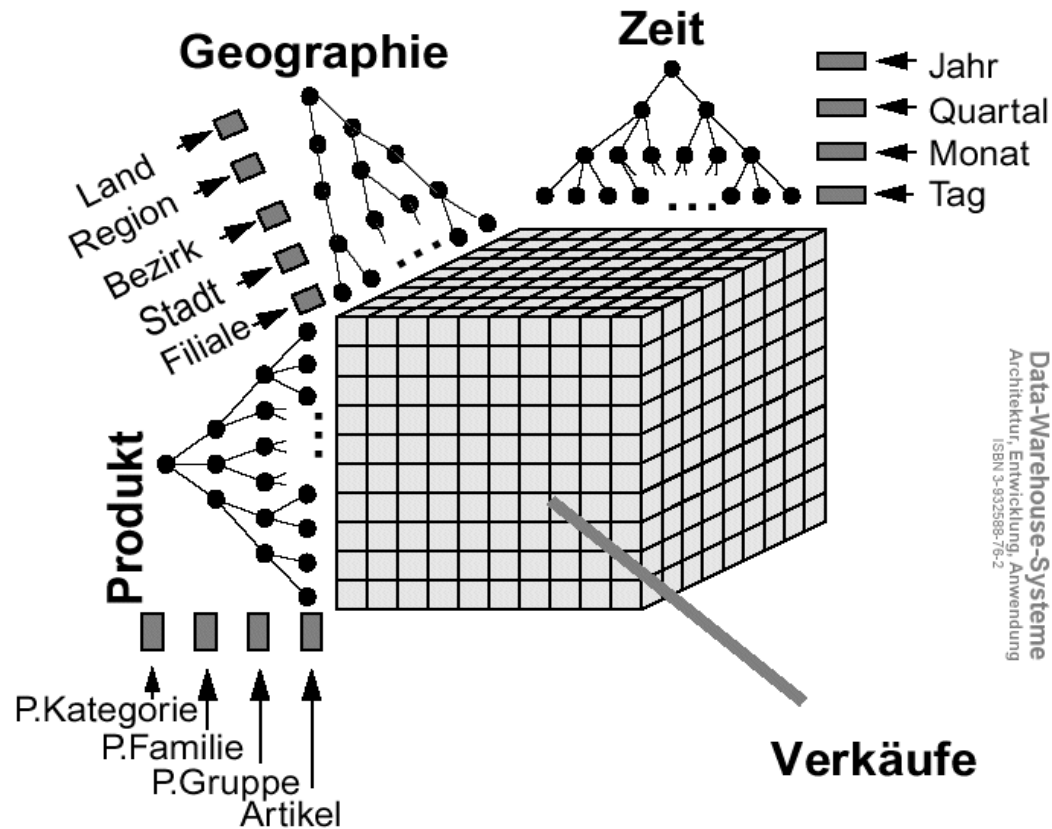
Artikel	Filiale	Tag	Verkäufe
Duett	Nürnberg, Breite Gasse	03.01.2000	7
Duett	München, Isartor	03.01.2000	3
Lavamat S	München, Isartor	03.01.2000	2

- Spalten (Attribute) ARTIKEL, FILIALE und TAG werden als Dimensionen interpretiert.
- Die Spalte VERKÄUFE wird als Kenngröße interpretiert.
- Man nennt eine solche Tabelle Faktentabelle.





Multidimensionales Datenmodell mit Dimensionen und Klassifikationshierarchien:



ROLAP: Spezielle Datenbank-Schemata für Data Warehouse Anwendungen



Snowflake-Schema

und



Star-Schema

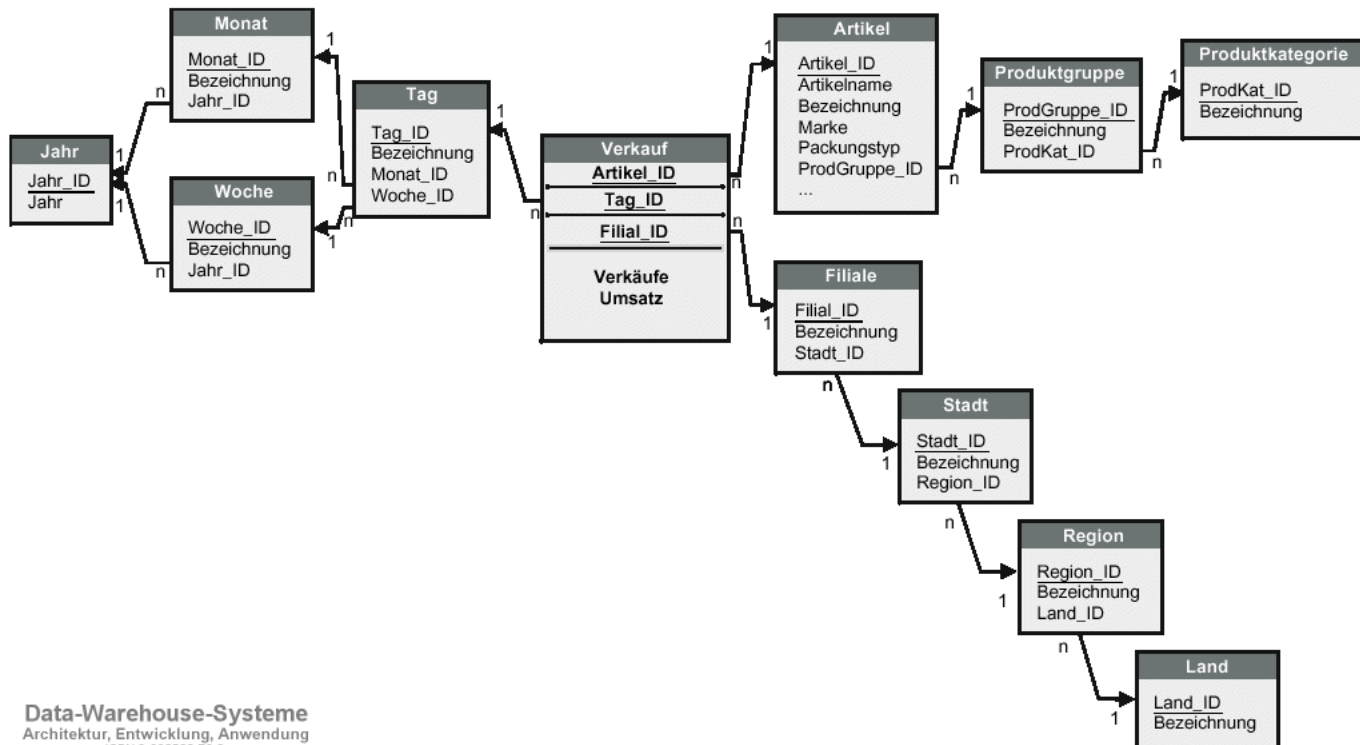


ROLAP: Datenbank-Schema Typ I:

Snowflake-Schema



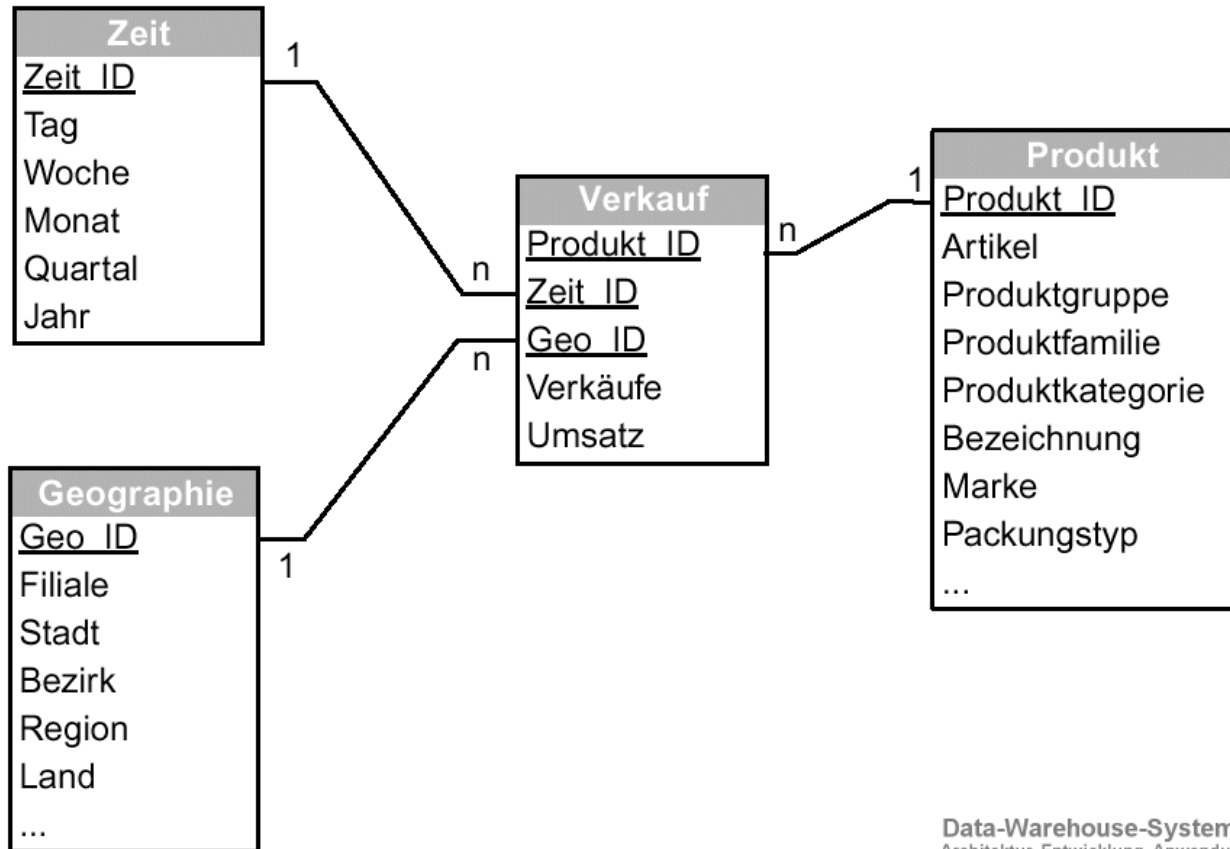
Einrichtung einer eigenen Tabelle für jede Klassifikationsstufe der Klassifikationshierarchie



ROLAP: Datenbank-Schema Typ II: Star-Schema



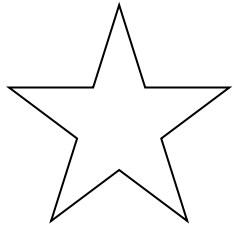
Beispiel



Data-Warehouse-Systeme
Architektur, Entwicklung, Anwendung
ISBN 3-932588-76-2

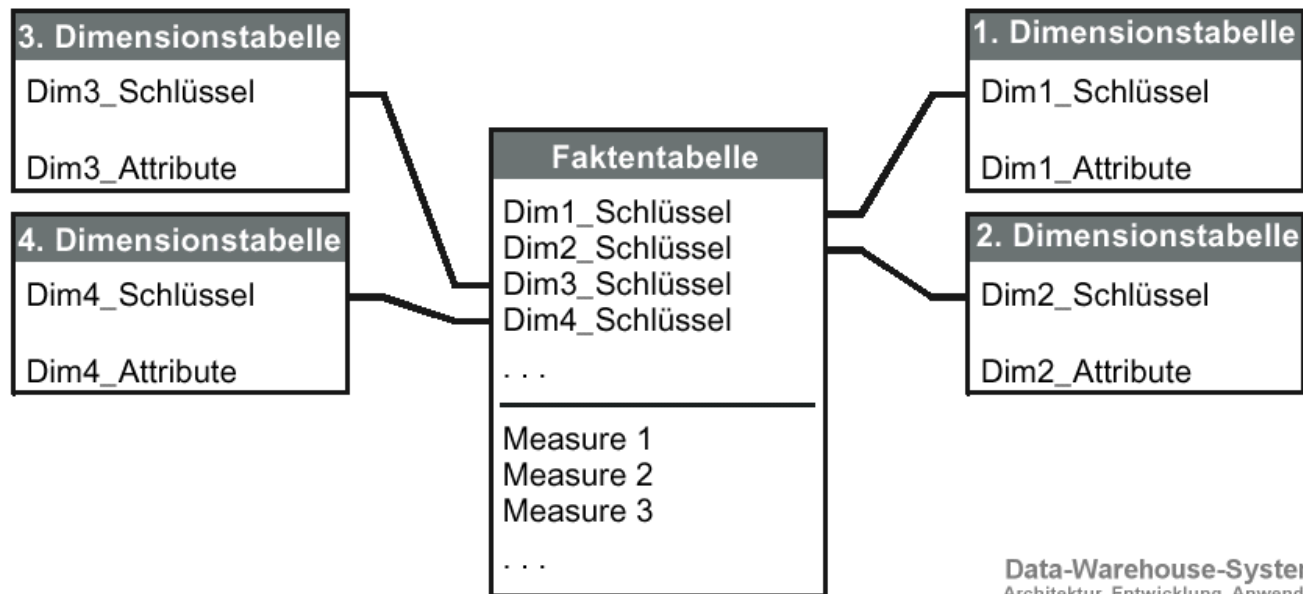


ROLAP: Datenbank-Schema Typ II:



Star-Schema

Die gesamte Klassifikationshierarchie einer Dimension wird in jeweils einer einzigen Tabelle zusammengefasst



Data-Warehouse-Systeme
Architektur, Entwicklung, Anwendung
ISBN 3-932588-76-2



ROLAP

Nachteile des Star-Schemas:

- Redundanzen
- Klassifikations-Abhängigkeiten nicht sichtbar

Vorteile des Star-Schemas:

- einfache Struktur
- einfache Darstellung von Klassifikations-Hierarchien
- effiziente Anfrageverarbeitung innerhalb der Dimensionen



ROLAP

Relationale Abfrageoperationen

- **GROUP BY** einfache Gruppierung
- **CUBE** interdimensional
- **ROLLUP** intradimensional



Zusammenfassung (1)

- Relationenmodell
- Schlüssel
- Normalisierung
 - ◆ ENF
 - ◆ ZNF
 - ◆ DNF
- Geregelttes Verfahren
- Denormalisierung



Zusammenfassung (2)

- **Relationenmodell** als Modellbaukasten
- **Relationes Modell** als Ergebnis eines Entwurfsprozesses
- **Relationales Datenbankmanagementsystem (RDBMS)** als Softwarekiste
- **Relationale Datenbank (RDB)** als Datenspeicher
- **Relationales Datenbanksystem (RDBS)**
als RDBMS+RDB



Zusammenfassung (3)

- Es gibt auch Anwendungen, für die eine Normalisierung ungünstig ist
- Data Warehouse: Datenbank für OLAP
- ROLAP: Relationales OLAP
 - ◆ Snowflake- und Star-Schema
 - ◆ CUBE und ROLLUP

