

Data Warehouses und Moderne Betriebliche Anwendungen von Datenbanksystemen

(Folien von A. Kemper zum Buch 'Datenbanksysteme')

Online Transaction Processing

**Betriebswirtschaftliche Standard-
Software (SAP R/3)**

Data Warehouse-Anwendungen

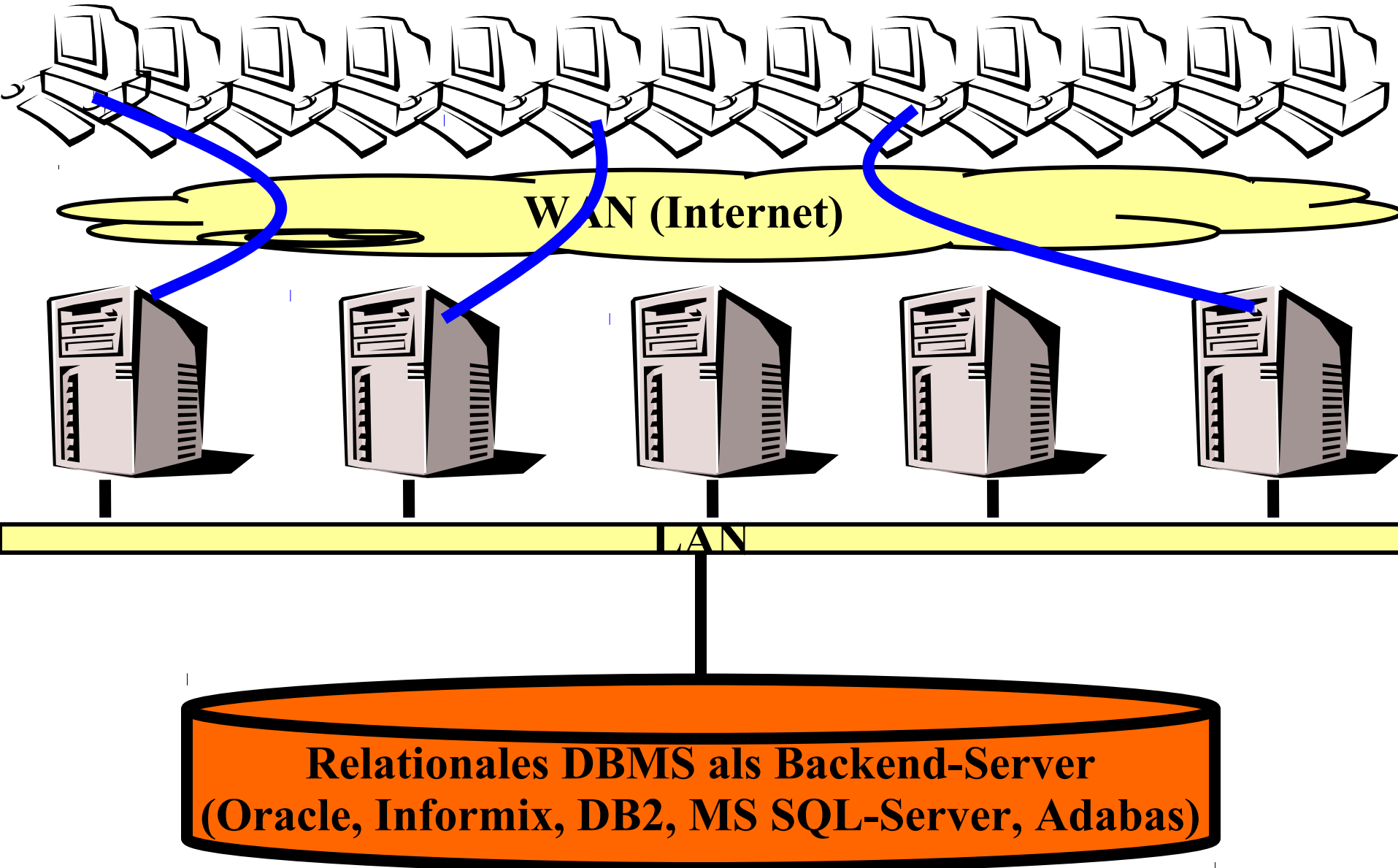
Data Mining

OLTP: Online Transaction

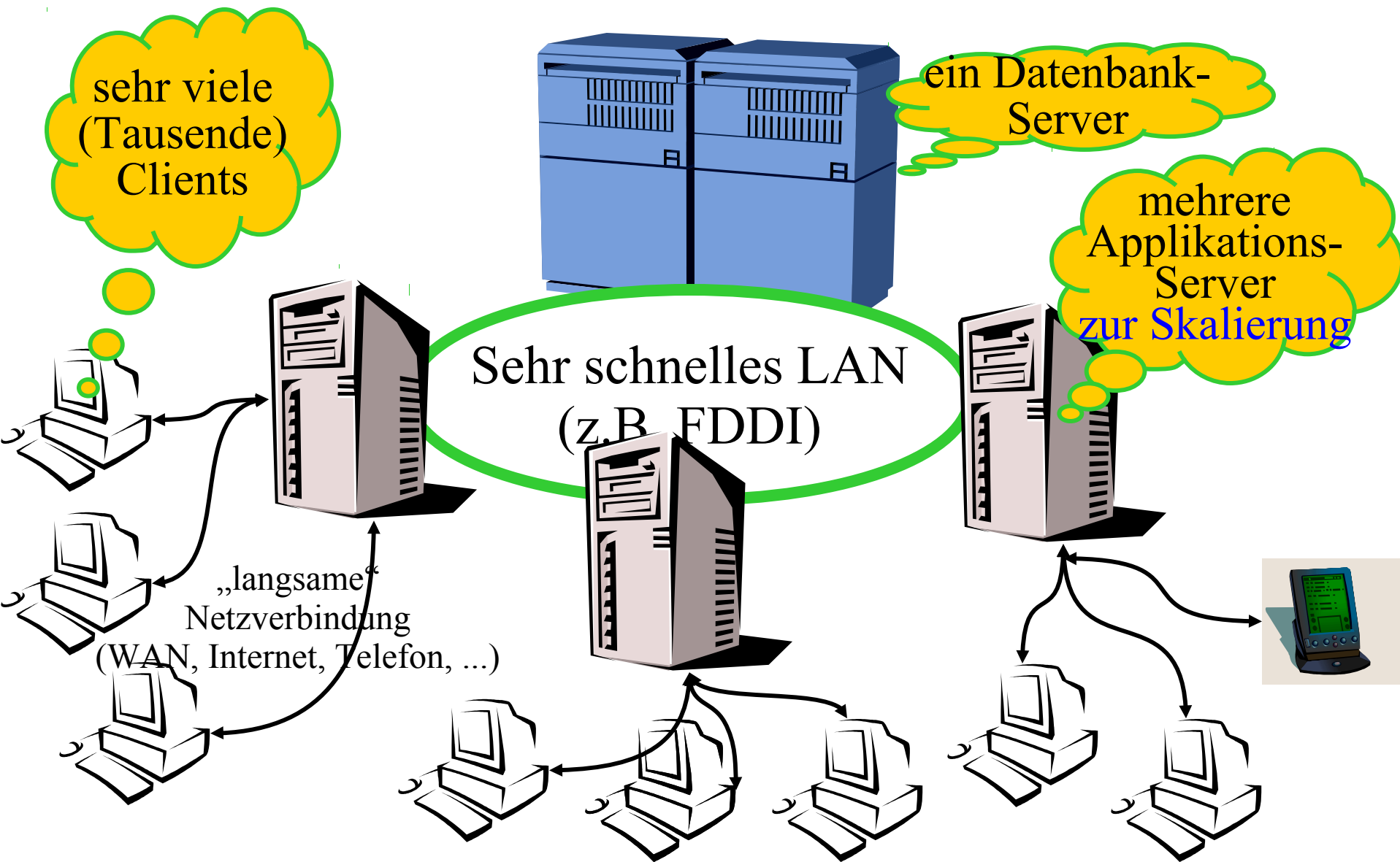
Processing

- Beispiele
 - Flugbuchungssystem
 - Bestellungen in einem Handelsunternehmen
- Charakterisierung
 - Hoher Parallelitätsgrad
 - Viele (Tausende pro Sekunde) kurze Transaktionen
 - TAs bearbeiten nur ein kleines Datenvolumen
 - „mission-critical“ für das Unternehmen
 - Hohe Verfügbarkeit muss gewährleistet sein
- Normalisierte Relationen (möglichst wenig Update-Kosten)
- Nur wenige Indexe (wegen Fortschreibungskosten)

SAP R/3: Enterprise Resource Modelling (ERP-System)



Dreistufige Client/Server-Architektur (3 Tier, SAP R/3)



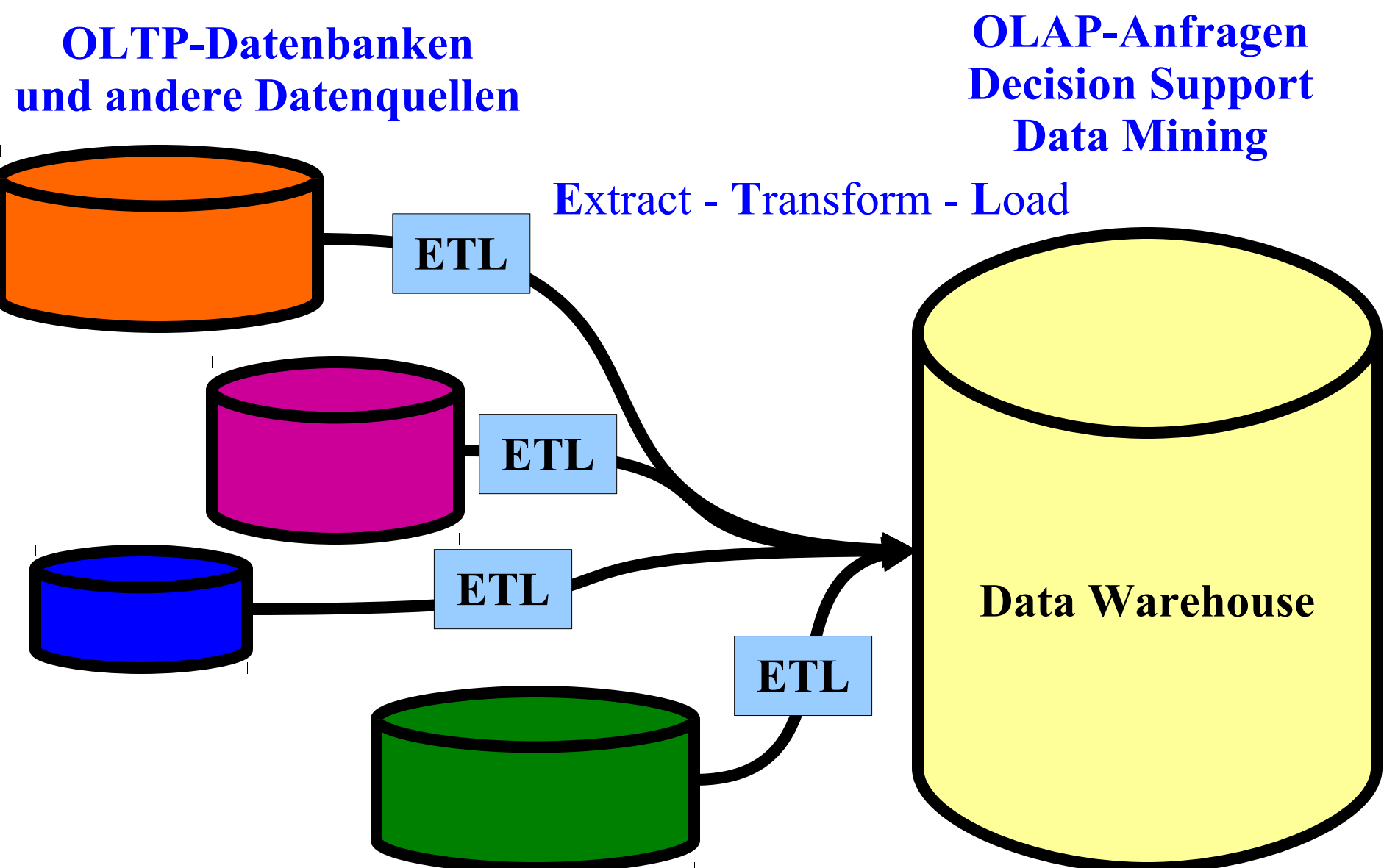
Data Warehouse-Anwendungen: OLAP~Online Analytical Processing

- Wie hat sich die Auslastung der Transatlantikflüge über die letzten zwei Jahre entwickelt?

oder

- Wie haben sich besondere offensive Marketingstrategien für bestimmte Produktlinien auf die Verkaufszahlen ausgewirkt?

Sammlung und periodische Auffrischung der Data Warehouse-Daten



Das Stern-Schema

Verkäufe					
VerkDatum	Filiale	Produkt	Anzahl	Kunde	Verkäufer
25-Jul-00	Passau	1347	1	4711	825
...

Filialen			
Filialenkennung	Land	Bezirk	...
Passau	D	Bayern	...
...

Kunden			
KundenNr	Name	wiealt	...
4711	Kemper	43	...
...

Verkäufer					
VerkäuferNr	Name	Fachgebiet	Manager	wiealt	...
825	Handyman	Elektronik	119	23	...
...

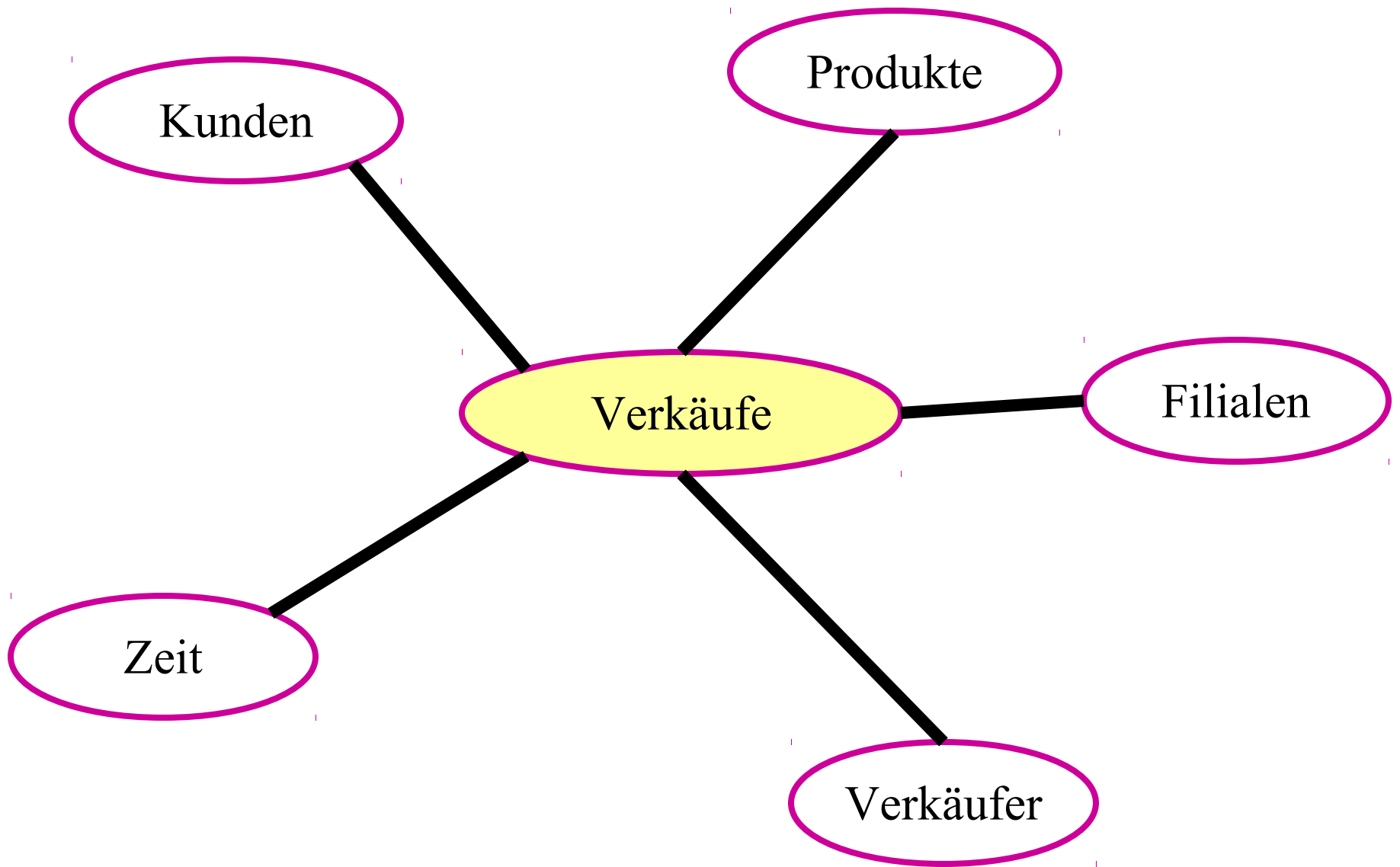
Zeit								
Datum	Tag	Monat	Jahr	Quartal	KW	Wochentag	Saison	...
...
25-Jul-00	25	Juli	2000	3	30	Dienstag	Hochsommer	...
...
18-Dec-01	18	Dezember	2001	4	52	Dienstag	Weihnachten	...
...

Produkte					
ProduktNr	Produkttyp	Produktgruppe	Produkthauptgruppe	Hersteller	...
1347	Handy	Mobiltelekom	Telekom	Siemens	...
...

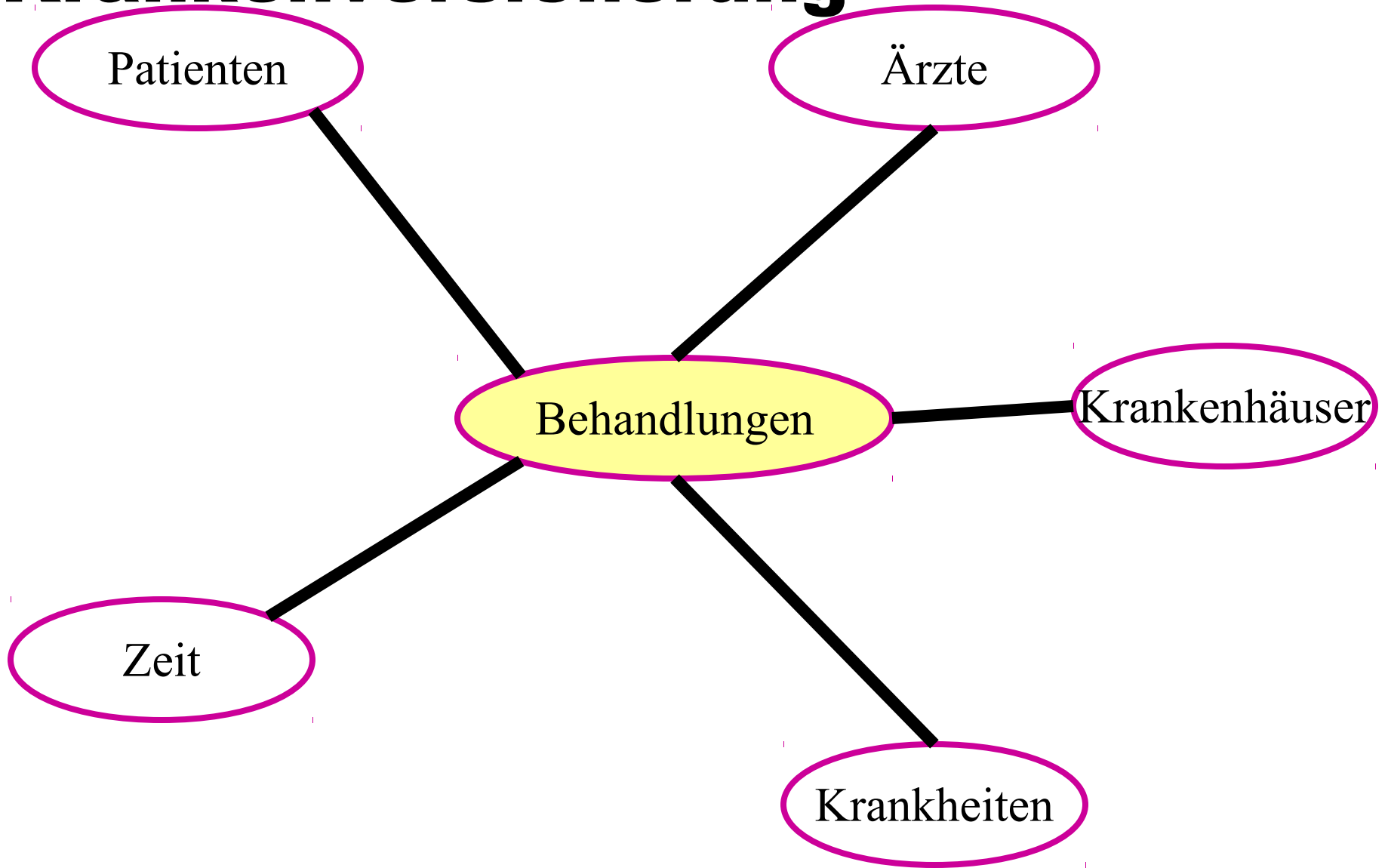
Stern-Schema bei Data Warehouse- Anwendungen

- Eine sehr große Faktentabelle
 - Alle Verkäufe der letzten drei Jahre
 - Alle Telefonate des letzten Jahres
 - Alle Flugreservierungen der letzten fünf Jahre
 - normalisiert
- Mehrere Dimensionstabellen
 - Zeit
 - Filialen
 - Kunden
 - Produkt
 - Oft nicht normalisiert

Das Stern-Schema: Handelsunternehmen



Das Stern-Schema: Krankenversicherung



Stern-Schema

Verkäufe					
VerkDatum	Filiale	Produkt	Anzahl	Kunde	Verkäufer
25-Jul-00	Passau	1347	1	4711	825
...

Faktentabelle (SEHR groß)

Filialen			
FilialenKennung	Land	Bezirk	...
Passau	D	Bayern	...
...

Kunden			
KundenNr	Name	wieAlt	...
4711	Kemper	43	...
...

Dimensionstabellen (relativ klein)

Verkäufer					
VerkäuferNr	Name	Fachgebiet	Manager	wieAlt	...
825	Handyman	Elektronik	119	23	...
...

Stern-Schema (cont'd)

Zeit							
Datum	Tag	Monat	Jahr	Quartal	KW	Wochentag	Saison
25-Jul-00	25	7	2000	3	30	Dienstag	Hochsommer
...		
18-Dec-01	18	12	2001	4	52	Dienstag	Weihnachten
...

Produkte					
ProduktNr	Produkttyp	Produktgruppe	Produkthauptgruppe	Hersteller	..
1347	Handy	Mobiltelekom	Telekom	Siemens	..
...

Nicht-normalisierte Dimensionstabellen: effizientere Anfrageauswertung

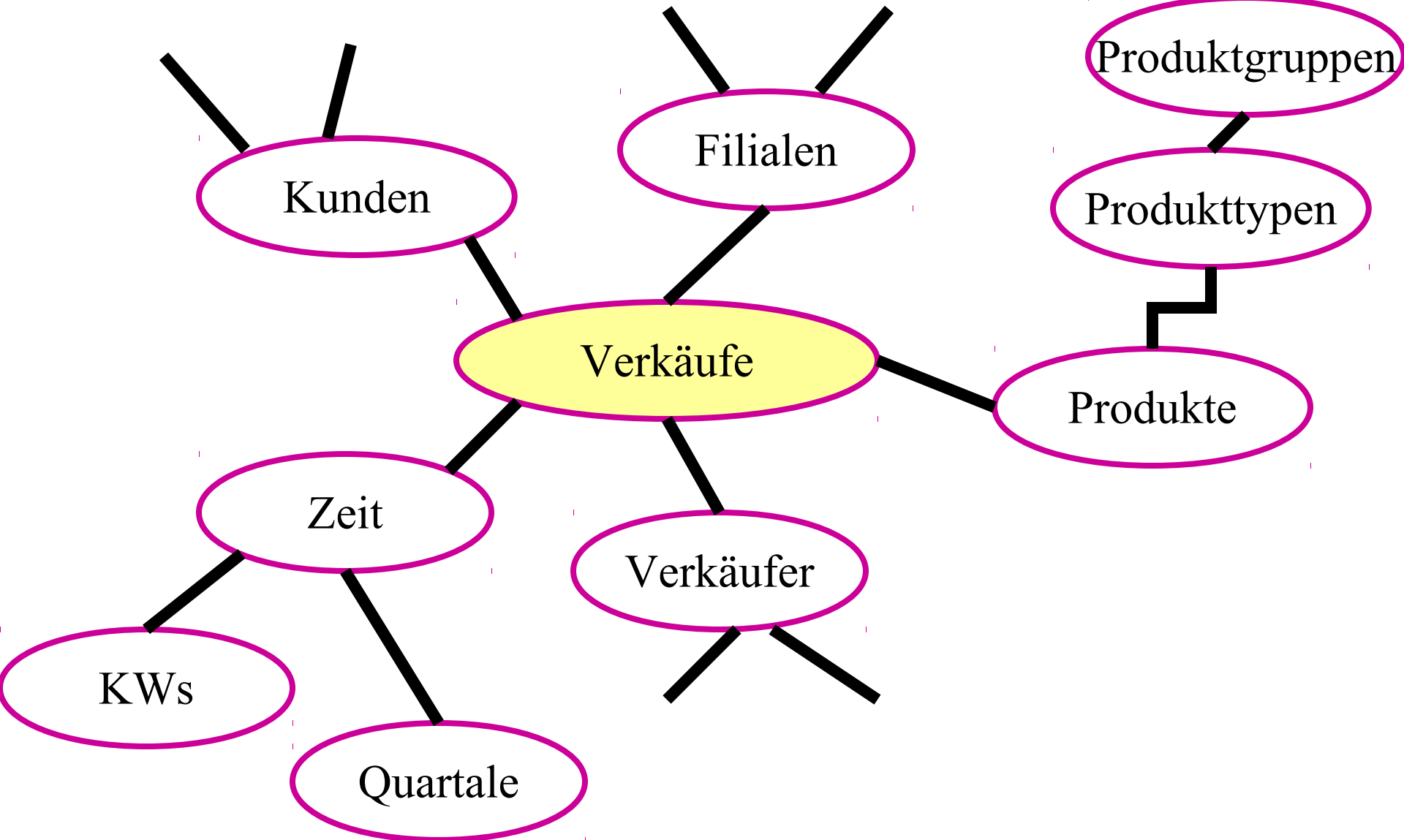
Zeit							
Datum	Tag	Monat	Jahr	Quartal	KW	Wochentag	Saison
25-Jul-00	25	7	2000	3	30	Dienstag	Hochsommer
...		
18-Dec-01	18	12	2001	4	52	Dienstag	Weihnachten
...

Datum → Monat → Quartal

Produkte					
ProduktNr	Produkttyp	Produktgruppe	Produkthauptgruppe	Hersteller	..
1347	Handy	Mobiltelekom	Telekom	Siemens	..
...

ProduktNr → Produkttyp → Produktgruppe → Produkthauptgruppe

Normalisierung führt zum Schneeflocken-Schema



Anfragen im Sternschema

select sum(v.Anzahl), p.Hersteller

from Verkäufe v, Filialen f, Produkte p, Zeit z, Kunden k

where z.Saison = 'Weihnachten' and

z.Jahr = 2001 and k.wieAlt < 30 and

p.Produkttyp = 'Handy' and f.Bezirk = 'Bayern' and

v.VerkDatum = z.Datum and v.Produkt = p.ProduktNr and

v.Filiale = f.FilialenKennung and v.Kunde = k.KundenNr

group by p.Hersteller;

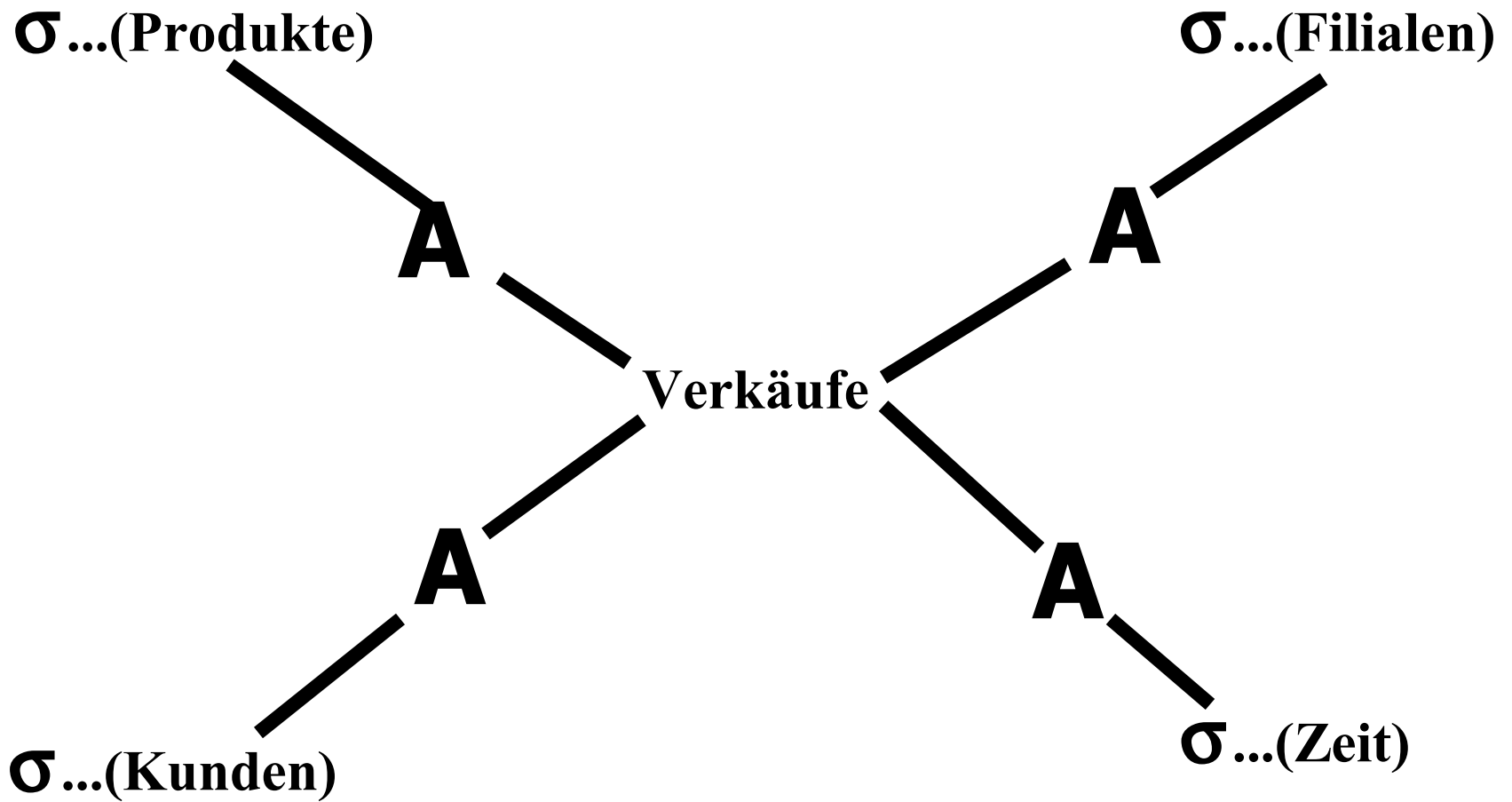


Einschränkung
der Dimensionen



Join-Prädikate

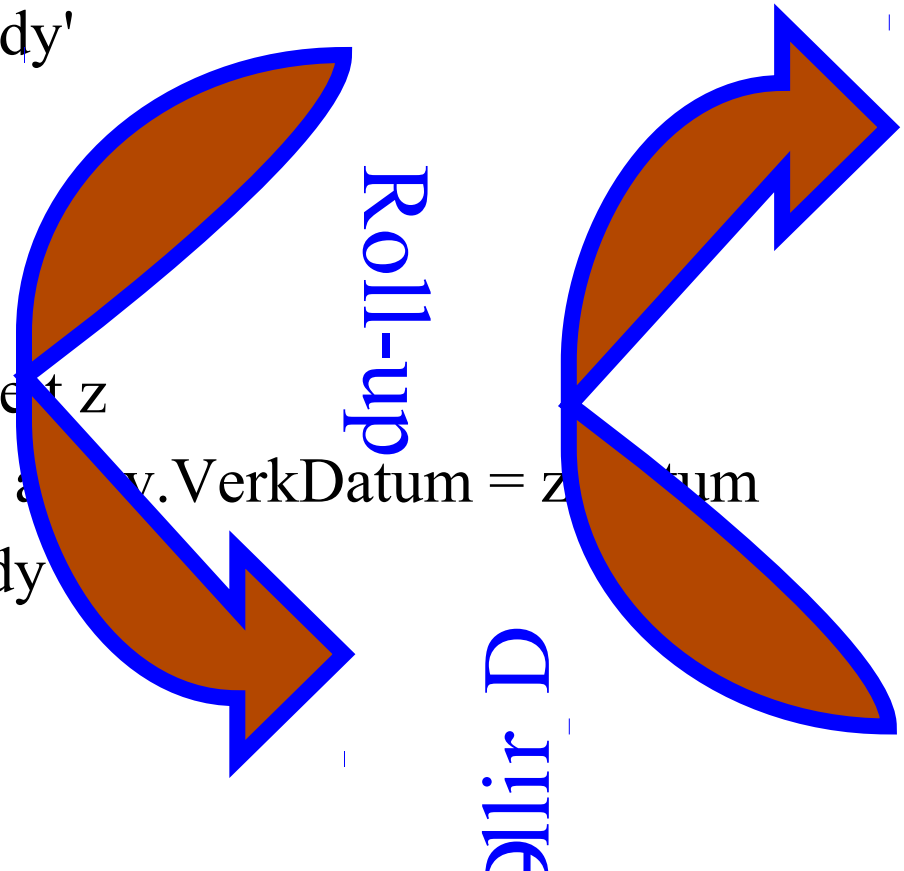
Algebra-Ausdruck



Roll-up/Drill-down-Anfragen

```
select Jahr, Hersteller, sum(Anzahl)
from Verkäufe v, Produkte p, Zeit z
where v.Produkt = p.ProduktNr and v.VerkDatum = z.Datum
      and p.Produkttyp = 'Handy'
group by p.Hersteller, z.Jahr;
```

```
select Jahr, sum(Anzahl)
from Verkäufe v, Produkte p, Zeit z
where v.Produkt = p.ProduktNr and v.VerkDatum = z.Datum
      and p.Produkttyp = 'Handy'
group by z.Jahr;
```



Ultimative Verdichtung

select sum(Anzahl)

from Verkäufe v, Produkte p

where v.Produkt = p.ProduktNr and p.Produkttyp = 'Handy';

Handyverkäufe nach Hersteller und Jahr		
Hersteller	Jahr	Anzahl
Siemens	1999	2.000
Siemens	2000	3.000
Siemens	2001	3.500
Motorola	1999	1.000
Motorola	2000	1.000
Motorola	2001	1.500
Bosch	1999	500
Bosch	2000	1.000
Bosch	2001	1.500
Nokia	1999	1.000
Nokia	2000	1.500
Nokia	2001	2.000

Handyverkäufe nach Jahr	
Jahr	Anzahl
1999	4.500
2000	6.500
2001	8.500

Handyverkäufe nach Hersteller	
Hersteller	Anzahl
Siemens	8.500
Motorola	3.500
Bosch	3.000
Nokia	4.500

Abb. 17.7: Analyse der Handyverkaufszahlen nach unterschiedlichen Dimensionen

Hersteller \ Jahr	1999	2000	2001	Σ
Siemens	2.000	3.000	3.500	8.500
Motorola	1.000	1.000	1.500	3.500
Bosch	500	1.000	1.500	3.000
Nokia	1.000	1.500	2.000	4.500
Σ	4.500	6.500	8.500	19.500

Abb. 17.8: Handyverkäufe nach Jahr und Hersteller

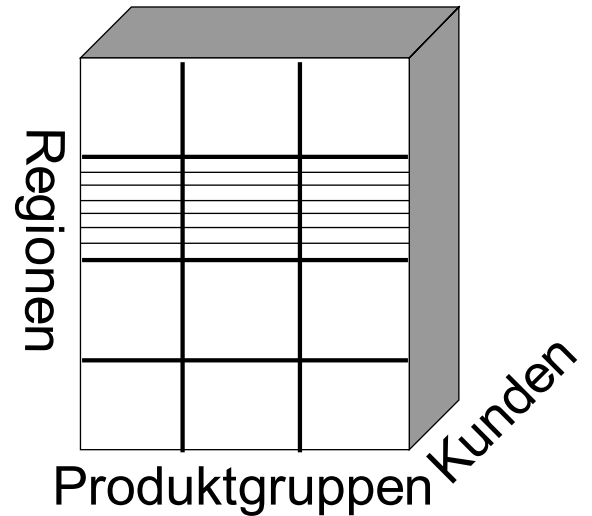
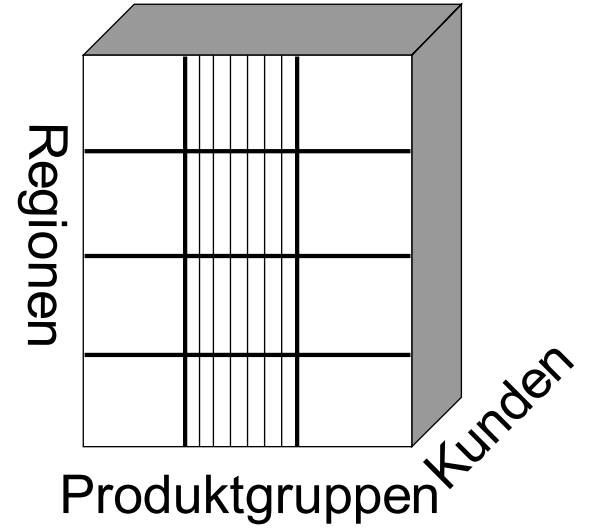
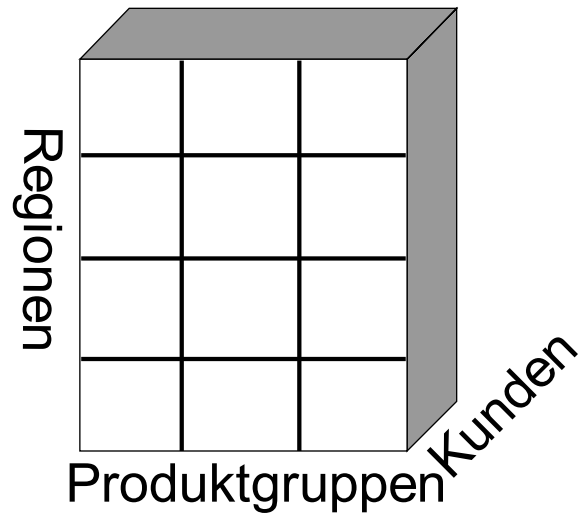
Handyverkäufe nach Hersteller und Jahr			Handyverkäufe nach Jahr	
Hersteller	Jahr	Anzahl	Jahr	Anzahl
Siemens	1999	2.000	1999	4.500
Siemens	2000	3.000	2000	6.500
Siemens	2001	3.500	2001	8.500
Motorola	1999	1.000		
Motorola	2000	1.000		
Motorola	2001	1.500		
Bosch	1999	500		
Bosch	2000	1.000		
Bosch	2001	1.500		
Nokia	1999	1.000		
Nokia	2000	1.500		
Nokia	2001	2.000		
			Hersteller	Anzahl
			Siemens	8.500
			Motorola	3.500
			Bosch	3.000
			Nokia	4.500

Abb. 17.7: Handyverkäufe nach Hersteller und Jahr

Hersteller \ Jahr	1999	2000	2001	Σ
Siemens	2.000	3.000	3.500	8.500
Motorola	1.000	1.000	1.500	3.500
Bosch	500	1.000	1.500	3.000
Nokia	1.000	1.500	2.000	4.500
Σ	4.500	6.500	8.500	19.500

Abb. 17.8: Handyverkäufe nach Jahr und Hersteller

Flexible Auswertungsmethoden: slice and dice



Materialisierung von

Aggregaten

insert into Handy2DCube

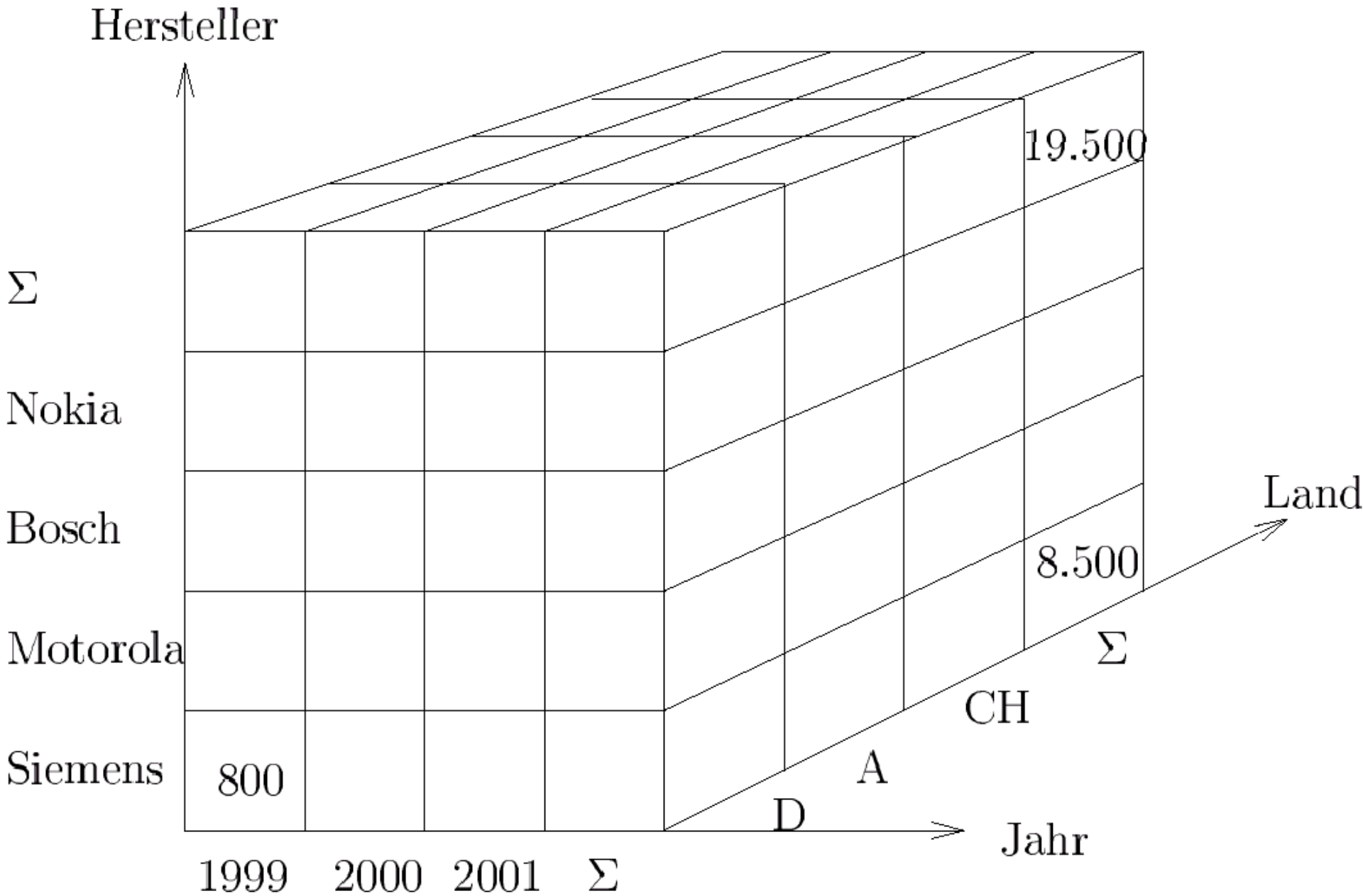
```
( select p.Hersteller, z.Jahr, sum(v.Anzahl)
from Verkäufe v, Produkte p, Zeit z
where v.Produkt = p.ProduktNr and p.Produkttyp = 'Handy'
      and v.VerkDatum = z.Datum
group by z.Jahr, p.Hersteller ) union
( select p.Hersteller, to_number(null), sum(v.Anzahl)
from Verkäufe v, Produkte p
where v.Produkt = p.ProduktNr and p.Produkttyp = 'Handy'
group by p.Hersteller ) union
( select null, z.Jahr, sum(v.Anzahl)
from Verkäufe v, Produkte p, Zeit z
where v.Produkt = p.ProduktNr and p.Produkttyp = 'Handy'
      and v.VerkDatum = z.Datum
group by z.Jahr ) union
( select null, to_number(null), sum(v.Anzahl)
from Verkäufe v, Produkte p
where v.Produkt = p.ProduktNr and p.Produkttyp = 'Handy' );
```

Relationale Struktur der Datenwürfel

Handy2DCube		
Hersteller	Jahr	Anzahl
Siemens	1999	2.000
Siemens	2000	3.000
Siemens	2001	3.500
Motorola	1999	1.000
Motorola	2000	1.000
Motorola	2001	1.500
Bosch	1999	500
Bosch	2000	1.000
Bosch	2001	1.500
Nokia	2000	1.000
Nokia	2001	1.500
Nokia	2001	2.000
null	1999	4.500
null	2000	6.500
null	2001	8.500
Siemens	null	8.500
Motorola	null	3.500
Bosch	null	3.000
Nokia	null	4.500
null	null	19.500

Handy3DCube			
Hersteller	Jahr	Land	Anzahl
Siemens	1999	D	800
Siemens	1999	A	600
Siemens	1999	CH	600
Siemens	2000	D	1.200
Siemens	2000	A	800
Siemens	2000	CH	1.000
Siemens	2001	D	1.400
...
Motorola	1999	D	400
Motorola	1999	A	300
Motorola	1999	CH	300
...
Bosch
...
null	1999	D	...
null	2000	D	...
...
Siemens	null	null	8.500
...
null	null	null	19.500

Würfeldarstellung



Der **cube**-Operator

```
select p.Hersteller, z.Jahr, f.Land, sum(v.Anzahl)
from Verkäufe v, Produkte p, Zeit z, Filialen f
where v.Produkt = p.ProduktNr and p.Produkttyp = 'Handy'
      and v.VerkDatum = z.Datum and v.Filiale = f.Filialenkennung
group by cube (z.Jahr, p.Hersteller, f.Land);
```

Wiederverwendung von Teil-Aggregaten

```
insert into VerkäufeProduktFilialeJahr
```

```
( select v.Produkt, v.Filiale, z.Jahr, sum(v.Anzahl)
```

```
from Verkäufe v, Zeit z
```

```
where v.VerkDatum = z.Datum
```

```
group by v.Produkt, v.Filiale, z.Jahr );
```

```
select v.Produkt, v.Filiale, sum(v.Anzahl)
```

```
from Verkäufe v
```

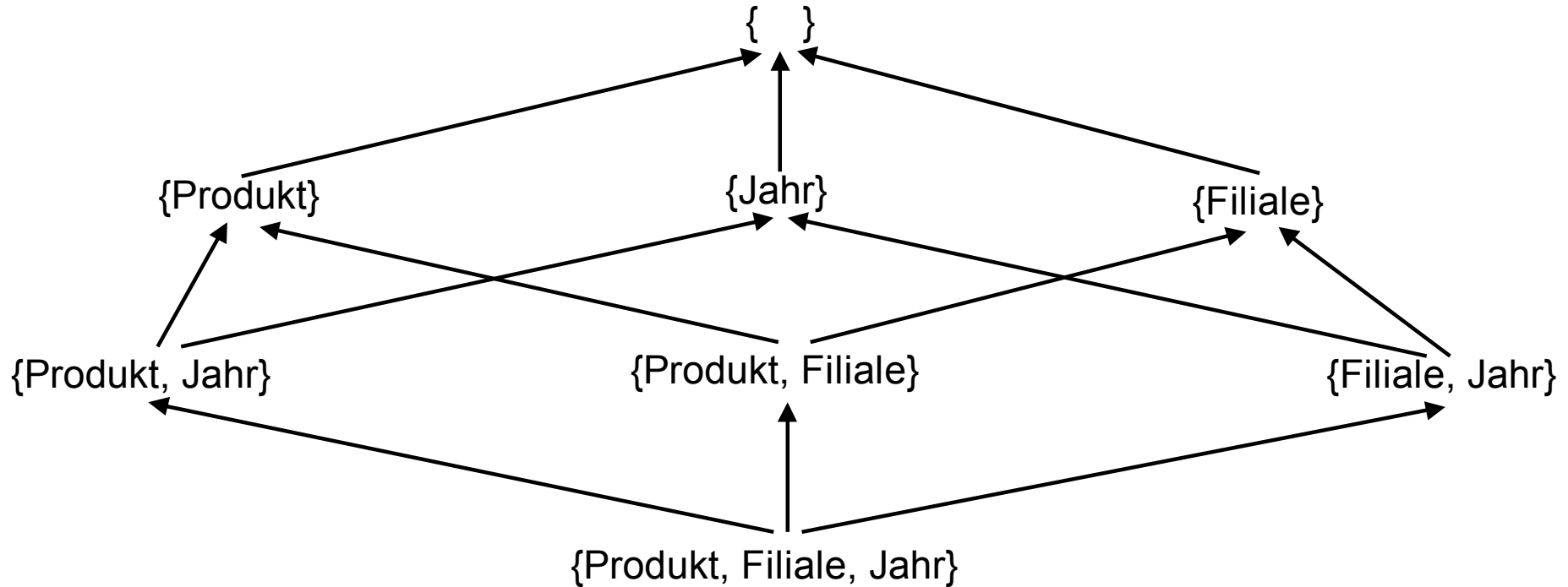
```
group by v.Produkt, v.Filiale
```

Wiederverwendung von Teil-Aggregaten

```
select v.Produkt, v.Filiale, sum(v.Anzahl)
from VerkäufeProduktFilialeJahr v
group by v.Produkt, v.Filiale
```

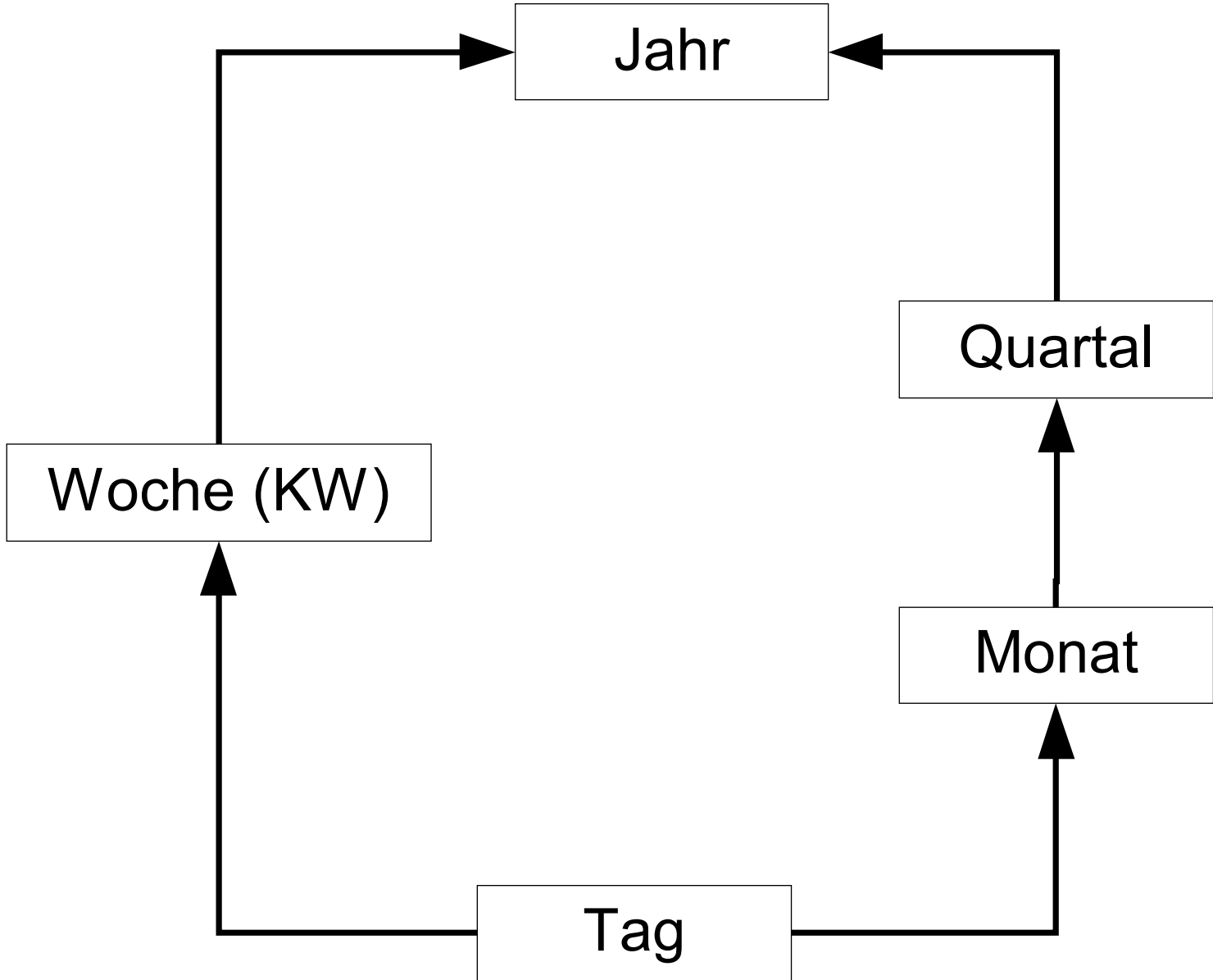
```
select v.Produkt, z.Jahr, sum(v.Anzahl)
from Verkäufe v, Zeit z
where v.VerkDatum = z.Datum
group by v.Produkt, z.Jahr
```

Die Materialisierungs-Hierarchie



- Teilaggregate T sind für eine Aggregation A wiederverwendbar wenn es einen gerichteten Pfad von T nach A gibt
- Also $T \rightarrow \dots \rightarrow A$
- Man nennt diese Materialisierungshierarchie auch einen Verband (Engl. *Lattice*)

Die Zeit-Hierarchie



Weitere Decision-Support Anfrage-Typen

- Top N-Anfragen
 - Ich will nur die N besten Treffer erhalten und nicht alle 5 Millionen
 - Muss bei der Anfrageoptimierung berücksichtigt werden
- Online Aggregation
 - Man berechnet das Ergebnis approximativ
 - Je länger die Anfrage läuft, desto genauer wird das Ergebnis

Top N-Anfragen

Select A.*

From Angestellte A, Abteilungen abt

Where A.Abteilung = abt.AbteilungsNr and abt.Ort = Passau

Order by A.Gehalt

Stop after 20

Online-Aggregation

```
Select abt.Ort, avg(A.Gehalt)
From Angestellte A, Abteilungen abt
Where A.Abteilung = abt.AbteilungsNr
Group by abt.Ort
```