Data Warehousing

DIS Exercise Course

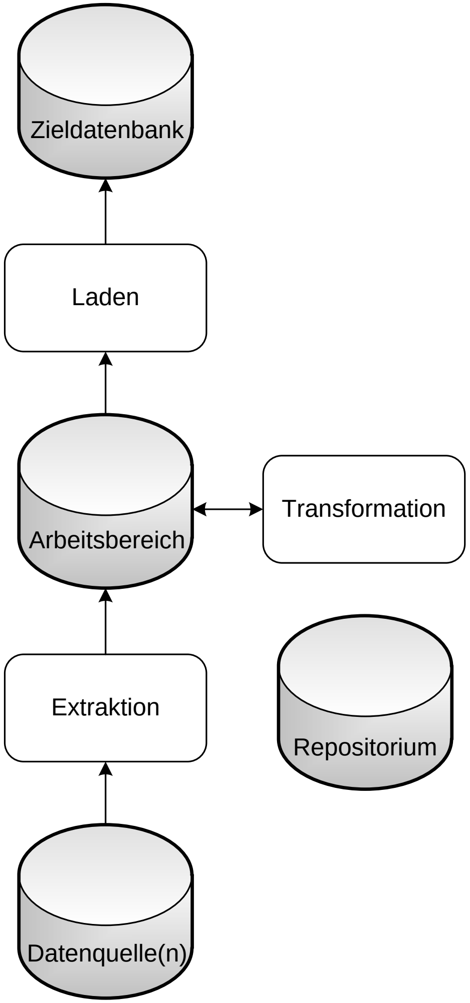
Data Warehouses

- **Example application**
  - A Manager wants to know various key figures concerning his departments:
    - Average salaries, number of employees per organisational unit and achievement of objectives – each grouped by year, age and gender.
  - The relevant data are stored in different systems for personnel management, project management and corporate management.

overview	number of employees by gender and age					salary		achievement of objectives
	total	< 40		>= 40		total	average	total
		male	female	male	female			
2009								
total	68942	22337	15443	21544	9618	126107773,82	1770,97	…
manufacturing	18614	8376	3257	5584	1397	32384781,88	1689,42	…
food	2329	466	583	682	598	3436462,79	1475,51	…
engineering	6523	2962	1231	1957	373	12333492,71	1890,77	…
other	9762	4948	1443	2945	426	16614826,38	1701,99	…
services	37918	10051	9562	12438	5867	72475706,84	1911,38	…
other	12410	3910	2624	3522	2354	21247285,10	1712,11	…
2010								
total	69037	21859	15672	22738	8768	1231036676	1699,23	…
…	…	…	…	…	…	…	…	…

Data Warehouses (2)

- **ETL process**
  - **E**xtract
    - … relevant data from various data sources
    - periodic, event-driven, query-driven
  - **T**ransform
    - … data into destination schema and format
    - syntactic transformation (date formats, strings, …)
    - semantic transformation (remove duplicates, convert values, …)
  - **L**oad
    - … data into the destination database

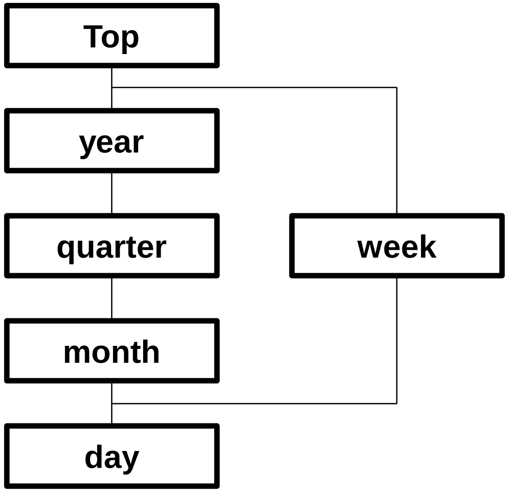


- **ETL is supported by repository:**
  - Contain rules for extraction and transformation

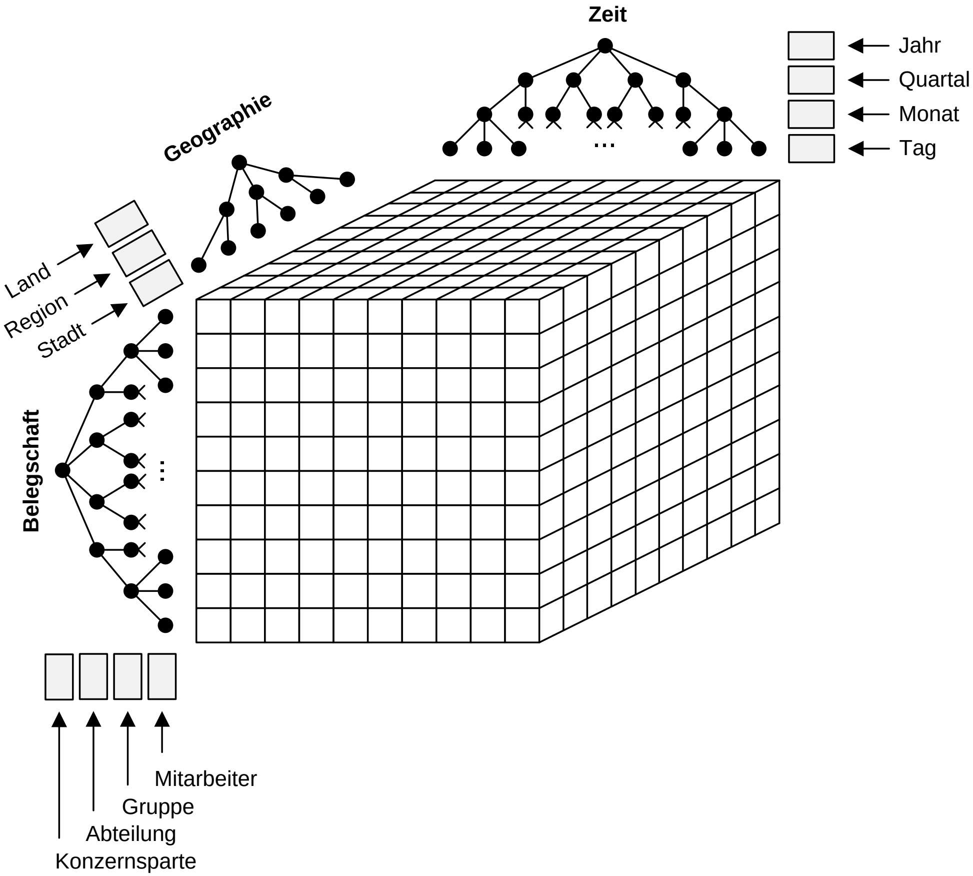
Multidimensional View

- **Data cube**
- **Data structures**
  - Qualifying data (categories)
  - Quantifying data

Dimensional Hierarchies



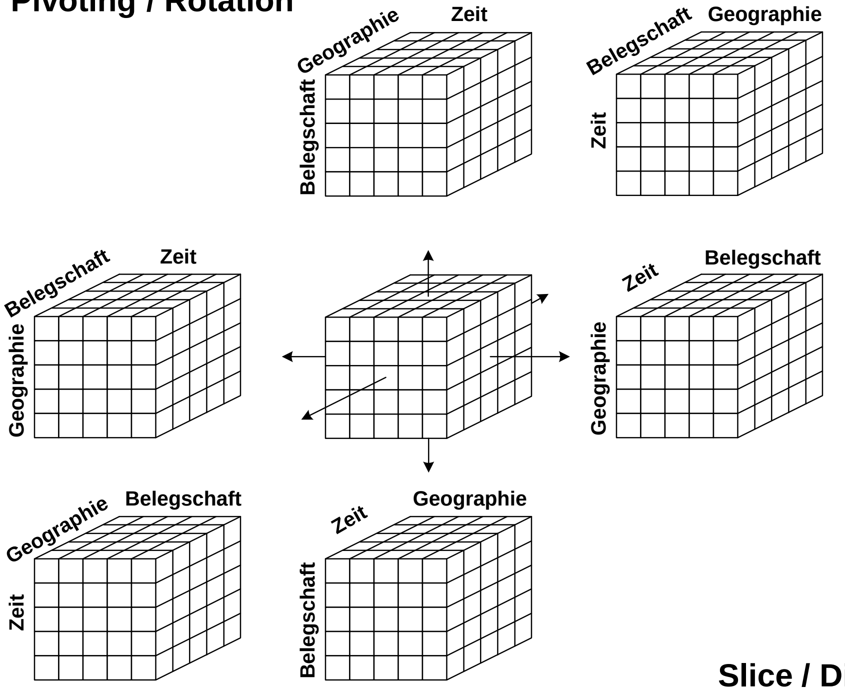
Data cube with several dimensions



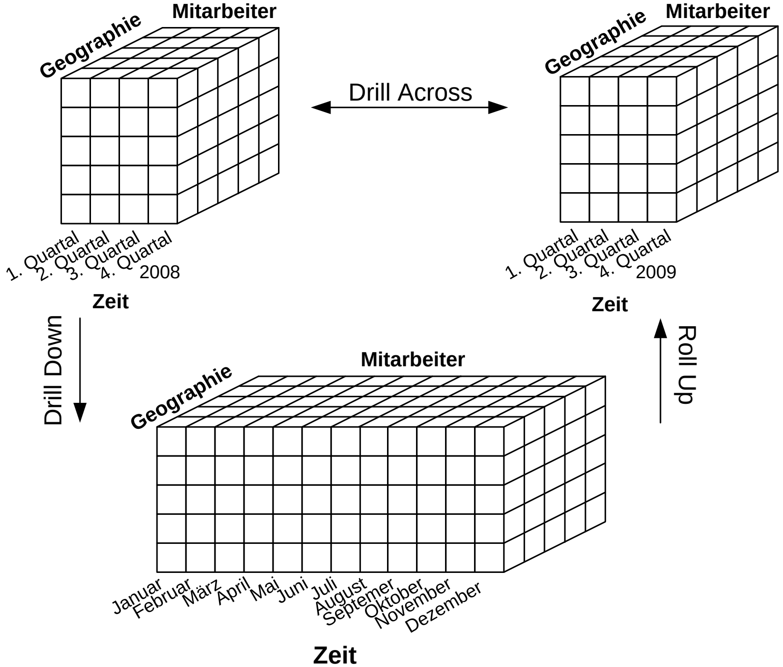
Multidimensional View (2)

- **Operations during data analysis**

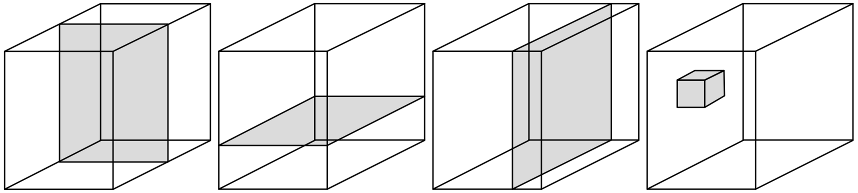
Pivoting / Rotation



Drill-Down / Roll-Up



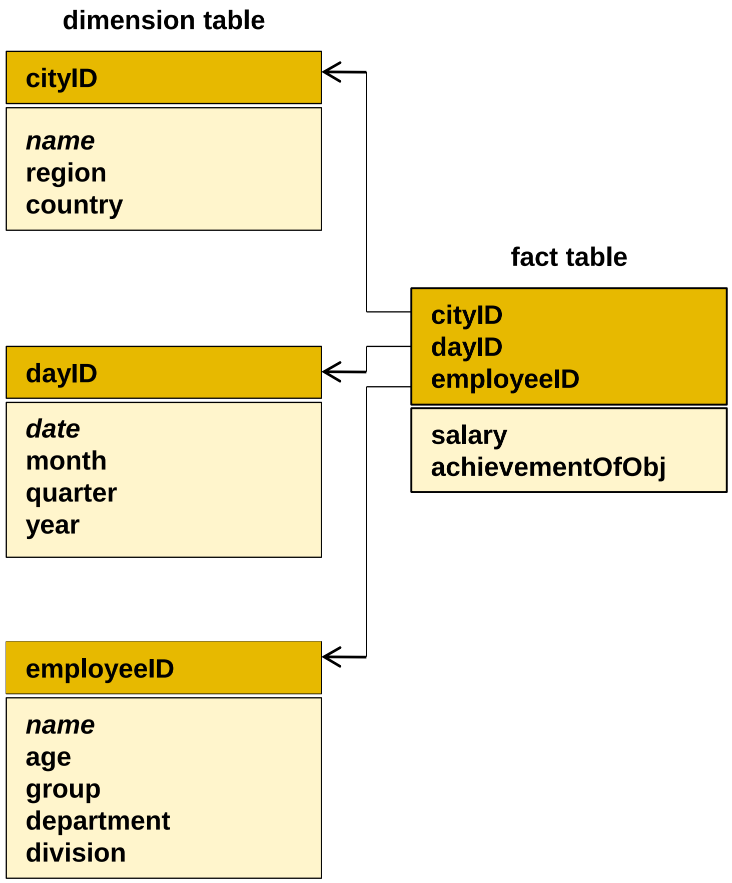
Slice / Dice



Relational Mapping of Multidimensional Data

- **Separation of structure and contents:**
  - Central **fact table**
    - Few columns, many tuples
    - partly quantifying attributes
  - Peripheral **dimension tables**
    - Feature and category attributes (structure)
    - Many columns, few tuples
    - Foreign keys connect fact and dimension tables

Star Schema



dimension table

cityID

name
region
country

dayID

date
month
quarter
year

employeeID

name
age
group
department
division

fact table

cityID
dayID
employeeID

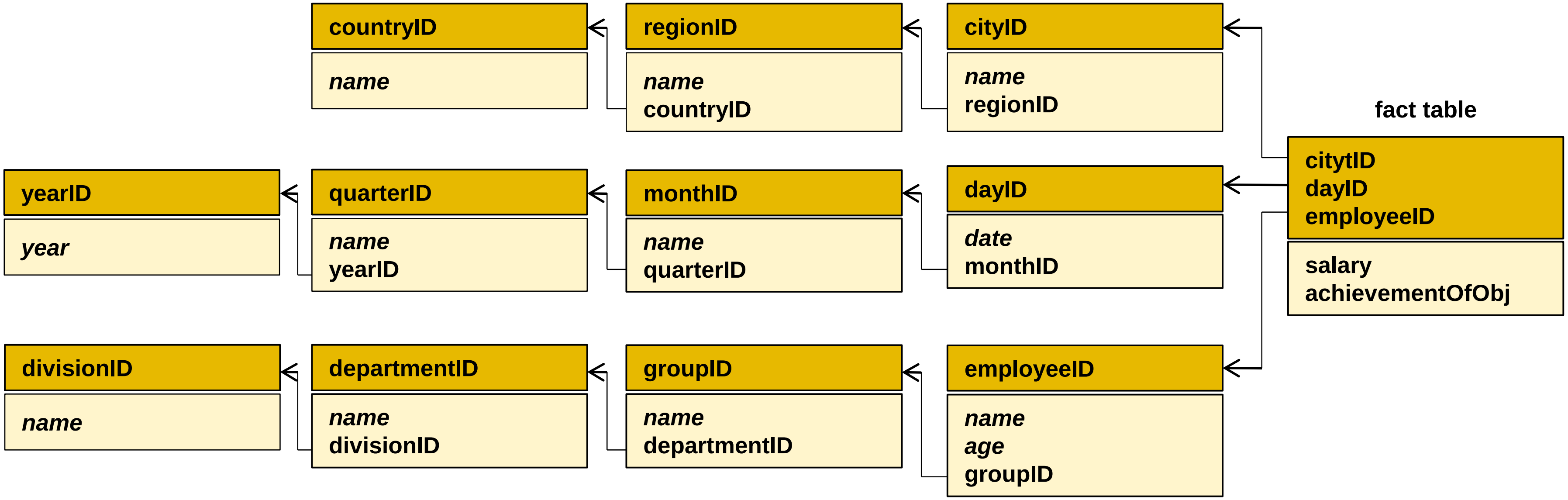
salary
achievementOfObj

Relational Mapping of Multidimensional Data(2)

- **Star Schema Problem:**
  - Redundancy

➔ **Normalised Star Schema**

➔ **(in case of forks: Snowflake Schema)**

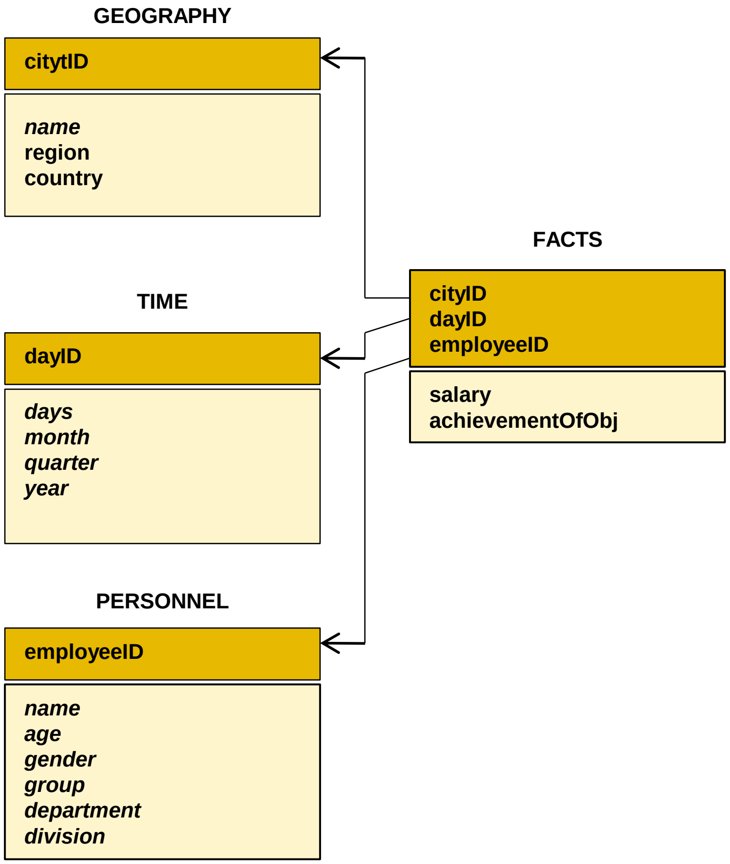


Star Query

- **Grouping queries on Star (or Snowflake) Schema**
- **Example:**

  total salary of employees by region and gender

```
SELECT          G.region, P.gender,
                SUM(salary)
FROM            facts
 NATURAL JOIN   geography G
 NATURAL JOIN   personnel P
WHERE           …
GROUP BY        (G.region, P.gender)
```



GEOGRAPHY

citytID

name
region
country

TIME

dayID

days
month
quarter
year

PERSONNEL

employeeID

name
age
gender
group
department
division

FACTS

cityID
dayID
employeeID

salary
achievementOfObj

Star Query (2) – Problem: Cross Table

- **Cross table**
  - Typical result output
  - Row and column sums
- **Grouping sets**
  - Several queries required (one query per granulation level, e.g. on Roll-Up)
  - 2^n possible combinations for n attributes
  - Combinations in the example (3 attributes = 2^3 = 8 combinations):
    - triple: (division, year, gender)
    - double: (division, year), (year, gender), (division, gender)
    - single / empty: (gender), (year), (division), ()

overview		number of employees by gender		
		total	male	female
2008	total	37780	22337	15443
	manufacturing	11633	8376	3257
	services	19613	10051	9562
	other	6534	3910	2624
2009	total	37688	22388	15300
	manufacturing	11590	8379	3211
	services	19522	10199	9323
	other	6576	3810	2766
total		75468	44725	30743

- **Idea: extension of GROUP BY to sets of granulation levels**

SQL Extensions: Grouping Sets

- **Extending the GROUP BY clause:**
  - Explicit enumeration of the desired grouping combinations

```
SELECT year, division, gender, SUM(salary) FROM …
GROUP BY GROUPING SETS ((division, year), (year, gender), (year), ())
```

```
year  division     gender  SUM(salary)
----- ------------ ------- ------------
2008  manufactur.  -       1200000
2008  services     -       2500000
2009  manufactur.  -       1500000
2009  services     -       3000000
2008  -            m       3000000
2008  -            f       7000000
2009  -            m       3000000
2009  -            f       1500000
2008  -            -       3700000
2009  -            -       4500000
-     -            -       8200000
```

  - Every GROUPING SET produces a separate set of tuples!

SQL Extensions (2)

- **Extending the GROUP BY clause:**
  - Abbreviation for the enumeration of all 2^n possible combinations

```
SELECT year, division, SUM(salary) FROM …
GROUP BY CUBE(year, division)
```

is equivalent to

```
SELECT year, division, SUM(salary) FROM …
GROUP BY GROUPING SETS ((year, division), (year), (division), ())
```

```
year  division     SUM(salary)
----- ------------ -------------------
2008  manufactur.  1200000
2008  services     2500000
2009  manufactur.  1500000
2009  services     3000000
2008  -            3700000
2009  -            4500000
-     manufactur   2700000
-     services     5500000
-     -            8200000
```