Introduction to Data Warehousing and Business Intelligence

Peter Scheuermann

What is Business Intelligence (BI)?

- From *Encyclopedia of Database Systems*:

  "[BI] refers to a set of tools and techniques that enable a company to transform its business data into timely and accurate information for the decisional process, to be made available to the right persons in the most suitable form."

What is Business Intelligence (BI)?

- BI is different from Artificial Intelligence (AI)
  - AI systems **make** decisions **for** the users
  - BI systems **help** the users make the **right** decisions, based on available data
- Combination of technologies
  - Data Warehousing (DW)
  - On-Line Analytical Processing (OLAP)
  - Data Mining (DM)
  - Data Visualization

BI and the Web

- The Web makes BI even more useful
  - Customers do not appear “physically” in a store; their behaviors cannot be observed by traditional methods
  - A website log is used to capture the behavior of each customer, e.g., sequence of pages seen by a customer, the products viewed
  - Idea: understand your customers using data and BI!
    - Utilize website logs, analyze customer behavior in more detail than before (e.g., what was **not** bought?)
    - Combine web data with traditional customer data

Case Study of an Enterprise

- Example of a chain (e.g., fashion stores or car dealers)
  - Each store maintains its own customer records and sales records
    - Hard to answer questions like: “find the total sales of Product X from stores in Evanston”
  - The same customer may be viewed as different customers for different stores; hard to detect duplicate customer information
  - Imprecise or missing data in the addresses of some customers
  - Purchase records maintained in the operational system for limited time (e.g., 6 months); then they are deleted or archived
  - The same “product” may have different prices, or different discounts in different stores

- Can you see the problems of using those data for business analysis?

Data Analysis Problems

- The same data found in many different systems
  - Example: customer data across different stores and departments
  - The same concept is defined differently
- Heterogeneous sources
  - Relational DBMS, On-Line Transaction Processing (OLTP)
  - Unstructured data in files (e.g., MS Word)
  - Legacy systems (IBM- IMS System)
  - …

Data Analysis Problems (cont')

- Data is suited for operational systems
  - Accounting, billing, etc.
  - Does not support analysis across business functions
- Data quality is bad
  - Missing data, imprecise data, different use of systems
- Data are "volatile"
  - Data deleted in operational systems (6 months)
  - Data changes over time – no historical information

Requirements for the data warehousing process

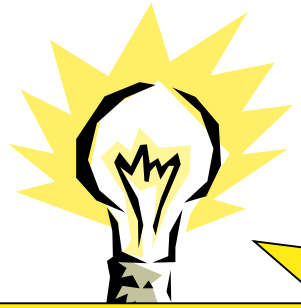
- Accessibility to users not very familiar with IT and data structures
- Integration of data on the basis of a standard enterprise model
- **Query flexibility** to maximize the advantages obtained from the existing information
- Information conciseness allowing for target-oriented and effective analyses
- **Multidimensional representation** giving users an intuitive and manageable view of information
- Correctness and completeness of integrated data

Query Types

- OLTP (On-Line Transactional Processing):
  - They execute transactions that generally read/write a small number of tuples from/to many tables connected by simple relations
  - The essential workload core is “frozen” in application programs, and ad hoc data queries are occasionally run for data maintenance
- OLAP (On-Line Analytical Processing):
  - Dynamic, multidimensional analyses that need to scan a huge amount of records to process a set of numeric data summing up the performance of an enterprise
  - Interactivity is an essential property for analysis sessions, so the actual workload constantly changes as time goes by

The key idea

- A mix of analytical queries with transactional routine queries inevitably slows down the system, and this does not meet the needs of users of both types of queries



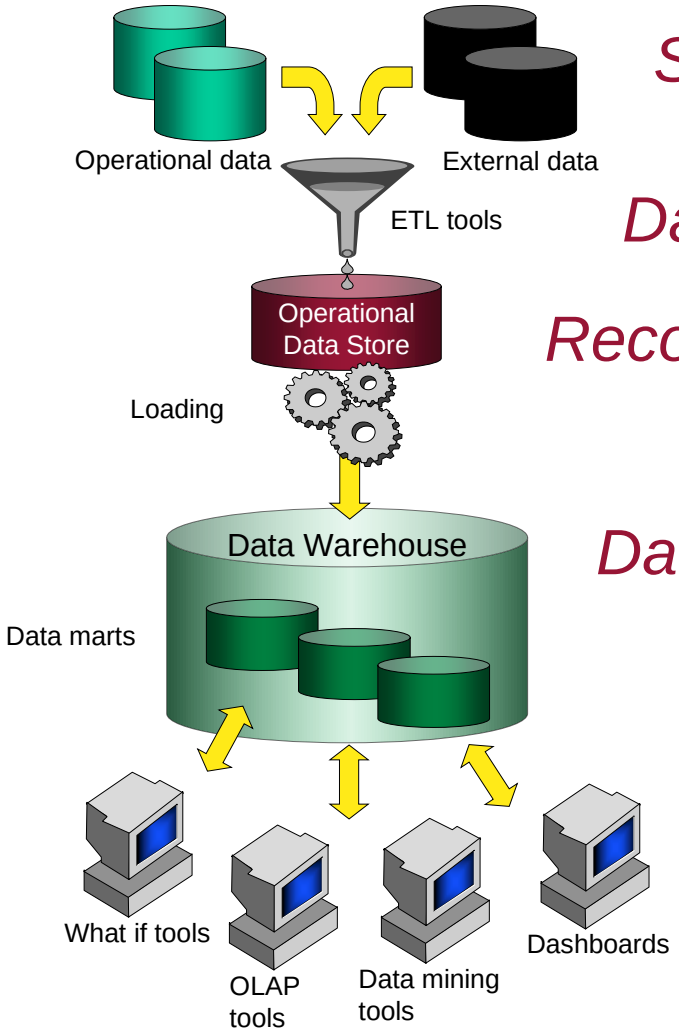
Separate *OLAP* from *OLTP* by creating a new repository that integrates data from various sources and then makes data available for analysis and evaluation aimed at decision-making processes

Data Warehousing

- Solution: Data Warehouse is a repository of information aimed at supporting the decision making process.
  - Subject oriented (versus function oriented)
  - Integrated (logically and physically)
  - Time variant (data can always be related to time)
  - Non-volatile (data not deleted, several versions)
  - Supporting management decisions (different organizations)

- Data from the operational systems are
  - Extracted
  - Cleansed
  - Transformed
  - Aggregated
  - Loaded into the DW
- A good DW is a **prerequisite** for successful BI

A reference DW architecture



Source layer

Data staging

Reconciled layer

Data warehouse layer

Analysis

EXTRACTION, TRANSFORMATION, AND LOADING:

ETL processes extract data from sources, transform and clean them, and finally load them in the ODS and in the data warehouse

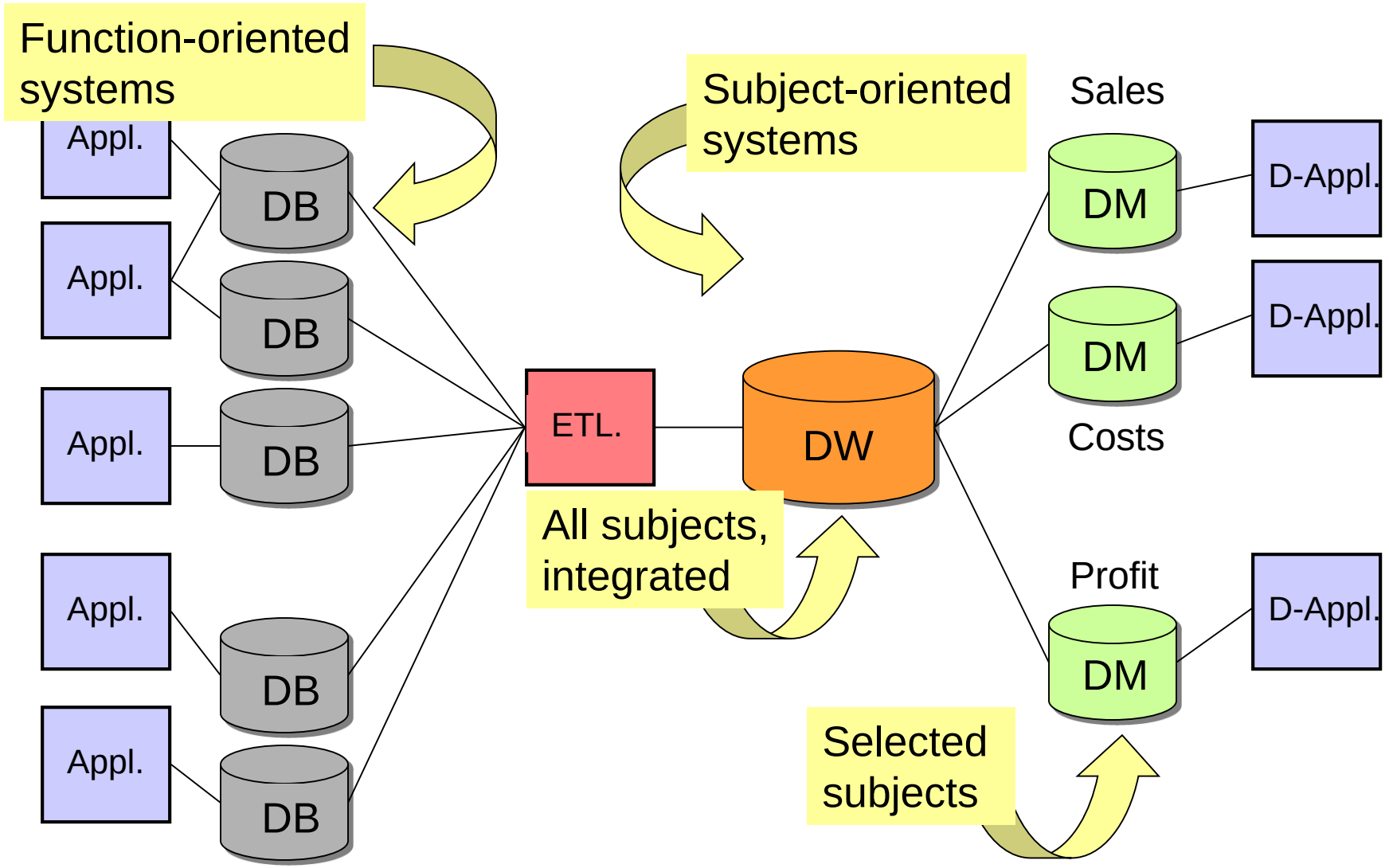
OPERATIONAL DATA STORE:

Operational data obtained after integrating and cleansing source data. As a result, those data are integrated, consistent, appropriate, current, and detailed

DATA MART:

A subset or an aggregation of the data stored to a primary data warehouse. It includes a set of information pieces relevant to a specific business area, corporate department, or category of users

Function vs. Subject Orientation



Hard/Infeasible Queries for OLTP

- Why not use the existing databases (OLTP) for business analysis?
- Business analysis queries
  - In the **past five years**, which product is the most profitable?
  - Which **public holiday** do we have the largest sales?
  - Which **week** do we have the largest sales?
  - Does the sales of **dairy products** increase over time?
- Difficult to express these queries in SQL
  - 3rd query: may extract the "week" value using a function
    - But the user has to learn many transformation functions …
  - 4th query: use a "special" table to store IDs of all dairy products, in advance
    - There can be many different dairy products; there can be many other product types as well …
- The need of multidimensional modeling

The Multidimensional model

- It is the key for representing and querying information in a DW
- *Facts* of interest are represented in *cubes* where:
  - each cell stores numerical *measures* that quantify the fact from different points of view
  - each axis is a *dimension* for analyzing measure values
  - each dimension can be the root of a *hierarchy* of attributes used to aggregated measure values

Multidimensional Modeling

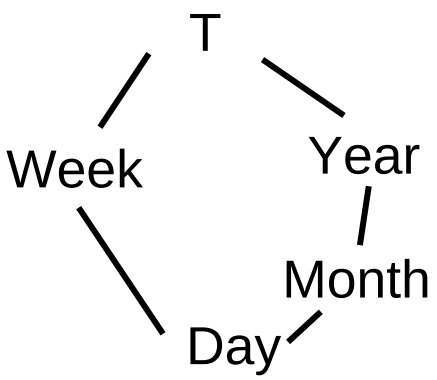
- Example: sales of supermarkets
- Facts and measures
  - Each sales record is a *fact*, and its sales value is a *measure*
- Dimensions
  - Group correlated attributes into the same dimension ➔ easier for analysis tasks
  - Each sales record is associated with its values of *Product*, *Store*, *Time*

Product	Type	Category	Store	City	County	Date	Sales
Top	Beer	Beverage	Trøjborg	Århus	Århus	25 May, 2009	5.75

Product Store Time

Multidimensional Modeling

- How do we model the *Time* dimension?
  - Hierarchies with multiple levels
  - Attributes, e.g., holiday, event



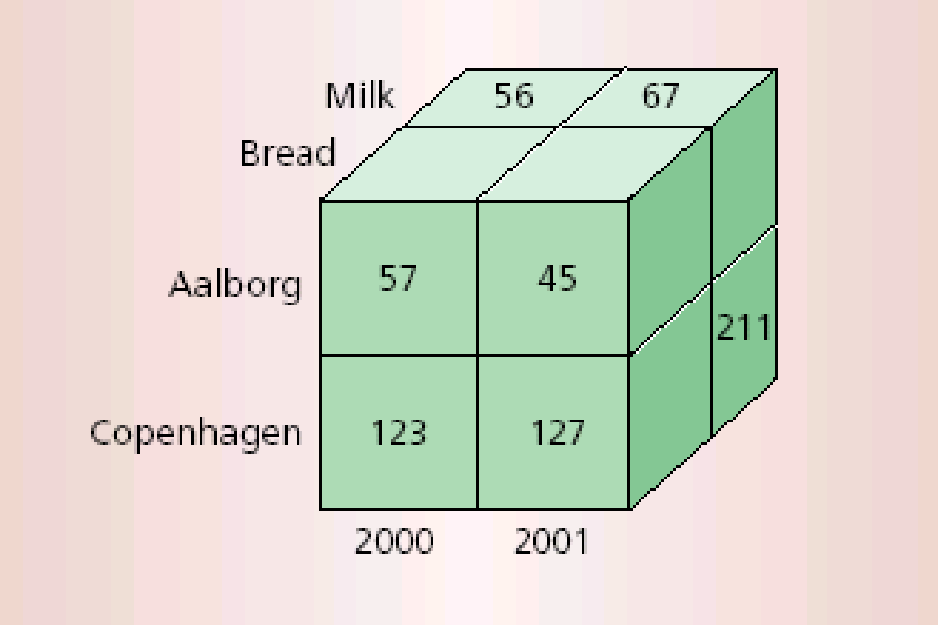
tid	day	day #	week #	month #	year	work day	…
1	January 1st 2009	1	1	1	2009	No	…
2	January 2nd 2009	2	1	1	2009	Yes	…
…		…	…	…	…	…	…

- Advantage of this model?
  - Easy to query (more about this later)
- Disadvantage?
  - More data redundancy (but controlled redundancy is acceptable)

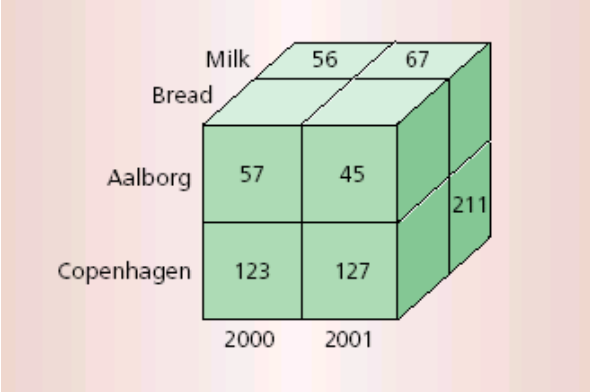
OLAP Data Cube

- Data cube
  - Useful data analysis tool in DW
  - Generalized GROUP BY queries
  - Aggregate facts based on chosen dimensions
    - Product, store, time dimensions
    - Sales represent facts
- Why data cube?
  - Good for visualization (i.e., text results hard to understand)
  - Multidimensional, intuitive
  - Supports interactive OLAP operations
- How is it different from a spreadsheet?

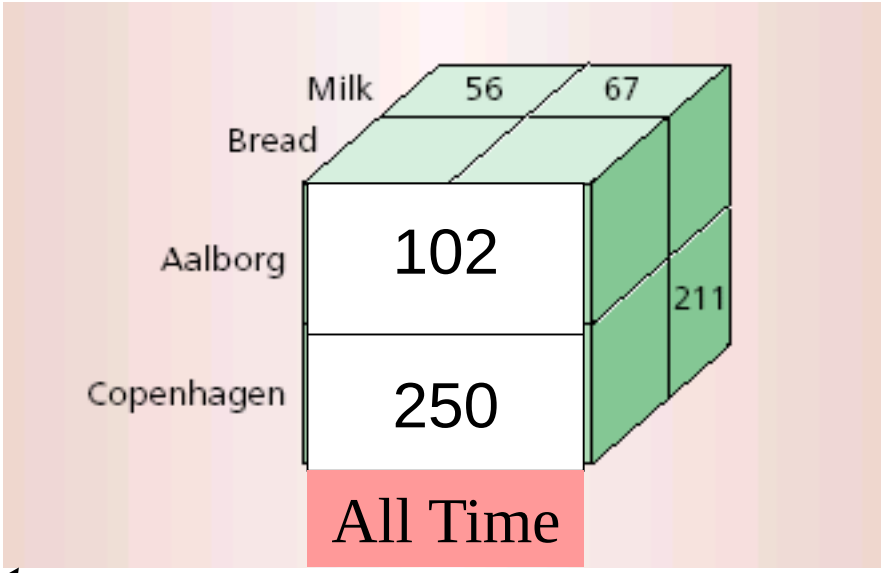
Store	Product	Year	Sales
Aalborg	Bread	2000	57
Aalborg	Milk	2000	56
Copenhagen	Bread	2000	123
…	…	…	…

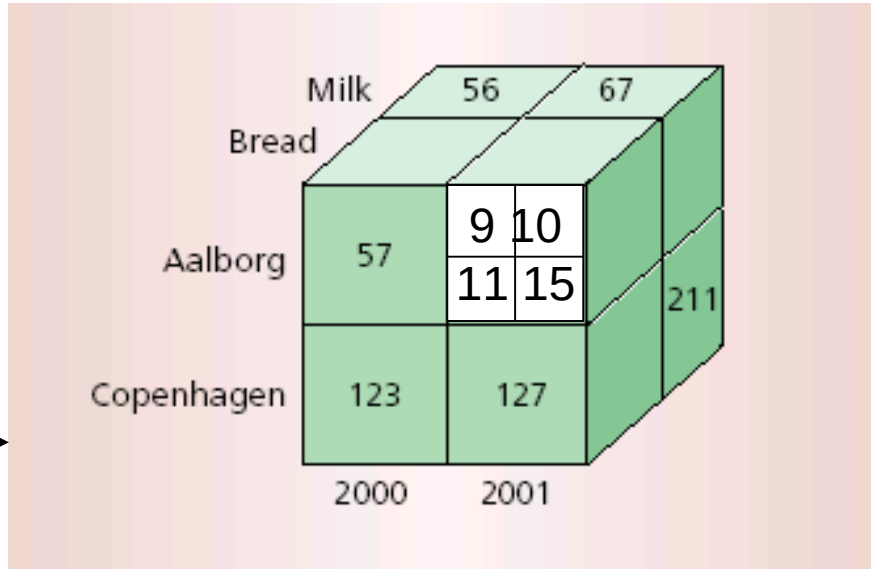


On-Line Analytical Processing (OLAP)

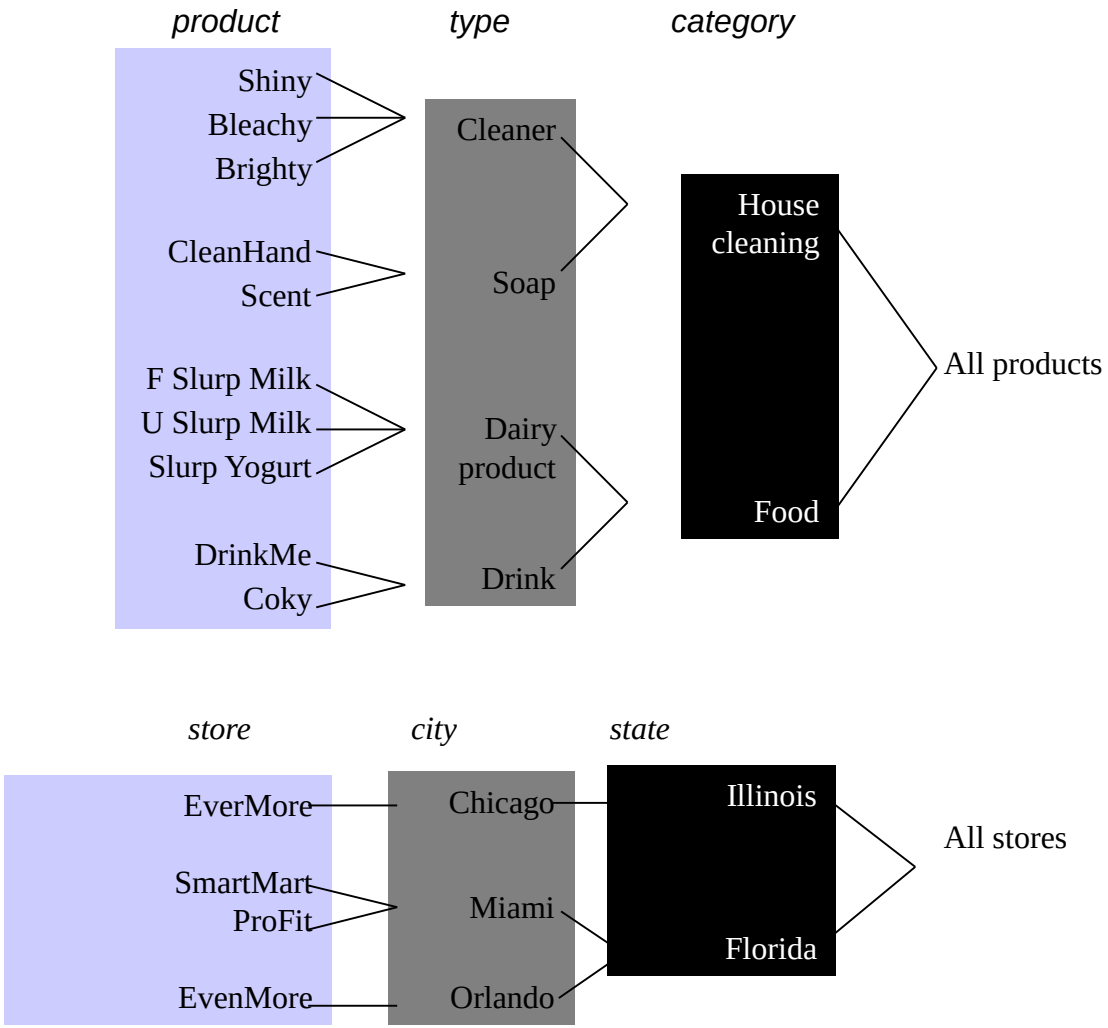


- On-Line Analytical Processing
  - Interactive analysis
  - Explorative discovery
  - Fast response times required
- OLAP operations/queries
  - Aggregation, e.g., SUM
  - Starting level, (Year, City)
    - Roll Up: Less detail
    - Drill Down: More detail
  - Slice/Dice: Selection, Year=2000

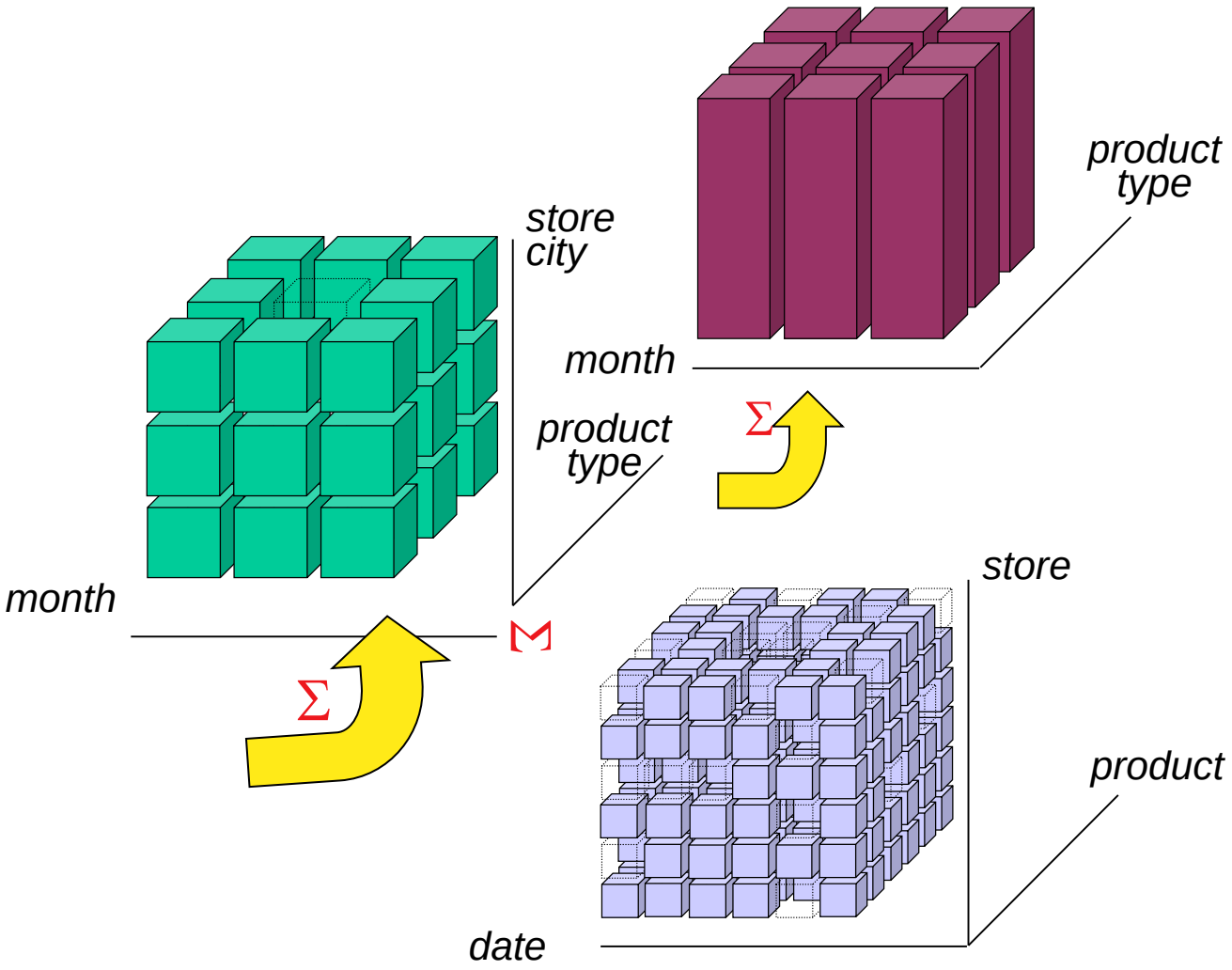




Dimension Hierarchies



Aggregation levels



Advanced Multidimensional Modeling

- Changing dimensions
  - Some dimensions are not static. They change over time.
    - A store moves to a new location with more space
    - The name of a product changes
    - A customer moves from Evanston to Wilmette
  - How do we handle these changes?
- Large-scale dimensional modeling
  - How do we coordinate the dimensions in different data cubes and data marts?

Bus architecture

Dimensions

Data marts	Time	Customer	Product	Supplier
Sales	+	+	+	
Costs			+	+
Profit	+	+	+	+

Extract, Transform, Load (ETL)

- "Getting multidimensional data into the DW"
- Problems
  1. Data from different sources
  2. Data with different formats
  3. Handling of missing data and erroneous data
  4. Query performance of DW
- ETL
  - Extract (for problem #1)
  - Transformations / cleansing (for problems #2, #3)
  - Load (for problem #4)
- The most time-consuming process in DW development
  - 80% of development time spent on ETL

Performance optimization---Materialization Example

- Imagine 1 billion sales rows, 1000 products, 100 locations
- CREATE VIEW TotalSales (pid, locid, total) AS
  SELECT s.pid, s.locid, SUM(s.sales)
  FROM Sales s
  GROUP BY s.pid, s.locid
- The materialized view has 100,000 rows
- Wish to answer the query:
  - SELECT p.category, SUM(s.sales)
    FROM Products p, Sales s WHERE p.pid=s.pid
    GROUP BY p.category
- Rewrite the query to use the view:
  - SELECT p.category, SUM(t.total)
    FROM Products p, **TotalSales t**
    WHERE p.pid=t.pid
    GROUP BY p.category
    - Query becomes 10,000 times faster!

Sales

tid	pid	locid	sales
1	1	1	10
2	1	1	20
3	2	3	40
…	…	…	…

1 billion rows

VIEW TotalSales

pid	locid	sales
1	1	30
2	3	40
…	…	…

100,000 rows