# An Overview of Data Warehousing and OLAP Technology

Slides modified by Marie (original: Otto Bian)
Discussion: Juntong

# Motivation

- **Data is used to make decisions**
  - Rapid grow of data, operational data and facts
  - Data is usually in different databases and in different physical places.

- Need for accessible, precise and comprehensive data
- Fast access regardless of the size of data
- Call for historical analysis of data
- **Goal:** provide support for decision-making rather than processing daily transactions

# Decision Support

- Computerized information systems that support decision-making
- Need for historical, summarized and consolidated data from heterogeneous sources
- **Goal:** Support knowledge workers with decision making

→ Traditional DBMSs targeted for OLTP (on-line transaction processing) not suitable for this

# Data Warehouse

*"subject-oriented, integrated, time-varying, non-volatile collection of data that is used primarily in organizational decision making."*

Main subject areas

Harmonized data from different sources

Reflect changes over time

Preservation of historical data

Operational Databases

Data Warehouse

| | OLTP | OLAP |
|---|---|---|
| **Users** | Clerk, IT professional | Knowledge worker |
| **Function** | Day to day operations | Decision support |
| **DB Design** | Application-oriented | Subject-oriented |
| **Data** | Current, up-to-date detailed. | Historical, summarized, multidimensional,… |
| **Usage** | Repetitive | Ad-hoc |
| **Access** | Read/write | Lots of scans |
| **Unit of work** | Short, simple transaction | Complex query |
| **# rec accessed** | Tens | Millions |
| **# users** | Thousands | Hundreds |
| **DB size** | 100 MB- GB | 100 GB-TB |
| **Metric** | Transaction throughput | Query throughput |

# Discussion (in pairs)

Now that we have discussed the differences between OLTP and OLAP.

- What are some real-world use cases of OLTP and OLAP?
- Which types of businesses require more out of OLAP vs OLTP?

# OLAP Architecture

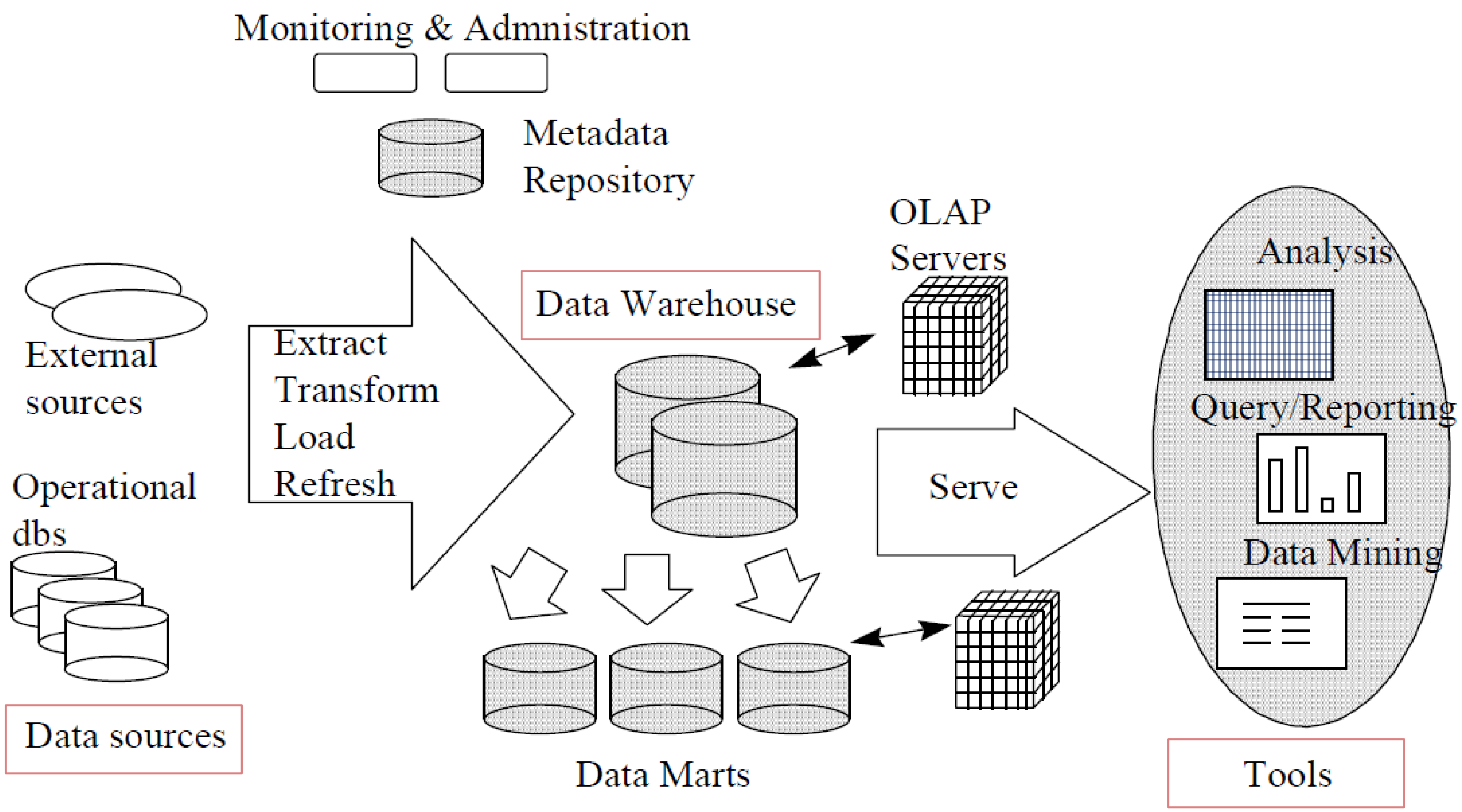



Figure 1. Data Warehousing Architecture

# Multidimensional data

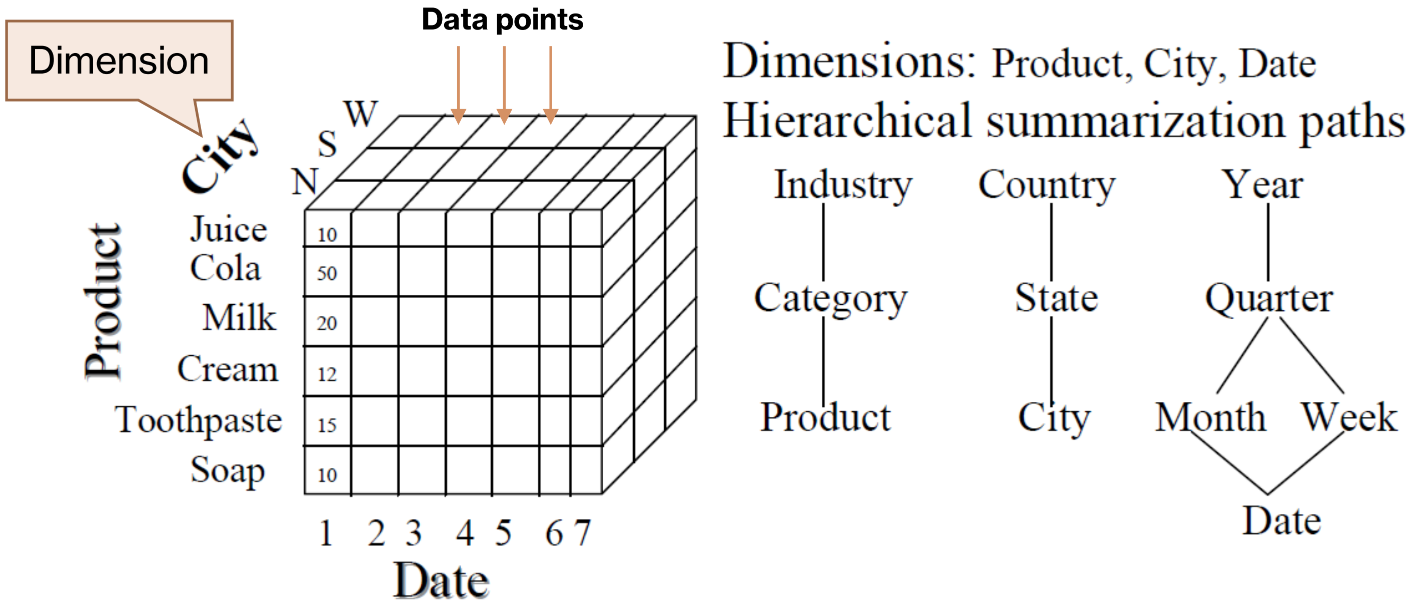



Figure 2. Multidimensional data

# OLAP Architecture

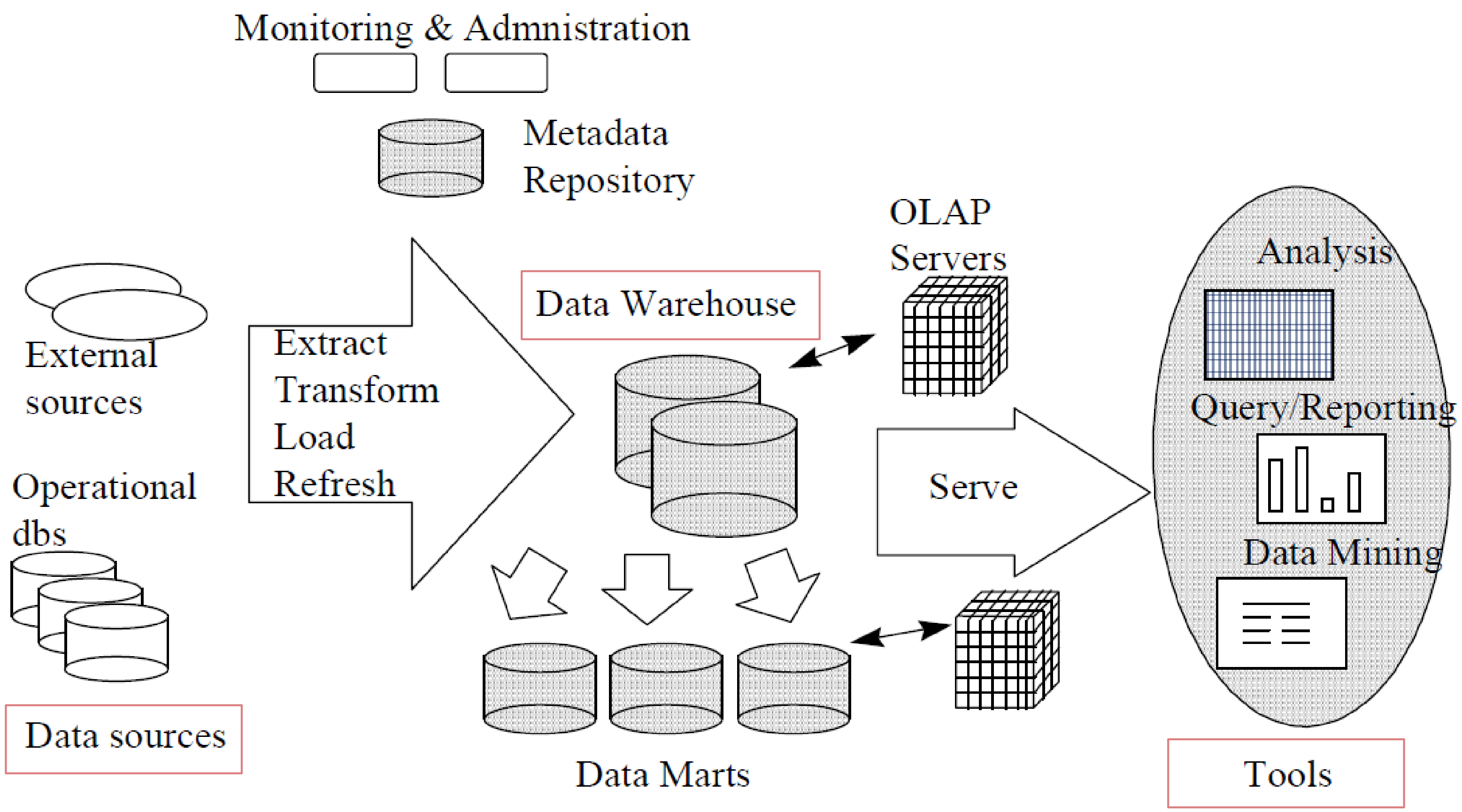



Figure 1. Data Warehousing Architecture

# Database Design Methodology

- **Multidimensionality** is core to facilitate complex analyses and visualizations
- **Star Schema**
    - **Fact table** has a pointer to each of the dimensions (acts as a multidimensional coordinate with numerical measures)
    - **Dimension tables** store attributes of the dimension
    - Fact table connects to all dimension tables with a multiple join

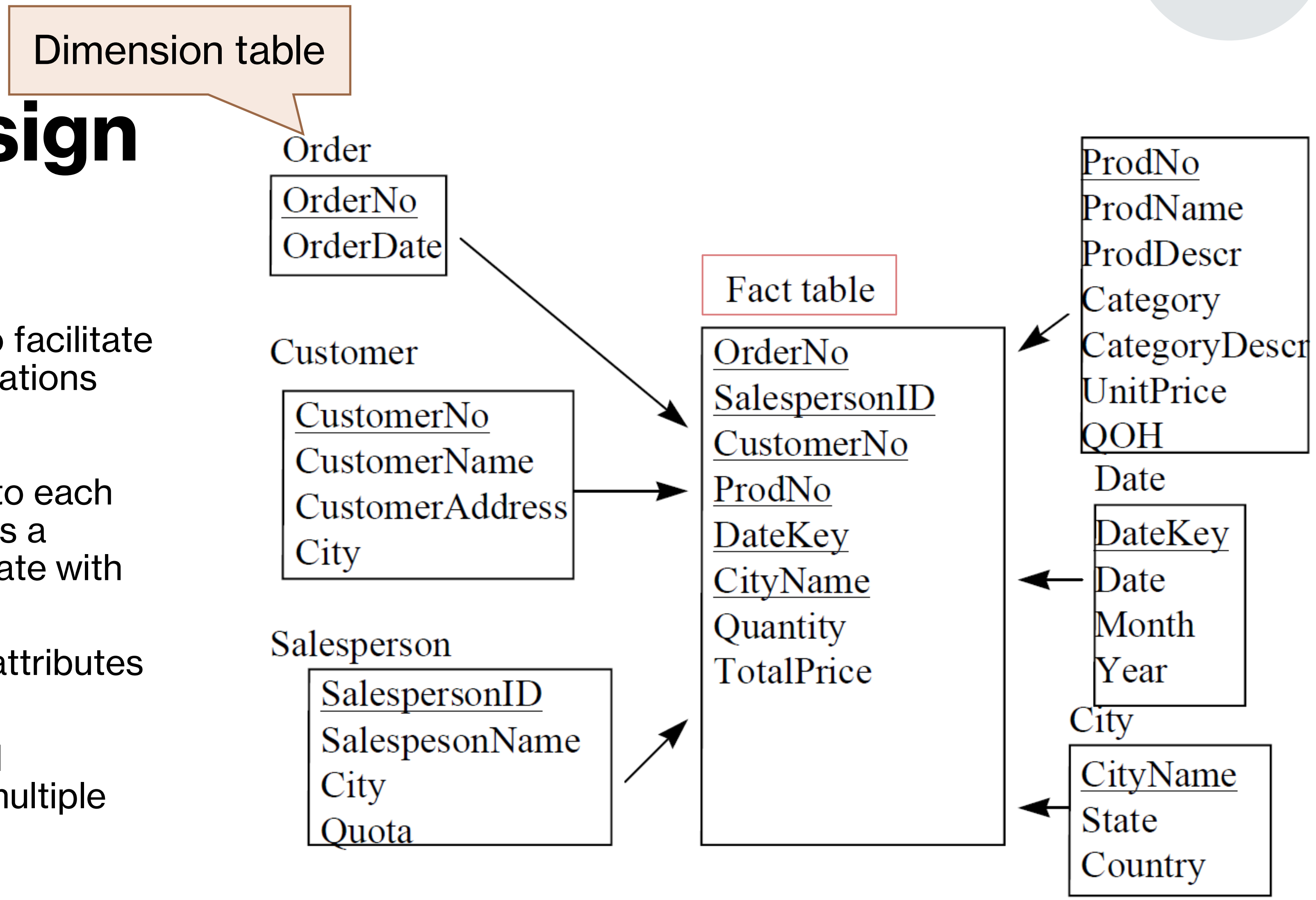



Figure 3. A Star Schema.

# Materialized Views

- Database object that contains the results of a query
- Challenges
  - Understanding which views to materialize.
  - Understand how to use such views to answer queries.
  - Efficiently updating materialized views during load and refresh.
- Choice can depend on workload characteristics, update costs, storage requirements.

# Metadata Requirements

- Reflection upon the use of data within the warehouse
- **Administrative metadata**
  - E.g. description of the source database
- **Business metadata**
  - E.g. business terms and definitions
- **Operational metadata**
  - Monitoring information

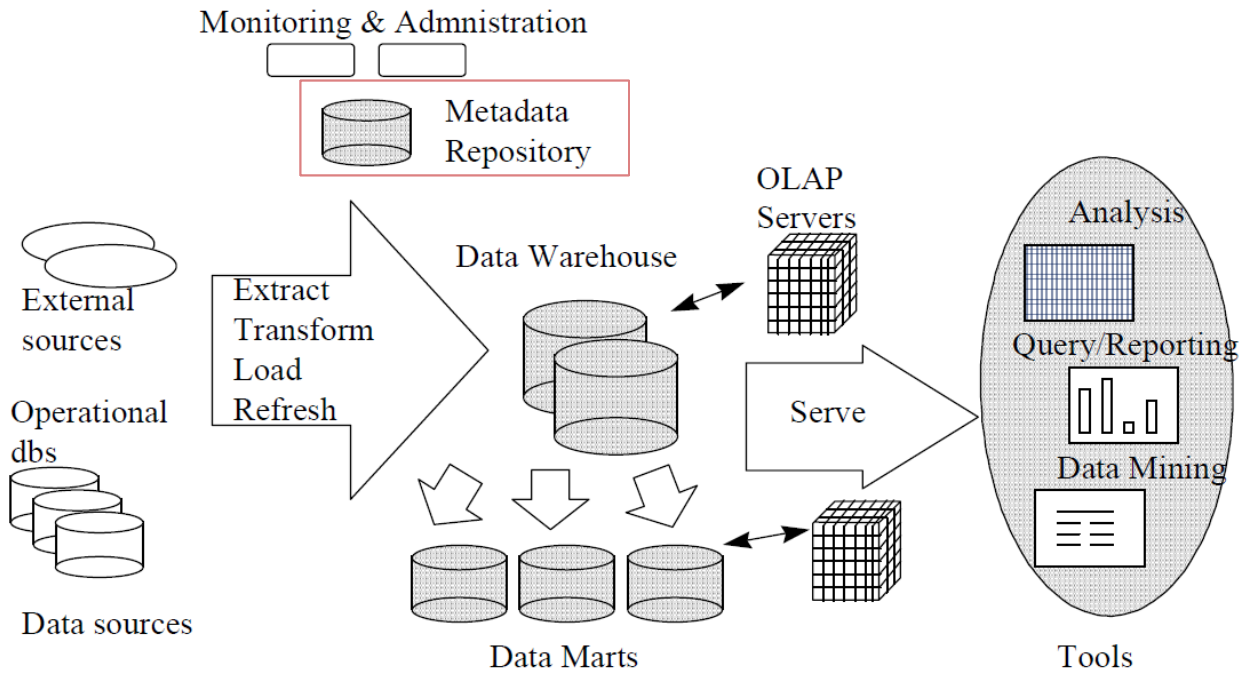



Figure 1. Data Warehousing Architecture

# Summary

- Data Warehouse for decision support
- Focuses on complex querying over large volume of data (OLAP)
- N-dimensional rather than relational table (Cube)
- Importance of metadata managing
- Possible to analyze trends (historical analysis)
- Challenging to develop efficient query processing (Materialized Views)

# Discussion (in groups of ~4)

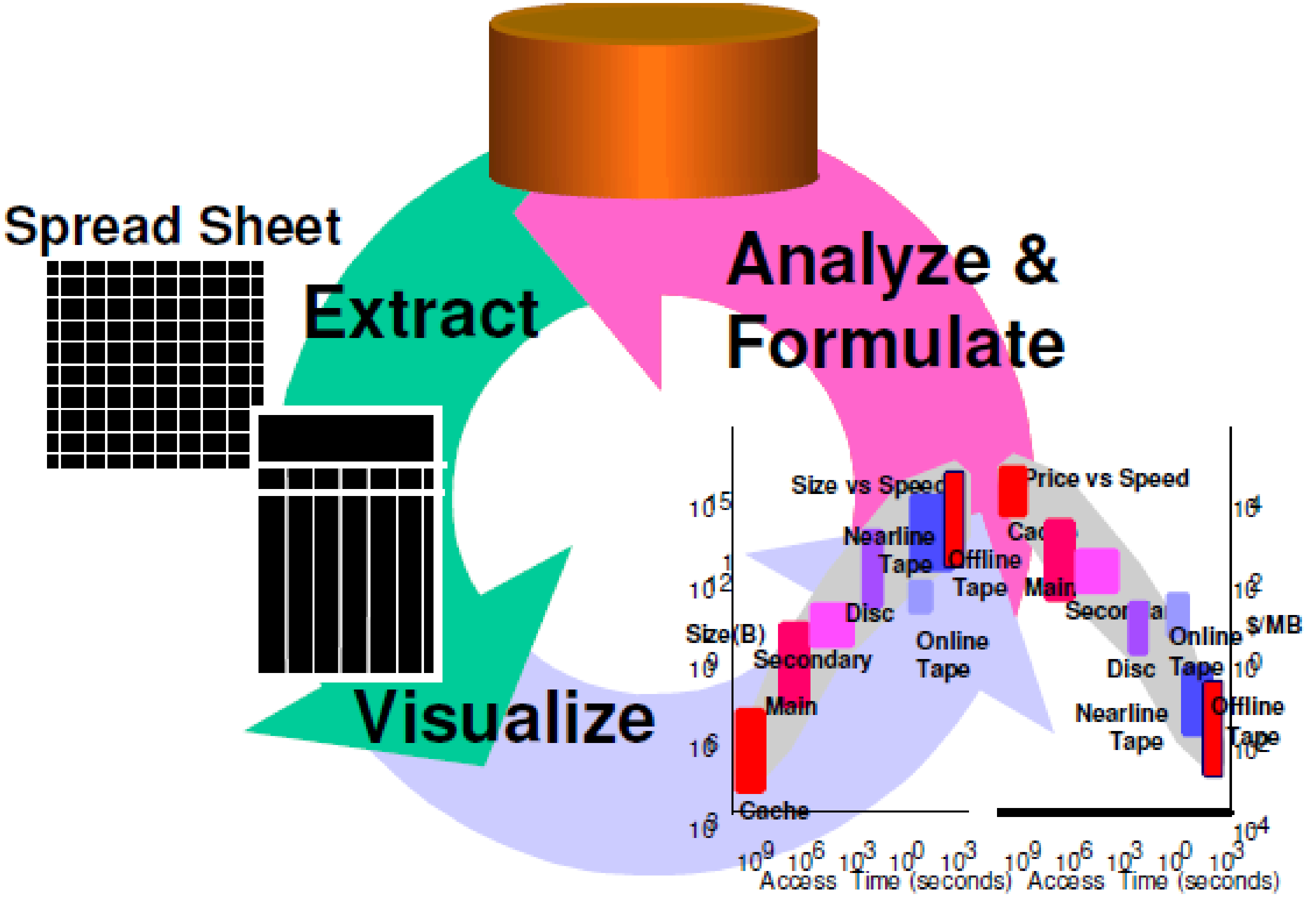
- How does the heterogeneity in data warehouses differ from the topics that we've discussed in data integration?
- What are some applications that you would use data integration for? What about a data warehouse?
    - Can you think of any applications for which *both* would be a good solution?

# Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Totals

Slides modified by Marie (original: Jim Cao)
Discussion: Juntong

# Data Analysis Applications

- **Looking for anomalies, unusual patterns**
- 4 steps:
  - **Formulating** a Query
  - **Extracting** Aggregated Data
  - **Visualizing** the Results
  - **Analyzing** the Results
- **Goal:** categorization of data values and trends, statistical information, contrast one category with another



# Discussion (in pairs)

This paper is a technical report from Microsoft.

- What is the purpose (or motivations) of such technical reports from the industry?
- Who are the targeted audience?
- What are some potential takeaways from this type of paper?

# Dimensionality Reduction

- Dimensionality reduction in data visualization for better comprehensibility
- Represent N-dimension data in 2- or 3-D

- Example – Car Sale:
    - Many different information: date of sale, sales company, color of car, model of car, year of car, etc.
    - Analyze subset of these attributes (e.g. color, model)

  → More manageable, focused dataset

# Relational Representation

- 2D flat files model an N-dimensional problem as a relation with N-attribute domains.

Dimensions

Table 1: Weather

| Time (UCT) | Latitude | Longitude | Altitude (m) | Temp (c) | Pres (mb) |
|---|---|---|---|---|---|
| 96/6/1:1500 | 37:58:33N | 122:45:28W | 102 | 21 | 1009 |
| many more rows like the ones above and below | | | | | |
| 96/6/7:1500 | 34:16:18N | 27:05:55W | 10 | 23 | 1024 |

# However, consider...

Not relational

- Data aggregated at a coarse level and then finer levels
  - **Rolling-up:** Going up the levels
  - **Drilling-down:** Going down the levels

- Table 3.a: Aggregated data at 3 distinct levels with subtotals
  - Model then Year then Color
- Sales rolled up by using totals and subtotals

- **Problems:**
  - Not relational data – the empty cells (NULL values) cannot form a key.
  - Exponential increase in number of aggregation columns when rolling up

- **Challenge with SQL:**
  - GROUP BY operator does not allow a direct construction of histograms
  - Possible inelegant solution: Multiple separate queries with different GROUP BY clauses

**Table 3.a:** Sales Roll Up by Model by Year by Color

| Model | Year | Color | Sales by Model by Year by Color | Sales by Model by Year | Sales by Model |
|---|---|---|---|---|---|
| Chevy | 1994 | black | 50 | | |
| | | white | 40 | | |
| | | | | 90 | |
| | 1995 | black | 85 | | |
| | | white | 115 | | |
| | | | | 200 | |
| | | | | | 290 |

# Date's Alternative

Relational, but…

- Displays aggregated data without additional columns for each roll-up level

  → Combines the data different levels

- **Problems:**
  - Enormous number of domains
  - Naming problems
  - Very long names
  - Same SQL challenges

**Table 3.b:** Sales Roll-Up by Model by Year by Color as recommended by Chris Date [Date1].

| Model | Year | Color | Sales | Sales by Model by Year | Sales by Model |
|---|---|---|---|---|---|
| Chevy | 1994 | black | 50 | 90 | 290 |
| Chevy | 1994 | white | 40 | 90 | 290 |
| Chevy | 1995 | black | 85 | 200 | 290 |
| Chevy | 1995 | white | 115 | 200 | 290 |

# Pivot Table

- Excel pivot table alternative to the traditional roll-up method (Table 3.a)
    - Transposes spreadsheet
    - Aggregate cells based on cell values

- **Problem:**
    - Size of the Pivot Table
    - N and M values → pivot table has N x M values
        - Many columns
        - Obtuse column

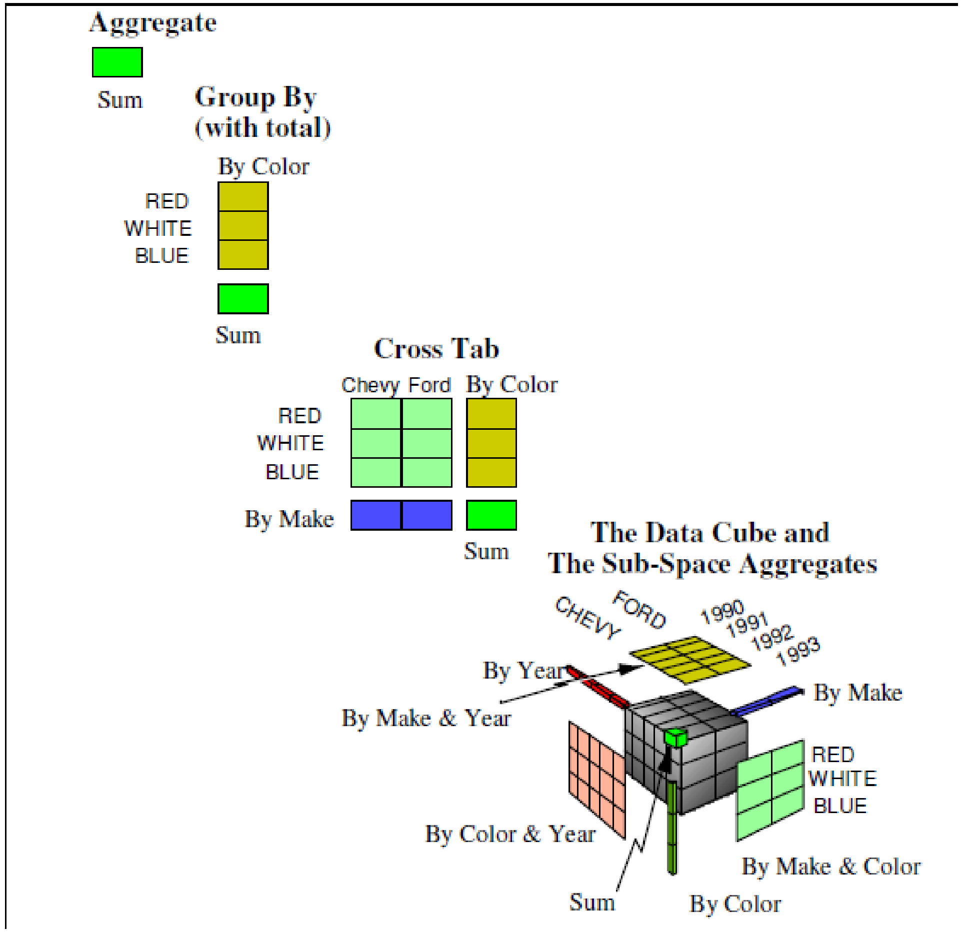
**Table 4:** An Excel pivot table representation of Table 3 with Ford sales data included.

| Sum | Year | Color | | | | | |
|---|---|---|---|---|---|---|---|
| Sales | 1994 | | 1994 Total | 1995 | | 1995 Total | Grand Total |
| Model | black | white | | black | white | | |
| Chevy | 50 | 40 | 90 | 85 | 115 | 200 | 290 |
| Ford | 50 | 10 | 60 | 85 | 75 | 160 | 220 |
| Grand Total | 100 | 50 | 150 | 170 | 190 | 360 | 510 |

# Data CUBE

- N-dimensional generalization of simple aggregate functions
- Data cube operator builds a table containing all these aggregate values
    - O-D data cube: a point.
    - 1-D data cube: a line & a point.
    - 2-D data cube: a cross tabulation, a plane, two lines, and a point.
    - 3-D data cube: a cube with three intersecting 2D cross tabs

- **Data Cube vs. SQL:**
    - Simultaneous aggregation across multiple dimensions possible
    - More complex and flexible analyzes possible



# The CUBE operator

- CUBE = relational operator
  - GROUP BY and ROLL UP: degenerate forms of the operator

- Power Set of Aggregation Columns
  - All possible subsets of grouping columns

- How does a CUBE operator work?
  - First Aggregate
  - Second Aggregate with UNIONS
  - Possible super-aggregate Values
  - Super-aggregates through ROLLUP

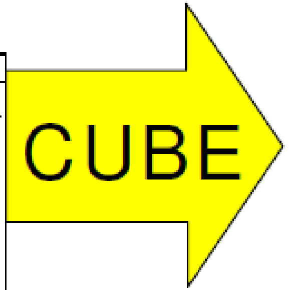
- Super-aggregate → aggregated data that summarizes multiple levels of data hierarchies

```
SELECT Model, Year, Color, SUM (Sales) AS Sales
FROM Sales
WHERE Model in ['Ford', 'Chevy']
  AND  Year BETWEEN 1994 AND 1995
GROUP BY CUBE Model, Year, Color;
```

```
SELECT Model, Year, Color, SUM(sales) AS Sales
FROM Sales
WHERE Model in {'Ford', 'Chevy'}
  AND Year BETWEEN 1990 AND 1992
GROUP BY CUBE Model, Year, Color;
```

SALES

| Model | Year | Color | Sales |
|---|---|---|---|
| Chevy | 1990 | red | 5 |
| Chevy | 1990 | white | 87 |
| Chevy | 1990 | blue | 62 |
| Chevy | 1991 | red | 54 |
| Chevy | 1991 | white | 95 |
| Chevy | 1991 | blue | 49 |
| Chevy | 1992 | red | 31 |
| Chevy | 1992 | white | 54 |
| Chevy | 1992 | blue | 71 |
| Ford | 1990 | red | 64 |
| Ford | 1990 | white | 62 |
| Ford | 1990 | blue | 63 |
| Ford | 1991 | red | 52 |
| Ford | 1991 | white | 9 |
| Ford | 1991 | blue | 55 |
| Ford | 1992 | red | 27 |
| Ford | 1992 | white | 62 |
| Ford | 1992 | blue | 39 |




DATA CUBE

| Model | Year | Color | Sales |
|---|---|---|---|
| Chevy | 1990 | blue | 62 |
| Chevy | 1990 | red | 5 |
| Chevy | 1990 | white | 95 |
| Chevy | 1990 | ALL | 154 |
| Chevy | 1991 | blue | 49 |
| Chevy | 1991 | red | 54 |
| Chevy | 1991 | white | 95 |
| Chevy | 1991 | ALL | 198 |
| Chevy | 1992 | blue | 71 |
| Chevy | 1992 | red | 31 |
| Chevy | 1992 | white | 54 |
| Chevy | 1992 | ALL | 156 |
| Chevy | ALL | blue | 182 |
| Chevy | ALL | red | 90 |
| Chevy | ALL | white | 236 |
| Chevy | ALL | ALL | 508 |
| Ford | 1990 | blue | 63 |
| Ford | 1990 | red | 64 |
| Ford | 1990 | white | 62 |
| Ford | 1990 | ALL | 189 |
| Ford | 1991 | blue | 55 |
| Ford | 1991 | red | 52 |
| Ford | 1991 | white | 9 |
| Ford | 1991 | ALL | 116 |
| Ford | 1992 | blue | 39 |
| Ford | 1992 | red | 27 |
| Ford | 1992 | white | 62 |
| Ford | 1992 | ALL | 128 |
| Ford | ALL | blue | 157 |
| Ford | ALL | red | 143 |
| Ford | ALL | white | 133 |
| Ford | ALL | ALL | 433 |
| ALL | 1990 | blue | 125 |
| ALL | 1990 | red | 69 |
| ALL | 1990 | white | 149 |
| ALL | 1990 | ALL | 343 |
| ALL | 1991 | blue | 106 |
| ALL | 1991 | red | 104 |
| ALL | 1991 | white | 110 |
| ALL | 1991 | ALL | 314 |
| ALL | 1992 | blue | 110 |
| ALL | 1992 | red | 58 |
| ALL | 1992 | white | 116 |
| ALL | 1992 | ALL | 284 |
| ALL | ALL | blue | 339 |
| ALL | ALL | red | 233 |
| ALL | ALL | white | 369 |
| ALL | ALL | ALL | 941 |

**Figure 4:** A 3D data cube (right) built from the table at the left by the `CUBE` statement at the top of the figure.

SELECT Model, Year, Color, SUM (Sales) AS Sales
FROM Sales
WHERE Model in ['Ford', 'Chevy']
  AND Year BETWEEN 1994 AND 1995
GROUP BY CUBE Model, Year, Color;

# Discussion (in groups of 4)

The abstract mentions that “many of the features are being added to the SQL standard”.

- Does this strike you as a big or a small change to SQL?
- How do we decide if a new feature should be added to the standard? What are the deciding factors?

# Data Cubes - Summary

- Generalizes and unifies aggregates, group by, histograms, roll-ups and drill-downs and cross tabs.
- Based on a relational representation of aggregate data
- Easy to compute for a wide class of functions
- Flexible and dynamic form of data analysis