

Fall 2004, CIS, Temple University

CIS527: Data Warehousing, Filtering, and Mining

Lecture 2

- Data Warehousing and OLAP Technology for Data Mining

Lecture slides taken/modified from:

- **Jiawei Han** (http://www-sal.cs.uiuc.edu/~hanj/DM_Book.html)



Chapter 2: Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation
- Further development of data cube technology
- From data warehousing to data mining



What is Data Warehouse?

- Defined in many different ways, but not rigorously.
 - A decision support database that is maintained **separately** from the organization's operational database
 - Support **information processing** by providing a solid platform of consolidated, historical data for analysis.
- Một kho dữ liệu là một bộ dữ liệu **hướng chủ đề, tích hợp, biến động theo thời gian, và không mất đi** được sử dụng để hỗ trợ quá trình ra quyết định quản lý .”—W. H. Inmon
- Data warehousing:
 - The process of constructing and using data warehouses



Data Warehouse—Subject-Oriented

- Qua việc cung cấp một khung nhìn xúc tích và đơn giản xung quanh các vấn đề của một chủ đề cụ thể. Chúng ta có thể thực hiện đặc điểm này bằng cách loại trừ các dữ liệu không hữu ích trong tiến trình hỗ trợ quyết định.
- Qua việc được tổ chức xung quanh các đối tượng chính, chẳng hạn như **customer, product, sales**.
- Tập trung vào mô hình hóa và phân tích các dữ liệu cho những người ra quyết định, không phải cho các hoạt động tác nghiệp hàng ngày hoặc cho xử lý giao dịch.



Data Warehouse—Integrated

- Constructed by integrating multiple, heterogeneous data sources
 - relational databases, flat files, on-line transaction records
- Data cleaning and data integration techniques are applied.
 - Ensure consistency in naming conventions, encoding structures, attribute measures, etc. among different data sources
 - E.g., Hotel price: currency, tax, breakfast covered, etc.
 - When data is moved to the warehouse, it is converted.



Data Warehouse—Time Variant

- The time horizon for the data warehouse is significantly longer than that of operational systems.
 - Operational database: current value data.
 - Data warehouse data: provide information from a historical perspective (e.g., past 5-10 years)
- Every key structure in the data warehouse
 - Contains an element of time, explicitly or implicitly
 - However, the key of operational data may or may not contain “time element”.



Data Warehouse—Non-Volatile

- A **physically separate store** of data transformed from the operational environment.
- Operational **update of data does not necessarily occur** in the data warehouse environment.
 - Does not require transaction processing, recovery, and concurrency control mechanisms
 - Often requires only two operations in data accessing:
 - *initial loading of data* and *access of data*.



Data Warehouse vs. Heterogeneous DBMS

- Traditional heterogeneous DB integration:
 - Build **wrappers/mediators** on top of heterogeneous databases
 - **Query driven** approach
 - A query posed to a client site is translated into queries appropriate for individual heterogeneous sites; The results are integrated into a global answer set
 - Involving complex information filtering
 - Competition for resources at local sources
- Data warehouse: **update-driven**, high performance
 - Information from heterogeneous sources is integrated in advance and stored in warehouses for direct query and analysis



Data Warehouse vs. Operational DBMS

- OLTP (on-line transaction processing)
 - Major task of traditional relational DBMS
 - Day-to-day operations: purchasing, inventory, banking, manufacturing, payroll, registration, accounting, etc.
- OLAP (on-line analytical processing)
 - Major task of data warehouse system
 - Data analysis and decision making
- Distinct features (OLTP vs. OLAP):
 - OLTP có định hướng người sử dụng còn OLAP có định hướng hệ thống: OLTP phục vụ khách hàng còn OLAP phục vụ thị trường
 - Data contents: current, detailed vs. historical, consolidated
 - Database design: mô hình ER + ứng dụng vs mô hình sao + chủ thể
 - View: current, local vs. evolutionary, integrated
 - Access patterns: update vs. read-only but complex queries



Why Separate Data Warehouse?

- High performance for both systems
 - DBMS—tuned for OLTP: access methods, indexing, concurrency control, recovery
 - Warehouse—tuned for OLAP: complex OLAP queries, multidimensional view, consolidation.
- Different functions and different data:
 - Decision support requires **historical data** which operational DBs do not typically maintain
 - Decision Support requires **consolidation** (aggregation, summarization) of data from heterogeneous sources
 - Different sources typically use **inconsistent data representations**, codes and formats which have to be reconciled



Chapter 2: Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation
- Further development of data cube technology
- From data warehousing to data mining

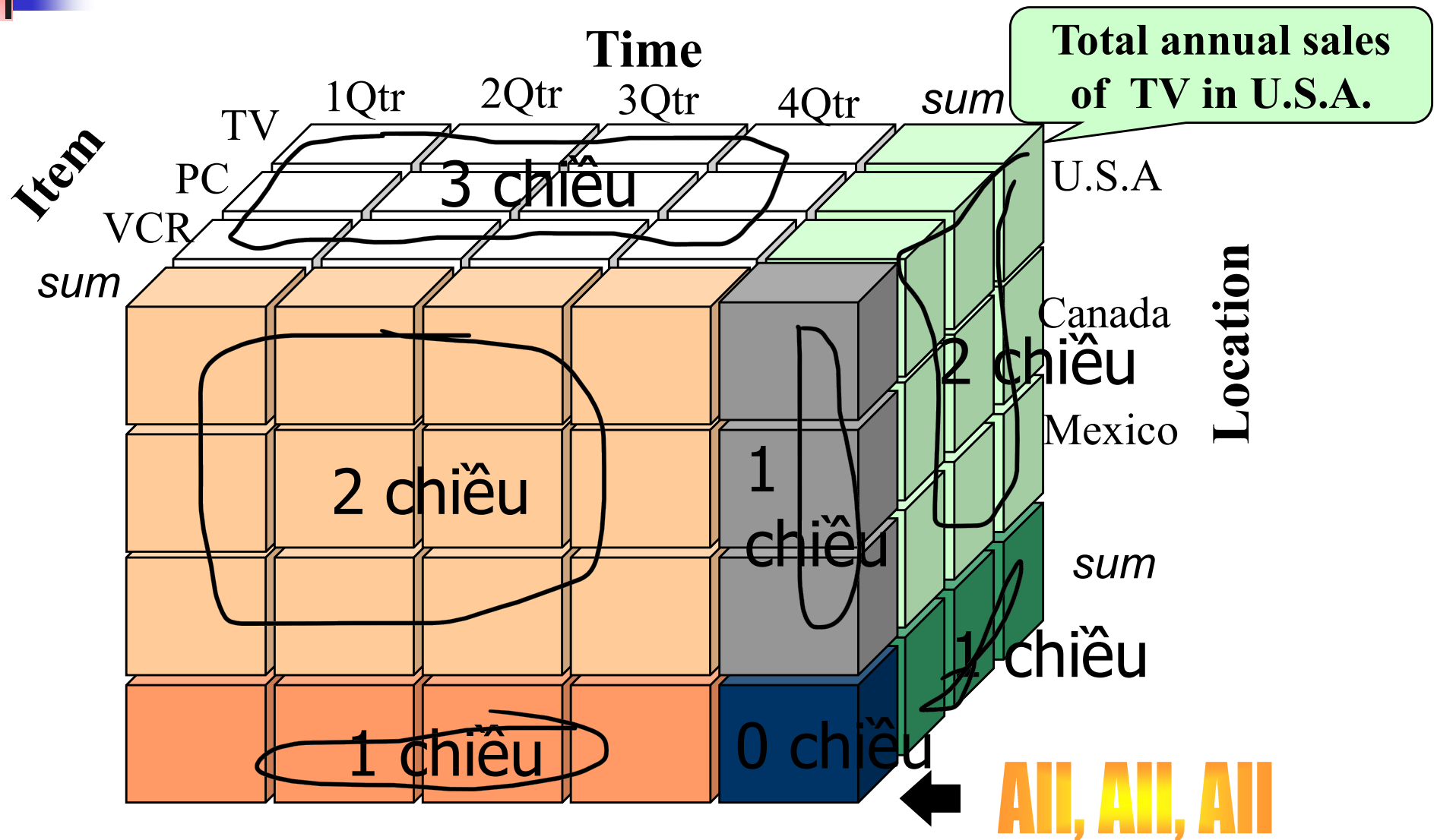


A Multi-Dimensional Data Model

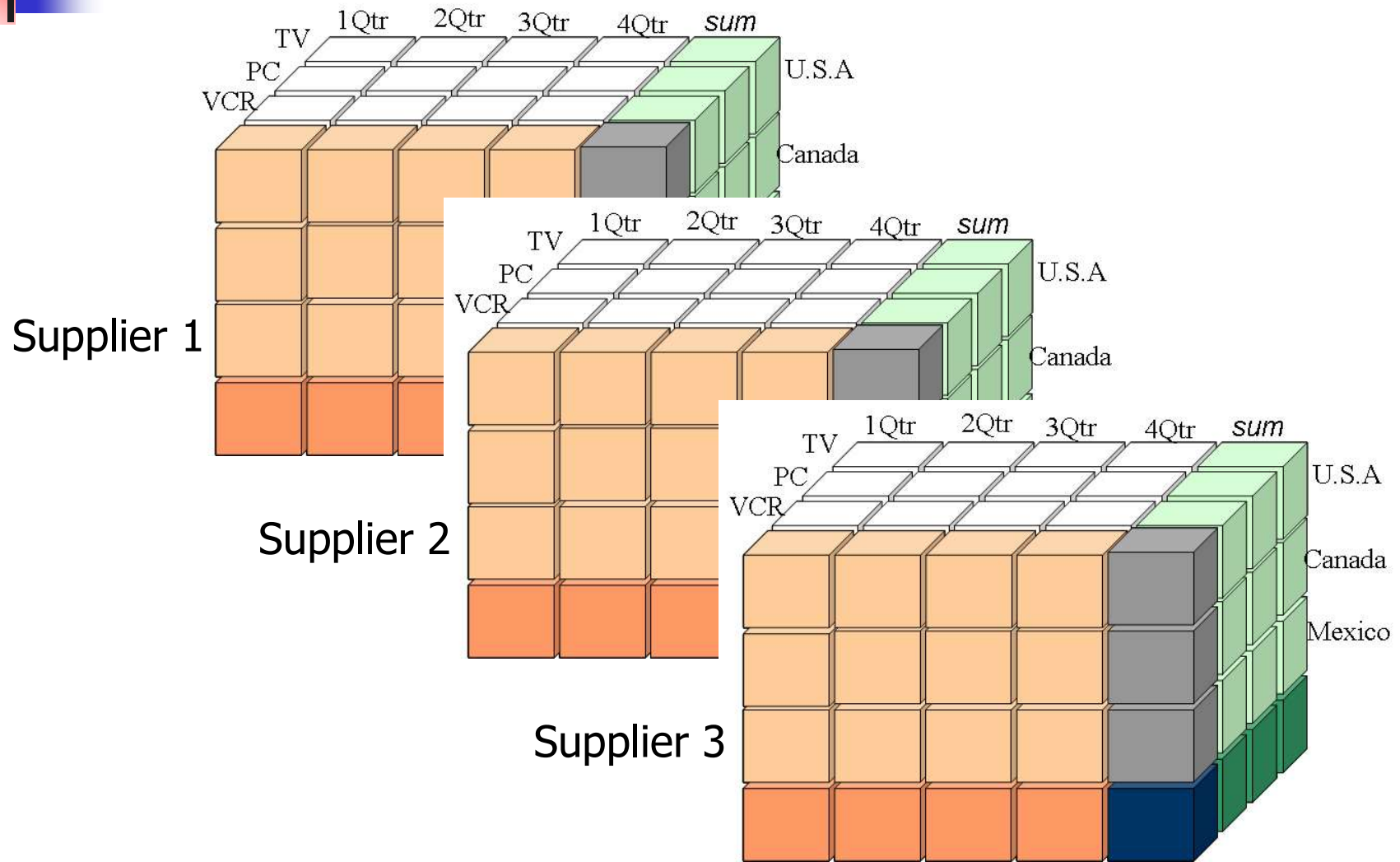
- A data warehouse is based on a **multidimensional data model** which views data in the form of a data cube
- A data cube allows data to be modeled and viewed in multiple dimensions **(chú ý sự khác nhau giữa data cube và cuboid)**
 - **Dimension tables**, such as item (item_name, brand, type), or time(day, week, month, quarter, year)
 - **Fact table** contains measures (such as dollars_sold) and keys to each of the related dimension tables
- In data warehousing literature, an n-D base cube is called a **base cuboid**. The top most 0-D cuboid, which holds the highest-level of summarization, is called the **apex cuboid**. The lattice of cuboids forms a **data cube**.

A Sample Data Cube

3 dimension
Độ đo là lượng
hàng bán được

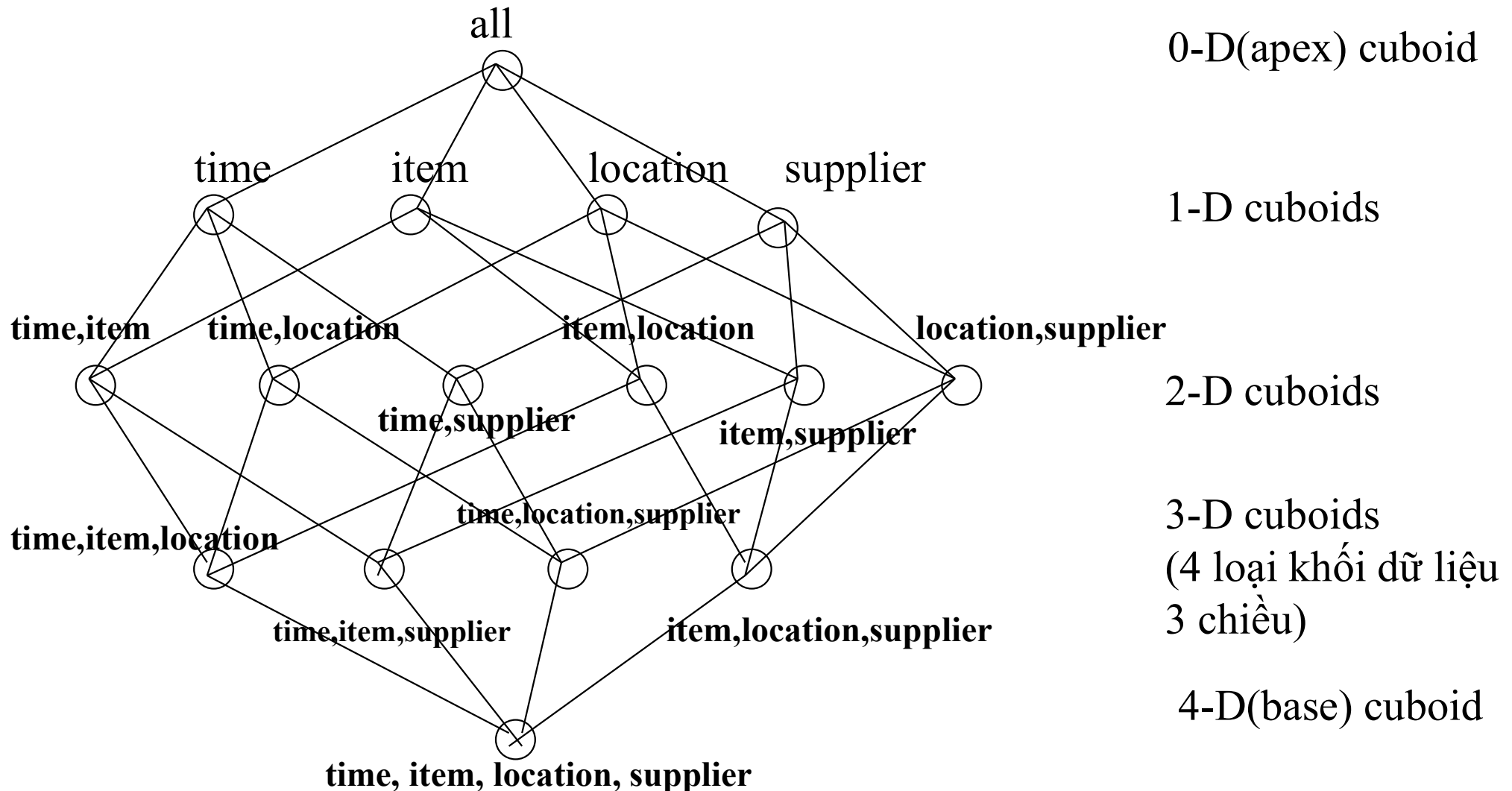


4-D Data Cube





Cube: A Lattice of Cuboids





Conceptual Modeling of Data Warehouses

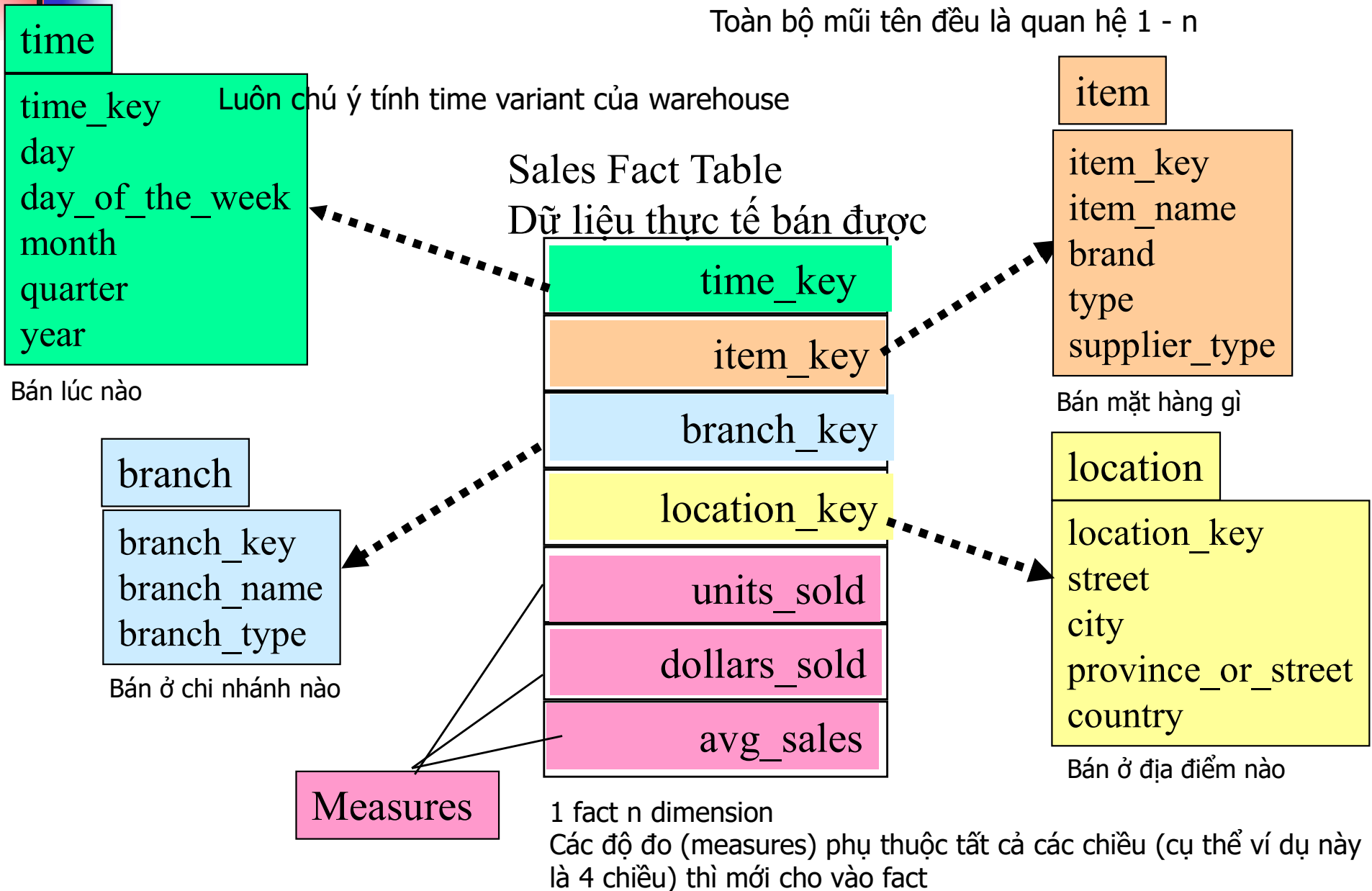
- Modeling data warehouses: dimensions & measures
 - **Star schema**: A fact table in the middle connected to a set of dimension tables
 - **Snowflake schema**: A refinement of star schema where some dimensional hierarchy is normalized into a set of smaller dimension tables, forming a shape similar to snowflake
 - **Fact constellations**: Multiple fact tables share dimension tables, viewed as a collection of stars, therefore called galaxy schema or fact constellation

Fact là table chủ yếu được đổ dữ liệu nhiều nhất

Warehouse chỉ lưu summarised data, không lưu chi tiết như CSDL tích hợp

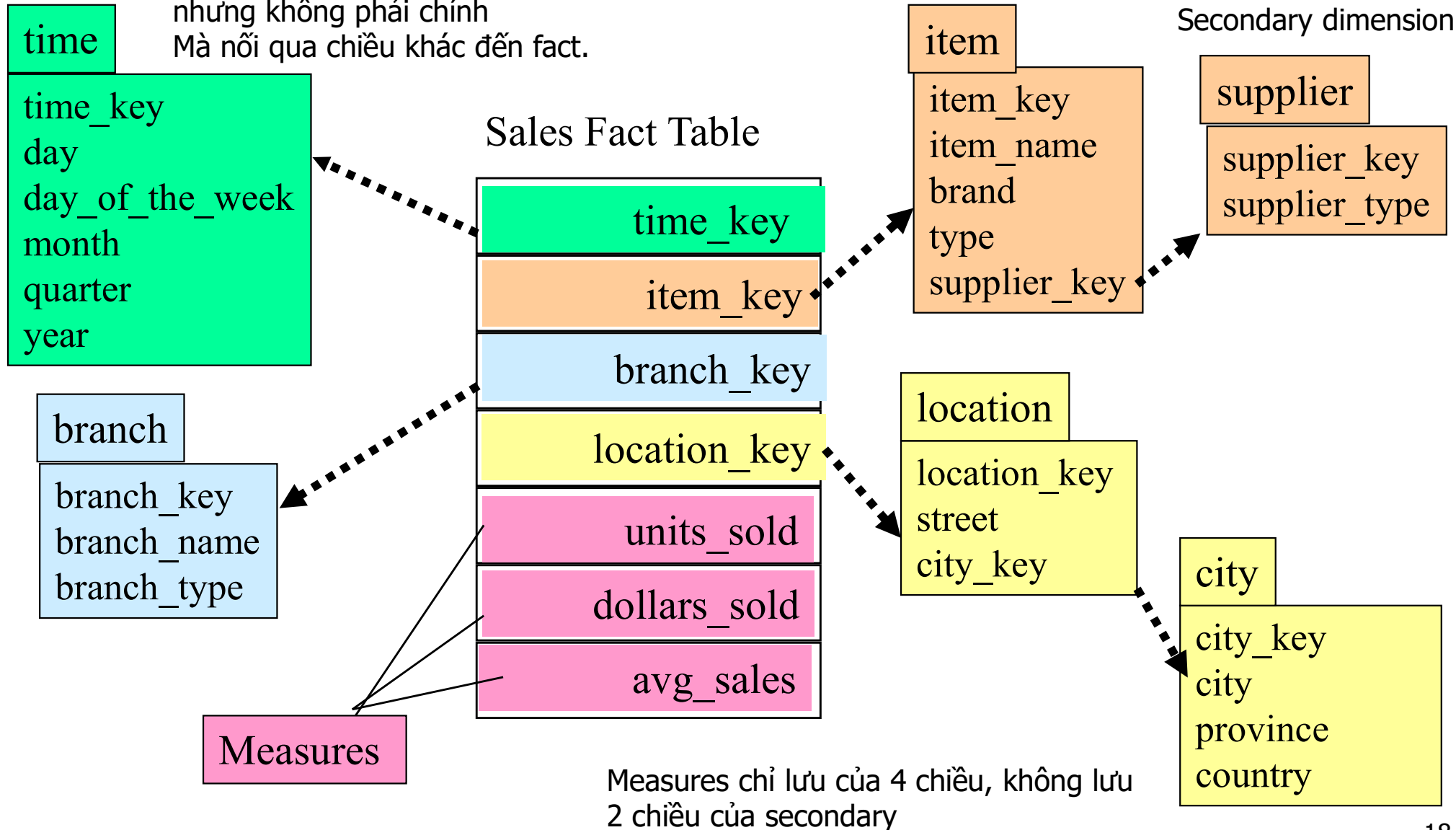
Example of Star Schema

Toàn bộ mũi tên đều là quan hệ 1 - n



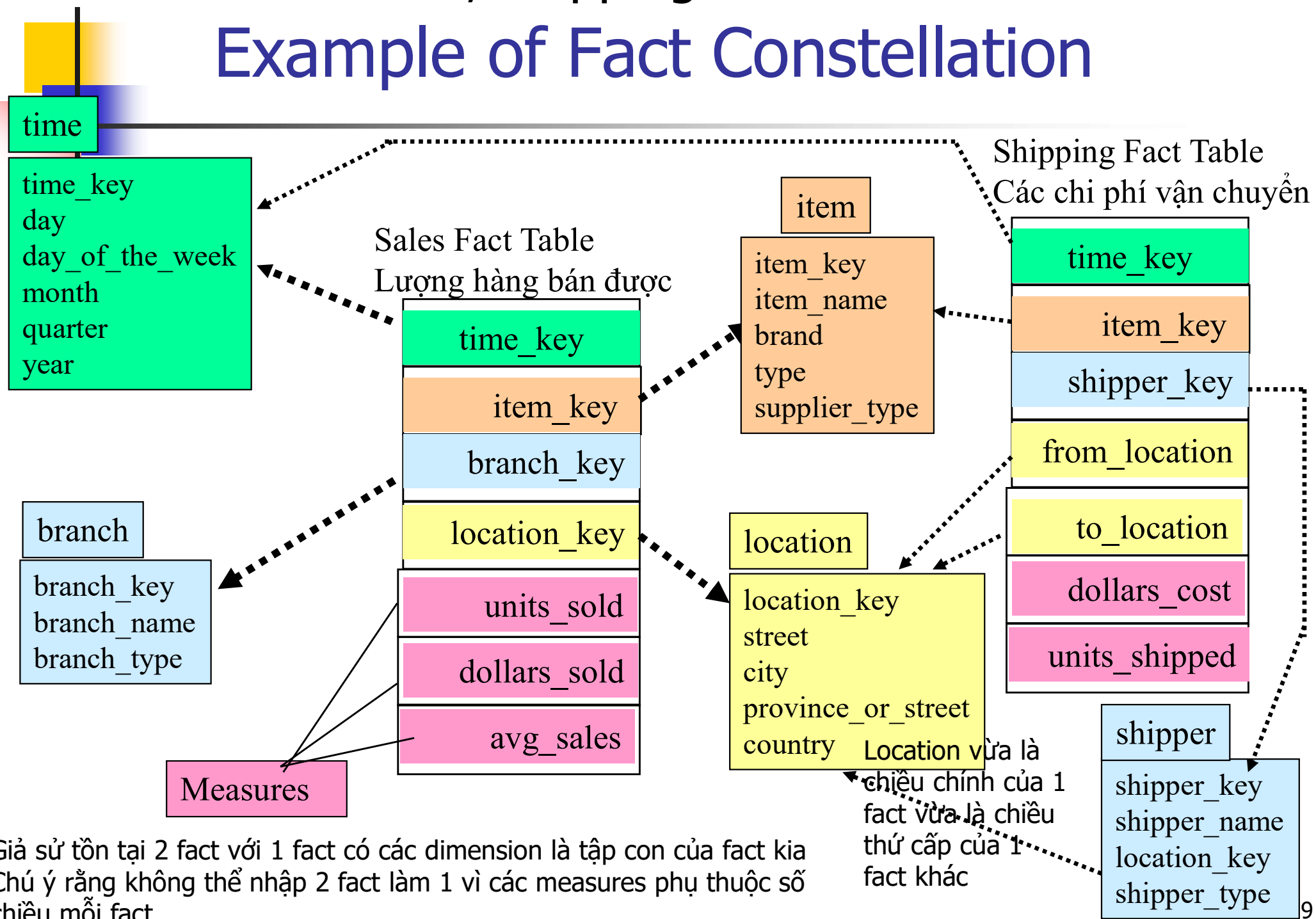
Example of Snowflake Schema

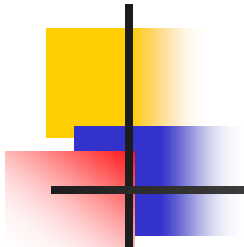
Các chiều thứ cấp vẫn tính là chiều (lược đồ 6 chiều),
nhưng không phải chính
Mà nối qua chiều khác đến fact.



Sales 4 chiều, Shipping 5 chiều

Example of Fact Constellation





A Data Mining Query Language, DMQL: Language Primitives

- Cube Definition (Fact Table)
`define cube <cube_name> [<dimension_list>]:
 <measure_list>`
- Dimension Definition (Dimension Table)
`define dimension <dimension_name> as
 (<attribute_or_subdimension_list>)`
- Special Case (Shared Dimension Tables)
 - First time as "cube definition"
 - `define dimension <dimension_name> as
 <dimension_name_first_time> in cube
 <cube_name_first_time>`



Defining a Star Schema in DMQL

```
define cube sales_star [time, item, branch, location]:  
    dollars_sold = sum(sales_in_dollars), avg_sales =  
        avg(sales_in_dollars), units_sold = count(*)  
define dimension time as (time_key, day, day_of_week,  
    month, quarter, year)  
define dimension item as (item_key, item_name, brand,  
    type, supplier_type)  
define dimension branch as (branch_key, branch_name,  
    branch_type)  
define dimension location as (location_key, street, city,  
    province_or_state, country)
```



Defining a Snowflake Schema in DMQL

```
define cube sales_snowflake [time, item, branch, location]:  
    dollars_sold = sum(sales_in_dollars), avg_sales =  
        avg(sales_in_dollars), units_sold = count(*)  
  
define dimension time as (time_key, day, day_of_week,  
    month, quarter, year)  
  
define dimension item as (item_key, item_name, brand, type,  
    supplier(supplier_key, supplier_type))  
  
define dimension branch as (branch_key, branch_name,  
    branch_type)  
  
define dimension location as (location_key, street,  
    city(city_key, province_or_state, country))
```



Defining a Fact Constellation in DMQL

```
define cube sales [time, item, branch, location]:  
    dollars_sold = sum(sales_in_dollars), avg_sales =  
        avg(sales_in_dollars), units_sold = count(*)  
define dimension time as (time_key, day, day_of_week, month, quarter, year)  
define dimension item as (item_key, item_name, brand, type, supplier_type)  
define dimension branch as (branch_key, branch_name, branch_type)  
define dimension location as (location_key, street, city, province_or_state,  
    country)  
define cube shipping [time, item, shipper, from_location, to_location]:  
    dollar_cost = sum(cost_in_dollars), unit_shipped = count(*)  
define dimension time as time in cube sales  
define dimension item as item in cube sales  
define dimension shipper as (shipper_key, shipper_name, location as location  
    in cube sales, shipper_type)  
define dimension from_location as location in cube sales  
define dimension to_location as location in cube sales
```



Measures: Three Categories

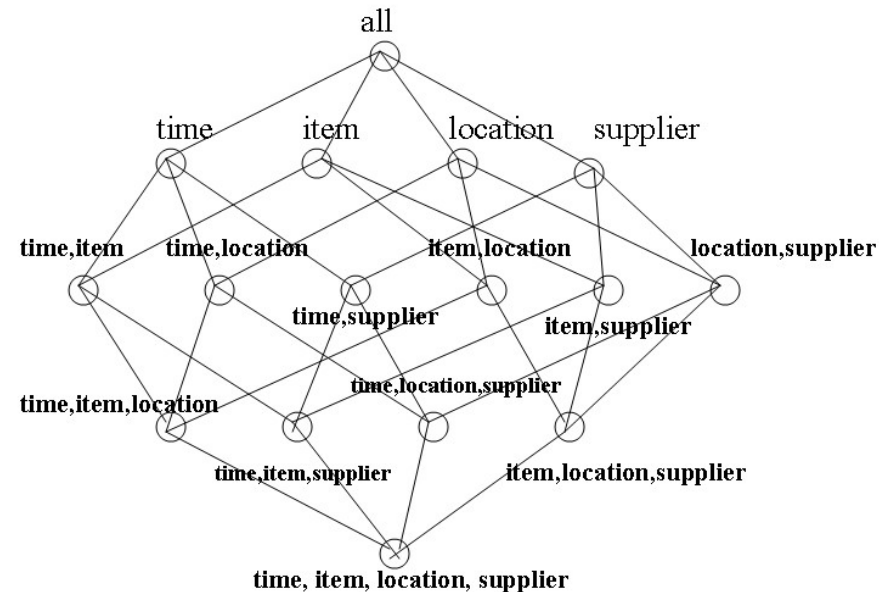
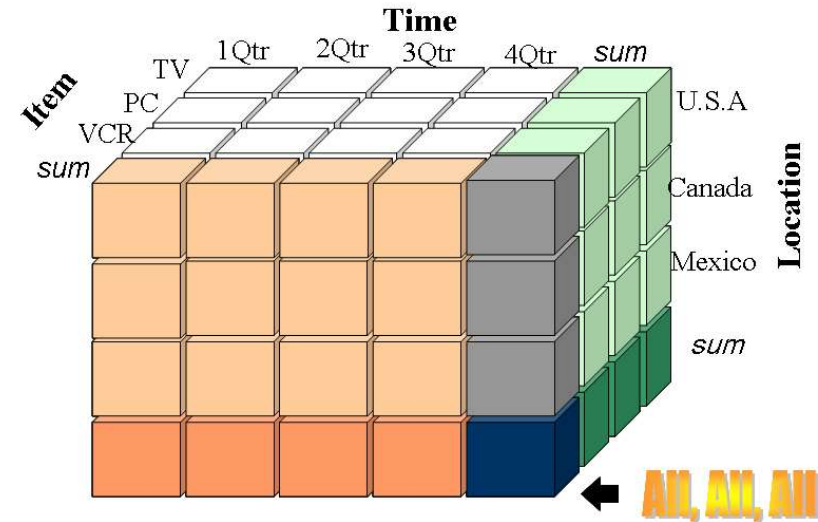
Measure: a function evaluated on aggregated data corresponding to given dimension-value pairs.

Measures can be:

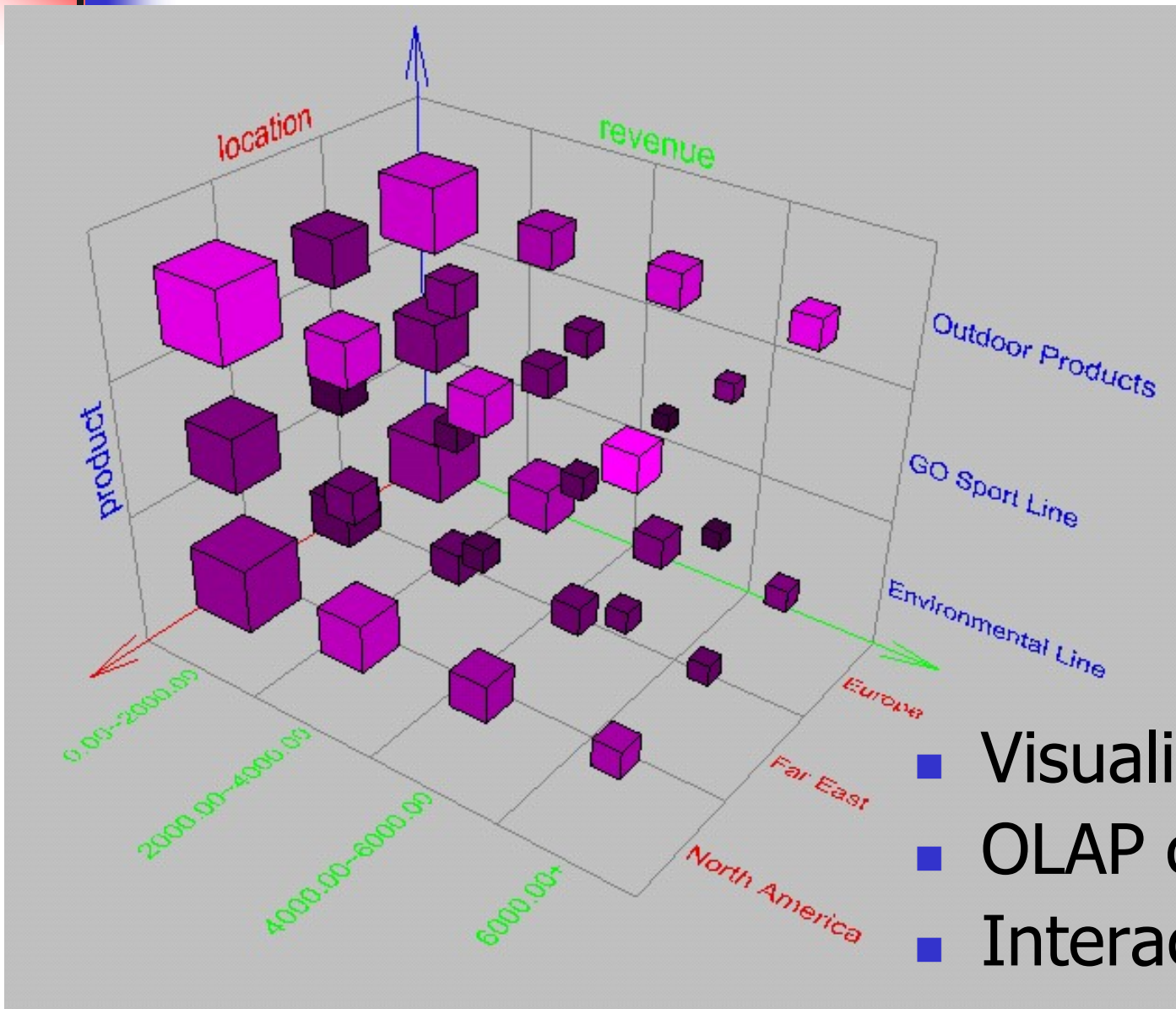
- distributive: if the measure can be calculated in a distributive manner.
 - E.g., `count()`, `sum()`, `min()`, `max()`.
- algebraic: if it can be computed from arguments obtained by applying distributive aggregate functions.
 - E.g., `avg()=sum()/count()`, `min_N()`, `standard_deviation()`.
- holistic: if it is not algebraic.
 - E.g., `median()`, `mode()`, `rank()`.

Measures: Three Categories

- Distributive and algebraic measures are ideal for data cubes.
- Calculated measures at **lower** levels can be used directly at **higher** levels.
- Holistic measures can be difficult to calculate efficiently.
- Holistic measures could often be efficiently **approximated**.



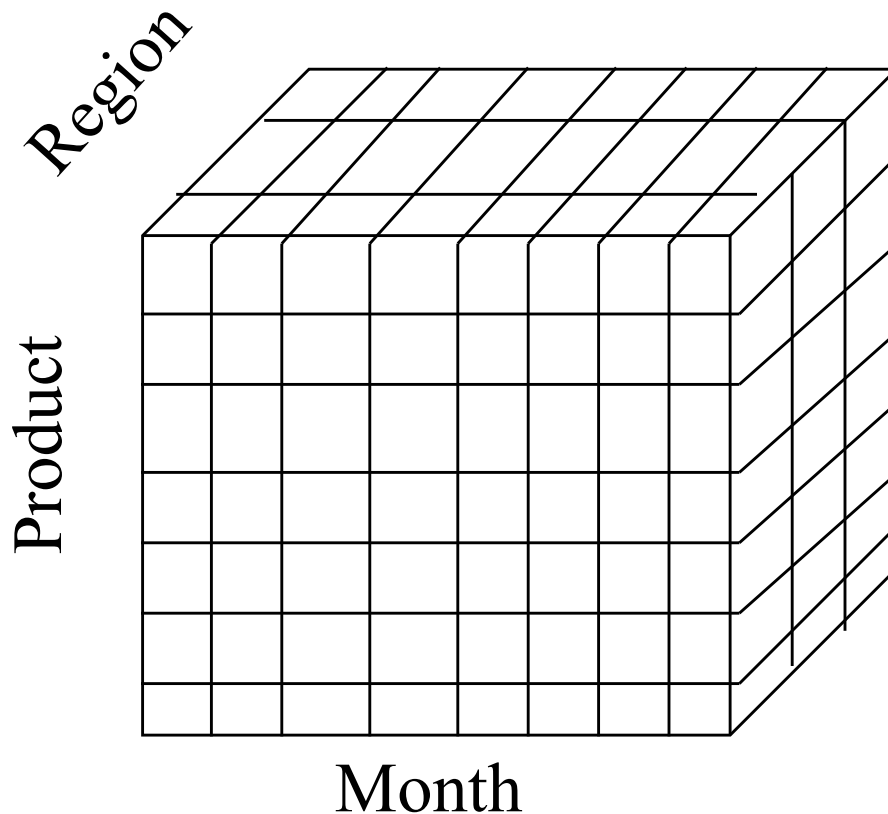
Browsing a Data Cube



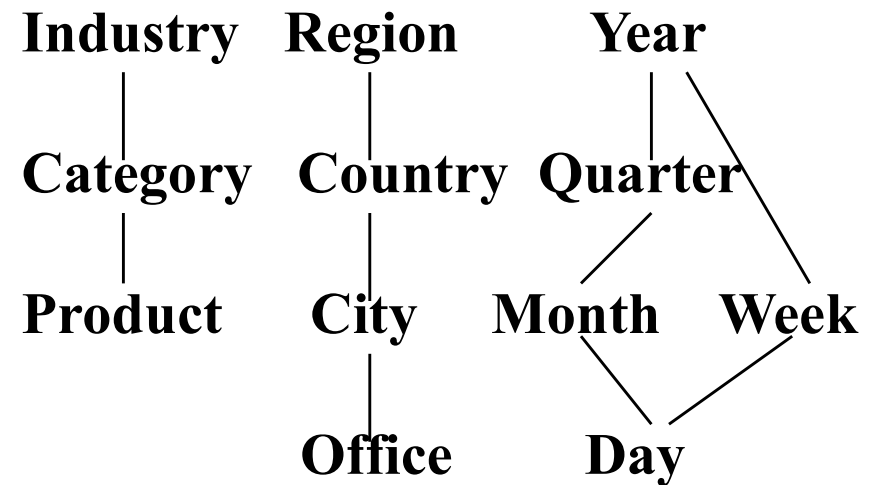
- Visualization
- OLAP capabilities
- Interactive manipulation

A Concept Hierarchy

- Concept hierarchies allow data to be handled at varying levels of abstraction



Dimensions: Product, Location, Time
Hierarchical summarization paths



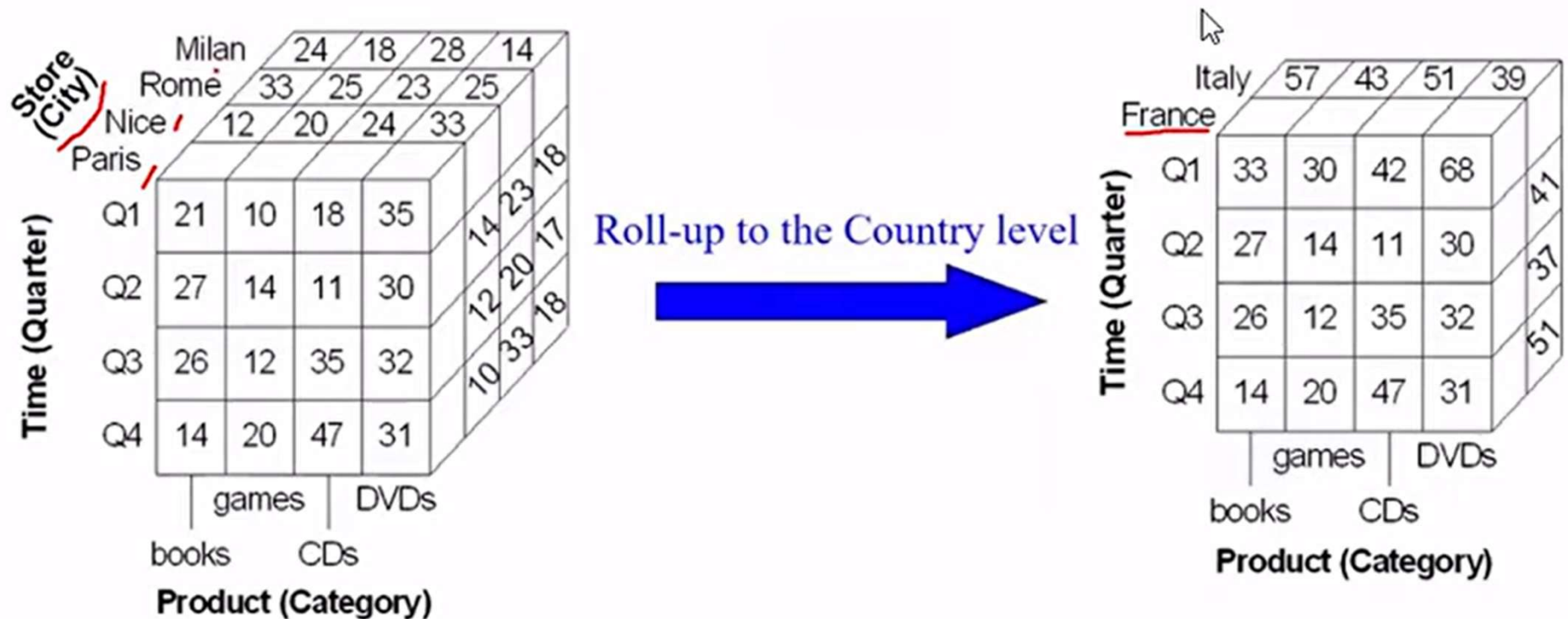
Demo BTL chỉ cần từ Roll up -> Pivot

Typical OLAP Operations (Fig 2.10)

- **Roll up (drill-up):** summarize data (tăng cấp từ thấp lên cao trong 1 phân cấp của 1 chiều hoặc giảm số chiều)
 - *by climbing up concept hierarchy or by dimension reduction*
- **Drill down (roll down):** reverse of roll-up (giảm cấp của 1 chiều hoặc tăng số chiều)
 - *from higher level summary to lower level summary or detailed data, or introducing new dimensions*
- **Dice and slice:**
 - *Project (chọn 1 tập con các cột) and select (1 tập con bản ghi thỏa mãn theo WHERE)*
- **Pivot (rotate):**
 - *reorient the cube, visualization, 3D to series of 2D planes. (hiểu đơn giản hàng -> cột, cột -> hàng)*
- **Other operations**
 - *drill across: involving (across) more than one fact table*
 - *drill through: through the bottom level of the cube to its back-end relational tables (using SQL)*

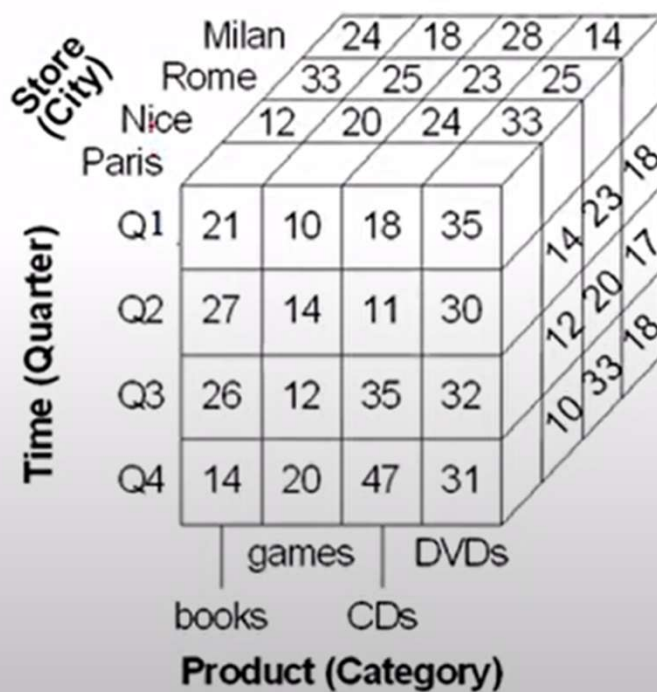
Roll up

- Transforms detailed measures into summarized ones when one moves up in a hierarchy

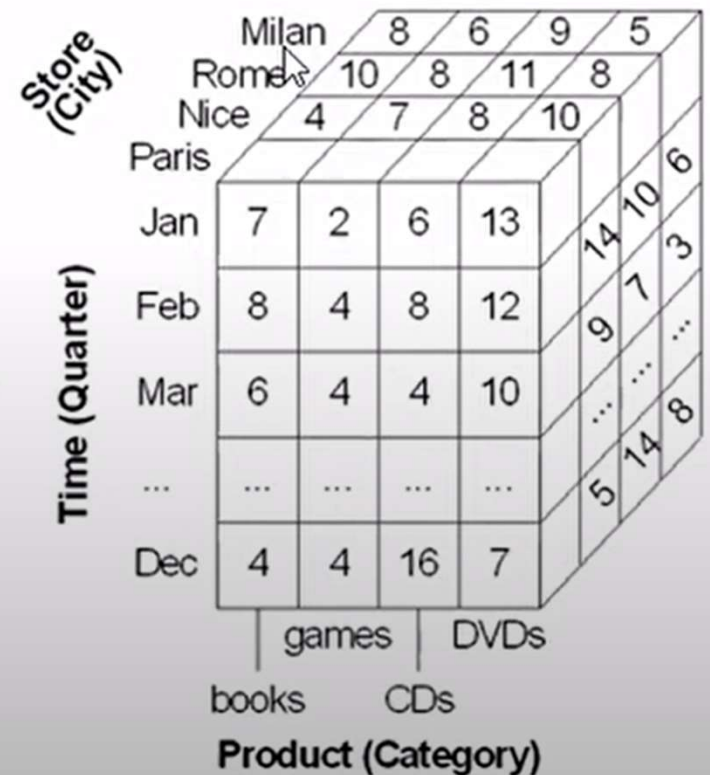


Drill down

- Opposite to the roll-up operation, i.e., it moves from a more general level to a detailed level in a hierarchy

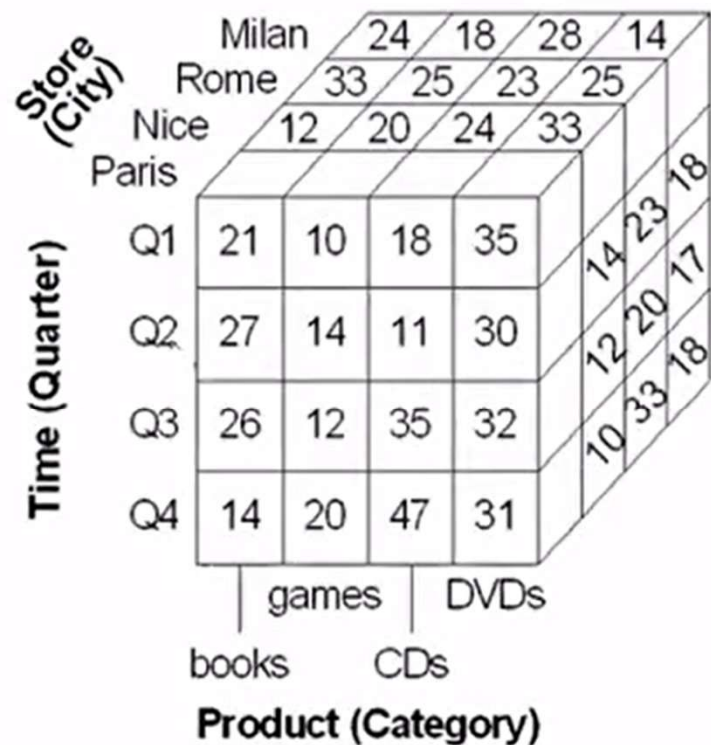


Drill-down to the Month level



Slice

- Performs a selection on one dimension of a cube, resulting in a subcube

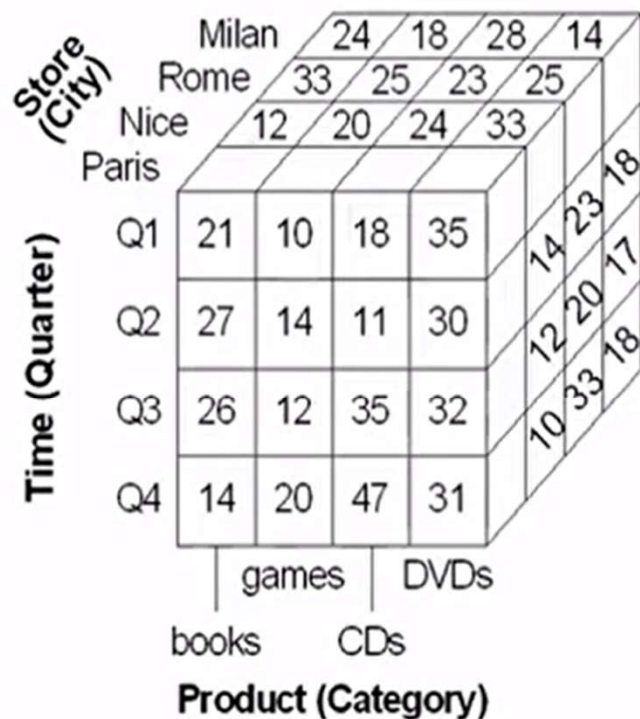


Slice on Store.City = 'Paris'

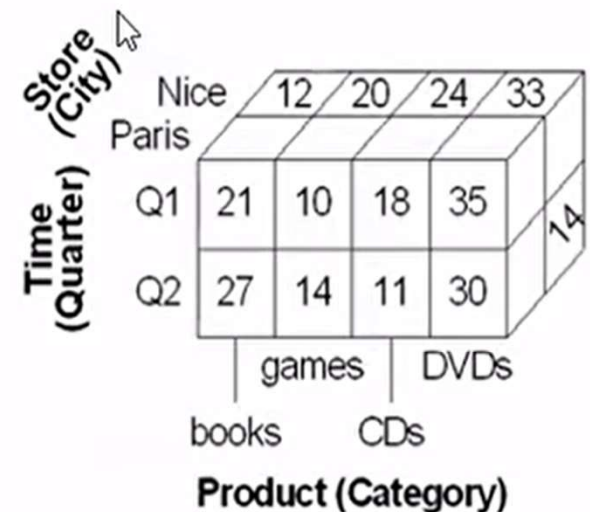
| Time (Quarter) | Product (Category) | | | |
|----------------|--------------------|-----|-------|------|
| | books | CDs | games | DVDs |
| Q1 | 21 | 10 | 18 | 35 |
| Q2 | 27 | 14 | 11 | 30 |
| Q3 | 26 | 12 | 35 | 32 |
| Q4 | 14 | 20 | 47 | 31 |

Dice

- Defines a selection on two or more dimensions, thus again defining a subcube

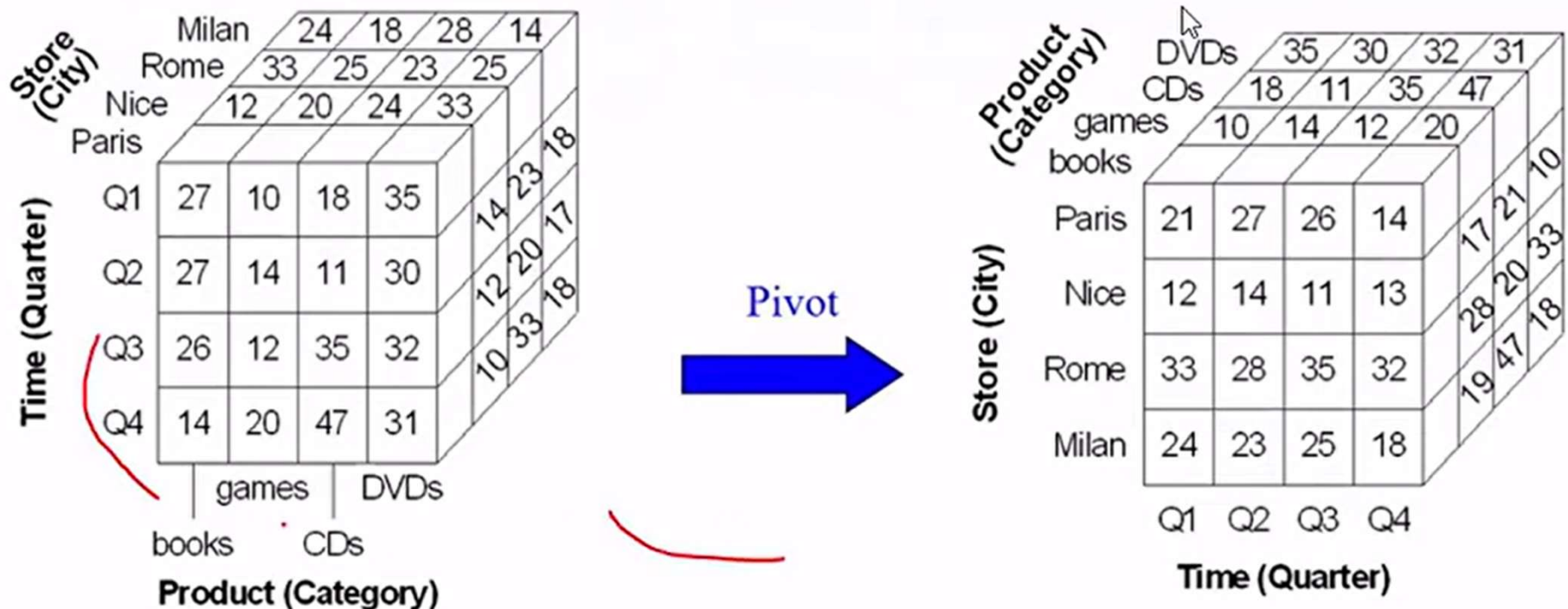


Dice on Store.Country = 'France'
and Time.Quarter = 'Q1' or 'Q2'



Pivot

- Rotates the axes of a cube to provide an alternative presentation of the data

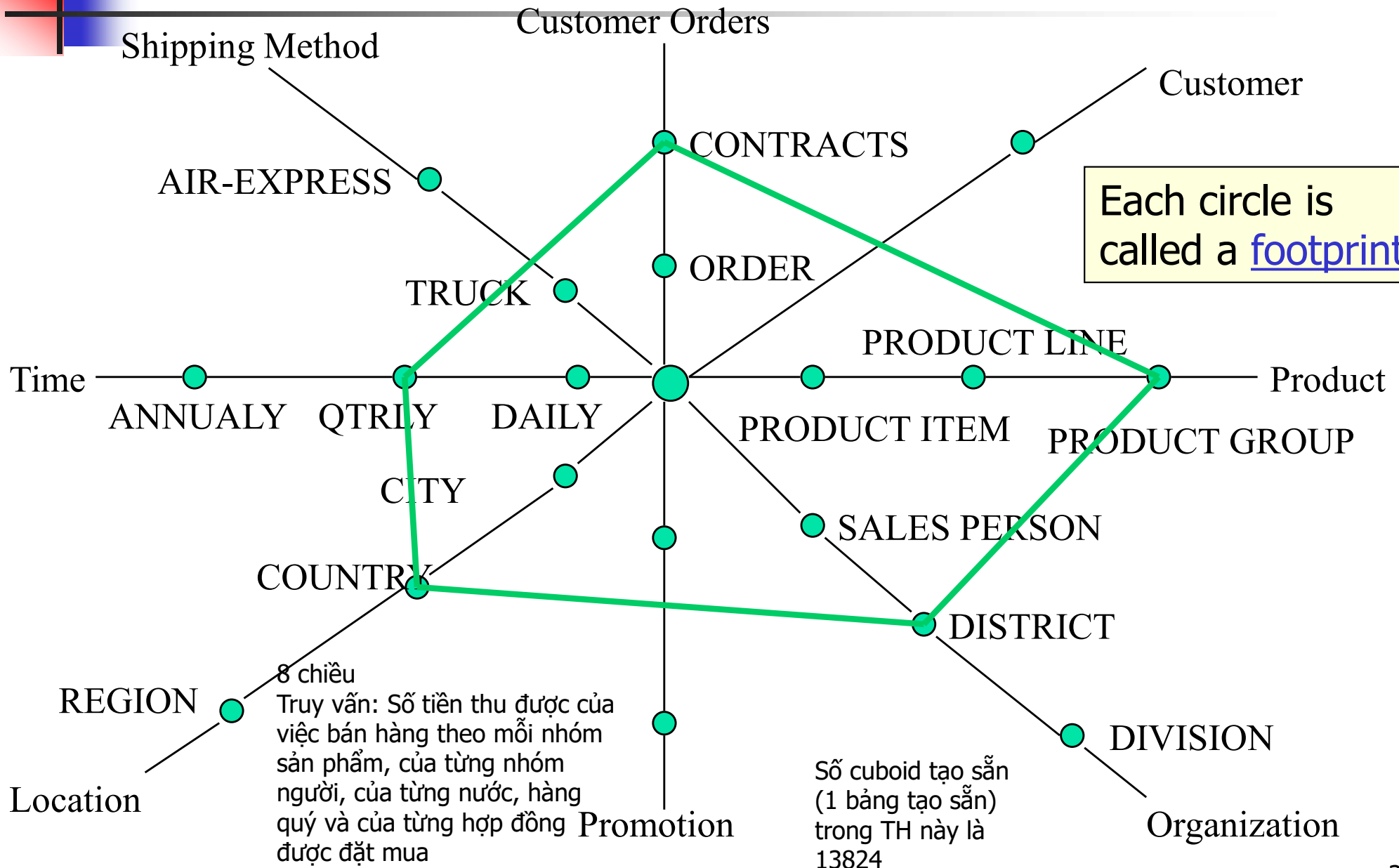




OLAP Operation - Summary

| Operation | Purpose | Description |
|------------|--|--|
| Slice | Focus attention on a subset of dimensions | Replace a dimension with a single member value or with a summary of its measure values |
| Dice | Focus attention on a subset of member values | Replace a dimension with a subset of members |
| Drill-down | Obtain more detail about a dimension | Navigate from a more general level to a more specific level |
| Roll-up | Summarize details about a dimension | Navigate from a more specific level to a more general level |
| Pivot | Present data in a different order | Rearrange the dimensions in a data cube |

Querying Using a Star-Net Model





Ví dụ truy vấn

Tính số lượng hàng bán được theo bang (State).

Cho rằng 3 chiều là Location (Đang đứng ở Store (trên là State)), Time (Đang đứng ở Month (trên là Quarter)), Item (Đang đứng ở số hàng bán được với mỗi loại item, trong từng tháng và trong từng cửa hàng (3 chiều)).

Truy vấn trên là 1 chiều.

Ta phải cuộn lên Item (giảm 1 chiều), cuộn lên Time (giảm 1 chiều).

Thành 1 chiều nhưng đang ở từng cửa hàng (Store) =>

Cuộn lên lần nữa tăng phân cấp

=> 3 phép toán, tức 3 cube, mỗi lần cuộn lên là chọn vào 1 cube khác để lấy ra màn hình



Chapter 2: Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation
- Further development of data cube technology
- From data warehousing to data mining

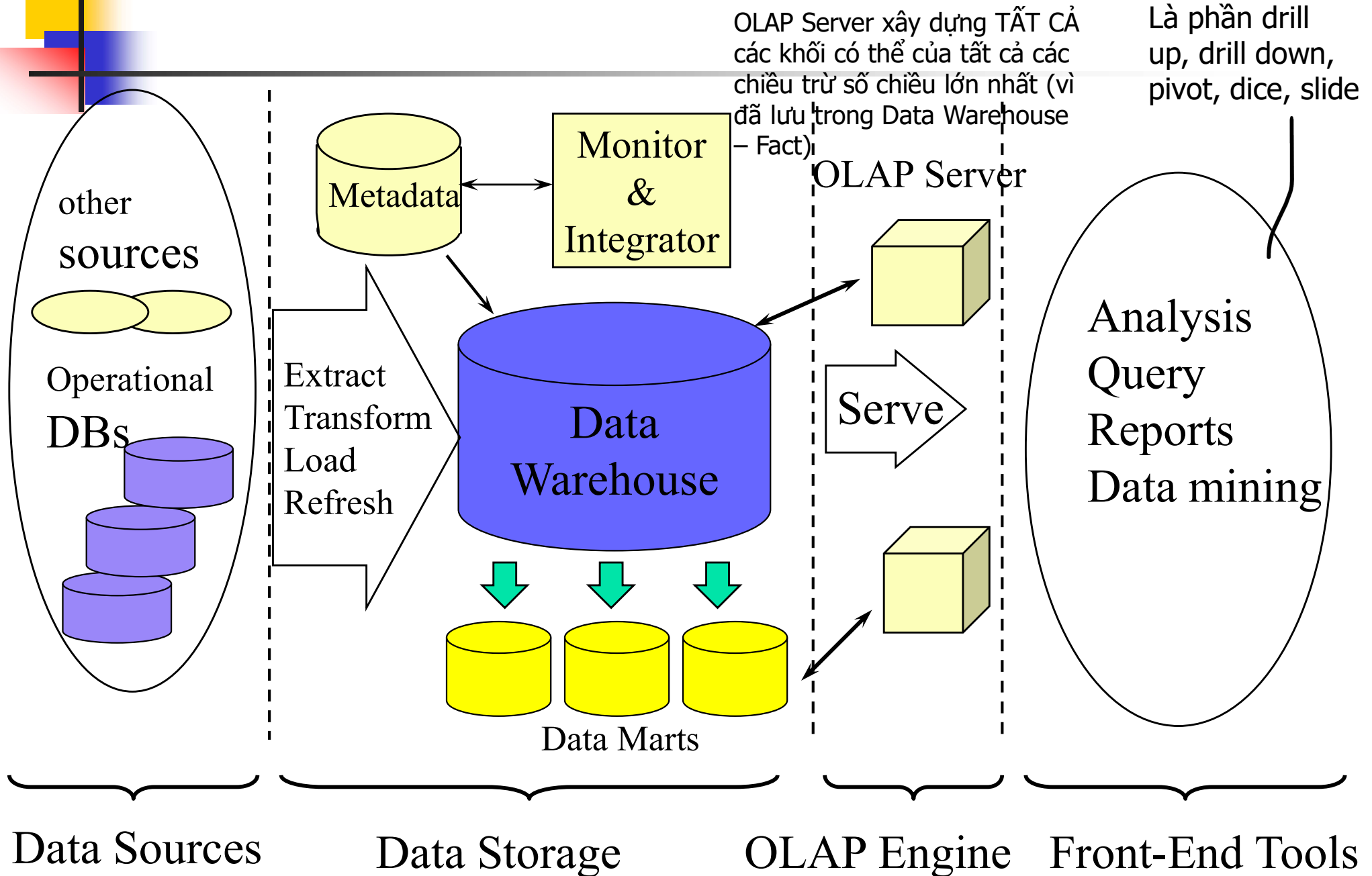


Data Warehouse Design Process

- Top-down, bottom-up approaches or a combination of both
 - Top-down: Starts with overall design and planning (mature)
 - Bottom-up: Starts with experiments and prototypes (rapid)
- From software engineering point of view
 - Waterfall: structured and systematic analysis at each step before proceeding to the next
 - Spiral: rapid generation of increasingly functional systems, quick modifications, timely adaptation of new designs and technologies
- Typical data warehouse design process
 - Choose a **business process** to model, e.g., orders, invoices, etc.
 - Choose the ***grain (atomic level of data)*** of the business process
 - Choose the **dimensions** that will apply to each fact table record
 - Choose the **measure** that will populate each fact table record

DW và OLAP Server sẽ trên 2 server khác nhau, nhưng demo btl chỉ cần 1 máy (1 server)

Multi-Tiered Architecture





Three Data Warehouse Models

- **Enterprise warehouse**

- collects all of the information about subjects spanning the entire organization

- **Data Mart**

- a subset of corporate-wide data that is of value to a specific groups of users. Its scope is confined to specific, selected groups, such as marketing data mart
 - Independent vs. dependent (directly from warehouse) data mart

- **Virtual warehouse**

- A set of views over operational databases
- Only some of the possible summary views may be materialized

OLAP Server Architectures

Relational OLAP (ROLAP)

- Use relational or extended-relational DBMS to store and manage warehouse data
- Include optimization of DBMS backend and additional tools and services
- greater scalability

Multidimensional OLAP (MOLAP)


- Array-based multidimensional storage engine (sparse matrix techniques)
- fast indexing to pre-computed summarized data

Hybrid OLAP (HOLAP)

- User flexibility (low level: relational, high-level: array)

Specialized SQL servers

- specialized support for SQL queries over star/snowflake schemas



| pid | timeid | locid | sales |
|-----|--------|-------|-------|
| 11 | 1 | 1 | 25 |
| 11 | 2 | 1 | 8 |
| 11 | 3 | 1 | 15 |
| 12 | 1 | 1 | 30 |
| 12 | 2 | 1 | 20 |
| 12 | 3 | 1 | 50 |
| 13 | 1 | 1 | 8 |
| 13 | 2 | 1 | 10 |
| 13 | 3 | 1 | 10 |
| 11 | 1 | 2 | 35 |

meid).

| pid | timeid | locid | sales |
|-----|--------|-------|-------|
| 11 | 1 | 1 | 25 |
| 11 | 2 | 1 | 8 |
| 11 | 3 | 1 | 15 |
| 12 | 1 | 1 | 30 |
| 12 | 2 | 1 | 20 |
| 12 | 3 | 1 | 50 |
| 13 | 1 | 1 | 8 |
| 13 | 2 | 1 | 10 |
| 13 | 3 | 1 | 10 |
| 11 | 1 | 2 | 35 |



Chapter 2: Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation
- Further development of data cube technology
- From data warehousing to data mining



Efficient Data Cube Computation

- Data cube can be viewed as a lattice of cuboids
 - Khối dữ liệu ở đáy dưới cùng của khối dữ liệu được xem là khối cơ sở
 - Khối trên đỉnh cao nhất của khối dữ liệu chỉ chứa 1 ô
 - Chúng ta cùng xác định xem có bao nhiêu khối lập phương trong khối dữ liệu n chiều với mỗi chiều có L mức phân cấp khác nhau. Ta có chiều thứ (i) có Li mức nên nhận Li + 1 giá trị. Vì thế tổng số khối lập n chiều là
$$T = \prod_{i=1}^n (L_i + 1)$$

Cube Operation

- Cube definition and computation in DMQL

define cube sales[item, city, year]: sum(sales_in_dollars)

compute cube sales

- Transform it into a SQL-like language (with a new operator **cube by**, introduced by Gray et al.'96)

SELECT item, city, year, SUM (amount)

FROM SALES

CUBE BY item, city, year

- Need compute the following Group-Bys

(date, product, customer),

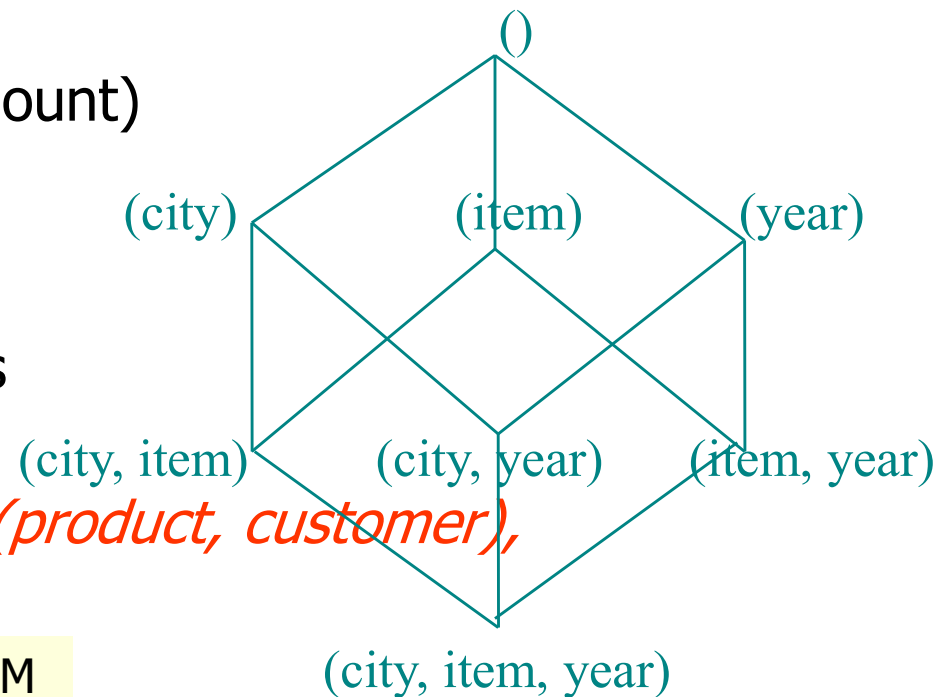
(date, product), (date, customer), (product, customer),

(date), (product), (customer)

()

```
SELECT item, city, year, SUM  
(amount)  
FROM SALES
```

GROUP BY item, year



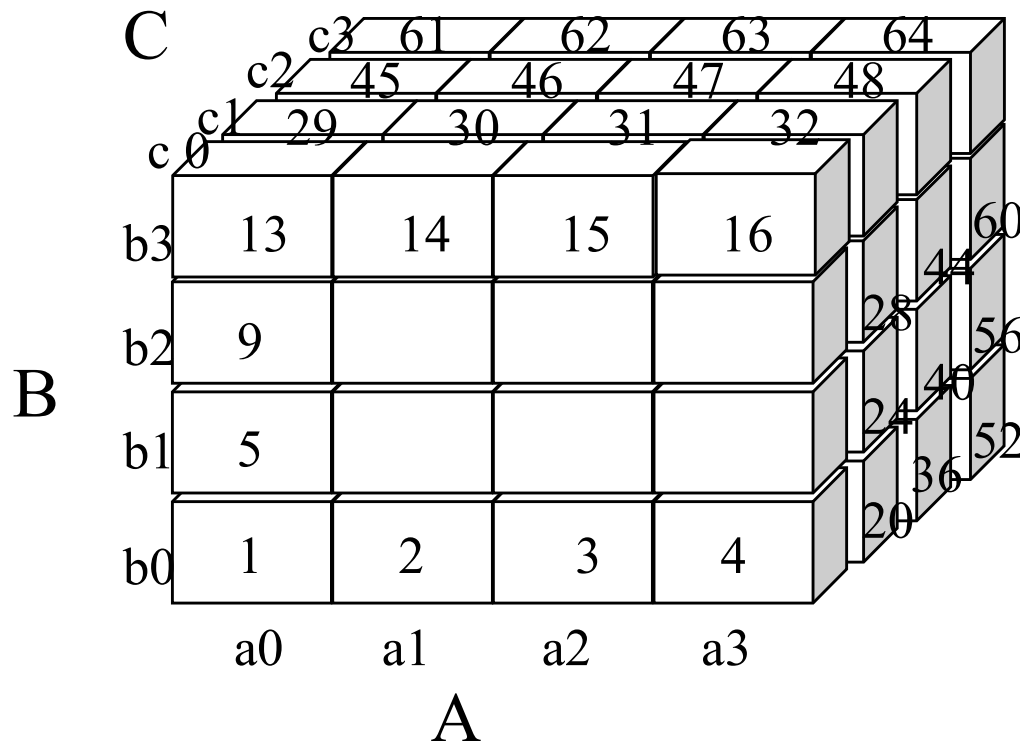


Cube Computation: ROLAP vs. MOLAP

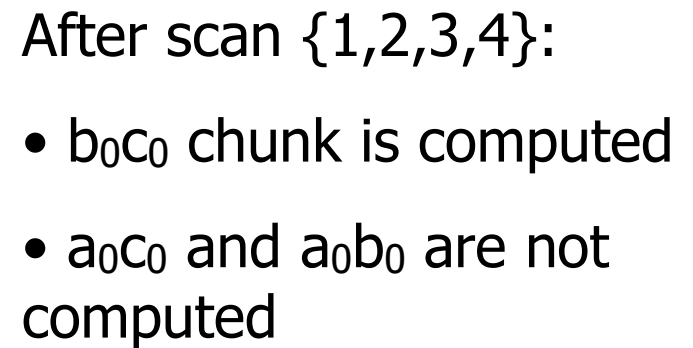
- ROLAP-based cubing algorithms
 - Key-based addressing
 - Sorting, hashing, and grouping operations are applied to the dimension attributes to reorder and cluster related tuples
 - Aggregates may be computed from previously computed aggregates, rather than from the base fact table
- MOLAP-based cubing algorithms
 - Direct array addressing
 - Partition the array into chunks that fit the memory
 - Compute aggregates by visiting cube chunks
 - Possible to exploit ordering of chunks for faster calculation

Multiway Array Aggregation for MOLAP

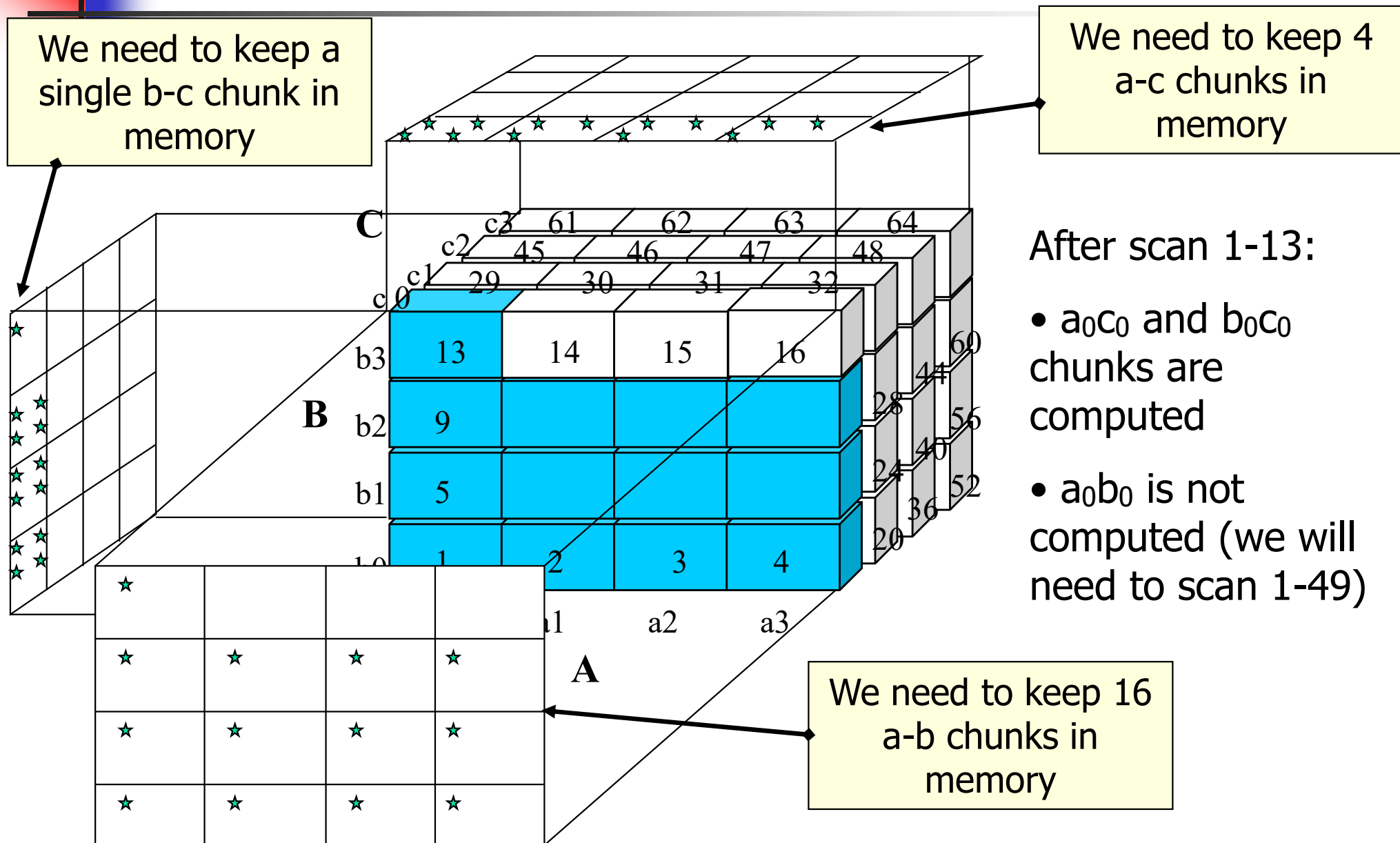
- Partition arrays into chunks (a small subcube which fits in memory).
- Compressed sparse array addressing: (chunk_id, offset)
- Compute aggregates in “multiway” by visiting cube cells in the order which minimizes the # of times to visit each cell, and reduces memory access and storage cost.



What is the best traversing order to do multi-way aggregation?



Multiway Array Aggregation for MOLAP





Multiway Array Aggregation for MOLAP

- Method: the planes should be sorted and computed according to their size in ascending order.
 - The proposed scan is optimal if $|C| > |B| > |A|$
 - See the details of Example 2.12 (pp. 75-78)
- MOLAP cube computation is faster than ROLAP
- Limitation of MOLAP: computing well only for a small number of dimensions
- If there are a large number of dimensions use the iceberg cube computation: process only “dense” chunks

Indexing OLAP Data: Bitmap Index

- Suitable for low cardinality domains
- Index on a particular column
- Each value in the column has a bit vector: bit-op is fast
- The length of the bit vector: # of records in the base table
- The i -th bit is set if the i -th row of the base table has the value for the indexed column

Base table

| Cust | Region | Type |
|------|---------|--------|
| C1 | Asia | Retail |
| C2 | Europe | Dealer |
| C3 | Asia | Dealer |
| C4 | America | Retail |
| C5 | Europe | Dealer |

Index on Region

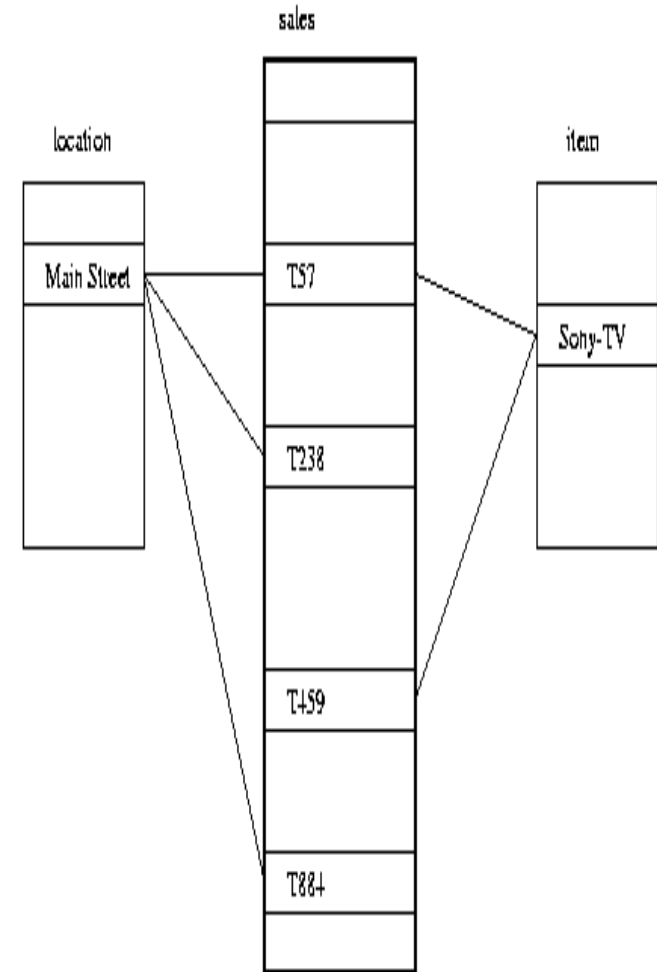
| RecID | Asia | Europe | America |
|-------|------|--------|---------|
| 1 | 1 | 0 | 0 |
| 2 | 0 | 1 | 0 |
| 3 | 1 | 0 | 0 |
| 4 | 0 | 0 | 1 |
| 5 | 0 | 1 | 0 |

Index on Type

| RecID | Retail | Dealer |
|-------|--------|--------|
| 1 | 1 | 0 |
| 2 | 0 | 1 |
| 3 | 0 | 1 |
| 4 | 1 | 0 |
| 5 | 0 | 1 |

Indexing OLAP Data: Join Indices

- **Join index** materializes relational join and speeds up relational join — a rather costly operation
- In data warehouses, join index relates the values of the **dimensions** of a star schema to **rows** in the fact table.
 - E.g. fact table: *Sales* and two dimensions *location* and *item*
 - A join index on *location* is a list of pairs $\langle \text{loc_name}, T_id \rangle$ sorted by location
 - A join index on *location-and-item* is a list of triples $\langle \text{loc_name}, \text{item_name}, T_id \rangle$ sorted by location and item names
- Search of a join index can still be slow
- **Bitmapped join index** allows speed-up by using bit vectors instead of dimension attribute names





Online Aggregation

- Consider an aggregate query:
"finding the average sales by state"
- Can we provide the user with some information before the exact average is computed for all states?
 - Solution: show the current "running average" for each state as the computation proceeds.
 - Even better, if we use statistical techniques and sample tuples to aggregate instead of simply scanning the aggregated table, we can provide bounds such as "the average for Wisconsin is 2000 ± 102 with 95% probability."



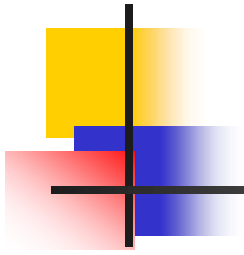
Efficient Processing of OLAP Queries

- Determine which operations should be performed on the available cuboids:
 - transform drill, roll, etc. into corresponding SQL and/or OLAP operations, e.g, dice = selection + projection
- Determine to which materialized cuboid(s) the relevant operations should be applied.
- Exploring indexing structures and compressed vs. dense array structures in MOLAP (trade-off between indexing and storage performance)



Metadata Repository

- Meta data is the data defining warehouse objects. It has the following kinds
 - Description of the structure of the warehouse
 - schema, view, dimensions, hierarchies, derived data definitions, data mart locations and contents
 - Operational meta-data
 - data lineage (history of migrated data and transformation path), currency of data (active, archived, or purged), monitoring information (warehouse usage statistics, error reports, audit trails)
 - The algorithms used for summarization
 - The mapping from operational environment to the data warehouse (cách ánh xạ từ lược đồ CSDL tích hợp vào DW)
 - Data related to system performance
 - warehouse schema, view and derived data definitions
 - Business data
 - business terms and definitions, ownership of data, charging policies



- BTL có metadata kiểu các bảng ?



Data Warehouse Back-End Tools and Utilities

- Data extraction:
 - get data from multiple, heterogeneous, and external sources
- Data cleaning:
 - detect errors in the data and rectify them when possible
- Data transformation:
 - convert data from legacy or host format to warehouse format
- Load:
 - sort, summarize, consolidate, compute views, check integrity, and build indices and partitions
- Refresh
 - propagate the updates from the data sources to the warehouse



Chapter 2: Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation
- Further development of data cube technology
- From data warehousing to data mining



Discovery-Driven Exploration of Data Cubes

- Hypothesis-driven: exploration by user, huge search space
- Discovery-driven (Sarawagi et al.'98)
 - pre-compute measures indicating exceptions, guide user in the data analysis, at all levels of aggregation
 - Exception: significantly different from the value anticipated, based on a statistical model
 - Visual cues such as background color are used to reflect the degree of exception of each cell
 - Computation of exception indicator can be overlapped with cube construction

Examples: Discovery-Driven Data Cubes

| | |
|--------|-----|
| item | all |
| region | all |

| Sum of sales | month | | | | | | | | | | | |
|--------------|-------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| | Jan | Feb | Mar | Apr | May | Jun | Jul | Aug | Sep | Oct | Nov | Dec |
| Total | | 1% | -1% | 0% | 1% | 3% | -1 | -9% | -1% | 2% | -4% | 3% |

| Avg sales item | month | | | | | | | | | | | |
|-------------------------|-------|-----|-----|-----|-----|-----|------|------|-----|------|------|------|
| | Jan | Feb | Mar | Apr | May | Jun | Jul | Aug | Sep | Oct | Nov | Dec |
| Sony b/w printer | | 9% | -8% | 2% | -5% | 14% | -4% | 0% | 41% | -13% | -15% | -11% |
| Sony color printer | | 0% | 0% | 3% | 2% | 4% | -10% | -13% | 0% | 4% | -6% | 4% |
| HP b/w printer | | -2% | 1% | 2% | 3% | 8% | 0% | -12% | -9% | 3% | -3% | 6% |
| HP color printer | | 0% | 0% | -2% | 1% | 0% | -1% | -7% | -2% | 1% | -5% | 1% |
| IBM home computer | | 1% | -2% | -1% | -1% | 3% | 3% | -10% | 4% | 1% | -4% | -1% |
| IBM laptop computer | | 0% | 0% | -1% | 3% | 4% | 2% | -10% | -2% | 0% | -9% | 3% |
| Toshiba home computer | | -2% | -5% | 1% | 1% | -1% | 1% | 5% | -3% | -5% | -1% | -1% |
| Toshiba laptop computer | | 1% | 0% | 3% | 0% | -2% | -2% | -5% | 3% | 2% | -1% | 0% |
| Logitech mouse | | 3% | -2% | -1% | 0% | 4% | 6% | -11% | 2% | 1% | -4% | 0% |
| Ergo-way mouse | | 0% | 0% | 2% | 3% | 1% | -2% | -2% | -5% | 0% | -5% | 8% |

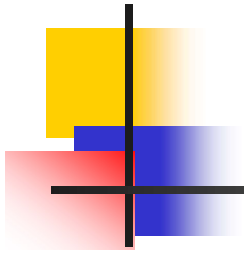
| | |
|------|-------------------|
| item | IBM home computer |
|------|-------------------|

| Avg sales region | month | | | | | | | | | | | |
|---------------------|-------|-----|-----|-----|-----|-----|------|------|------|-----|-----|-----|
| | Jan | Feb | Mar | Apr | May | Jun | Jul | Aug | Sep | Oct | Nov | Dec |
| North | | -1% | -3% | -1% | 0% | 3% | 4% | -7% | 1% | 0% | -3% | -3% |
| South | | -1% | 1% | -9% | 6% | -1% | -39% | 9% | -34% | 4% | 1% | 7% |
| East | | -1% | -2% | 2% | -3% | 1% | 18% | -2% | 11% | -3% | -2% | -1% |
| West | | 4% | 0% | -1% | -3% | 5% | 1% | -18% | 8% | 5% | -8% | 1% |



Chapter 2: Data Warehousing and OLAP Technology for Data Mining

- What is a data warehouse?
- A multi-dimensional data model
- Data warehouse architecture
- Data warehouse implementation
- Further development of data cube technology
- From data warehousing to data mining



- Demo giống Window Explorer ?
- Bấm dấu cộng trái xuống (các thư mục con)
- Kiểu khi bấm vào City thì rải xuống là các bang của nó, chọn bang nào thì hiển thị ra



Data Warehouse Usage

- Three kinds of data warehouse applications
 - Information processing
 - supports querying, basic statistical analysis, and reporting using crosstabs, tables, charts and graphs
 - Analytical processing
 - multidimensional analysis of data warehouse data
 - supports basic OLAP operations, slice-dice, drilling, pivoting
 - Data mining
 - knowledge discovery from hidden patterns
 - supports associations, constructing analytical models, performing classification and prediction, and presenting the mining results using visualization tools.
- Differences among the three tasks



From On-Line Analytical Processing to On Line Analytical Mining (OLAM)

- Why online analytical mining?
 - High quality of data in data warehouses
 - DW contains integrated, consistent, cleaned data
 - Available information processing structure surrounding data warehouses
 - ODBC, OLEDB, Web accessing, service facilities, reporting and OLAP tools
 - OLAP-based exploratory data analysis
 - mining with drilling, dicing, pivoting, etc.
 - On-line selection of data mining functions
 - integration and swapping of multiple mining functions, algorithms, and tasks.
- Architecture of OLAM



Summary

- **Data warehouse**
 - A subject-oriented, integrated, time-variant, and nonvolatile collection of data in support of management's decision-making process
- A **multi-dimensional model** of a data warehouse
 - Star schema, snowflake schema, fact constellations
 - A data cube consists of dimensions & measures
- **OLAP** operations: drilling, rolling, slicing, dicing and pivoting
- OLAP servers: ROLAP, MOLAP, HOLAP
- Efficient computation of data cubes
 - Partial vs. full vs. no materialization
 - Multiway array aggregation
 - Bitmap index and join index implementations
- Further development of data cube technology
 - Discovery-drive and multi-feature cubes
 - From OLAP to OLAM (on-line analytical mining)