

# ***Business Intelligence***

## **Week 3**

### **Data Warehousing**

- Introduction to Data Warehouse
- OLTP vs OLAP
- Data Warehouse Architecture and Models
- ETL
- Data Warehouse Design

**Tilahun Melak(PhD)**



**April, 2026**

# Objectives

At the end of this lecture students will be able to :

- Understand concepts of Data Warehouse
- Identify Data Warehouse Architecture, Design and Models
- Differentiate OLTP and OLAP
- Explain ELT Process

# Introduction

- Data warehouse;
  - 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.
- "A data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile collection of data in support of management's decision-making process."—W. H. Inmon
- Data warehousing:
  - The process of constructing and using data warehouses

# Introduction....

- **Subject-Oriented**
  - Organized around major subjects, such as **customer, product, sales**
  - Focusing on the modeling and analysis of data for decision makers, not on daily operations or transaction processing
  - Provide **a simple and concise** view around particular subject issues by **excluding data that are not useful in the decision support process**

# Introduction....

- **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.

# Introduction....

## ■ 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
  - But the key of operational data may or may not contain "time element"

# Introduction....

- **Nonvolatile**

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

# OLTP vs OLAP

	<b>OLTP</b>	<b>OLAP</b>
<b>users</b>	clerk, IT professional	knowledge worker
<b>function</b>	day to day operations	decision support
<b>DB design</b>	application-oriented	subject-oriented
<b>data</b>	current, up-to-date detailed, flat relational isolated	historical, summarized, multidimensional integrated, consolidated
<b>usage</b>	repetitive	ad-hoc
<b>access</b>	read/write index/hash on prim. key	lots of scans
<b>unit of work</b>	short, simple transaction	complex query
<b># records accessed</b>	tens	millions
<b>#users</b>	thousands	hundreds
<b>DB size</b>	100MB-GB	100GB-TB
<b>metric</b>	transaction throughput	query throughput, response

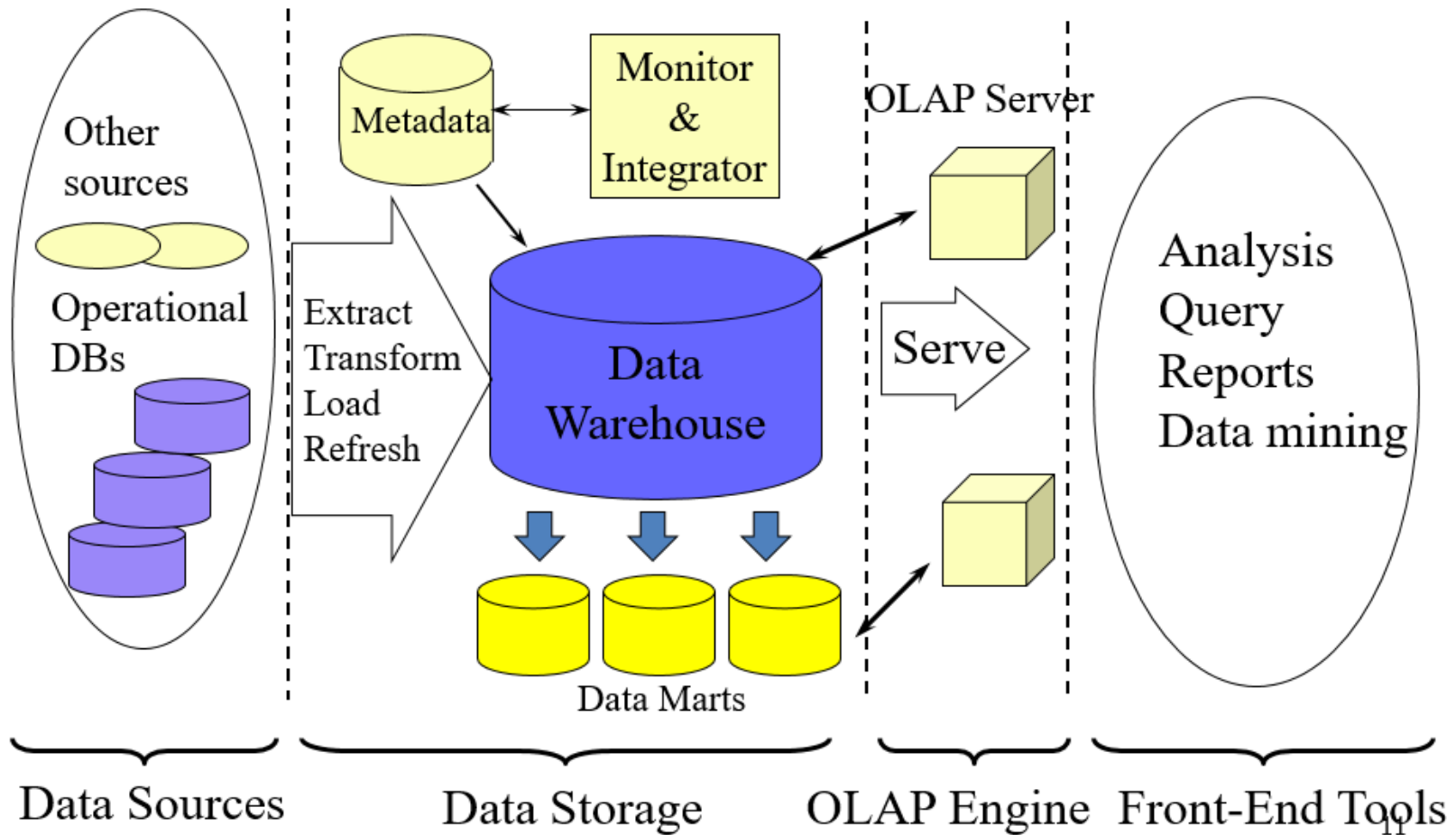
# Why a Separate Data Warehouse?

- 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

# Why a Separate Data Warehouse?

- Different functions and different data:
  - missing data: Decision support requires historical data which operational DBs do not typically maintain
  - data consolidation: DS requires consolidation (aggregation, summarization) of data from heterogeneous sources
  - data quality: different sources typically use inconsistent data representations, codes and formats which have to be reconciled
- Note: There are more and more systems which perform OLAP analysis directly on relational databases

# Data Warehouse: A 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

# Extraction, Transformation, and Loading (ETL)

- **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

# Metadata Repository

- **Meta data** is the data defining warehouse objects. It stores:
  - Description of the **structure** of the data warehouse
    - schema, view, dimensions, hierarchies, derived data defn, 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)

# Metadata Repository...

- The **algorithms** used for summarization
- The **mapping** from operational environment to the data warehouse
- Data related to **system performance**
  - warehouse schema, view and derived data definitions
- **Business data**
  - business terms and definitions, ownership of data, charging policies

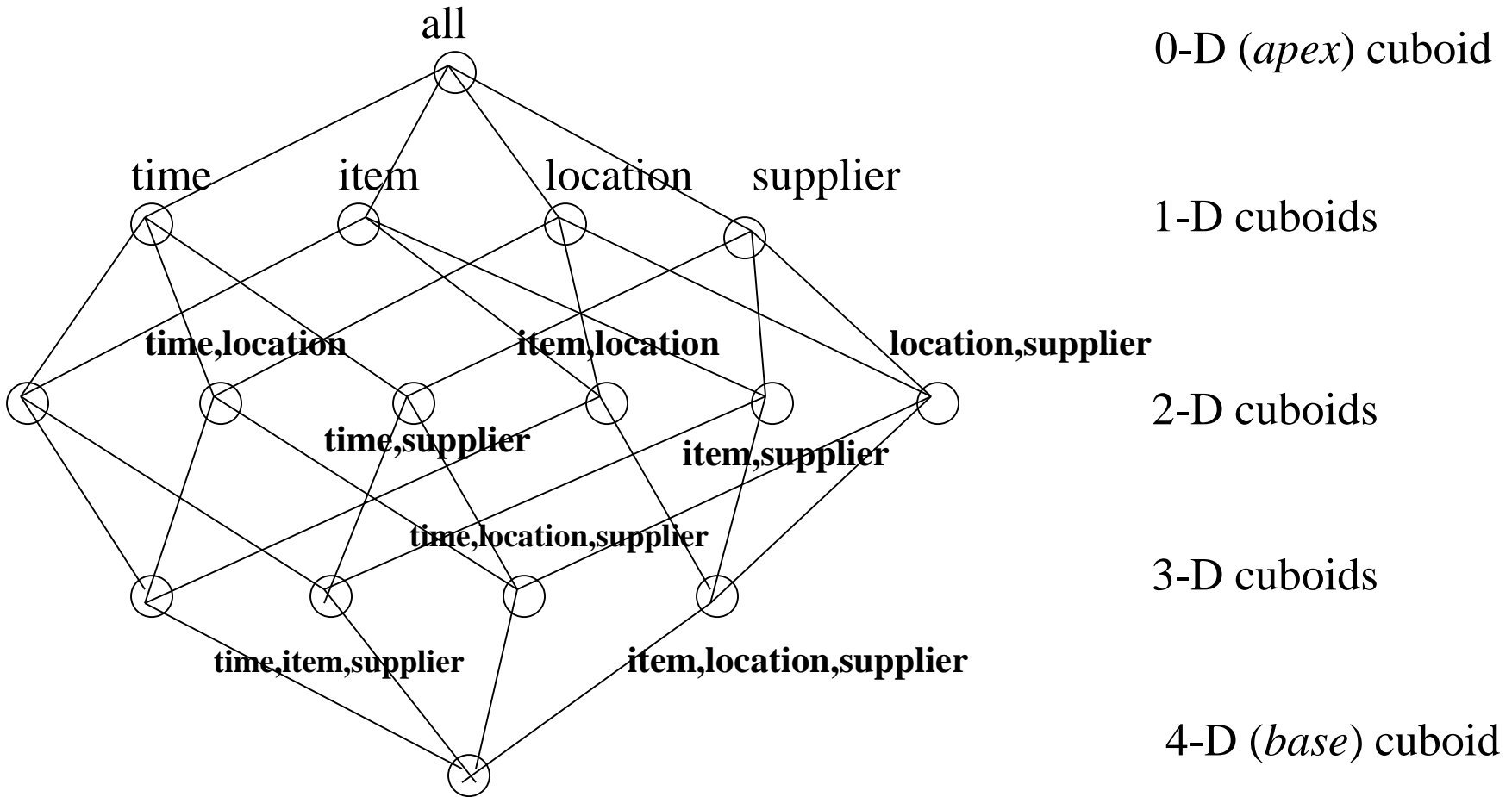
# From Tables and Spreadsheets to Data Cubes

- A **data warehouse** is based on a **multidimensional data model** which views data in the form of a data cube
- A data cube, such as **sales**, allows data to be modeled and viewed in multiple dimensions
  - **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

# From Tables and Spreadsheets to Data Cubes

- 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**.

# Cube: A Lattice of Cuboids



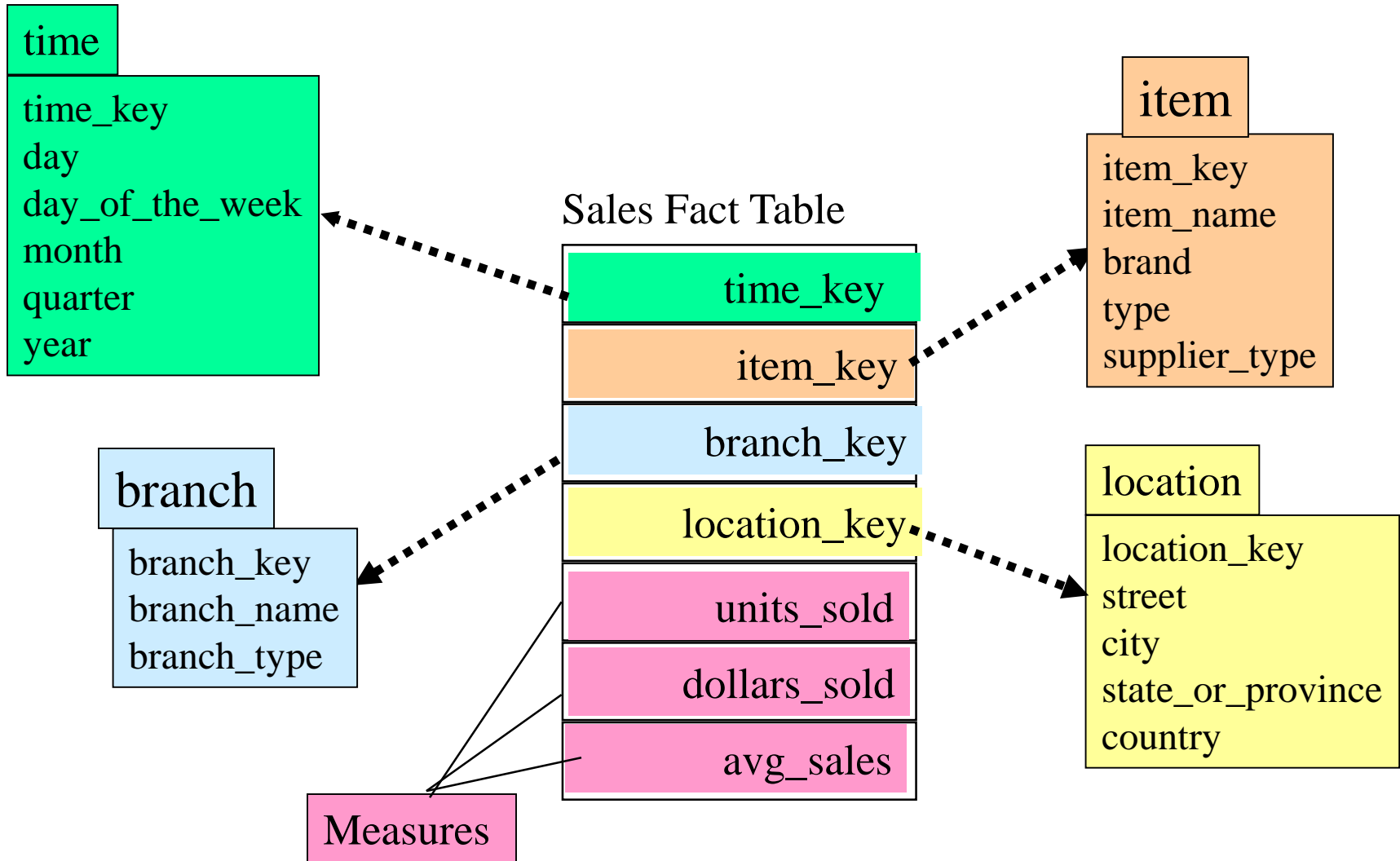
# Conceptual Modeling of Data Warehouses

- Modeling data warehouses: dimensions & measures
  - Star schema: A fact table in the middle connected to a set of dimension tables

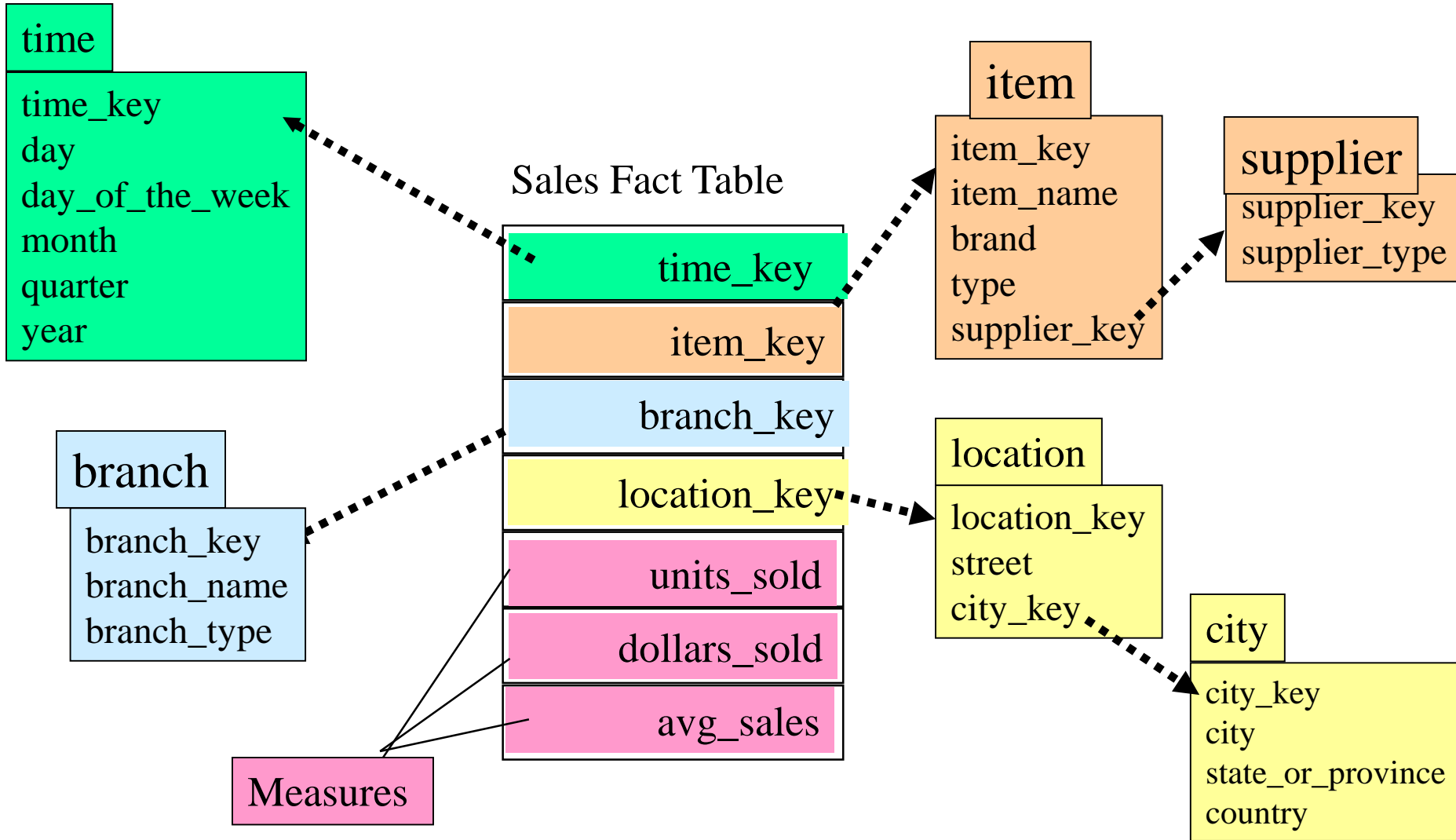
# Conceptual Modeling of Data Warehouses

- Modeling data warehouses: dimensions & measures
  - 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

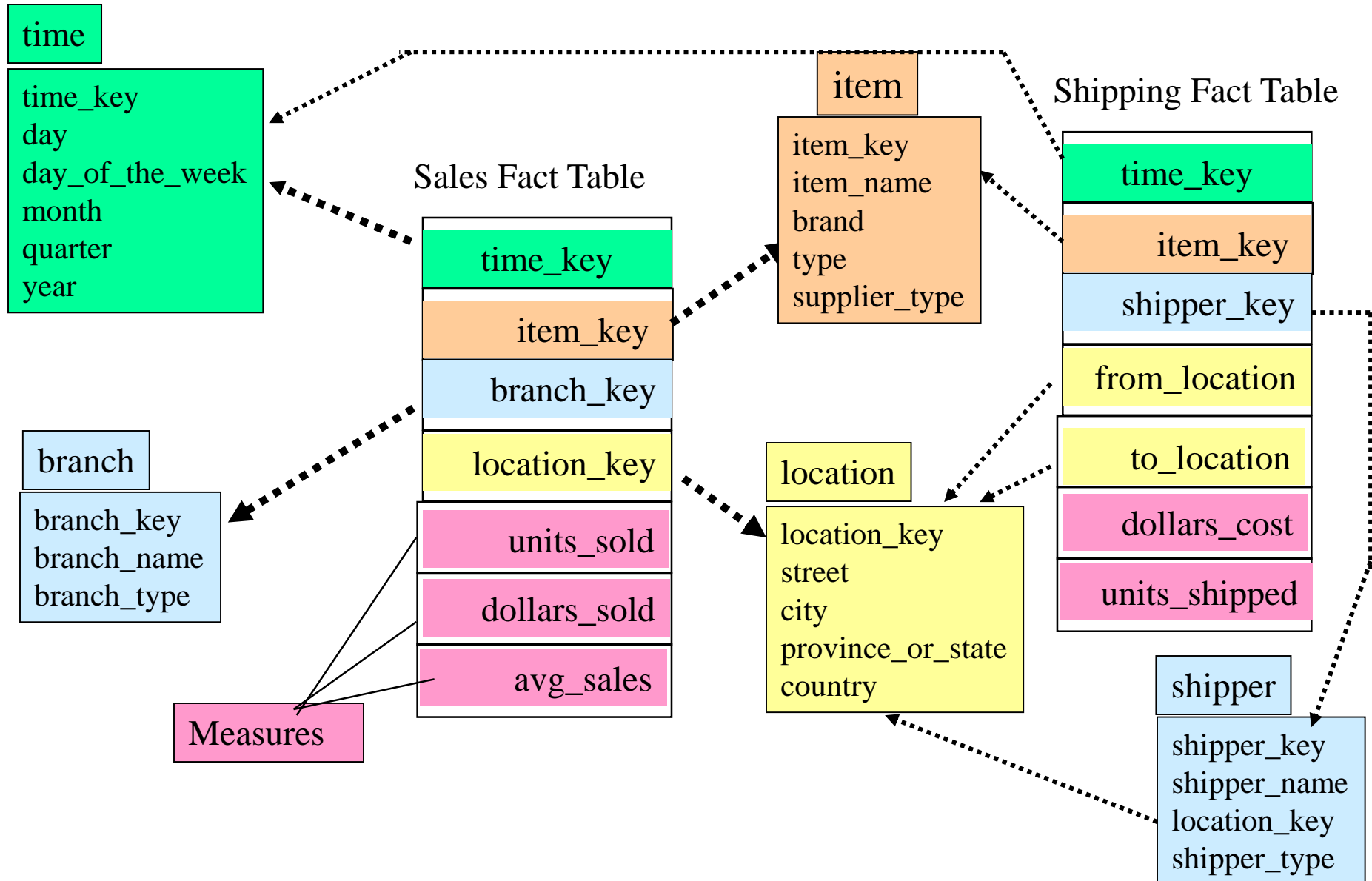
# Example of Star Schema



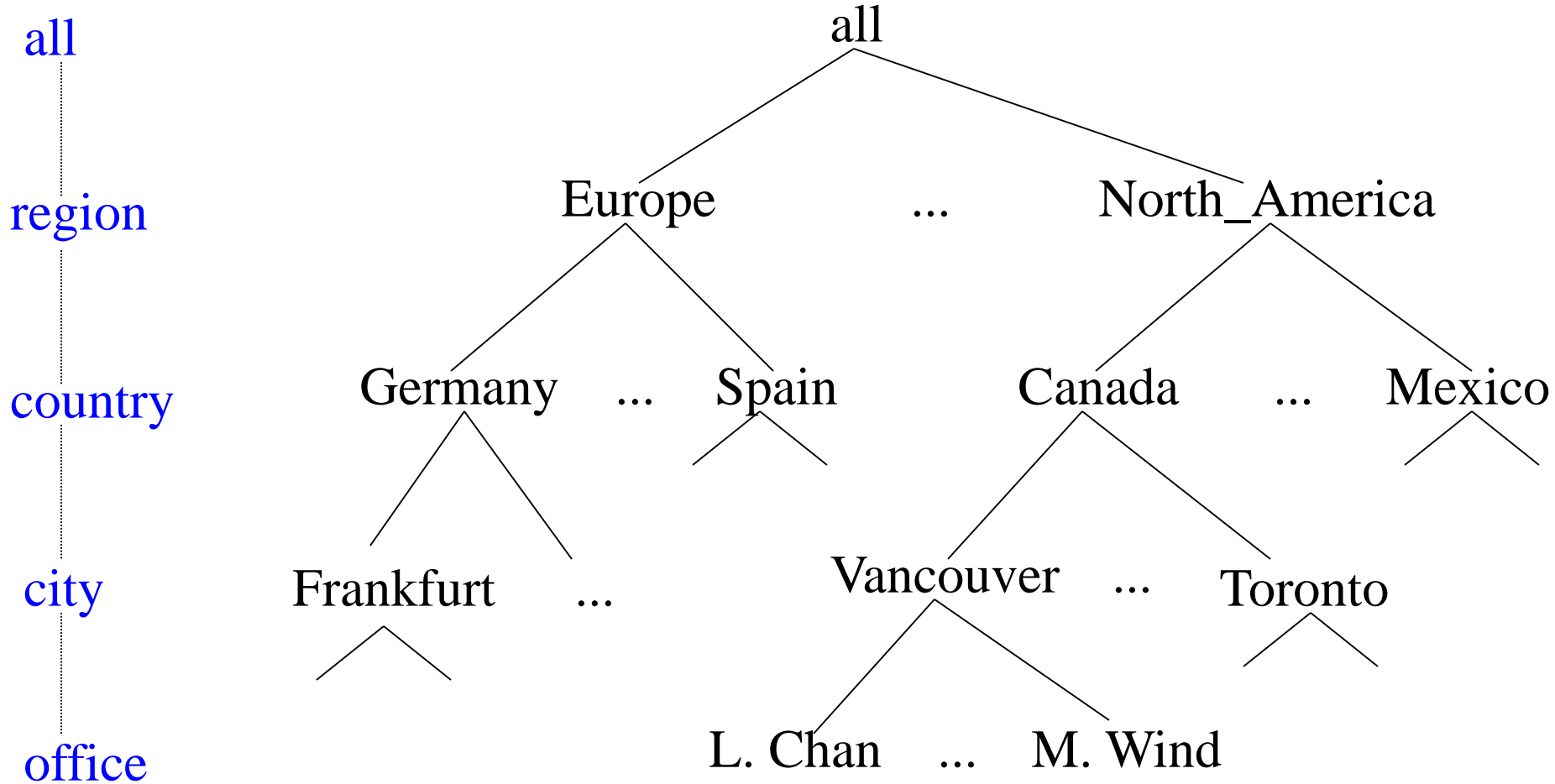
# Example of Snowflake Schema



# Example of Fact Constellation



# A Concept Hierarchy: Dimension (location)



# Data Cube Measures: Three Categories

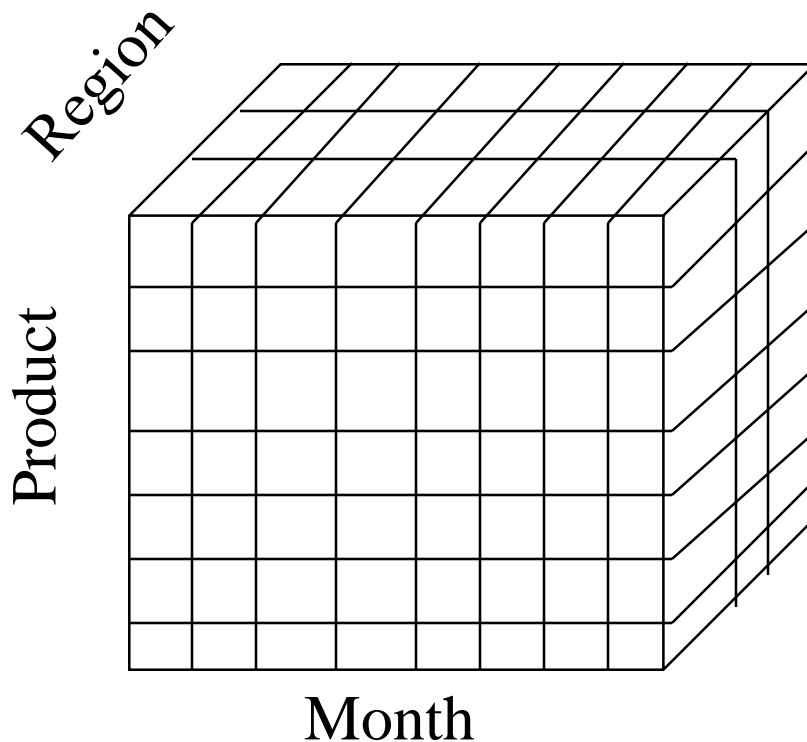
- **Distributive**: if the result derived by applying the function to  $n$  aggregate values is the same as that derived by applying the function on all the data without partitioning
  - E.g., `count()`, `sum()`, `min()`, `max()`

# Data Cube Measures: Three Categories

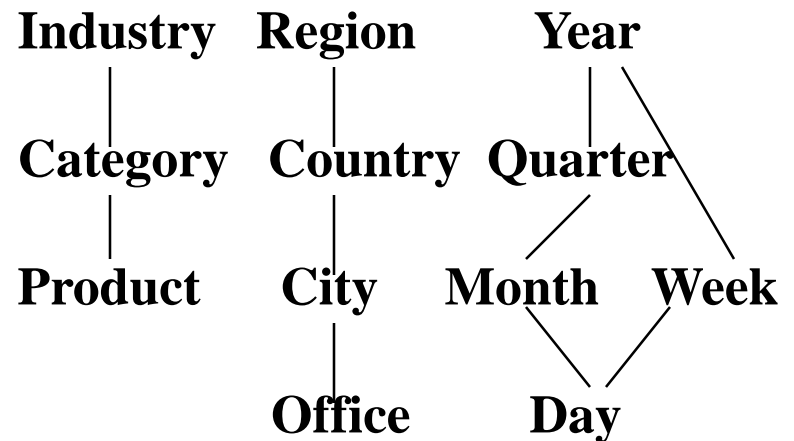
- **Algebraic**: if it can be computed by an algebraic function with  $M$  arguments (where  $M$  is a bounded integer), each of which is obtained by applying a distributive aggregate function
  - E.g., `avg()`, `min_N()`, `standard_deviation()`
- **Holistic**: if there is no constant bound on the storage size needed to describe a subaggregate.
  - E.g., `median()`, `mode()`, `rank()`

# Multidimensional Data

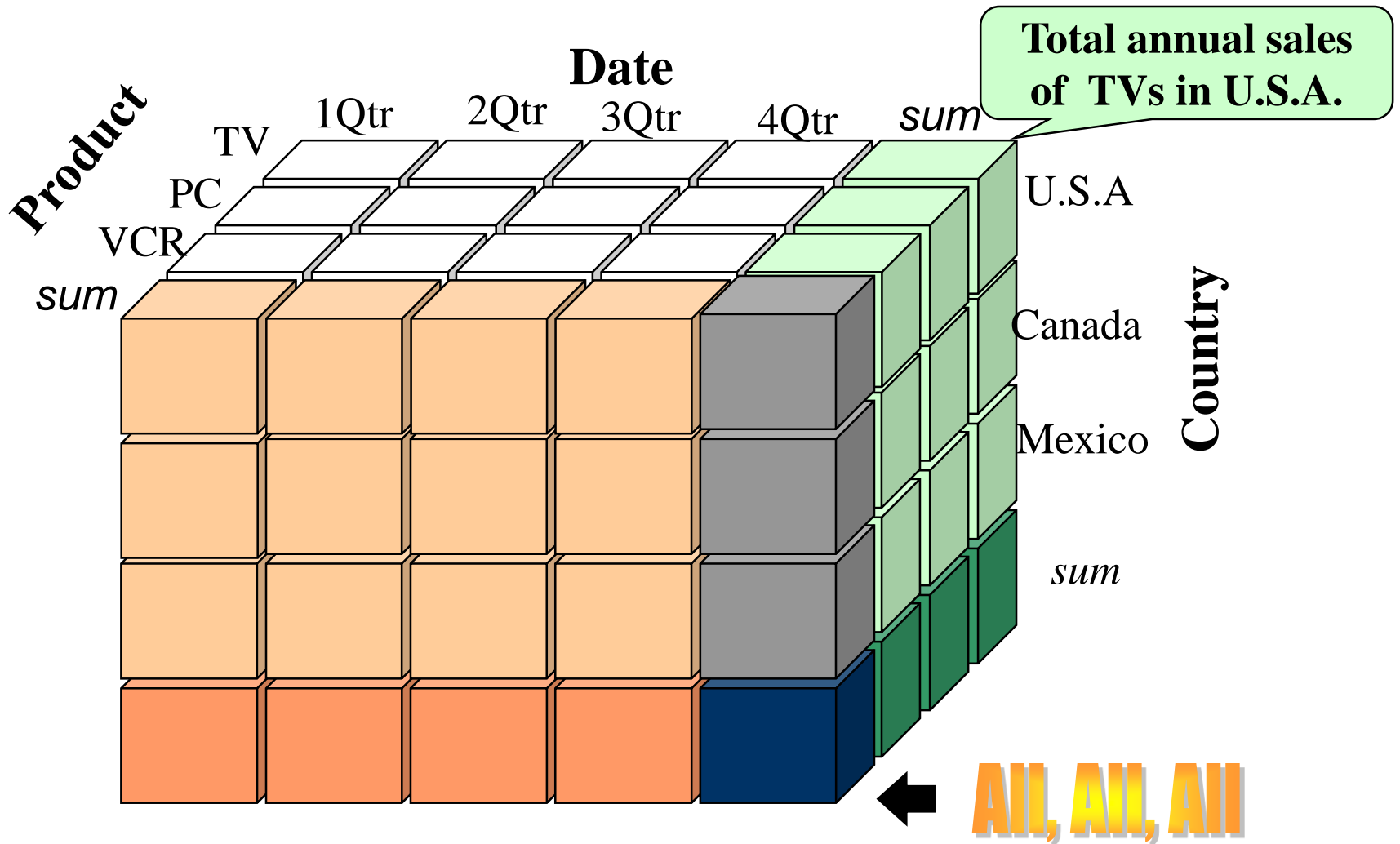
- Sales volume as a function of product, month, and region



**Dimensions:** *Product, Location, Time*  
**Hierarchical summarization paths**



# A Sample Data Cube



# 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, short turn around time, quick turn around

# Data Warehouse Design Process

- **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

# Summary

- In today's lecture we have discussed about;
  - Key concepts of data warehousing
  - Including understanding its fundamentals
  - Recognizing its architecture, design, and models
  - Distinguishing between OLTP and OLAP systems, and
  - Explaining the ELT (Extract, Load, Transform) process

# References

- Inmon, W. H. (2005). *Building the Data Warehouse* (4th ed.). Wiley.
- Kimball, R., & Ross, M. (2013). *The Data Warehouse Toolkit* (3rd ed.). John Wiley & Sons.
- Jiawei Han, Micheline Kamber, & Jian Pei. (2012). *Data mining: Concepts and techniques* (3rd ed.). Morgan Kaufmann.
- Kimball, R., & Ross, M. (2013). *The data warehouse toolkit: The definitive guide to dimensional modeling* (3rd ed.). Wiley.