Module-1

DATA WAREHOUSING & MODELLING

Data warehouses generalize and consolidate data in multidimensional space. The construction of data warehouses involves data cleaning, data integration, and data transformation, and can be viewed as an important preprocessing step for data mining. Moreover, data warehouses provide online analytical processing (OLAP) tools for the interactive analysis of multidimensional data of varied granularities, which facilitates effective data generalization and data mining.

Many other data mining functions, such as association, classification, prediction, and clustering, can be integrated with OLAP operations to enhance interactive mining of knowledge at multiple levels of abstraction. Hence, the data warehouse has become an increasingly important platform For data analysis and OLAP and will provide an effective platform for datamining. Therefore ,data warehousing and OLAP form an essential step in the knowledge discovery process.

1.1 Data Warehouse: Basic Concepts

1.1.1 What Is a Data Warehouse?

Data warehousing provides architectures and tools for business executives to systematically organize, understand, and use their data to make strategic decisions. Data warehouse systems are valuable tools in today's competitive, fast-evolving world. In the last several years, many firms have spent millions of dollars in building enterprise-wide data warehouses. Many people feel that with competition mounting in every industry, data warehousing is the latest must-have marketing weapon—a way to retain customers by learning more about their needs.

A data warehouse refers to a data repository that is maintained separately from an organization's operational databases. Data warehouse systems allow for integration of a variety of application systems. They support information processing by providing a solid platform of consolidated historic data for analysis.

According to William H. Inmon, a leading architect in the construction of data warehouse systems, "A data warehouse is a subject-oriented, integrated, time-variant, and nonvolatile collection of data in support of management's decision making process".

Key features:

Subject-Oriented: A data warehouse can be used to analyze a particular subject area. For example, "sales" can be a particular subject.

Integrated: A data warehouse integrates data from multiple data sources. For example, source A and source B may have different ways of identifying a product, but in a data warehouse, there will be only a single way of identifying a product.

Time-Variant: Historical data is kept in a data warehouse. For example, one can retrieve data from 3 months, 6 months, 12 months, or even older data from a data warehouse. This contrasts with a transactions system, where often only the most recent data is kept. For example, a transaction system may hold the most recent address of a customer, where a data warehouse can hold all addresses associated with a customer.

Non-volatile: Once data is in the data warehouse, it will not change. So, historical data in a data warehouse should never be altered.

1.1.2 Differences between Operational Database Systems and Data Warehouses

The major task of online operational database systems is to perform online transaction and query processing. These systems are called **online transaction processing (OLTP)** systems. They cover most of the day-to-day operations of an organization such as purchasing, inventory, manufacturing, banking, payroll, registration, and accounting.

Data warehouse systems, on the other hand, serve users or knowledge workers in the role of data analysis and decision making. Such systems can organize and present data in various formats in order to accommodate the diverse needs of different users. These systems are known as **online analytical processing(OLAP**) systems. The major distinguishing features of OLTP and OLAP are summarized as follows:

Table 4.1 Comparison of OLTP and OLAP Systems

Feature	*OLTP*	*OLAP*
Characteristic	operational processing	informational processing
Orientation	transaction	analysis
User	clerk, DBA, database professional	knowledge worker (e.g., manager, executive, analyst)
Function	day-to-day operations	long-term informational requirements decision support
DB design	ER-based, application-oriented	star/snowflake, subject-oriented
Data	current, guaranteed up-to-date	historic, accuracy maintained over time
Summarization	primitive, highly detailed	summarized, consolidated
View	detailed, flat relational	summarized, multidimensional
Unit of work	short, simple transaction	complex query
Access	read/write	mostly read
Focus	data in	information out
Operations	index/hash on primary key	lots of scans
Number of records accessed	tens	millions
Number of users	thousands	hundreds
DB size	GB to high-order GB	≥ TB
Priority	high performance, high availability	high flexibility, end-user autonomy
Metric	transaction throughput	query throughput, response time

1.1.3 Data Warehousing: A Multitiered Architecture

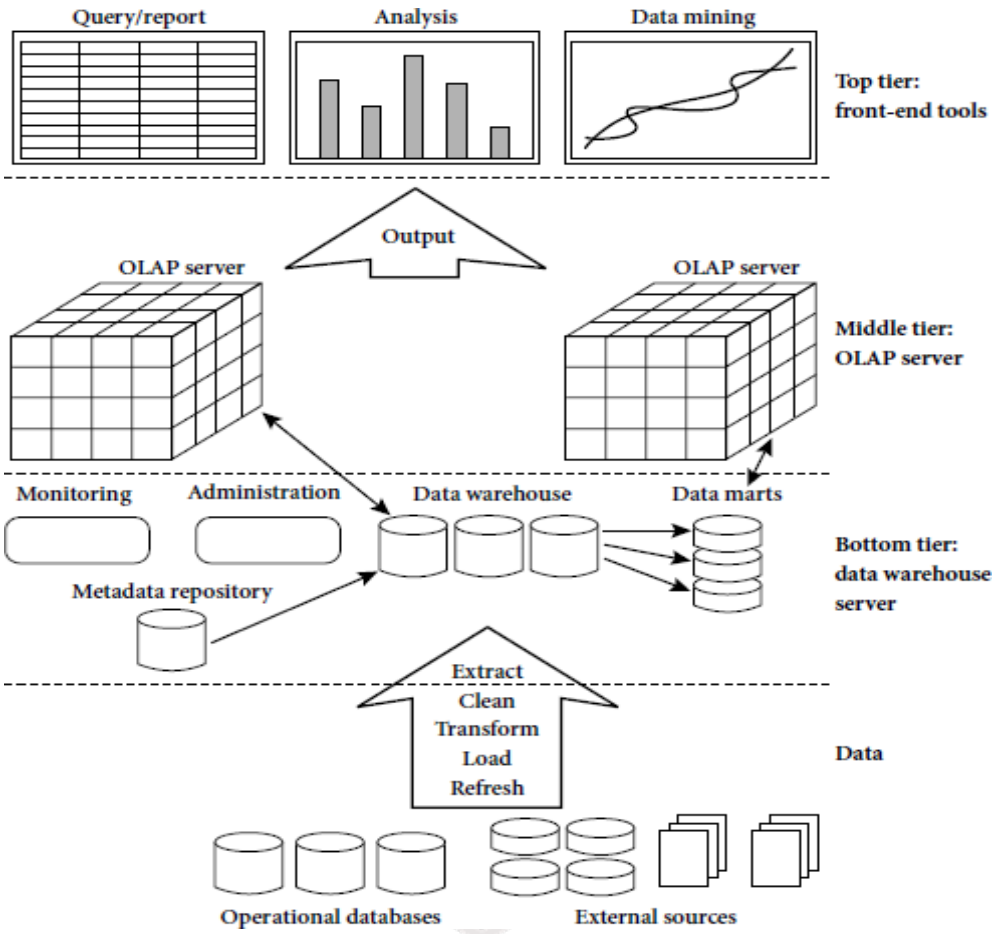
Tier-1:

The bottom tier is a warehouse database server that is almost always a relational database system. Back-end tools and utilities are used to feed data into the bottom tier from operational databases or other external sources (such as customer profile information provided by external consultants). These tools and utilities perform data extraction, cleaning, and transformation (e.g., to merge similar data from different sources into a unified format), as well as load and refresh functions to update the data warehouse . The data are extracted using application program interfaces known as gateways. A gateway is supported by the underlying DBMS and allows client programs to generate SQL code to be executed at a server.

Examples of gateways include ODBC (Open Database Connection) and

OLEDB (Open Linkingand Embedding for Databases) by Microsoft and JDBC (Java Database Connection).

This tier also contains a metadata repository, which stores information aboutthe data warehouse and its contents.



A Three Tier Data Warehouse Architecture:

Tier-2:

The middle tier is an OLAP server that is typically implemented using either a relational OLAP (ROLAP) model or a multidimensional OLAP.

- OLAP model is an extended relational DBMS thatmaps operations on multidimensional data to standard relational operations.
- A multidimensional OLAP (MOLAP) model, that is, a special-purpose server that directly implements multidimensional data and operations.

Tier-3:

The top tier is a front-end client layer, which contains query

and reporting tools, analysis tools, and/or data mining tools (e.g., trend analysis, prediction, and so on).

1.1.4 Data Warehouse Models:

There are three data warehouse models.

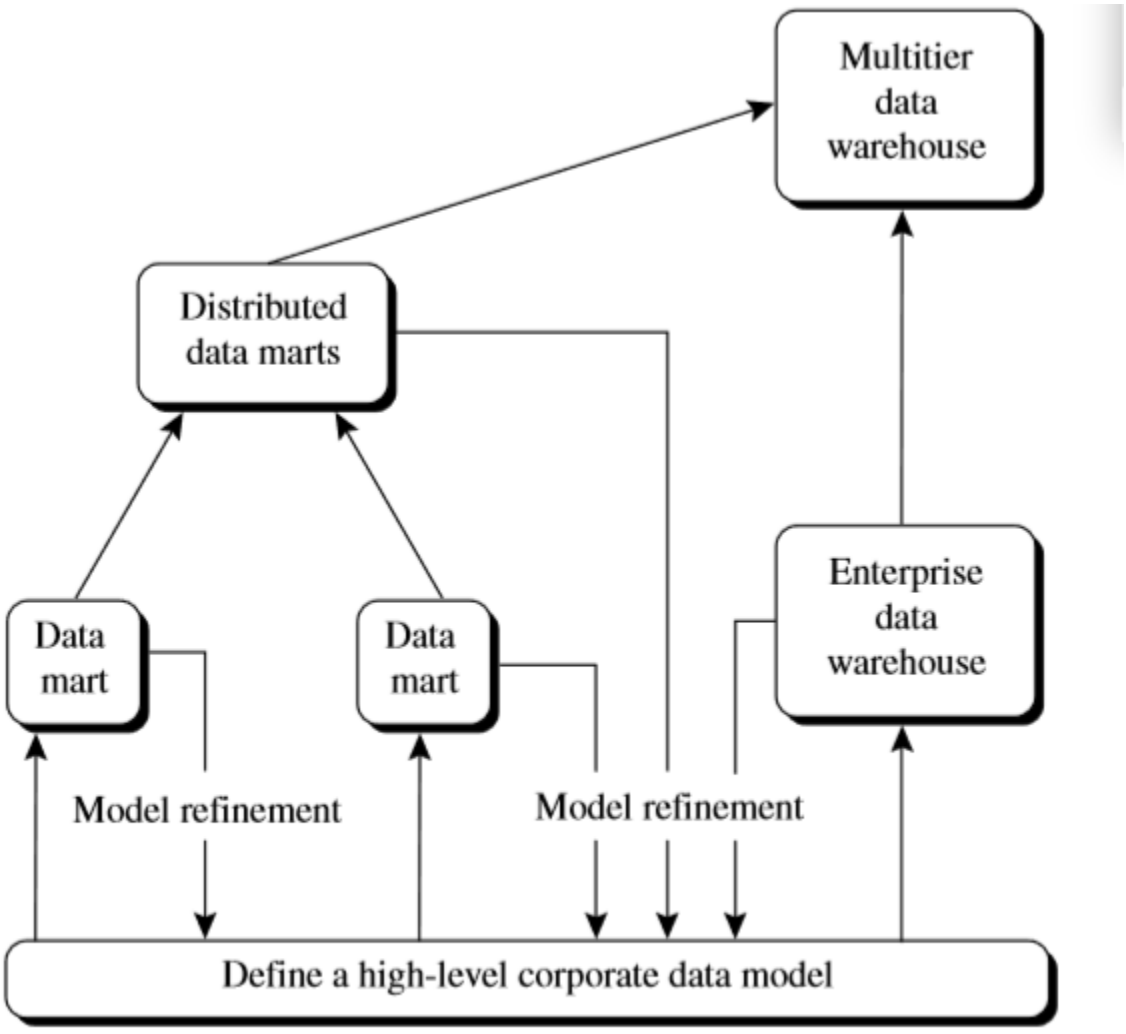
1. *Enterprise warehouse:*
   - An enterprise warehouse collects all of the information about subjects spanning the entire organization.
   - It provides corporate-wide data integration, usually from one or more operational systems or external information providers, and is cross-functional in scope.
   - It typically contains detailed data aswell as summarized data, and can range in size from a few gigabytes to hundreds of gigabytes, terabytes, or beyond.
   - An enterprise data warehouse may be implemented on traditional mainframes, computer super servers, or parallel architecture platforms. It requires extensive business modeling and may take years to design and build.
2. *Data mart:*
   - A data mart contains a subset of corporate-wide data that is of value to a specific group of users. The scope is confined to specific selected subjects. For example, a marketing data mart may confine its subjects to customer, item, and sales. The data contained in data marts tend to be summarized.
   - Data marts are usually implemented on low-cost departmental servers that are UNIX/LINUX- or Windows-based. The implementation cycle of a data mart is more likely to be measured in weeks rather than months or years. However, it may involve complex integration in the long run if its design and planning were not enterprise-wide.

- Depending on the source of data, data marts can be categorized as independent or dependent. Independent data marts are sourced from data captured from one or more operational systems or external information providers, or from data generated locally within a particular department or geographic area. Dependent data marts are sourced directly from

enterprise data warehouses.

3. *Virtual warehouse:*

- A virtual warehouse is a set of views over operational databases. For efficient query processing, only some of the possible summary views may be materialized.
- A virtual warehouse is easy to build but requires excess capacity on operational database servers.



4.2 A recommended approach for data warehouse development.

1.1.5 Extraction, Transformation, and Loading

Data warehouse systems use back-end tools and utilities to populate and refresh their data .These tools and utilities include the following functions:

Data extraction, which typically gathers data from multiple, heterogeneous, and external sources.

Data cleaning, which detects errors in the data and rectifies them when possible.

Data transformation, which converts data from legacy or host format to warehouse format.

Load, which sorts, summarizes, consolidates, computes views, checks integrity, and builds indices and partitions.

Refresh, which propagates the updates from the data sources to the warehouse.

1.1.6 Meta Data Repository:

Metadata are data about data. When used in a data warehouse, metadata are the data that define warehouse objects. Metadata are created for the data names and definitions of the given warehouse. Additional metadata are created and captured for timestamping any extracted data, the source of the extracted data, and missing fields that have been added by data cleaning or integration processes.

A metadata repository should contain the following:

- A description of the *structure of the data warehouse*, which includes the warehouse schema, view, dimensions, hierarchies, and derived data definitions, as well as data mart locations and contents.
- *Operational metadata*, which include data lineage (history of migrated data and the sequence of transformations applied to it), currency of data (active, archived, or purged), and monitoring information (warehouse usage statistics, error reports, and audit trails).
- The algorithms used for summarization, which include measure and dimension definition algorithms, data on granularity, partitions, subject areas, aggregation, summarization, and predefined queries and reports.
- *The mapping from the operational environment to the data warehouse*, which includes source databases and their contents, gateway descriptions, data partitions, data extraction, cleaning, transformation rules and defaults, data refresh and purging rules, and security (user authorization and access control).
- *Data related to system performance*, which include indices and profiles that improve data access and retrieval performance, in addition to rules for the timing and scheduling of refresh, update, and replication cycles.

- *Business metadata*, which include business terms and definitions, data ownership information, and charging policies.

1.2 Data Warehouse Modeling: Data Cube and OLAP

Data warehouses and OLAP tools are based on a **multidimensional data** model. This model views data in the form of a data cube.

1.2.1 Data Cube : A multidimensional Data model

- 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
- 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.

Table 4.2 2-D View of Sales Data for *AllElectronics* According to *time* and *item*

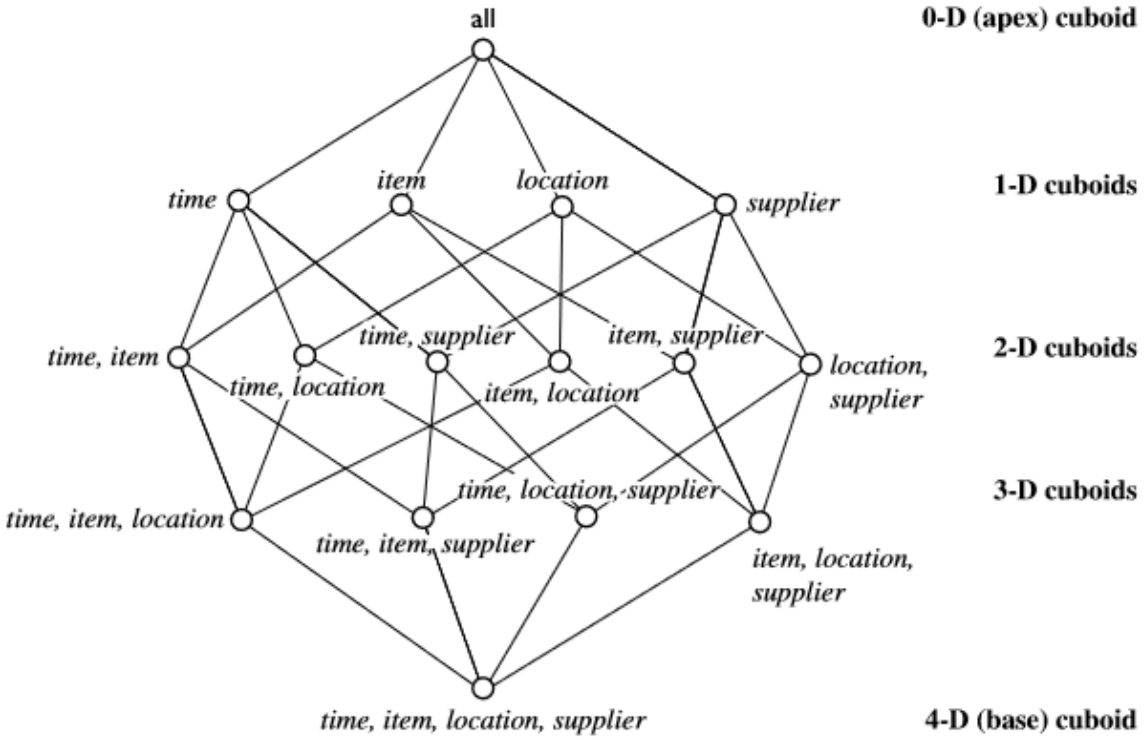
location = "Vancouver"				
	item (type)			
time (quarter)	home entertainment	computer	phone	security
Q1	605	825	14	400
Q2	680	952	31	512
Q3	812	1023	30	501
Q4	927	1038	38	580

Note: The sales are from branches located in the city of Vancouver. The measure displayed is *dollars_sold* (in thousands).

Table 4.3 3-D View of Sales Data for *AllElectronics* According to *time*, *item*, and *location*

	location = "Chicago"				location = "New York"				location = "Toronto"				location = "Vancouver"			
	item				item				item				item			
	home				home				home				home			
time	ent.	comp.	phone	sec.	ent.	comp.	phone	sec.	ent.	comp.	phone	sec.	ent.	comp.	phone	sec.
Q1	854	882	89	623	1087	968	38	872	818	746	43	591	605	825	14	400
Q2	943	890	64	698	1130	1024	41	925	894	769	52	682	680	952	31	512
Q3	1032	924	59	789	1034	1048	45	1002	940	795	58	728	812	1023	30	501
Q4	1129	992	63	870	1142	1091	54	984	978	864	59	784	927	1038	38	580

Given a set of dimensions, we can generate a cuboid for each of the possible subsets of the given dimensions. The result would form a lattice of cuboids, each showing the data at a different level of summarization, or group-by. The lattice of cuboids is then referred to as a data cube. Figure shows a lattice of cuboids forming a data cube for the dimensions time, item, location, and supplier. The cuboid that holds the lowest level of summarization is called the base cuboid.



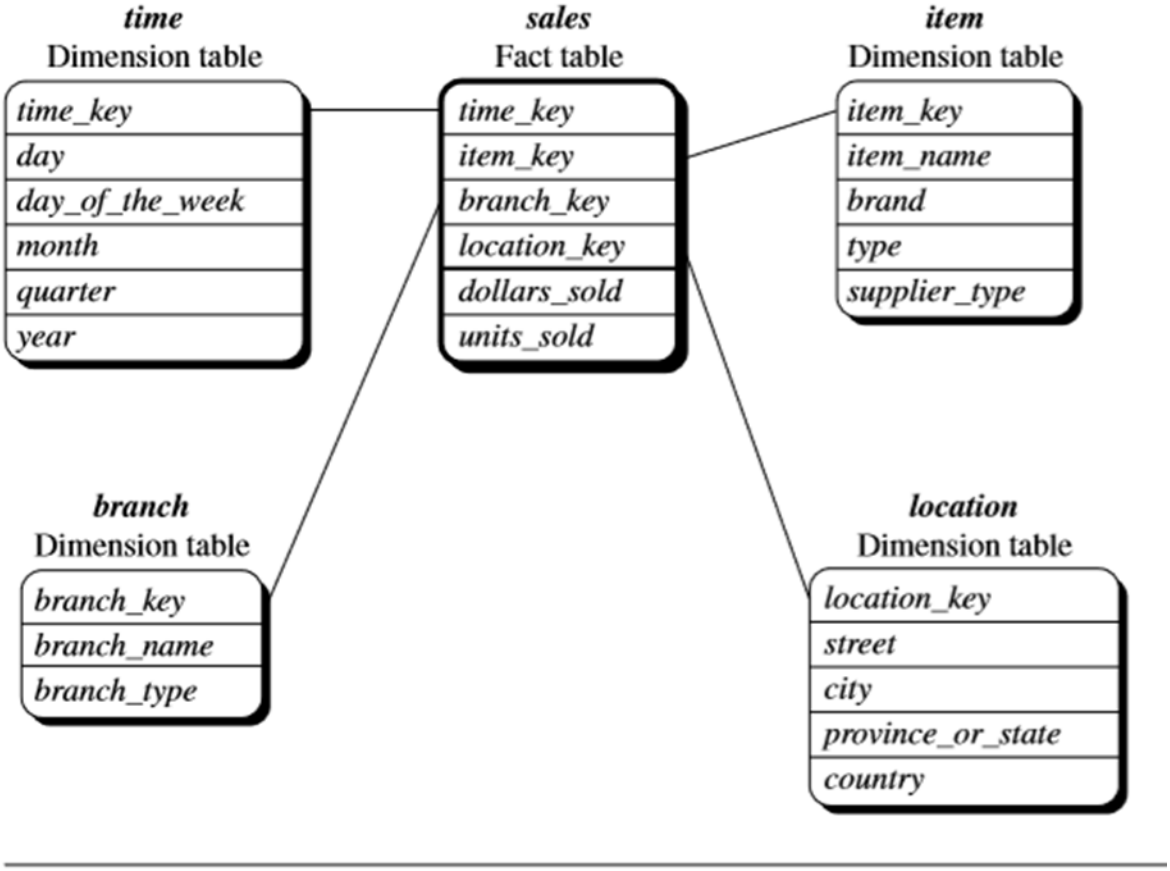
Lattice of cuboids, making up a 4-D data cube for *time*, *item*, *location*, and *supplier*. Each cuboid represents a different degree of summarization.

1.2.2 Stars, Snowflakes, and Fact Constellations: Schemas for Multidimensional Data Models

The most popular data model for a data warehouse is a multidimensional model, which can exist in the form of a star schema , a snow flake schema, or a fact constellation schema**.**

- Schemas for multidimensional data models
  - 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

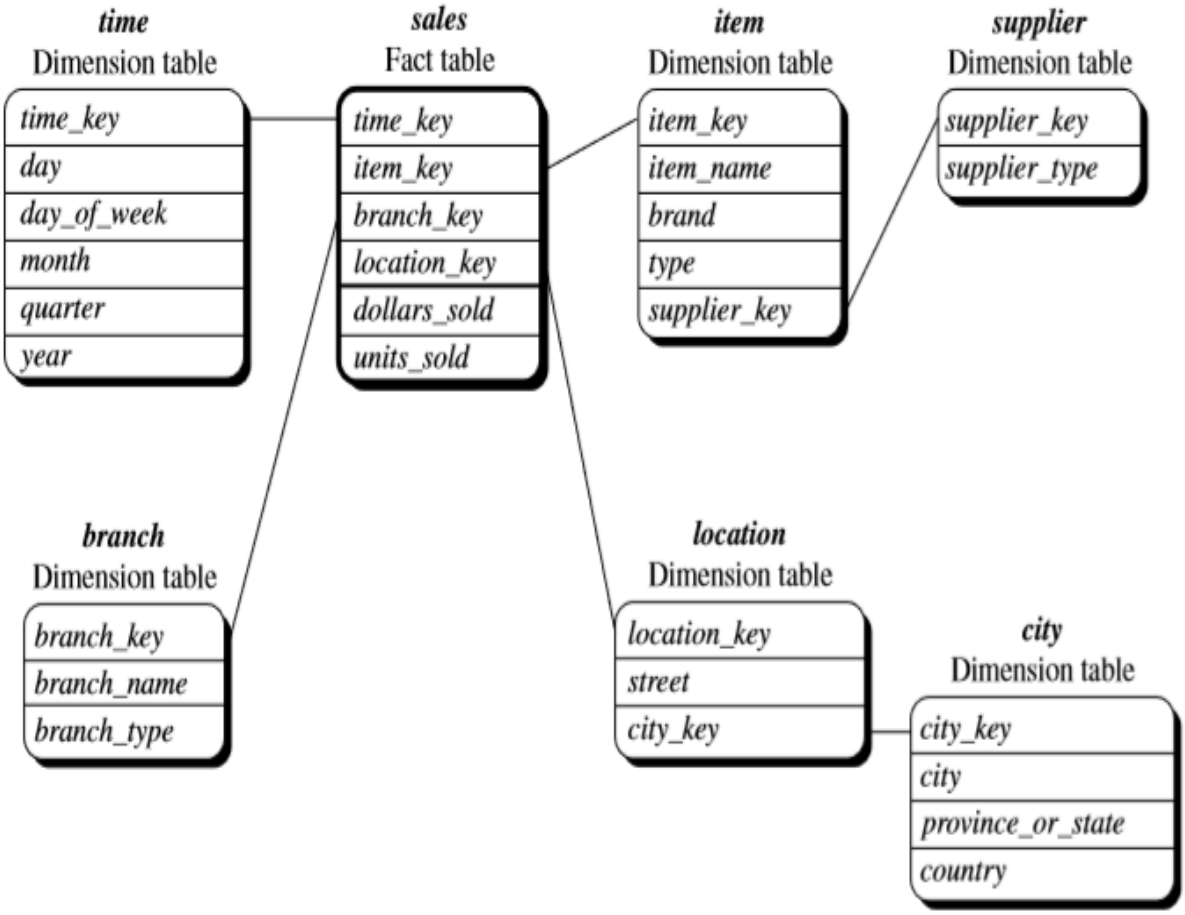
Star schema: The most common modeling paradigm is the star schema, in which the data warehouse contains (1) a large central table (fact table) containing the bulk of the data, with no redundancy, and (2) a set of smaller attendant tables (dimension tables), one for each dimension. The schema graph resembles a starburst, with the dimension tables displayed in a radial pattern around the central fact table.



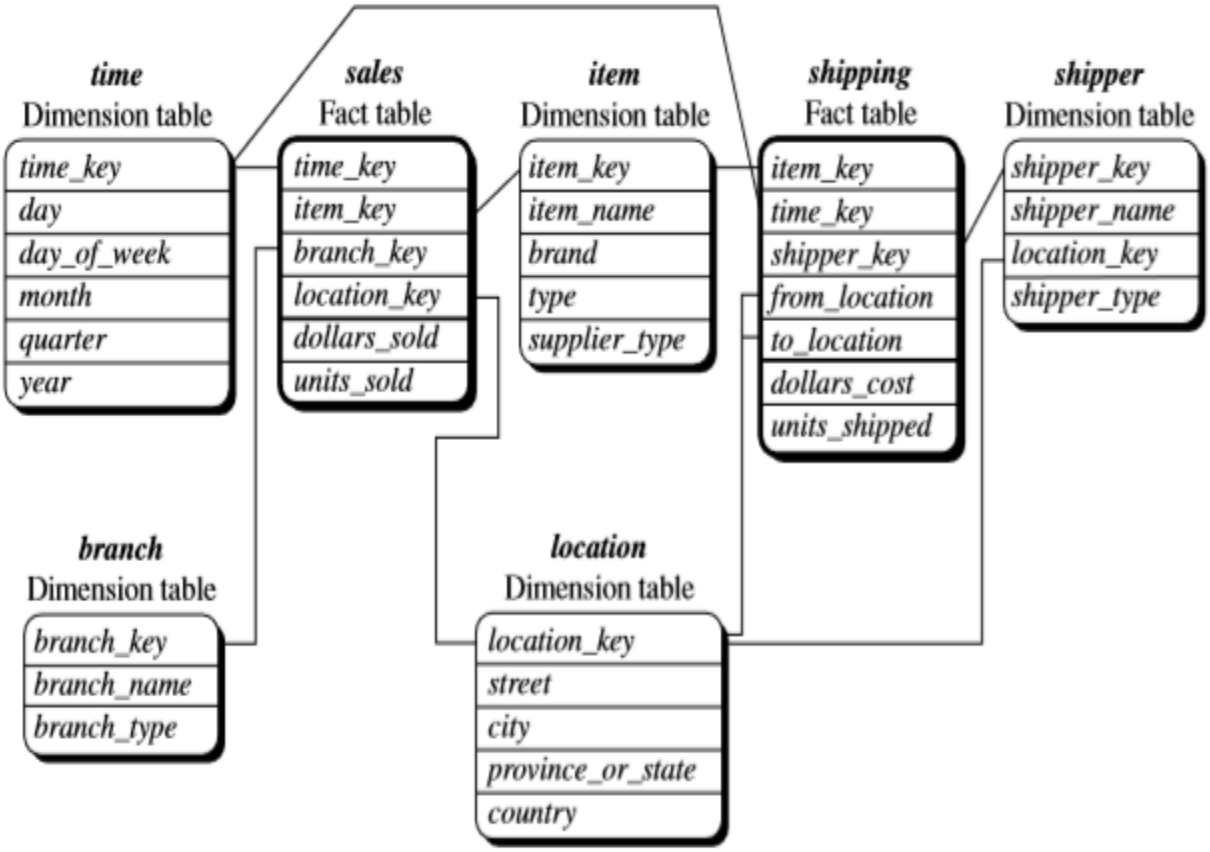
Star schema of *sales* data warehouse.

Snowflake schema: The snowflake schema is a variant of the star schema model, where some dimension tables are normalized, thereby further splitting the data into additional tables. The resulting schema graph forms a shape similar to a snowflake.

Fact constellation: Sophisticated applications may require multiple fact tables to share dimension tables. This kind of schema can be viewed as a collection of stars, and hence is called a galaxy schema or a fact constellation.



Snowflake schema of a *sales* data warehouse.

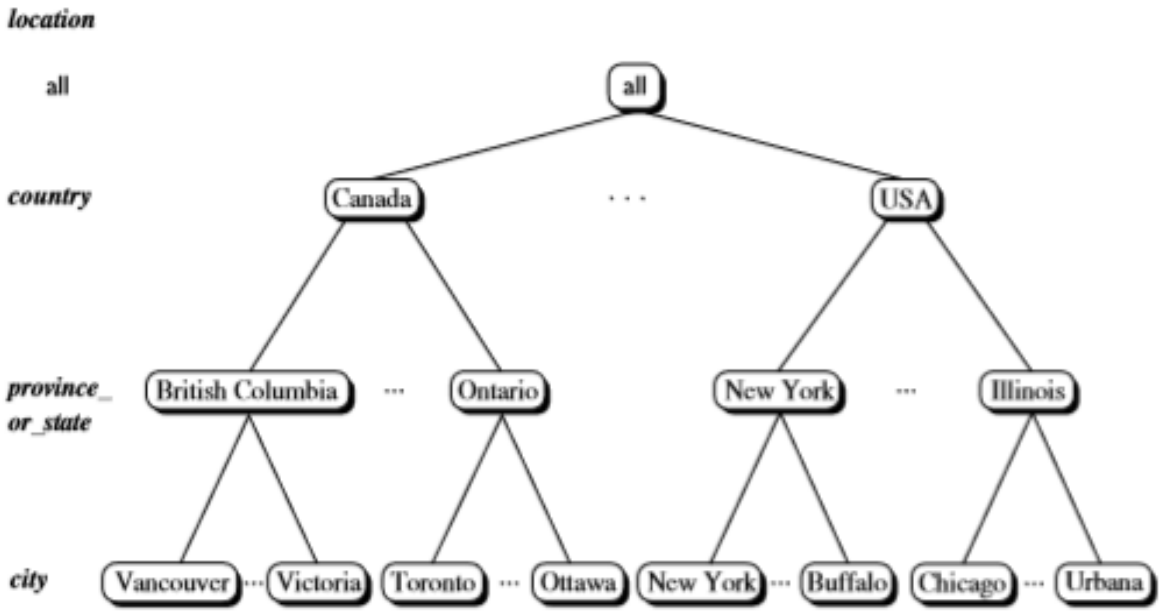


Fact constellation schema of a sales and shipping data warehouse.

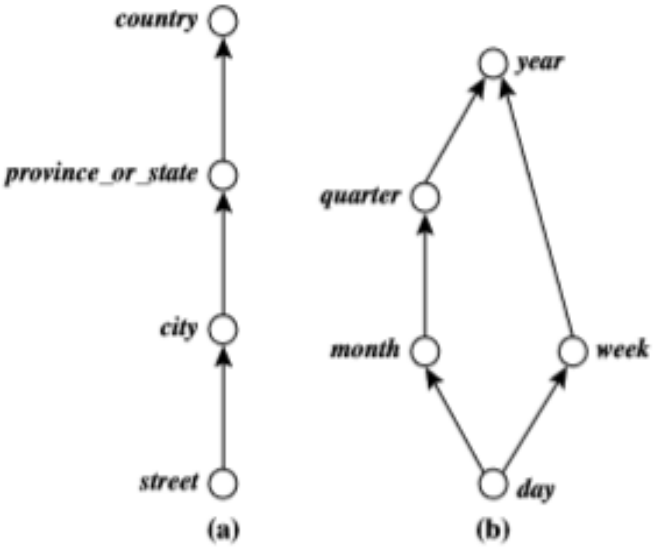
1.2.3 Dimensions: The Role of Concept Hierarchies

A concept hierarchy defines a sequence of mappings from a set of low-level concepts to higher-level, more general concepts. Consider a concept hierarchy for the dimension location. City values for location include Vancouver, Toronto, New York, and Chicago. Each city, however, can be mapped to the province or state to which it belongs

For example, suppose that the dimension location is described by the attributes number, street, city, province or state, zip code, and country. These attributes are related by a total order, forming a concept hierarchy such as "street < city < province or state < country." This hierarchy is shown in Figure



9 A concept hierarchy for *location*. Due to space limitations, not all of the hierarchy nodes are shown, indicated by ellipses between nodes.



1.2.4 Measures: Their Categorization and Computation

- **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()
- **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()

1.2.5 Typical OLAP Operations

- **ROLL-UP**

This is like zooming-out on the data-cube This is required when the user needs further abstraction or less detail. • Initially, the location-hierarchy was "street < city < province < country". • On rolling up, the data is aggregated by ascending the location-hierarchy from the level-of city to level-of-country.

- **DRILL DOWN**

This is like zooming-in on the data. This is the reverse of roll-up. • This is an appropriate operation → when the user needs further details or → when the user wants to partition more finely or → when the user wants to focus on some particular values of certain dimensions. • This adds more details to the data. • Initially, the time-hierarchy was "day < month < quarter < year". • On drill-up, the time dimension is descended from the level-of-quarter to the level-of-month

- **PIVOT (OR ROTATE)**

• This is used when the user wishes to re-orient the view of the data-cube. This may involve → swapping the rows and columns or → moving one of the row-dimensions into the column-dimension.

- **SLICE & DICE**

These are operations for browsing the data in the cube. • These operations allow ability to look at information from different viewpoints. • A slice is a subset of cube corresponding to a single value for 1 or more members of dimensions..

A dice operation is done by performing a selection of 2 or more dimensions.

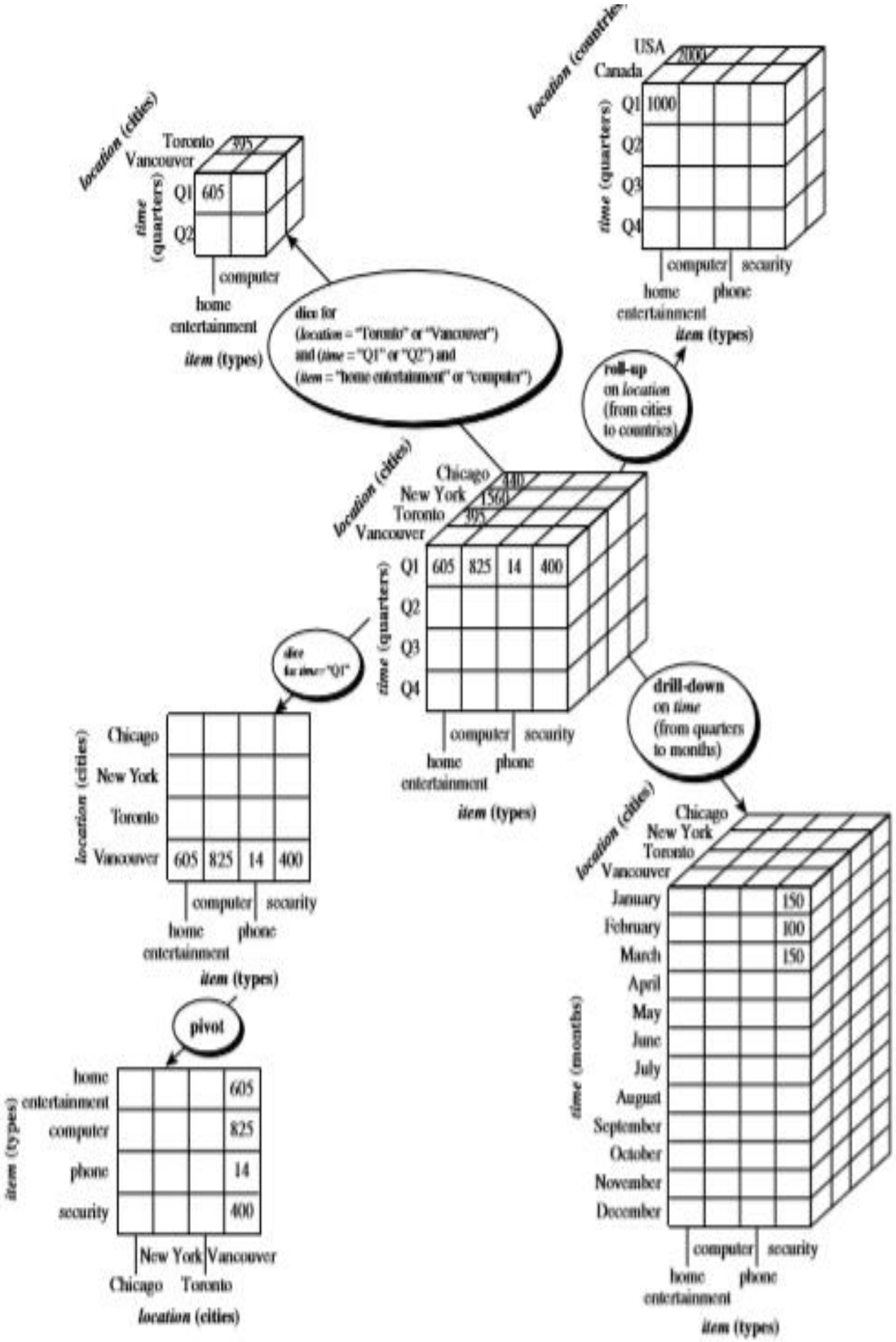


Figure 4.12 Examples of typical OLAP operations on multidimensional data.