Data Warehousing

- Basic Concepts: Why? What?
- DW Modeling: Data Cube and OLAP
- DW Design and Usage
- DW Implementation Issues
- Summary

Basic Concepts

Data, Data everywhere, yet ...

- I can't find the data I need
  - data is scattered over the network
  - many versions, subtle differences
- I can't get the data I need
  - need an expert to get the data
- I can't understand the data I found
  - available data poorly documented
- I can't use the data I found
  - results are unexpected
  - data needs to be transformed from one form to other

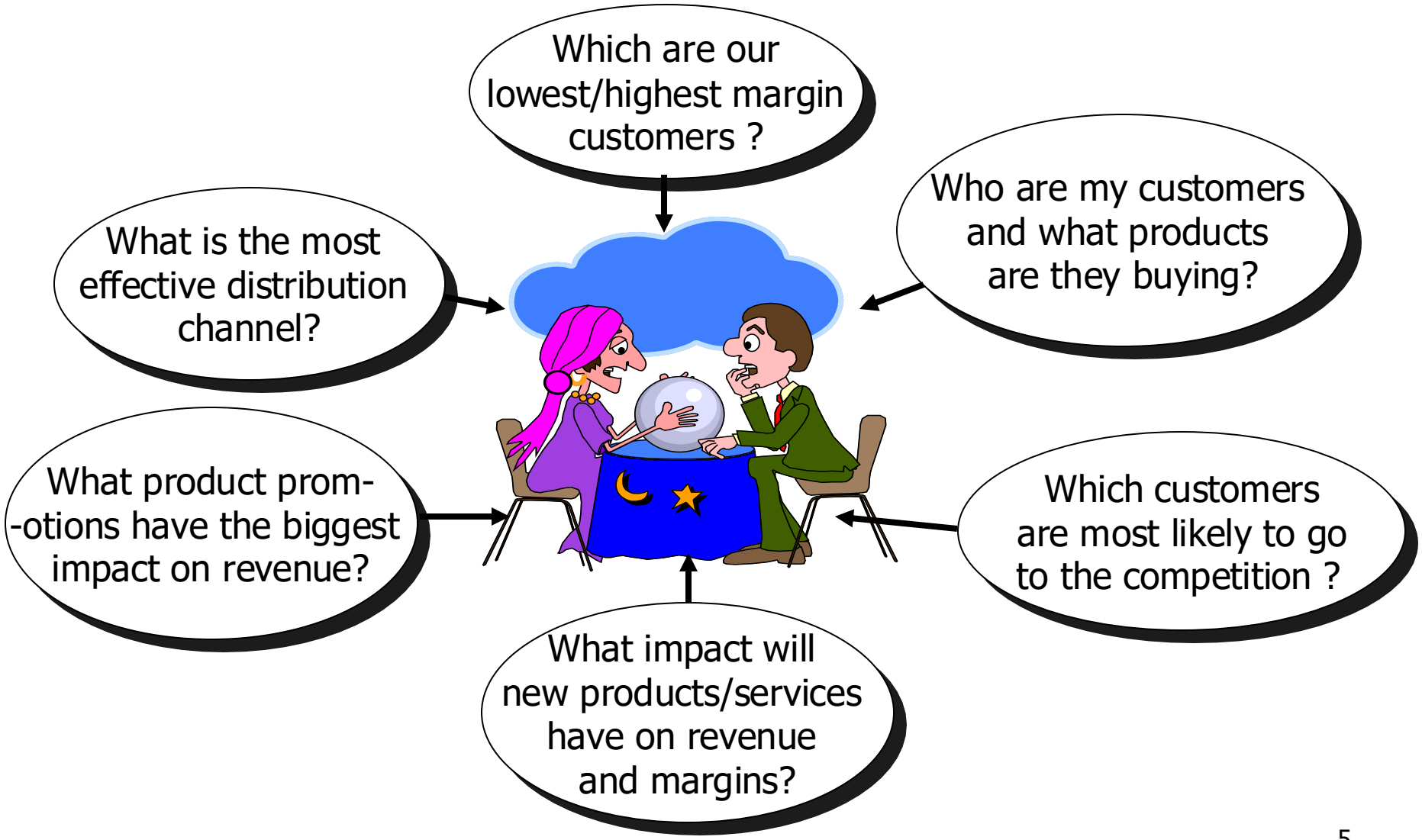
What do decision makers typically look for?

BEST OF BRITISH: UK TOP TRENDING DISHES 2022

Here's a list of the UK's top ten trending dishes on Deliveroo for 2022 - for a list of the UK's top 30, see editor's notes:

- Burrito from Chipotle, London
- Pad Thai from Ting Thai Caravan, Edinburgh
- Cheeseburger from Five Guys, London
- Build Your Own Salad Bowl from atis, London
- Regular Fried Chicken Strips from Clucking Oinks, York
- The Spicy Chicken Sandwich from Popeyes, London
- Perfectly Ripe Avocados from Waitrose, London
- House Black Daal from Dishoom, London
- Original Frozen Yogurt from Snog, London
- Jerk Chicken and Chips from White Men Can't Jerk, London

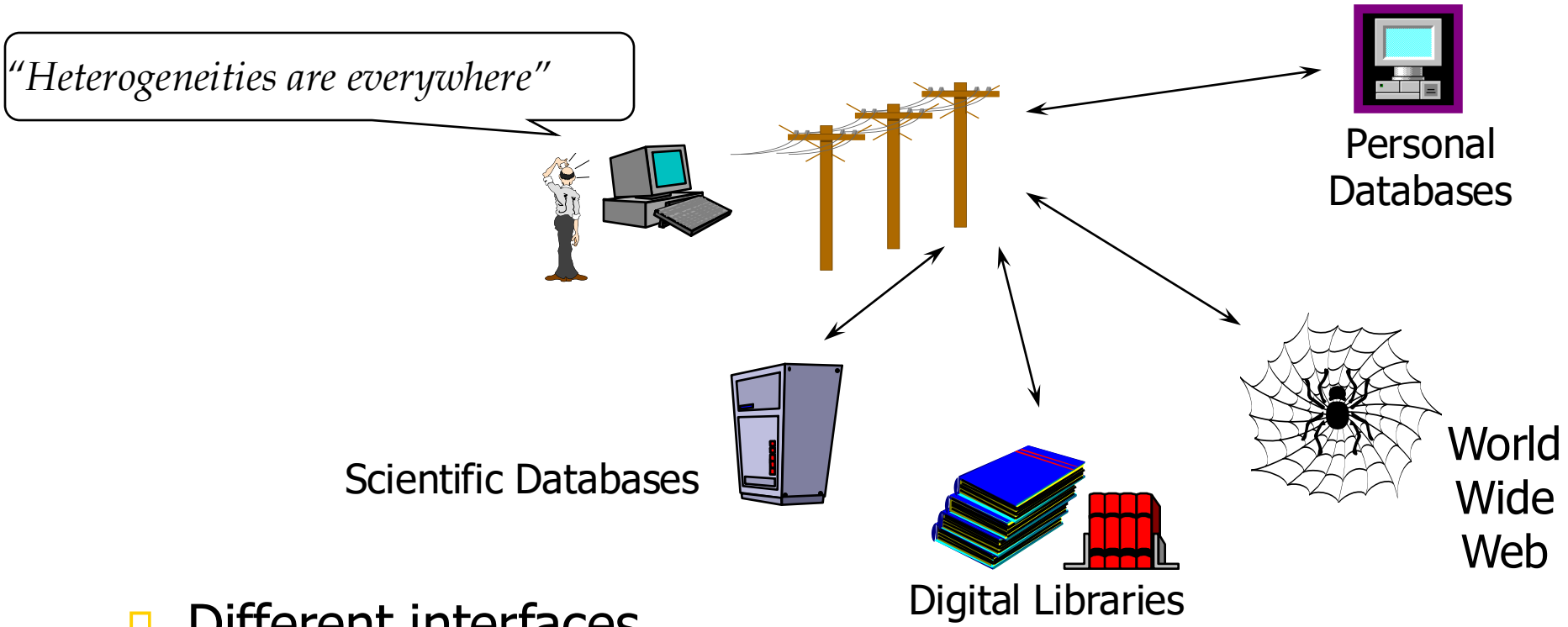
Why Data Warehousing?



Why Do We Need Data Warehouses?

- Consolidation of information resources
- Improved query performance
- Separate research and decision support functions from the operational systems
- Foundation for data mining, data visualization, advanced reporting and OLAP tools

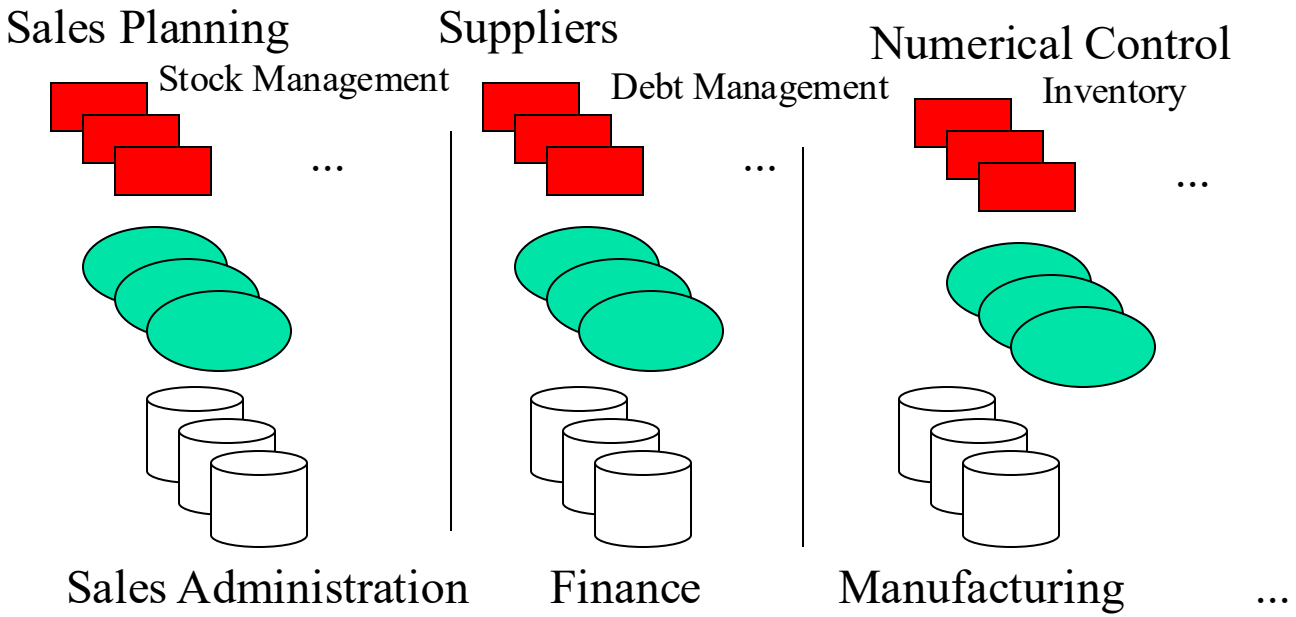
Root of the Problem: Heterogeneous Information Sources



- Different interfaces
- Different data representations
- Duplicate and inconsistent information

Additional Problem: Data Management in Large Enterprises

- Vertical fragmentation of informational systems (vertical stove pipes)
- Result of application (user)-driven development of operational systems

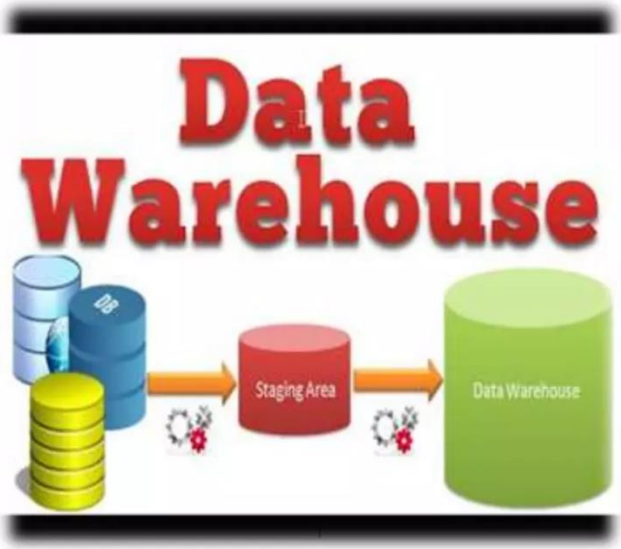


What is a Data Warehouse?

- 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.
- "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

What is a Data Warehouse? (cont.)

- Data warehouse is an environment, not a product.
- Data is often scattered across different database, it needs DW to get complete information
- It is aimed at effective integration of operational databases that enables strategic use of data.



What is a Data Warehouse Used for?

- In many organizations, we want a central "store" of all of our entities, concepts, metadata, and historical information
  - For doing data validation, complex mining, analysis, prediction, etc.
- One of the "modern" uses of the data warehouse is not only to support **analytics** but to serve as a reference to all of the entities in the organization
  - A cleaned, validated repository of what we know

    … which can be linked to by data sources

    … which may help with data cleaning

    … and which may be the basis of **data governance** (processes by which data is created and modified in a systematic way, e.g., to comply with gov't regulations)

What is a Data Warehouse Used for?

- Knowledge discovery
  - Making consolidated reports
  - Finding relationships and correlations
  - Data mining
  - Examples
    - Banks identifying credit risks
    - Insurance companies searching for fraud
- OLAP (Online Analytical Processing)
  - It contrasts with OLTP (on-line transaction processing) used to deal with the everyday running of one aspect of an enterprise.
  - OLTP systems are usually designed independently of each other and it is difficult for them to share information.

Data Warehouse Usage Examples

- Investment and Insurance Companies – Used to analyze customer and market trends and allied data pattern
- Retail Chains – Used for marketing and distribution to trace items, examine pricing policies and analyze buying trends of customers
- Healthcare – Used to generate treatment reports, share data with insurance companies and medical units.
- Airline – Used to carry out operation purposes like crew assignment, route profitability, frequent flyer program promotions, etc.

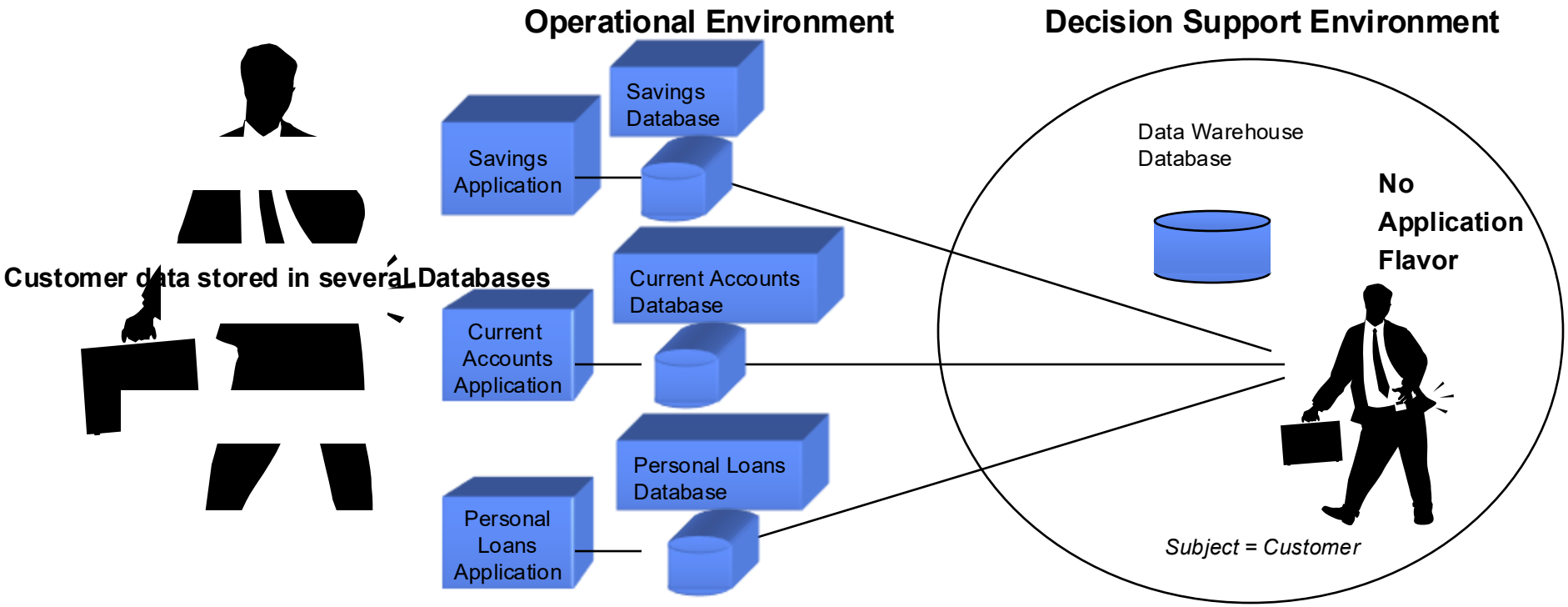
Data Warehouse—Subject-Oriented

- 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

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—Integrated (cont.)



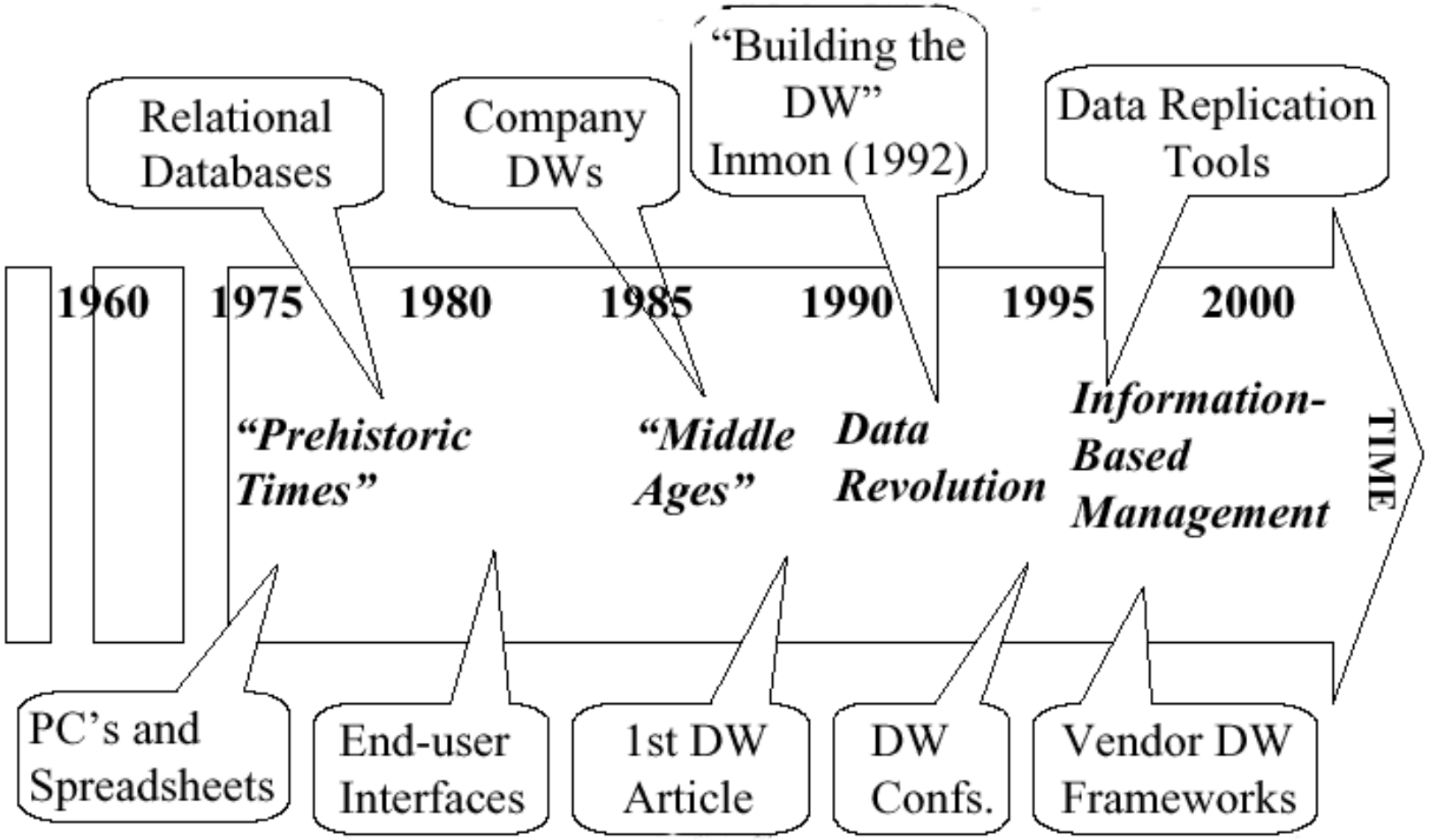
Data Warehouse—Time Variant

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

Data Warehouse—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*

History of Data Warehouse



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):
  - User and system orientation: customer vs. market
  - Data contents: current, detailed vs. historical, consolidated
  - Database design: ER + application vs. star + subject
  - View: current, local vs. evolutionary, integrated
  - Access patterns: update vs. read-only but complex queries

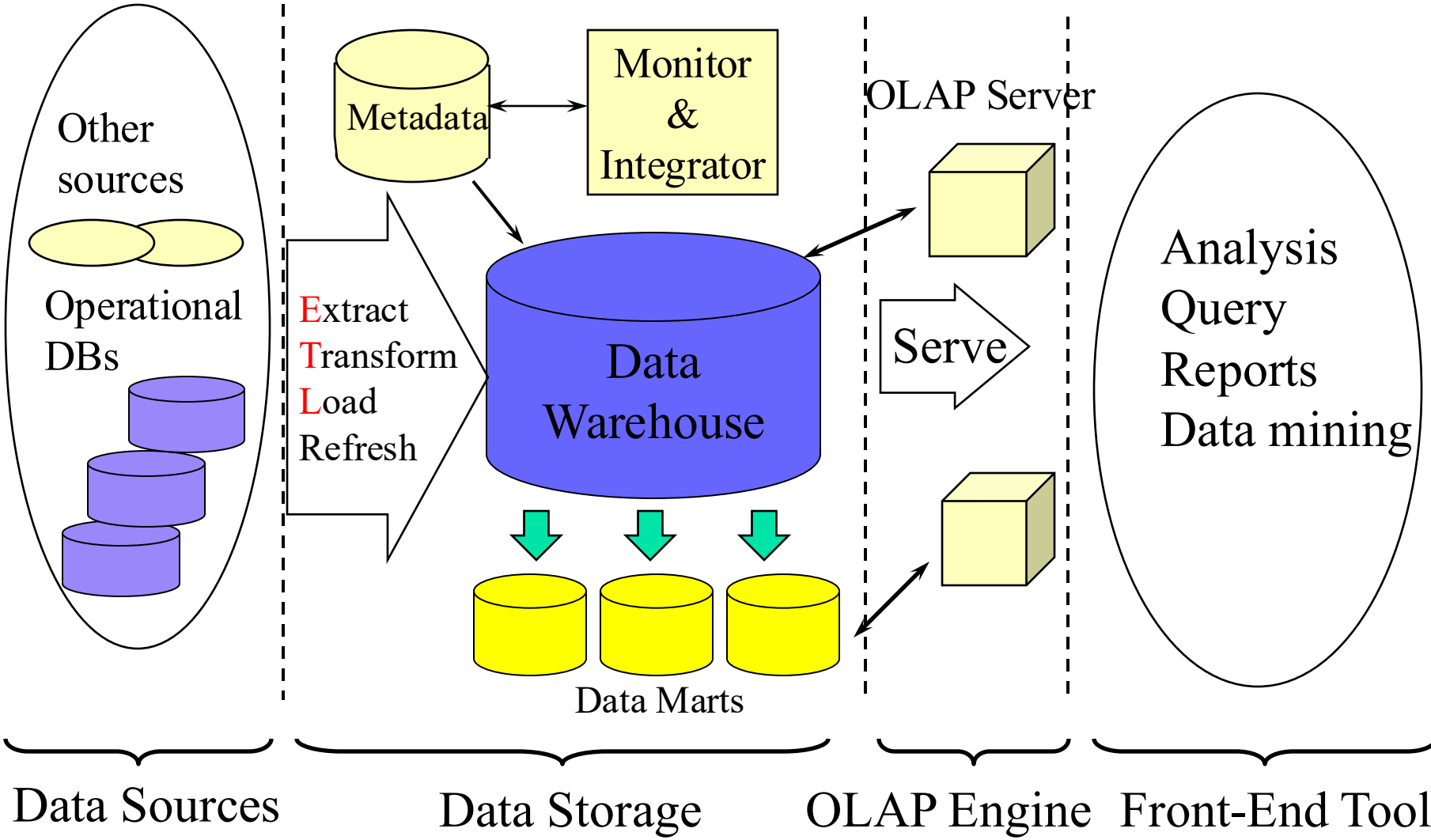
OLTP vs. OLAP

	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, flat relational isolated	historical, summarized, multidimensional integrated, consolidated
usage	repetitive	ad-hoc
access	read/write index/hash on prim. key	lots of scans
unit of work	short, simple transaction	complex query
# records accessed	tens	millions
#users	thousands	hundreds
DB size	100MB-GB	100GB-TB
metric	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
- 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



Types of Data Warehouse

- Enterprise warehouse
  - collects all of the information about subjects spanning the entire organization, offering a unified approach to data
- 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 indicies and partitions
- **Refresh**
  - propagate the updates from the data sources to the warehouse

Metadata Repository

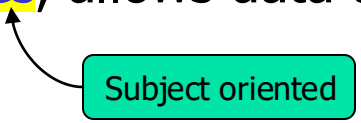
- **Meta data** is the data defining warehouse objects. It stores:
- Description of the structure of the data warehouse
  - schema, view, dimensions, hierarchies, derived data defn, data mart locations and contents
- Operational meta-data
  - data lineage (history of migrated data and transformation path), currency of data (active, archived, or purged), monitoring information (warehouse usage statistics, error reports, audit trails)
- 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

DW Modeling-

Data Cube and OLAP

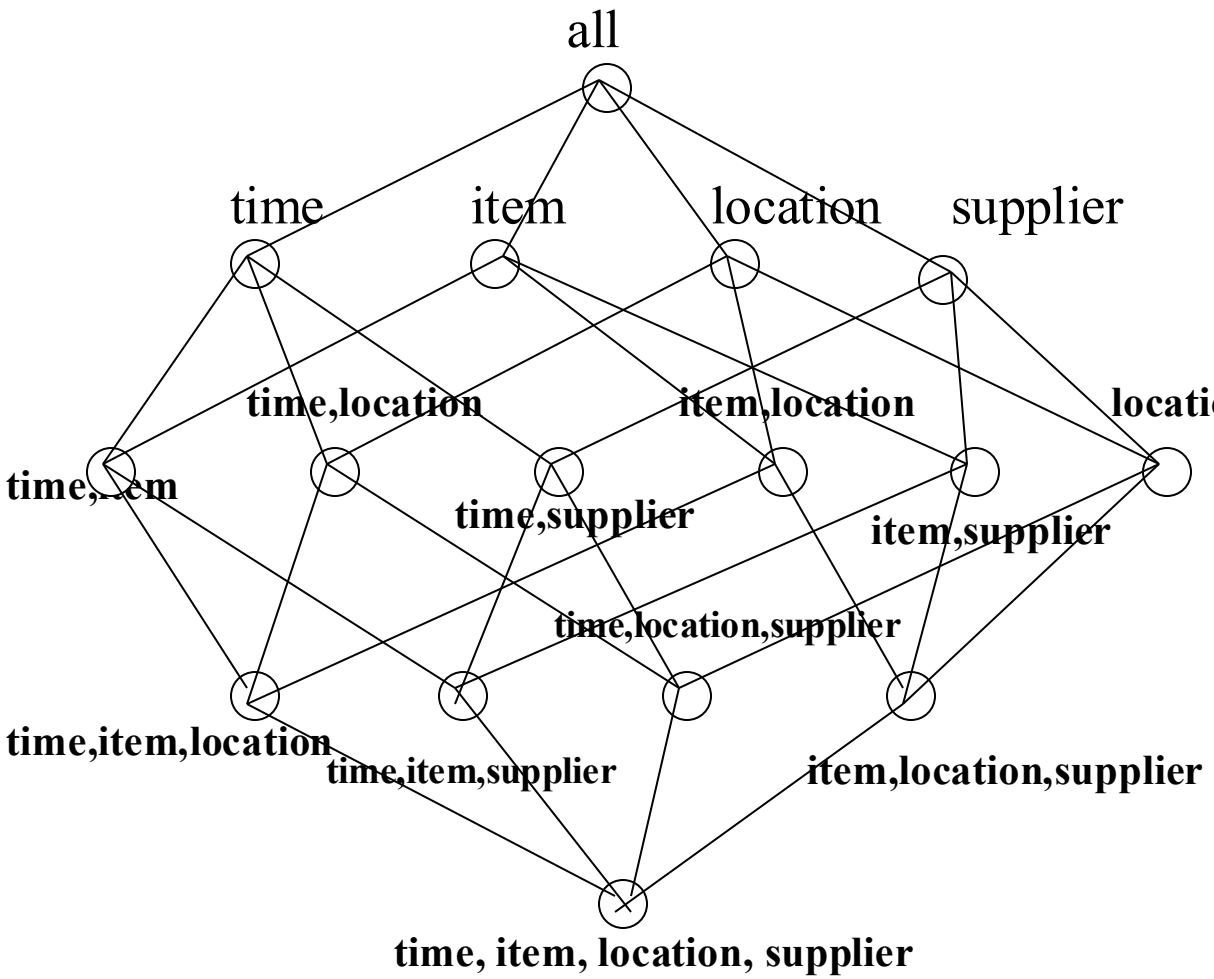
From Tables and Spreadsheets to Data Cubes

- 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
- 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



0-D (*apex*) cuboid

1-D cuboids

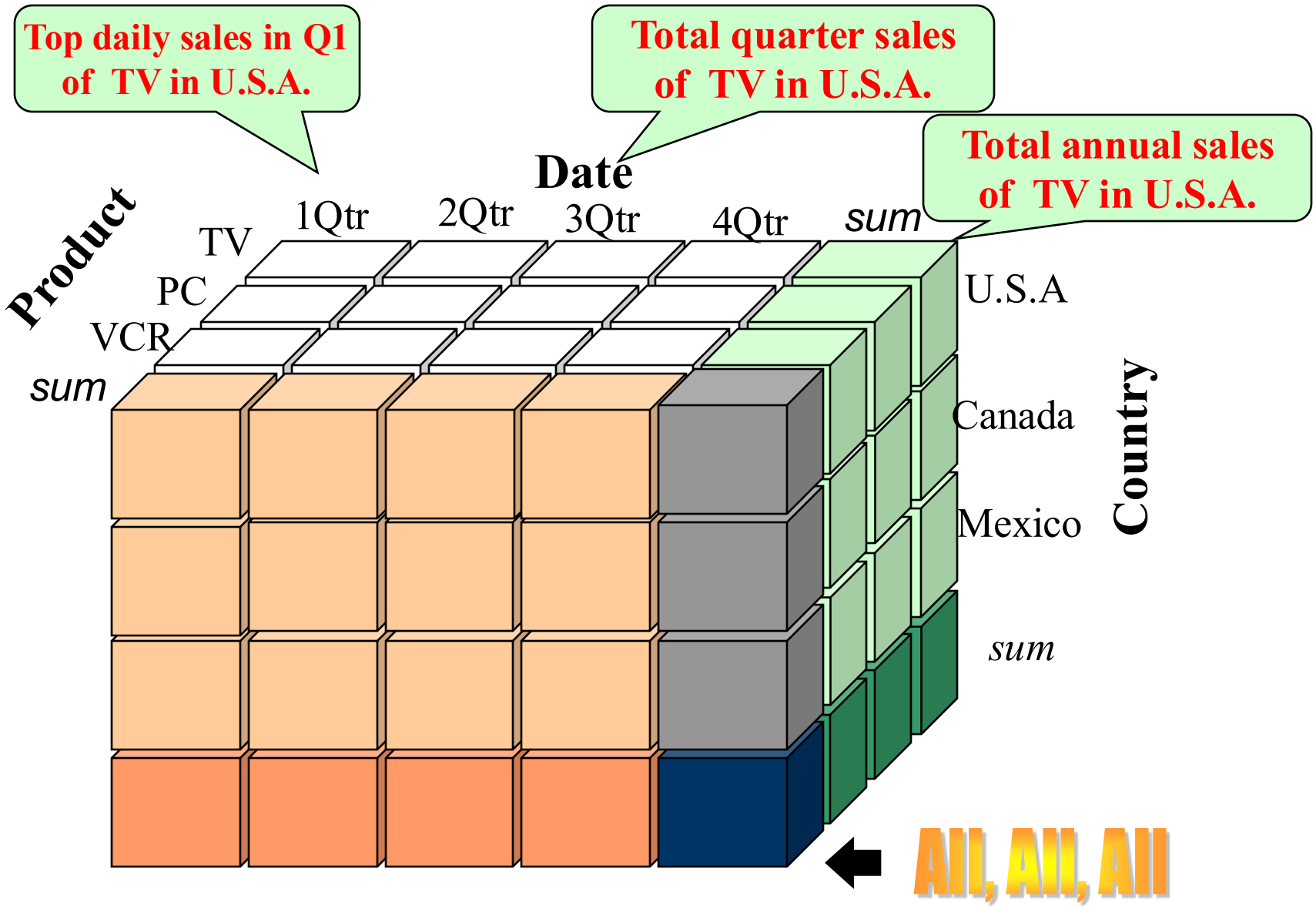
2-D cuboids

3-D cuboids

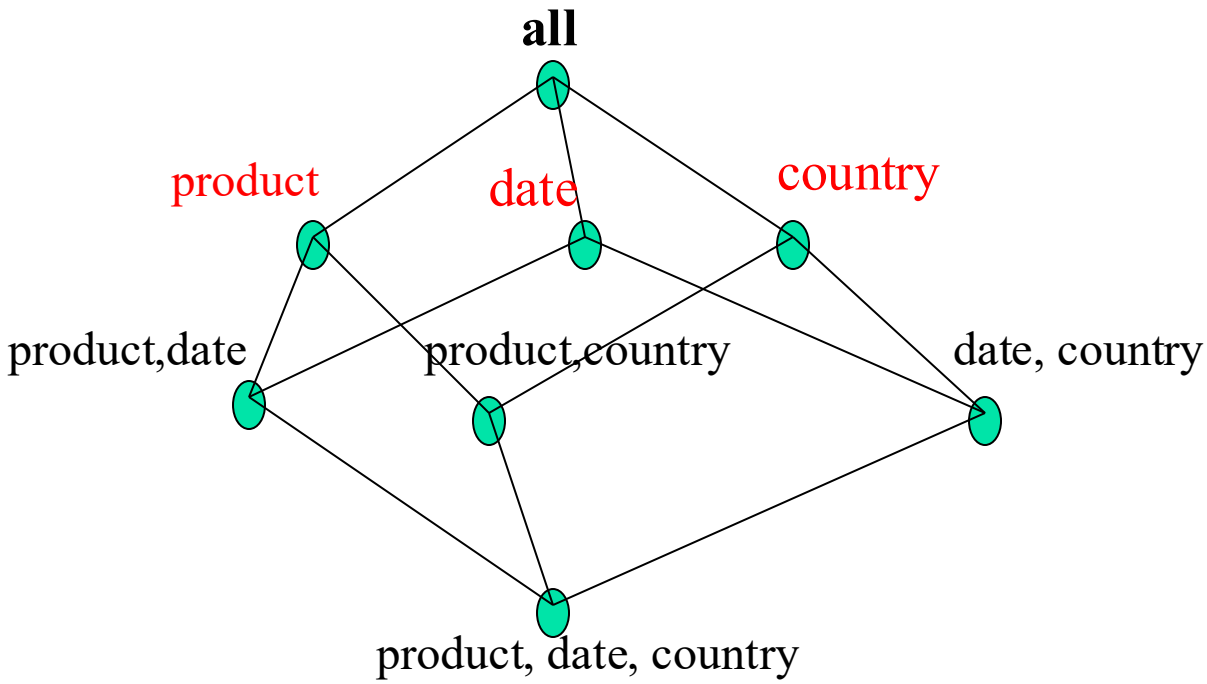
4-D (*base*) cuboid

Most ordered food in 2014 vs 2023

A Sample Data Cube



Cuboids Corresponding to the Cube



0-D (*apex*) cuboid

1-D cuboids

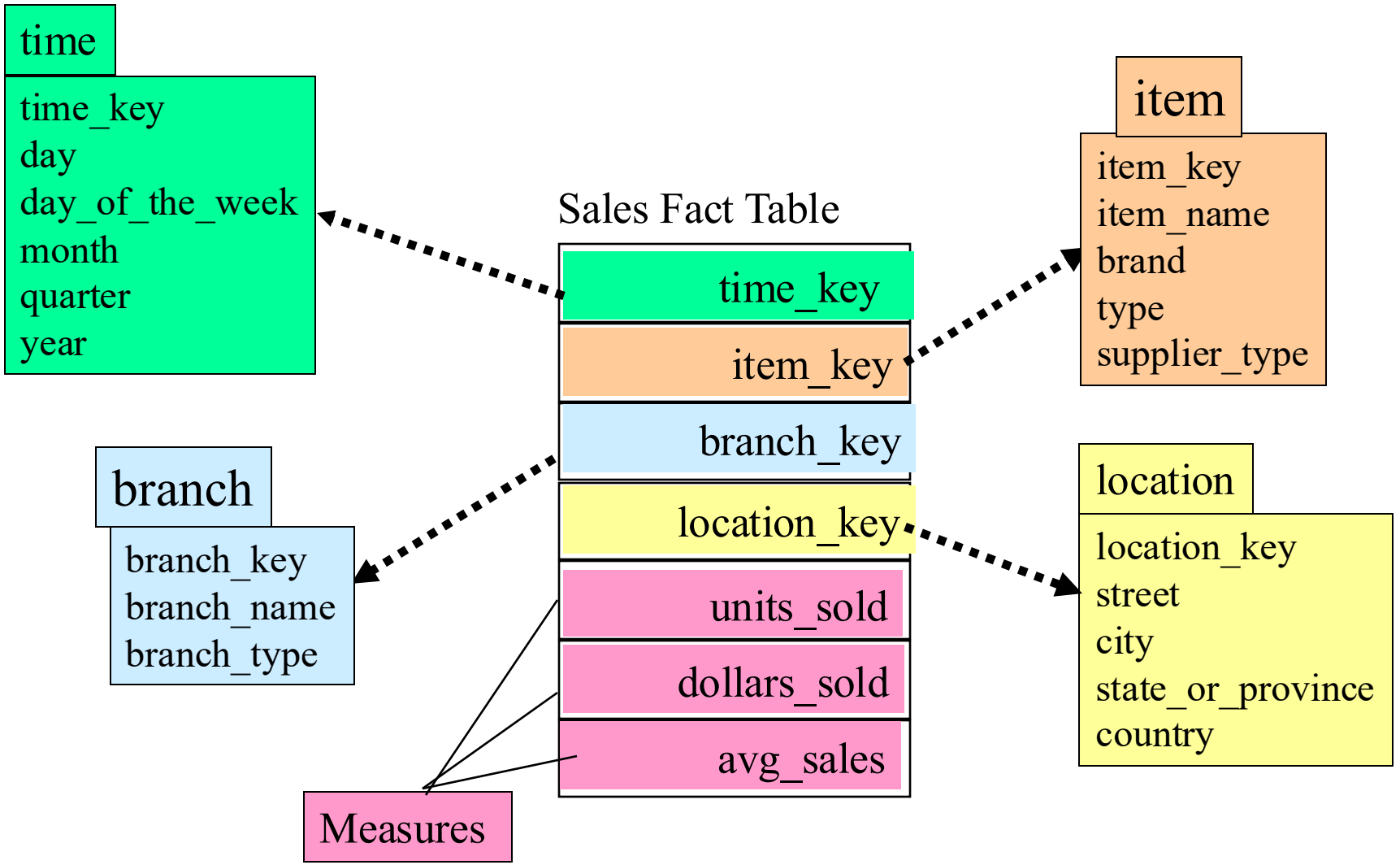
2-D cuboids

3-D (*base*) cuboid

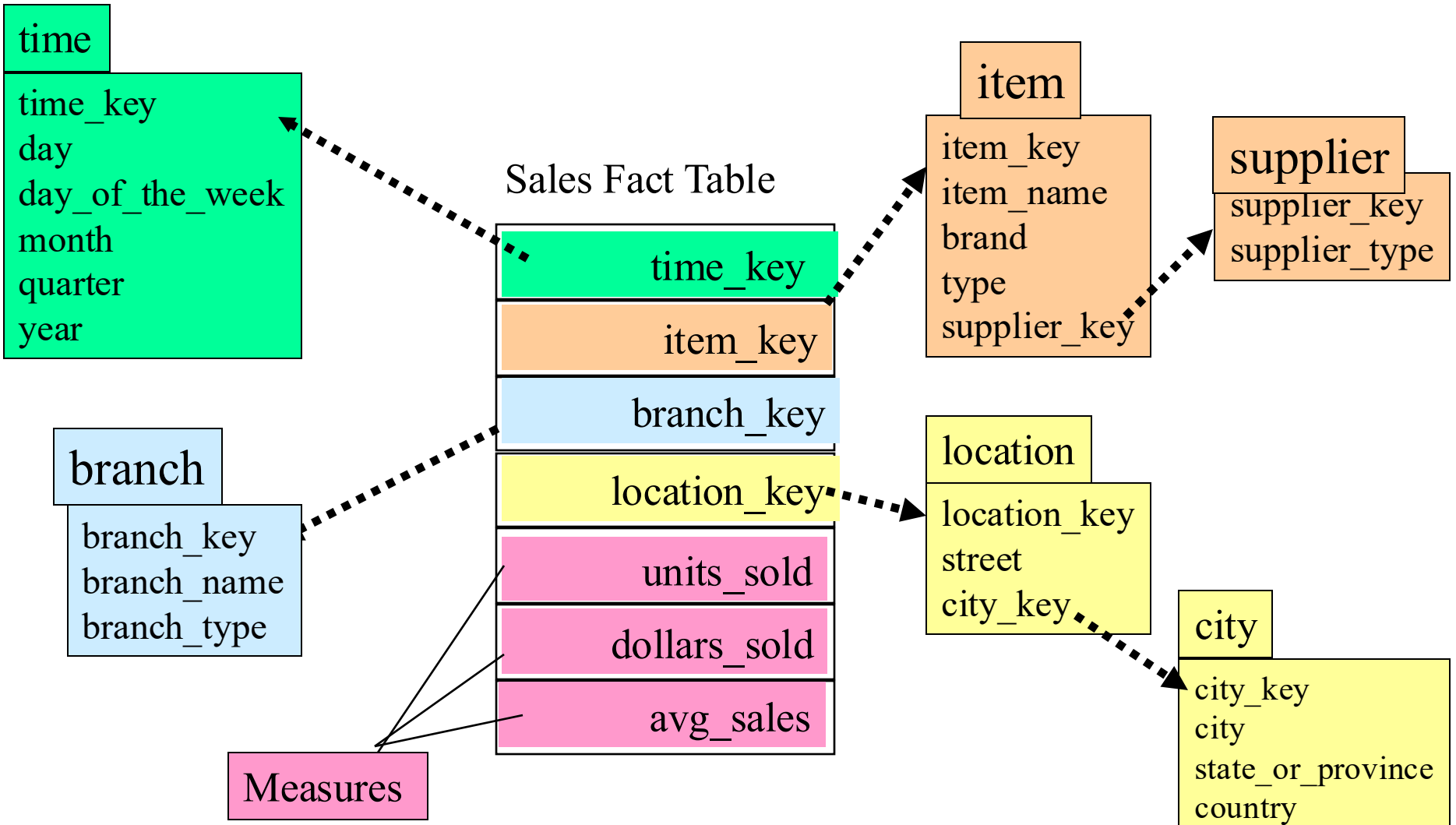
Conceptual Modeling of Data Warehouses

- 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

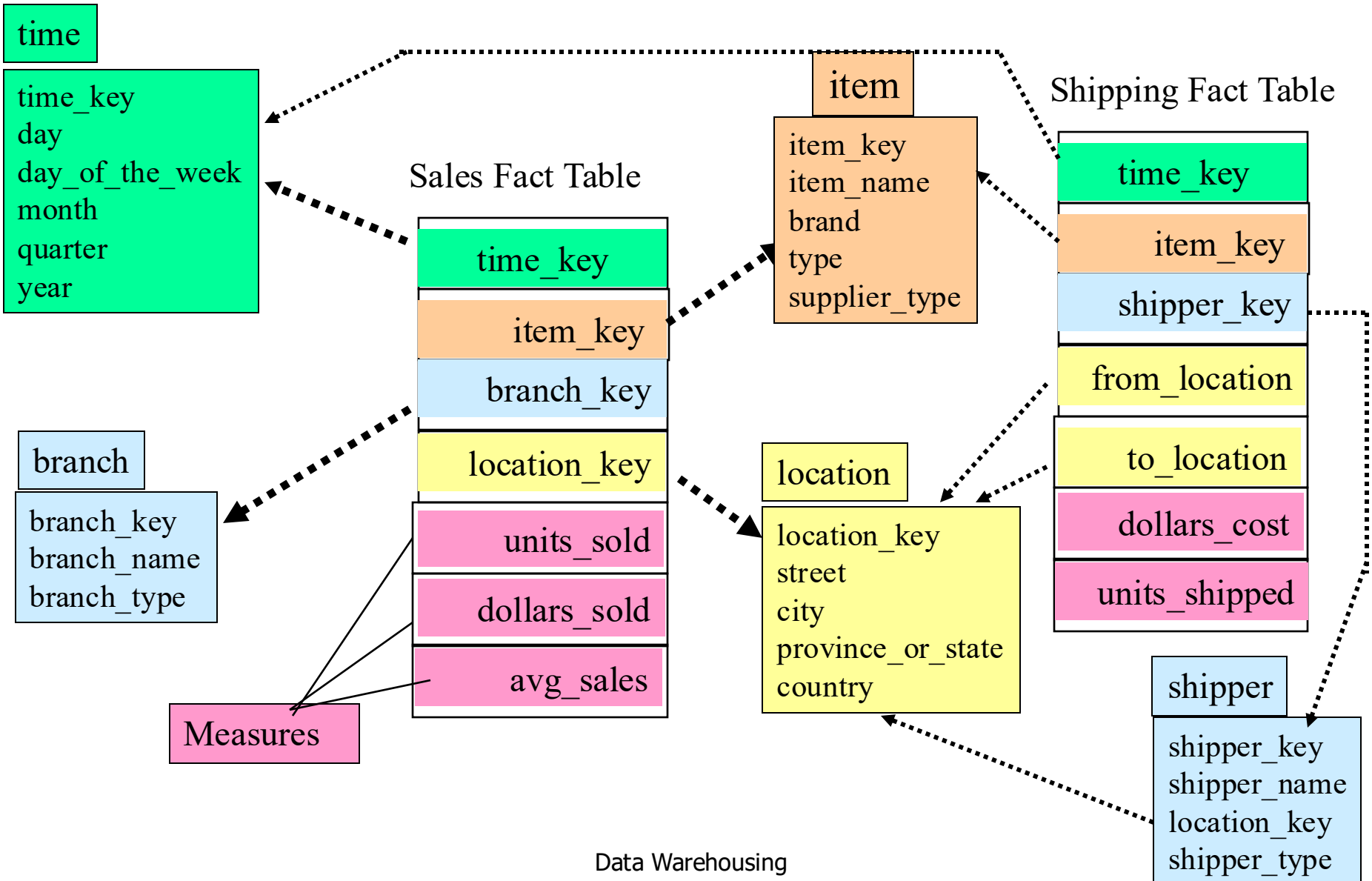
Example of Star Schema



Example of Snowflake Schema



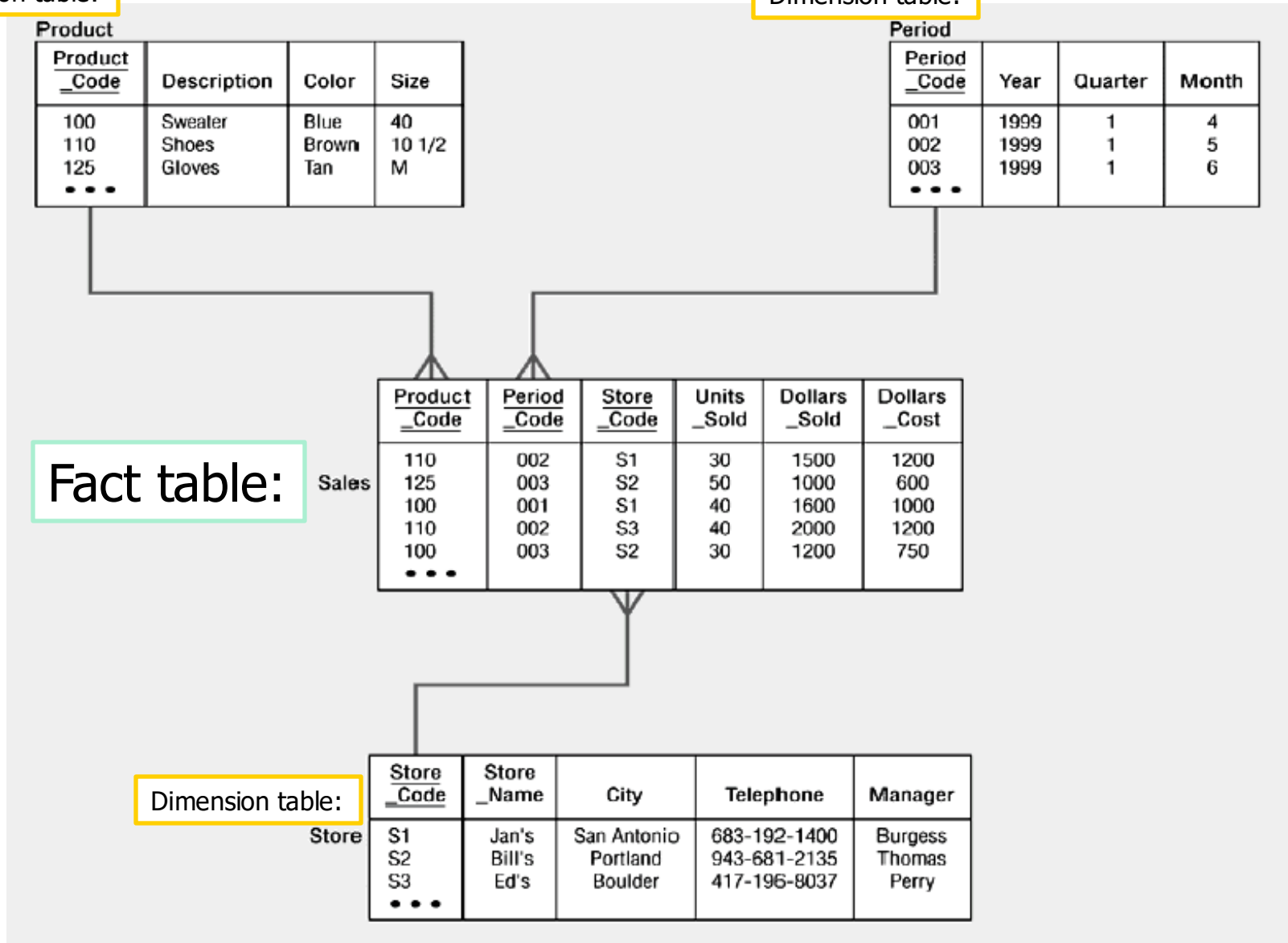
Example of Fact Constellation



Example of Star Schema with Data

Dimension table:

Dimension table:



Product

Product_Code	Description	Color	Size
100	Sweater	Blue	40
110	Shoes	Brown	10 1/2
125	Gloves	Tan	M
• • •			

Period

Period_Code	Year	Quarter	Month
001	1999	1	4
002	1999	1	5
003	1999	1	6
• • •			

Fact table:

Sales

Product_Code	Period_Code	Store_Code	Units_Sold	Dollars_Sold	Dollars_Cost
110	002	S1	30	1500	1200
125	003	S2	50	1000	600
100	001	S1	40	1600	1000
110	002	S3	40	2000	1200
100	003	S2	30	1200	750
• • •					

Dimension table:

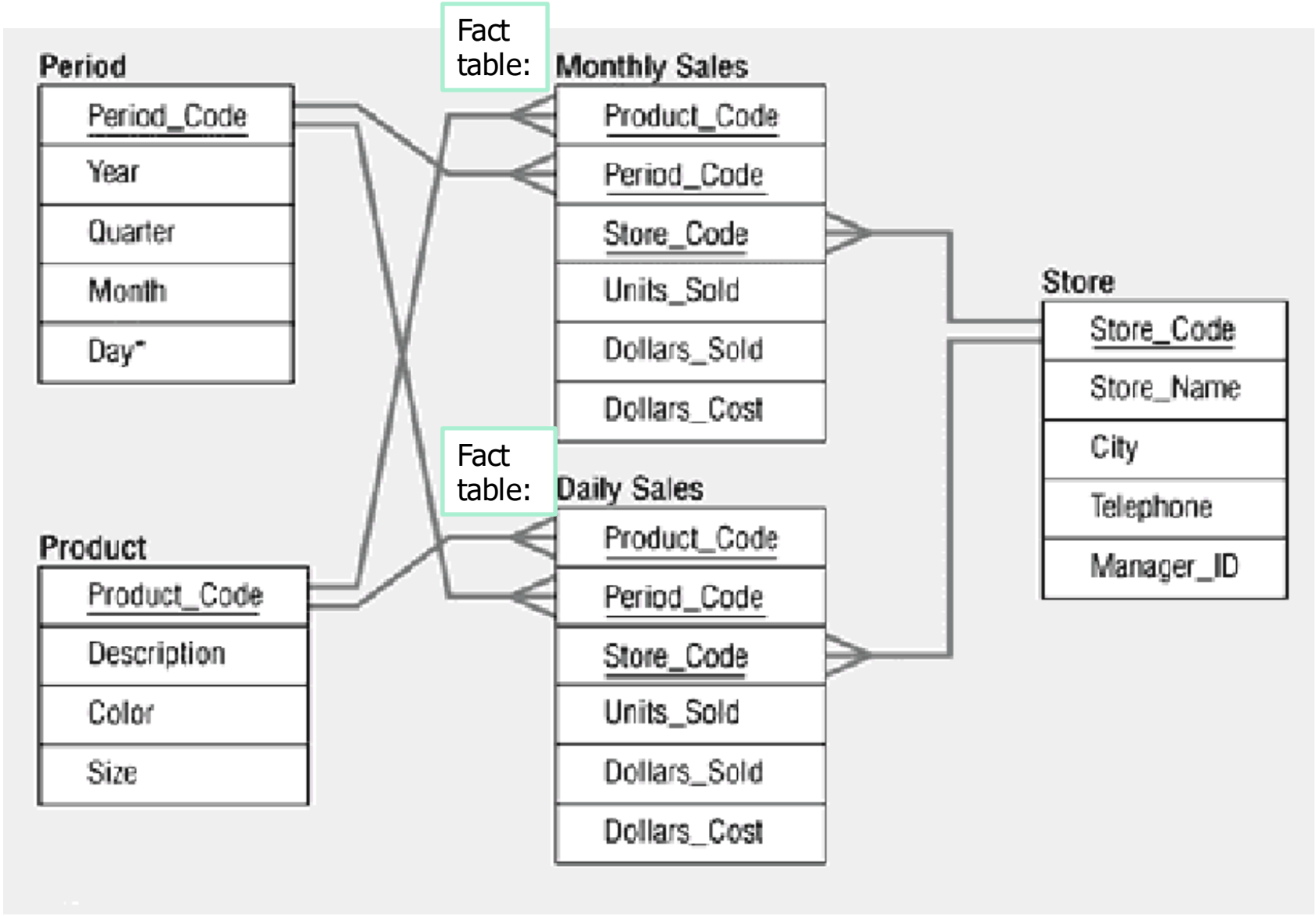
Store

Store_Code	Store_Name	City	Telephone	Manager
S1	Jan's	San Antonio	683-192-1400	Burgess
S2	Bill's	Portland	943-681-2135	Thomas
S3	Ed's	Boulder	417-196-8037	Perry
• • •				

Multiple Fact Tables ⇒ Galaxy Schema

- For performance or other reasons, we can define multiple fact tables in a given star schema
  - e.g. various users require different levels of aggregation
- Performance can be improved by defining a different fact table for each level of aggregation (see the example in next slide)
- Designers of DW need decide whether increased storage requirements are justified by the prospective performance improvement

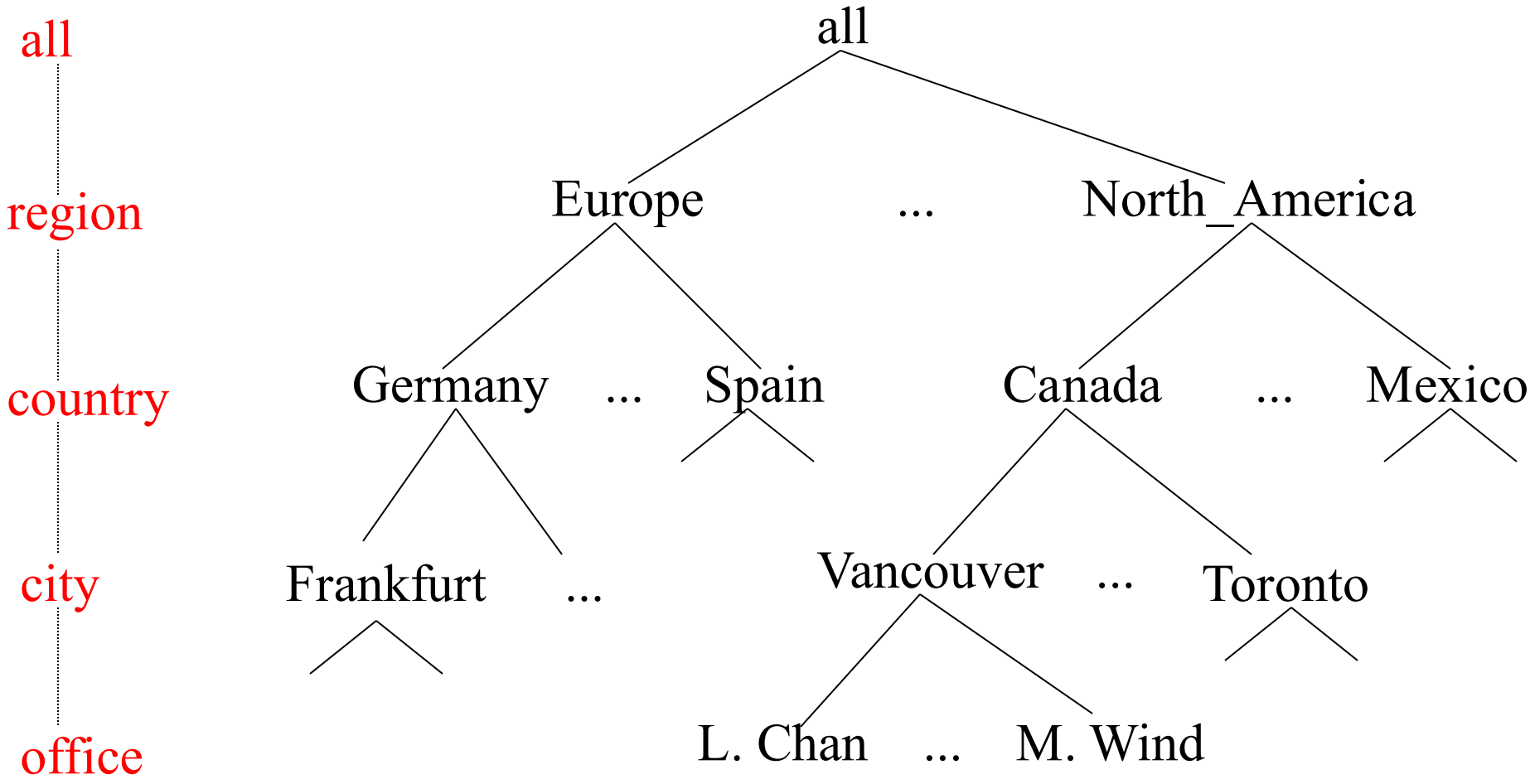
Star Schema with Two Fact Tables



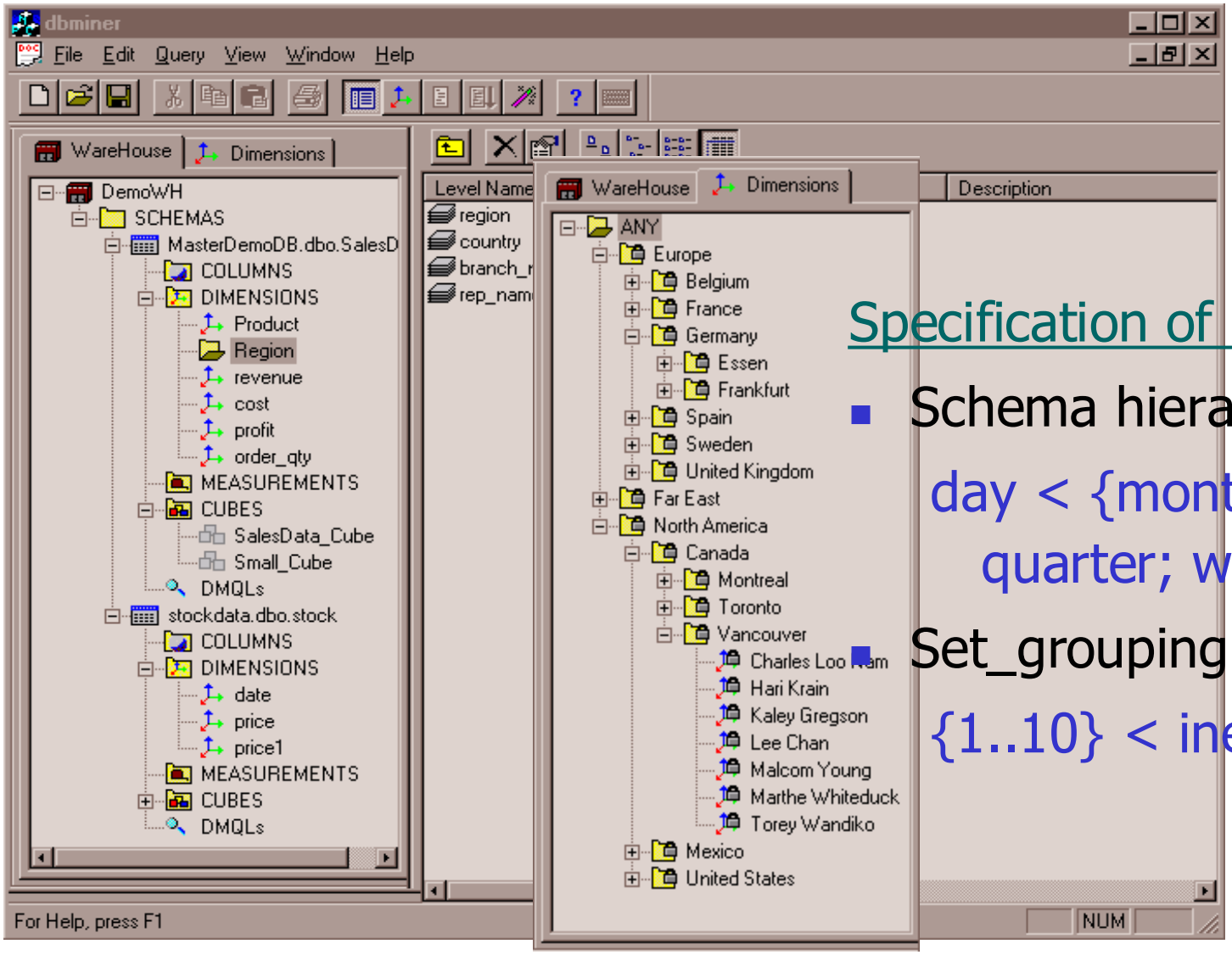
Snowflake Schema

- Sometimes a dimension in a star schema forms a natural hierarchy
  - e.g. a dimension named Market has geographic hierarchy:
    - several markets within a state
    - several markets within a region
    - several markets within a country
- When a dimension participates in a hierarchy, the designer has two basic choices.
  - Include all the information for the hierarchy in a single table
    - i.e., a big flat table
  - normalize the tables
    - resulting in an expanded schema $\Rightarrow$ the *snowflake schema*!
- A snowflake schema is an expanded version of a star schema in which all of the tables are fully normalized.

A Concept Hierarchy: Dimension (location)



View of Warehouses and Hierarchies



Specification of hierarchies

- Schema hierarchy

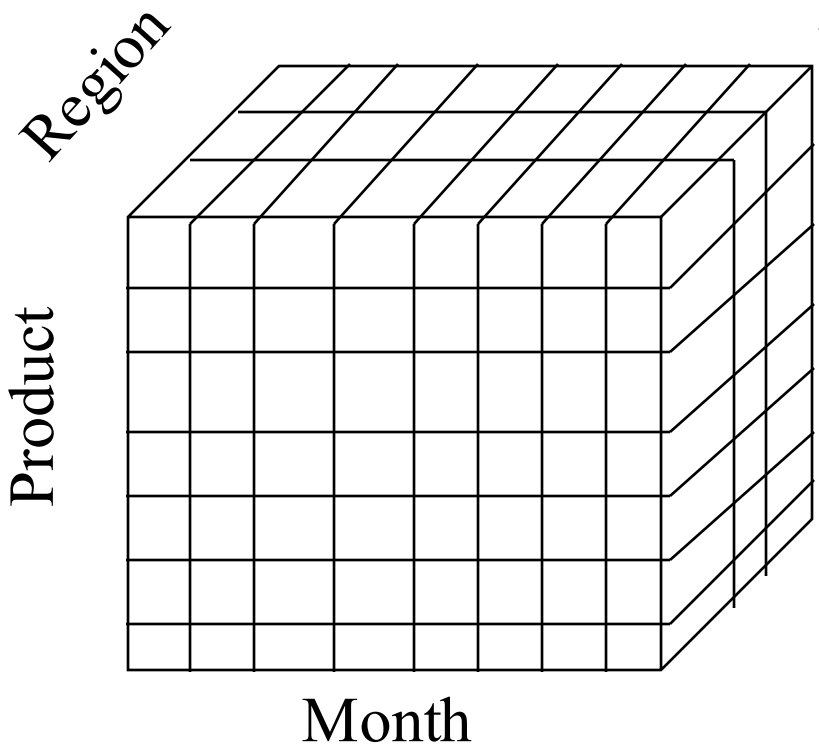
  day < {month < quarter; week} < year

- Set_grouping hierarchy

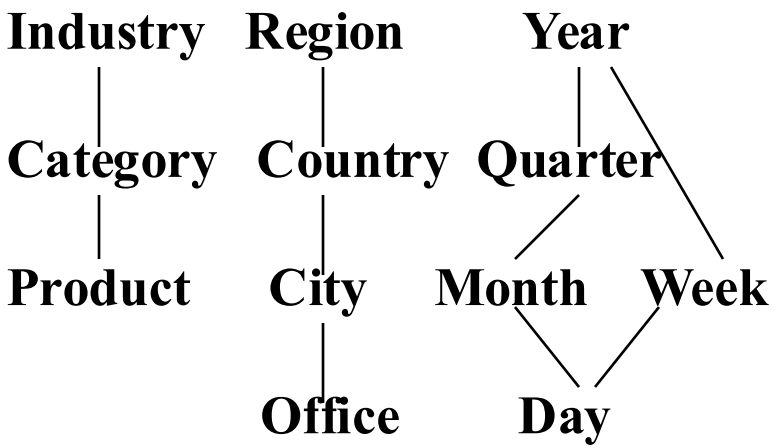
  {1..10} < inexpensive

Multidimensional Data

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



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



More data cube example: 4-D Data Cube

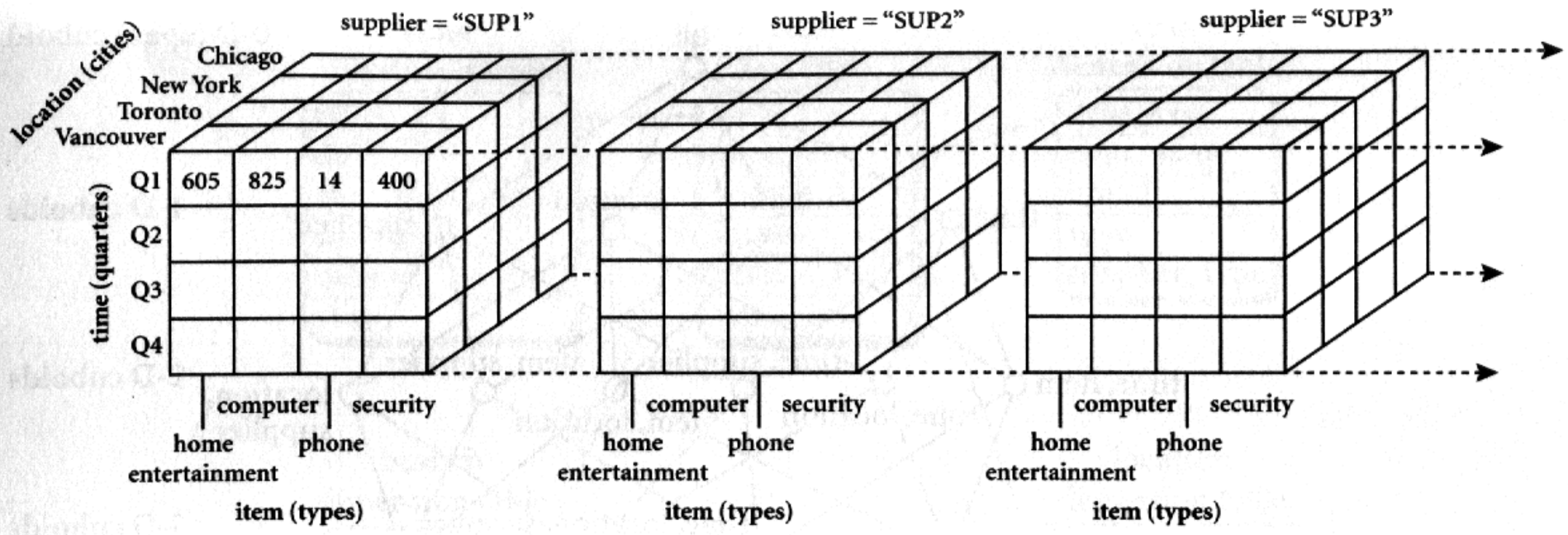


Figure 2.2 A 4-D data cube representation of sales data, according to the dimensions *time*, *item*, *location*, and *supplier*. The measure displayed is *dollars_sold* (in thousands). For improved readability, only some of the cube values are shown.

More data cube example:
3-D Data Cube

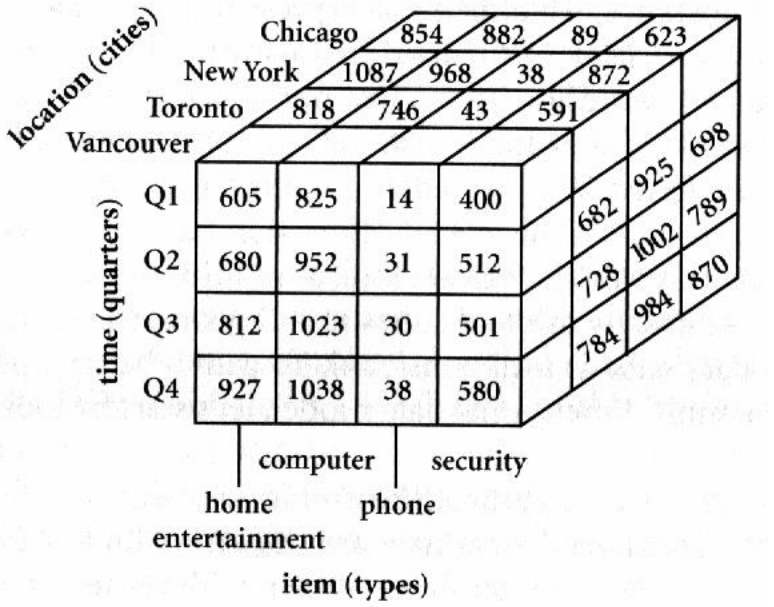


Table 2.3 A 3-D view of sales data for *AllElectronics*, according to the dimensions *time*, *item*, and *location*. The measure displayed is *dollars_sold* (in thousands).

	location = *"Chicago"*				***location*** = *"New York"*				***location*** = *"Toronto"*				***location*** = *"Vancouver"*			
	item				***item***				***item***				***item***			
time	*home ent.*	*comp.*	*phone*	*sec.*	*home ent.*	*comp.*	*phone*	*sec.*	*home ent.*	*comp.*	*phone*	*sec.*	*home 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

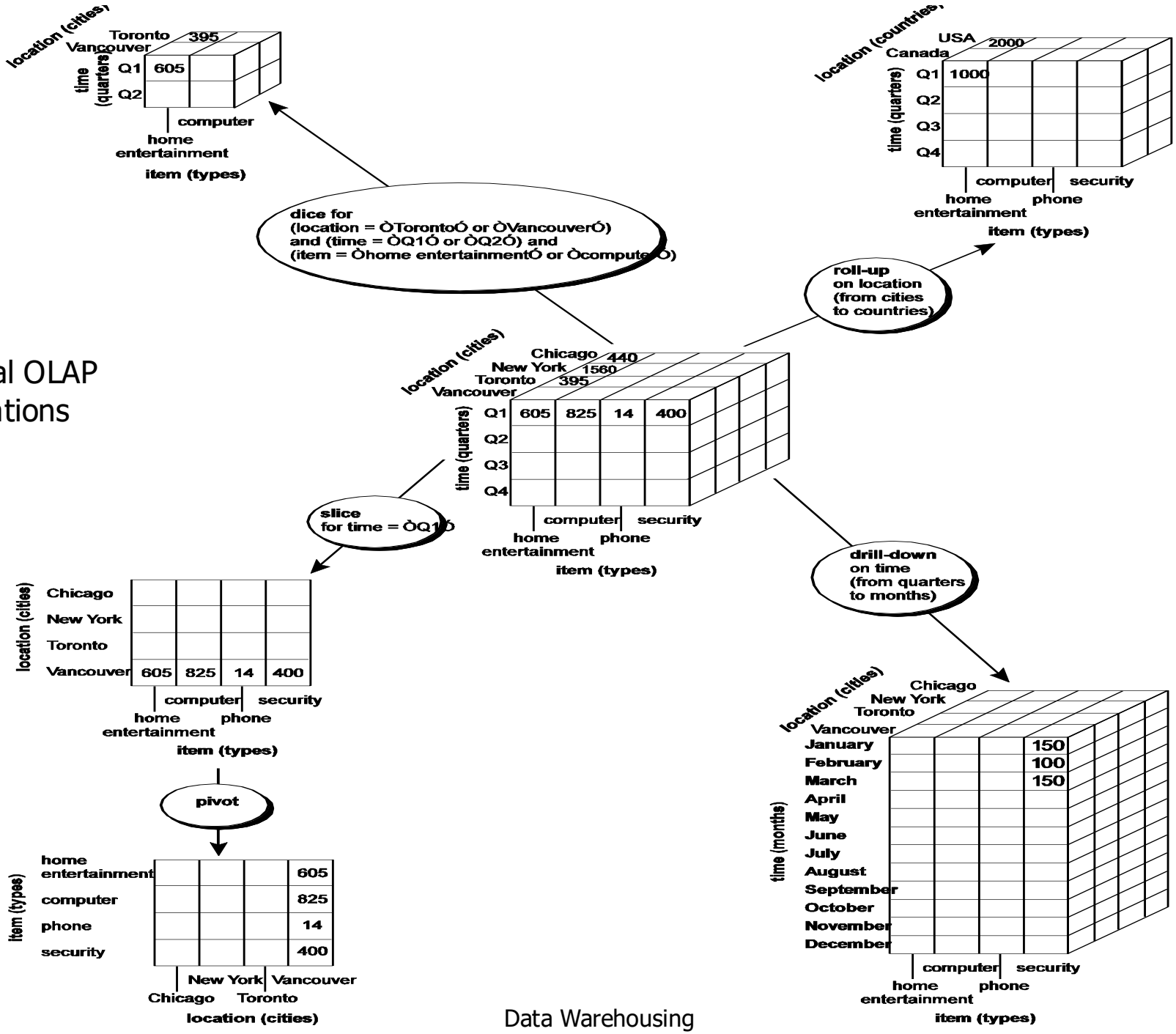
Data Cube Measures: Three Categories

- 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()
- 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()

Typical OLAP Operations

- Roll up (drill-up): summarize data
  - *by climbing up hierarchy or by dimension reduction*
- Drill down (roll down): reverse of roll-up
  - *from higher level summary to lower level summary or detailed data, or introducing new dimensions*
- Slice and dice: *project and select*
- Pivot (rotate):
  - *reorient the cube, visualization, 3D to series of 2D planes*
- 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)*

Typical OLAP Operations



Roll-up Operation

- Roll-up operation corresponds to taking the current aggregation level of fact values and doing a further aggregation on one (or more) of the dimensions
- That is equivalent to doing GROUP BY to this dimension(s) by using attribute hierarchy
- Roll-up operation can be understood as lowering the number of dimensions
- In this case, the measure is calculated without regard to dimensions to be omitted.

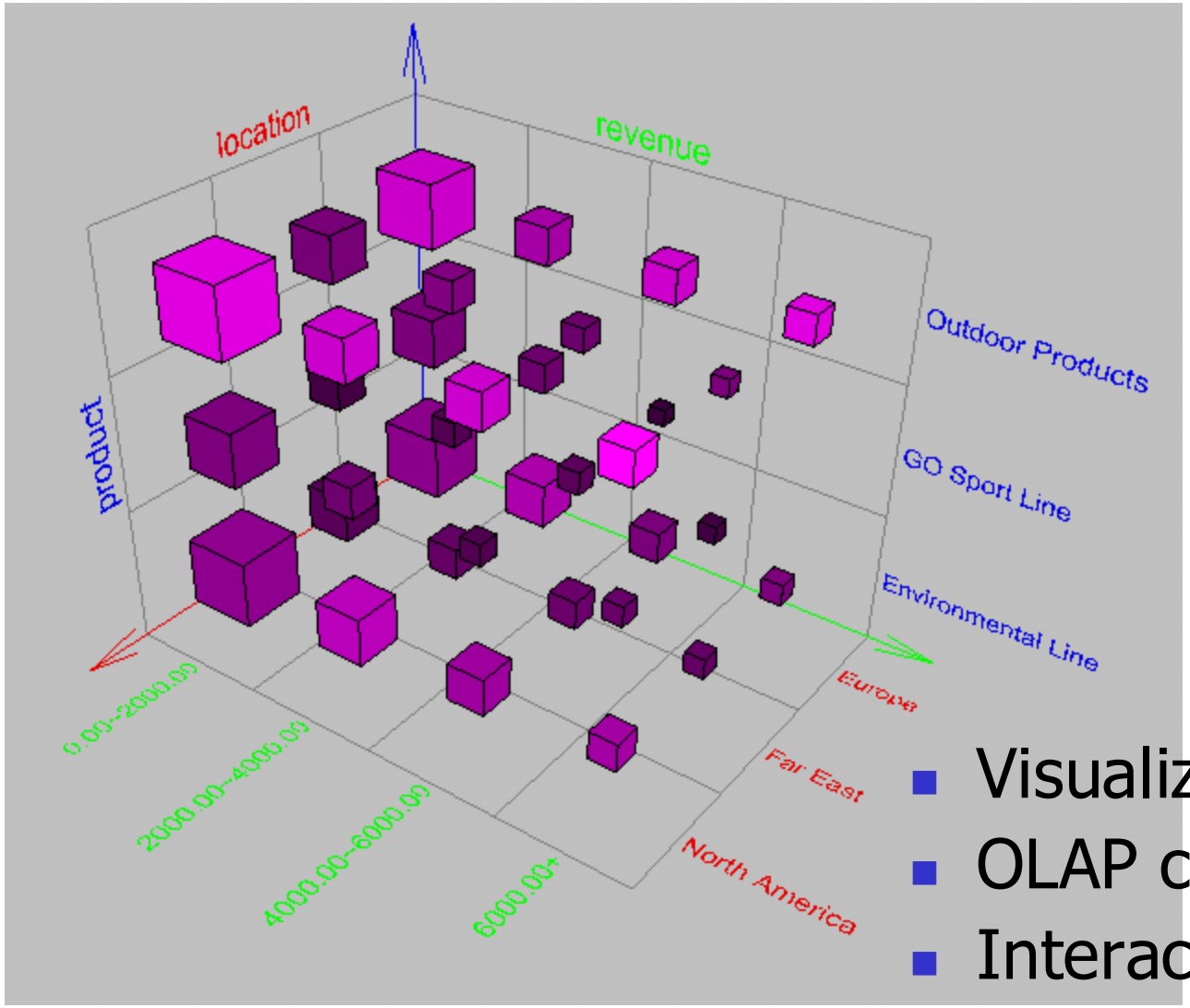
Drill Down Operation

- Analyzing a set of data at a finer level of detail
  - e.g. a summary report for the total sales of three package sizes for a given brand of paper towels
  - Further breakdown of sales by color within each of these package sizes
- A drill-down presentation is equivalent to adding another column to the original report (a color column)
- Executing a drill-down may require that the OLAP tool "reach back" to the DW to obtain the detailed data necessary for the drill-down
- Some tools even permit the OLAP tool to reach back to the operational data if necessary for a given query

Slicing and Dicing a Cube

- Slicing the data cube to produce a simple 2-D table or view
  - e.g. A slice is for the product named shoes
  - other views developed by simple "drag and drop"
  - This type of operation is often called "slicing and dicing" the cube
- Slice-and-dice operations reduce the number of dimensions by taking a projection of facts on a subset of dimensions and for some selected values of dimensions that are being dropped.
- Closely related to slicing and dicing is data pivoting
  - This term refers to rotating the view for a particular data point, to obtain another perspective
  - The analyst could pivot this view to obtain the sale of shoes by store for the same month

Browsing a Data Cube



- Visualization
- OLAP capabilities
- Interactive manipulation

Data Warehouse

Design and Usage

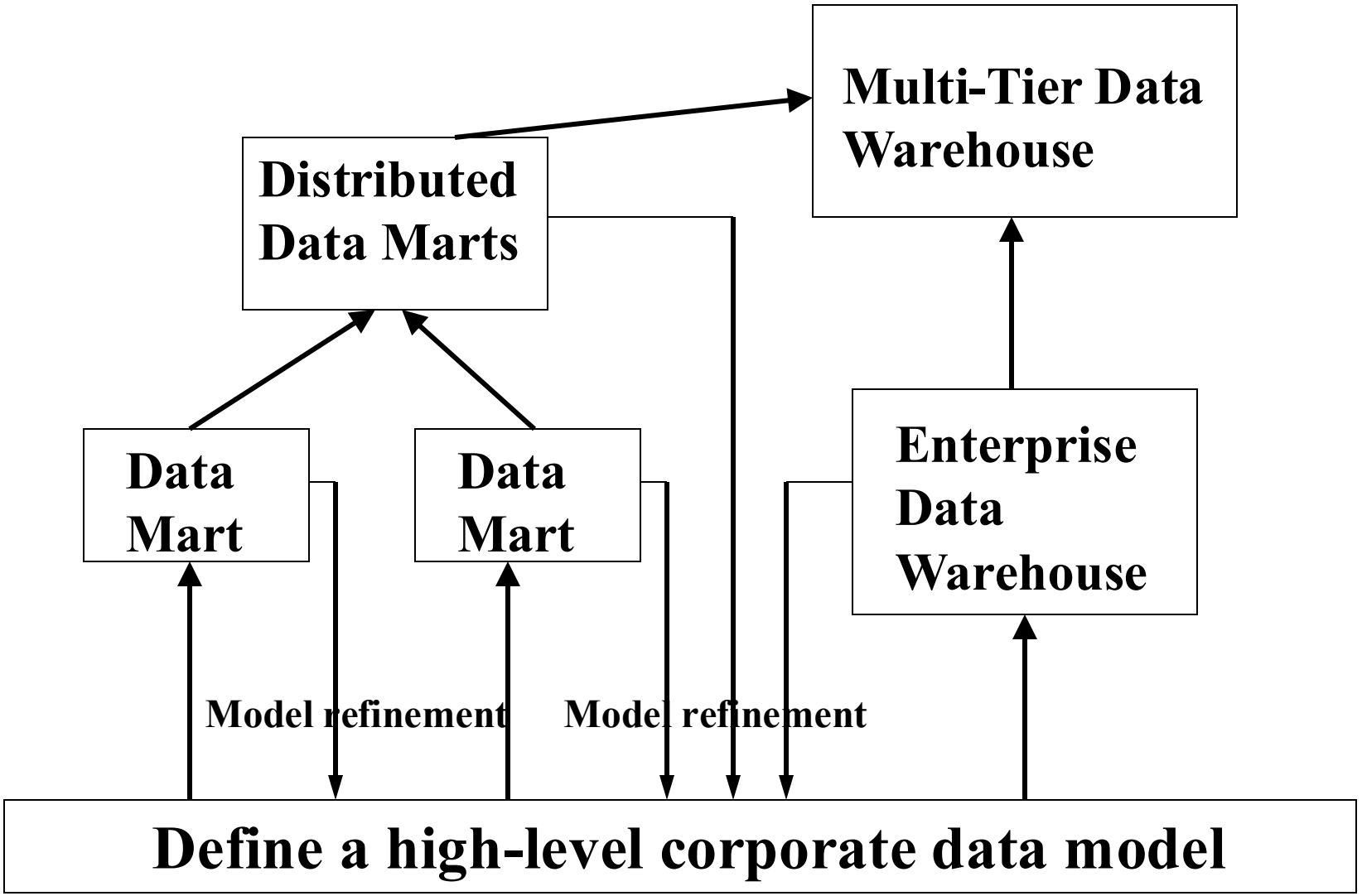
Design of Data Warehouse: A Business Analysis Framework

- Four views regarding the design of a data warehouse
  - Data warehouse view
    - consists of fact tables and dimension tables
  - Top-down view
    - allows selection of the relevant information necessary for the data warehouse
  - Data source view
    - exposes the information being captured, stored, and managed by operational systems
  - Business query view
    - sees the perspectives of data in the warehouse from the view of end-user

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
- **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

Data Warehouse Development: A Recommended Approach



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

From On-Line Analytical Processing (OLAP) 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

OLAP Server Architectures

- Relational OLAP (ROLAP)
  - Use relational or extended-relational DBMS to store and manage warehouse data and OLAP middle ware
  - Include optimization of DBMS backend, implementation of aggregation navigation logic, and additional tools and services
  - Greater scalability
- Multidimensional OLAP (MOLAP)
  - Sparse array-based multidimensional storage engine
  - Fast indexing to pre-computed summarized data
- Hybrid OLAP (HOLAP) (e.g., Microsoft SQLServer)
  - Flexibility, e.g., low level: relational, high-level: array
- Specialized SQL servers (e.g., Redbricks)
  - Specialized support for SQL queries over star/snowflake schemas

Data Warehouse

Implementation Issues

Views, OLAP and Materialization

- Views are frequently used in decision support systems to allow data analyst to consider just his/her part of the business
- Decision support queries are typically aggregate queries over very large fact tables
- To allow fast answers, view materialization is a viable alternative
- The DW itself is considered to be a (materialized) view of the operational databases and external data sources
- When deciding which views to materialize, one should consider the following issues:
  - How many queries potentially can be speeded up?
  - How much space will be required to store the views?, and
  - How will the views influence the DW maintenance (update)?

Choosing Views to Materialize

- The choice of views to materialize is complex, because the range of views that can be used for a query evaluation is very broad
- On the other hand, materialized views strongly influence the storage occupancy and DW maintenance time
- It is the goal to materialize a small, carefully chosen set of views that can be used to evaluate the majority of important queries
- Conversely, once the set of materialized views is determined, the query processor has to choose one of them to evaluate a given query
- ROLAP Engines offer automatic advise to a DW/DB administrator on which materialized views to build and which ones to drop
- That advising is done on the base of statistical data gathered during a Data Warehouse querying

Maintenance of Materialized Views

- Making a view consistent with its DW base tables is called view refreshing
- If the cost of algorithms for view refreshing is proportional to the change of the view, they are said to be incremental
- A view maintenance policy is a decision about when a view has to be refreshed

View Updates

- View update can be:
  - Immediate (within the same transaction that updates the base table), and
  - Deferred (some times after the base tables are updated)
- Deferred update can be done:
  - During the time view is used for a query evaluation for the first time after base table update,
  - Periodically, in regular time intervals
  - Forced, after a certain number of base table updates

View Refreshing and Aggregates

- A special consideration is needed when aggregate views are refreshed
- Views containing distributive aggregates are refreshed without any problem
- Views containing algebraic aggregates are easily refreshed if they contain all other necessary data
- Views containing holistic aggregates are hard to refresh, they are rather built from scratch, again

To summarize (view materialization)

- OLAP queries are typically aggregate queries over very large fact tables
- To allow fast answers, view materialization is a viable alternative
- Materialized views strongly influence the storage occupancy and DW maintenance time
- It is the goal to materialize a small, carefully chosen set of views that can be used to evaluate most of the important queries

Populating and Updating a DW

- Data warehousing systems use a variety of software tools for:
  - Data extraction
  - Data cleaning
  - DW loading, and
  - DW refreshing

Typically referred as ETL
(Extract, Transform, and Load)

- All these tools have the goal to provide data of high quality for the decision making purpose

Data Extraction

- Data from operational databases and external sources are extracted by using gateways
- A gateway is an application program interface that allows a client program to generate SQL statements to be executed at a server
- Common examples of gateways are:
  - Open Database Connectivity (ODBC)
  - Object Loading and Embedding for Databases (OLE), and
  - Java Database Connectivity (JDBC)

Data Cleaning Tools

Data cleaning tools transforms, cleans, and discovers violation of constraints in input data

- *Data migration* tools allow simple data transformation rules to be specified:
  - Replace *Surname* by *Last_Name*
  - Convert *pound* to *kg*
- *Data scrubbing* tools are more sophisticated, they use domain specific knowledge (rules of behavior in the real system) to do cleaning of data from various sources
  - Use functional dependency *ProductID->Prod_Name* to clean product data from production and marketing databases,
  - Convert country code number part of a telephone number into country name (e.g. 852 into Hong Kong)
  - Fill in missing *Address* data
- *Data auditing* tools are used to scan data and discover strange patterns (data mining)
  - Products that have been never sold;
  - Exceptionally large attribute values (although within limits allowed)

Data Loading

- Before loading data some additional data preprocessing has to be done:
  - Sorting
  - Summarization,
  - Aggregation,
  - Building indexes, and
  - Building materialized views
- The load utilities have to deal with very large volumes of data during small time slots (night)
- Sequential loads would take weeks (or more), so pipelined and parallel are exploited instead

Loading data

- Doing a full load has advantage of using the current version of a data warehouse for queries during the time the load is in progress
- But doing a full load can last too long
- To reduce the amount of data, incremental loading during refresh is used instead
- Only the updated operational tuples influence data to be inserted

Data Refreshing

- Refreshing a DW consists of propagating updates on source data (operational and external) to corresponding updates of base and derived data (in the DW)
- The DW refresh policy has to concern two issues:
  - Frequency, and
  - Procedures

  of data refreshing

Data Refreshing Frequency

- Data refreshing frequency depends on user needs and OLTP traffic
- Usually, a DW is refreshed periodically (daily or weekly)
- But, if users need current data, it is necessary to propagate every relevant update from OLTP data to OLAP data
- Also, if the OLTP update traffic is high and the DW refreshment frequency is low, data volumes during refreshment may overwhelm the refreshment utility
- So, OLTP update traffic also influence refreshment policy (high traffic leads to frequent updates)

Data Refreshing Procedures

- Generally, DW refreshing is made using one of the following two techniques:
  - Data shipping and
  - Transaction shipping
- Both techniques suppose that the operational DBMS support replication servers that incrementally propagate updates from a primary database to replicas
- If the operational database system is a legacy one, and does not support replication, extracting the whole source database can be the only choice

To summarize (DW Population)

- Data extraction is done by means of gateways
- Data cleaning software reconciles inconsistency and discovers integrity violations and suspicious data patterns
- Data loading can be either full, or incremental
- Some terminology:
  - Source data are data from operational DBs and external sources
  - Base data are DW fact table or dimension table data
  - Derived data is DW data produced by materializing views and building auxiliary access structures (indexes)

Summary

- Data warehousing: A multi-dimensional model of a data warehouse
  - A data cube consists of *dimensions* & *measures*
  - Star schema, snowflake schema, fact constellations
  - OLAP operations: drilling, rolling, slicing, dicing and pivoting
- Data Warehouse Architecture, Design, and Usage
  - Multi-tiered architecture
  - Business analysis design framework
  - Information processing, analytical processing, data mining, OLAM (Online Analytical Mining)
- Implementation Issues: Efficient computation of data cubes
  - Partial vs. full vs. no materialization
  - Populating DW

Acknowledgement

- Slides/Materials of
  - J. Han et al.'s DM: Concepts and Techniques textbook
  - https://web2.utc.edu/~djy471/
- Photos from Internet

References

- S. Chaudhuri and U. Dayal. An overview of data warehousing and OLAP technology. ACM SIGMOD Record, 26:65-74, 1997
- A. Gupta and I. S. Mumick. Materialized Views: Techniques, Implementations, and Applications. MIT Press, 1999.