Data Warehousing, Dimensional Modeling & OLAP

Agenda

- The Need for Data Warehousing;
- Data Warehouse Defined;
- Benefits of Data Warehousing ;
- Data Warehouse Architecture;
- Data Warehouse and Data Marts;
- The Star Schema; The Snowflake Schema;
- Fact Constellation Schema or Families of Star
- Need for Online Analytical Processing; OLTP vs OLAP; OLAP Operations in a cube: Roll-up, Drilldown, Slice, Dice, Pivot ;

Need for Data Warehousing

- Depending on the industries the various applications are:
  - Order processing, general ledger, inventory, human resources, payroll, in-patient billing, checking accounts, insurance claims, and many more…
- These applications are important systems that **run businesses.**
  - They process orders, maintain inventory, keep the accounting books, service the clients, receive payments, and process claims. Without these computer systems, no **modern business can survive.**
  - As an enterprise grows larger, hundreds of computer applications are needed to support the various business processes.
  - They gather, store, and process all the data needed to successfully perform the daily routine operations.
  - They provide online information and produce a variety of reports to **monitor and run the business.**

Need for Data Warehousing

- Since 1990s, as **businesses grew more complex**, corporations spread **globally**, and **competition became fiercer**, business executives became desperate for information to stay competitive and **improve the bottom line.**
- The operational computer systems did provide information to run the day-to-day operations but what the **executives** needed were different kinds of information that could be used readily to make **strategic decisions.**
    - The decision makers wanted to know which geographic regions to focus on, which product lines to expand, and which markets to strengthen.
    - They needed the type of information with proper content and format that could help them make such strategic decisions.
    - We may call this type of information strategic information as different from operational information. The operational systems, important as they were, **could not provide strategic information.**

Need for Data Warehousing

- **Data warehousing** is a new paradigm specifically intended to provide vital **strategic information.**
  - Who needs strategic information in an enterprise?
  - What exactly do we mean by strategic information?
- The executives and managers who are responsible for keeping the enterprise competitive need information to make proper decisions.
- They need information to:
  - Formulate the business strategies,
  - Establish goals,
  - Set objectives, and
  - Monitor results.

Need for Data Warehousing

- **Some examples of business objectives:**
  - Retain the present customer base
  - Increase the customer base by 15% over the next 5 years
  - Improve product quality levels in the top five product groups
  - Gain market share by 10% in the next 3 years
  - Bring three new products to market in 2 years
  - Increase sales by 15% in the North East Division

Demand for Strategic Information

- Strategic information is **not for running the day-to-day operations** of the business.
- It is not intended to produce an invoice, make a shipment, settle a claim, or post a withdrawal from a bank account.
- Strategic information is far more important for the continued health and survival of the corporation.
- **Critical business decisions depend on the availability of proper strategic information in an enterprise.**

Characteristics of Strategic Information

- The desired characteristics of strategic information.

INTEGRATED	**Must have a single, enterprise-wide view.**
DATA INTEGRITY	**Information must be accurate and must conform to business rules.**
ACCESSIBLE	**Easily accessible with intuitive access paths, and responsive for analysis.**
CREDIBLE	**Every business factor must have one and only one value.**
TIMELY	**Information must be available within the stipulated time frame.**

Inability to Provide Information

- IT receives too many **ad hoc requests**, resulting in a large overload. With limited resources, IT is unable to respond to the numerous requests in a timely fashion.
- Requests are too numerous; they also **keep changing** all the time. The users need more reports to expand and understand the earlier reports.
- The users find that they get into the spiral of asking for more and **more supplementary reports,** so they sometimes adapt by asking for every possible combination, which only increases the IT load even further.
- The users have to **depend on IT** to provide the information. They are not able to access the information themselves interactively.
- The information environment ideally suited for strategic decision making has to be very **flexible and advantageous for analysis**. IT has been unable to provide such an environment.

Operational versus Decision Support System

- The fundamental reason for the inability to provide strategic information is that we have been trying all along to provide strategic information from the **operational systems.**
- These operational systems such as order processing, inventory control, claims processing, outpatient billing, and so on are not designed or intended to provide strategic information.
- If we need the ability to provide strategic information, we must get the information from altogether different types of systems.
- Only specially designed decision support systems or informational systems can provide strategic information.

Operational and Informational Systems

	OPERATIONAL	INFORMATIONAL
Data Content	Current values	Archived, Derived, Summarized
Data Structure	Optimized for transactions	Optimized for complex queries
Access Frequency	High	Medium to low
Access Type	Read, update, delete	Read
Usage	Predictable, repetitive	Ad hoc, random, heuristic
Response Time	Sub-seconds	Several seconds to minutes
Users	Large number	Relatively small number

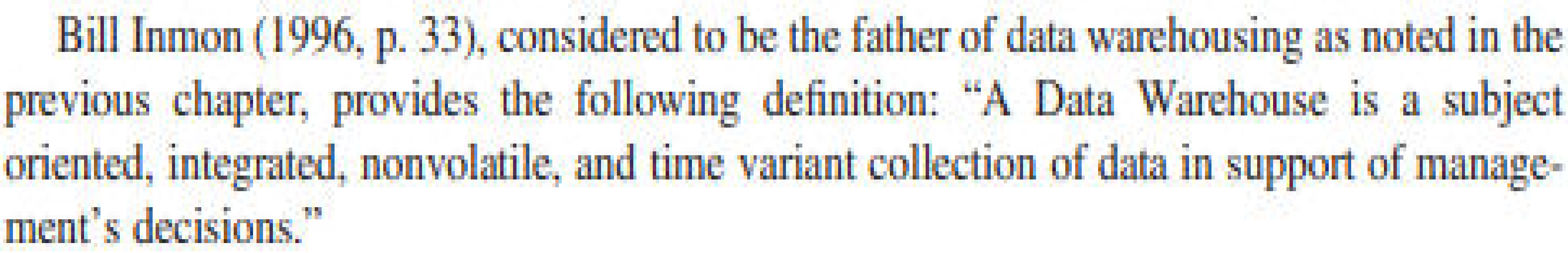
Processing Requirements in the New Environment

- The processing in the new environment for strategic information will have to be **analytical**. There are at least four levels of analytical processing requirements:
  1. Running of simple queries and reports against **current and historical data.**
  2. Ability to perform "what if" analysis in many different ways.
  3. Ability to query, step back, analyze, and then continue the process to any desired length.
  4. Ability to spot historical trends and apply them in future interactive processes.

This new system environment that users desperately need to obtain strategic information happens to be the new paradigm of data warehousing.

Data Warehouse Defined

- The strong conclusion that data warehousing is the only viable solution for providing strategic information.
- The data warehouse is an informational environment that:
  - Provides an integrated and total view of the enterprise.
  - Makes the enterprise's current and historical information easily available for strategic decision making.
  - Makes decision-support transactions possible without hampering operational systems.
  - Renders the organization's information consistent.
  - Presents a flexible and interactive source of strategic information.



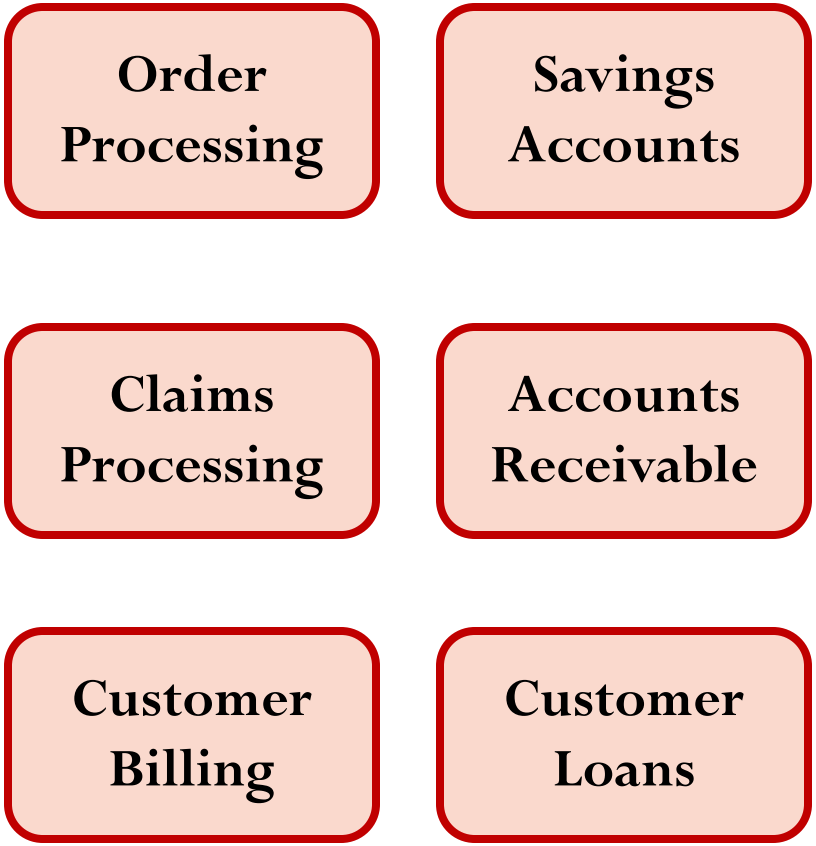
Bill Inmon (1996, p. 33), considered to be the father of data warehousing as noted in the previous chapter, provides the following definition: "A Data Warehouse is a subject oriented, integrated, nonvolatile, and time variant collection of data in support of management's decisions."

Subject-Oriented Data

- In every industry, data sets are organized around individual applications to support those particular operational systems.
- These individual data sets have to provide data for the specific applications to perform the specific functions efficiently.
- Therefore, the data sets for each application need to be organized around that specific application.
- In striking contrast, in the **data warehouse**, data is stored by real-world business subjects or events, not by applications.
- The data in a data warehouse is organized in such away that all the data sets relating to the same real-world business subject or event is tied together.

Subject-Oriented Data cont..

- **What are business subjects?**
  - Business subjects differ from enterprise to enterprise. These are the subjects critical for the enterprise.
  - For a manufacturing company, sales, shipments, and inventory are critical business subjects.
  - For a retail store, sales at the check-out counter would be a critical business subject.



Order Processing

Savings Accounts

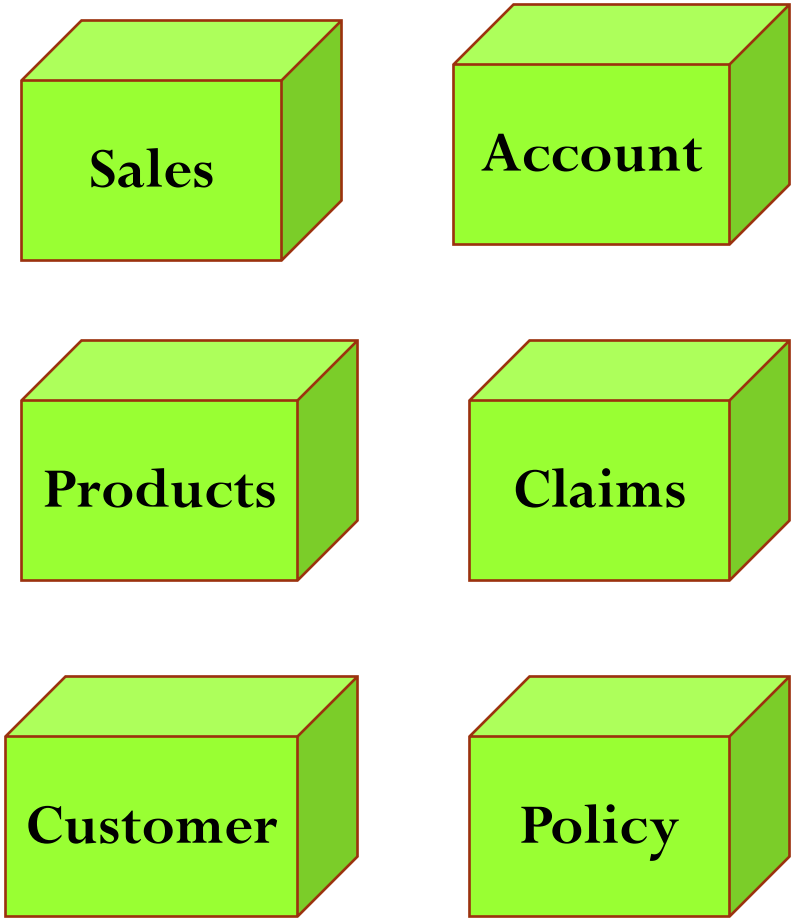
Claims Processing

Accounts Receivable

Customer Billing

Customer Loans

Operational Applications



Sales

Account

Products

Claims

Customer

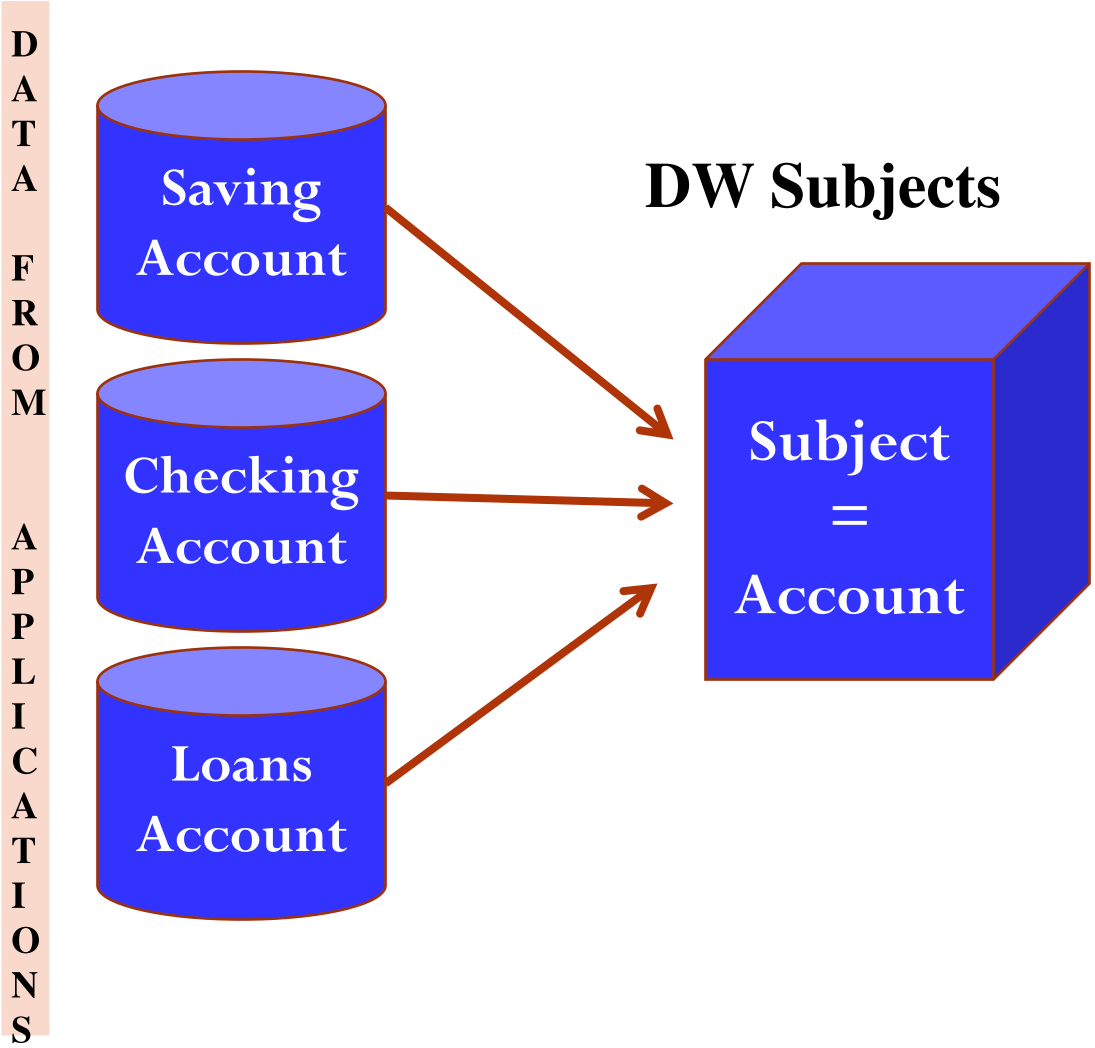
Policy

Data-Warehouse Subjects

Integrated Data

- For proper decision making, you need to pull together all the relevant data from the various applications.
- The data in the data warehouse comes from several operational systems.
- Source data reside in different databases, files, and data segments.
- These are disparate applications, so the operational platforms and operating systems could be different.
- The file layouts, character code representations, and field naming conventions all could be different.
- In addition to data from internal operational systems, for many enterprises, **data from outside sources** is likely to be very important.

Data inconsistencies are removed;
data from diverse operational applications is integrated.



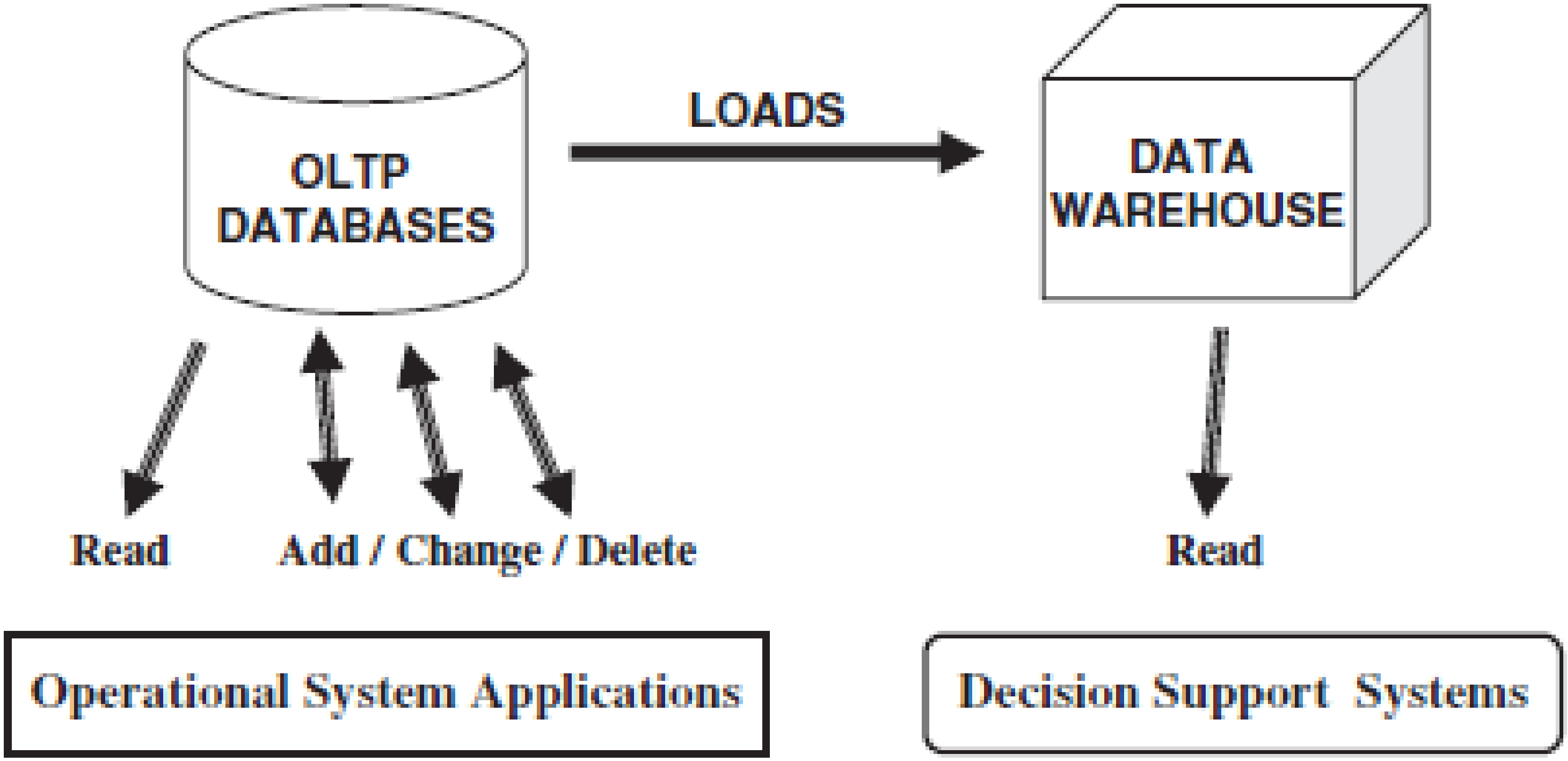
The data warehouse is integrated

Time-Variant Data

- For an operational system, the stored data contains the current values.
- On the other hand, the data in the data warehouse is meant for analysis and decision making.
- If a user is looking at the buying pattern of a specific customer, the user needs data not only about the current purchase, but on the past purchases as well.
- A data warehouse, because of the very nature of its purpose, has to contain historical data, not just current values.
- The time-variant nature of the data in a data warehouse
  - Allows for analysis of the past
  - Relates information to the present
  - Enables forecasts for the future

Nonvolatile Data

Usually the data in the data warehouse
is not updated or deleted.



OLTP DATABASES

LOADS

DATA WAREHOUSE

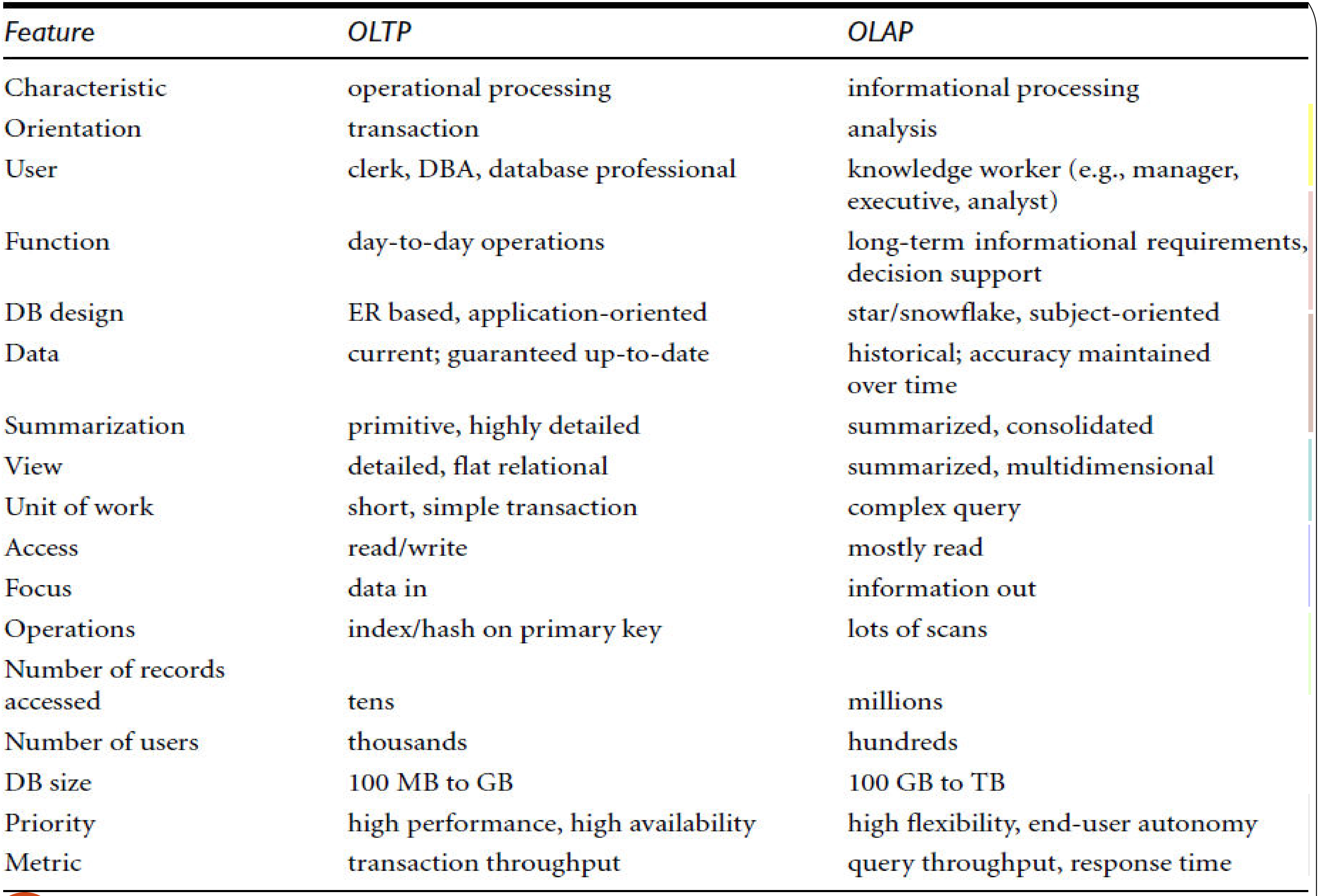
Read

Add / Change / Delete

Read

Operational System Applications

Decision Support 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	historical; 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	100 MB to GB	100 GB to TB
Priority	high performance, high availability	high flexibility, end-user autonomy
Metric	transaction throughput	query throughput, response time

DATA WAREHOUSES AND DATA MARTS

- Before deciding to build a data warehouse for your organization, you need to ask the following basic and fundamental questions and address the relevant issues:
  - Top-down or bottom-up approach?
  - Enterprise-wide or departmental?
  - Which first—data warehouse or data mart?
  - Build pilot or go with a full-fledged implementation?
  - Dependent or independent data marts?

DATA WAREHOUSES vs DATA MARTS

DATA WAREHOUSE	DATA MART
Corporate/Enterprise-wide	Departmental
Union of all data marts	A single business process
Data received from staging area	STAR join (facts & dimensions)
Structure for corporate view of data	Structure to suit the departmental view of data

Top-Down Versus Bottom-Up Approach

- **Top-Down Approach**
- The advantages of this approach are:
  - A truly corporate effort, an enterprise view of data
  - Inherently architected, not a union of disparate data marts
  - Single, central storage of data about the content
  - Centralized rules and control
  - May see quick results if implemented with iterations
- The disadvantages are:
  - Takes longer to build even with an iterative method
  - High exposure to risk of failure
  - Needs high level of cross-functional skills
  - High outlay without proof of concept

Top-Down Versus Bottom-Up Approach

- **Bottom-Up Approach**
- The advantages of this approach are:
  - Faster and easier implementation of manageable pieces
  - Favorable return on investment and proof of concept
  - Less risk of failure
  - Inherently incremental; can schedule important data marts first
  - Allows project team to learn and grow
- The disadvantages are:
  - Each data mart has its own narrow view of data
  - Permeates redundant data in every data mart
  - Perpetuates inconsistent and irreconcilable data
  - Proliferates unmanageable interfaces

Practical Approach

- Although the top-down and the bottom-up approaches each have their own advantages and drawbacks, a compromise approach accommodating both views appears to be practical.
- In this approach we do not lose sight of the overall big picture for the entire enterprise. We base our planning on this overall big picture. This aspect is from the top-down approach.
- Then we adopt the principles of the bottom-up approach and build the conformed data marts based on a priority scheme.
- The steps in this practical approach are as follows:

1. Plan and define requirements at the overall corporate level
2. Create a surrounding architecture for a complete warehouse
3. Conform and standardize the data content
4. Implement the data warehouse as a series of supermarts, one at a time

Architectural Types

- Centralized Data Warehouse
- Independent Data Marts
- Federated
- Hub-and-Spoke
- Data-Mart Bus

Architectural Types

- **Centralized Data Warehouse**
  - This architectural type takes into account the enterprise-level information requirements.
  - An overall infrastructure is established.
  - Atomic level normalized data at the lowest level of granularity is stored in the third normal form.
  - Occasionally, some summarized data is included.
  - Queries and applications access the normalized data in the central data warehouse.
  - There are no separate data marts.

Architectural Types

- **Independent Data Marts**
  - This architectural type evolves in companies where the organizational units develop their own data marts for their own specific purposes.
  - Although each data mart serves the particular organizational unit, these separate data marts do not provide "a single version of the truth."
  - The data marts are independent of one another. As a result, these different data marts are likely to have inconsistent data definitions and standards.
    - For example, if there are two independent data marts, one for sales and the other for shipments, although sales and shipments are related subjects, the independent data marts would make it difficult to analyze sales and shipments data together.

Architectural Types

- **Federated**
  - Some companies get into data warehousing with an existing legacy of an assortment of decision-support structures in the form of operational systems, extracted datasets, primitive data marts, and so on.
  - For such companies, it may not be prudent to discard all that huge investment and start from scratch.
  - The practical solution is a federated architectural type where data may be physically or logically integrated through shared key fields, overall global metadata, distributed queries, and such other methods.
  - In this architectural type, there is no one overall data warehouse.

Architectural Types

- **Hub-and-Spoke**
- This is the **Inmon** Corporate Information Factory approach. Similar to the centralized data warehouse architecture, that is an overall enterprise-wide data warehouse.
- Atomic data in the third normal form is stored in the centralized data warehouse.
- **The major and useful difference is the presence of dependent data marts in this architectural type.**
- Dependent data marts obtain data from the centralized data warehouse. The centralized data warehouse forms the hub to feed data to the data marts on the spokes.
- The dependent data marts may be developed for a variety of purposes: departmental analytical needs, specialized queries, data mining, and so on.
- Each dependent dart mart may have normalized, denormalized, summarized, or dimensional data structures based on individual requirements.
- Most queries are directed to the dependent data marts although the centralized data warehouse may itself be used for querying.
- This architectural type results from adopting a top-down approach to data warehouse development.

Architectural Types

- **Data-Mart Bus**
  - This is the **Kimbal** conformed supermarts approach.
  - Begin with analyzing requirements for a specific business subject such as orders, shipments, billings, insurance claims, car rentals, and so on.
  - Build the first data mart (supermart) using business dimensions and metrics.
  - These business dimensions will be shared in the future data marts.
  - The principal notion is that by conforming dimensions among the various data marts, the result would be logically integrated supermarts that will provide an enterprise view of the data.
  - The data marts contain atomic data organized as a dimensional data model.
  - This architectural type results from adopting an enhanced bottom-up approach to data warehouse development.

DW: Metadata

- Types of Metadata
- Metadata in a data warehouse fall into three major categories:
    - Operational metadata
    - Extraction and transformation metadata
    - End-user metadata

DW: Metadata

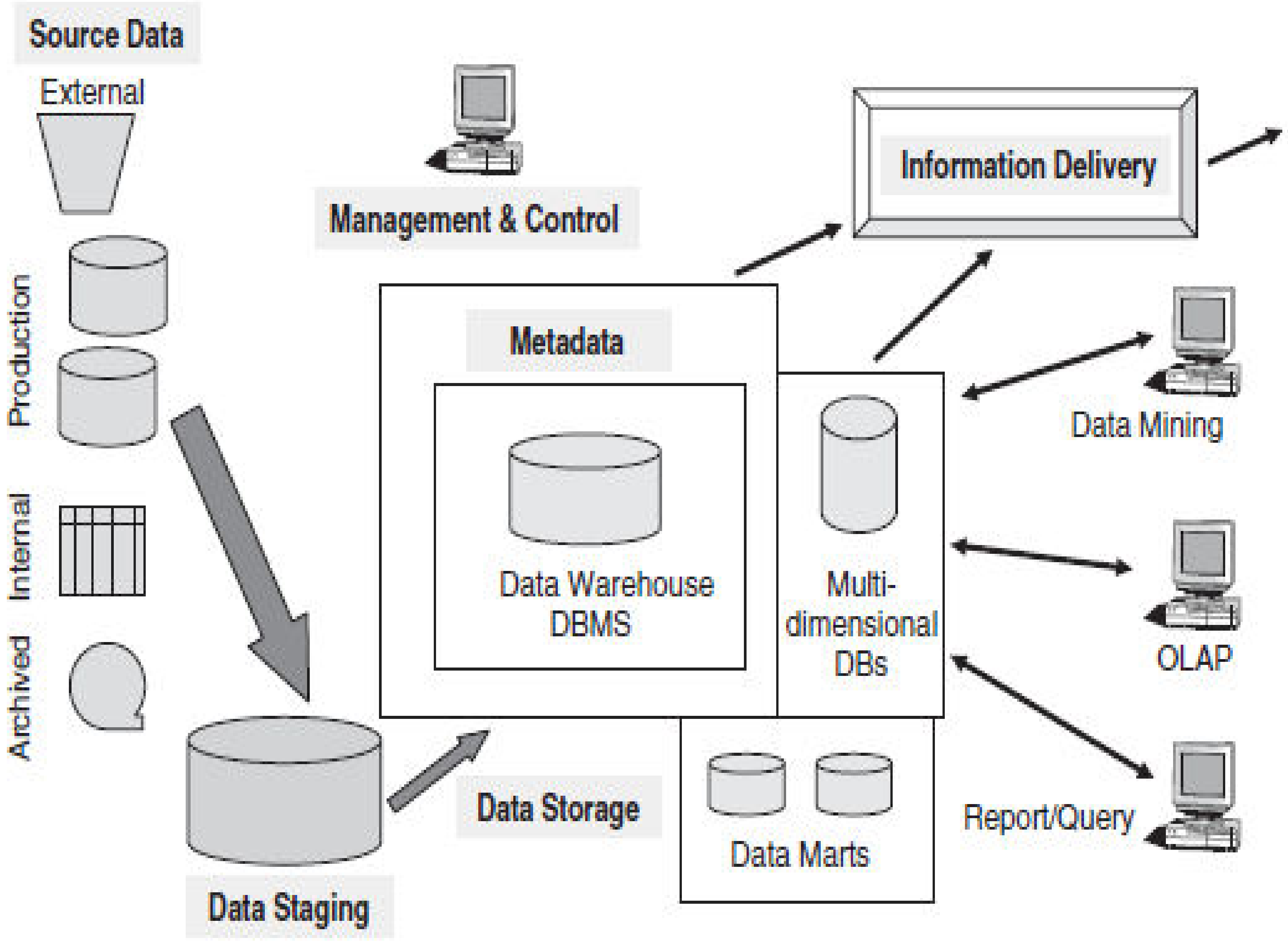
- **Operational Metadata**
- The data for the data warehouse comes from several operational systems of the enterprise. These source systems contain **different data structures**.
- The data elements selected for the data warehouse have various field **lengths and data types**.
- In selecting data from the source systems for the data warehouse, you **split** records, **combine** parts of records from different source files, and deal with multiple coding schemes and field lengths.
- When you deliver information to the end-users, you must be able to tie that back to the original source data sets.
- Operational metadata contain all of this information about the operational data sources.

DW: Metadata

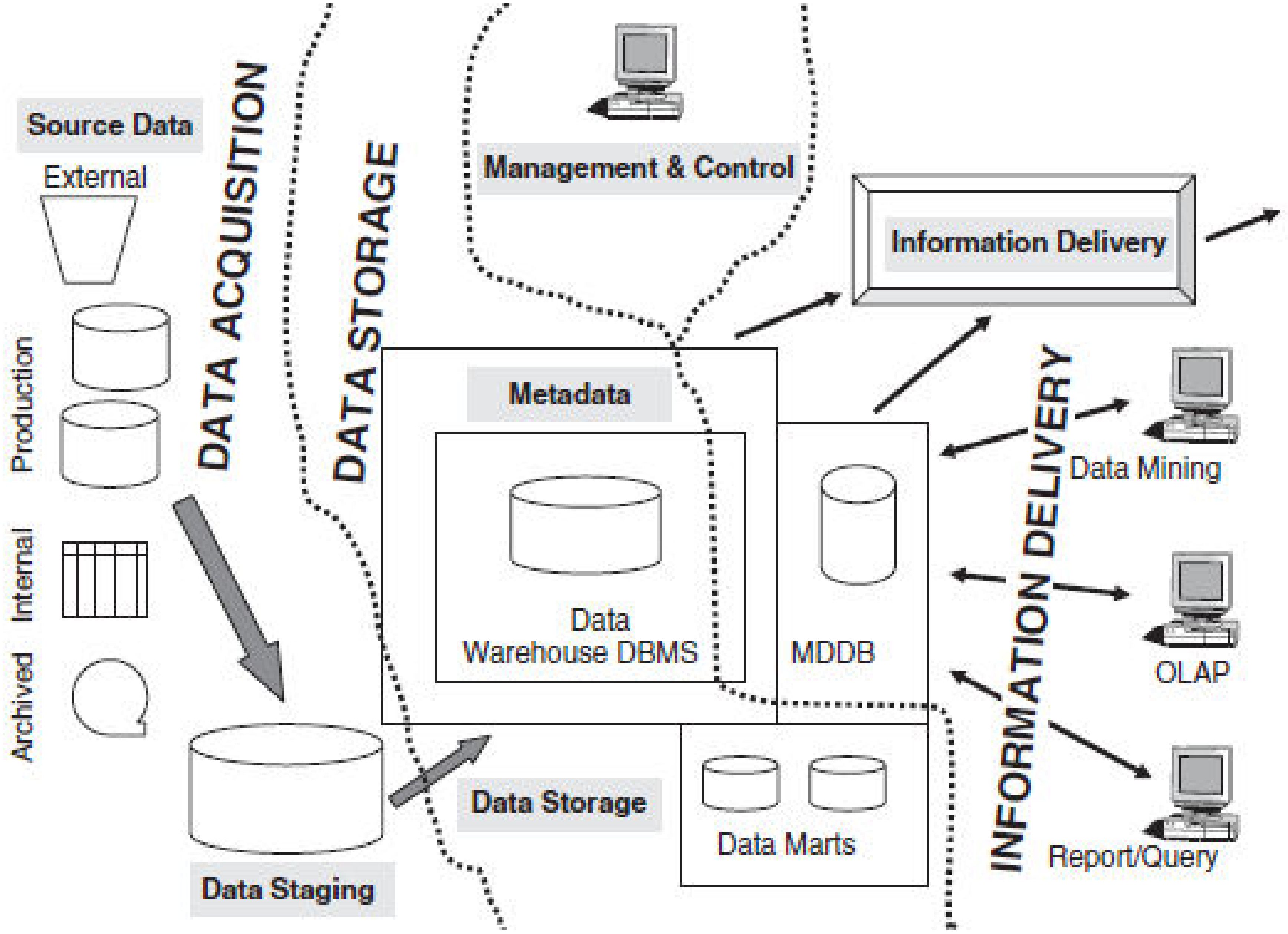
- **Extraction and Transformation Metadata**
  - Extraction and transformation metadata contain data about the extraction of data from the source systems, namely, the extraction frequencies, extraction methods, and business rules for the data extraction.
  - This category of metadata also contains information about all the data transformations that take place in the **data staging area**.
- **End-User Metadata**
  - The end-user metadata is the navigational map of the data warehouse.
  - It enables the end-users to find information from the data warehouse.
  - The end-user metadata allows the end-users to use their own business terminology and look for information in those ways in which they normally think of the business.

DW: Metadata

- **Special Significance**
- Why is metadata especially important in a data warehouse?
  - First, it acts as the glue that connects all parts of the data warehouse.
  - Next, it provides information about the contents and structures to the developers.
  - Finally, it opens the door to the end-users and makes the contents recognizable in their own terms.



Data warehouse: building blocks or components.



Architectural components in the three major areas.

DW: Architecture

- **Source Data Component**
- Source data coming into the data warehouse may be grouped into four broad categories
  - Production Data
  - Internal Data
  - Archived Data
  - External Data

DW: Architecture

- **Data Staging Component**
  - After you have extracted data from various operational systems and from external sources, you have to prepare the data for storing in the data warehouse.
  - The extracted data coming from several disparate sources needs to be changed, converted, and made ready in a format that is suitable to be stored for querying and analysis.
- Why do you need a separate place or component to perform the data preparation?
  - The need is:
  - In a data warehouse you pull in data from many source operational systems.
  - Remember that data in a data warehouse is **subject-oriented and cuts across operational applications.** A separate staging area, therefore, is a necessity for preparing data for the data warehouse.

DW: Architecture

- **Data Extraction**
- This function has to deal with numerous data sources.
- You have to employ the appropriate technique for each data source.
- Source data may be from different source machines in diverse data formats.
  - Part of the source data may be in relational database systems.
  - Some data may be on other legacy network and hierarchical data models.
  - Many data sources may still be in flat files.
  - You may want to include data from spreadsheets and local departmental data sets.
- Data extraction may become quite complex.

DW: Architecture

- **Data Transformation**
- Data for a data warehouse comes from many disparate sources.
- A number of individual tasks as part of data transformation.
  - First, you clean the data extracted from each source.
  - Cleaning may be
    - Just be correction of misspellings, or
    - May include resolution of conflicts between state codes and zip codes in the source data, or
    - May deal with providing default values for missing data elements, or
    - Elimination of duplicates when you bring in the same data from multiple source systems.
- Standardization of data elements forms a large part of data transformation.
  - Standardize the data types and field lengths for same data elements retrieved from the various sources.

DW: Architecture

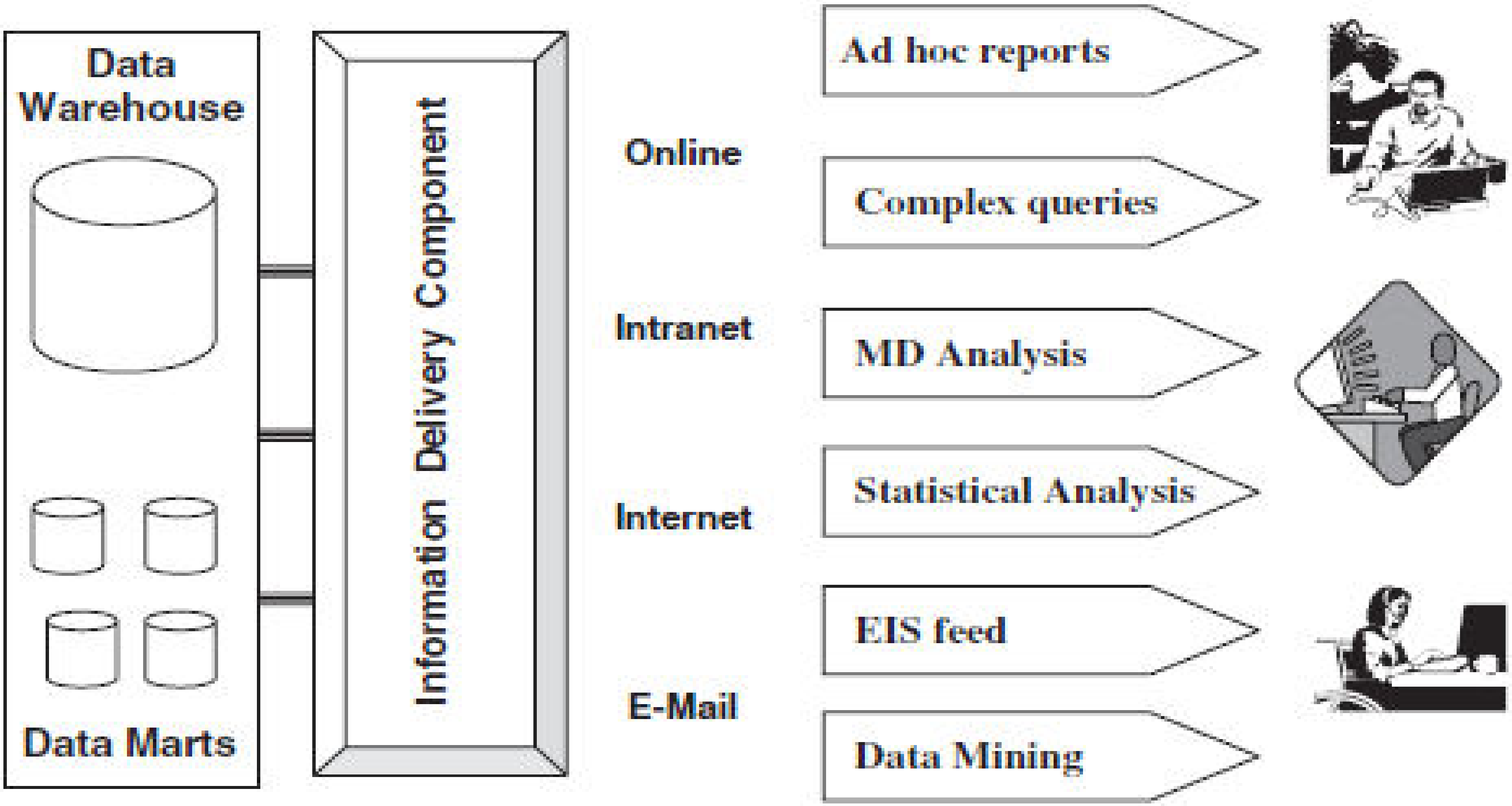
- **Data Loading**
- Two distinct groups of tasks form the data loading function.
  - When you complete the design and construction of the data warehouse and go live for the first time, you do the initial loading of the data into the data warehouse storage.
    - The initial load moves large volumes of data using up substantial amounts of time.
- As the data warehouse starts functioning, you continue to extract the changes to the source data, transform the data revisions, and feed the **incremental data revisions** on an ongoing basis.

DW: Architecture

- **Data Storage Component**
  - The data storage for the data warehouse is a separate repository.
  - The data repository for a data warehouse, need to keep large volumes of historical data for analysis.
  - Further, to keep the data in the data warehouse in structures suitable for analysis, and not for quick retrieval of individual pieces of information.
  - Therefore, the data storage for the data warehouse is kept separate from the data storage for operational systems.

DW: Architecture

- Information Delivery Component



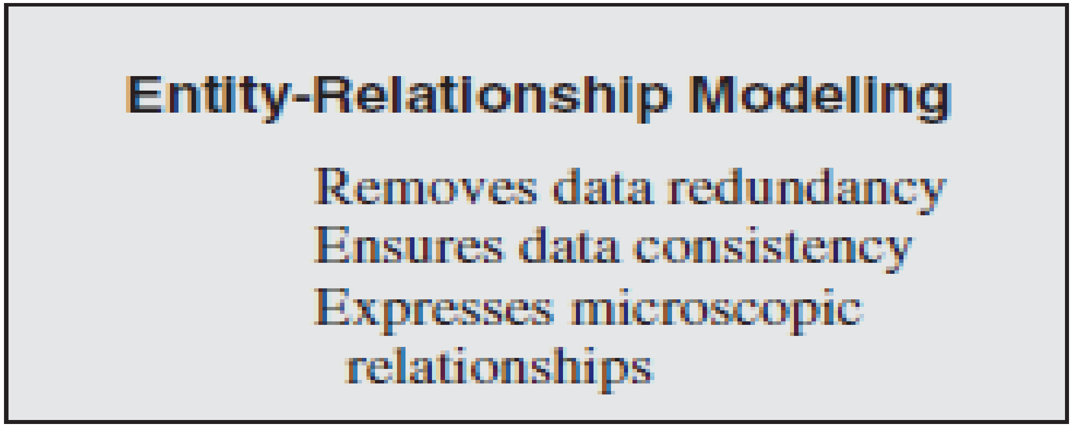
Executive Information Systems (EIS) is meant for senior executives

Dimensional Model

- **Cont…**

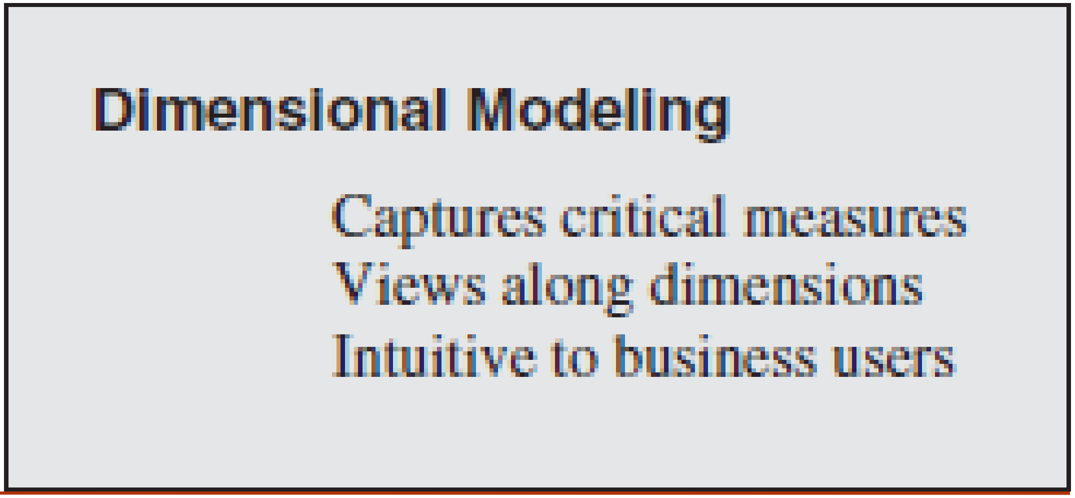
- Depict the way in which the **fac**
- Allow equal interaction of every
- Enable the users to perform **dimension hierarchies.**

- In DW we can form the dimensi
  - **Star Schema**
  - **Snowflake Schema**
  - **Fact constellation Schema**



Entity-Relationship Modeling

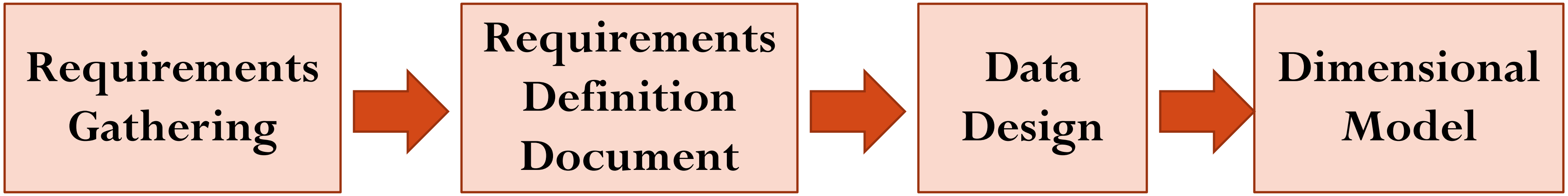
Removes data redundancy
Ensures data consistency
Expresses microscopic relationships



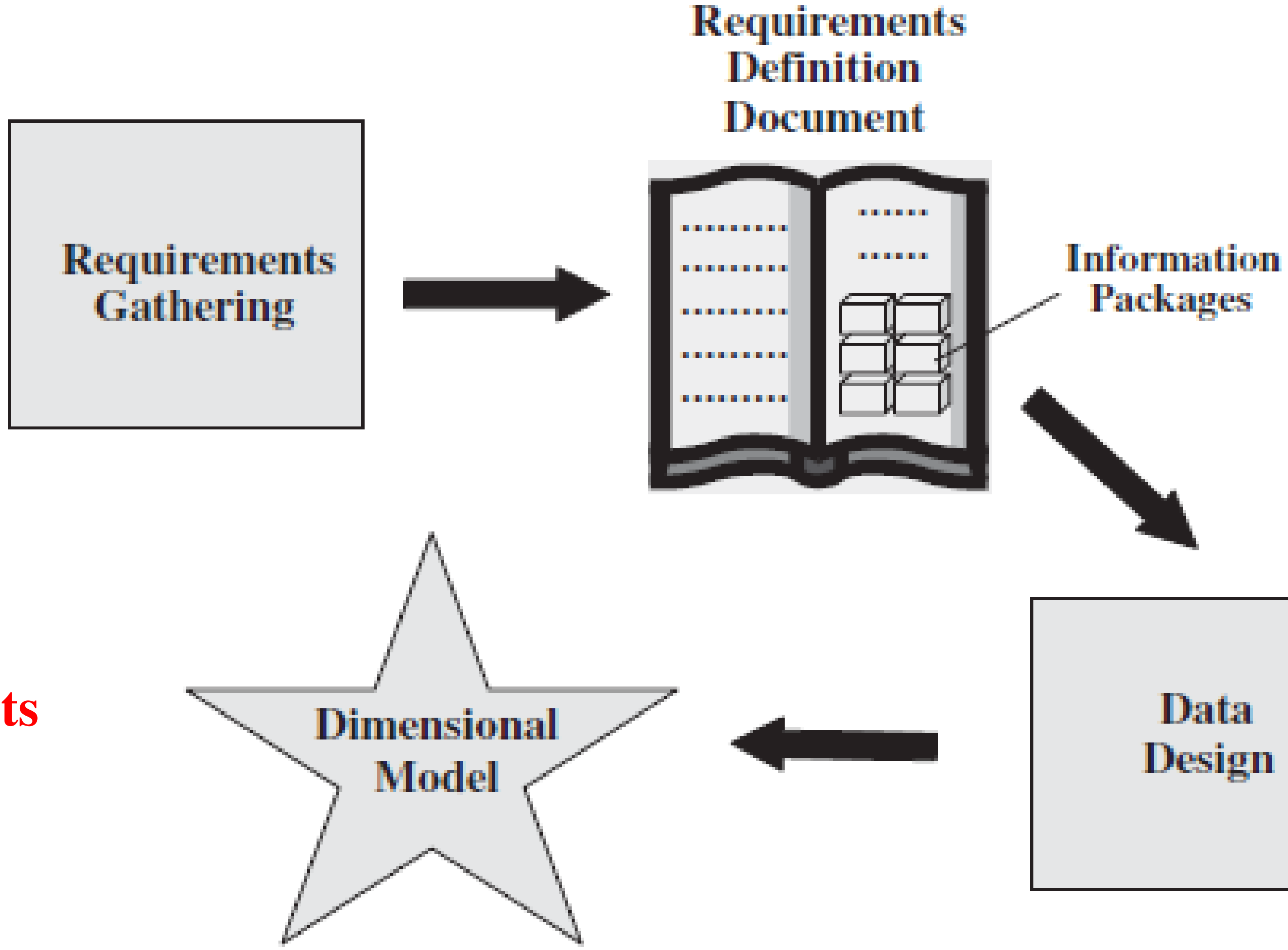
Dimensional Modeling

Captures critical measures
Views along dimensions
Intuitive to business users

Source: Reema Thareja



Requirements Gathering → Requirements Definition Document → Data Design → Dimensional Model



From requirements to data design.

Source: PAULRAJ PONNIAH

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

Source: Han & Kamber (2006)

STAR Schema

- The results of the requirements gathering phase is documented in detail in the requirements definition document.
- An essential component of this document is the set of information package diagrams.
- The information package diagrams - data marts.

STAR Schema

- Design Decisions
- Before designing the dimensional data model, the design decisions we have to make:
  - **Choosing the Process.**
    - Selecting the subjects from the information packages for the first set of logical structures to be designed.
  - **Choosing the Grain.**
    - Determining the level of detail for the data in the data structures.
  - **Identifying and Conforming the Dimensions.**
    - Choosing the business dimensions (such as product, market, time, etc.) to be included in the first set of structures and making sure that each particular data element in every business dimension is conformed to one another.

  - **Cont…**

STAR Schema

- **Choosing the Facts.**
  - Selecting the metrics or units of measurements (such as product sale units, dollar sales, dollar revenue, etc.) to be included in the first set of structures.
- **Choosing the Duration of the Database.**
  - Determining how far back in time you should go for historical data.

STAR Schema

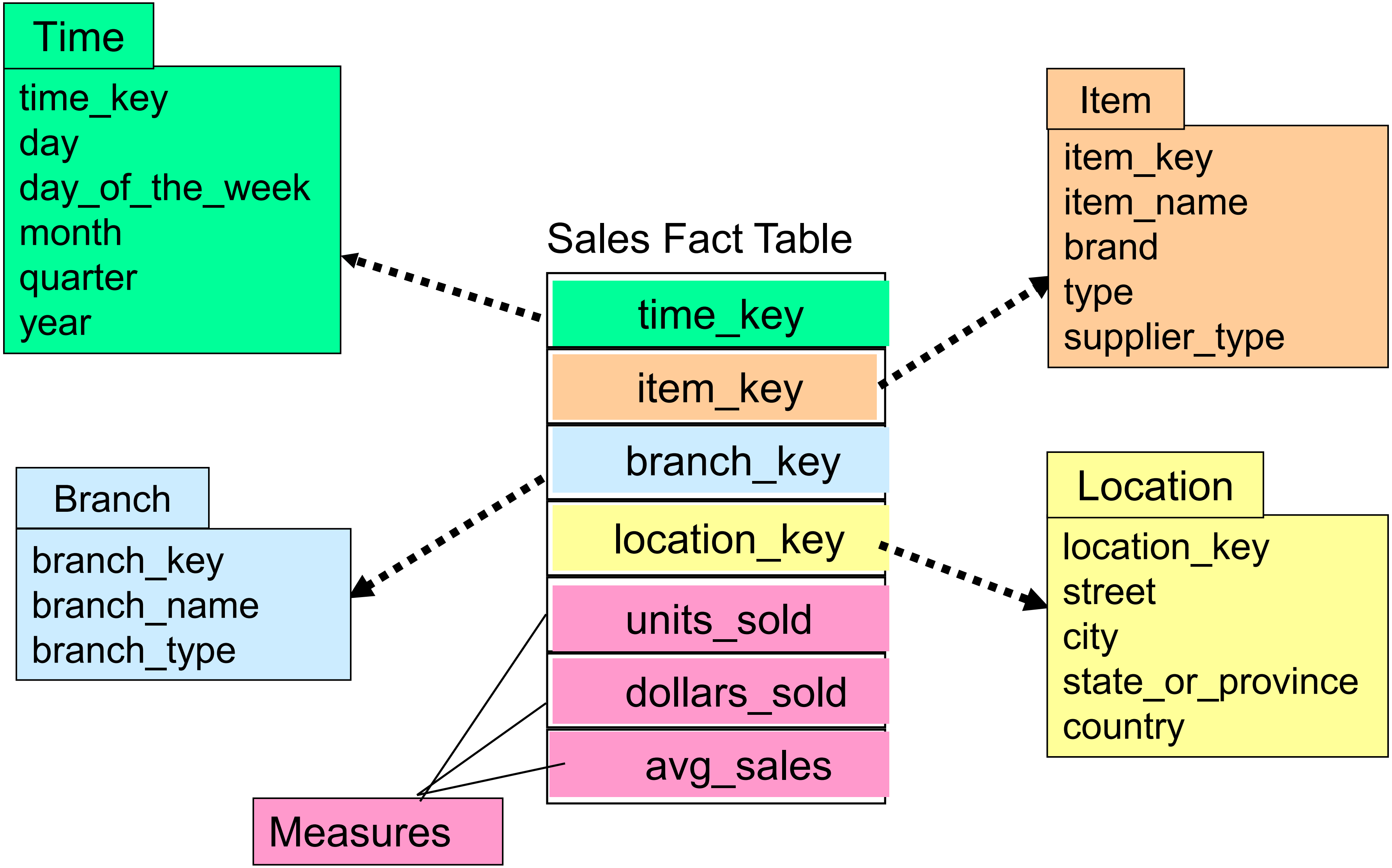
- A dimensional model with the fact table in the middle and dimension tables arranged around the fact table.
- This model represents star formation with the **fact table at the core** and the dimension tables along the spikes of the STAR.
- This arrangement is thus called a STAR schema.
- In STAR schema, every dimension table has a direct relationship with the fact table in the middle thereby allowing every dimension table with its attributes to have an equal chance of participating in a query to analyze the attribute in the fact table.
- STAR schema is perhaps the simplest logical schema of DW.
- The fact table contains primary information in the DW.
- Cont…

STAR Schema

- Dimension tables contains information about the entries for a particular attribute in the fact table.
- Each dimension table is joined with fact table using PK-FK join. But no join for dimension tables.

- **How does a query Execute?**
- When a query is executed against the STAR schema, the results of the query are produced by combining or joining one or more dimension tables with the fact table.

Example of Star Schema



Time

time_key
day
day_of_the_week
month
quarter
year

Sales Fact Table

time_key
item_key
branch_key
location_key
units_sold
dollars_sold
avg_sales

Item

item_key
item_name
brand
type
supplier_type

Branch

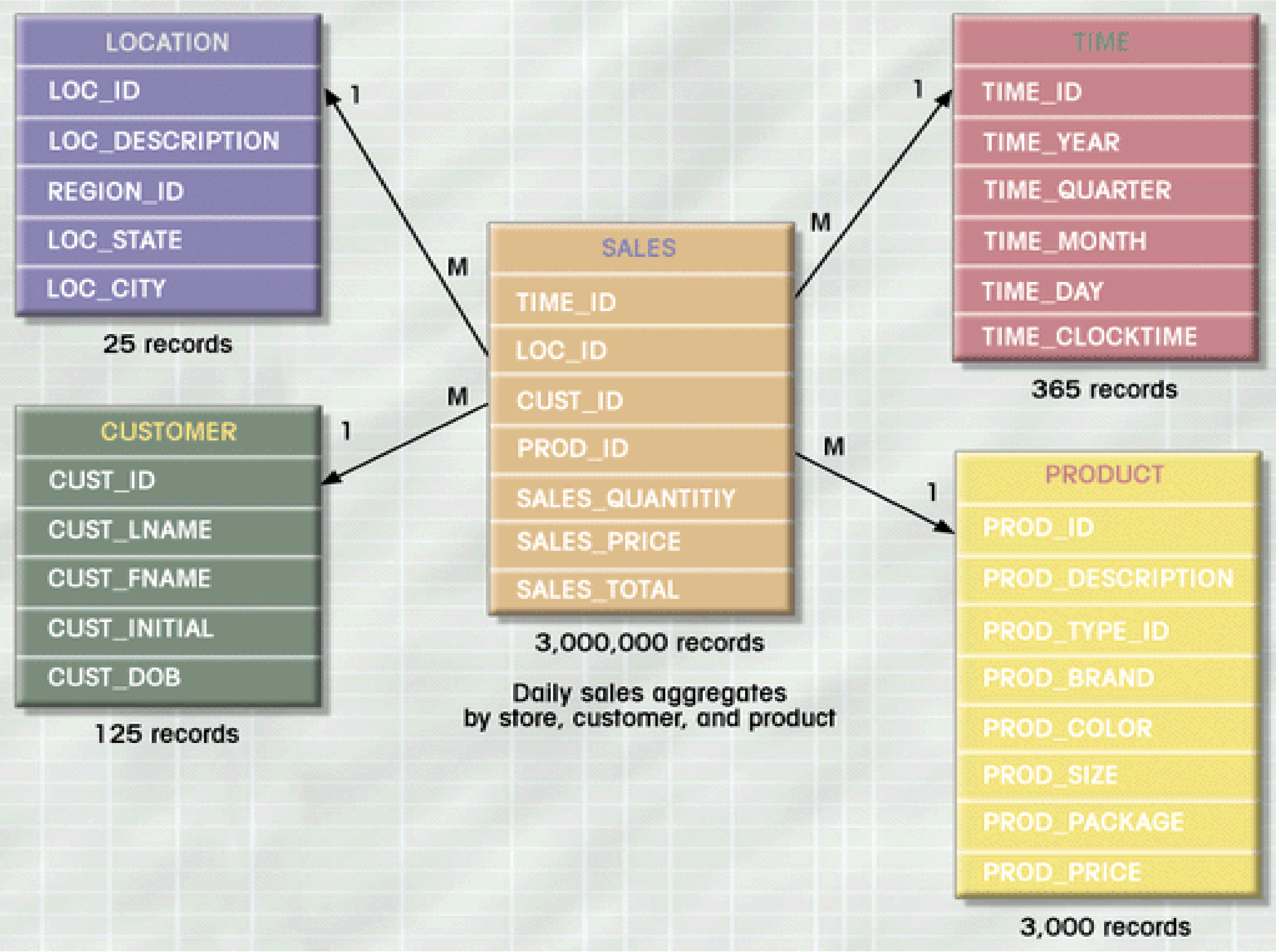
branch_key
branch_name
branch_type

Location

location_key
street
city
state_or_province
country

Measures

Source: Han & Kamber (2006)



LOCATION

LOC_ID
LOC_DESCRIPTION
REGION_ID
LOC_STATE
LOC_CITY

25 records

TIME

TIME_ID
TIME_YEAR
TIME_QUARTER
TIME_MONTH
TIME_DAY
TIME_CLOCKTIME

365 records

SALES

TIME_ID
LOC_ID
CUST_ID
PROD_ID
SALES_QUANTITIY
SALES_PRICE
SALES_TOTAL

3,000,000 records

Daily sales aggregates
by store, customer, and product

CUSTOMER

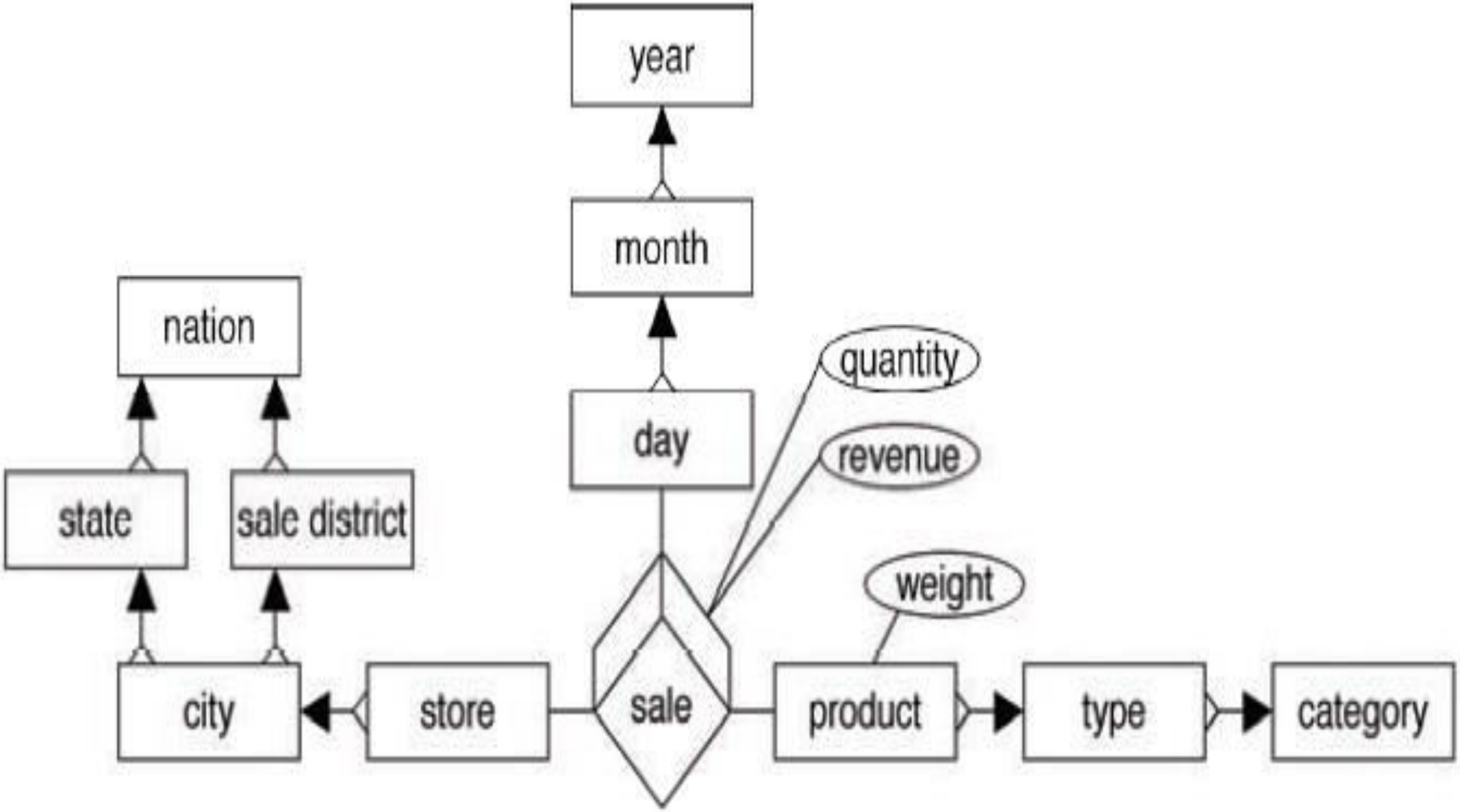
CUST_ID
CUST_LNAME
CUST_FNAME
CUST_INITIAL
CUST_DOB

125 records

PRODUCT

PROD_ID
PROD_DESCRIPTION
PROD_TYPE_ID
PROD_BRAND
PROD_COLOR
PROD_SIZE
PROD_PACKAGE
PROD_PRICE

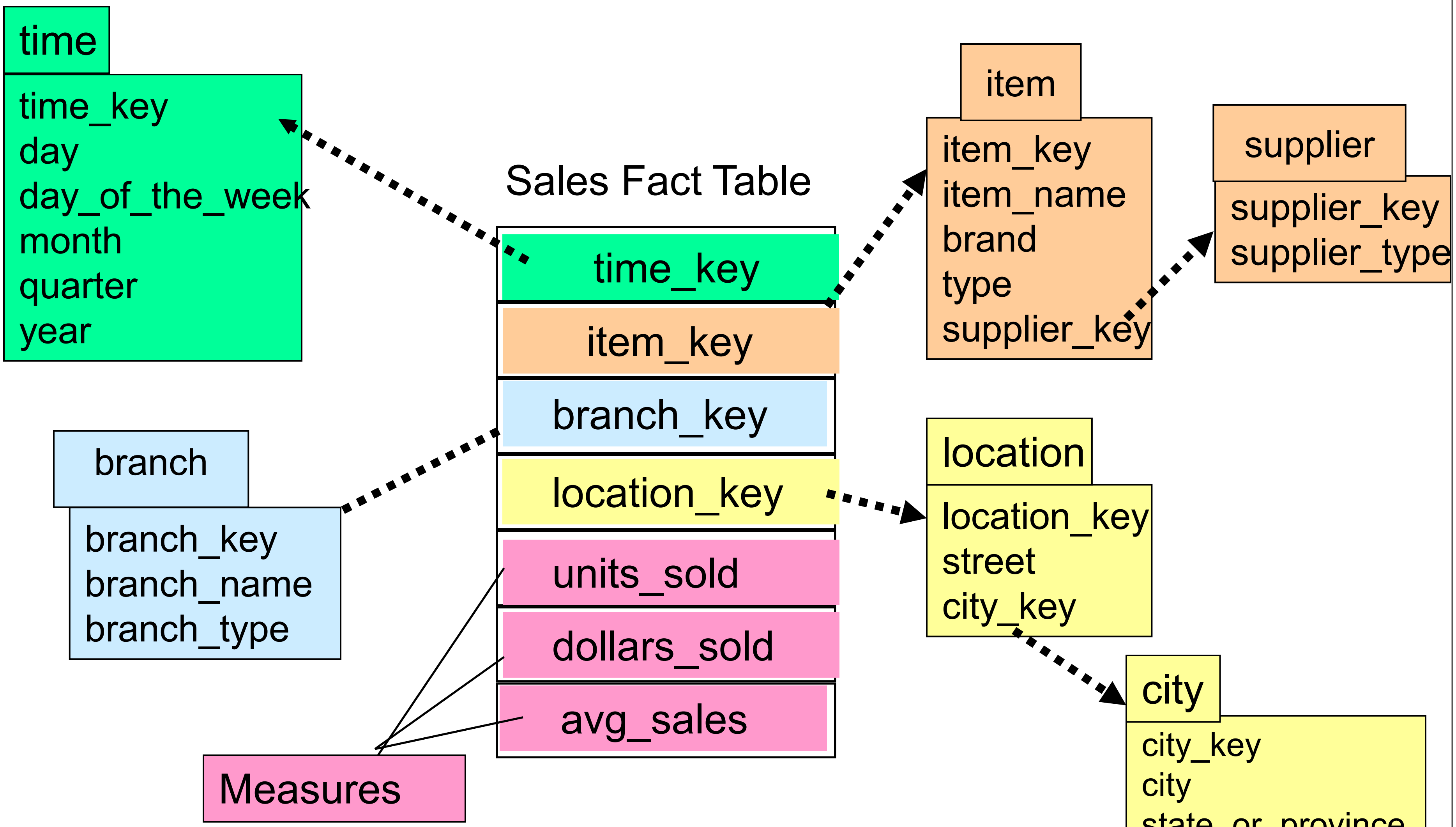
3,000 records


year
month
nation
quantity
day
revenue
state
sale district
weight
city
store
sale
product
type
category

SnowFlake Schema

- Variant of star schema model.
- A single,large and central fact table and one or more tables for each dimension.
- Dimension tables are normalized i.e. split dimension table data into additional tables

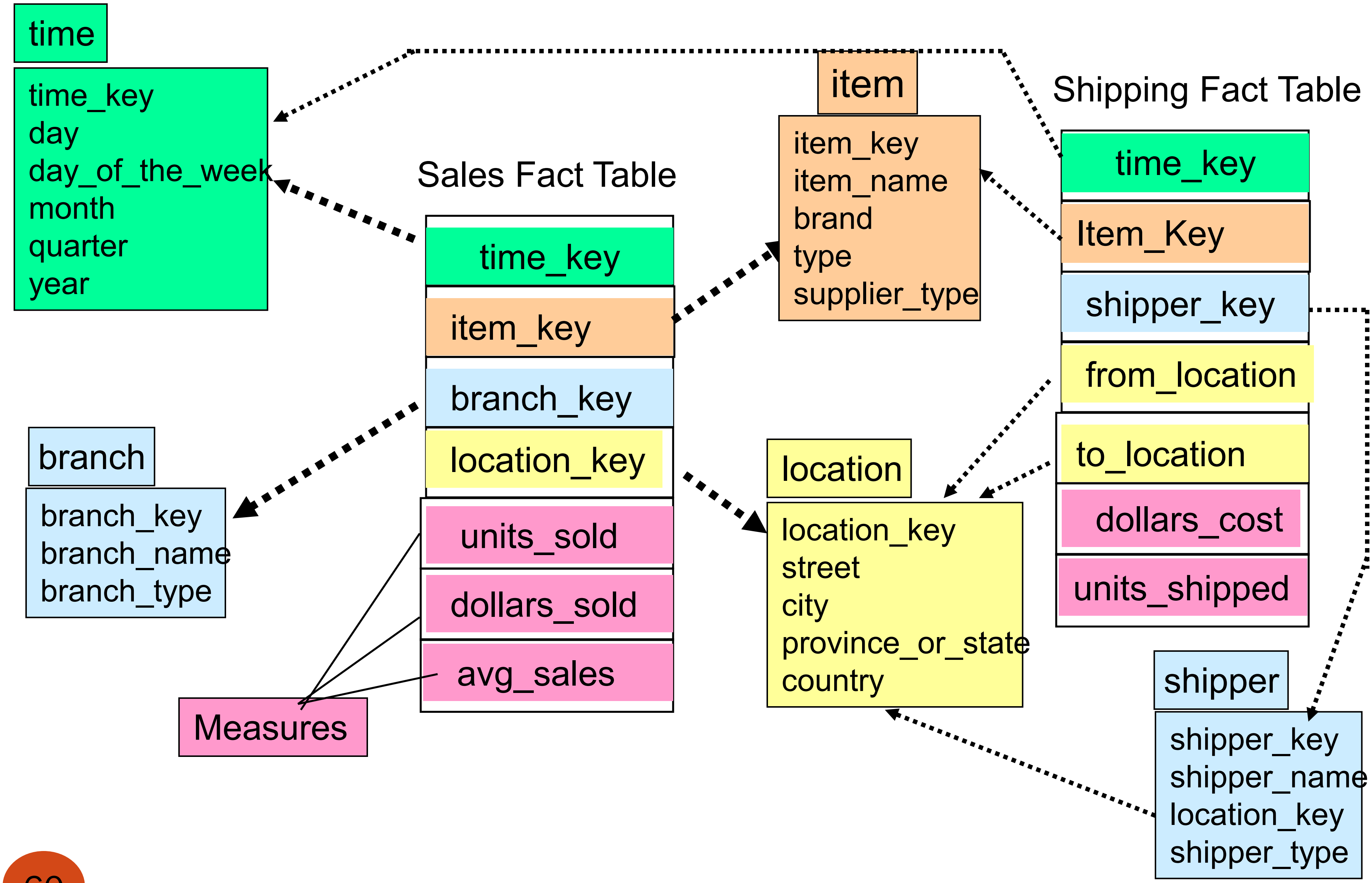
Example of Snowflake Schema



Fact Constellation

- Multiple fact tables share dimension tables.
- This schema is viewed as collection of stars hence called **galaxy schema** or fact constellation.
- Sophisticated application requires such schema.

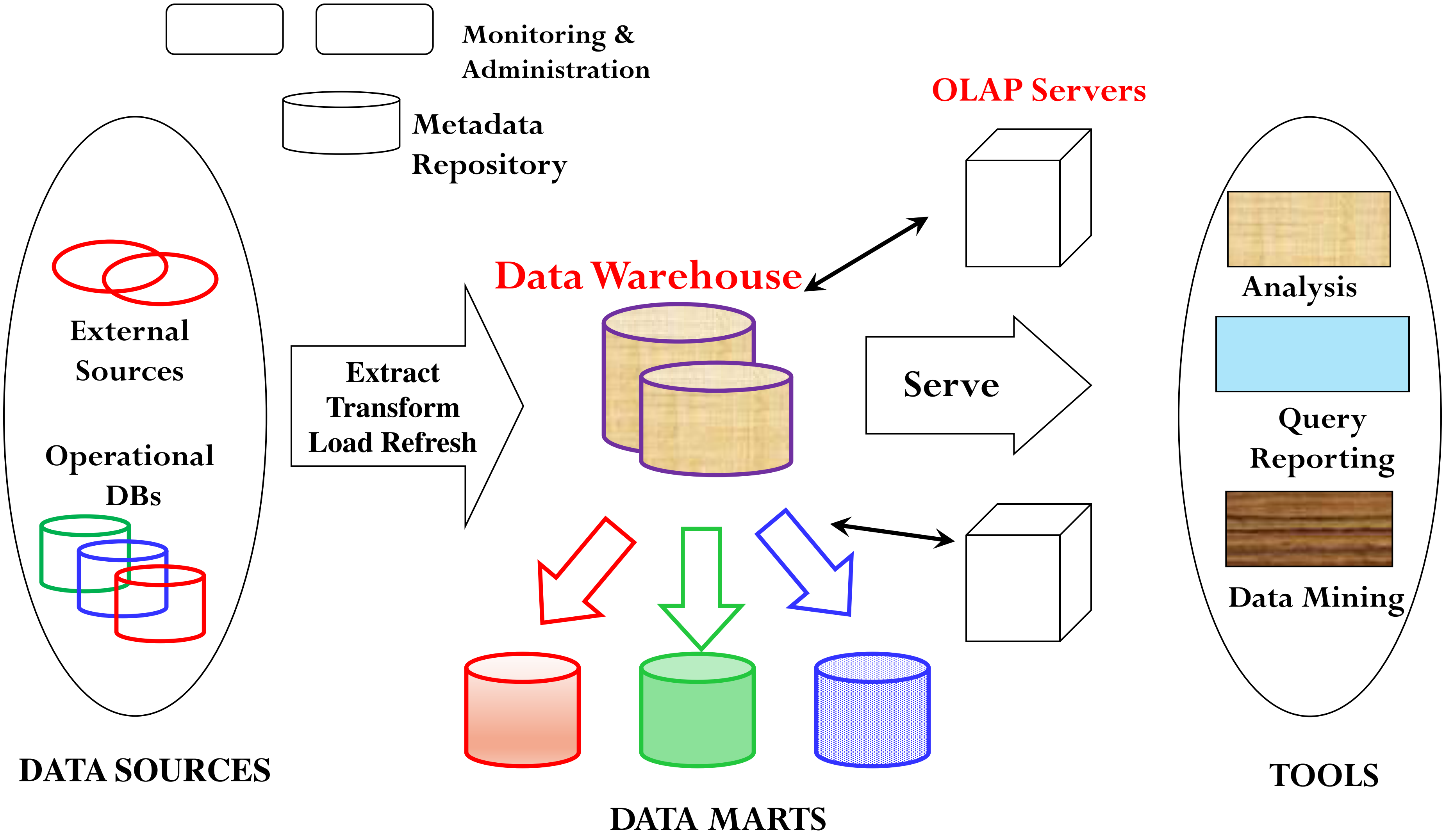
Example of Fact Constellation



RECAP

- Bill Inmon Definition
- Data Marts
- Top-Down Approach
- Bottom-Up Approach
- Practical Approach
- Architectural Types
- Metadata Types
- Modeling data warehouses: Dimensions & Measures
  - Star_schema
  - Snowflake schema
  - Fact constellations

Data Warehousing Architecture



Building Data Warehouse

- Data Selection
- Data Preprocessing
  - Fill missing values
  - Remove inconsistency
- Data Transformation & Integration
- Data Loading

Data in warehouse is stored in form of fact tables and dimension tables.

Case Study

- Afco Foods & Beverages is a new company which produces dairy, bread and meat products with production unit located at Baroda.
- There products are sold in North, North West and Western region of India.
- They have sales units at Mumbai, Pune, Ahemdabad, Delhi and Baroda.
- The President of the company wants sales information.

Sales Information

Report: The number of units sold.

113

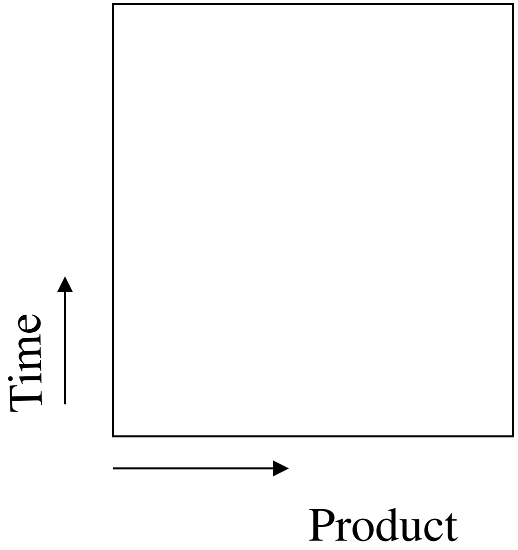
Report: The number of units sold over time

January	February	March	April
14	41	33	25

Sales Information

Report : The number of items sold for each product with time

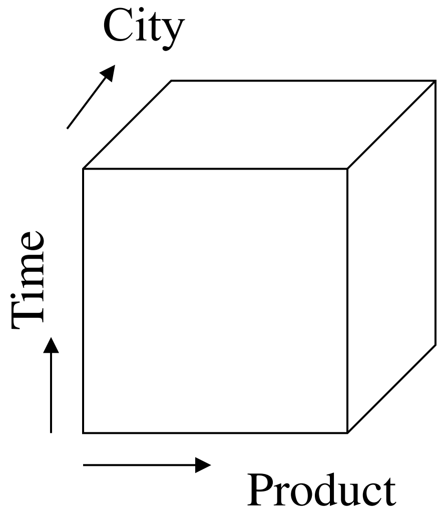
	Jan	Feb	Mar	Apr
Wheat Bread			6	17
Cheese	6	16	6	8
Swiss Rolls	8	25	21	



Sales Information

Report: The number of items sold in each City for each product with time

		Jan	Feb	Mar	Apr
Mumbai	Wheat Bread			3	10
	Cheese	3	16	6	
	Swiss Rolls	4	16	6	
Pune	Wheat Bread			3	7
	Cheese	3			8
	Swiss Rolls	4	9	15	



Sales Information

Report: The number of items sold and income in each region for each product with time.

		Jan		Feb		Mar		Apr	
		Rs	U	Rs	U	Rs	U	Rs	U
Mumbai	Wheat Bread					7.44	3	24.80	10
	Cheese	7.95	3	42.40	16	15.90	6		
	Swiss Rolls	7.32	4	29.98	16	10.98	6		
Pune	Wheat Bread					7.44	3	17.36	7
	Cheese	7.95	3					21.20	8
	Swiss Rolls	7.32	4	16.47	9	27.45	15		

Sales Measures & Dimensions

- Measure – Units sold, Amount.
- Dimensions – Product, Time, Region.

Sales Data Warehouse Model

Fact Table

City	Product	Month	Units	Rupees
Mumbai	Wheat Bread	January	3	7.95
Mumbai	Cheese	January	4	7.32
Pune	Wheat Bread	January	3	7.95
Pune	Cheese	January	4	7.32
Mumbai	Swiss Rolls	February	16	42.40

Sales Data Warehouse Model

City_ID	Prod_ID	Month	Units	Rupees
1	589	1/1/1998	3	7.95
1	1218	1/1/1998	4	7.32
2	589	1/1/1998	3	7.95
2	1218	1/1/1998	4	7.32
1	589	2/1/1998	16	42.40

Sales Data Warehouse Model

Product Dimension Tables

Prod_ID	Product_Name	Product_Category_ID
589	Wheat Bread	1
590	White Bread	1
288	Coconut Cookies	2

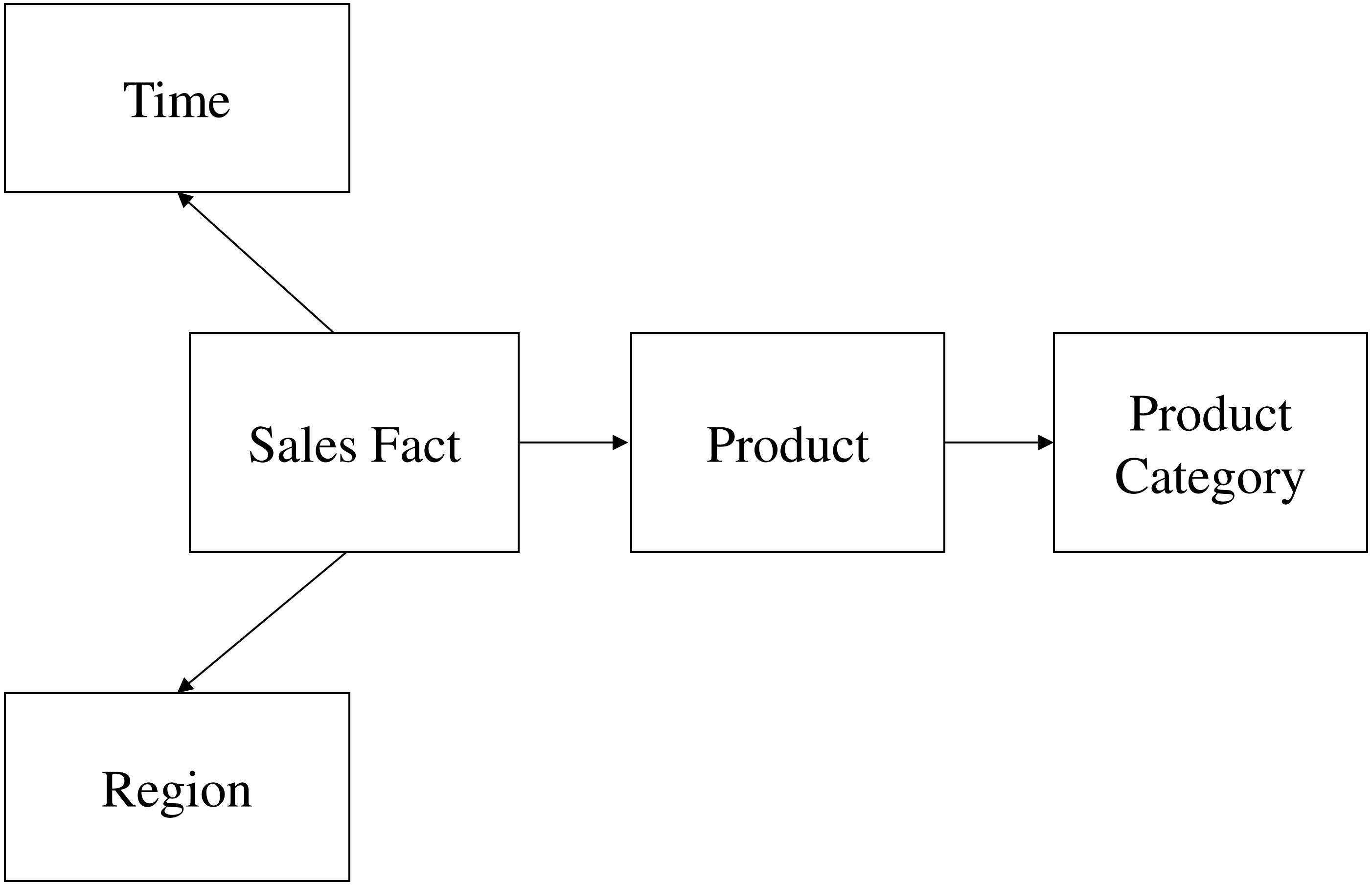
Product_Category_ID	Product_Category
1	Bread
2	Cookies

Sales Data Warehouse Model

Region Dimension Table

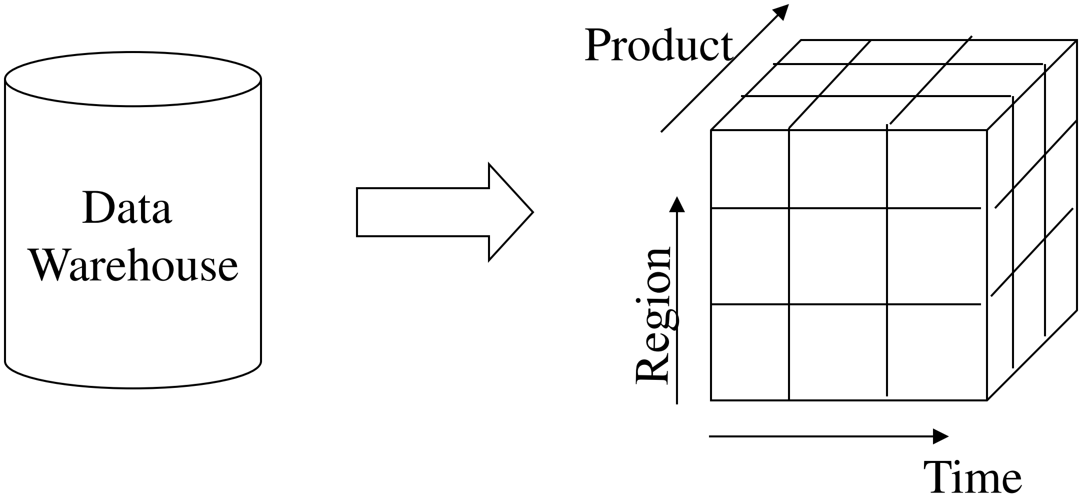
City_ID	City	Region	Country
1	Mumbai	West	India
2	Pune	NorthWest	India

Sales Data Warehouse Model



Online Analysis Processing(OLAP)

- It enables analysts, managers and executives to gain insight into data through fast, consistent, interactive access to a wide variety of possible views of information that has been transformed from raw data to reflect the real dimensionality of the enterprise as understood by the user.

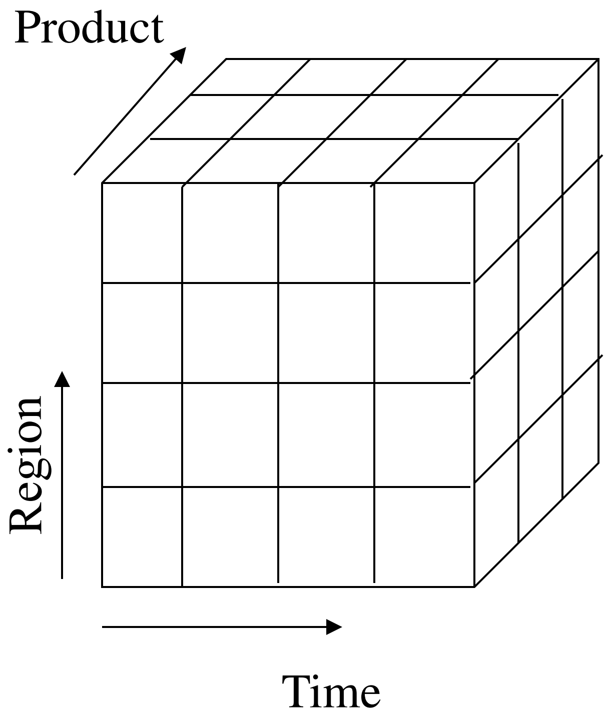


OLAP Cube

City	Product	Time	Units	Dollars
All	All	All	113	251.26
Mumbai	All	All	64	146.07
Mumbai	White Bread	All	38	98.49
Mumbai	Wheat Bread	All	13	32.24
Mumbai	Wheat Bread	Qtr1	3	7.44
Mumbai	Wheat Bread	March	3	7.44

OLAP Operations

Drill Down



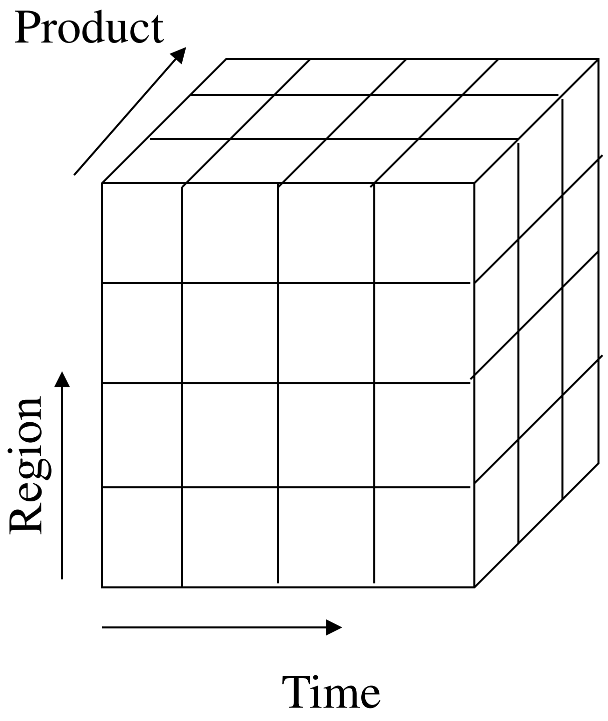
Category e.g Electrical Appliance

Sub Category e.g Kitchen

Product e.g Toaster

OLAP Operations

Drill Up



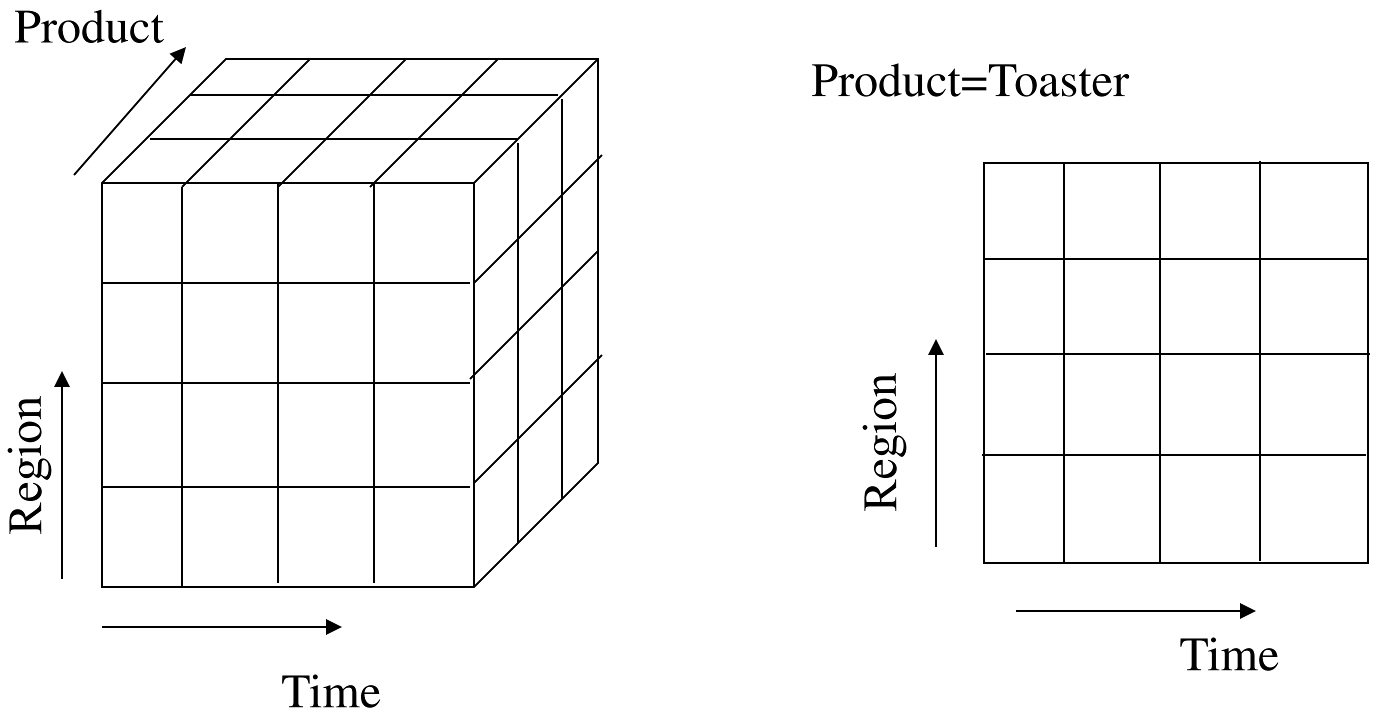
Category e.g Electrical Appliance

Sub Category e.g Kitchen

Product e.g Toaster

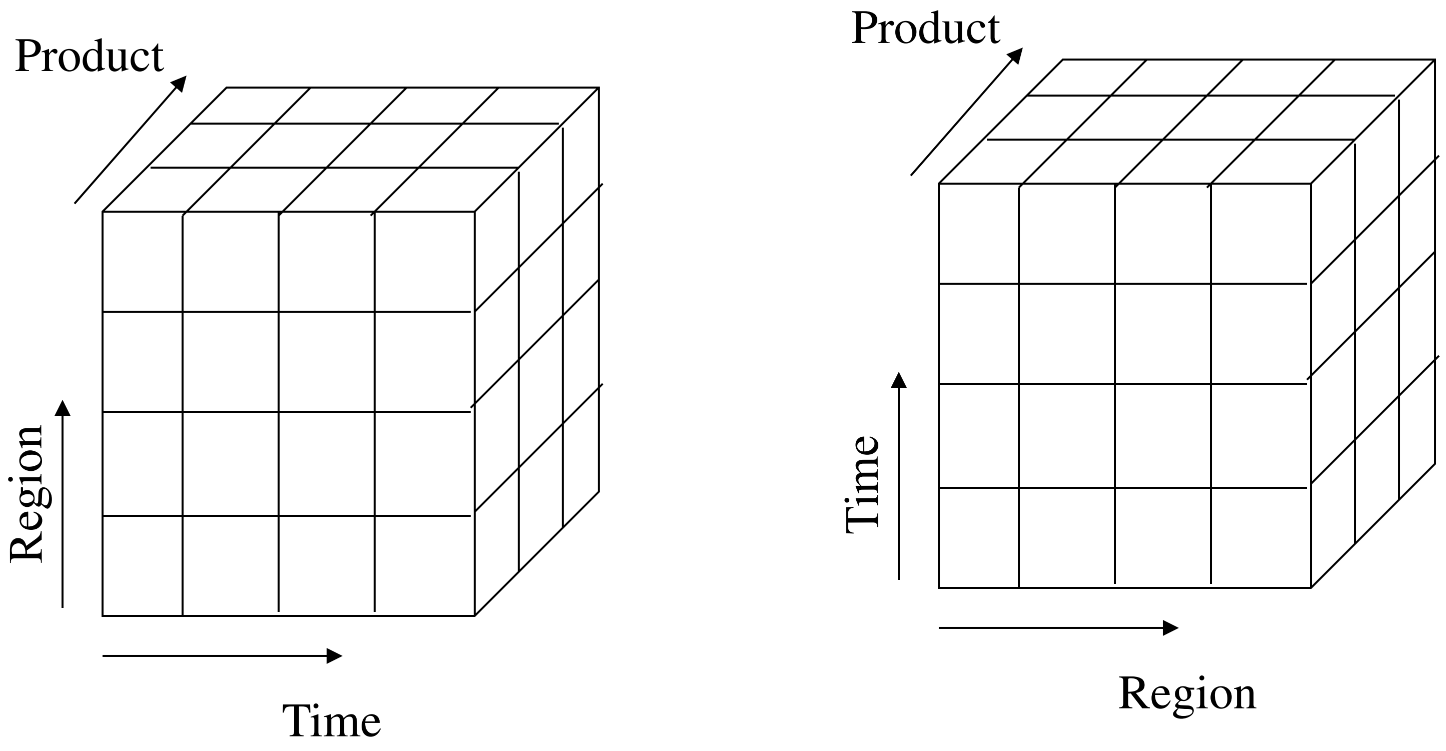
OLAP Operations

Slice and Dice



OLAP Operations

Pivot



AGGREGATES

- Add up amounts for day 1
- In SQL: SELECT sum(amt) FROM SALE WHERE date = 1

sale	prodId	storeId	date	amt
	p1	s1	1	12
	p2	s1	1	11
	p1	s3	1	50
	p2	s2	1	8
	p1	s1	2	44
	p1	s2	2	4

→ **81**

Aggregates

- Add up amounts by day
- In SQL: SELECT date, sum(amt) FROM SALE
  GROUP BY date

sale	prodId	storeId	date	amt
	p1	s1	1	12
	p2	s1	1	11
	p1	s3	1	50
	p2	s2	1	8
	p1	s1	2	44
	p1	s2	2	4

→

ans	date	sum
	1	81
	2	48

AGGREGATES: ANOTHER EXAMPLE

- Add up amounts by day, product
- In SQL: SELECT date, sum(amt) FROM SALE
  GROUP BY date, prodId

sale	prodId	storeId	date	amt
	p1	s1	1	12
	p2	s1	1	11
	p1	s3	1	50
	p2	s2	1	8
	p1	s1	2	44
	p1	s2	2	4

sale	prodId	date	amt
	p1	1	62
	p2	1	19
	p1	2	48

—— rollup ——→

←—— drill-down ——

POINTS TO BE NOTICED ABOUT ROLAP

- Defines complex, multi-dimensional data with simple model
- Reduces the number of joins a query has to process
- Allows the data warehouse to evolve with rel. low maintenance
- Can contain both detailed and summarized data.
- ROLAP is based on familiar, proven, and already selected technologies.
- BUT!!!
- SQL for multi-dimensional manipulation of calculations.

MOLAP: Dimensional Modeling Using the Multi Dimensional Model

- MDDB: a special-purpose data model
- Facts stored in multi-dimensional arrays
- Dimensions used to index array
- Sometimes on top of relational DB
- Products
  - Pilot, Arbor Essbase, Gentia

The MOLAP Cube

Fact table view:

sale	prodId	storeId	amt
	p1	s1	12
	p2	s1	11
	p1	s3	50
	p2	s2	8

↔

Multi-dimensional cube:

	s1	s2	s3
p1	12		50
p2	11	8	

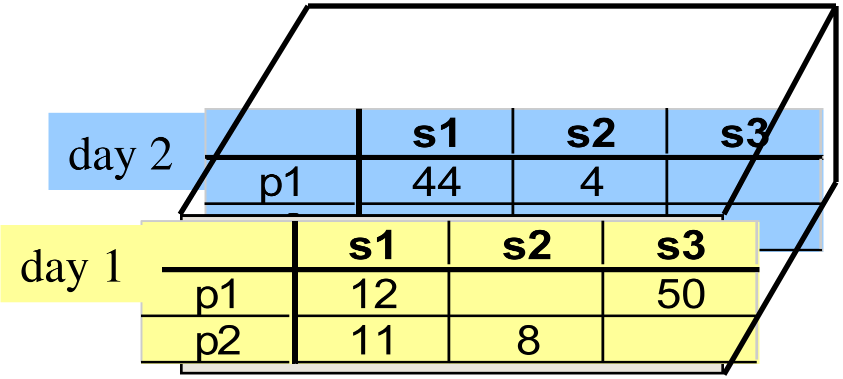
Dimensions = 2

3-D Cube

Fact table view:

sale	prodId	storeId	date	amt
	p1	s1	1	12
	p2	s1	1	11
	p1	s3	1	50
	p2	s2	1	8
	p1	s1	2	44
	p1	s2	2	4

Multi-dimensional cube:



day 2

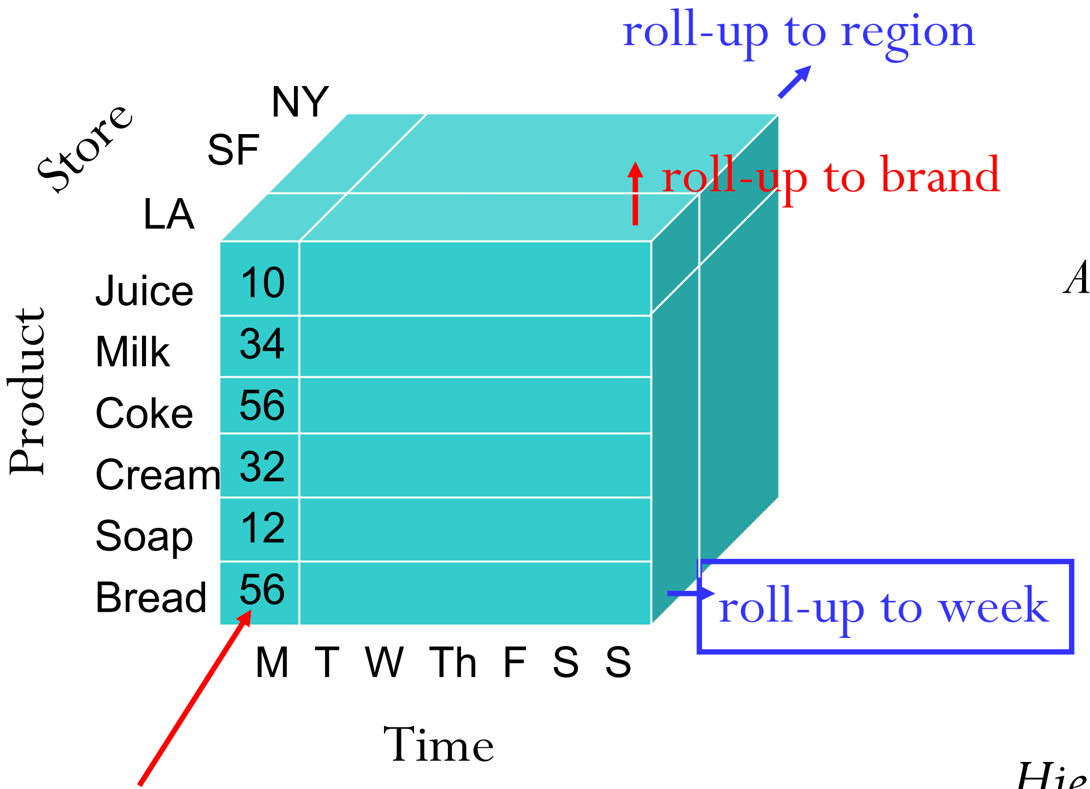
	s1	s2	s3
p1	44	4	

day 1

	s1	s2	s3
p1	12		50
p2	11	8	

Dimensions = 3

Example



roll-up to region

roll-up to brand

roll-up to week

56 units of bread sold in LA on M

Dimensions:

Time, Product, Store

Attributes:

Product (price, …)

Store (name, location, …)

Hierarchies:

Product → Brand → …

Day → Week → Quarter

Store → Region → Country

day 2

	s1	s2	s3
p1	44	4	

day 1

	s1	s2	s3
p1	12		50
p2	11	8	

	s1	s2	s3
sum	67	12	50

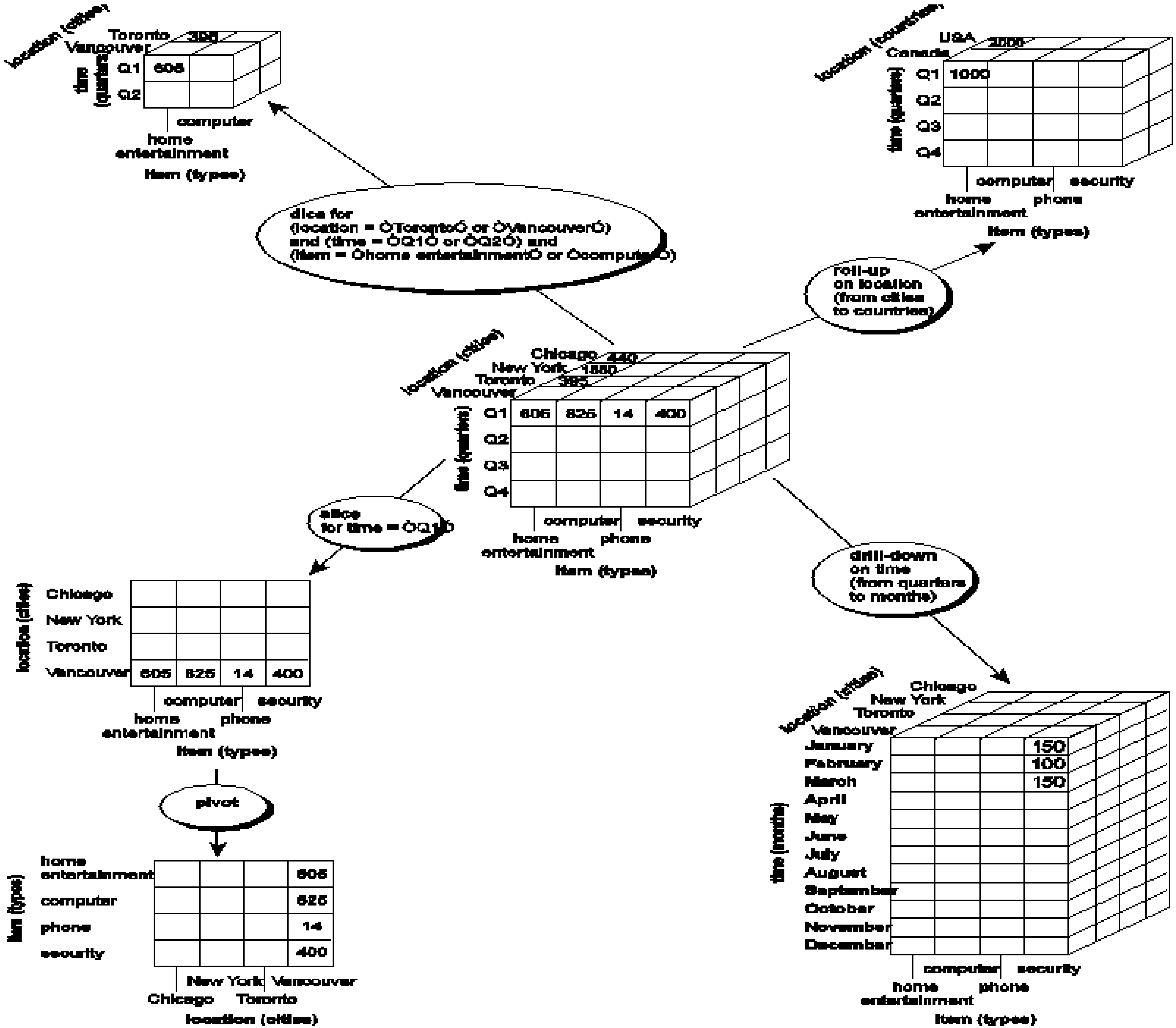
	s1	s2	s3
p1	56	4	50
p2	11	8	

129

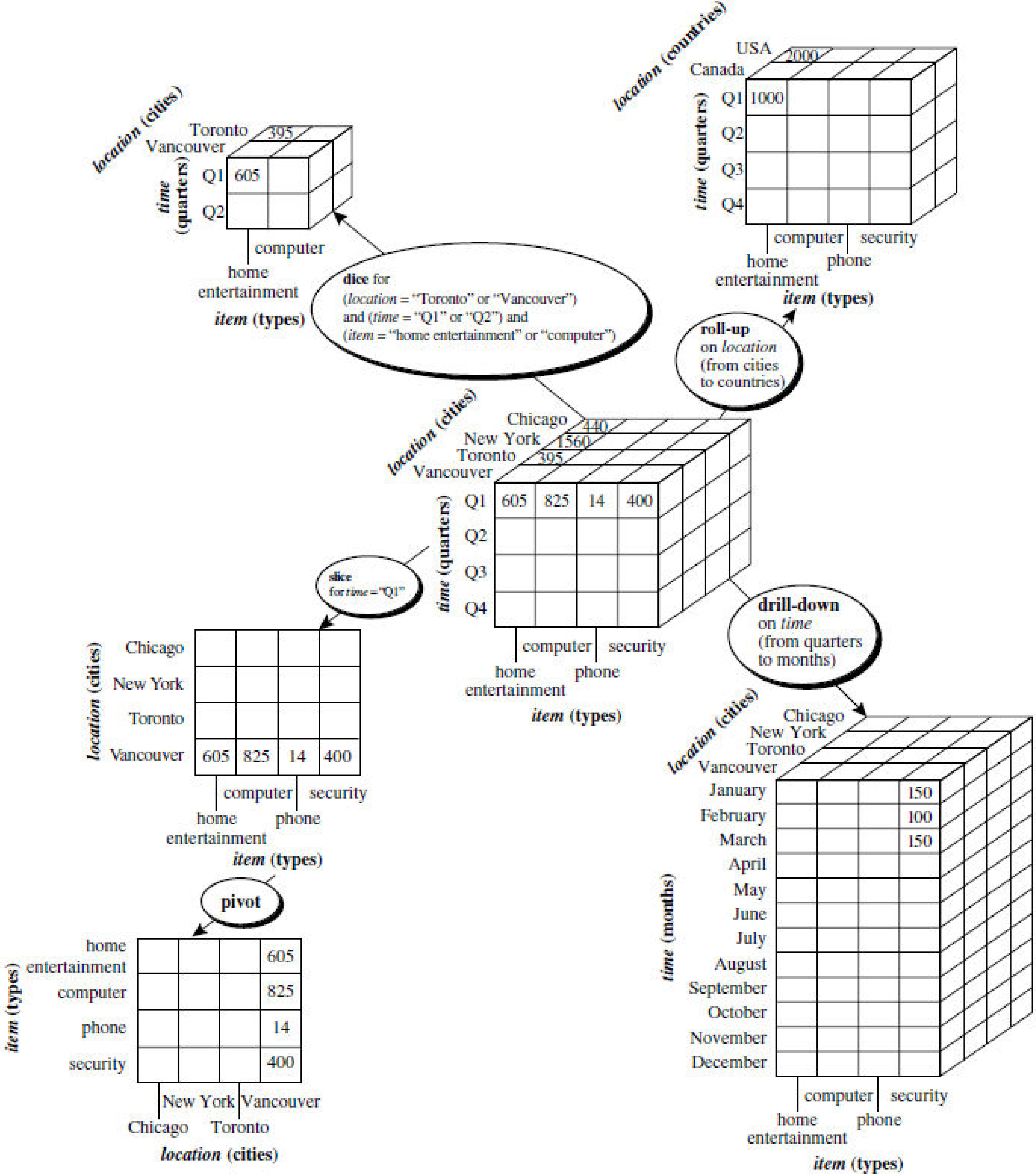
	sum
p1	110
p2	19

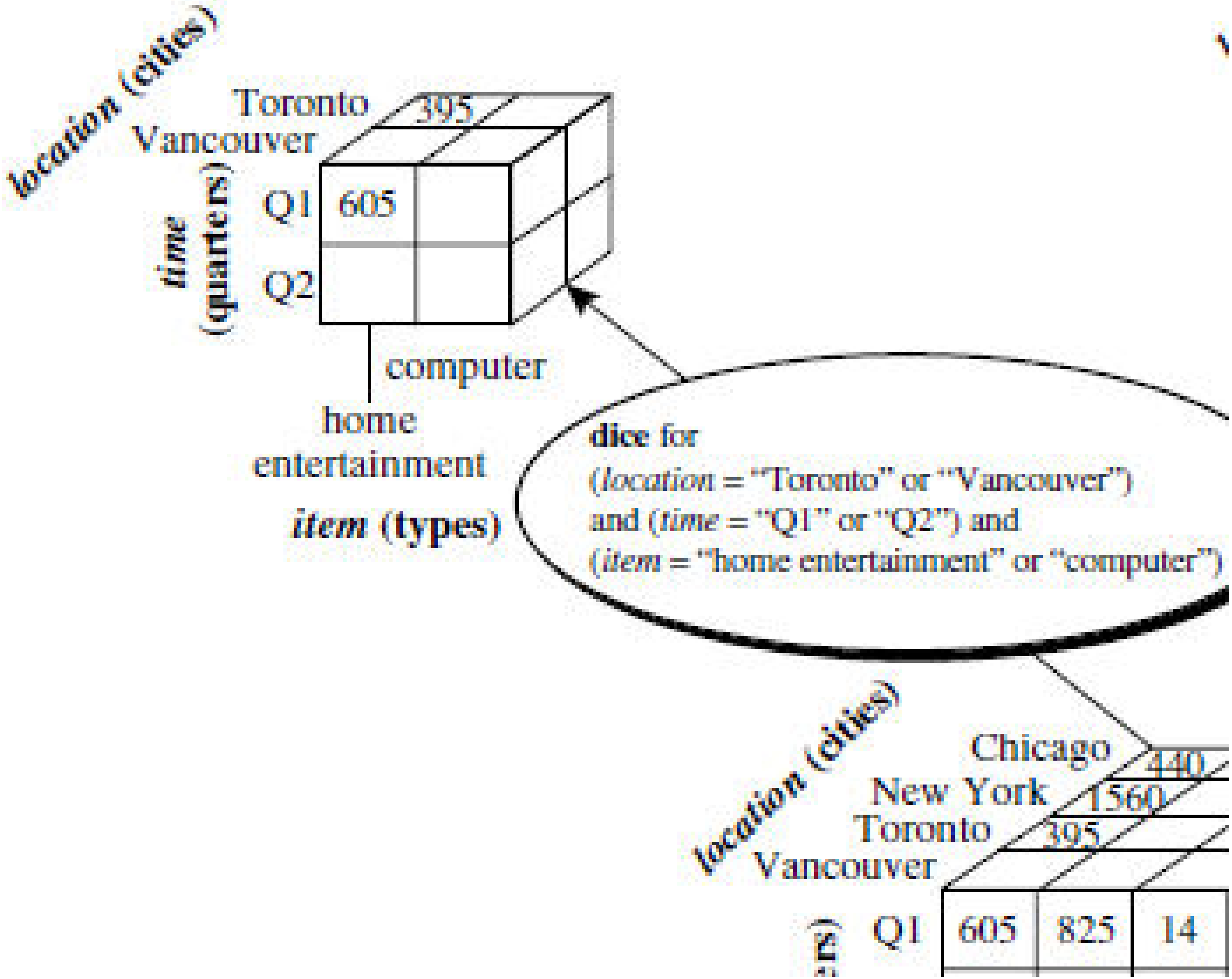
rollup →

← drill-down

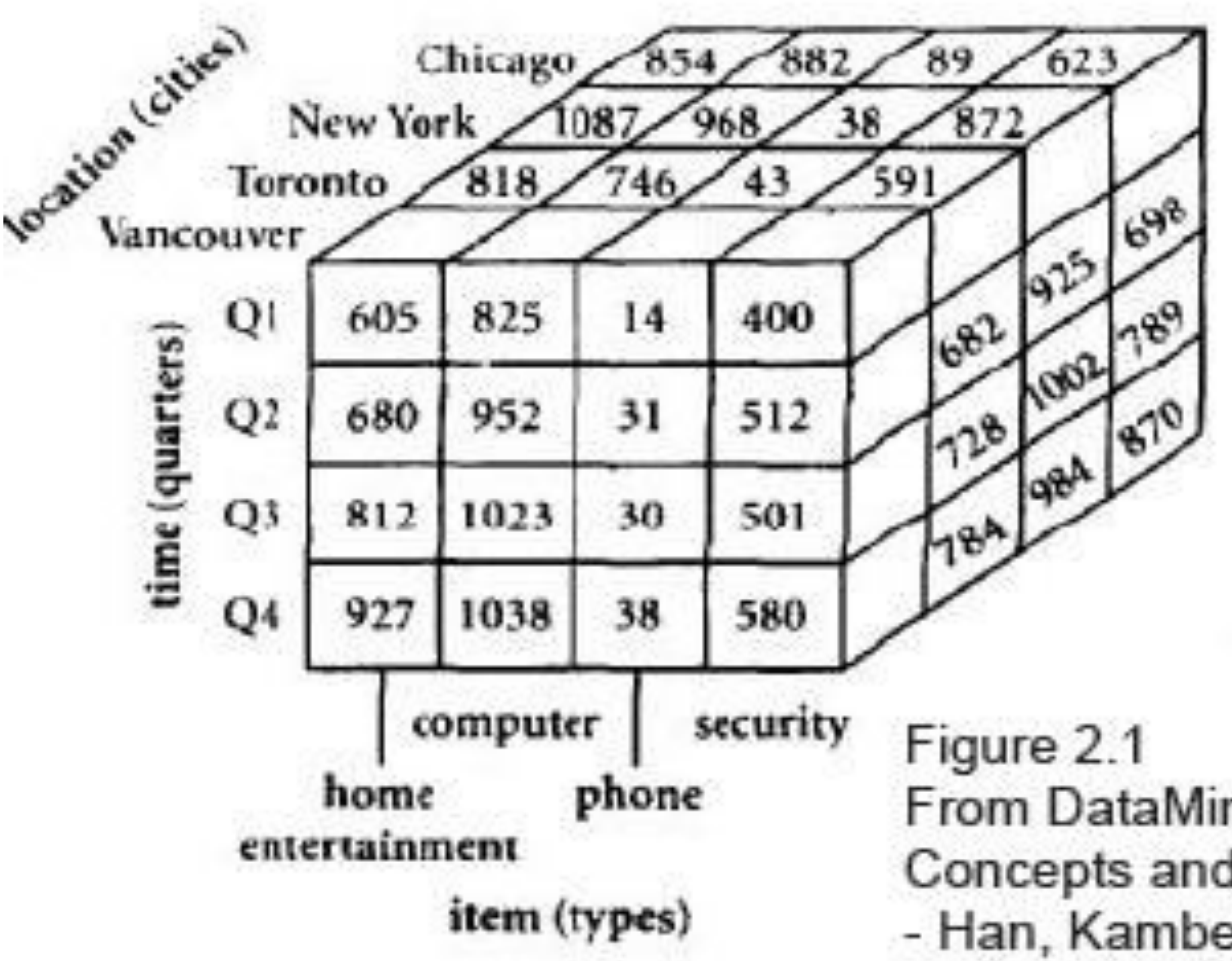


location (cities)
Chicago 440
New York 1560
Toronto 395
Vancouver
time (quarters)
Q1 605 825 14 400
Q2
Q3
Q4
home entertainment
computer
phone
security
item (types)
dice for
(location = ÒTorontoÓ or ÒVancouverÓ)
and (time = ÒQ1Ó or ÒQ2Ó) and
(item = Òhome entertainmentÓ or ÒcomputerÓ)
location (cities)
Toronto 395
Vancouver
time (quarters)
Q1 605
Q2
home entertainment
computer
item (types)
roll-up
on location
(from cities
to countries)
location (countries)
USA 2000
Canada
time (quarters)
Q1 1000
Q2
Q3
Q4
home entertainment
computer
phone
security
item (types)
slice
for time = ÒQ1Ó
location (cities)
Chicago
New York
Toronto
Vancouver 605 825 14 400
home entertainment
computer
phone
security
item (types)
pivot
item (types)
home entertainment 605
computer 825
phone 14
security 400
Chicago
New York
Toronto
Vancouver
location (cities)
drill-down
on time
(from quarters
to months)
location (cities)
Chicago
New York
Toronto
Vancouver
time (months)
January 150
February 100
March 150
April
May
June
July
August
September
October
November
December
home entertainment
computer
phone
security
item (types)


location (cities)
Toronto
Vancouver
395
Q1
605
Q2
time (quarters)
computer
home entertainment
item (types)
dice for (location = "Toronto" or "Vancouver") and (time = "Q1" or "Q2") and (item = "home entertainment" or "computer")
location (countries)
USA
Canada
2000
Q1
1000
Q2
Q3
Q4
time (quarters)
computer
security
home entertainment
phone
item (types)
roll-up on location (from cities to countries)
location (cities)
Chicago
New York
Toronto
Vancouver
440
1560
395
Q1
605
825
14
400
Q2
Q3
Q4
time (quarters)
computer
security
home entertainment
phone
item (types)
slice for time = "Q1"
location (cities)
Chicago
New York
Toronto
Vancouver
605
825
14
400
computer
security
home entertainment
phone
item (types)
pivot
item (types)
home entertainment
605
computer
825
phone
14
security
400
New York
Vancouver
Chicago
Toronto
location (cities)
drill-down on time (from quarters to months)
location (cities)
Chicago
New York
Toronto
Vancouver
January
150
February
100
March
150
April
May
June
July
August
September
October
November
December
time (months)
computer
security
home entertainment
phone
item (types)

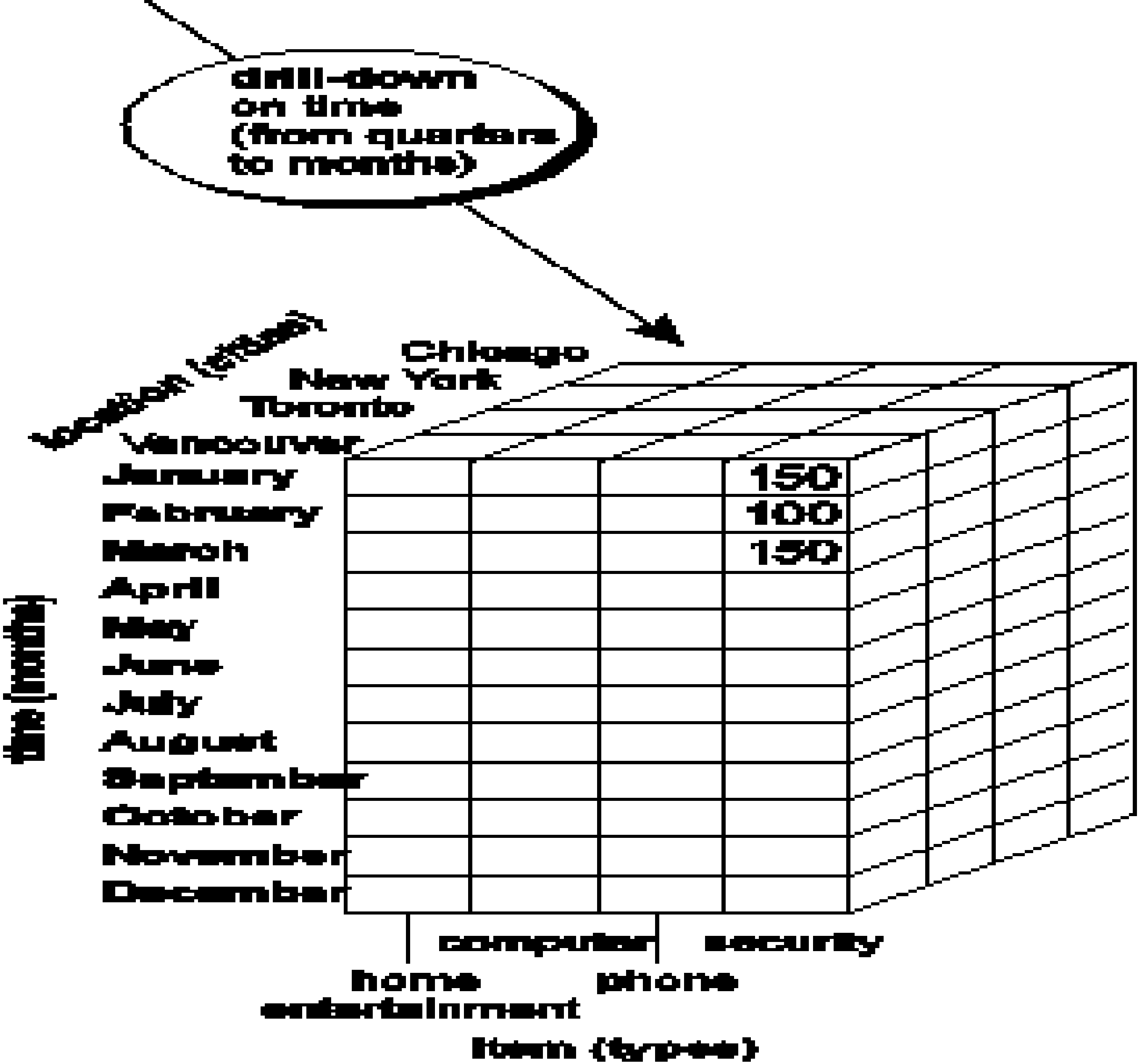


dice for
(*location* = "Toronto" or "Vancouver")
and (*time* = "Q1" or "Q2") and
(*item* = "home entertainment" or "computer")



	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

Figure 2.1
From DataMining:
Concepts and tech.
- Han, Kamber



drill-down on time (from quarters to months)