Module 1

Data Warehouse (DWH) Fundamentals
Introduction to Data Mining

Data Warehouse (DWH) Fundamentals

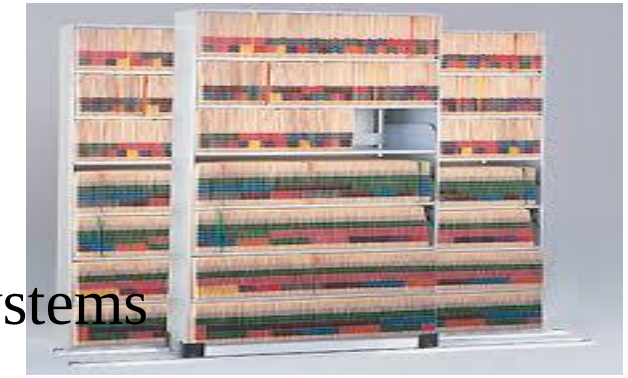
Introduction and Overview

Data warehousing – related to ***organized*** storage of a ***large volume*** of data so as to ease retrieval of ***useful*** information

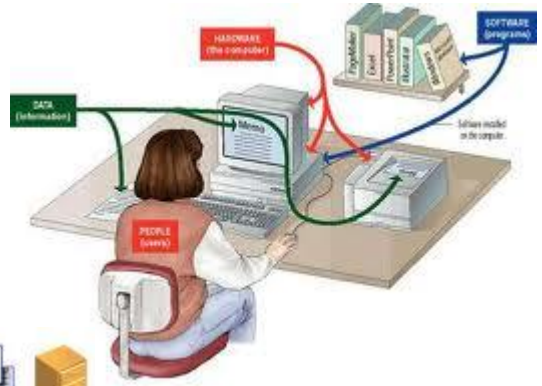
Data mining – related to ***intelligent*** information retrieval from a ***large source*** of data so as to extract ***useful*** information

Brief History of Information Technology

- The dark age

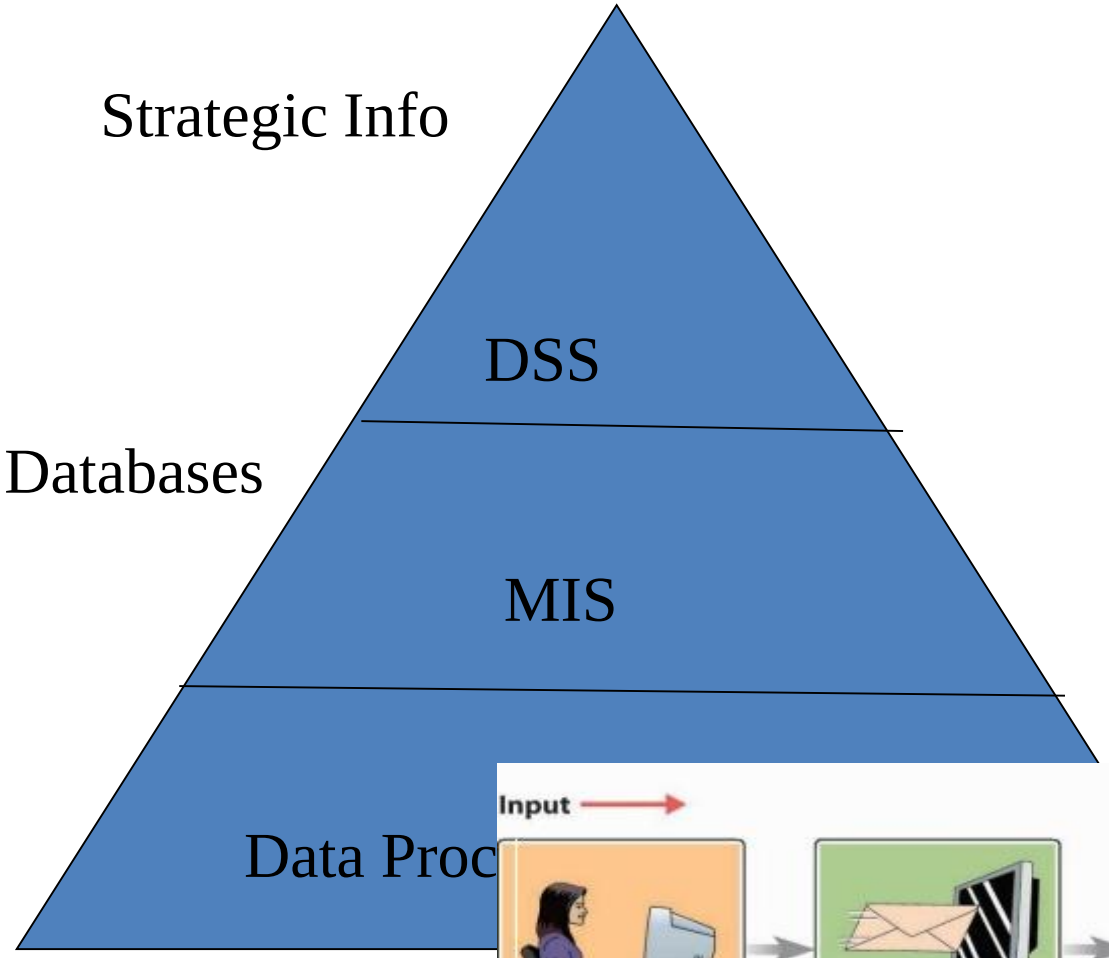


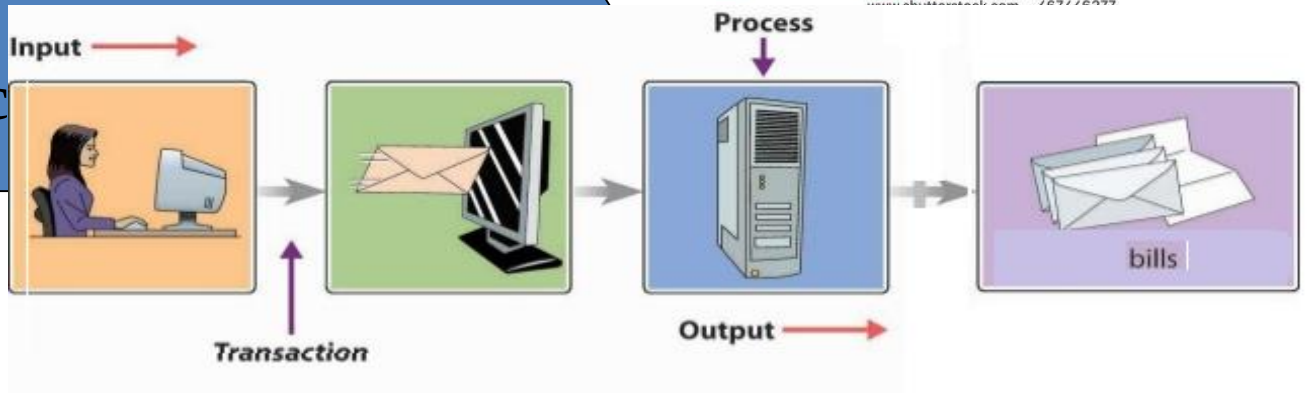
- Computerized systems emerge



- The next step: use data fo decision-making

Information System Pyramid





Questions That Are Important

- Who are my most profitable customers?
- Is the productivity of our Sales Channel improving?
- How can I provision orders faster?
- Which areas are doing well and which are not
- Are certain periods of time bad for business
- How will product sales impact the network?
- Can distribution be better
- How should I apply upselling/cross-selling?

Hierarchy Of Information Structures

- Data
- Bits (bytes)
- Fields
- Records
- Files
- Databases
- RDBMS
- ??? Data Warehouses – (Knowledge Structures)

Motivating Example

Chain of stores

Typical RDBMS Design

- create table product_categories (product_category_id integer primary key, product_category_name varchar(100) not null);
- create table manufacturers (manufacturer_id integer primary key, manufacturer_name varchar(100) not null);
- create table products (product_id integer primary key, product_name varchar(100) not null, product_category_id references product_categories, manufacturer_id references manufacturers);
- create table cities (city_id integer primary key, city_name varchar(100) not null, state varchar(100) not null, population integer not null);
- create table stores (store_id integer primary key, city_id references cities, store_location varchar(200) not null, phone_number varchar(20));
- create table sales (product_id not null references products, store_id not null references stores, quantity_sold integer not null, date_time_of_sale date not null);

Populate the Database

- insert into product_categories values (1, 'toothpaste');
- insert into product_categories values (2, 'soda');
- insert into manufacturers values (68, 'Colgate');
- insert into manufacturers values (5, 'Coca Cola');
- insert into products values (567, 'Colgate Gel Pump 6.4 oz.', 1, 68);
- insert into products values (219, 'Diet Coke 12 oz. can', 2, 5);
- insert into cities values (34, 'San Francisco', 'California', 700000);
- insert into cities values (58, 'East Fishkill', 'New York', 30000);
- insert into stores values (16, 34, '510 Main Street', '415-555-1212');
- insert into stores values (17, 58, '13 Maple Avenue', '914-555-1212');
- insert into sales values (567, 17, 1, to_date('1997-10-22 09:35:14', 'YYYY-MM-DD HH24:MI:SS'));
- insert into sales values (219, 16, 4, to_date('1997-10-22 09:35:14', 'YYYY-MM-DD HH24:MI:SS'));
- insert into sales values (219, 17, 1, to_date('1997-10-22 09:35:17', 'YYYY-MM-DD HH24:MI:SS'));

The Database

SALES table

product id	store id	qty sold	date/time of sale
567	17	1	1997-10-22 09:35:14
219	16	4	1997-10-22 09:35:14
219	17	1	1997-10-22 09:35:17

...

PRODUCTS table

prod id	prod name	prod category	man id
567	**Colgate Gel Pump 6.4 oz.**	1	68
219	**Diet Coke 12 oz. can**	2	5

The Database

PRODUCT_CATEGORIES table

product category id	product category name
1	toothpaste
2	soda

MANUFACTURERS table

manufacturer id	manufacturer name
68	Colgate
5	Coca Cola

The Database

STORES table

store id	city id	store location	phone number
16	34	510 Main Street	415-555-1212
17	58	13 Maple Avenue	914-555-1212
...			

CITIES table

city id	city name	state	population
34	San Francisco	California	700,000
58	East Fishkill	New York	30,000
...			

Strategic Information Retrieval

- After a few months of stuffing data into these tables,
- Assume the following query say after a recent Colgate promotion, directed at people who live in small towns.
- ***How much Colgate toothpaste did we sell in those towns yesterday?***
- ***And how much on the same day a month ago?"***

The Query

```
select sum(sales.quantity_sold) from sales, products,
    product_categories, manufacturers, stores, cities where
    manufacturer_name = 'Colgate' & product_category_name =
    'toothpaste' & cities.population < 40000 and
    trunc(sales.date_time_of_sale) = trunc(sysdate-1) -- restrict to
    yesterday and sales.product_id = products.product_id &
    sales.store_id = stores.store_id and products.product_category_id
    = product_categories.product_category_id &
    products.manufacturer_id = manufacturers.manufacturer_id &
    stores.city_id = cities.city_id
```

Query processing

- This query would be tough for a novice to read and, being a 6-way JOIN of some fairly large tables, might take quite a while to execute. Also, these tables are being updated as the query is executed.
- At some times any attempt to update the database results in the computer freezing up for 20 minutes. Eventually the database administrators realize that

  the system collapses every time the toothpaste query gets run !!!!!!!!

- Conclusion : The system is an on-line transaction processing (OLTP) system. ***You can't feed it a decision support system (DSS) query and expect things to work!"***

- ***We Need A different Store Of data***

The problem

- The preceding type of queries will arise commonly in DSS
- The query will take a very long time for execution each time
- All these suggest a different type of store of data
- Enter **“*The Data Warehouse*”**

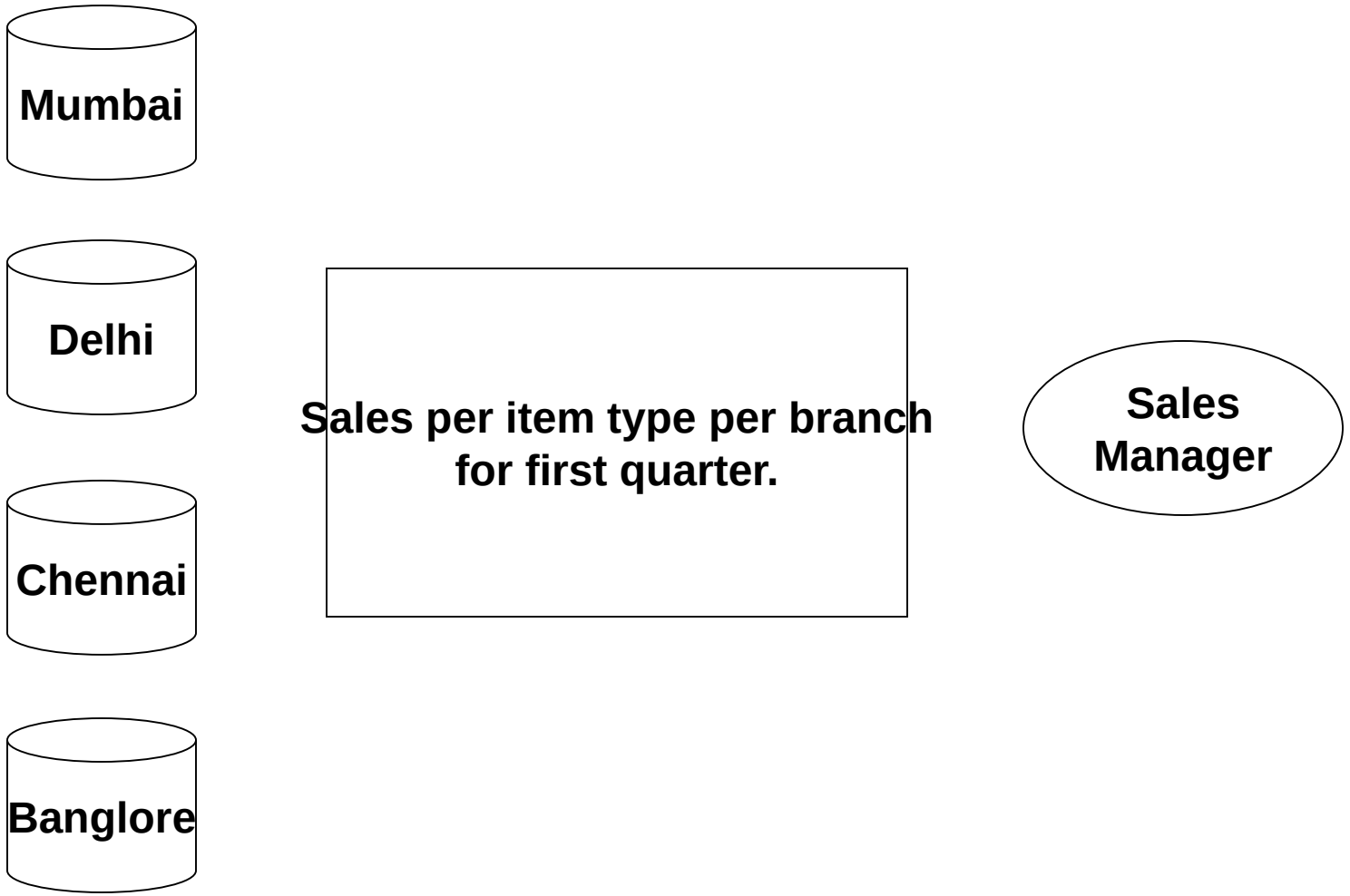
Evolution of Data Warehousing

- Organizations now focus on ways to use operational data to support decision-making, as a means of gaining competitive advantage.
- However, operational systems were never designed to support such business activities.

Scenario 1

ABC Pvt Ltd is a company with branches at Mumbai, Delhi, Chennai and Banglore. The Sales Manager wants quarterly sales report. Each branch has a separate operational system.

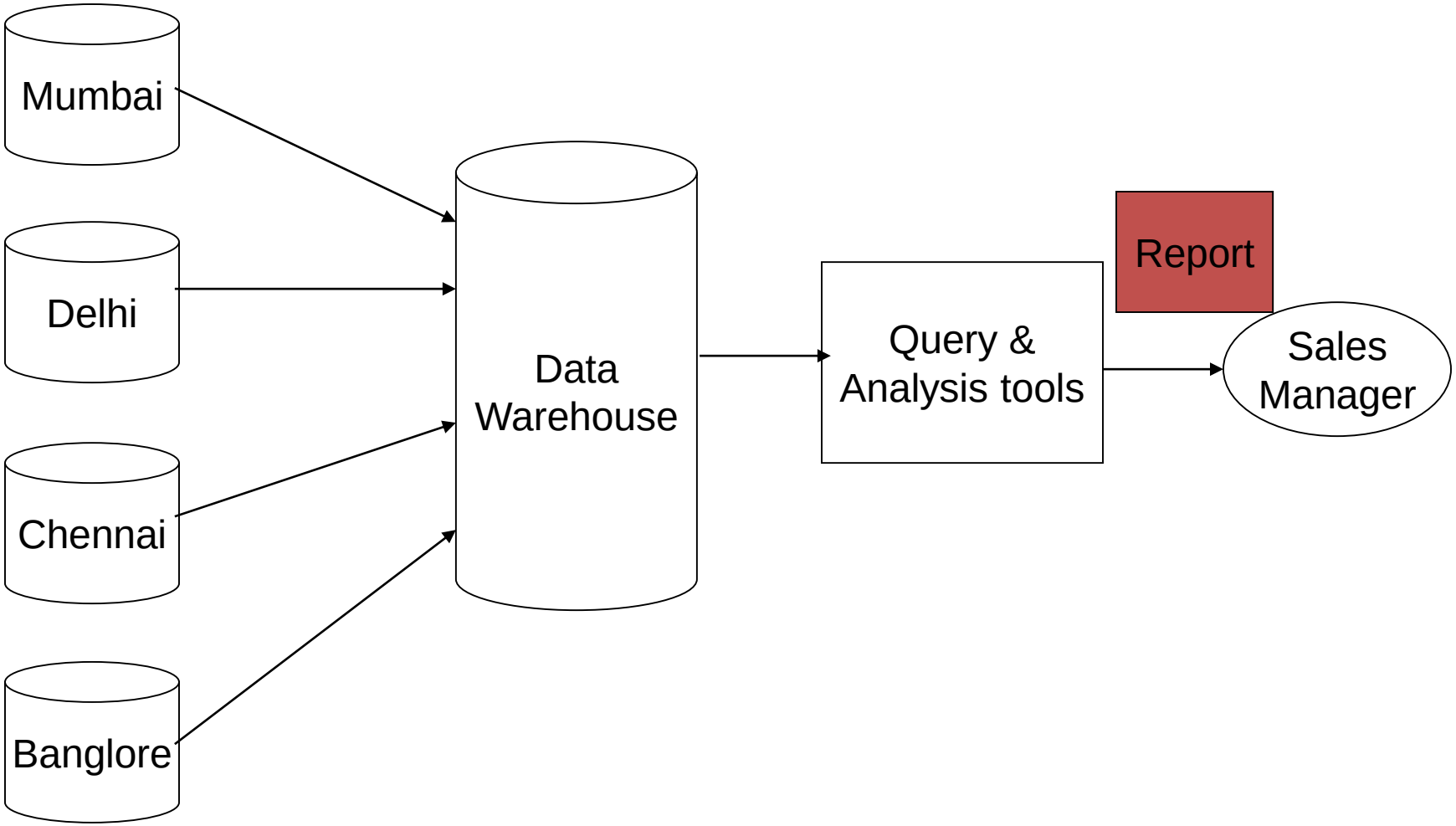
Scenario 1 : ABC Pvt Ltd.



Solution 1:ABC Pvt Ltd.

- Extract sales information from each database.
- Store the information in a common repository at a single site.

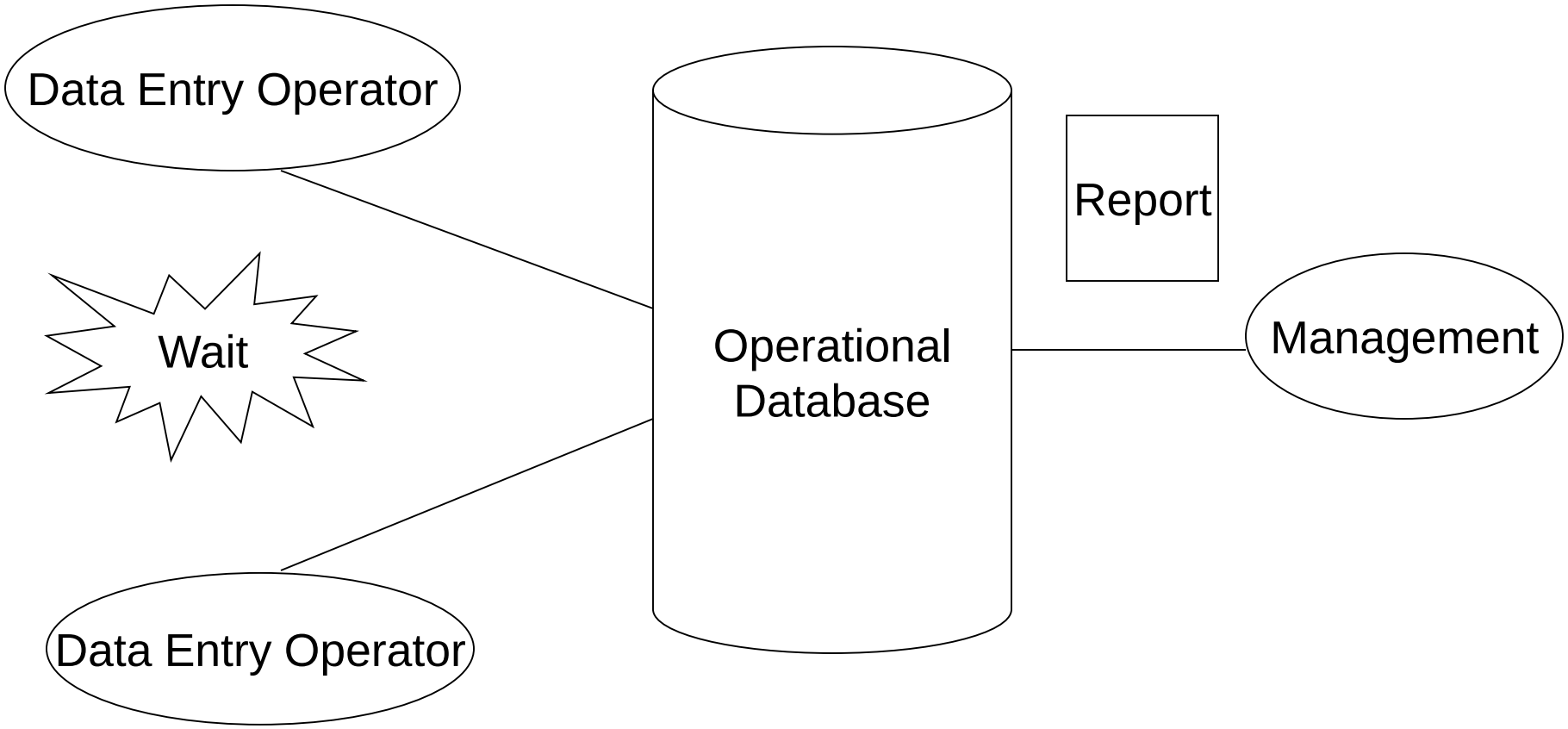
Solution 1:ABC Pvt Ltd.



Scenario 2

One Stop Shopping Super Market has huge operational database.Whenever Executives wants some report the OLTP system becomes
slow and data entry operators have to wait for some time.

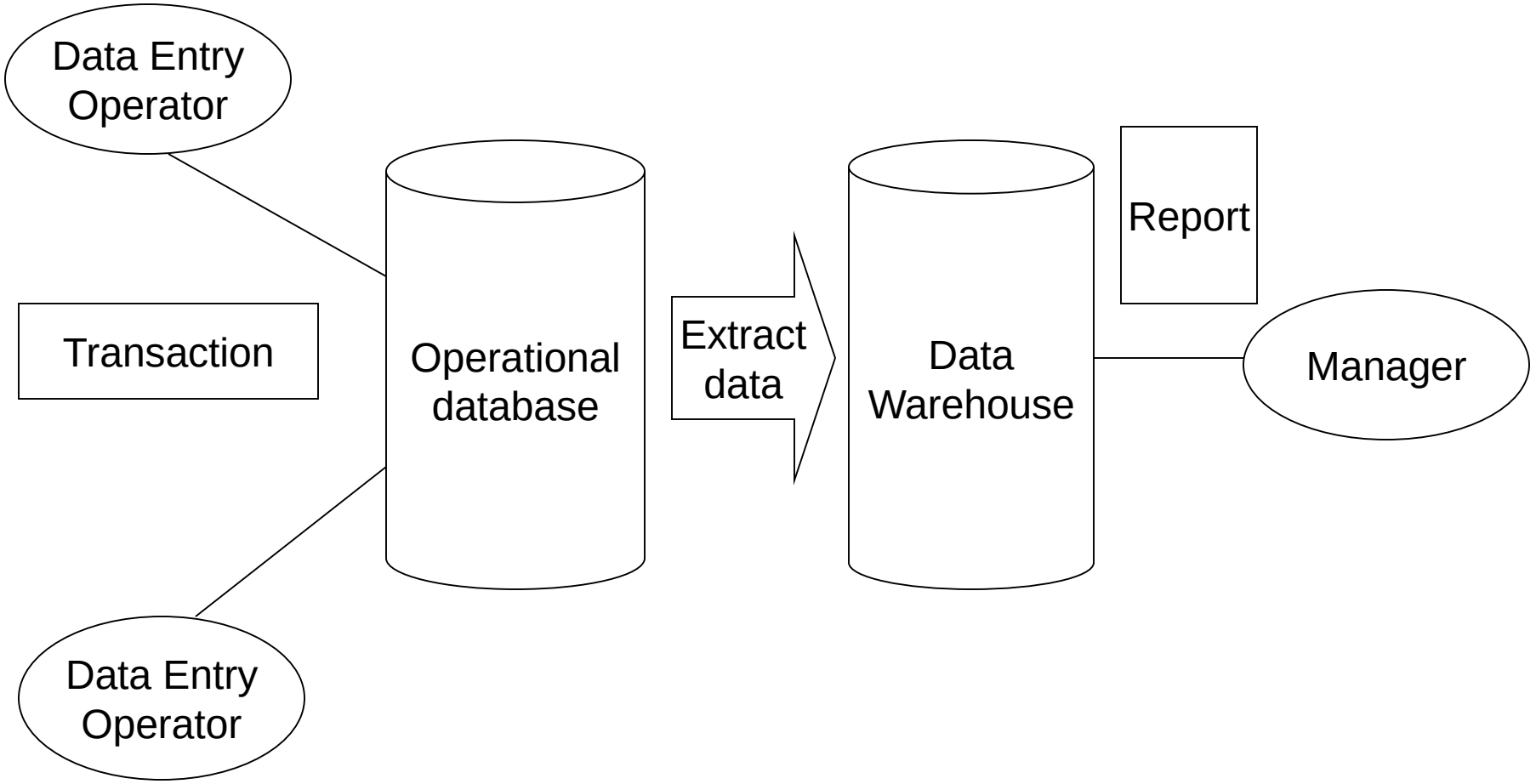
Scenario 2 : One Stop Shopping



Solution 2

- Extract data needed for analysis from operational database.
- Store it in warehouse.
- Refresh warehouse at regular interval so that it contains up to date information for analysis.
- Warehouse will contain data with historical perspective.

Solution 2


Data Entry Operator
Transaction
Data Entry Operator
Operational database
Extract data
Data Warehouse
Report
Manager

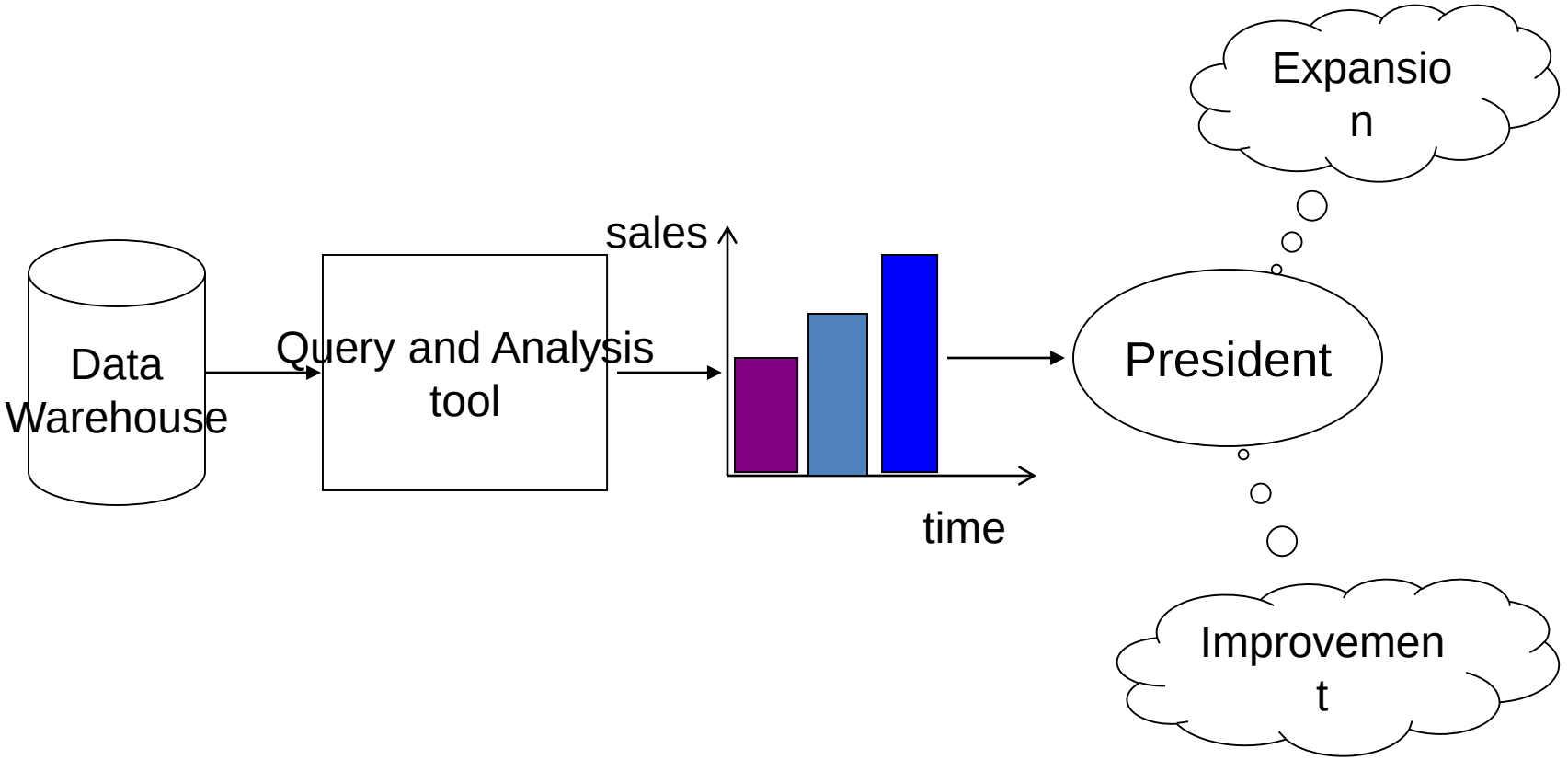
Scenario 3

Cakes & Cookies is a small,new company.President of the company wants his company should grow.He needs information so that he can make correct decisions.

Solution 3

- Improve the quality of data before loading it into the warehouse.
- Perform data cleaning and transformation before loading the data.
- Use query analysis tools to support adhoc queries.

Solution 3


Expansio
n
sales
Data
Warehouse
Query and Analysis
tool
President
time
Improvemen
t

What is Data Warehouse??

Data warehouse

A data warehouse is

-subject-oriented,

-integrated,

-time-variant,

-nonvolatile

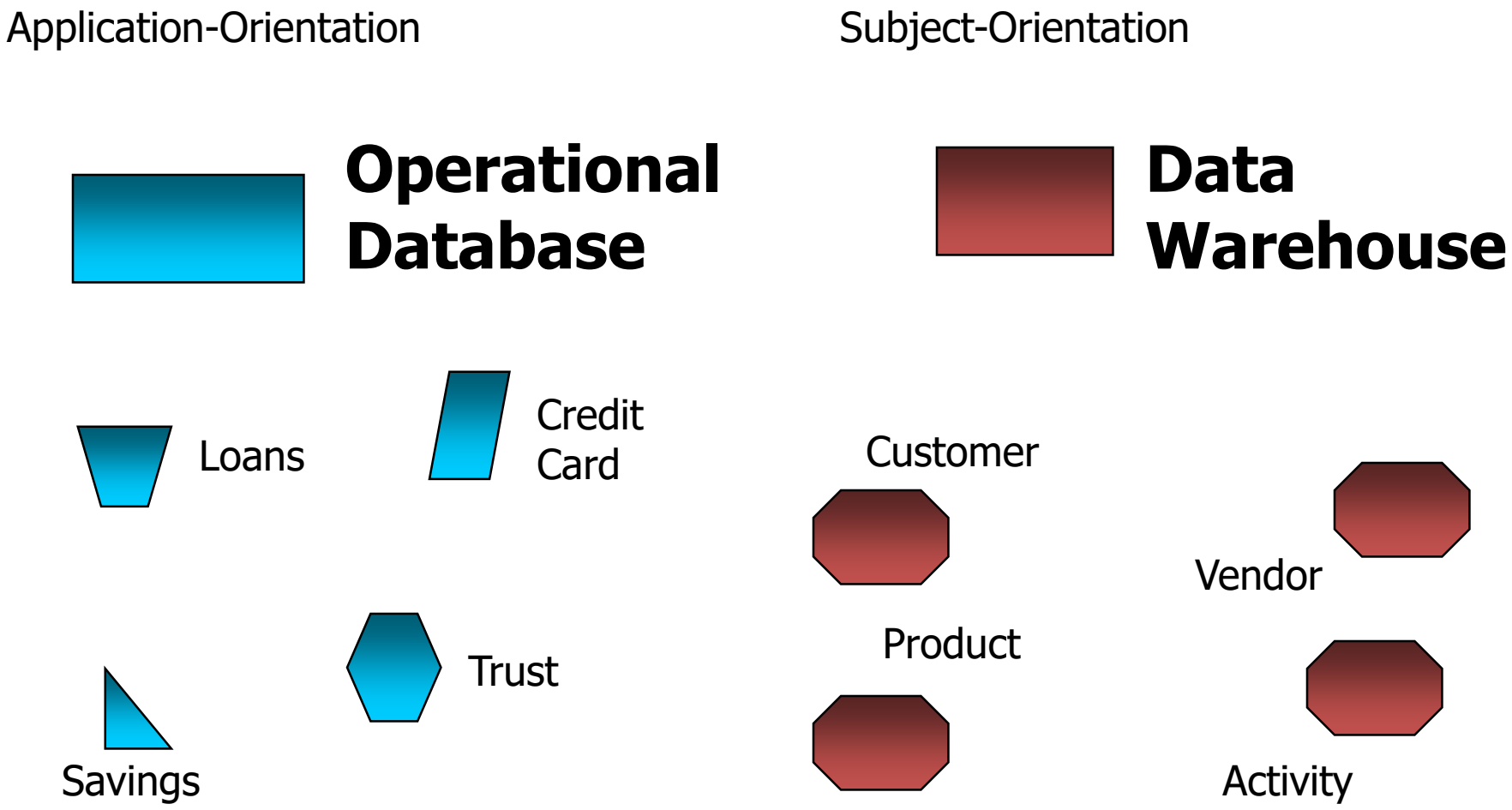
collection of data in support of management's decision making process.

---Bill Inmon

Subject-oriented

- Data warehouse is organized around subjects such as sales,product,customer.
- It focuses on modeling and analysis of data for decision makers.
- Excludes data not useful in decision support process.
- Subject areas can be determined by *5W1H rules*
  - **When, where, who, what, why, how**

Application-Orientation vs. Subject-Orientation



Subject Vs Operational data

- Example : Order Processing
- Operational Data:
  - Order entry
  - Validation
  - Shipping
  - Accts Payable etc

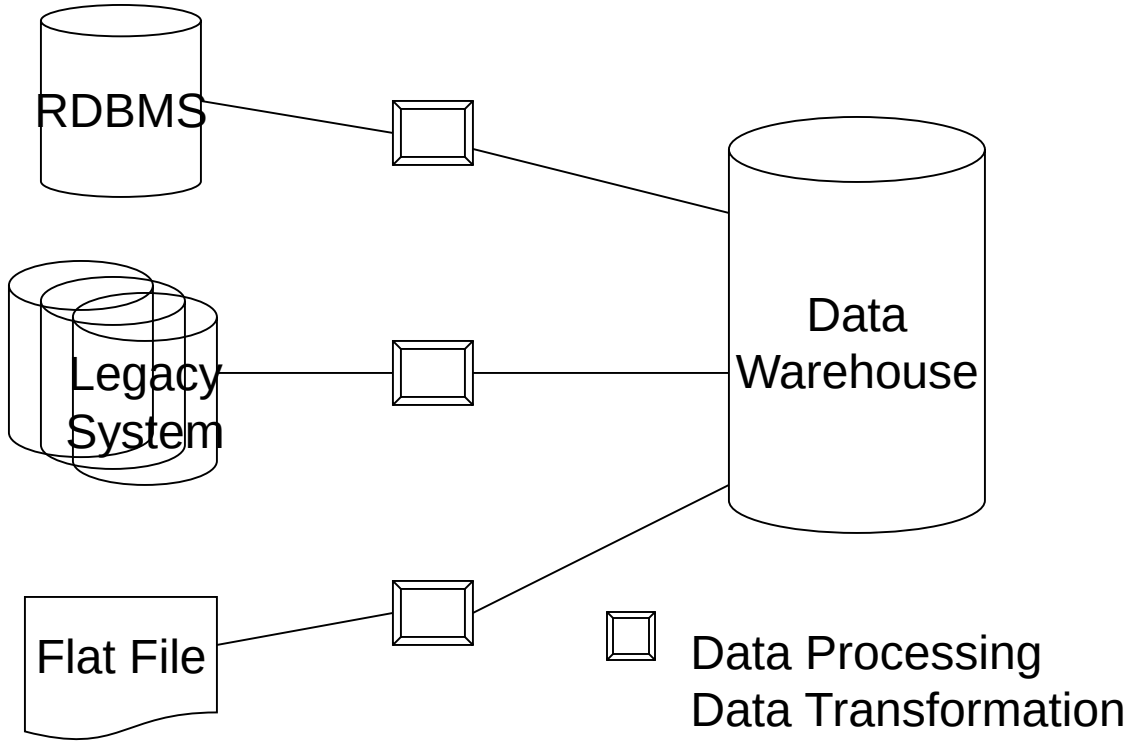
- Subject Oriented:
  - Customer
  - Sales
  - Product
  - Duration etc

Subject-oriented

- For an insurance company, the applications may be auto, health, life, and casualty.
  - The major subject areas of the insurance corporation might be customer, policy, premium, and claim.
- For a manufacturer,
  - the major subject areas might be product, order, vendor, bill of material, and raw goods.
- For a retailer,
  - the major subject areas may be product, SKU, sale, vendor, and so forth.
- Each type of company has its own unique set of subjects

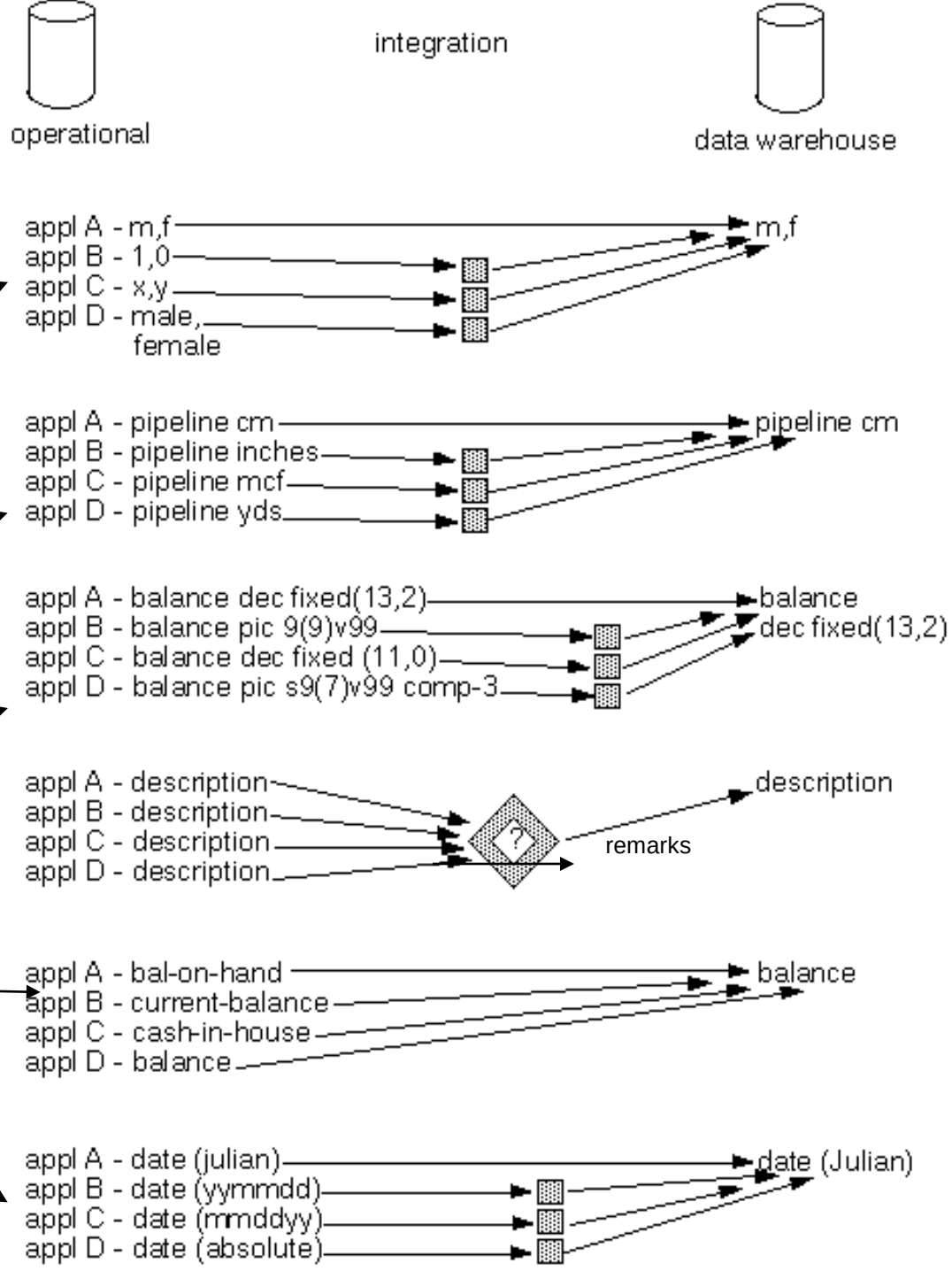
Integration

- Data Warehouse is constructed by integrating multiple heterogeneous sources.
- Data Preprocessing are applied to ensure consistency.



Integration

- In terms of data.
  - encoding structures.
  - Measurement of attributes.
  - physical attribute. of data
  - naming conventions.
  - Data type format



DW Definition…

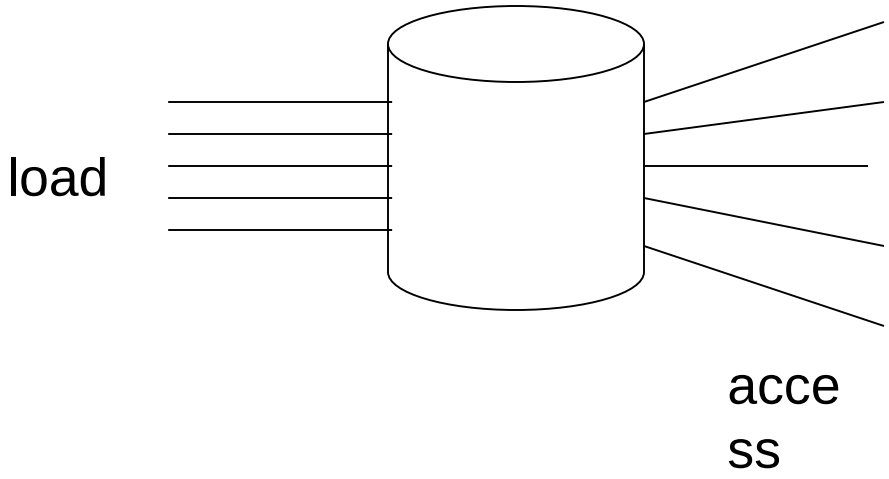
- Time-variant
  - The data in the warehouse contain a time dimension so that they may be used as a historical record of the business
- The periodically addition of data to a Data Warehouse is called refreshing
- During refreshing, all time dependent aggregations are computed again
- Example:
  - If the sales data from the last time period are loaded into a Data Warehouse, aggregated sales data for products, customers, etc have to be recalculated

Time-variant

- Provides information from historical perspective e.g. past 5-10 years

Nonvolatile

- Data once recorded cannot be updated.
- Data warehouse requires two operations in data accessing
  - Initial loading of data
  - Access of data



Data Warehouse—Non-Volatile

- 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**.

Non Volatility

- Non volatility pertains to the fact that a Data Warehouse data are (rarely) deleted and relatively rarely updated
- There are mainly only new data added
- This way a Data Warehouse grows all the time
- So, a Data Warehouse has to manage a many Terabyte database
- In practice, a Data Warehouse is often “resynchronized” to keep its volume manageable
  - **Re-synchronization implies that a Data Warehouse stores data for a fixed number of time periods**
  - **After a new period is added, the oldest is deleted**

Non Volatile Data

- Data in warehouse not intended for **day to day business**
- Data from ODS are moved to the warehouse at specific intervals. This is called a warehouse **“refresh”**
- Data in warehouse is primarily for **query** and **analysis**

Data Marts

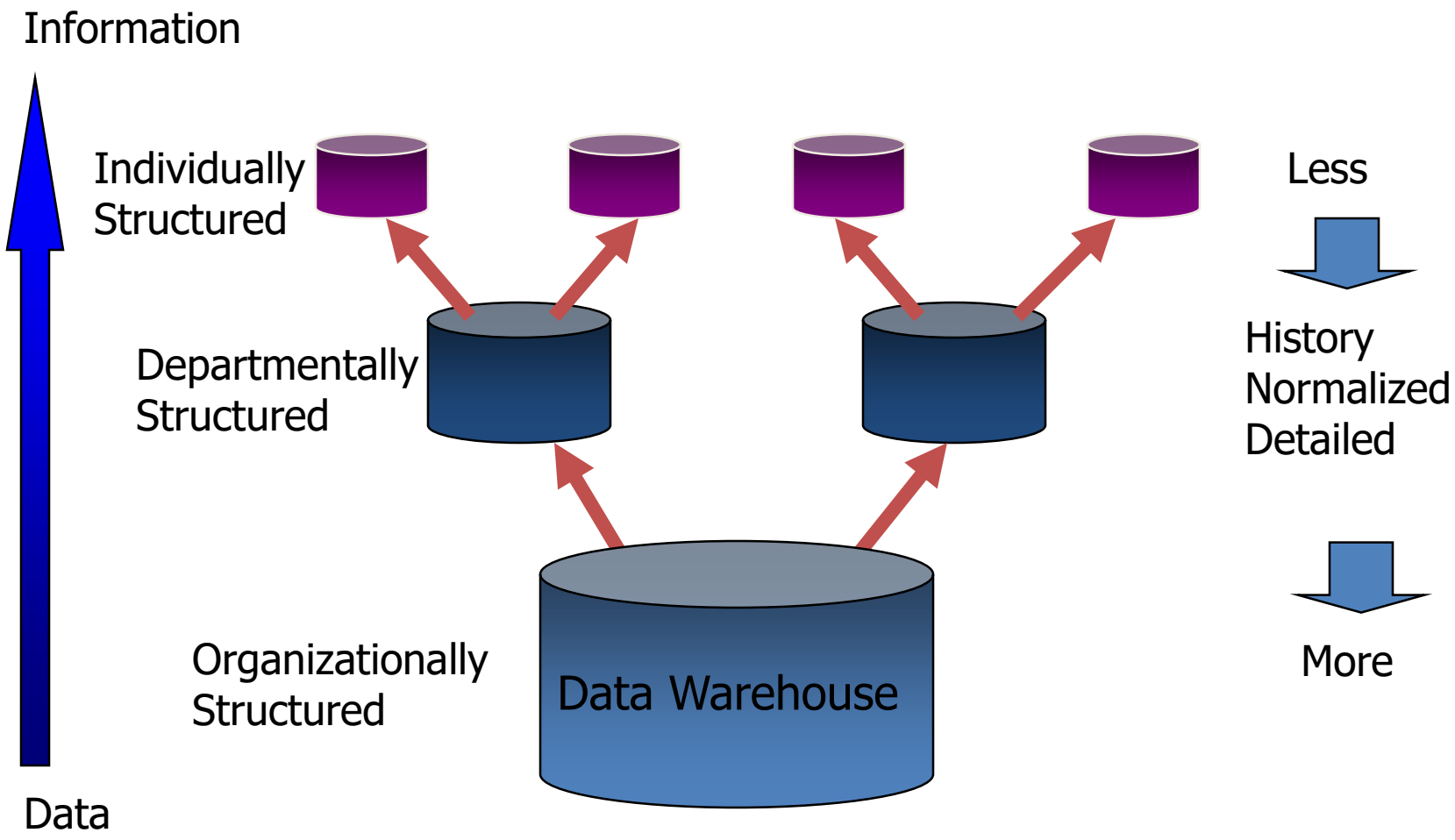
Data Mart

- A Data Mart is a smaller, more focused Data Warehouse – a mini-warehouse.
- A Data Mart typically reflects the business rules of a specific business unit within an enterprise.

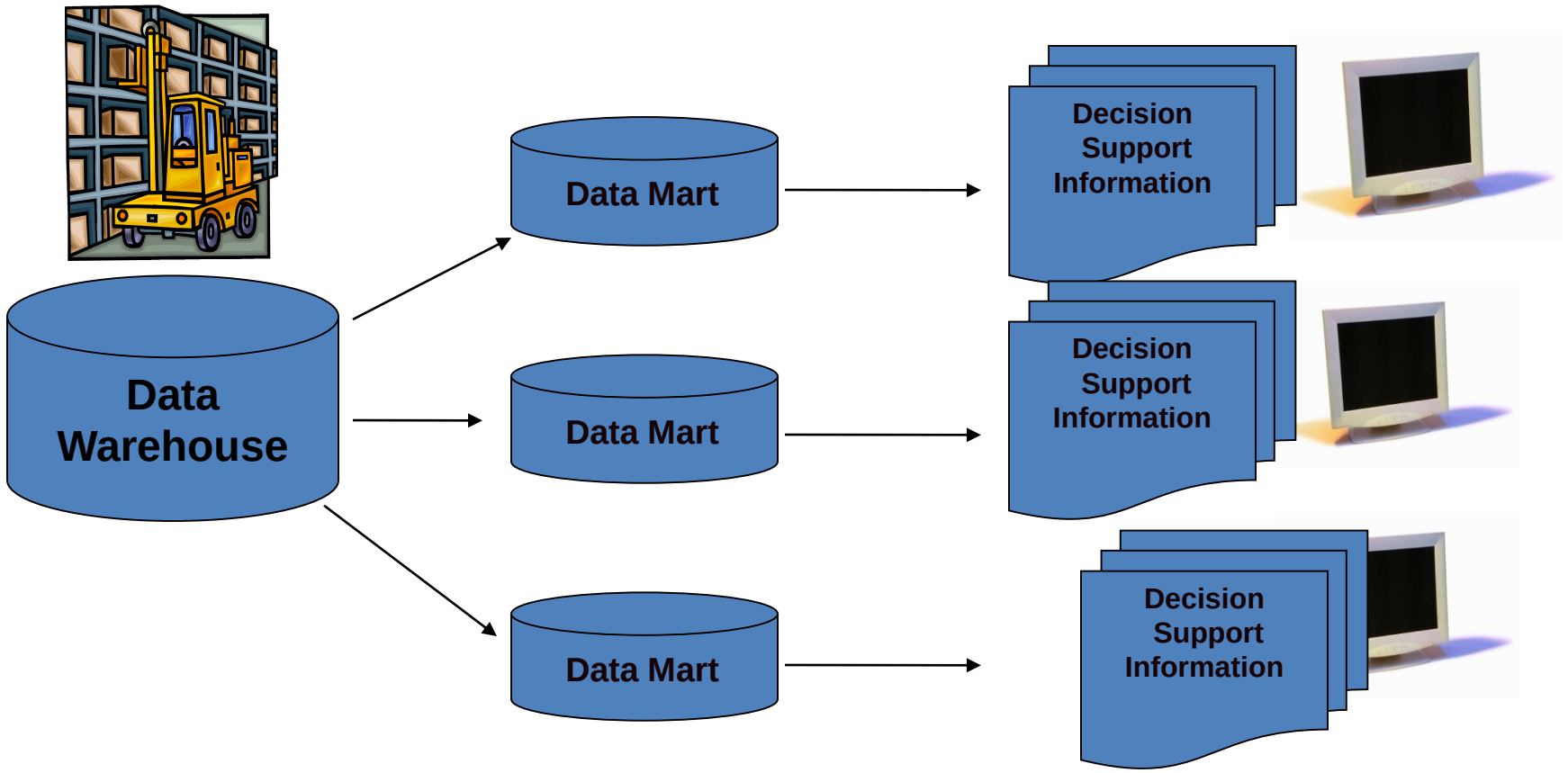
Data Mart

- A subset of a data warehouse that supports the requirements of a particular department or business function.
- Characteristics include:
  - Focuses on only the requirements of one department or business function.
  - Do not normally contain detailed operational data unlike data warehouses.
  - More easily understood and navigated.

From the Data Warehouse to Data Marts



Data Warehouse to Data Mart



Parameter	Data Warehouse	Data Mart
Definition	A Data Warehouse is a large repository of data collected from different organizations or departments within a corporation.	A data mart is an only subtype of a Data Warehouse. It is designed to meet the need of a certain user group.
Usage	It helps to take a strategic decision.	It helps to take tactical decisions for the business.
Objective	The main objective of Data Warehouse is to provide an integrated environment and coherent picture of the business at a point in time.	A data mart mostly used in a business division at the department level.
Designing	The designing process of Data Warehouse is quite difficult.	The designing process of Data Mart is easy.
	May or may not use in a dimensional model. However, it can feed dimensional models.	It is built focused on a dimensional model using a start schema.
Data Handling	Data warehousing includes large area of the corporation which is why it takes a long time to process it.	Data marts are easy to use, design and implement as it can only handle small amounts of data.
Focus	Data warehousing is broadly focused all the departments. It is possible that it can even represent the entire company.	Data Mart is used at a department level.

Data type	The data stored inside the Data Warehouse are always detailed when compared with data mart.	Data Marts are built for particular user groups. Therefore, data short and limited.
Subject-area	The main objective of Data Warehouse is to provide an integrated environment and coherent picture of the business at a point in time.	Mostly hold only one subject area- for example, Sales figure.
Data storing	Designed to store enterprise-wide decision data, not just marketing data.	Dimensional modeling and star schema design employed for optimizing the performance of access layer.
Data value	Read-Only from the end-users standpoint.	Transaction data regardless of grain fed directly from the Data Warehouse.
Scope	Data warehousing is more helpful as it can bring information from any department.	Data mart contains data, of a specific department of a company. There are maybe separate data marts for sales, finance, marketing, etc. Has limited usage
Source	In Data Warehouse Data comes from many sources.	In Data Mart data comes from very few sources.
Size	The size of the Data Warehouse may range from 100 GB to 1 TB+.	The Size of Data Mart is less than 100 GB.
Implementation time	The implementation process of Data Warehouse can be extended from months to years.	The implementation process of Data Mart is restricted to few months.

OLTP v/s OLAP

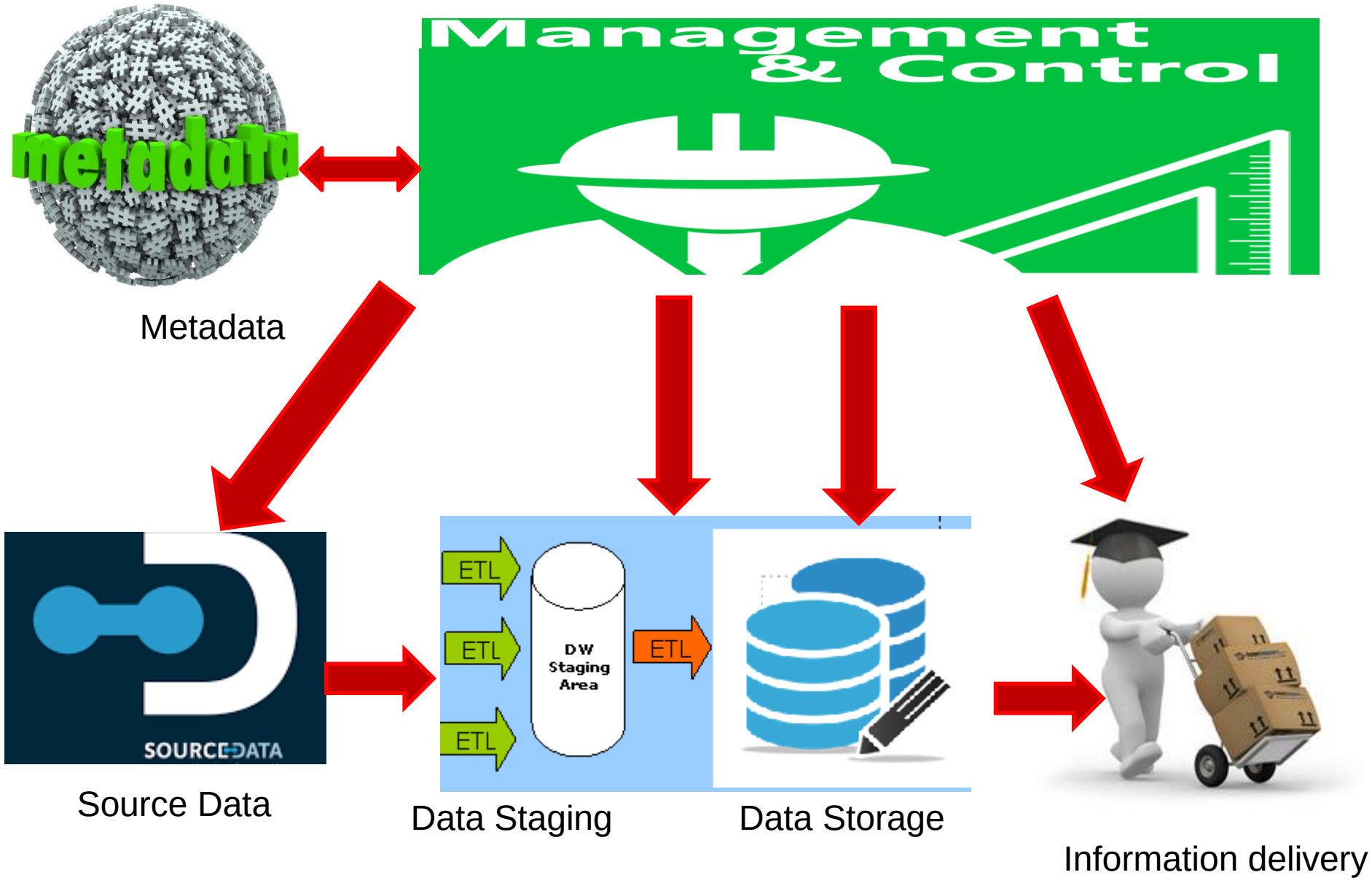
Features	OLTP	OLAP
Characteristics	Operational processing	Informational processing
Orientation	Transaction	Analysis
User	Clerk,DBA,database professional	Knowledge workers
Function	Day to day operation	Decision support
Data	Current	Historical
View	Detailed,flat relational	Summarized, multidimensional
DB design	Application oriented	Subject oriented
Unit of work	Short ,simple transaction	Complex query
Access	Read/write	Mostly read

Features	Operational	OLAP
Focus	Data in	Information out
Number of records accessed	tens	millions
Number of users	thousands	hundreds
DB size	100MB to GB	100 GB to TB
Priority	High performance, high availability	High flexibility,end-user autonomy
Metric	Transaction throughput	Query througput

Sr. No.	Data Warehouse (OLAP)	Operational Database(OLTP)
1	It involves historical processing of information.	It involves day-to-day processing.
2	OLAP systems are used by knowledge workers such as executives, managers, and analysts.	OLTP systems are used by clerks, DBAs, or database professionals.
3	It is used to analyze the business.	It is used to run the business.
4	It focuses on Information out.	It focuses on Data in.
5	It is based on Star Schema, Snowflake Schema, and Fact Constellation Schema.	It is based on Entity Relationship Model.
6	It focuses on Information out.	It is application oriented.
7	It contains historical data.	It contains current data.
8	It provides summarized and consolidated data.	It provides primitive and highly detailed data.
9	It provides summarized and multidimensional view of data.	It provides detailed and flat relational view of data.
10	The number of users is in hundreds.	The number of users is in thousands.
11	The number of records accessed is in millions.	The number of records accessed is in tens.
12	The database size is from 100GB to 100 TB.	The database size is from 100 MB to 100 GB.
13	These are highly flexible.	It provides high performance.

Data Warehousing Architecture

Components of datawarehouse



Metadata

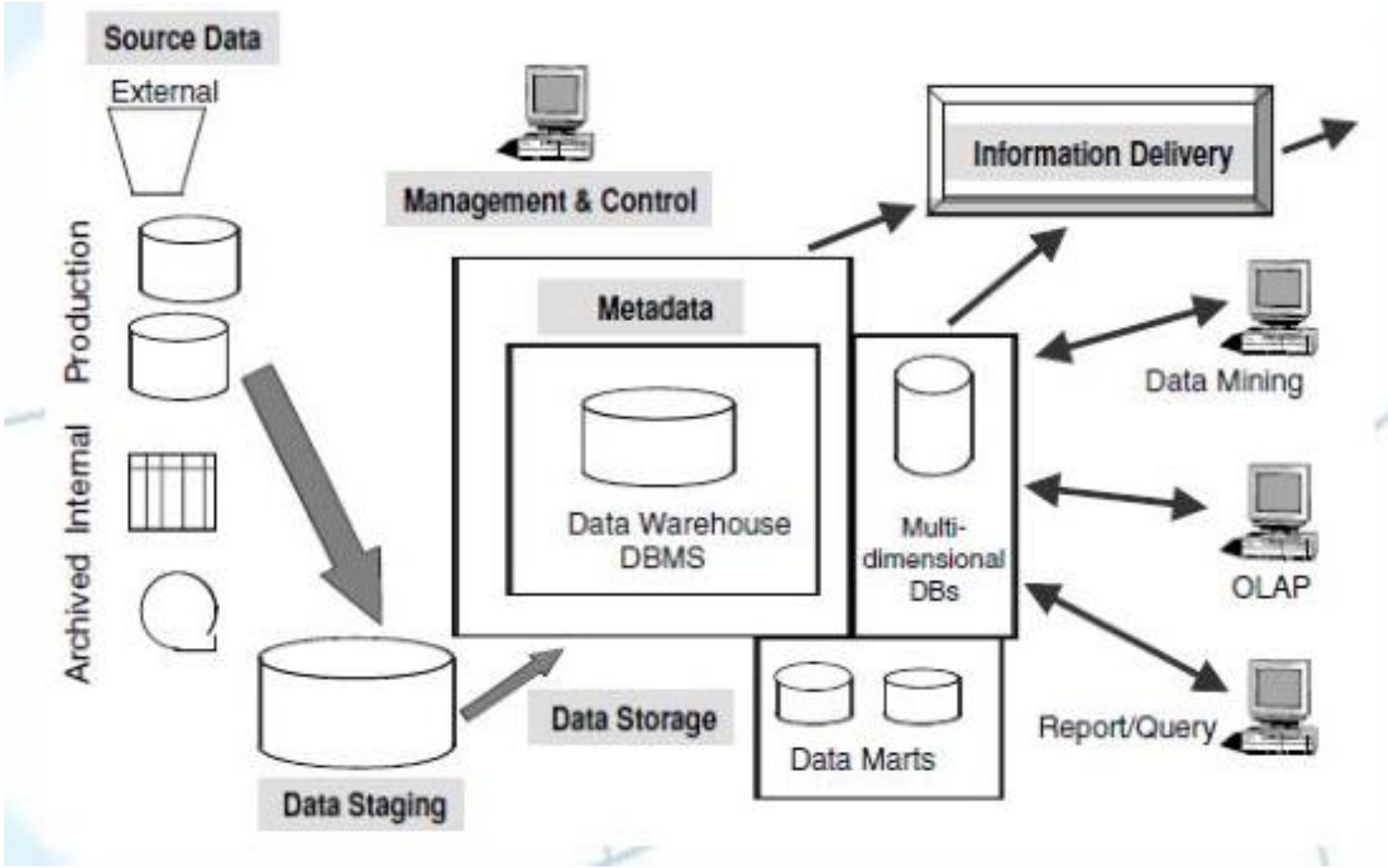
Source Data

Data Staging

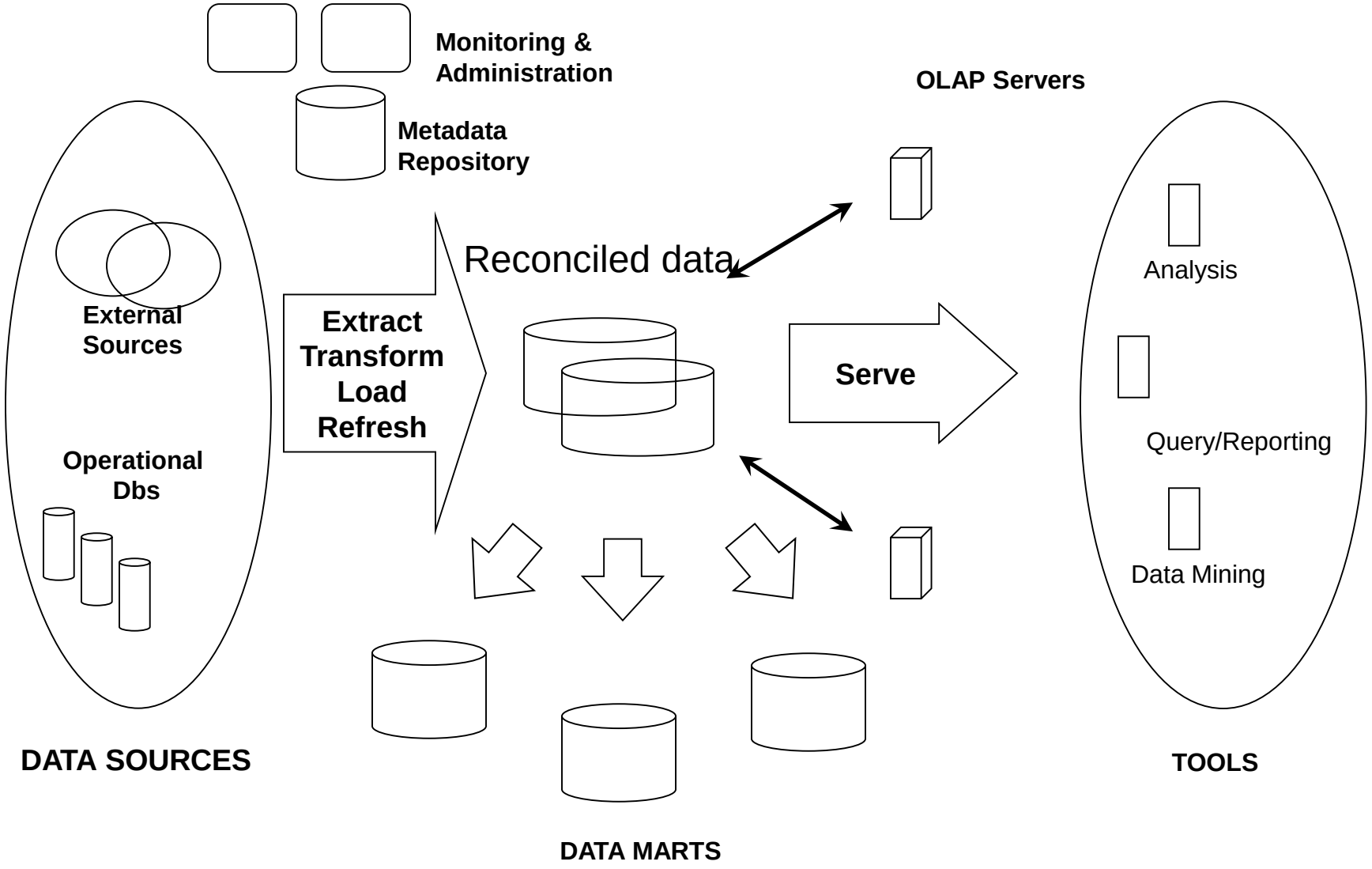
Data Storage

Information delivery

Data Warehousing Architecture



Data Warehousing Architecture

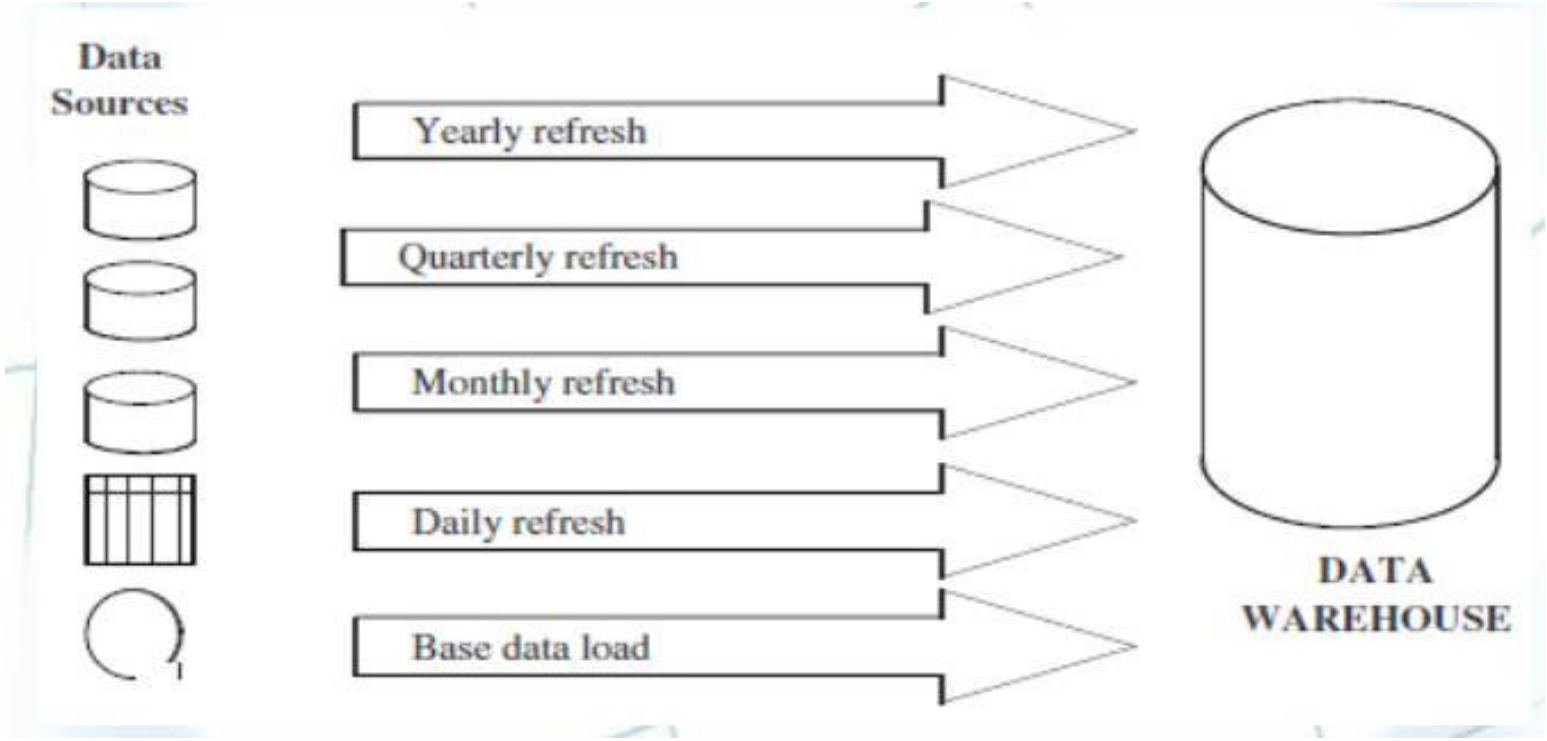


Source Data Component

- Source data coming into the data warehouse may be grouped into four broad categories, as following:
  - **Production Data:** the various operational systems of the enterprise
  - **Internal Data**:"private" spreadsheets, documents, customer profiles, and sometimes even departmental databases.
  - **Archived Data:**
  - **External Data:**Most executives depend on data from external sources for a high percentage of the information they use. They use statistics relating to their industry produced by external agencies. They use market share data of competitors. They use standard values of financial indicators for their business to check on their performance.

Data Staging Component

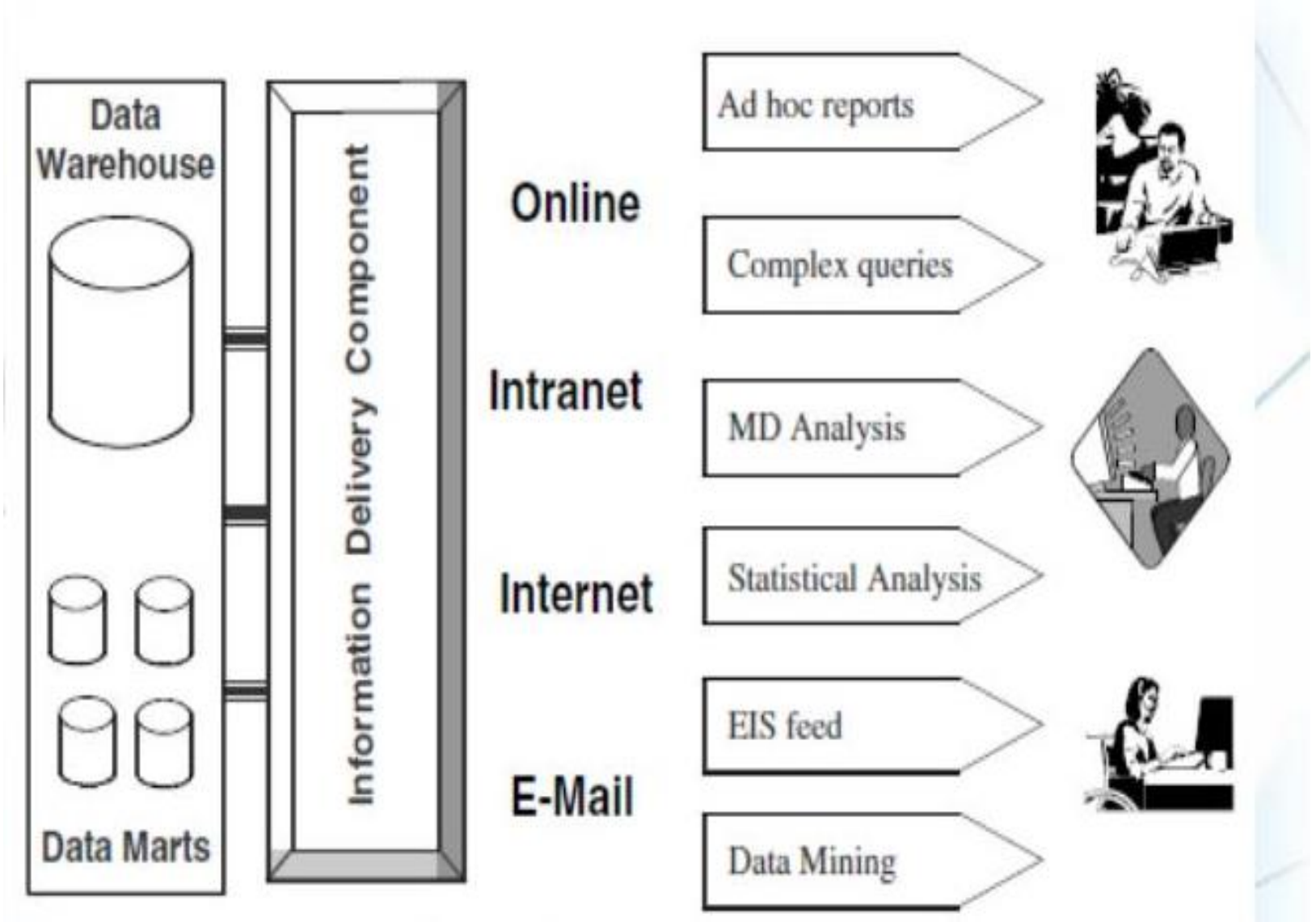
- The extracted data coming from several disparate sources needs to be changed, converted, and made ready in a format that is suitable to be stored for querying and analysis.
- Three major functions need to be performed for getting the data ready.
  - You have to extract the data,
  - transform the data,
  - and then load the data into the data warehouse storage.
- These three major functions of extraction, transformation, and preparation for loading take place in a staging



common types of data movements from the staging area to the data warehouse storage

Data Storage Component

- The data storage for the data warehouse is a separate repository.
- large volumes of historical data for analysis.
- keep the data in the data warehouse in structures suitable for analysis, and not for quick retrieval of individual pieces of information.
- When your analysts use the data in the data warehouse for analysis, they need to know that the data is stable and that it represents snapshots at specified periods.
- the data warehouses are “read-only” data repositories.



Information Delivery Component

Information Delivery Component

- Who are the users that need information from the data warehouse?
- In order to provide information to the wide community of data warehouse users, the information delivery component includes different methods of information delivery.
- The different information delivery methods.
  - Ad hoc reports are predefined reports primarily meant for novice and casual users.
  - Provision for complex queries,
  - multidimensional (MD) analysis, and
  - statistical analysis cater to the needs of the business analysts and power users.
  - Information fed into Executive Information Systems (EIS) is meant for senior executives and high-level managers.
  - Some data warehouses also provide data to data-mining applications.

Metadata Component

- Metadata in a data warehouse is similar to the data dictionary or the data catalog in a database management system.
- In the data dictionary, you keep the information about the logical data structures, the information about the files and addresses, the information about the indexes, and so on. The data dictionary contains data about the data in the database.
- Similarly, the metadata component is the data about the data in the data warehouse.

Management and Control Component

- This component of the data warehouse architecture sits on top of all the other components.
- Coordinates the services and activities within the data warehouse.
- Controls the data transformation and the data transfer into the data warehouse storage.
- Moderates the information delivery to the users.
- Works with the database management systems and enables data to be Properly stored in the repositories.
- Monitors the movement of data into the staging area and from there into the data warehouse storage itself.
- Interacts with the metadata component to perform the management and control functions.

Dimensional Modeling

Dimensional Modeling

- **Dimensional modeling** (DM) names a set of techniques and concepts used in data warehouse design.
- Dimensional modeling is one of the methods of data modeling, that help us store the data in such a way that it is relatively easy to retrieve the data from the database.
- Dimensional modeling always uses the concepts of facts (measures), and dimensions (context).

Entity-Relationship vs. Dimensional Models

Relational Modeling	Dimensional Modeling
Data is stored in RDBMS	Data is stored in RDBMS or Multidimensional databases
Tables are units of storage	Cubes are units of storage
Data is normalized and used for OLTP. Optimized for OLTP processing	Data is de normalized and used in data warehouse and data mart. Optimized for OLAP
Several tables and chains of relationships among them	Few tables and fact tables are connected to dimensional tables
Volatile (several updates) and time variant Detailed level of transactional data	Non volatile and time invariant Summary of bulky transactional data (Aggregates and Measures) used in business decisions
SQL is used to manipulate data Normal Reports	MDX is used to manipulate data User friendly, interactive, drag and drop multidimensional OLAP Reports

Entity-Relationship vs. Dimensional Models

- One table per entity
- Minimize data redundancy
- Optimize update
- The Transaction Processing Model

- One fact table for data organization
- Maximize understandability
- Optimized for retrieval
- The data warehousing model

Facts & Dimensions

- There are two main types of objects in a dimensional model
  - **Facts** are quantitative measures that we wish to analyse and report on.
  - **Dimensions** contain textual descriptors of the business. They provide *context* for the facts.

Dimensional Modeling

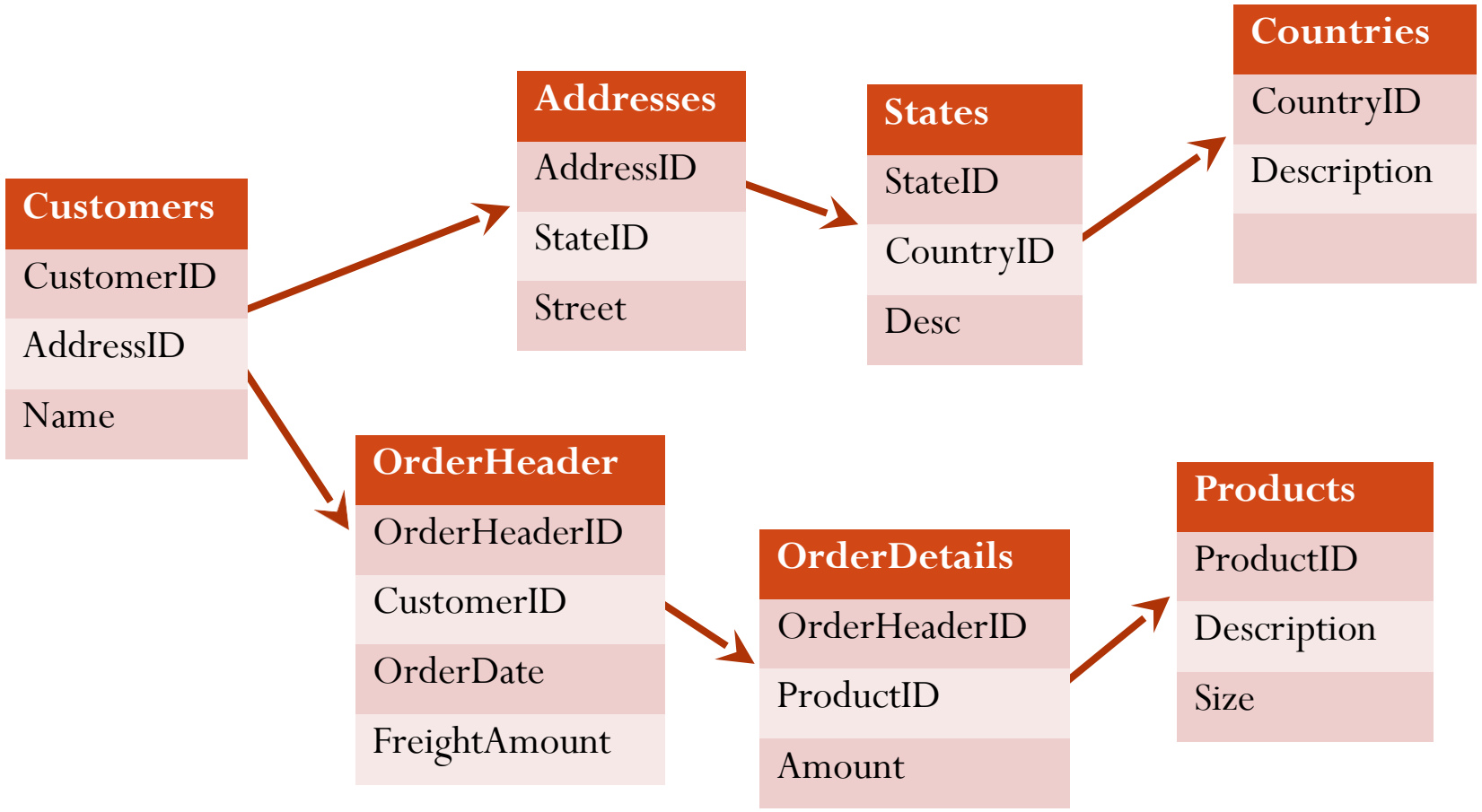
- Models the data around two basic concepts: Facts & Dimensions.
- Facts
  - Facts are numeric measurements (values) that represent a specific business aspect or activity.
  - Facts can be computed or derived at run-time (metrics).
  - Examples : Unit Cost, Sale Amount, Quantity Sold
- Dimensions
  - Dimensions are qualifying characteristics that provide additional perspectives to a given fact.
  - Examples: Date (Day, Month, Qtr,Year), Product (Type, Category)

Dimensional Modeling (Contd.)

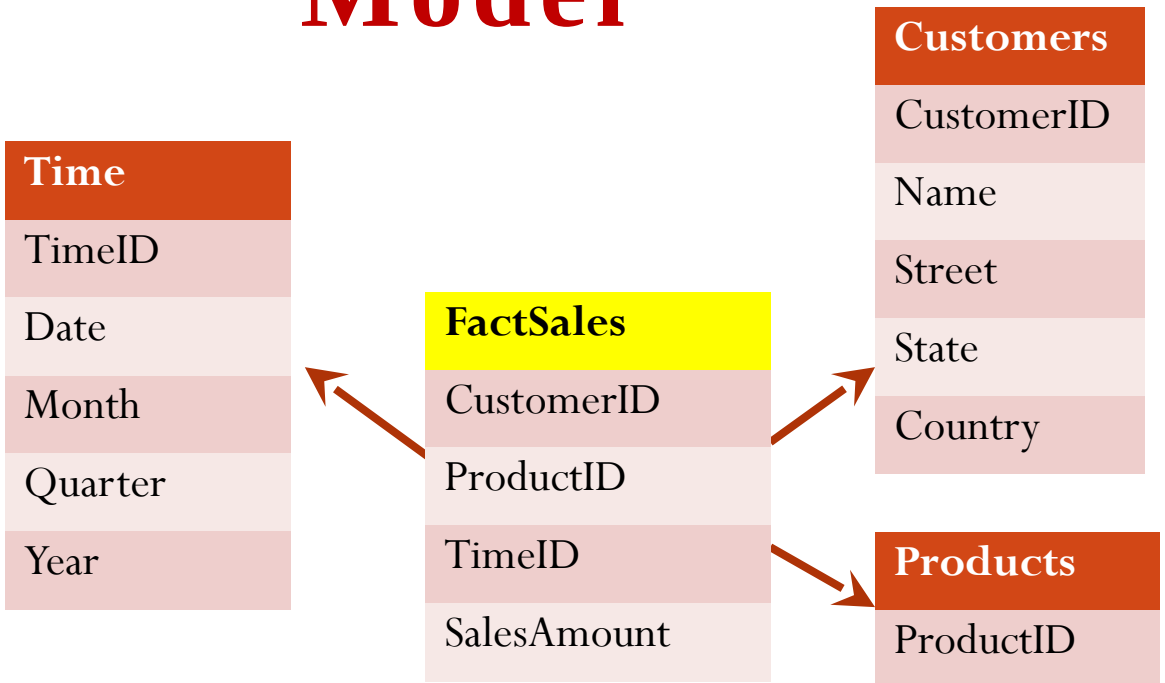
- Every dimensional model (DM) is composed of one (or more) fact tables, and a set of smaller dimension tables.

- Forms ‘star-like’ structure, which is called a star schema or star join.

A Transactional Database



A Dimensional Model



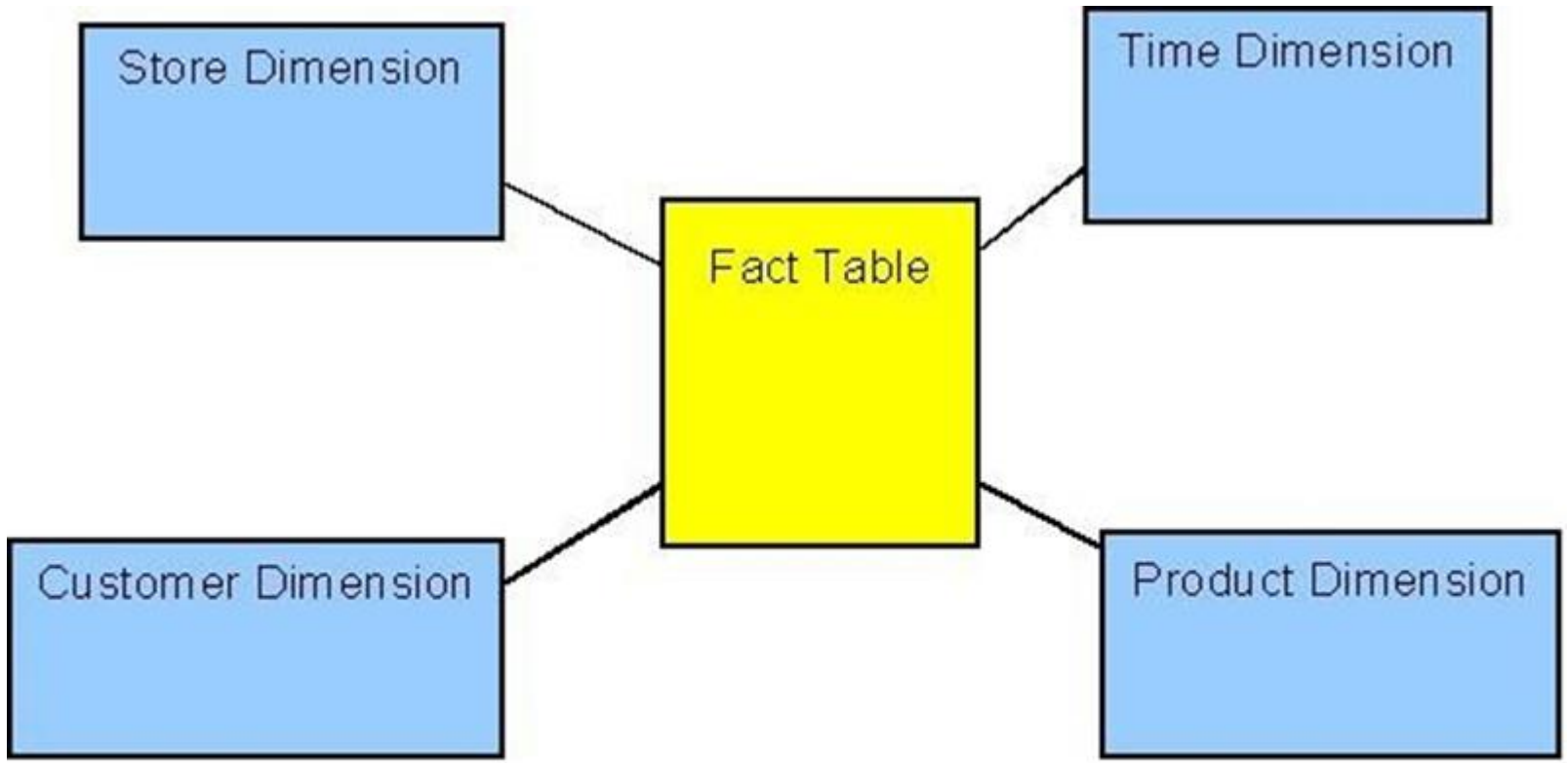
Look on Fact table through one (or more) dimensions.

What is the sale amount in Consumer Product category, for customers who live in Maharashtra and in the second quarter of 2004?

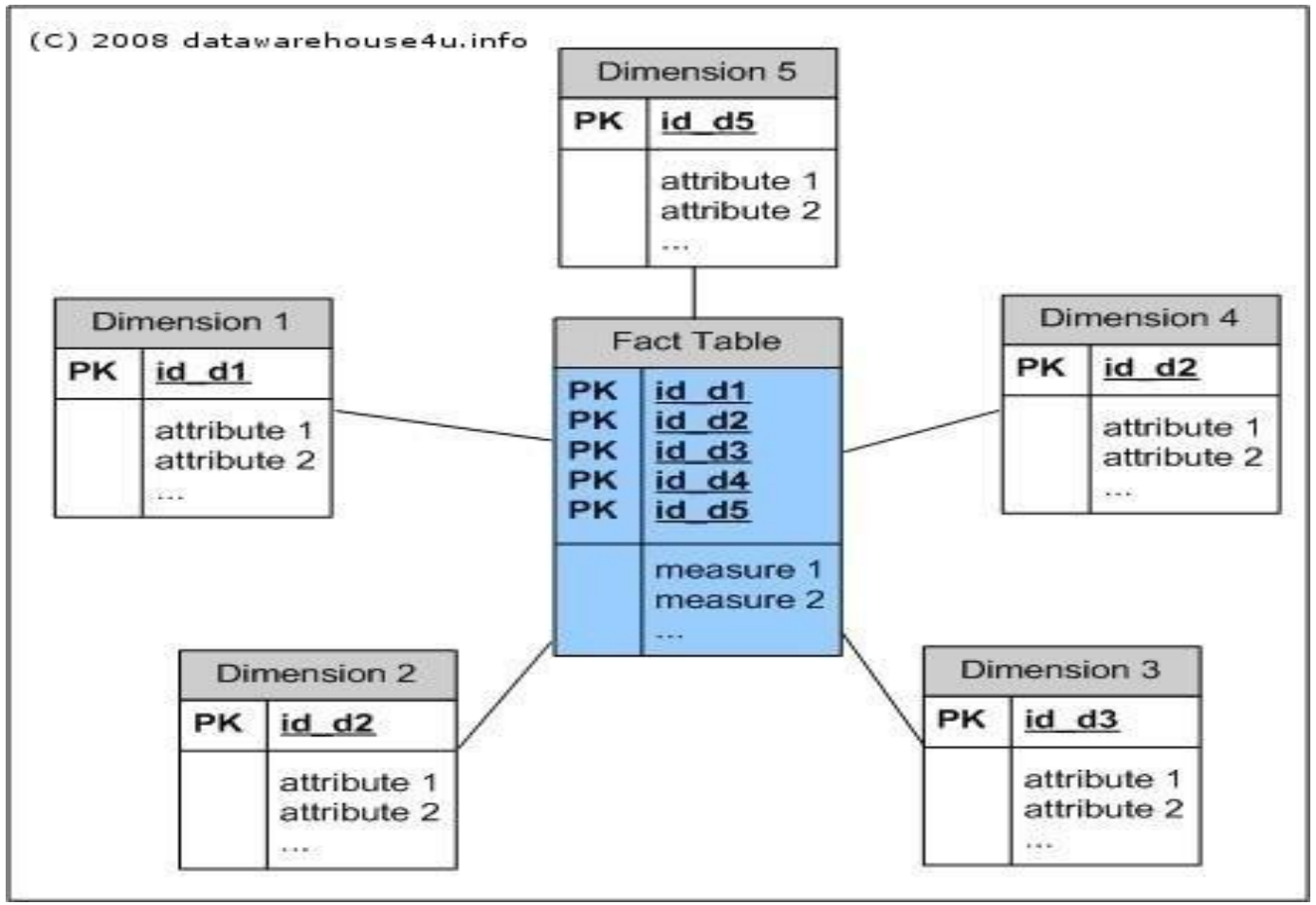
Star schema

- The star schema architecture is the simplest data warehouse schema.
- It is called a star schema because the diagram resembles a star with points radiating from a center.
- The center of the star consists of fact table and the points of the star are the dimension tables.
- Usually the fact tables in a star schema are in third normal form(3NF) whereas dimensional tables are de-normalized.

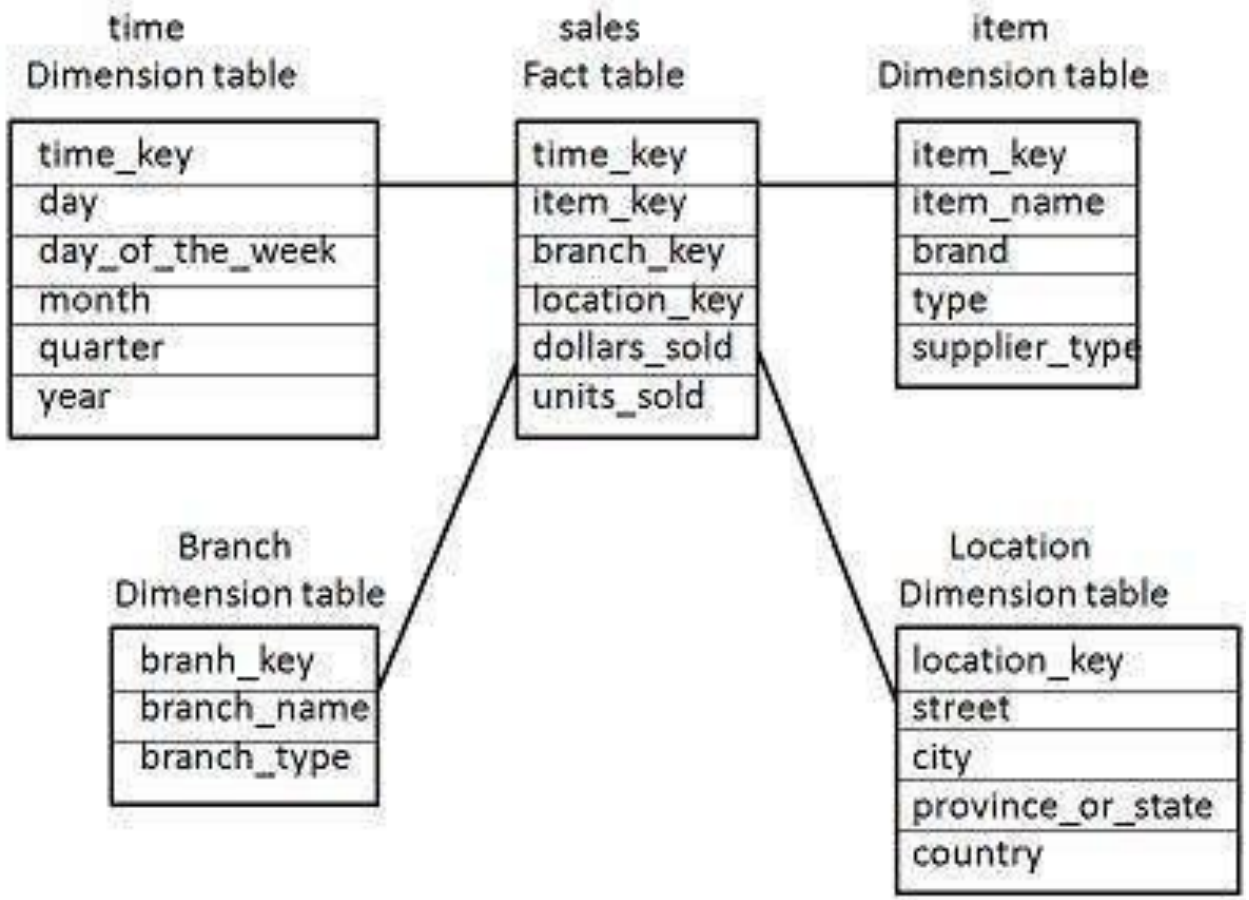
Star schema


Store Dimension
Time Dimension
Fact Table
Customer Dimension
Product Dimension

Star schema


(C) 2008 datawarehouse4u.info
Dimension 5
PK
id_d5
attribute 1
attribute 2
...
Dimension 1
PK
id_d1
attribute 1
attribute 2
...
Fact Table
PK
id_d1
PK
id_d2
PK
id_d3
PK
id_d4
PK
id_d5
measure 1
measure 2
...
Dimension 4
PK
id_d2
attribute 1
attribute 2
...
Dimension 2
PK
id_d2
attribute 1
attribute 2
...
Dimension 3
PK
id_d3
attribute 1
attribute 2
...

Star schema

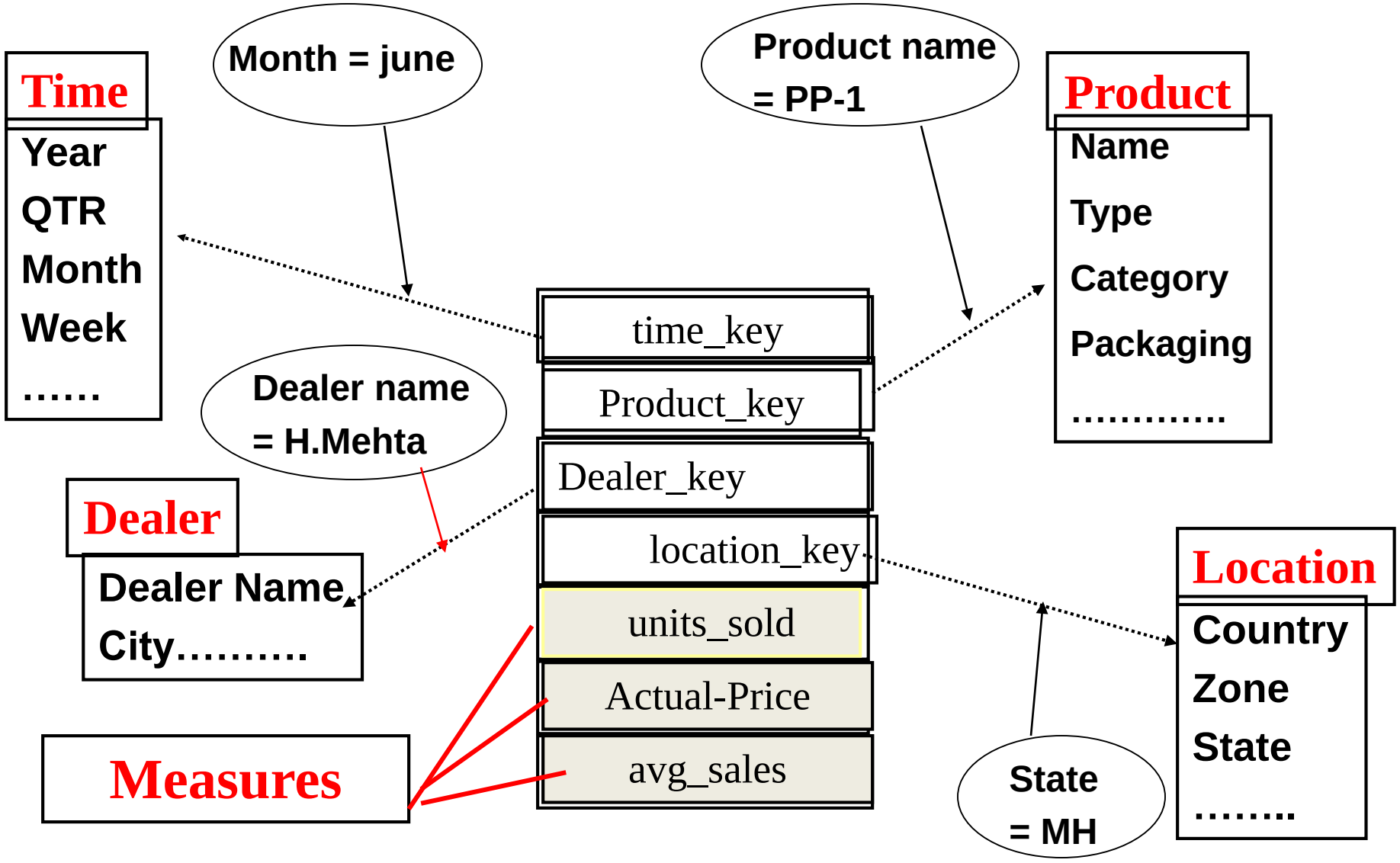

time
Dimension table
time_key
day
day_of_the_week
month
quarter
year
sales
Fact table
time_key
item_key
branch_key
location_key
dollars_sold
units_sold
item
Dimension table
item_key
item_name
brand
type
supplier_type
Branch
Dimension table
branh_key
branch_name
branch_type
Location
Dimension table
location_key
street
city
province_or_state
country

Querying Against a Star Schema

- Say you want to find sales during **june 2006**
  - Go to the **Time dim table**,
  - find year **2006** and month **june**,
  - extract the **keys** for all such records
  - Use this set of keys to query the
  - **Fact table**

- **Thus the dimension tables act as filters into the fact table**

Querying Against a Star Schema



Example

The business objective is to create a data model that can store and report number of burgers and fries sold from a specific McDonalds outlet per day.

- **Step 1: Identify the dimensions**
- Dimensions are the object or context.
- dimensions are the 'things' about which something is being spoken.
- 3 different things - we are speaking about some "food", some specific McDonalds "store" and some specific "day". So we have 3 dimensions - "food" (e.g. burgers and fries), "store" and "day".
- Burgers and fries are 2 different members of "food" dimension.
- create separate tables for separate dimensions.

- **Step 2: Identify the measures**
- Measures are the quantifiable subjects and these are often numeric in nature.
- the number of burgers/fries sold is a measure.
- Measures are not stored in the dimension tables.
- A separate table is created for storing measures.
- This table is called Fact Table.

Step 3: Identify the attributes or properties of dimensions

different attributes of food - e.g. names of the food, price of the food, total calories in the food, color of the food and so on

Food

KEY	NAME
1	Burger
2	Fries

Store

KEY	NAME
1	Store 1
2	Store 2
…	…

Similarly, the structure of our store and day dimensions will be like this: **Store**

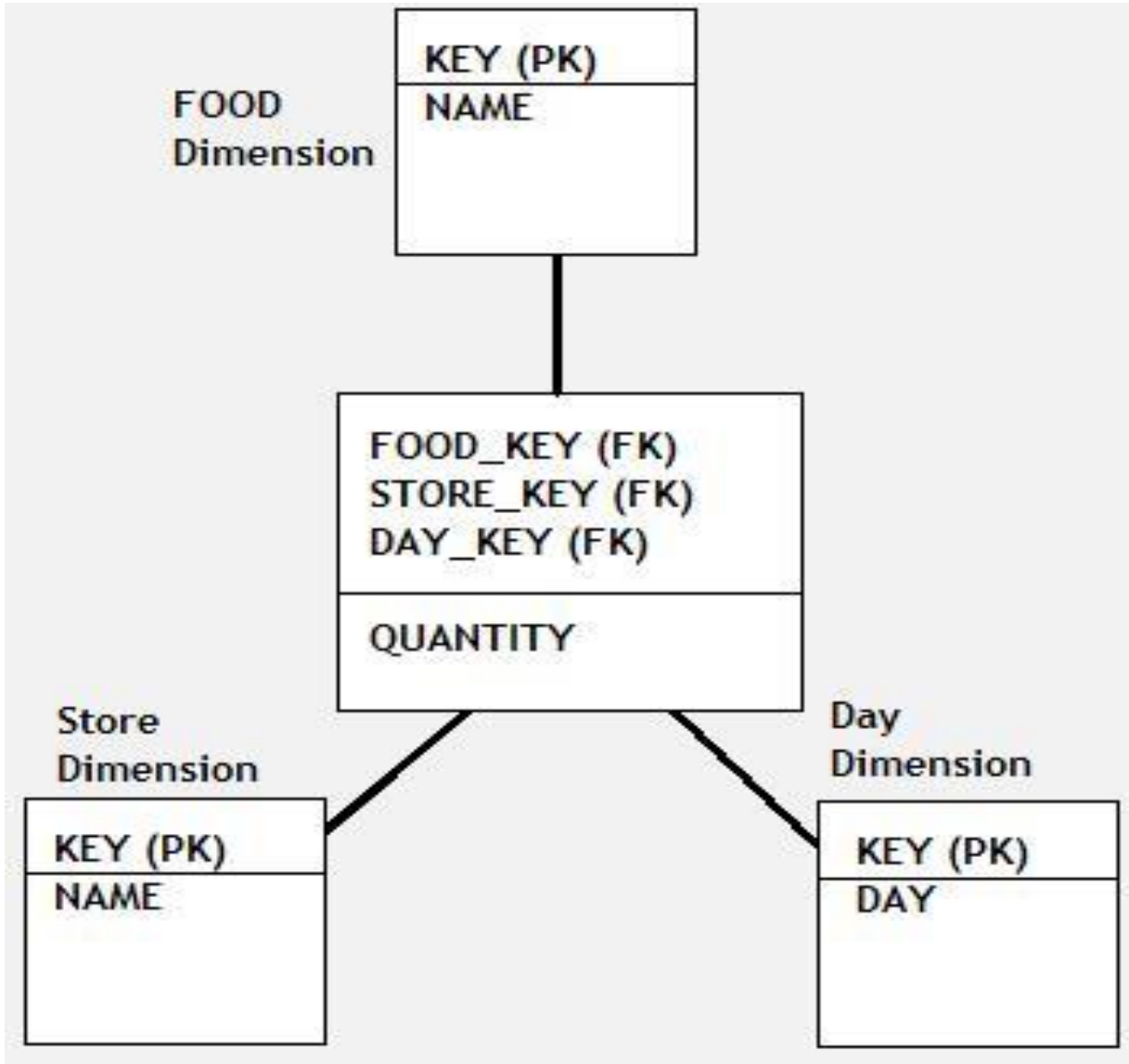
Day

KEY	DAY
1	01 Jan 2012
2	02 Jan 2012
3	03 Jan 2012
…	…

Step 4: Identify the granularity of the measures

- Granularity refers to the lowest (or most granular) level of information stored in any table.
- If a table contains sales data for each and every day, then it has a daily granularity.
- If a table contains total sales data for each month, then it has monthly granularity

The grain conveys the level of detail that is associated with the fact table measurements.


FOOD
Dimension
KEY (PK)
NAME
FOOD_KEY (FK)
STORE_KEY (FK)
DAY_KEY (FK)
QUANTITY
Store
Dimension
KEY (PK)
NAME
Day
Dimension
KEY (PK)
DAY

- As your dimension tables start growing in terms of number of attributes as well as data, you may want to normalize the data then,

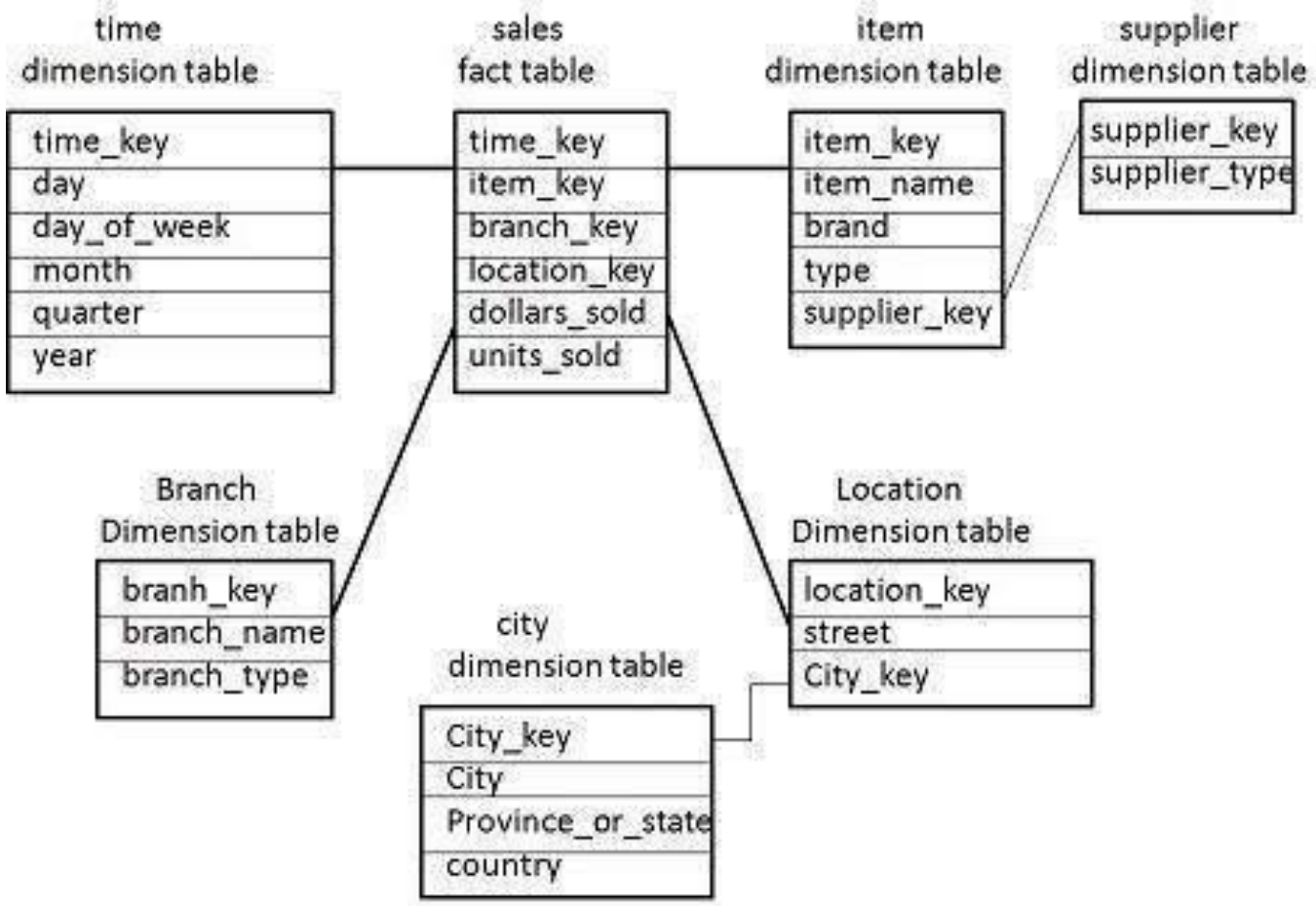
?

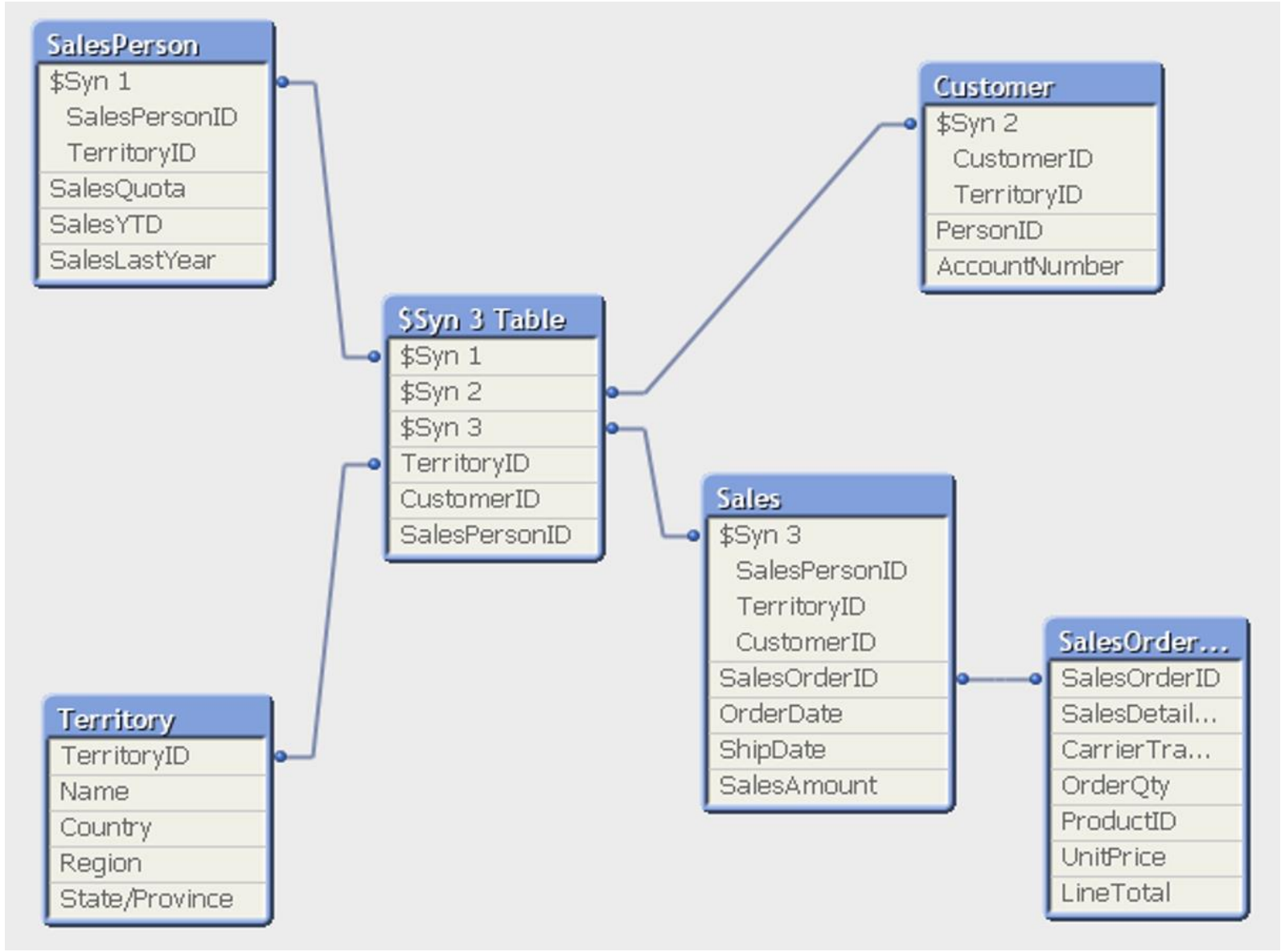
Snowflake Schema

Snowflake Schema

- Some dimension tables in the Snowflake schema are normalized.
- The normalization splits up the data into additional tables.
- Unlike Star schema, the dimensions table in a snowflake schema are normalized.
- For example, the item dimension table in star schema is normalized and split into two dimension tables, namely item and supplier table.

Snowflake Schema




SalesPerson
$Syn 1
SalesPersonID
TerritoryID
SalesQuota
SalesYTD
SalesLastYear
Customer
$Syn 2
CustomerID
TerritoryID
PersonID
AccountNumber
$Syn 3 Table
$Syn 1
$Syn 2
$Syn 3
TerritoryID
CustomerID
SalesPersonID
Sales
$Syn 3
SalesPersonID
TerritoryID
CustomerID
SalesOrderID
OrderDate
ShipDate
SalesAmount
SalesOrder...
SalesOrderID
SalesDetail...
CarrierTra...
OrderQty
ProductID
UnitPrice
LineTotal
Territory
TerritoryID
Name
Country
Region
State/Province

Consider a data warehouse for a hospital, where there are three dimensions:

(i) Doctor

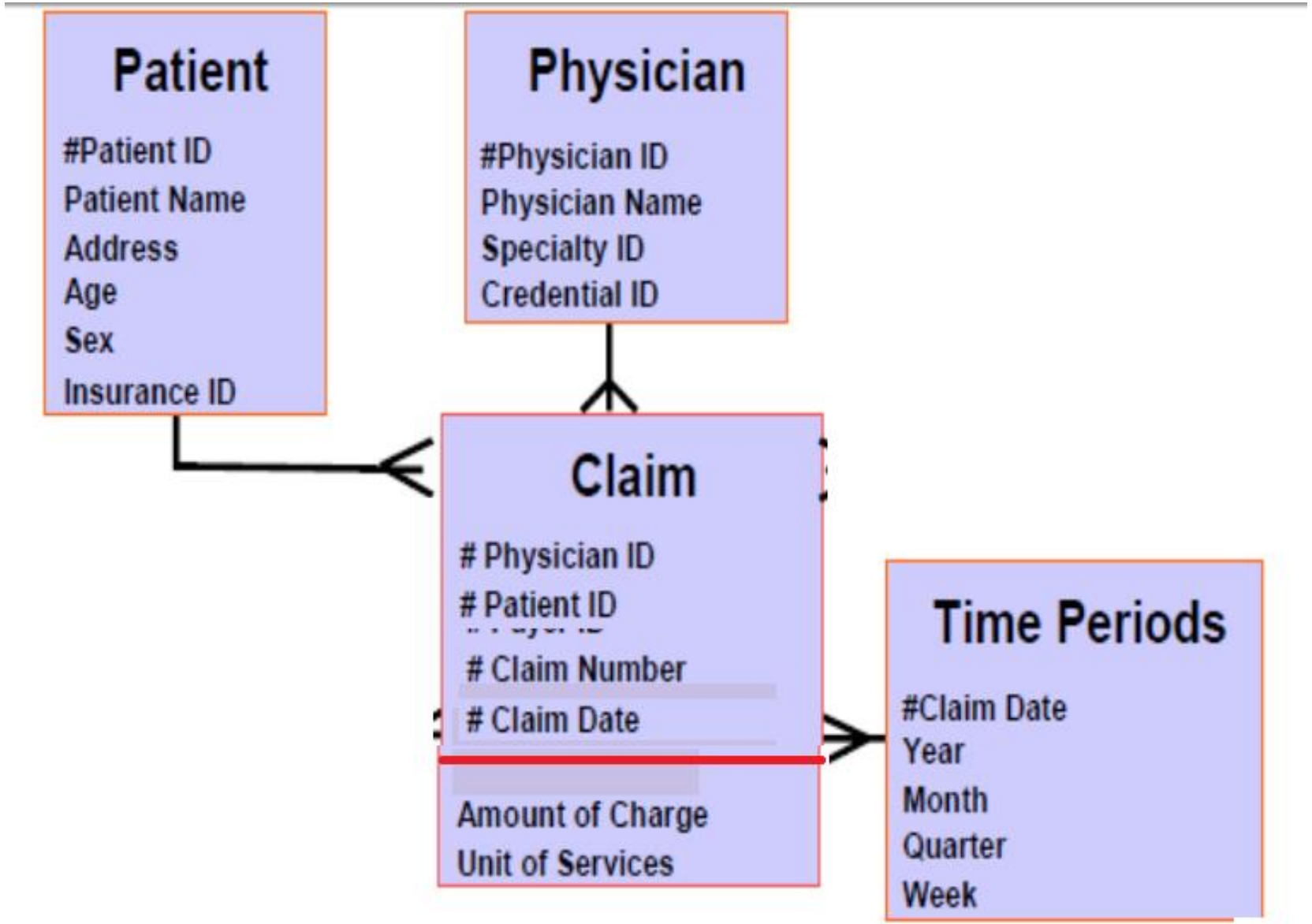
(ii) Patient and

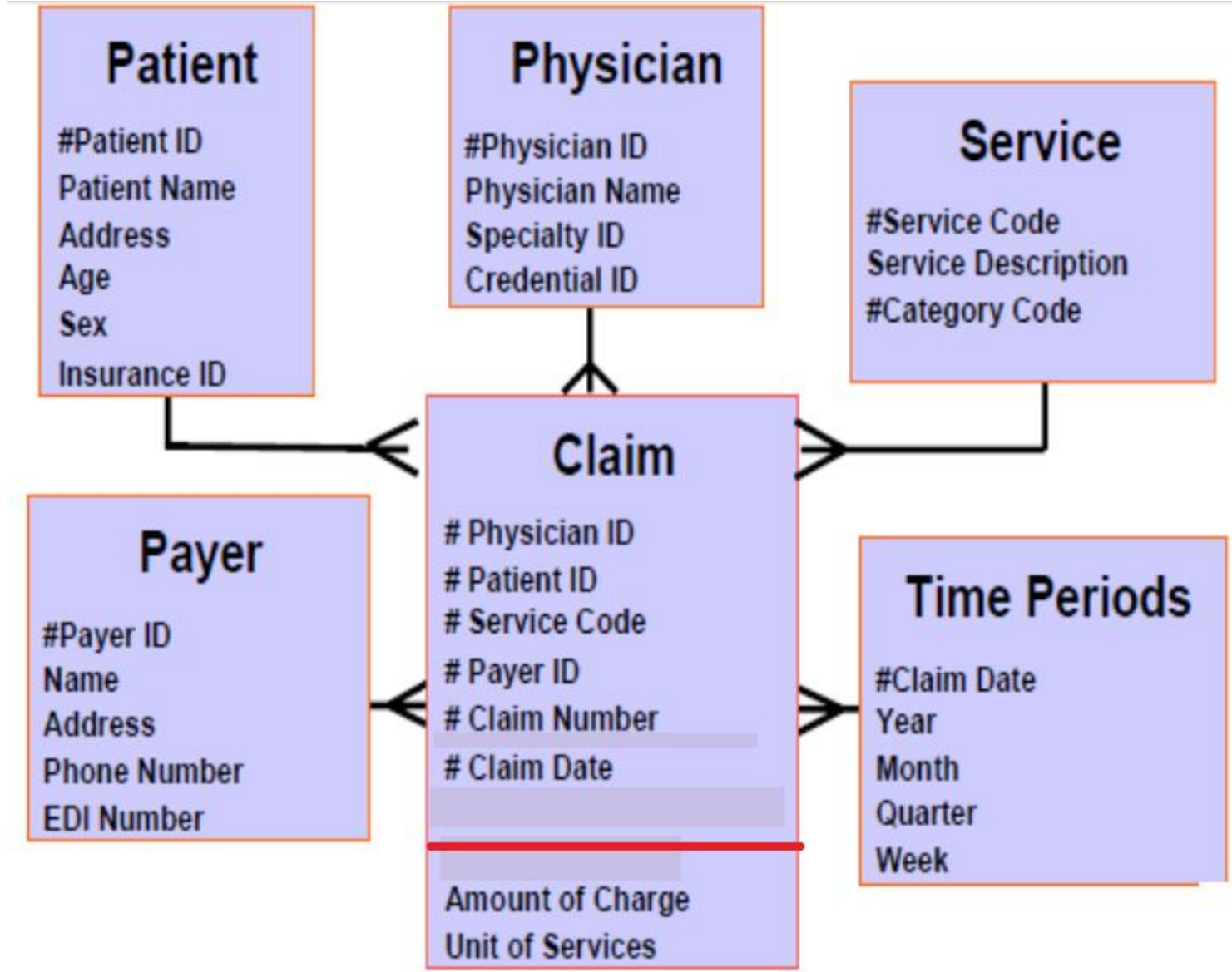
(iii) Time

and two measures:

(i) Count and

(ii) Charge.

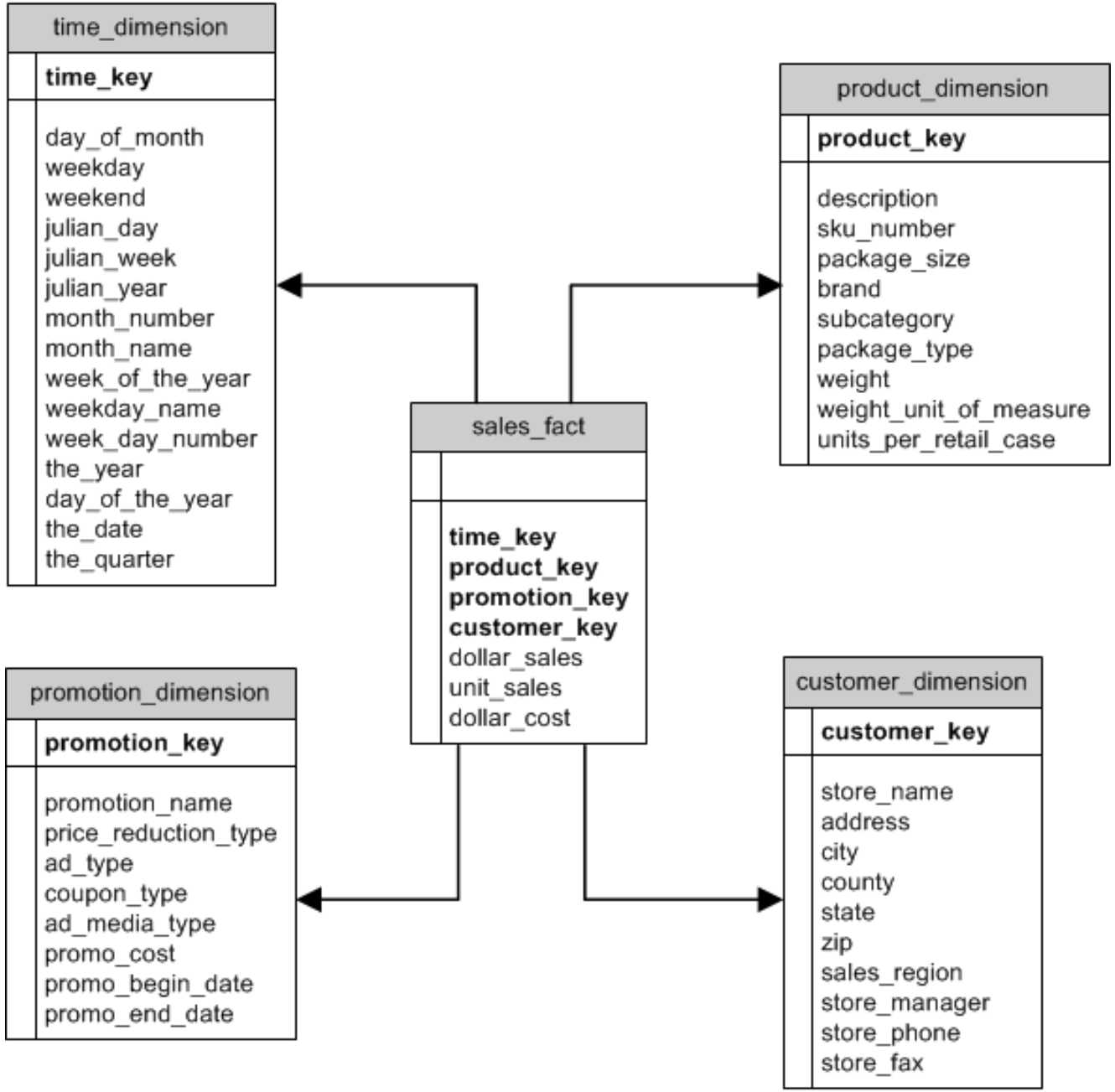

Patient
#Patient ID
Patient Name
Address
Age
Sex
Insurance ID
Physician
#Physician ID
Physician Name
Specialty ID
Credential ID
Claim
Physician ID
Patient ID
Claim Number
Claim Date
Amount of Charge
Unit of Services
Time Periods
#Claim Date
Year
Month
Quarter
Week


Patient
#Patient ID
Patient Name
Address
Age
Sex
Insurance ID
Physician
#Physician ID
Physician Name
Specialty ID
Credential ID
Service
#Service Code
Service Description
#Category Code
Claim
Physician ID
Patient ID
Service Code
Payer ID
Claim Number
Claim Date
Amount of Charge
Unit of Services
Payer
#Payer ID
Name
Address
Phone Number
EDI Number
Time Periods
#Claim Date
Year
Month
Quarter
Week

Examples of facts in different industries include:

• Retail -- number of units sold, sales amount

Examples of dimensions in different industries include:
• Retail -- store name, zip code, product name, product category, day of week
•


time_dimension
time_key
day_of_month
weekday
weekend
julian_day
julian_week
julian_year
month_number
month_name
week_of_the_year
weekday_name
week_day_number
the_year
day_of_the_year
the_date
the_quarter
product_dimension
product_key
description
sku_number
package_size
brand
subcategory
package_type
weight
weight_unit_of_measure
units_per_retail_case
sales_fact
time_key
product_key
promotion_key
customer_key
dollar_sales
unit_sales
dollar_cost
promotion_dimension
promotion_key
promotion_name
price_reduction_type
ad_type
coupon_type
ad_media_type
promo_cost
promo_begin_date
promo_end_date
customer_dimension
customer_key
store_name
address
city
county
state
zip
sales_region
store_manager
store_phone
store_fax

Examples of facts in different industries include:

• Telecommunications -- length of call in minutes, average number of calls

Examples of dimensions in different industries include:
• Telecommunications -- call origin, call destination

Examples of facts in different industries include:

• Banking -- average monthly balance

Examples of dimensions in different industries include:
• Banking -- customer name, account number, branch, account officer

Examples of facts in different industries include:

- Insurance -- claims amount

Examples of dimensions in different industries include:

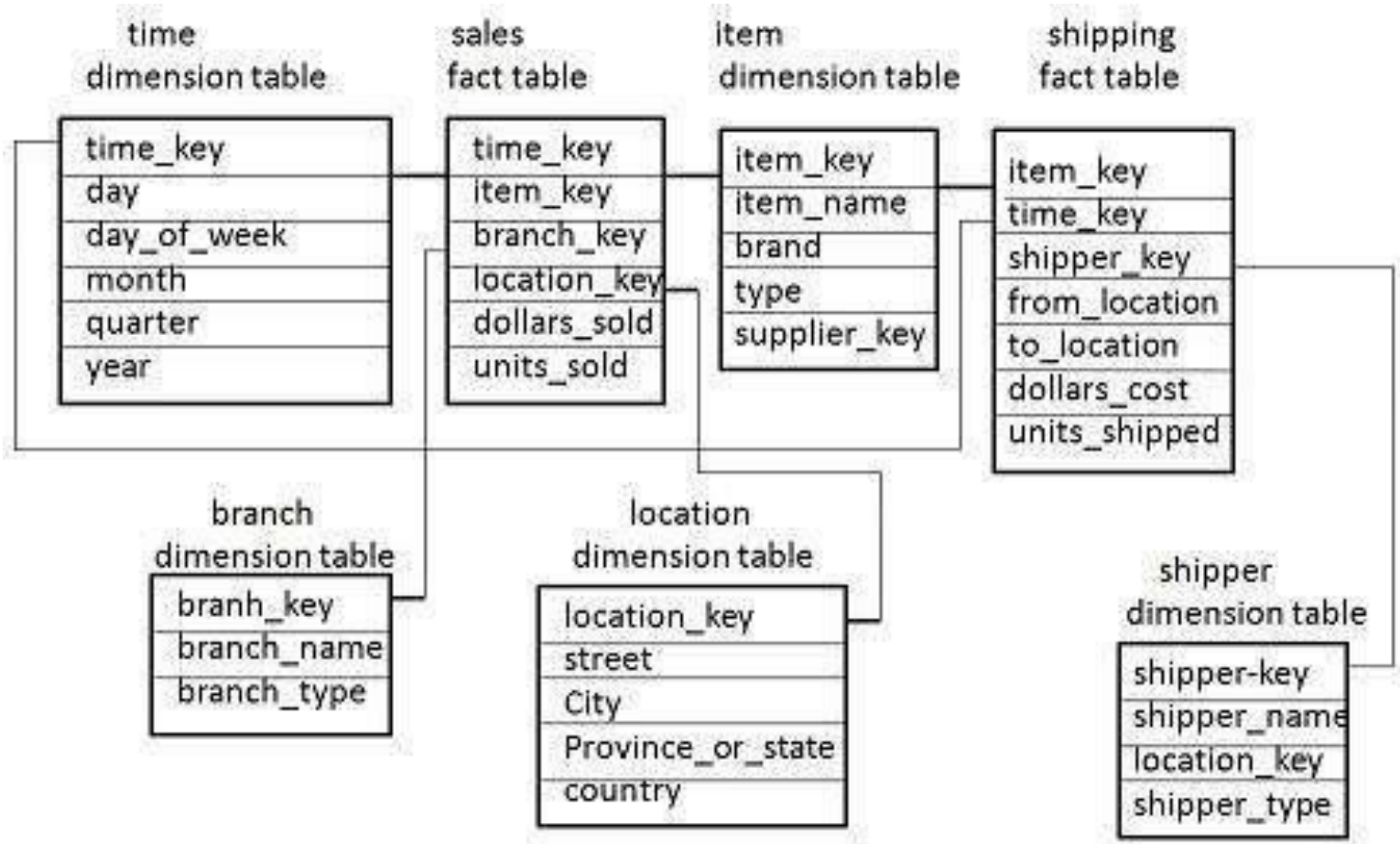
- Insurance -- policy type, insured party

Information Package Diagram

Hierarchies/Categories	Dimensions			
	Time	**Locations**	**Products**	…
	T_ID	Lc_Id	Prod_ID	
	Day	City	Prod_Name	
	Week	State	Prod_Category	
	Quarter	Country	Prod_Subcategory	
	Measures/Facts: Sales_Amount, No. of products sold			

Fact Constellation Schema

- A fact constellation has multiple fact tables. It is also known as galaxy schema.
- The following diagram shows two fact tables, namely sales and shipping.
- It is also possible to share dimension tables between fact tables. For example, time, item, and location dimension tables are shared between the sales and shipping fact table.


time
dimension table
time_key
day
day_of_week
month
quarter
year
sales
fact table
time_key
item_key
branch_key
location_key
dollars_sold
units_sold
item
dimension table
item_key
item_name
brand
type
supplier_key
shipping
fact table
item_key
time_key
shipper_key
from_location
to_location
dollars_cost
units_shipped
branch
dimension table
branh_key
branch_name
branch_type
location
dimension table
location_key
street
City
Province_or_state
country
shipper
dimension table
shipper-key
shipper_name
location_key
shipper_type

“Factless” Fact Table

- A ***factless fact table*** is fact table that does not contain fact.
- Table contain only dimensional keys and it captures events that happen only at information level but not included in the calculations level. just an information about an event that happen over a period.

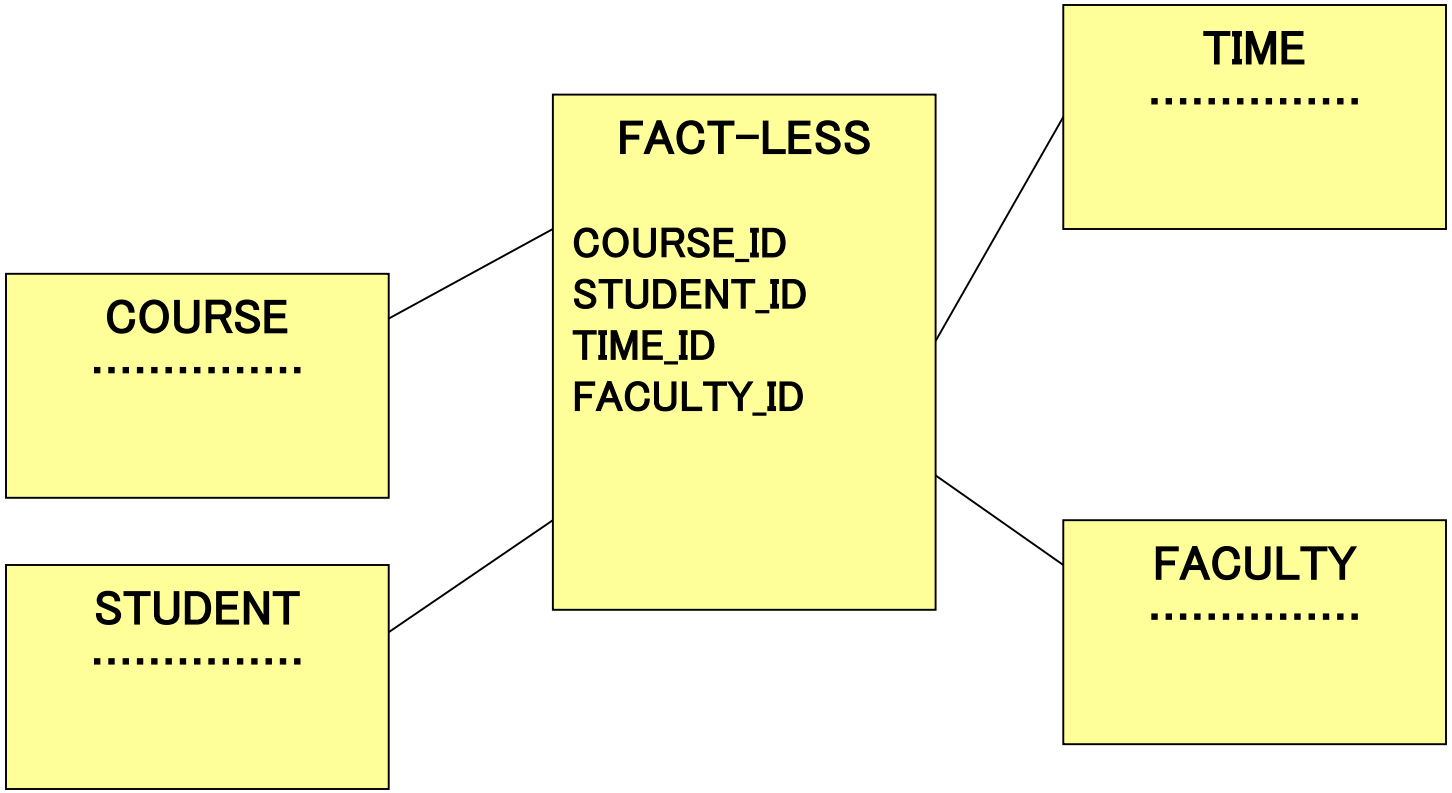
- A factless fact table captures the many-to-many relationships between dimensions, but contains no numeric or textual facts.
- They are often used to record events or coverage information.

Fact-less Fact Tables

- Think of facts as values that arise when a particular combination of dimension instances occur.
  - **e.g. SALE AMOUNT fact arise when a particular CUSTOMER buys a particular PRODUCT on a particular DATE**.
- In some situations, a combination of dimension instance occur without any fact.
- Factless" fact table
  - **A fact table without numeric fact columns**
  - **Used to capture relationships between dimensions**

Fact-less Fact Tables (example)

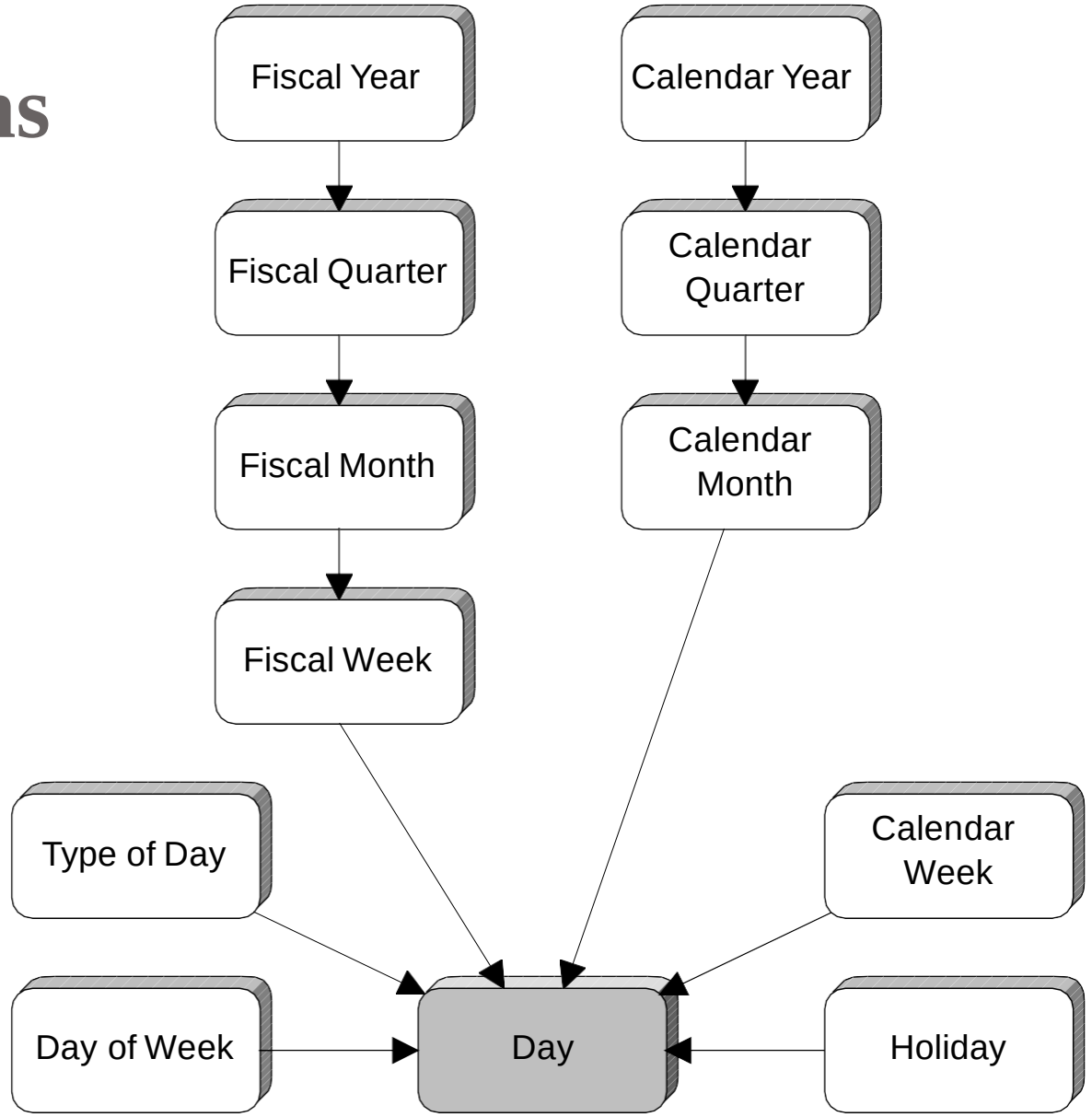
- A particular **STUDENT** attending a particular **COURSE** on a particular **DATE** by a particular **FACULTY MEMBER**.
- What are the facts in this attendance scenario?
- Create a Fact-less fact table for recording this event.



Fact-less Fact Tables (example)

- Which course has maximum number of attendance?
- Which teacher teaches maximum number of students?
- All the above queries are based on the **COUNT (), MAX()** with **GROUP BY**.

Date Dimensions

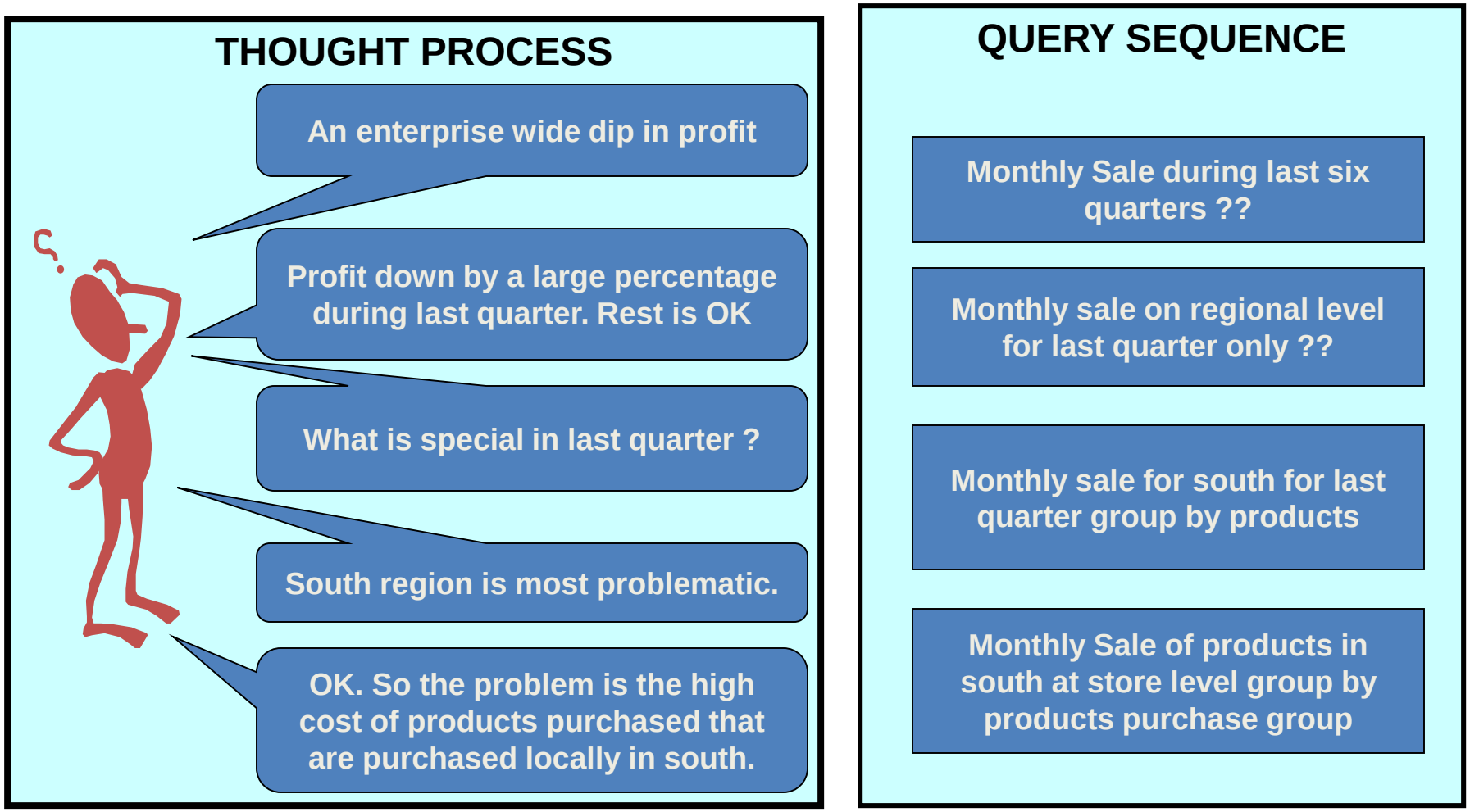


OLAP IN THE DATA WAREHOUSE

What Is OLAP ?

- OLAP is a powerful graphics-oriented tool used to access the data warehouse
- OLAP supports
  - **Business analysis queries**
  - **Data visualization**
  - **Trend analysis**
  - **Scenario analysis**
  - **User defined queries**

Supporting Human Thought Process



- **How many such query sequences you can program beforehand?**

OLAP Is The Answer

- Enables executives , managers to gain useful insights from presentation of data
- Can reorganize metrics along several dimensions and allow data to be viewed from different perspectives
- Supports multidimensional analysis
- Supports drill down, roll up, etc
- Also visual presentation for result comprehension
- Can be implemented on the web
- Highly interactive analysis can be done

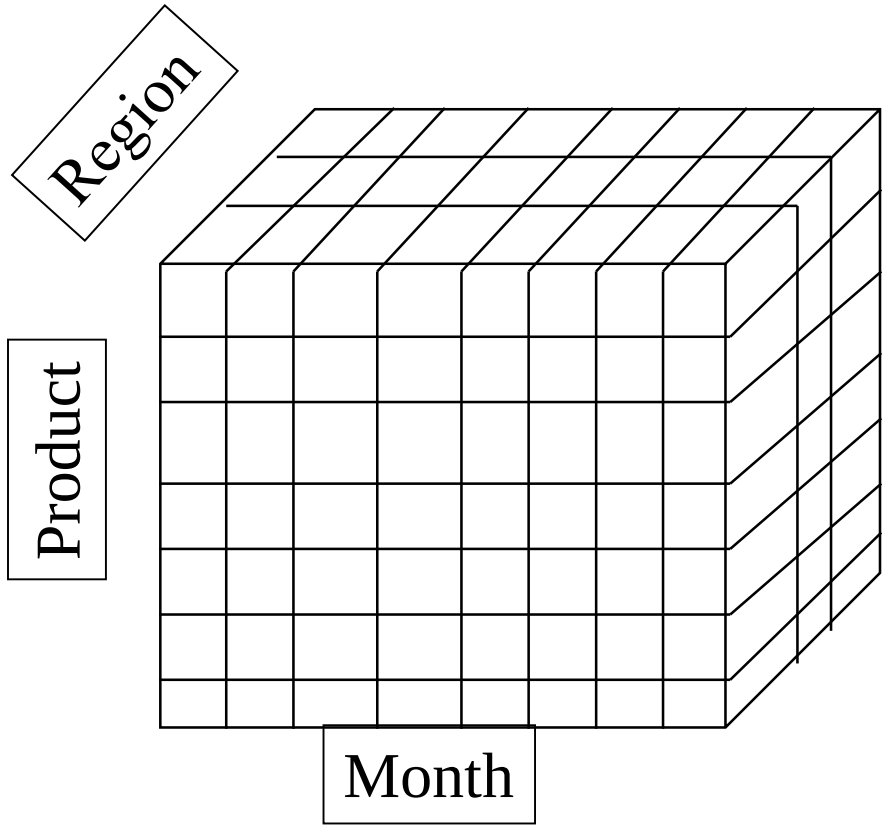
Multidimensional Aggregations At a Particular Level

- A Cube Structure to Handle This
- Sales volume as a function of product, month, and region

Dimensions: Product, Location, Time

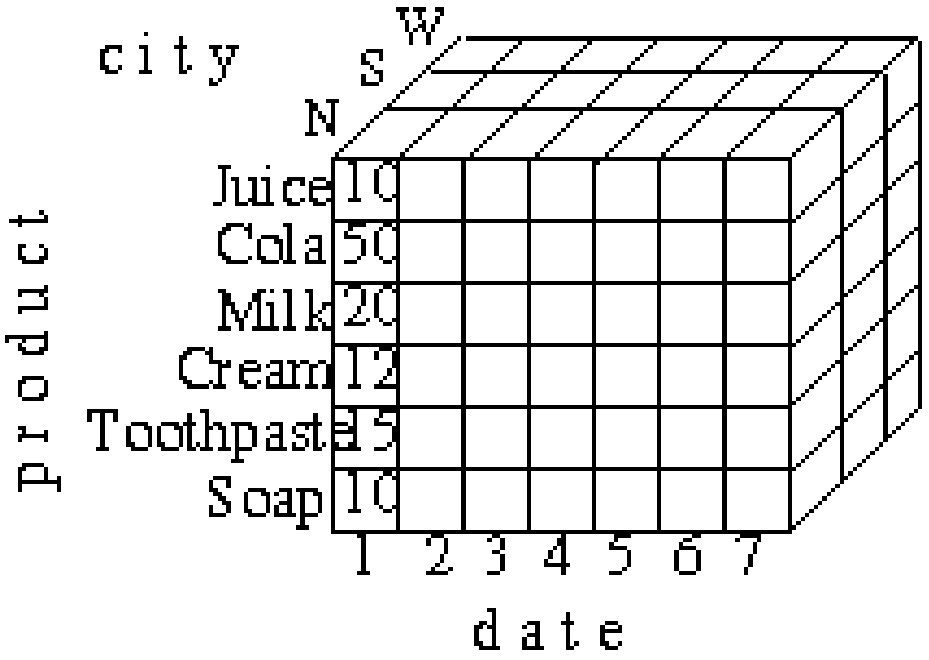
Hierarchical summarization paths

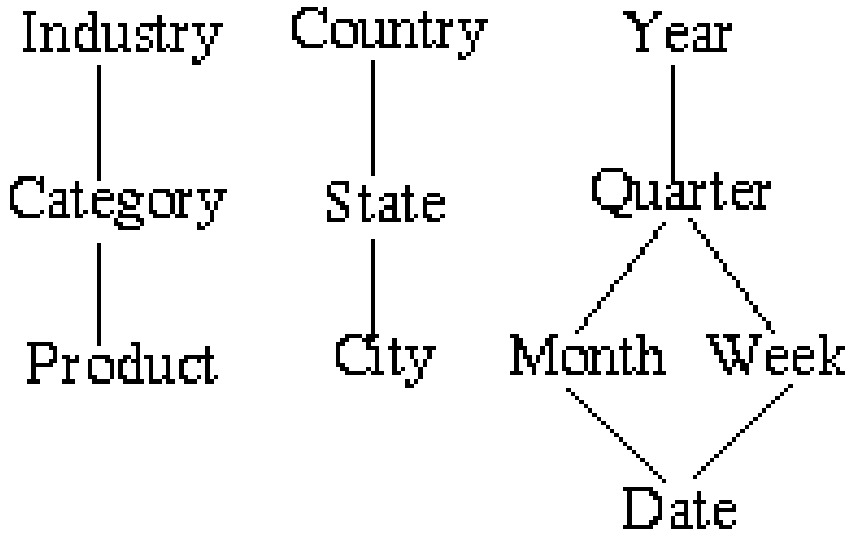
Industry	**Region**	**Year**
Category	**City**	**Quarter**
Product	**Section**	**Month**
	Branch	**Week**
		Day



Multidimensional Data

- Sales volume as a function of product, time, and geography





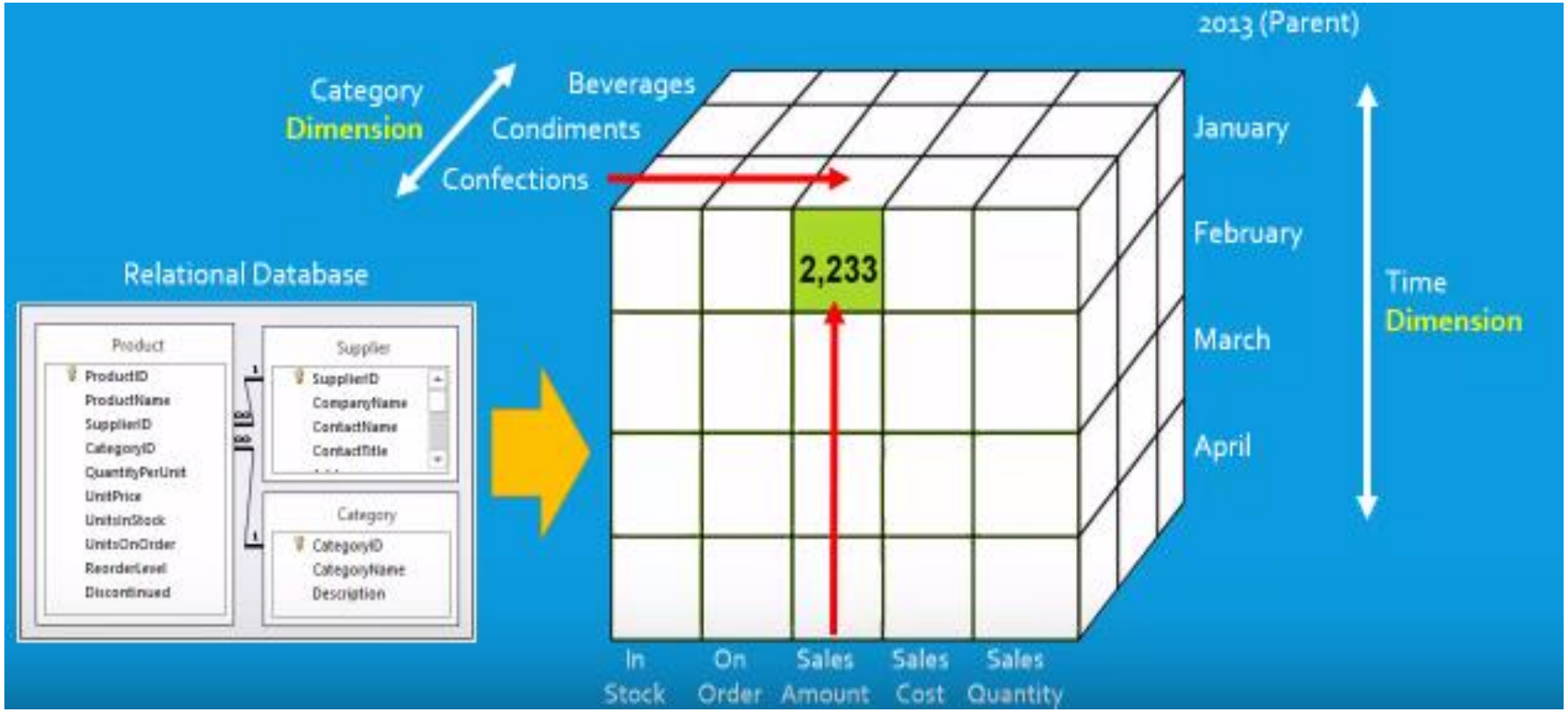
Dimensions:

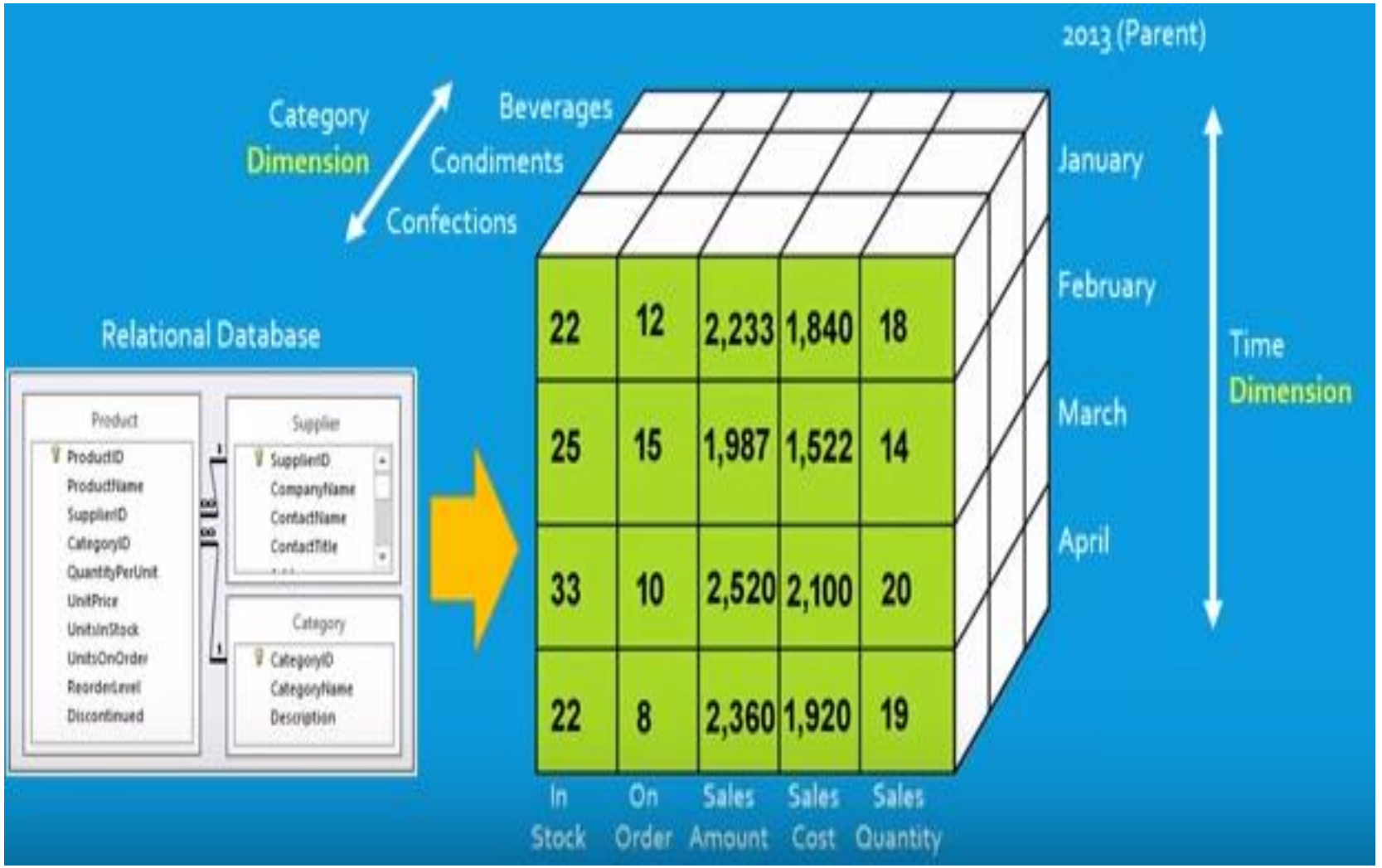
Product, City, Date

Hierachies

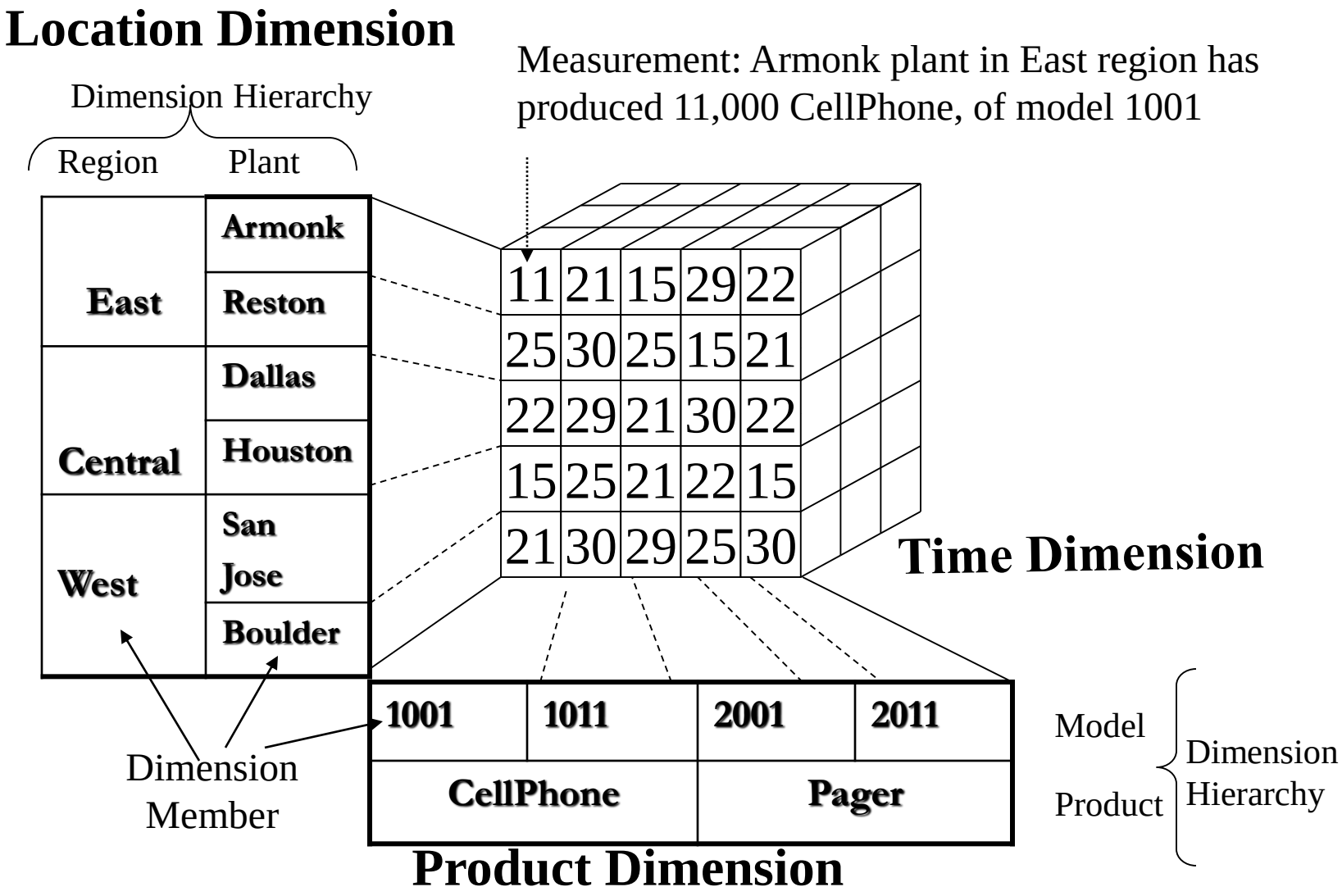
On Line Analytical Processing

- A category of applications and technologies for collecting, managing, processing and presenting multidimensional data for analysis and management purposes.
- OLAP Cube
  - **A multidimensional structure that forms basis for OLAP applications.**
  - **A variety of cross-dimensional calculations and aggregations are possible within a cube.**
  - **Despite the name, most OLAP cubes have more than three dimensions.**


2013 (Parent)
Category
Dimension
Beverages
Condiments
Confections
Relational Database
Product
ProductID
ProductName
SupplierID
CategoryID
QuantityPerUnit
UnitPrice
UnitsInStock
UnitsOnOrder
ReorderLevel
Discontinued
Supplier
SupplierID
CompanyName
ContactName
ContactTitle
Category
CategoryID
CategoryName
Description
2,233
January
February
March
April
Time
Dimension
In Stock
On Order
Sales Amount
Sales Cost
Sales Quantity


2013 (Parent)
Category
Dimension
Beverages
Condiments
Confections
Relational Database
Product
ProductID
ProductName
SupplierID
CategoryID
QuantityPerUnit
UnitPrice
UnitsInStock
UnitsOnOrder
ReorderLevel
Discontinued
Supplier
SupplierID
CompanyName
ContactName
ContactTitle
Category
CategoryID
CategoryName
Description
January
February
March
April
Time
Dimension
22 12 2,233 1,840 18
25 15 1,987 1,522 14
33 10 2,520 2,100 20
22 8 2,360 1,920 19
In Stock
On Order
Sales Amount
Sales Cost
Sales Quantity

Visualization of a Dimensional Model



OLAP CUBE

- Axes of the cube represent attributes of the data records
  - Generally discrete-valued / categorical
  - e.g. color, month, state
  - Called **dimensions**
- Cells hold aggregated measurements
  - e.g. total $ sales, number of autos sold
  - Called **facts**

Basic Operations for OLAP

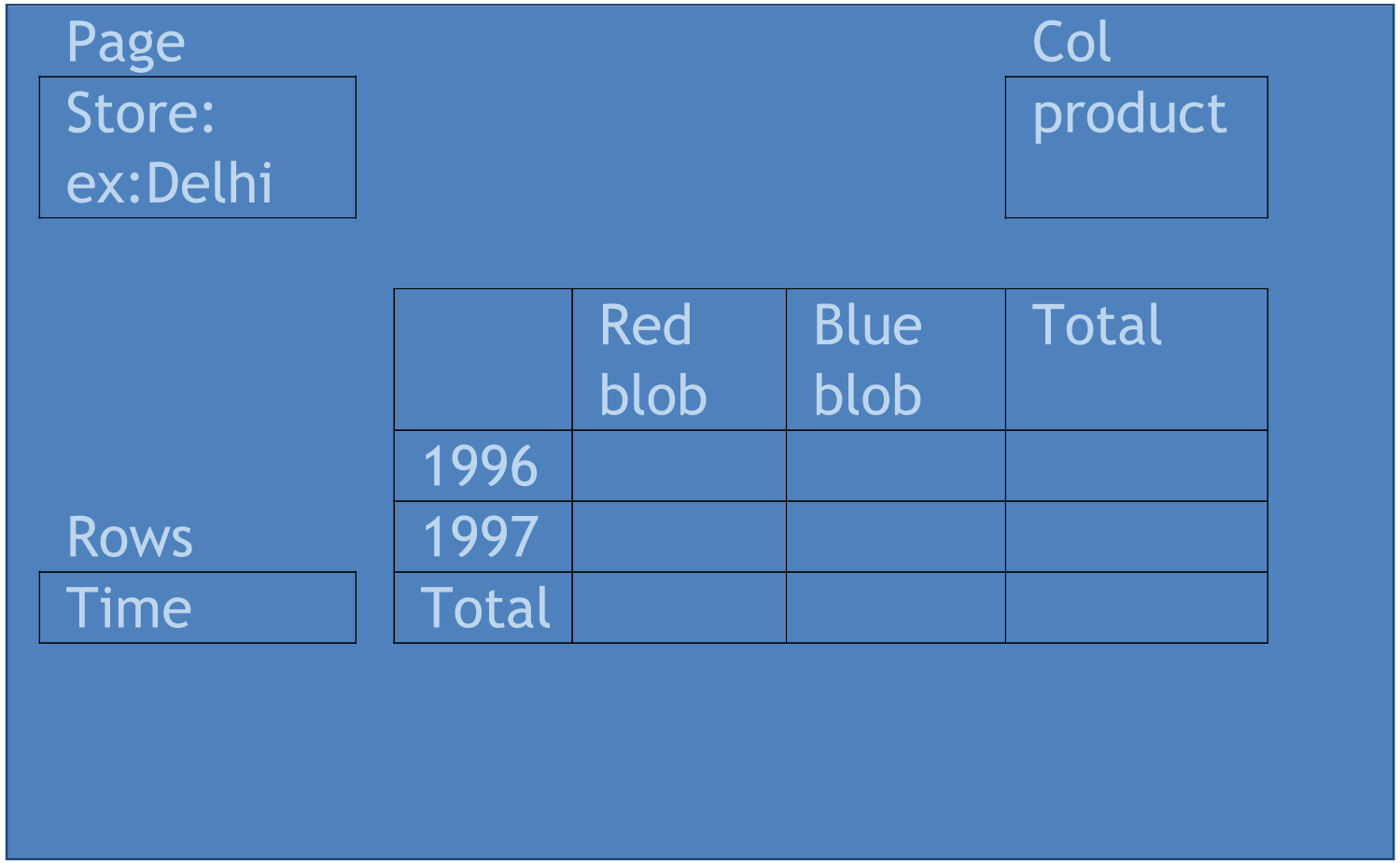
- Drill Down
  - Navigate to higher levels of detail
  - Example: from regional analysis to specific plant analysis, further to team analysis, …
- Roll Up
  - Navigate to lower levels of detail
  - Example: from month analysis to a quarter analysis
- Slice
  - Cut through the cube, so that users can focus on some specific perspectives
  - Example: only analyzing on the product CellPhone

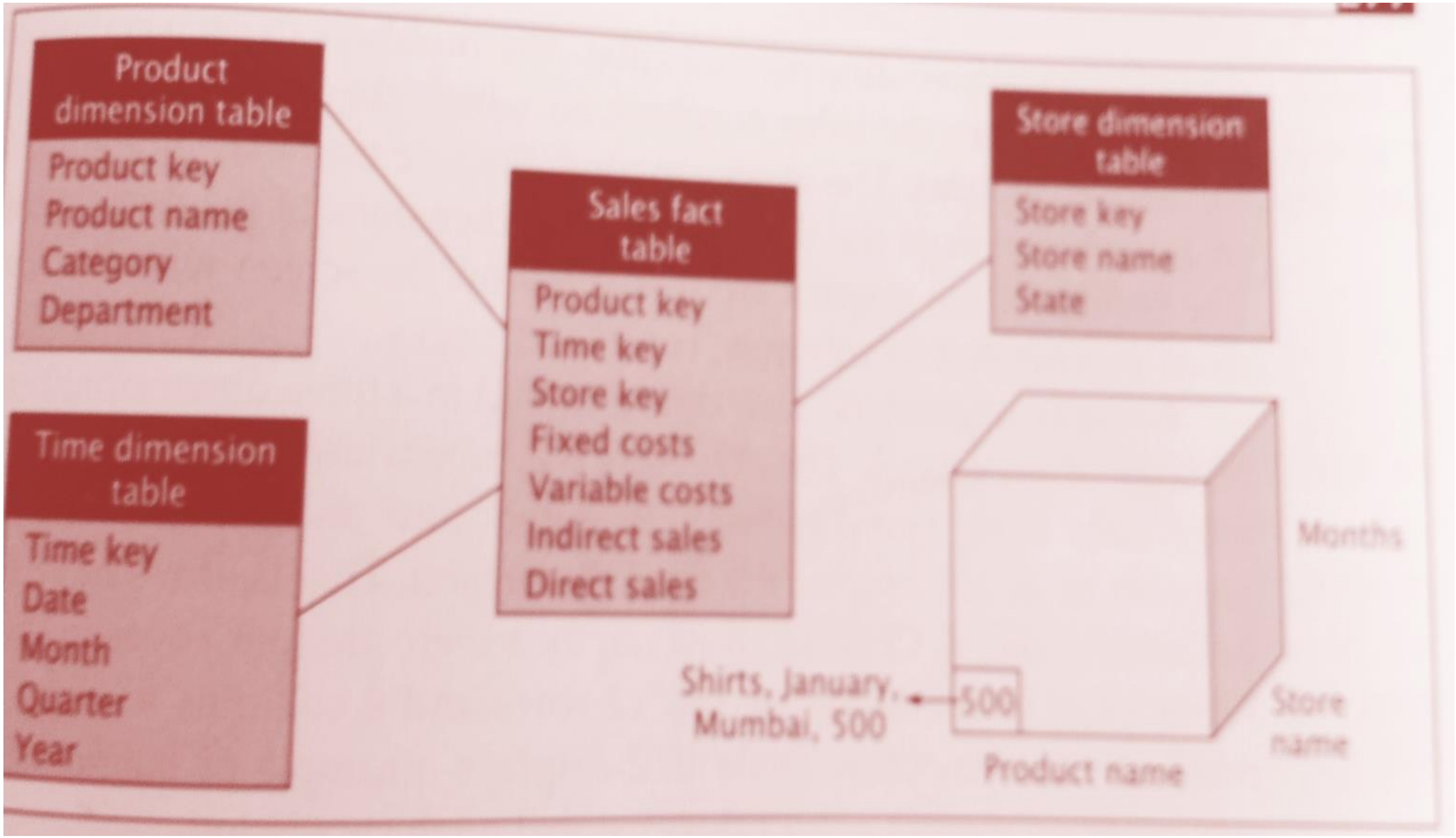
Basic Operations for OLAP (cont.)

- Dice
  - Get one cell from the cube (the smallest slice)
  - Example: get the production volume of Armonk, for CellPhone 1001, in January (here, we suppose *plant*, *product model* and *month* are the smallest members in **Location**, **Product**, **Time** dimensions respectively)
- Pivot
  - Rotate the cube
  - Example: change the perspective from “Region X Product” to “Region X Time”

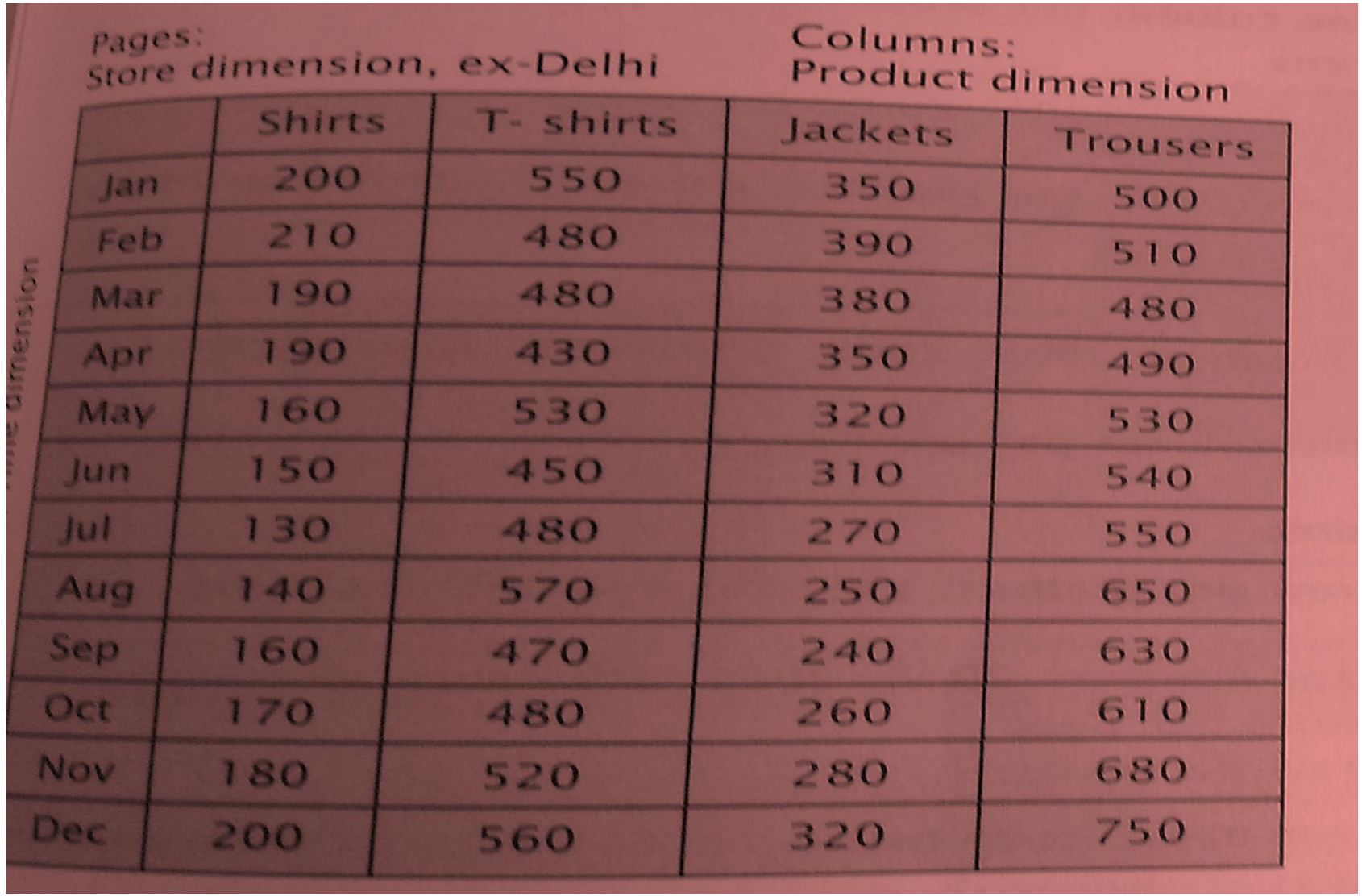
OLAP Cube - 3

Three-Dimensional Cube Display




Product dimension table
Product key
Product name
Category
Department
Time dimension table
Time key
Date
Month
Quarter
Year
Sales fact table
Product key
Time key
Store key
Fixed costs
Variable costs
Indirect sales
Direct sales
Store dimension table
Store key
Store name
State
Shirts, January, Mumbai, 500
500
Months
Store name
Product name

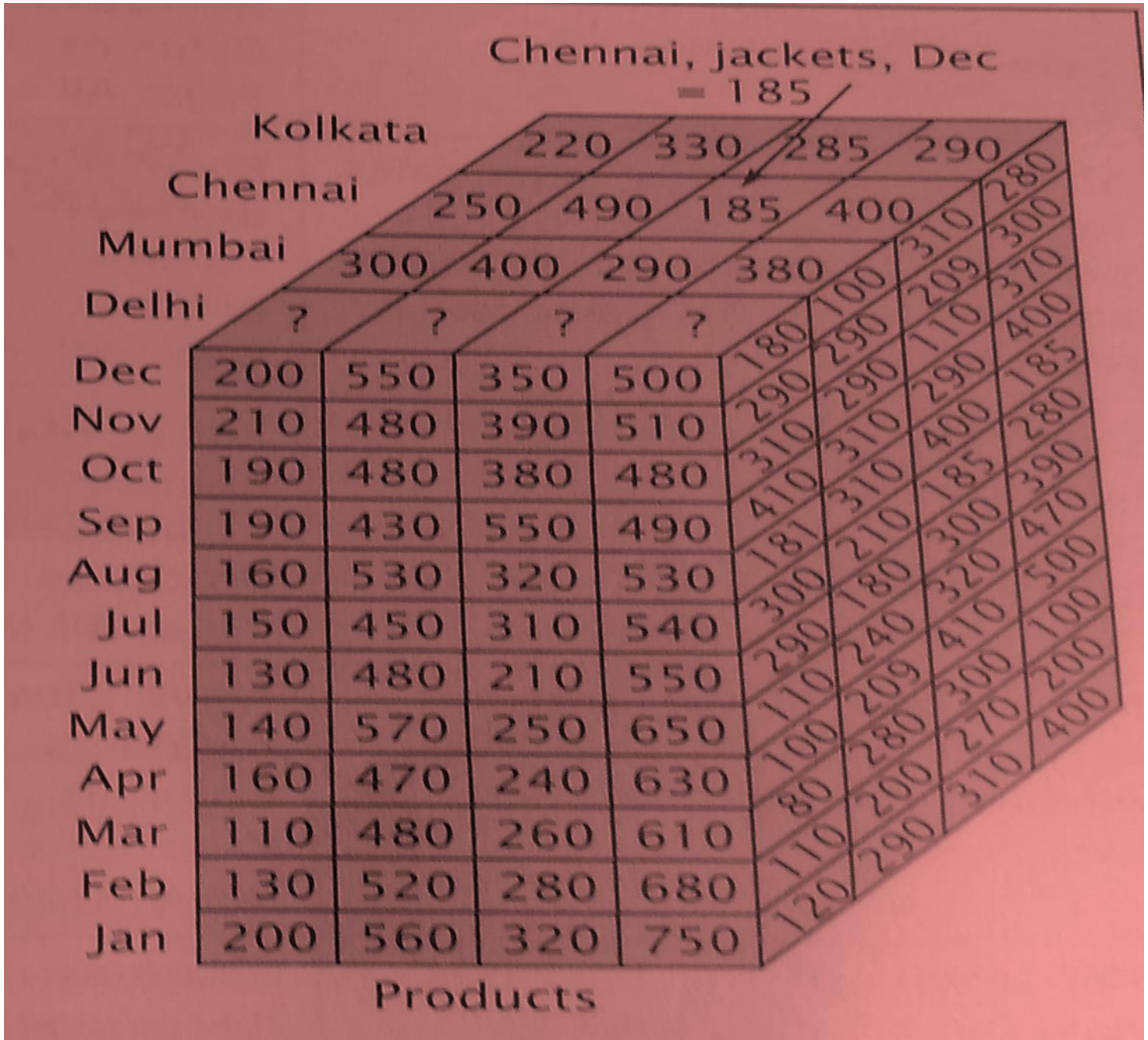
Row:Time Dimension

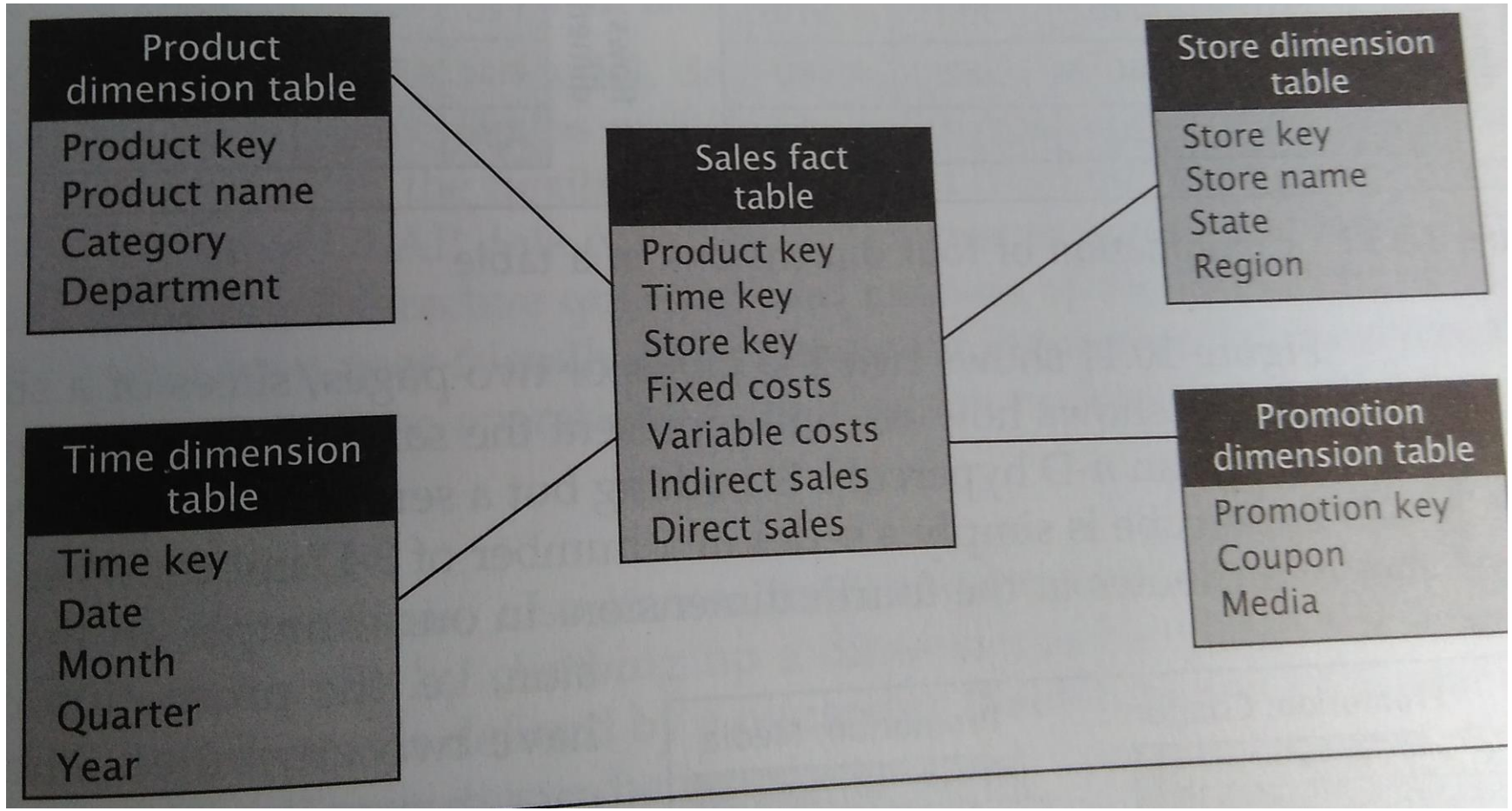


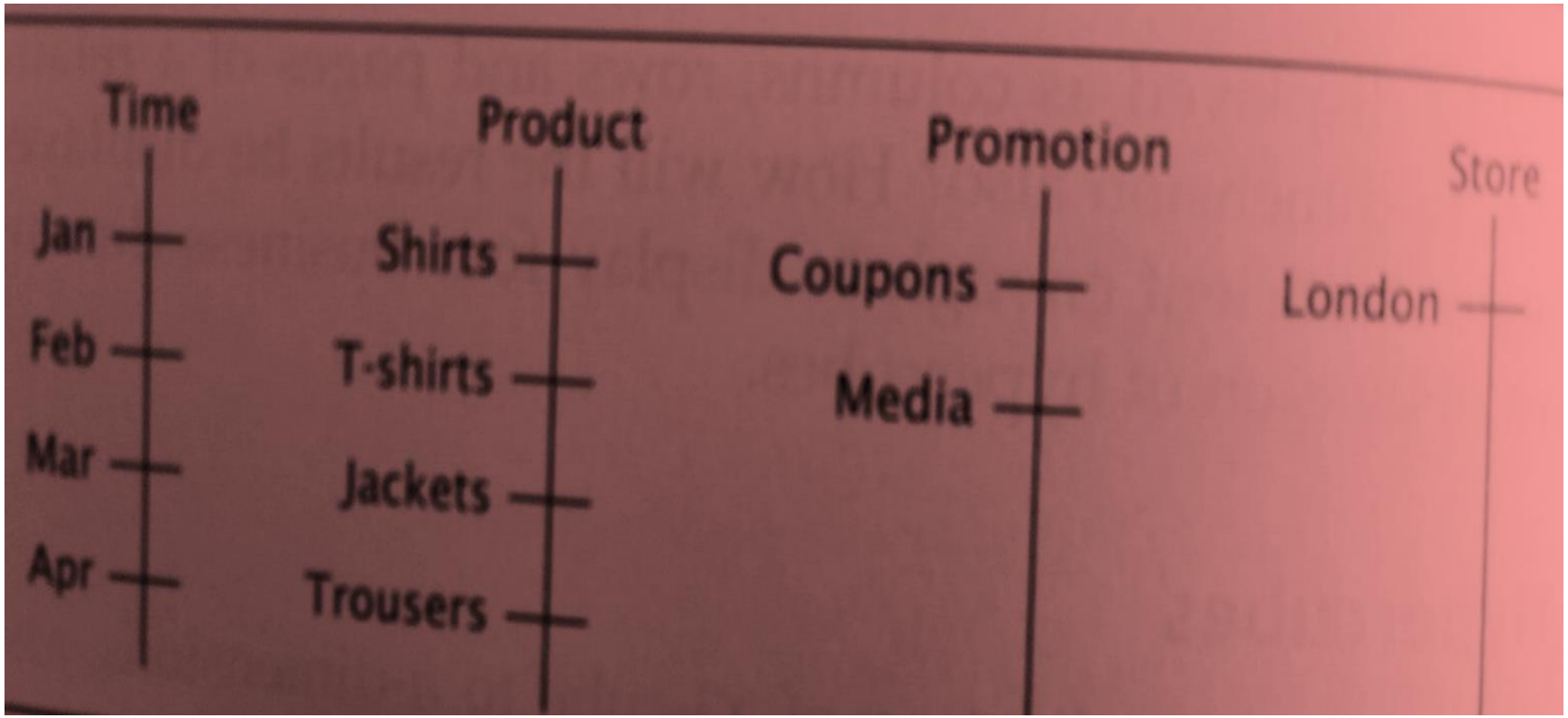
Pages:
Store dimension, ex-Delhi

Columns:
Product dimension

	Shirts	T- shirts	Jackets	Trousers
Jan	200	550	350	500
Feb	210	480	390	510
Mar	190	480	380	480
Apr	190	430	350	490
May	160	530	320	530
Jun	150	450	310	540
Jul	130	480	270	550
Aug	140	570	250	650
Sep	160	470	240	630
Oct	170	480	260	610
Nov	180	520	280	680
Dec	200	560	320	750


Chennai, jackets, Dec = 185
Kolkata
Chennai
Mumbai
Delhi
Dec
Nov
Oct
Sep
Aug
Jul
Jun
May
Apr
Mar
Feb
Jan
Products


Product dimension table
Product key
Product name
Category
Department
Time dimension table
Time key
Date
Month
Quarter
Year
Sales fact table
Product key
Time key
Store key
Fixed costs
Variable costs
Indirect sales
Direct sales
Store dimension table
Store key
Store name
State
Region
Promotion dimension table
Promotion key
Coupon
Media


Time
Jan
Feb
Mar
Apr
Product
Shirts
T-shirts
Jackets
Trousers
Promotion
Coupons
Media
Store
London

Pages: Store and promotion dimension combined
London: Coupon
Columns: Product dimension

Rows: Time dimension

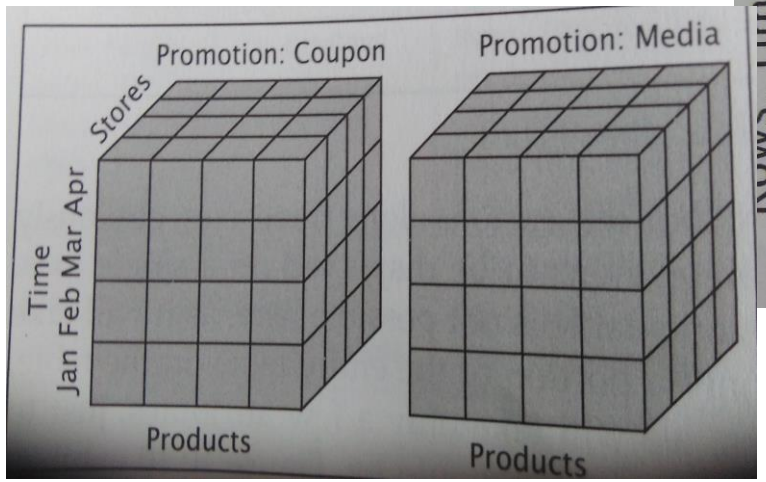
	Shirts	T- shirts	Jackets	Trousers
Jan	200	550	350	500
Feb	210	480	390	510
Mar	190	480	380	480
Apr	190	430	350	490

n-D hypercube is nothing but a series of (n-1)-D cubes

Pages: Store and promotion dimension combined
London: Media
Columns: Product dimension

Rows: Time dimension

	Shirts	T- shirts	Jackets	Trousers
Jan	150	450	310	540
Feb	130	480	270	550
Mar	140	570	250	650
Apr	160	470	240	630

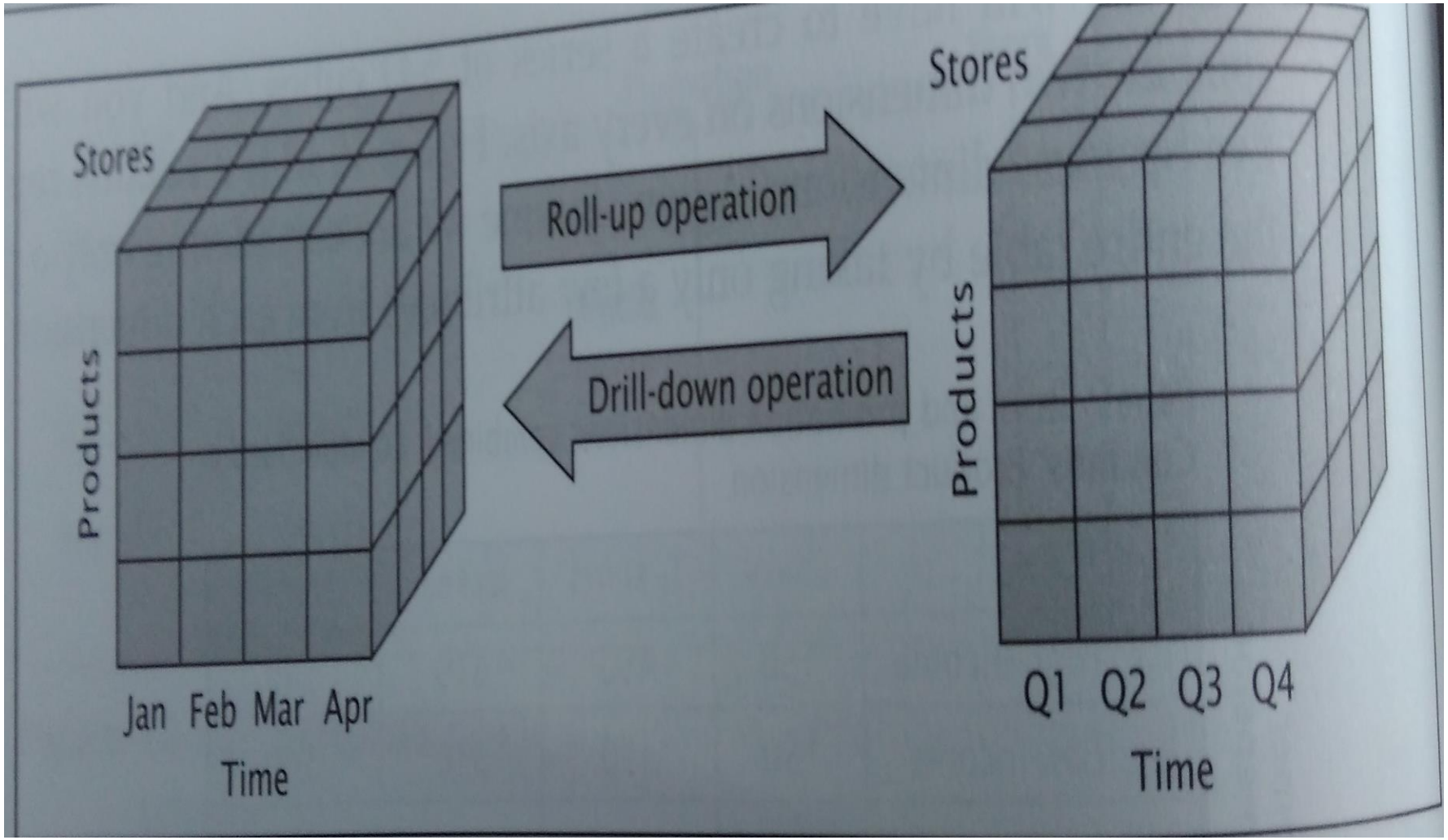


4-D hypercube is simply a series of 3-D cubes where x is number of attributes in fourth dimension

Pages: Store and promotion dimensions combined, London: Media
Columns: Product dimension

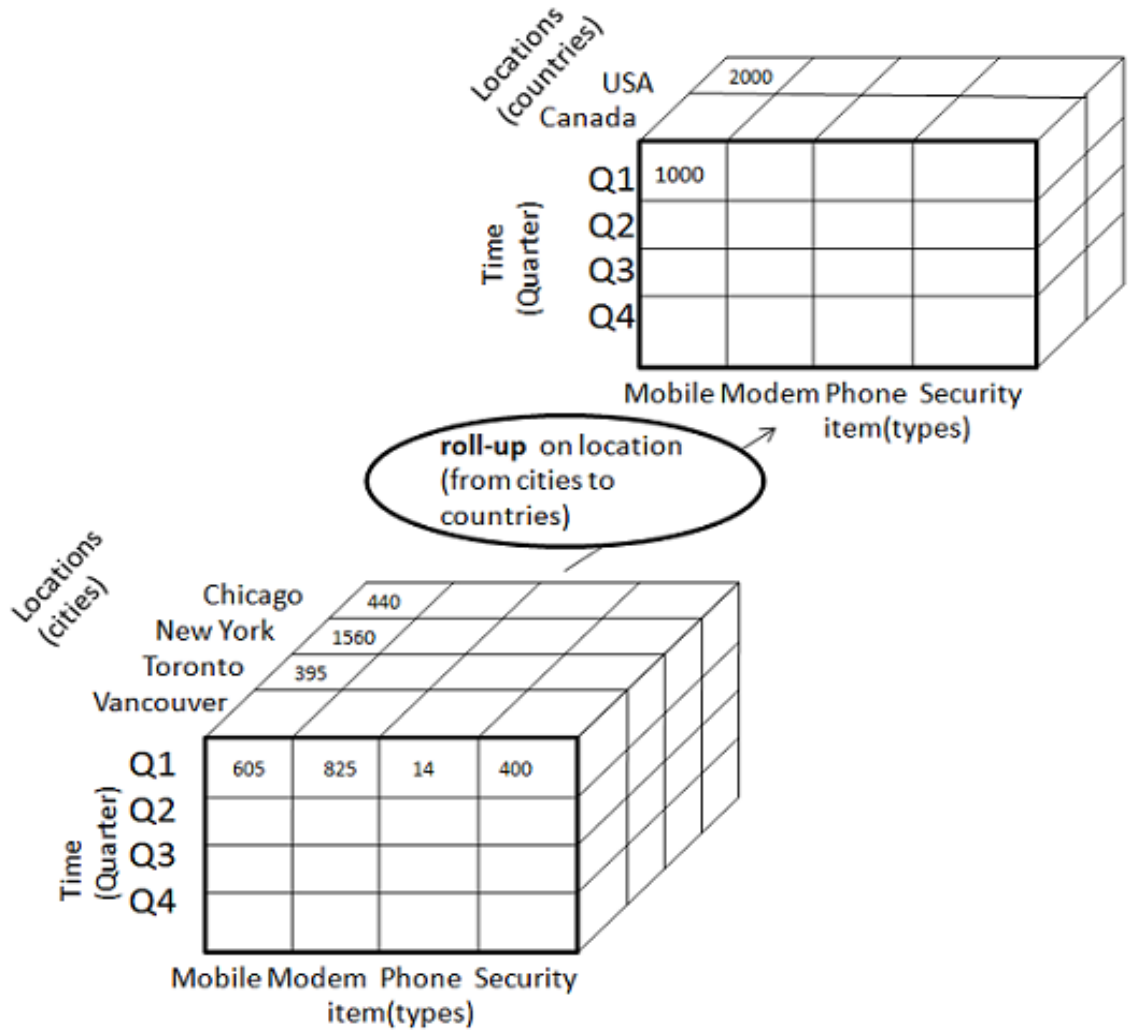
Rows: Time customer demographics dimension

	Shirts	T- shirts	Jackets	Trousers
Jan: High-income	150	450	310	540
Jan: Low-income	150	450	310	540
Feb: High-income	130	480	270	550
Feb: Low-income	130	480	270	550
Mar: High-income	140	570	250	650
Mar: Low-income	140	570	250	650
Apr: High-income	160	470	240	630
Apr: Low-income	160	470	240	630

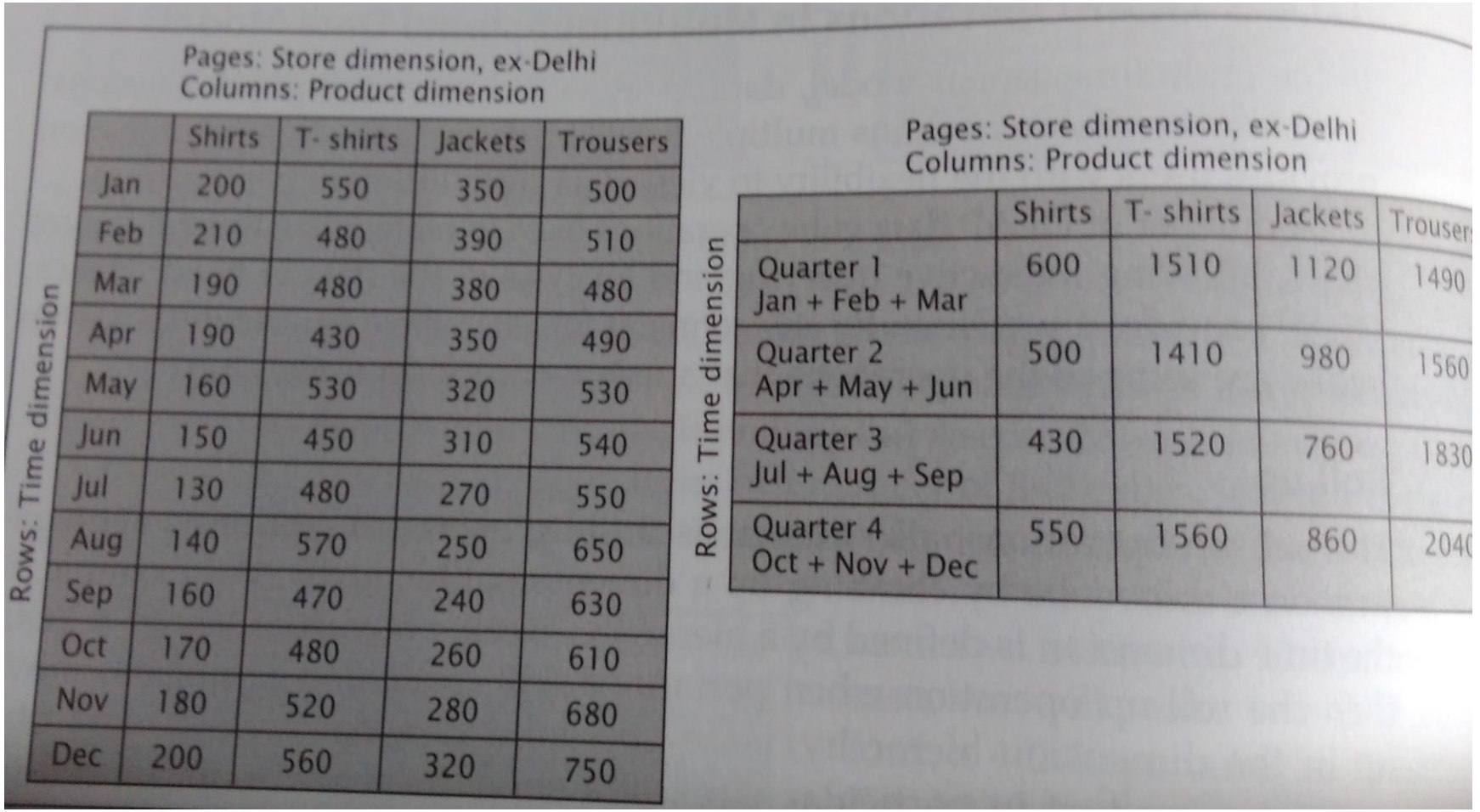

Stores
Products
Jan Feb Mar Apr
Time
Roll-up operation
Drill-down operation
Stores
Products
Q1 Q2 Q3 Q4
Time

Roll-up

- Roll- up performs aggregation on a data cube by climbing up a dimensional hierarchy



Drill down navigates from less detailed data to more detailed data



Pages: Store dimension, ex-Delhi
Columns: Product dimension

Rows: Time dimension

	Shirts	T- shirts	Jackets	Trousers
Jan	200	550	350	500
Feb	210	480	390	510
Mar	190	480	380	480
Apr	190	430	350	490
May	160	530	320	530
Jun	150	450	310	540
Jul	130	480	270	550
Aug	140	570	250	650
Sep	160	470	240	630
Oct	170	480	260	610
Nov	180	520	280	680
Dec	200	560	320	750

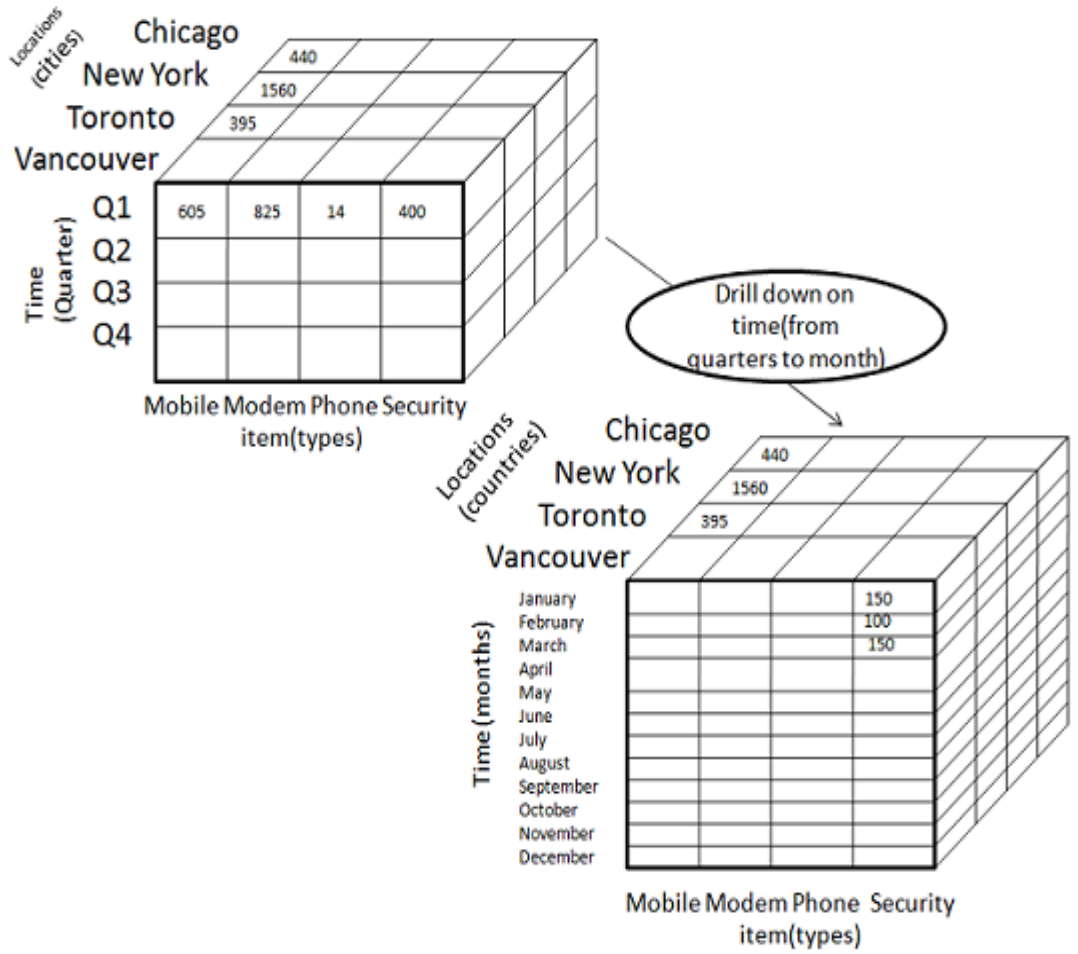
Pages: Store dimension, ex-Delhi
Columns: Product dimension

Rows: Time dimension

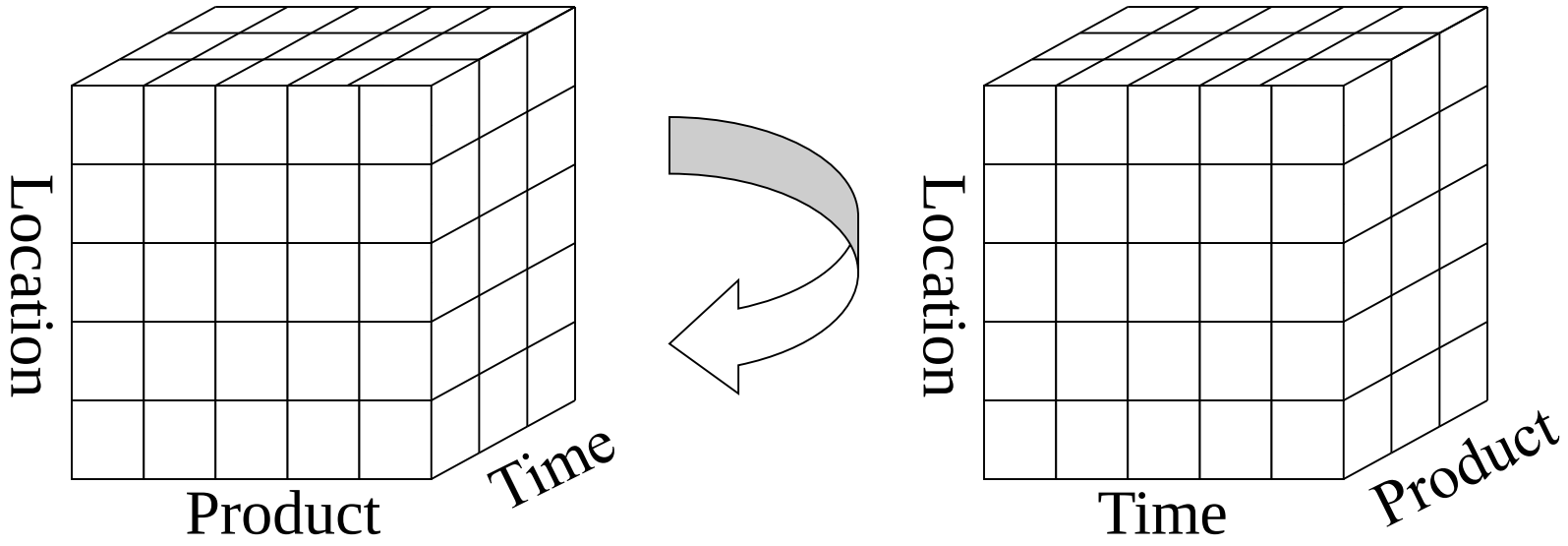
	Shirts	T- shirts	Jackets	Trouser
Quarter 1 Jan + Feb + Mar	600	1510	1120	1490
Quarter 2 Apr + May + Jun	500	1410	980	1560
Quarter 3 Jul + Aug + Sep	430	1520	760	1830
Quarter 4 Oct + Nov + Dec	550	1560	860	204(

Drill Down

- Drill Down can be realized by stepping down the dimensional hierarchy
- Drill down navigates from less detailed data to more detailed data

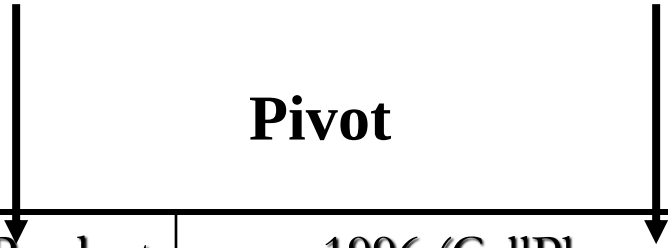


Pivot



Pivot (cont.)

Volume of Product (numbers in 1000)		CellPhone		Pager	
		1001	1011	2001	2011
West	San Jose	33	12	8	12
	Boulder	45	34	20	23



Volume of Product (numbers in 1000)		1996 (CellPhone & Pager)			
		Qtr1	Qtr2	Qtr3	Qtr4
West	San Jose	78	45	34	56
	Boulder	90	67	87	91

Pages: Store dimension, ex-Delhi
Columns: Product dimension

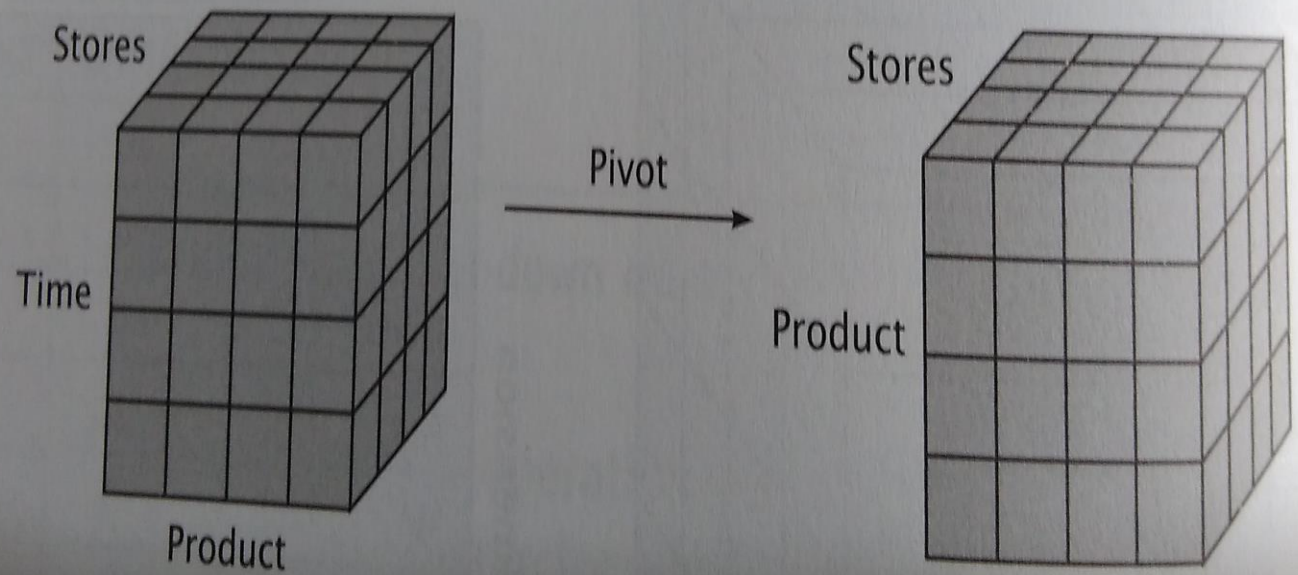
Rows: Time dimension

	Shirts	T- shirts	Jackets	Trousers
Quarter 1	600	1510	1120	1490
Quarter 2	500	1410	980	1560
Quarter 3	430	1520	760	1830
Quarter 4	550	1560	860	2040

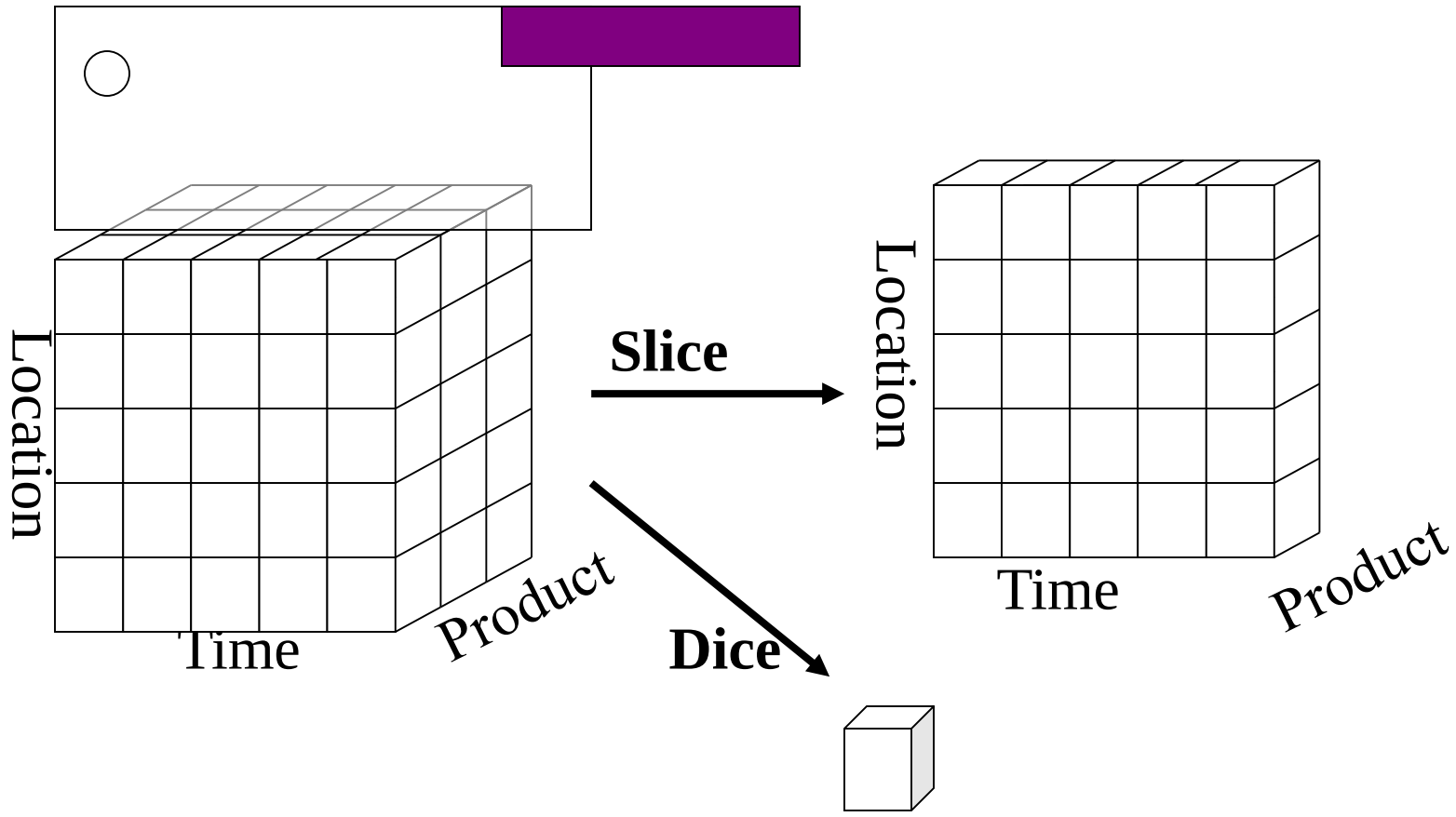
Pages: Store dimension, ex-Delhi
Columns: Product dimension

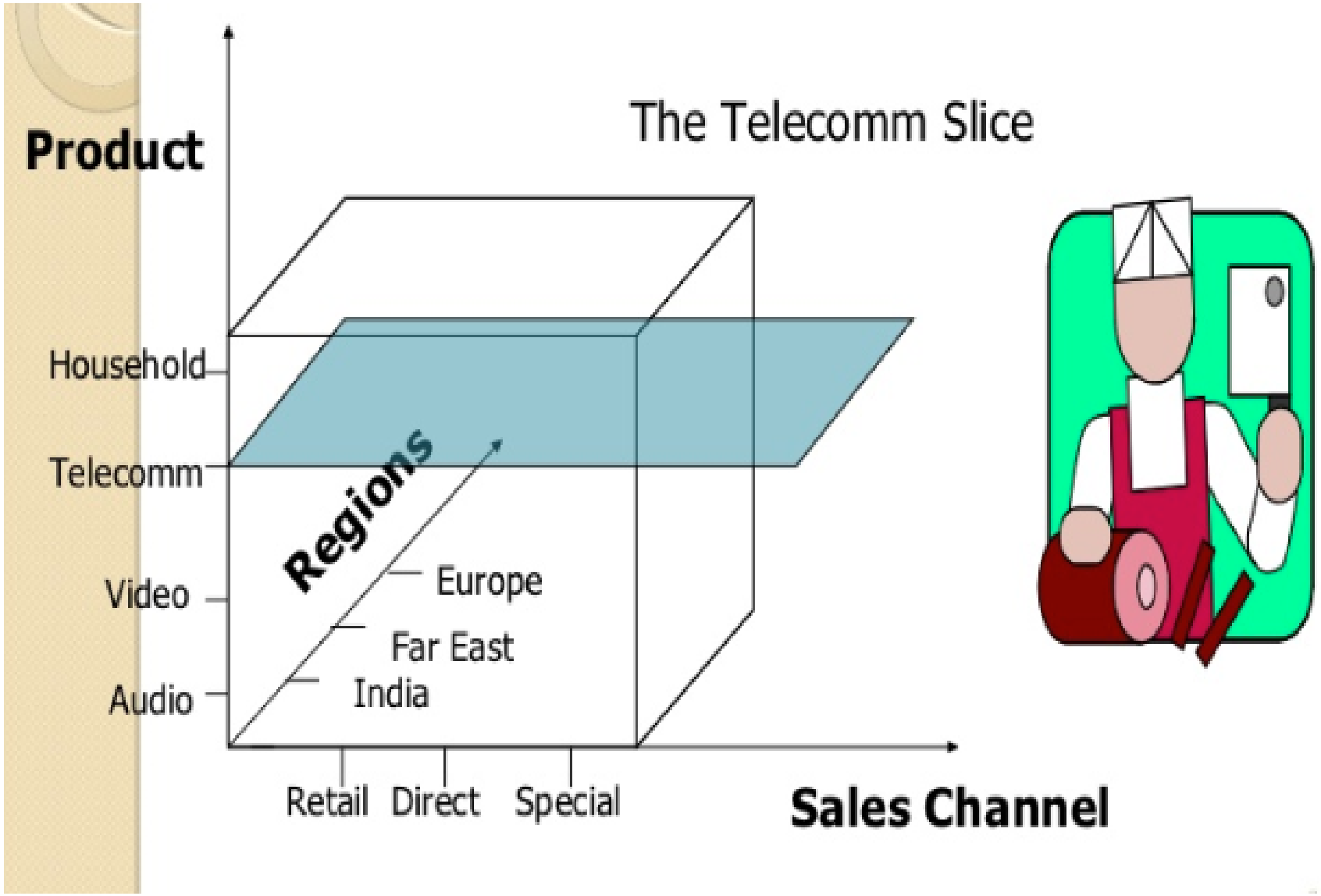
Rows: Time dimension

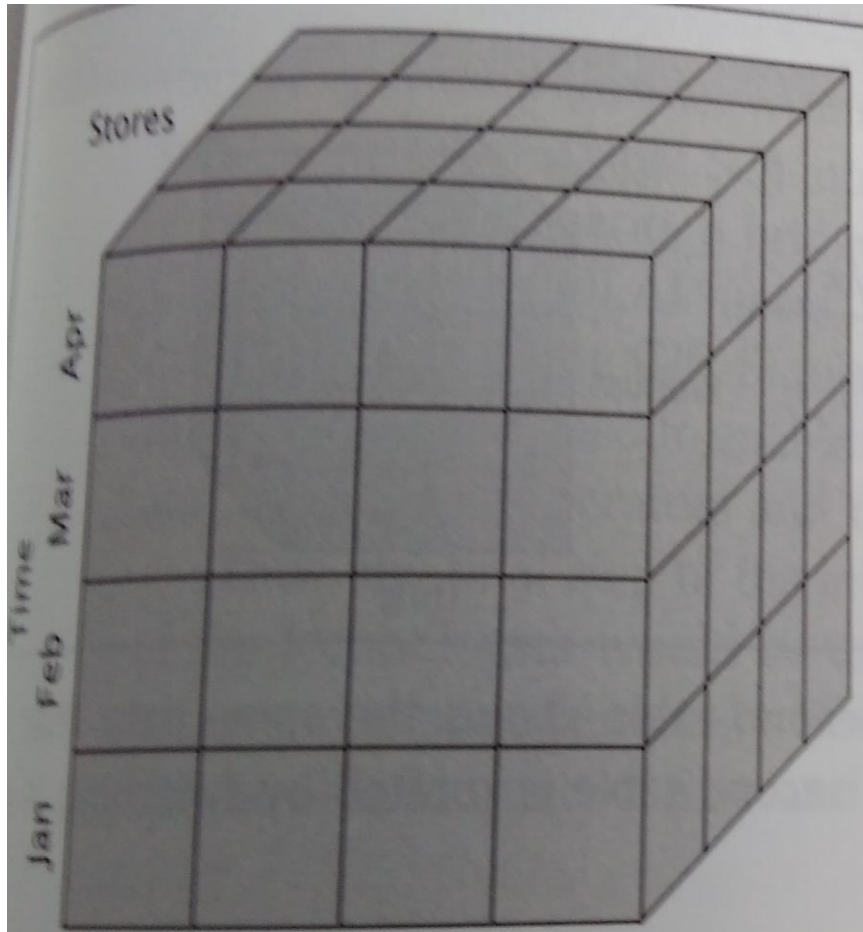
	Quarter 1	Quarter 2	Quarter 3	Quarter 4
Shirts	600	500	430	550
T-shirts	1510	1410	1520	1560
Jackets	1120	980	760	860
Trousers	1490	1560	1830	2040



Slice & Dice




The Telecomm Slice
Product
Household
Telecomm
Video
Audio
Regions
Europe
Far East
India
Retail
Direct
Special
Sales Channel



Store: London

	Jackets
March	350

(a)

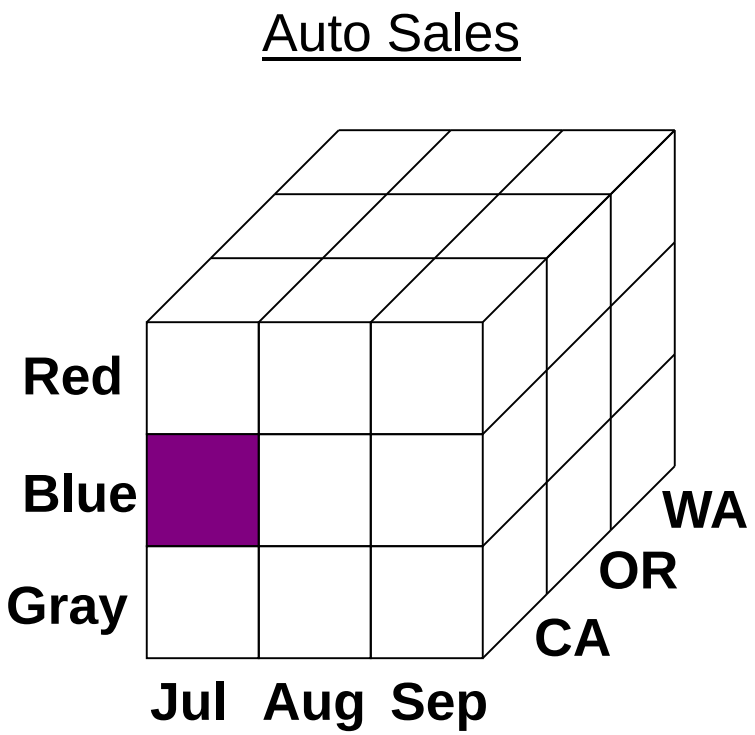
Pages: Store dimension, ex-Delhi
Columns: Product dimension

Rows: Time dimension

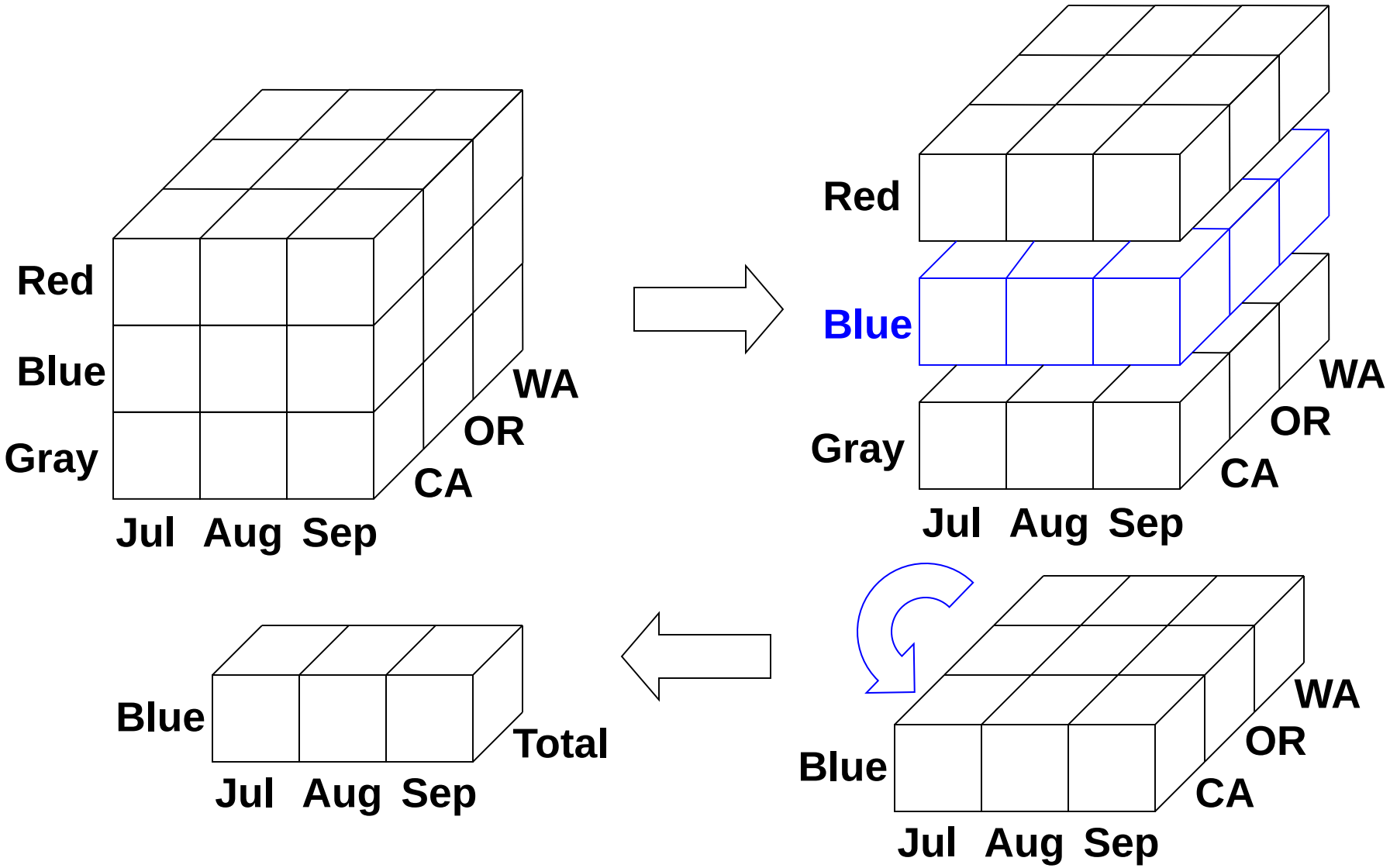
	Shirts	T- shirts	Jackets	Trousers
Jan	200	550	350	500
Feb	210	480	390	510
Mar	190	480	380	480
Apr	190	430	350	490
May	160	530	320	530
Jun	150	450	310	540
Jul	130	480	270	550
Aug	140	570	250	650
Sep	160	470	240	630
Oct	170	480	260	610
Nov	180	520	280	680
Dec	200	560	320	750

(b)

Data Cube Another Example



Slicing and Dicing



Querying the Data Cube

- Operations on a cross-tab
  - Roll up (further aggregation)
  - Drill down (less aggregation)

Number of Autos Sold

	CA	OR	WA	Total
Jul	45	33	30	108
Aug	50	36	42	128
Sep	38	31	40	109
Total	133	100	112	345

Roll Up and Drill Down

Number of Autos Sold

	CA	OR	WA	Total
Jul	45	33	30	108
Aug	50	36	42	128
Sep	38	31	40	109
Total	133	100	112	345

Roll up
by Month

Number of Autos Sold

CA	OR	WA	Total
133	100	112	345

Drill down
by Color

Number of Autos Sold

	CA	OR	WA	Total
Red	40	29	40	109
Blue	45	31	37	113
Gray	48	40	35	123
Total	133	100	112	345

Example: Olap of an Automobile Marketer

The Story

An automobile marketer wants to improve business activity. Therefore he wants to view sales figures from different perspectives.

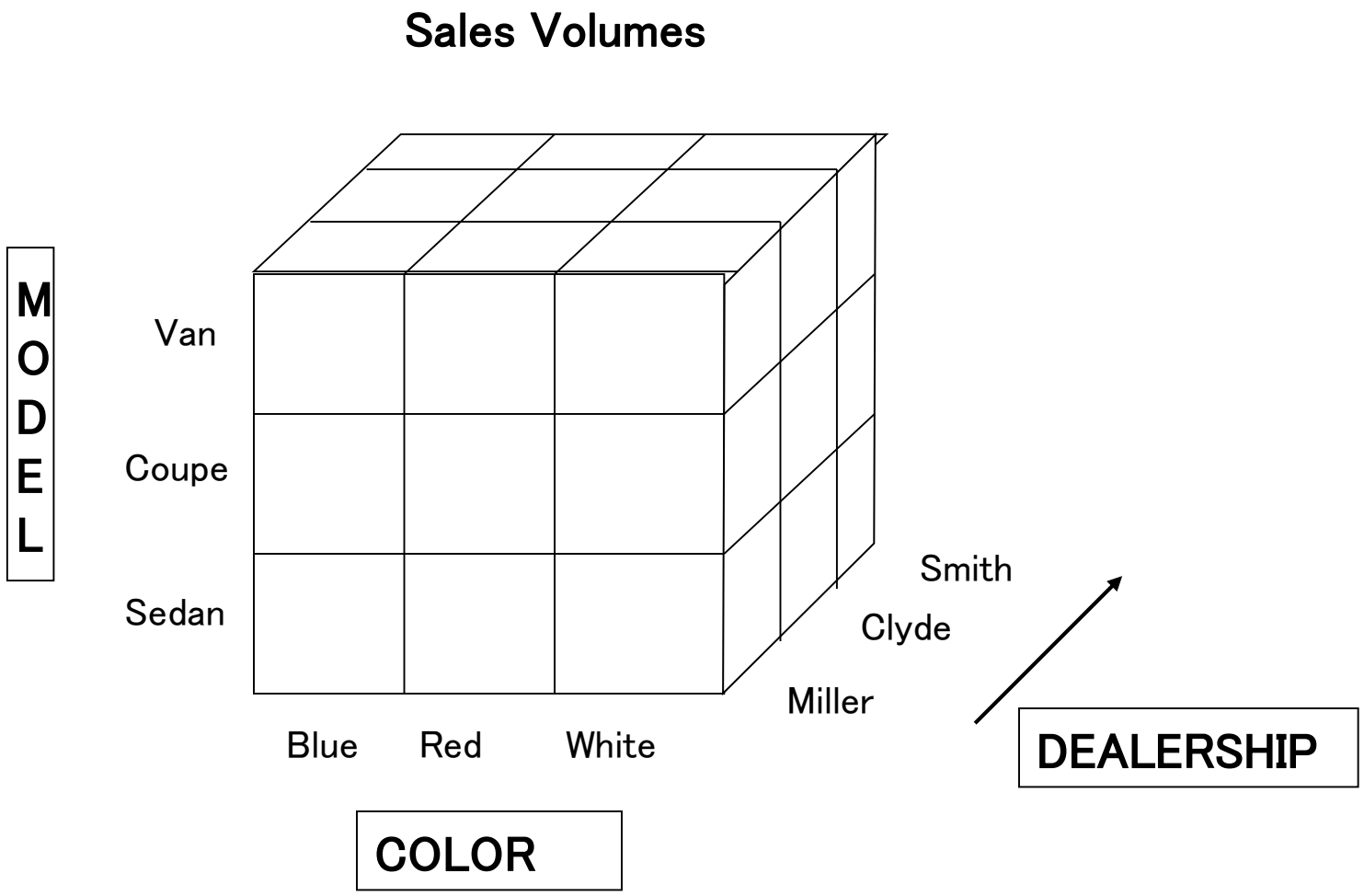
The Data Needs

- Sales by model
- Sales by dealership
- Sales by color
- Sales over time
- etc.

A Question

What is the trend in sales volumes over a period of time for a specific model and color across a specific group of dealerships ?

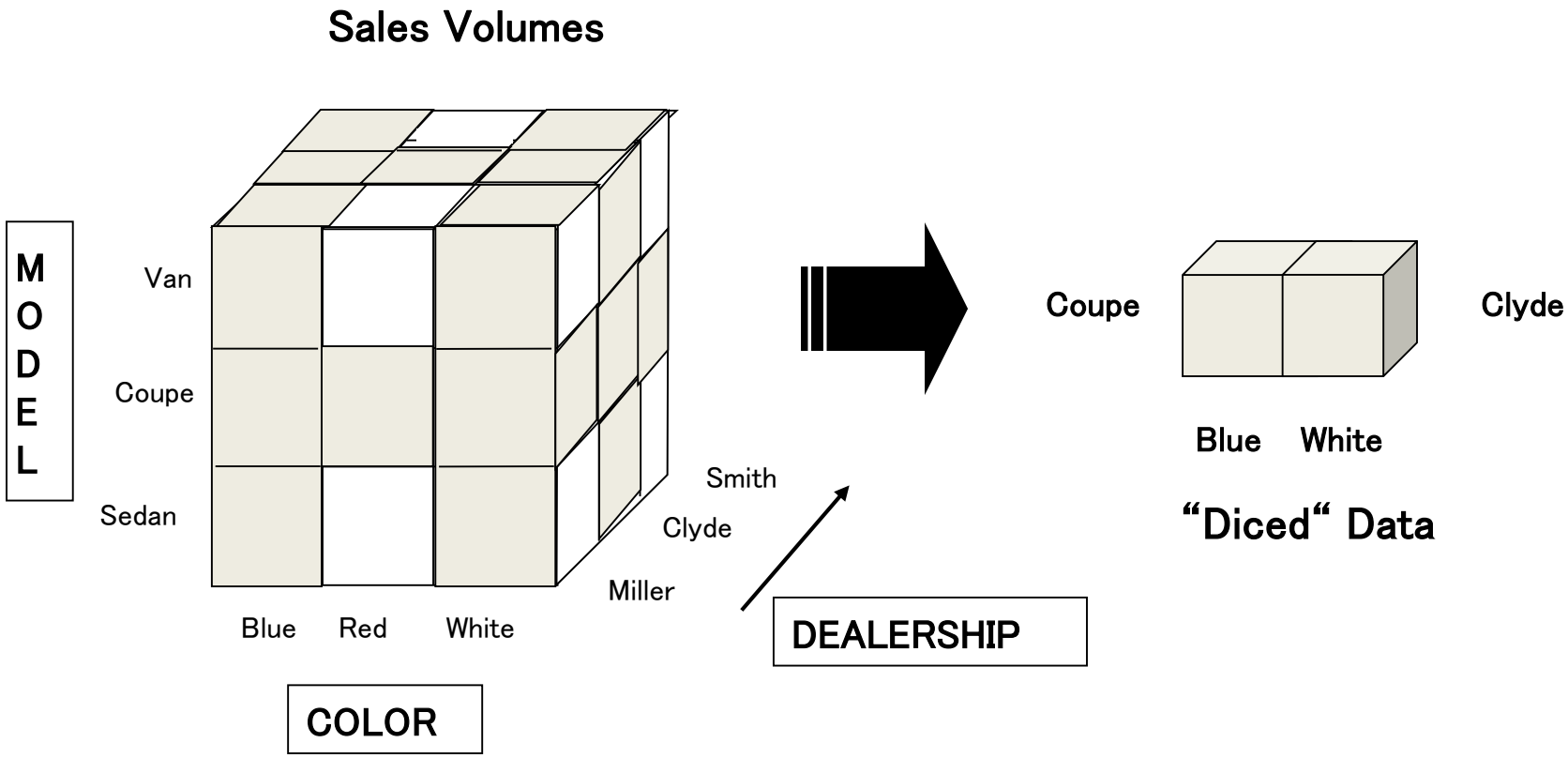
Example: The Multidimensional Database Used



OLAP Features: “Dicing“ – Ranging of Data

Choosing a range out of each dimension:

- Color: Blue and White
- Model: Coupe only
- Dealership: Clyde only



OLAP Features: "Slicing" and Rotation of Data

Different Users will require different views (cross tables or "slices") of the multidimensional cube – OLAP allows easy rotation of data

Dealership=Miler

Sales Volumes

MODEL

Van
Coupe
Sedan

Blue Red White

COLOR

Sales Volumes

MODEL

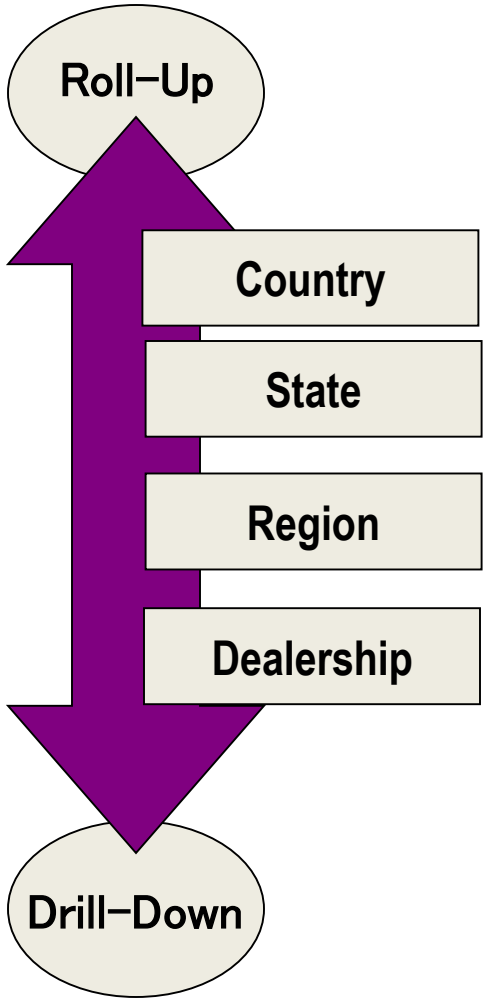
Van
Coupe
Sedan

Miller Smith Clyde

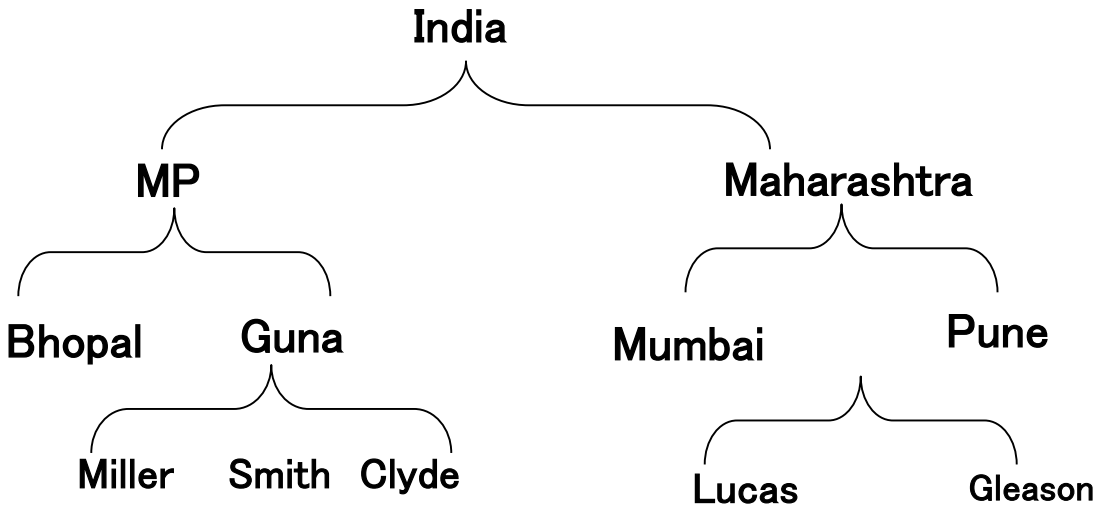
DEALERSHIP

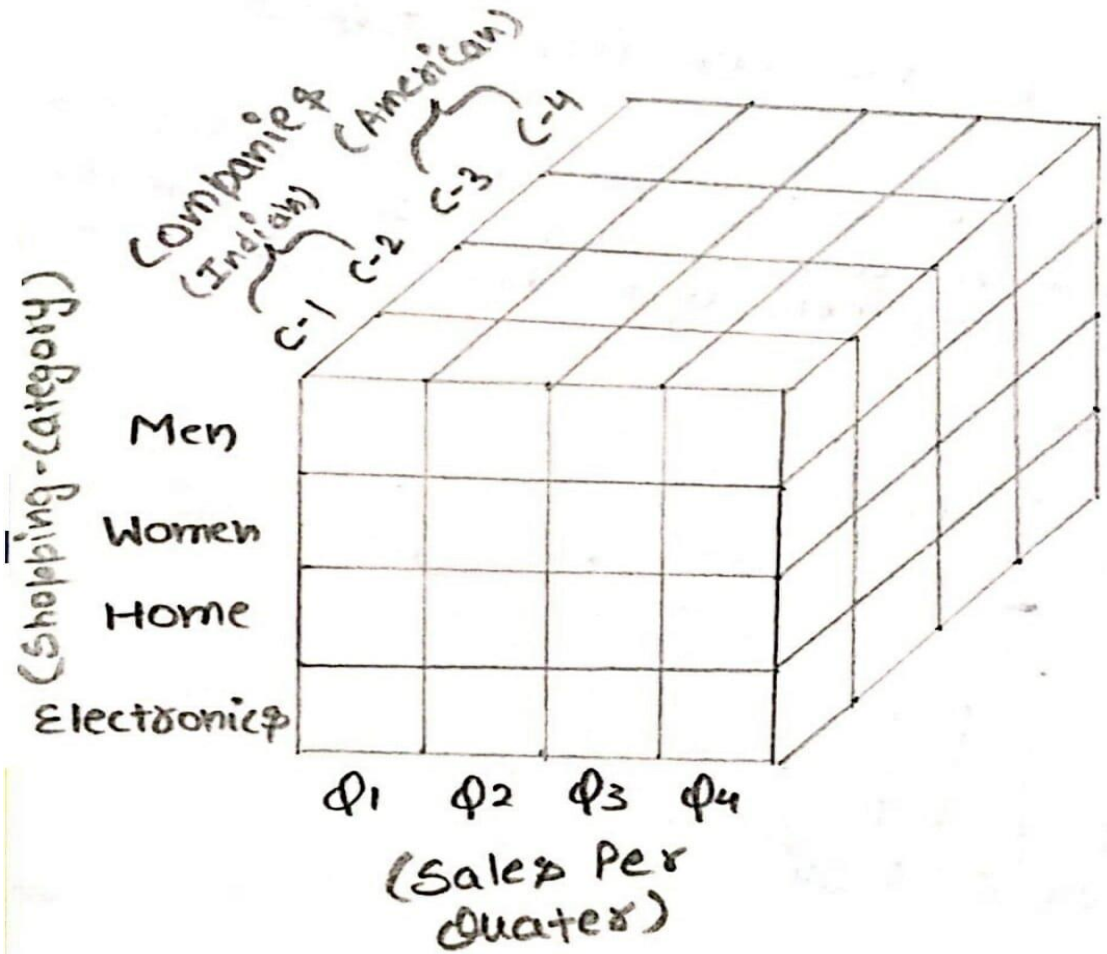
OLAP Features: Drill-Down and Roll-Up

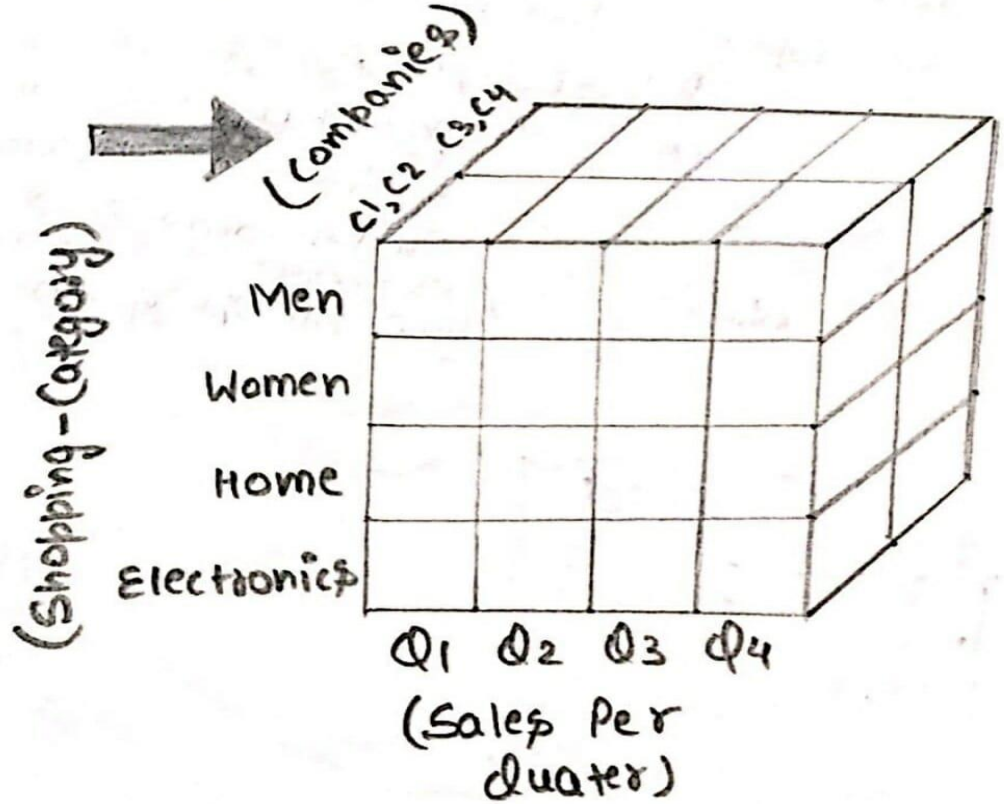
Data can be disaggregated and aggregated along a dimension according to their natural hierarchy

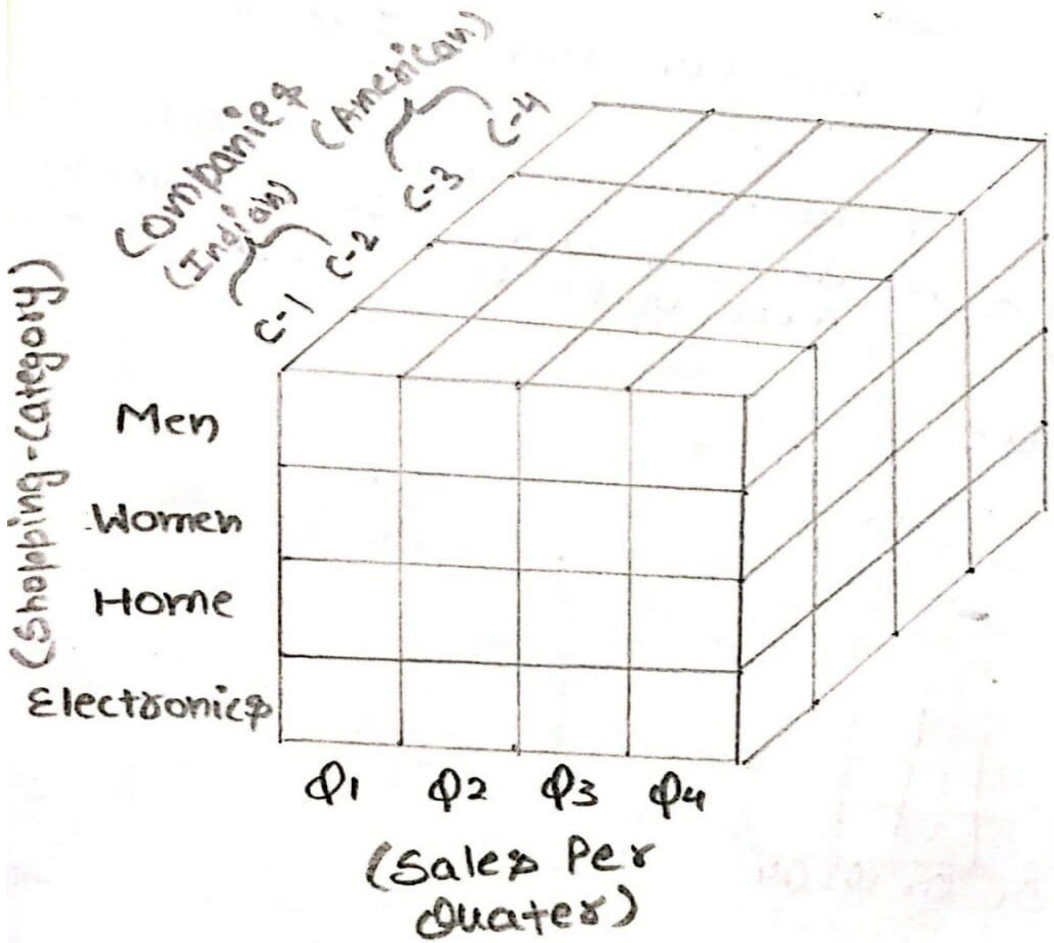


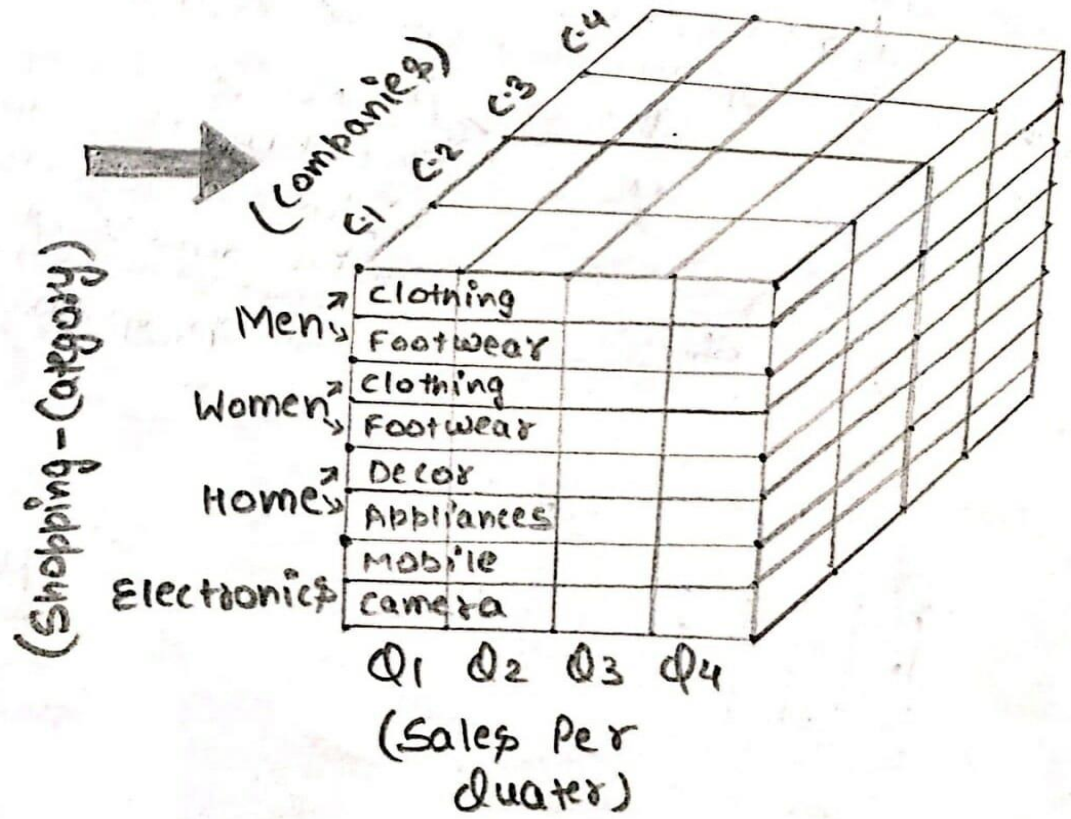
Sales Volumes by Organization Dimension
– four level hierarchy –

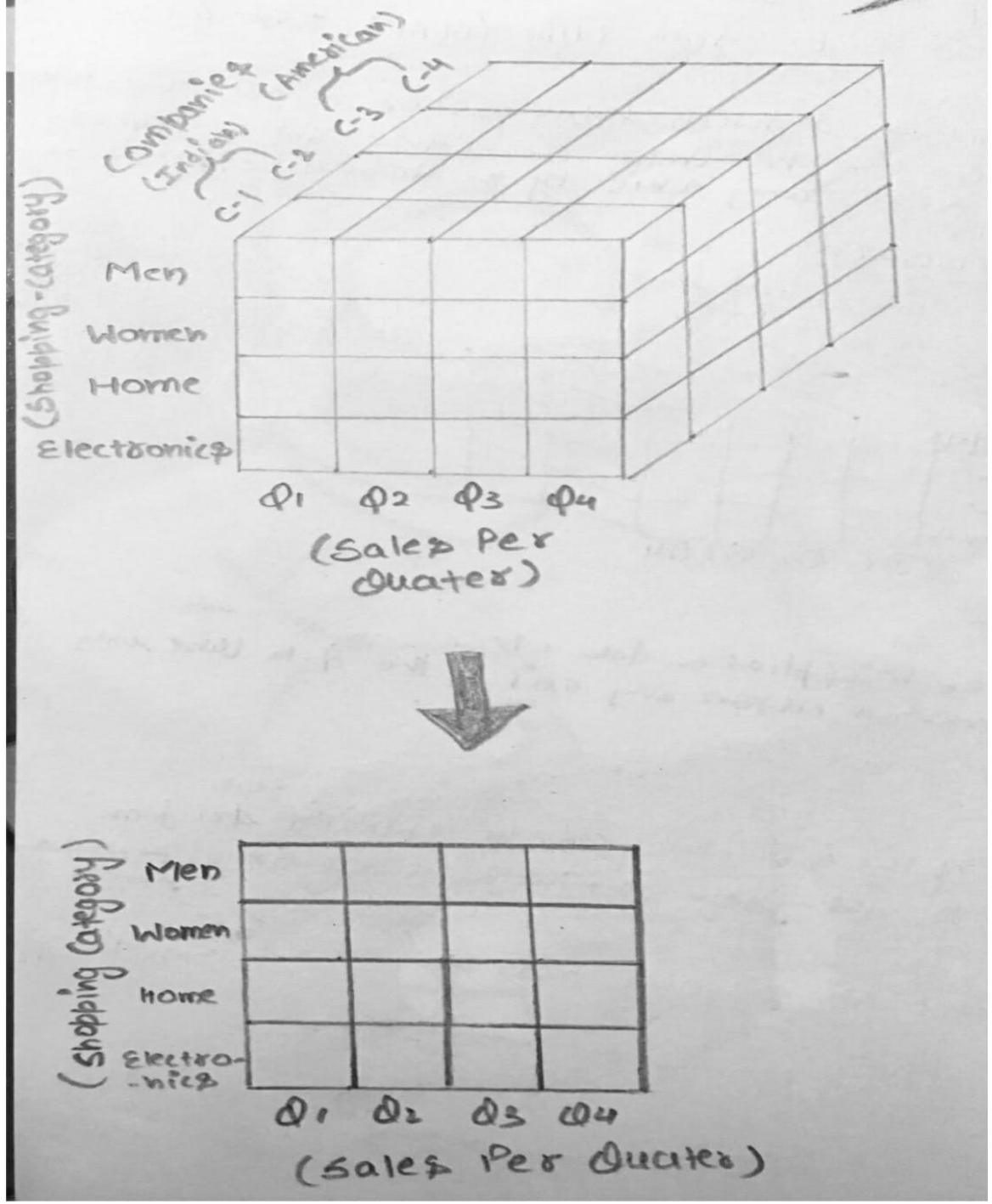


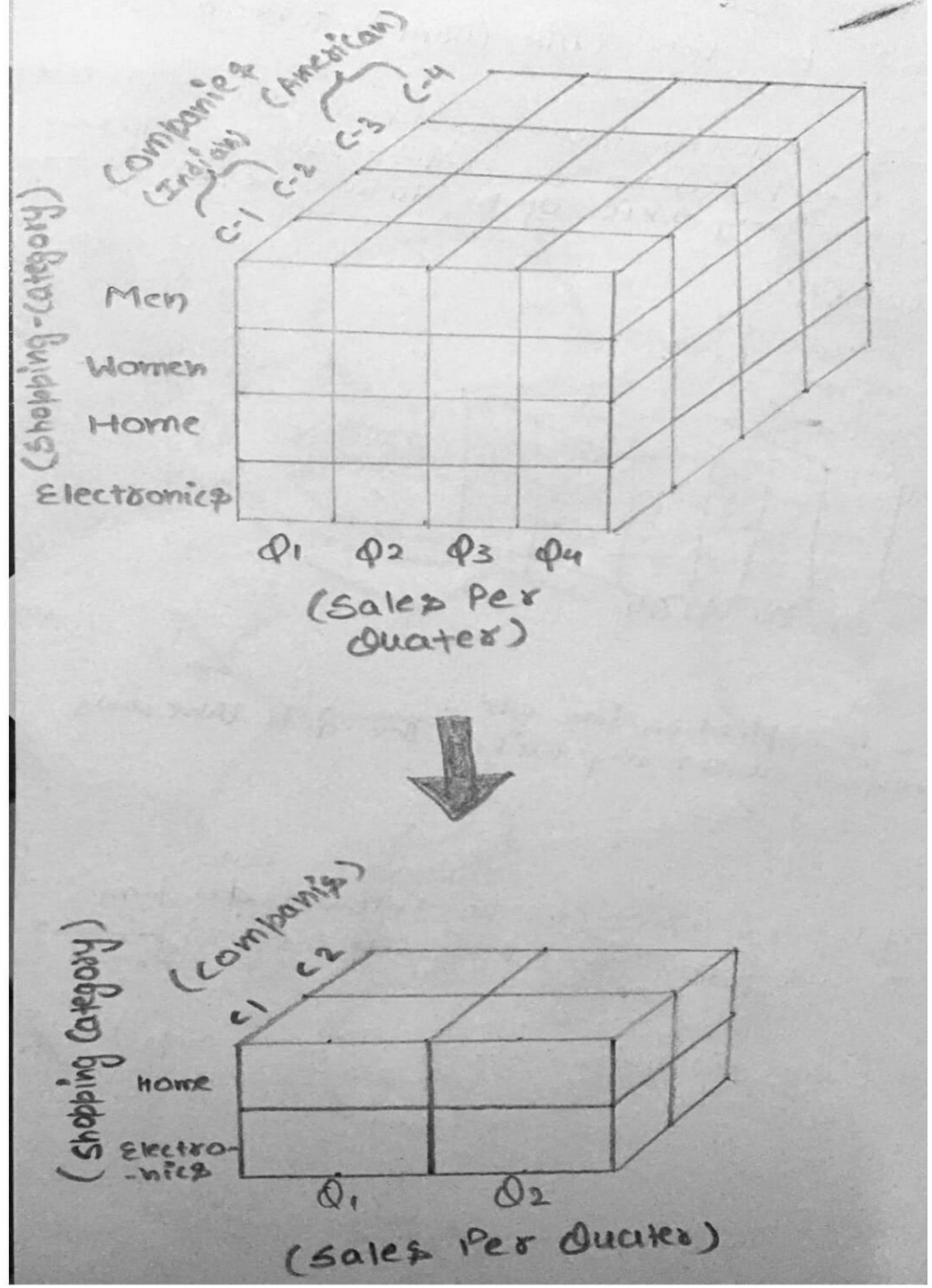

(Companies)
(Indian)
(American)
C-1
C-2
C-3
C-4
(Shopping-Category)
Men
Women
Home
Electronics
Q1
Q2
Q3
Q4
(Sales Per Quater)


(Companies)
C1,C2
C3,C4
(Shopping-Category)
Men
Women
Home
Electronics
Q1
Q2
Q3
Q4
(Sales Per Quater)


(Companies)
(Indian)
(American)
C-1
C-2
C-3
C-4
(Shopping-Category)
Men
Women
Home
Electronics
Q1
Q2
Q3
Q4
(Sales Per Quater)


(Companies)
C-1
C-2
C-3
C-4
(Shopping-Category)
Men
Clothing
Footwear
Women
Clothing
Footwear
Home
Decor
Appliances
Electronics
Mobile
Camera
Q1
Q2
Q3
Q4
(Sales Per Quater)


(Companies
(Indian)
(American)
C-1
C-2
C-3
C-4
(Shopping-Category)
Men
Women
Home
Electronics
Q1
Q2
Q3
Q4
(Sales Per Quater)
(Shopping Category)
Men
Women
Home
Electro-nics
Q1
Q2
Q3
Q4
(Sales Per Quater)


(Companies)
(Indian)
(American)
C-1
C-2
C-3
C-4
(Shopping-Category)
Men
Women
Home
Electronics
Q1
Q2
Q3
Q4
(Sales Per Quater)
(Companies)
C1
C2
(Shopping Category)
Home
Electro-nics
Q1
Q2
(Sales Per Quater)

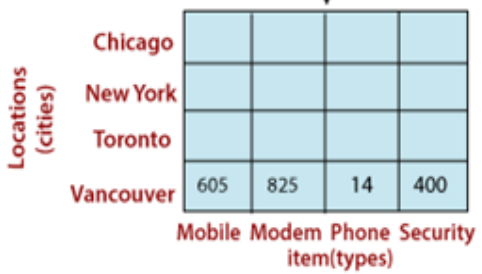
Dice

The dice operation describes a subcube by operating a selection on two or more dimension.

Dice for (location="Toronto" or "Vancouver") and(time="Q1"or"Q2")and (item="Mobile" or "Modem")

Slice

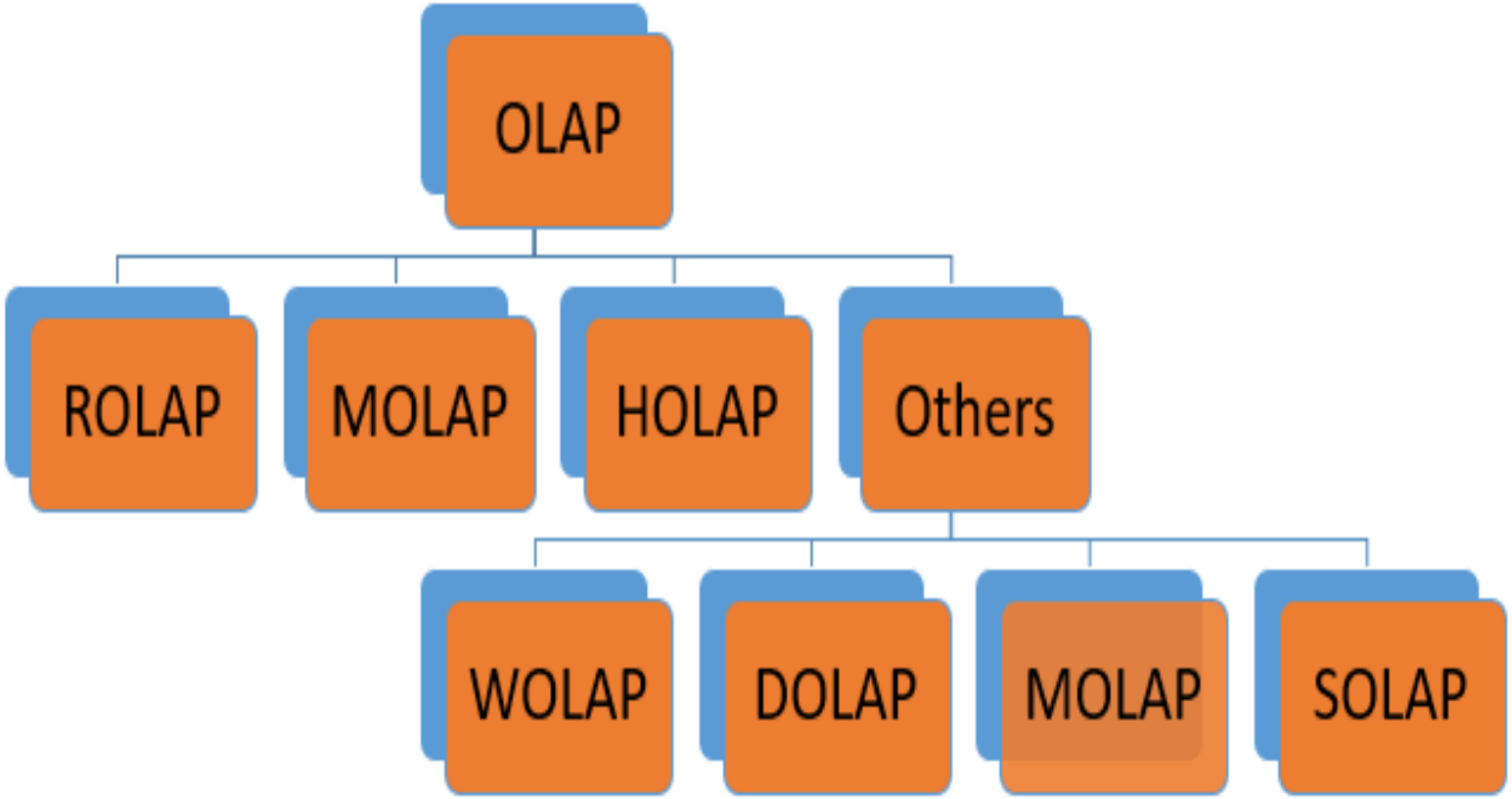
slice for time ="Q1"



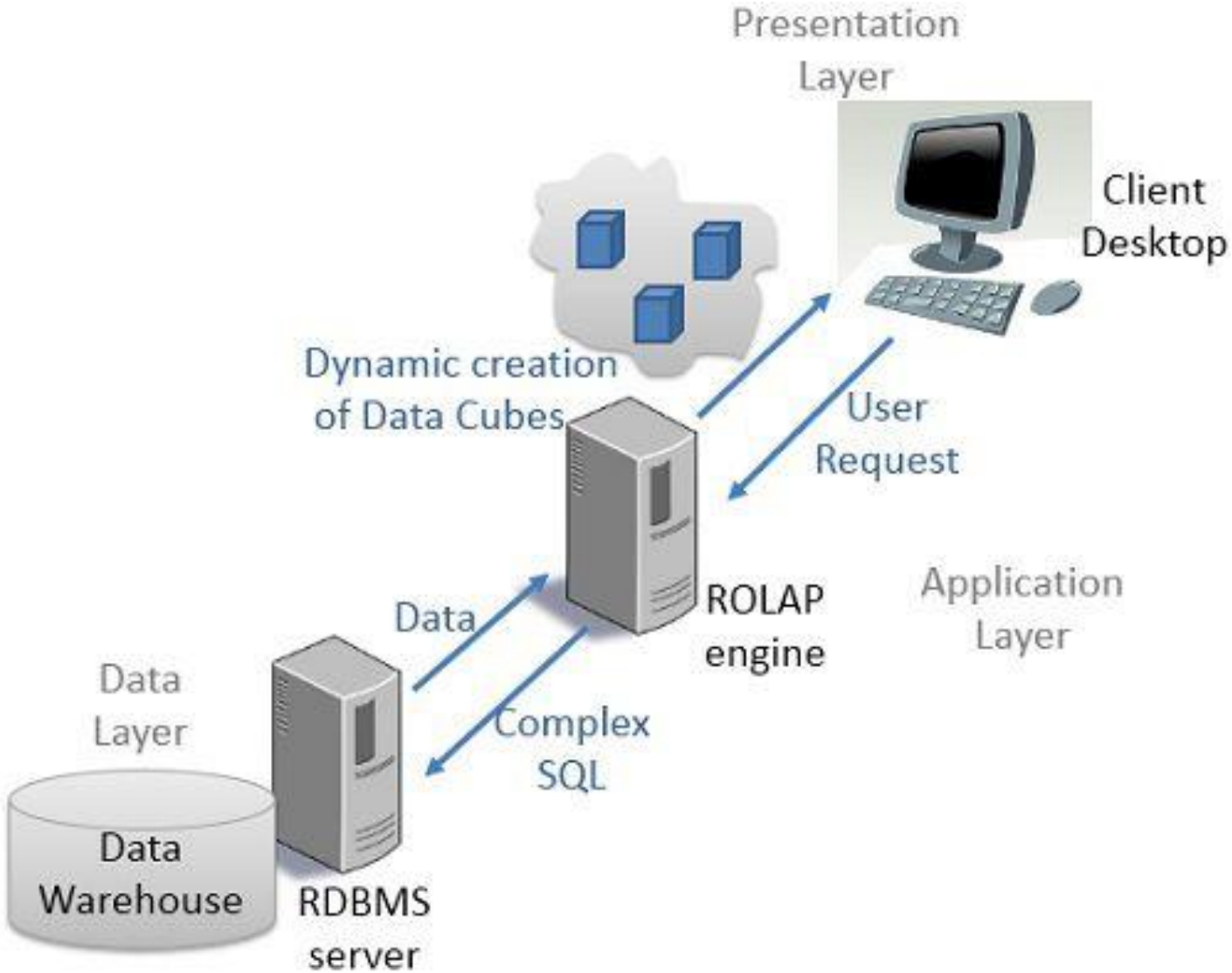
For example, a slice operation is executed when the customer wants a selection on one dimension of a three-dimensional cube resulting in a two-dimensional site. So, the Slice operations perform a selection **on one dimension of the given cube,** thus resulting in a subcube.

Here Slice is functioning for the dimensions "time" using the criterion time = "Q1".
It will form a new sub-cubes by selecting one or more dimensions.

OLAP Models



- **ROLAP**
- ROLAP works with data that exist in a relational database. Facts and dimension tables are stored as relational tables. It also allows multidimensional analysis of data and is the fastest growing OLAP.


Presentation
Layer
Client
Desktop
Dynamic creation
of Data Cubes
User
Request
Data
ROLAP
engine
Application
Layer
Data
Layer
Complex
SQL
Data
Warehouse
RDBMS
server

ROLAP Model

Advantages of ROLAP model:

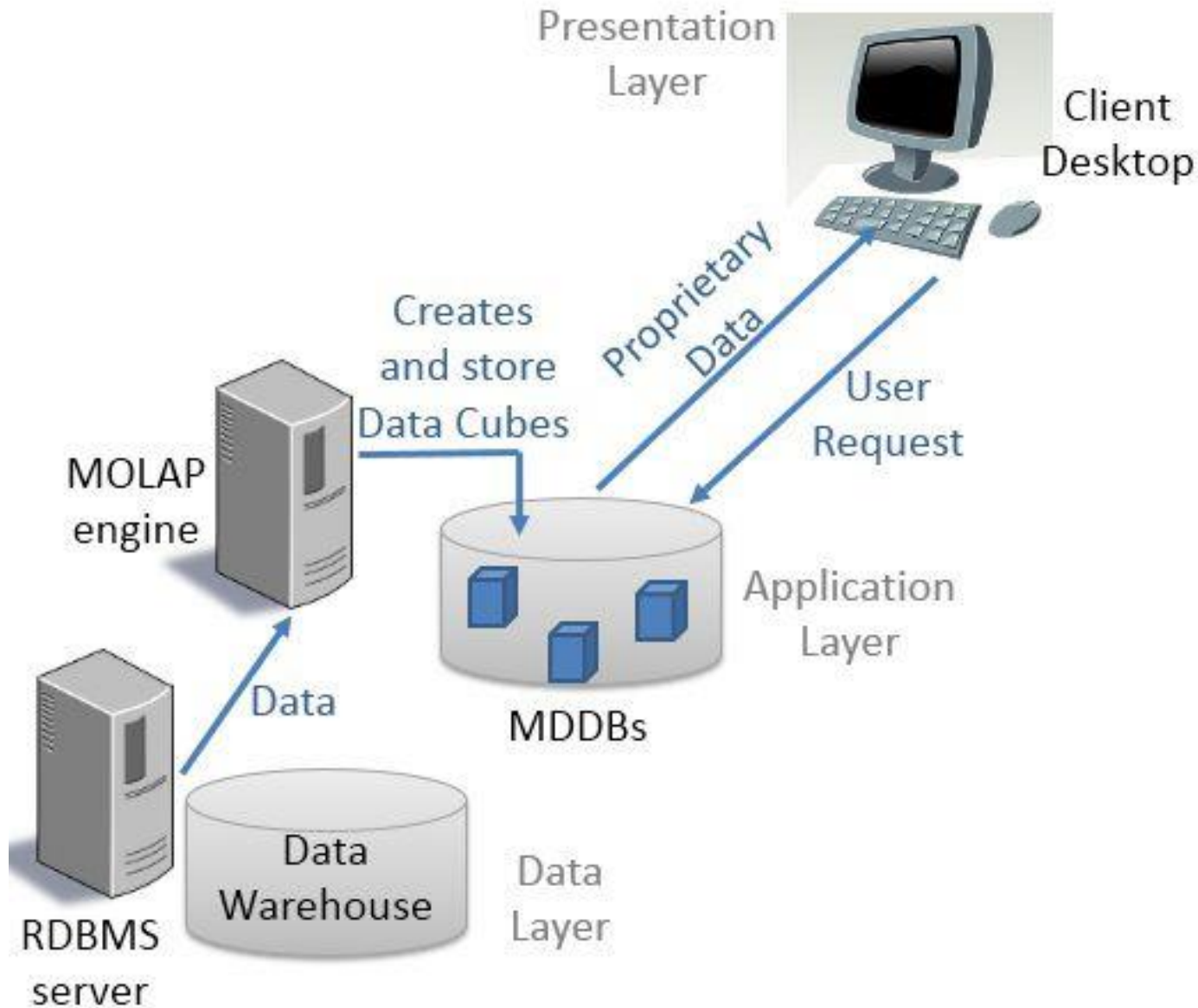
- **High data efficiency.** It offers high data efficiency because query performance and access language are optimized particularly for the multidimensional data analysis.
- **Scalability.** This type of OLAP system offers scalability for managing large volumes of data, and even when the data is steadily increasing.

Drawbacks of ROLAP model:

- **Demand for higher resources:** ROLAP needs high utilization of manpower, software, and hardware resources.
- **Aggregately data limitations.** ROLAP tools use SQL for all calculation of aggregate data. However, there are no set limits to the for handling computations.
- **Slow query performance.** Query performance in this model is slow when compared with MOLAP

Multidimensional OLAP (MOLAP)

- Multidimensional OLAP (MOLAP) is a classical OLAP that facilitates data analysis by using a multidimensional data cube. Data is pre-computed,pre-summarized, and stored in a MOLAP.
- Using a MOLAP, a user can use multidimensional view data with different facets.
- MOLAP has all possible combinations of data already stored in a multidimensional array.
- MOLAP can access this data directly. Hence, MOLAP is faster compared to Relational Online Analytical Processing (ROLAP).


Presentation
Layer
Client
Desktop
Proprietary
Data
Creates
and store
Data Cubes
User
Request
MOLAP
engine
Application
Layer
MDDBs
Data
Data
Warehouse
Data
Layer
RDBMS
server

MOLAP Model

MOLAP Advantages

- MOLAP can manage, analyze and store considerable amounts of multidimensional data.
- Fast Query Performance due to optimized storage, indexing, and caching.
- Smaller sizes of data as compared to the relational database.
- Automated computation of higher level of aggregates data.
- Help users to analyze larger, less-defined data.
- MOLAP is easier to the user that's why It is a suitable model for inexperienced users.
- MOLAP cubes are built for fast data retrieval and are optimal for slicing and dicing operations.
- All calculations are pre-generated when the cube is created.

MOLAP Disadvantages

- One major weakness of MOLAP is that it is less scalable than ROLAP as it handles only a limited amount of data.
- The MOLAP also introduces data redundancy as it is resource intensive
- MOLAP Solutions may be lengthy, particularly on large data volumes.
- MOLAP products may face issues while updating and querying models when dimensions are more than ten.
- MOLAP is not capable of containing detailed data.
- The storage utilization can be low if the data set is highly scattered.
- It can handle the only limited amount of data therefore, it's impossible to include a large amount of data in the cube itself.

Hybrid OLAP

- Hybrid OLAP is a mixture of both ROLAP and MOLAP. It offers fast computation of MOLAP and higher scalability of ROLAP. HOLAP uses two databases.
- Aggregated or computed data is stored in a multidimensional OLAP cube
- Detailed information is stored in a relational database.

Benefits of Hybrid OLAP:

- This kind of OLAP helps to economize the disk space, and it also remains compact which helps to avoid issues related to access speed and convenience.
- Hybrid HOLAP's uses cube technology which allows faster performance for all types of data.
- ROLAP are instantly updated and HOLAP users have access to this real-time instantly updated data. MOLAP brings cleaning and conversion of data thereby improving data relevance. This brings best of both worlds.

Drawbacks of Hybrid OLAP:

- Greater complexity level**:** The major drawback in HOLAP systems is that it supports both ROLAP and MOLAP tools and applications. Thus, it is very complicated.
- Potential overlaps**:** There are higher chances of overlapping especially into their functionalities.

Key Differences Between ROLAP and MOLAP

- ROLAP stands for Relational Online Analytical Processing whereas;
- In both the cases, ROLAP and MOLAP data is stored in the main warehouse. In ROLAP data is directly fetched from the main warehouse
- In ROLAP, data is stored in the form of relational tables
- ROLAP deals with large volumes of data whereas,.

- MOLAP stands for Multidimensional Online Analytical Processing.
- In both the cases, ROLAP and MOLAP data is stored in the main warehouse,in MOLAP data is fetched from the proprietary databases MDDBs.
- In MOLAP data is stored in the form of a multidimensional array made of data cubes.
- MOLAP deals with limited data summaries kept in MDDBs

- ROLAP engines use complex SQL to fetch data from the data warehouse.
- ROLAP engine creates a multidimensional view of data dynamically
- As ROLAP creates a multidimensional view of data dynamically, it is slower than MOLAP
- Mondrian (Open Source) offered by Pentaho, Oracle OLAP.

- MOLAP engine creates prefabricated and precalculated datacubes to present multidimensional view of data to a user
- MOLAP statically stores multidimensional view of data in proprietary databases MDDBs for a user to view it from there.
- MOLAP which do not waste time in creating a multidimensional view of data. And thus it is faster
- Commercial products that use MOLAP:Microsoft analysis Services,ESSbase,and TM1,Palo

Type of OLAP	Explanation
Relational OLAP(ROLAP):	ROLAP is an extended RDBMS along with multidimensional data mapping to perform the standard relational operation.
Multidimensional OLAP (MOLAP)	MOLAP Implementes operation in multidimensional data.
Hybrid OnlineAnalytical Processing (HOLAP)	In HOLAP approach the aggregated totals are stored in a multidimensional database while the detailed data is stored in the relational database. This offers both data efficiency of the ROLAP model and the performance of the MOLAP model.

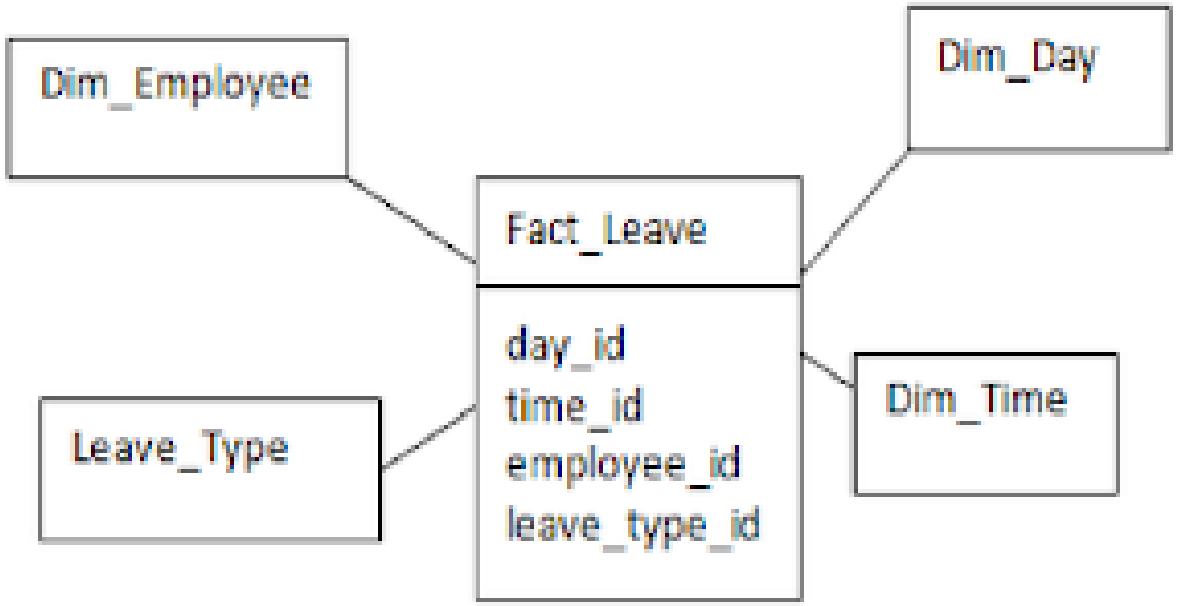
Type of OLAP	Explanation
Desktop OLAP (DOLAP)	In Desktop OLAP, a user downloads a part of the data from the database locally, or on their desktop and analyze it. DOLAP is relatively cheaper to deploy as it offers very few functionalities compares to other OLAP systems.
Web OLAP (WOLAP)	Web OLAP which is OLAP system accessible via the web browser. WOLAP is a three-tiered architecture. It consists of three components: client, middleware, and a database server.
Mobile OLAP:	Mobile OLAP helps users to access and analyze OLAP data using their mobile devices
Spatial OLAP :	SOLAP is created to facilitate management of both spatial and non-spatial data in a Geographic Information system (GIS)

Fact-less Fact Tables

- There are *two types of factless fact tables:*
  - those that describe events, and
  - those that describe conditions.
-

Factless fact tables for Events

- This factless fact table is a table that records an event.
- Many event-tracking tables in dimensional data warehouses turn out to be factless.
- Events or activities occur that you wish to track, but you find no measurements. In situations like this, build a standard transaction-grained fact table that contains no facts.
-

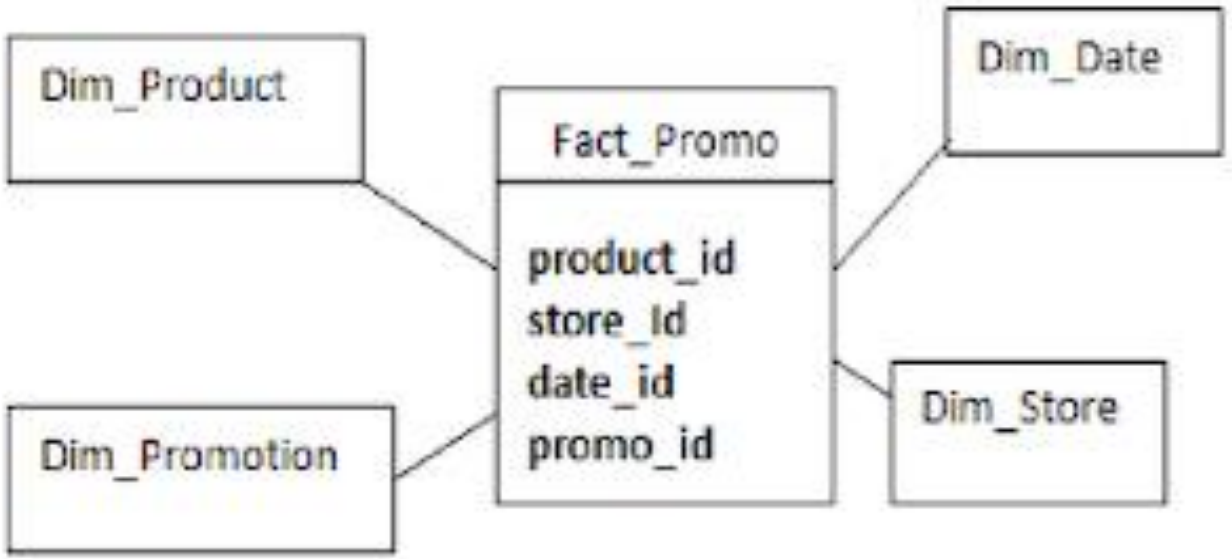


- Whenever an employee takes leave, a record is created with the dimensions.
- Using the fact FACT_LEAVE we can answer many questions like
  - Number of leaves taken by an employee
  - Type of leave an employee takes
  - Details of the employee who took leave

Factless fact tables for Conditions

- Factless fact tables are also used to model conditions or other important relationships among dimensions.
- In these cases, there are no clear transactions or events. It is used to support negative analysis report.
- For example a Store that did not sell a product for a given period. To produce such report, you need to have a fact table to capture all the possible combinations. You can then figure out what is missing.
- This kind of factless fact table is used to track conditions, coverage or eligibility. It is also called a "coverage table."

- fact_promo gives the information about the products which have promotions but still did not sell



- This fact answers the below questions:
  - To find out products that have promotions.
  - To find out products that have promotion that sell.
  - The list of products that have promotion but did not sell.

- A fact-less-fact table can only answer ‘optimistic’ queries (positive query) but cannot answer a negative query.
- Coverage fact is used to support negative analysis reports.
- For example, an electronic store did not sell any product for give period of time.
- If you consider the student-teacher relation table, the event capturing fact table cannot answer ‘which teacher did not teach any student?’ Coverage fact attempts to answer this question by adding extra flag 0 for negative condition and 1 for positive condition.
- If the student table has 20 records and teacher table has 3 records then coverage fact table will store 20 * 3 = 60 records for all possible combinations. If any teacher is not teaching particular student then that record will have flag 0 in it.