Data Warehousing & Mining

UNIT – I

Syllabus of Unit - I

- DSS-Uses, definition, Operational Database.
- Introduction to DATA Warehousing. Data-Mart,
- Concept of Data-Warehousing,
- Multi Dimensional Database Structures.
- Client/Server Computing Model & Data Warehousing
- Parallel Processors & Cluster Systems. Distributed DBMS implementations.

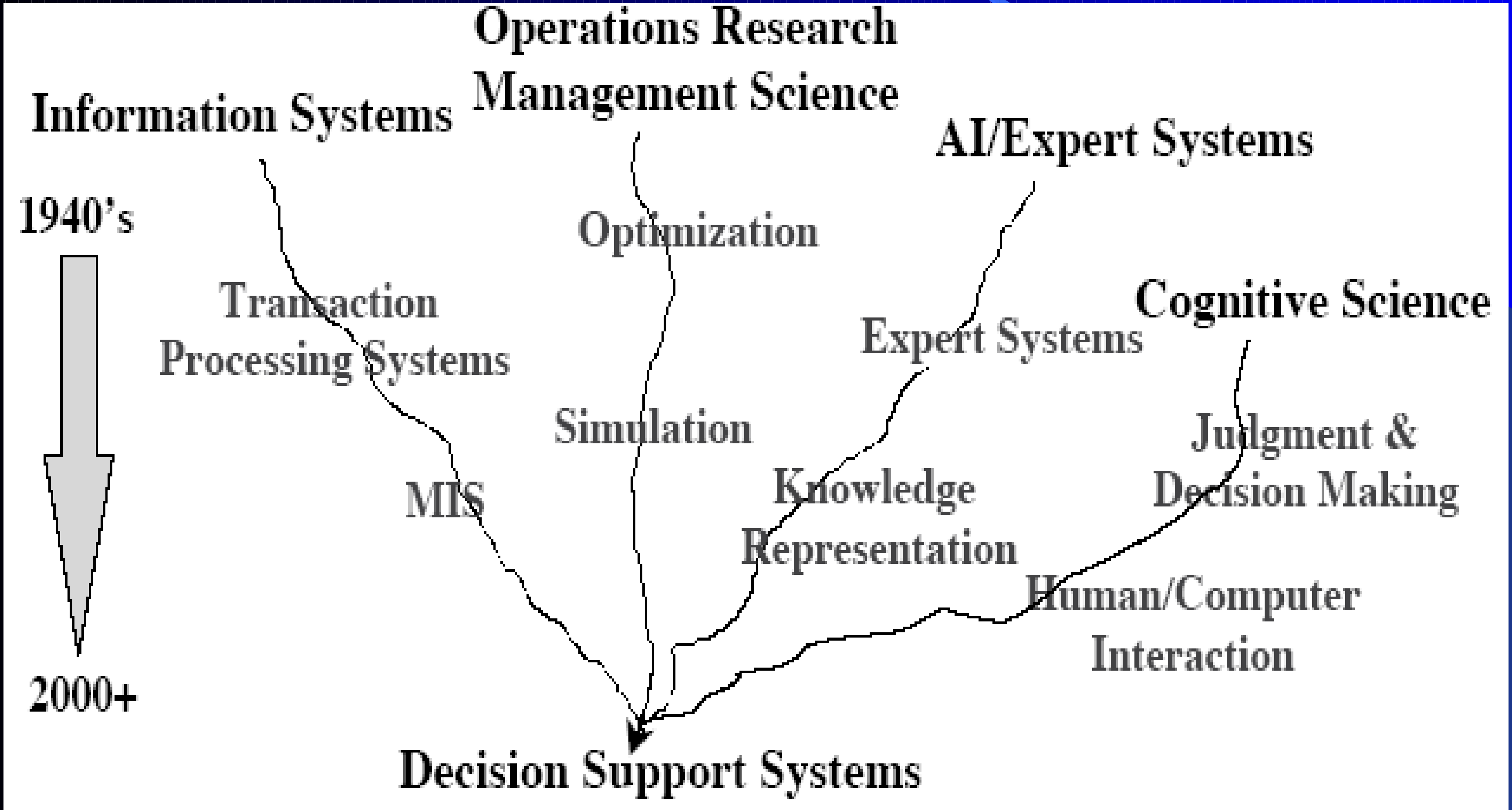
Introduction – Decision Support System (DSS)

- A **Decision Support System (DSS)** is an interactive computer-based system or subsystem intended to help decision makers use communications technologies, data, documents, knowledge and/or models to identify and solve problems, complete decision process tasks, and make decisions.
- It is clear that DSS belong to an environment with multidisciplinary foundations, including (but not exclusively):
  - Database research,
  - Artificial intelligence,
  - Human-computer interaction,
  - Simulation methods,
  - Software engineering, and
  - Telecommunications.

DSS

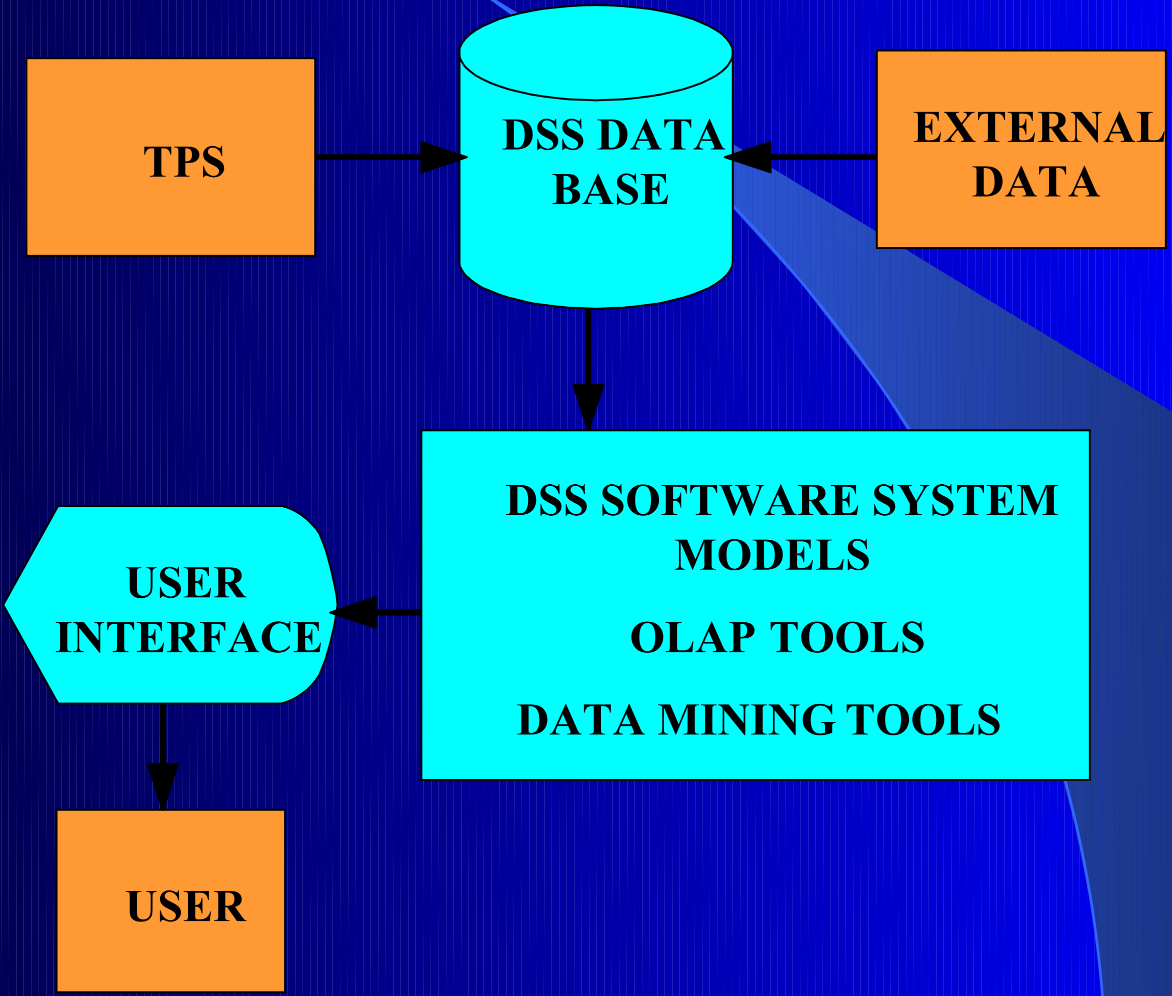
- A **Decision Support System (DSS)** is a computer-based information system that supports business or organizational decision-making activities.
- DSSs serve the management, operations, and planning levels of an organization (usually mid and higher management) and help to make decisions, which may be rapidly changing and not easily specified in advance (Unstructured and Semi-Structured decision problems).
- Decision support systems can be either fully computerized, human or a combination of both.

Historical Evolution of DSS



Typical DSS Architecture

- **TPS**: transaction processing system
- MODEL: representation of a problem
- **OLAP:** on-line analytical processing
- **USER INTERFACE:** how user enters problem & receives answers
- **DSS DATABASE:** current data from applications or groups
- **DATA MINING:** technology for finding relationships in large data bases for prediction

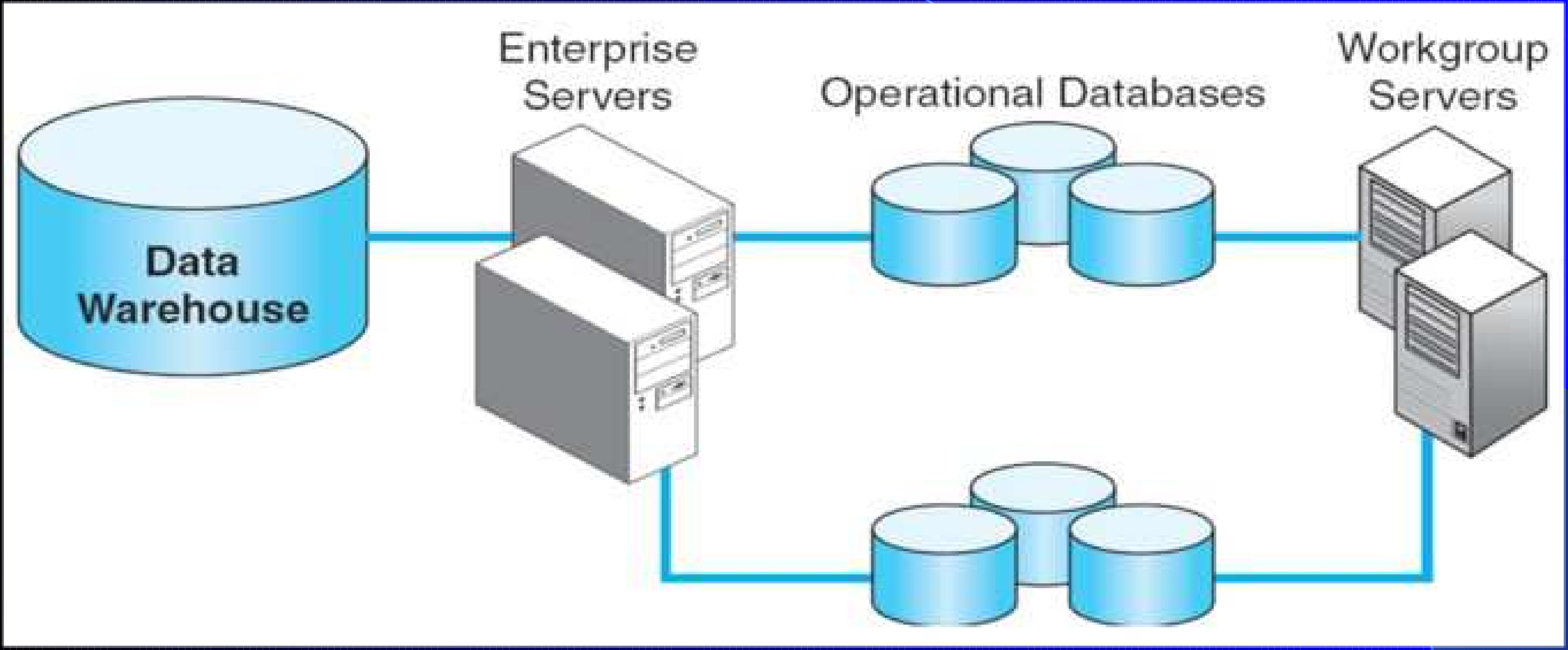


Why DSS?

- Increasing complexity of decisions
  - Technology
  - Information:
    - "Data, data everywhere, and not the time to think!"
  - Number and complexity of options
  - Pace of change
- Increasing availability of computerized support
  - Inexpensive high-powered computing
  - Better software
  - More efficient software development process
- Increasing usability of computers

Operational Databases

- Operational database management systems (also referred to as OLTP databases), are used to manage dynamic data in real-time.
- These types of databases allow you to do more than simply view archived data. Operational databases allows to modify that data (add, change or delete data), doing it in real-time.
- Since the early 90's, the operational database software market has been largely taken over by SQL engines.
- Today, the operational DBMS market (formerly OLTP) is evolving dramatically, with new, innovative entrants and incumbents supporting the growing use of unstructured data and NoSQL DBMS engines, as well as XML databases and NewSQL databases.
- Operational databases are increasingly supporting distributed database architecture that provides high availability and fault tolerance through replication and scale out ability.


Enterprise Servers
Operational Databases
Workgroup Servers
Data Warehouse

Differences between the Databases and Data Warehouses

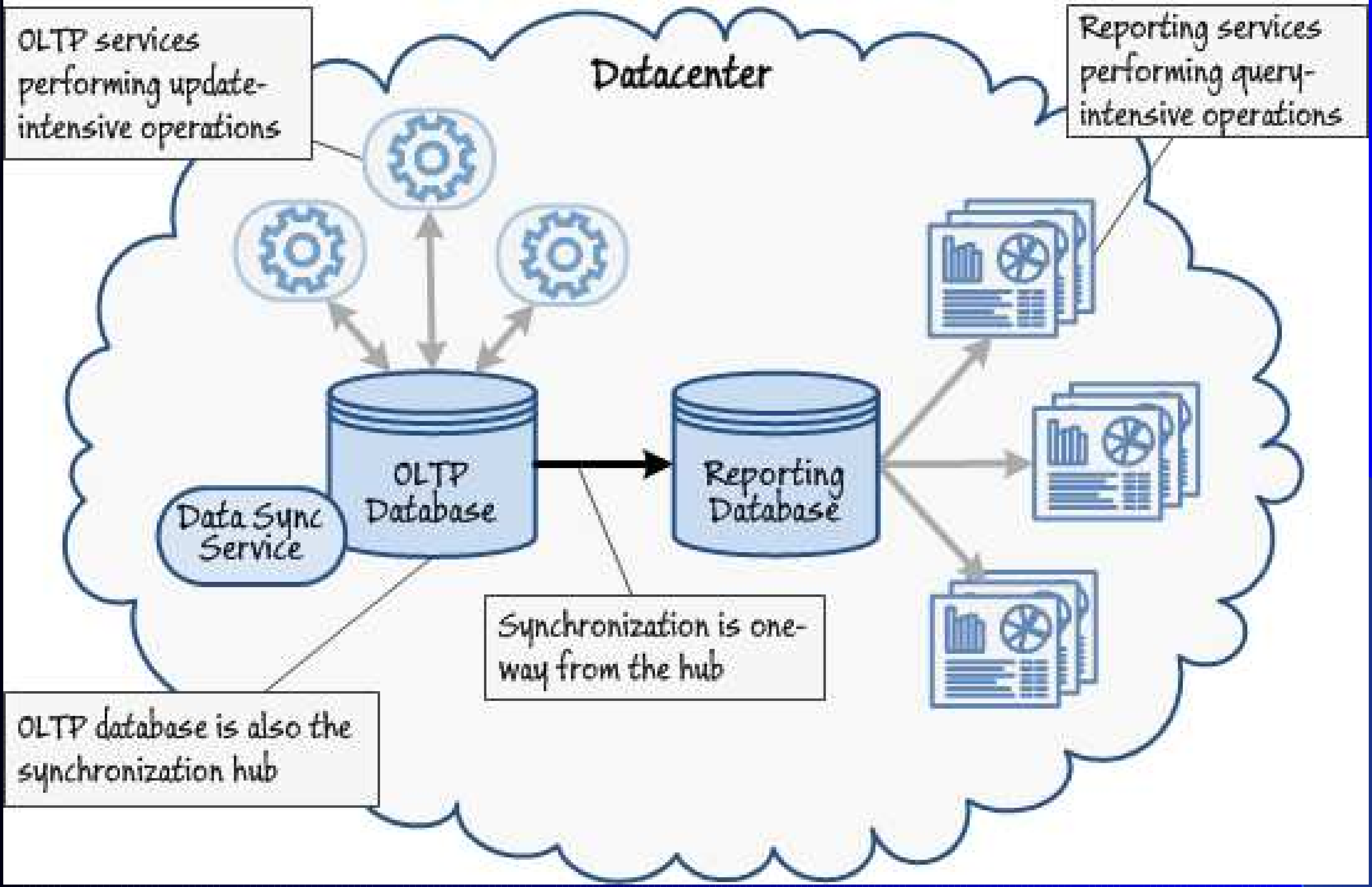
FEATURES	DATABASE	DATA WAREHOUSE
Characteristic	It is based on Operational Processing.	It is based on Informational Processing.
Data	It mainly stores the Current data which always guaranteed to be up-to-date.	It usually stores the Historical data whose accuracy is maintained over time.
Function	It is used for day-to-day operations.	It is used for long-term informational requirements and decision support.
User	The common users are clerk, DBA, database professional.	The common users are knowledge worker (e.g., manager, executive, analyst)
Unit of work	Its work consists of short and simple transaction.	The operations on it consists of complex queries..
Focus	The focus is on "Data IN"	The focus is on "Information OUT"
Orientation	The orientation is on Transaction.	The orientation is on Analysis.
DB design	The designing of database is ER based and application-oriented.	The designing is done using star/snowflake schema and its subject-oriented.
Summarization	The data is primitive and highly detailed.	The data is summarized and in consolidated form.
View	The view of the data is flat relational.	The view of the data is multidimensional.

FEATURES	DATABASE	DATA WAREHOUSE
Function	It is used for day-to-day operations.	It is used for long-term informational requirements and decision support.
User	The common users are clerk, DBA, database professional.	The common users are knowledge worker (e.g., manager, executive, analyst)
Access	The most frequent type of access type is read/write.	It mostly use the read access for the stored data.
Operations	The main operation is index/hash on primary key.	For any operation it needs a lot of scans.
Number of records accessed	A few tens of records.	A bunch of millions of records.
Number of users	In order of thousands.	In the order of hundreds only.
DB size	100 MB to GB.	100 GB to TB.
Priority	High performance, high availability	High flexibility, end-user autonomy
Metric	To measure the efficiency, transaction throughput is measured.	To measure the efficiency, query throughput and response time is measured.

DATA Warehousing - Introduction

A data warehouse is a subject-oriented, integrated, nonvolatile, time-variant collection of data in support of management's decisions.

- WH Inmon


OLTP services performing update-intensive operations
Datacenter
Reporting services performing query-intensive operations
OLTP Database
Data Sync Service
Reporting Database
Synchronization is one-way from the hub
OLTP database is also the synchronization hub

Data Warehouse Usage

- Three kinds of data warehouse applications
  - **Information processing**
    - supports querying, basic statistical analysis, and reporting using crosstabs, tables, charts and graphs
  - **Analytical processing**
    - multidimensional analysis of data warehouse data
    - supports basic OLAP operations, slice-dice, drilling, pivoting
  - **Data mining**
    - knowledge discovery from hidden patterns
    - supports associations, constructing analytical models, performing classification and prediction, and presenting the mining results using visualization tools.
- Differences among the three tasks

Data Warehouse: Subject-Oriented

- Organized around major subjects, such as customer, product, sales.

- **Focusing on the modeling and analysis of data for decision makers, not on daily operations or transaction processing.**

- Provide a **simple and concise view around particular subject issues by excluding data that are not useful in the decision support process**.

Data Warehouse—Integrated

- Constructed by integrating multiple, heterogeneous data sources
  - relational databases, flat files, on-line transaction records

- Data cleaning and data integration techniques are applied.
  - Ensure consistency in naming conventions, encoding structures, attribute measures, etc. among different data sources
    - E.g., Hotel price: currency, tax, breakfast covered, etc.
  - When data is moved to the warehouse, it is converted.

Data Warehouse—Time Variant

- The time horizon for the data warehouse is significantly longer than that of operational systems.
  - Operational database: current value data.
  - Data warehouse data: provide information from a historical perspective (e.g., past 5-10 years)
- Every key structure in the data warehouse
  - Contains an element of time, explicitly or implicitly
  - But the key of operational data may or may not contain "time element".

Data Warehouse—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*.

Data Warehouse vs. Heterogeneous DBMS

- Traditional heterogeneous DB integration:
  - Build wrappers/mediators on top of heterogeneous databases
  - Query driven approach
    - When a query is posed to a client site, a meta-dictionary is used to translate the query into queries appropriate for individual heterogeneous sites involved, and the results are integrated into a global answer set
    - Complex information filtering, compete for resources
- Data warehouse: update-driven, high performance
  - Information from heterogeneous sources is integrated in advance and stored in warehouses for direct query and analysis

Data Warehouse vs. Operational DBMS

- OLTP (on-line transaction processing)
  - Major task of traditional relational DBMS
  - Day-to-day operations: purchasing, inventory, banking, manufacturing, payroll, registration, accounting, etc.
- OLAP (on-line analytical processing)
  - Major task of data warehouse system
  - Data analysis and decision making
- Distinct features (OLTP vs. OLAP):
  - User and system orientation: customer vs. market
  - Data contents: current, detailed vs. historical, consolidated
  - Database design: ER + application vs. star + subject
  - View: current, local vs. evolutionary, integrated
  - Access patterns: update vs. read-only but complex queries

OLTP vs. OLAP

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

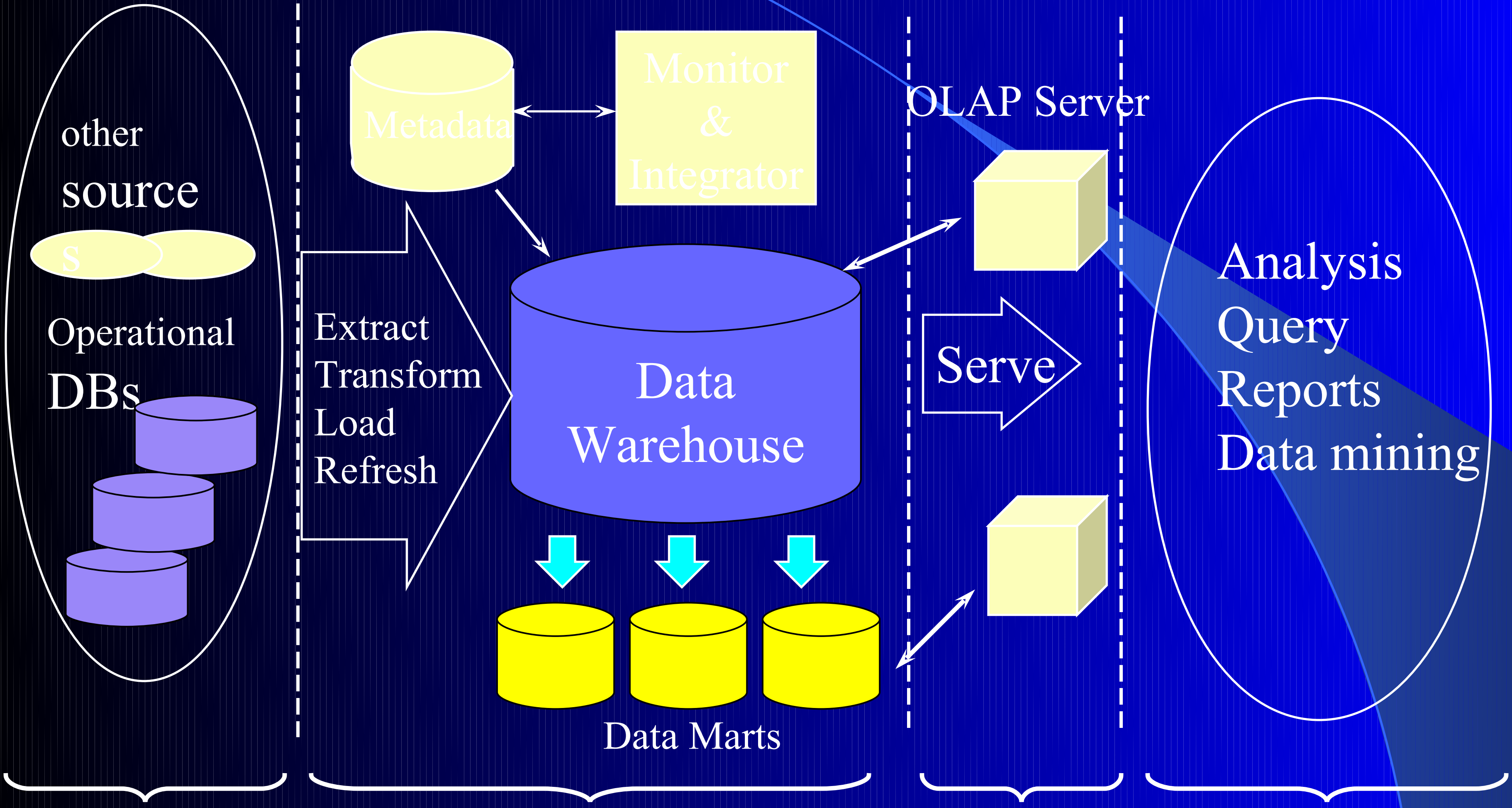
Why Separate Data Warehouse?

- High performance for both systems
  - DBMS— tuned for OLTP: access methods, indexing, concurrency control, recovery
  - Warehouse—tuned for OLAP: complex OLAP queries, multidimensional view, consolidation.
- Different functions and different data:
  - **missing data**: Decision support requires historical data which operational DBs do not typically maintain
  - **data consolidation:** DS requires consolidation (aggregation, summarization) of data from heterogeneous sources
  - **data quality:** different sources typically use inconsistent data representations, codes and formats which have to be reconciled

Data Mart

Concept of Data-Warehousing

Multi-Tiered Architecture



Data Sources

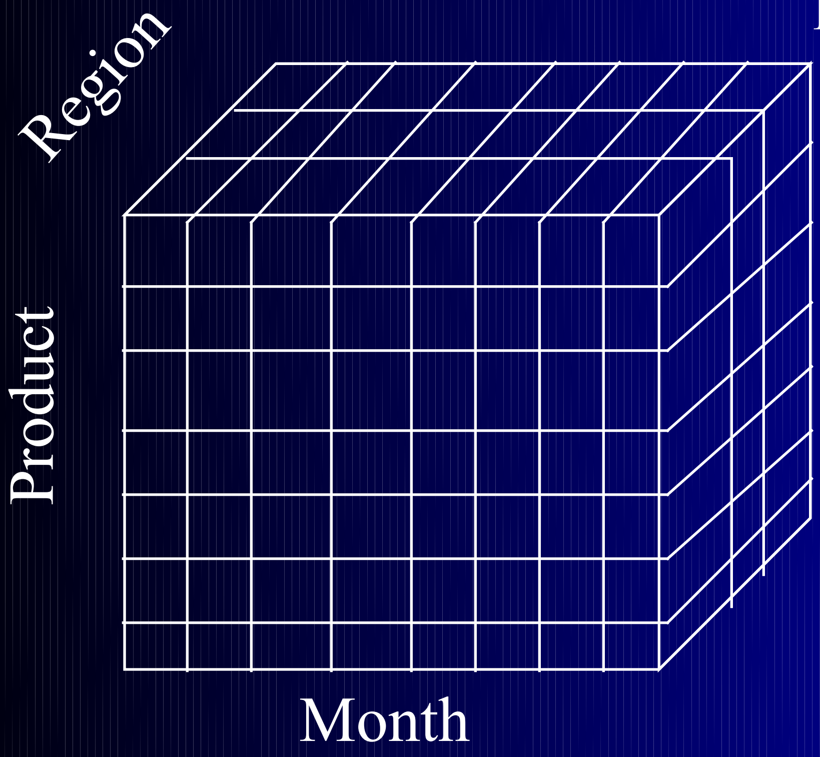
Data Storage

OLAP Engine

Front-End Tools

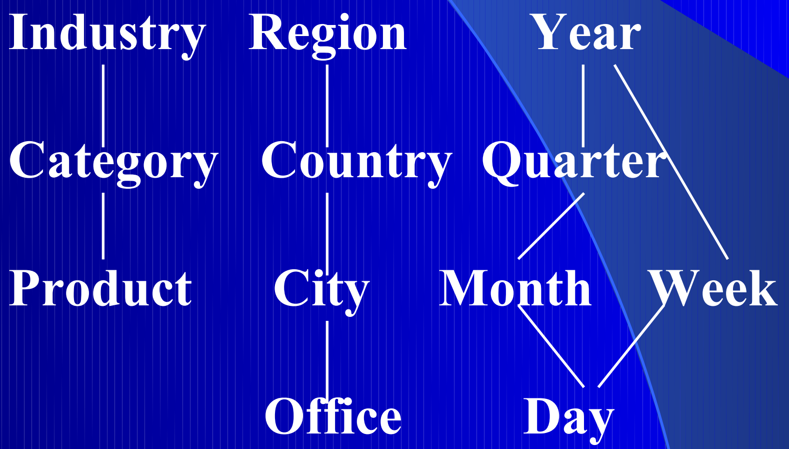
Multi Dimensional Database Structures

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



Dimensions: Product, Location, Time

Hierarchical summarization paths


Industry
Category
Product

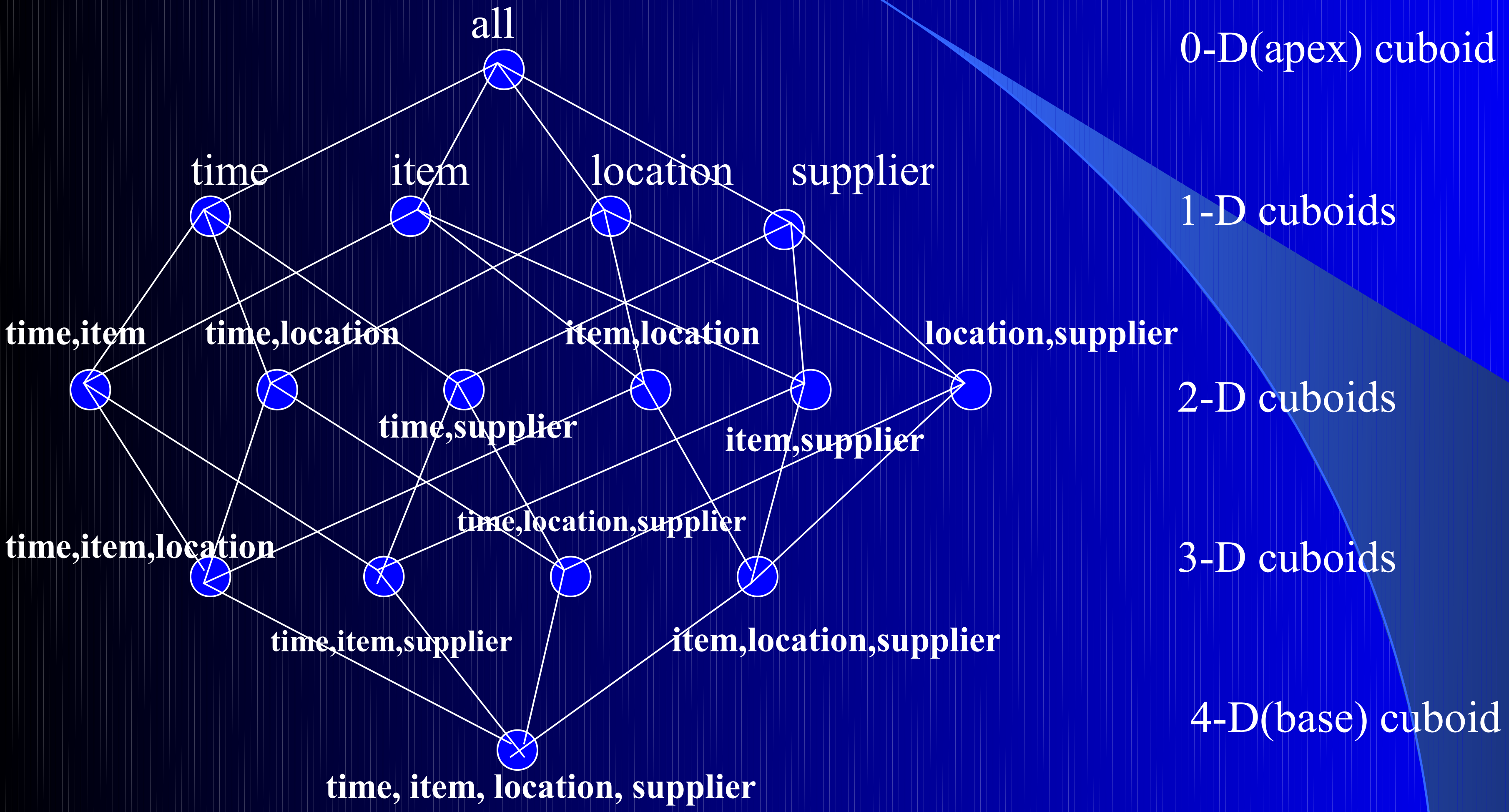
Region
Country
City
Office

Year
Quarter
Month
Week
Day

From Tables and Spreadsheets to Data Cubes

- A data warehouse is based on a **multidimensional data model** which views data in the form of a data cube
- A data cube, such as sales, allows data to be modeled and viewed in multiple dimensions
  - Dimension tables, such as item (item_name, brand, type), or time(day, week, month, quarter, year)
  - Fact table contains measures (such as dollars_sold) and keys to each of the related dimension tables
- In data warehousing literature, an n-D base cube is called a **base cuboid**. The top most 0-D cuboid, which holds the highest-level of summarization, is called the **apex cuboid**. The lattice of cuboids forms a **data cube**.

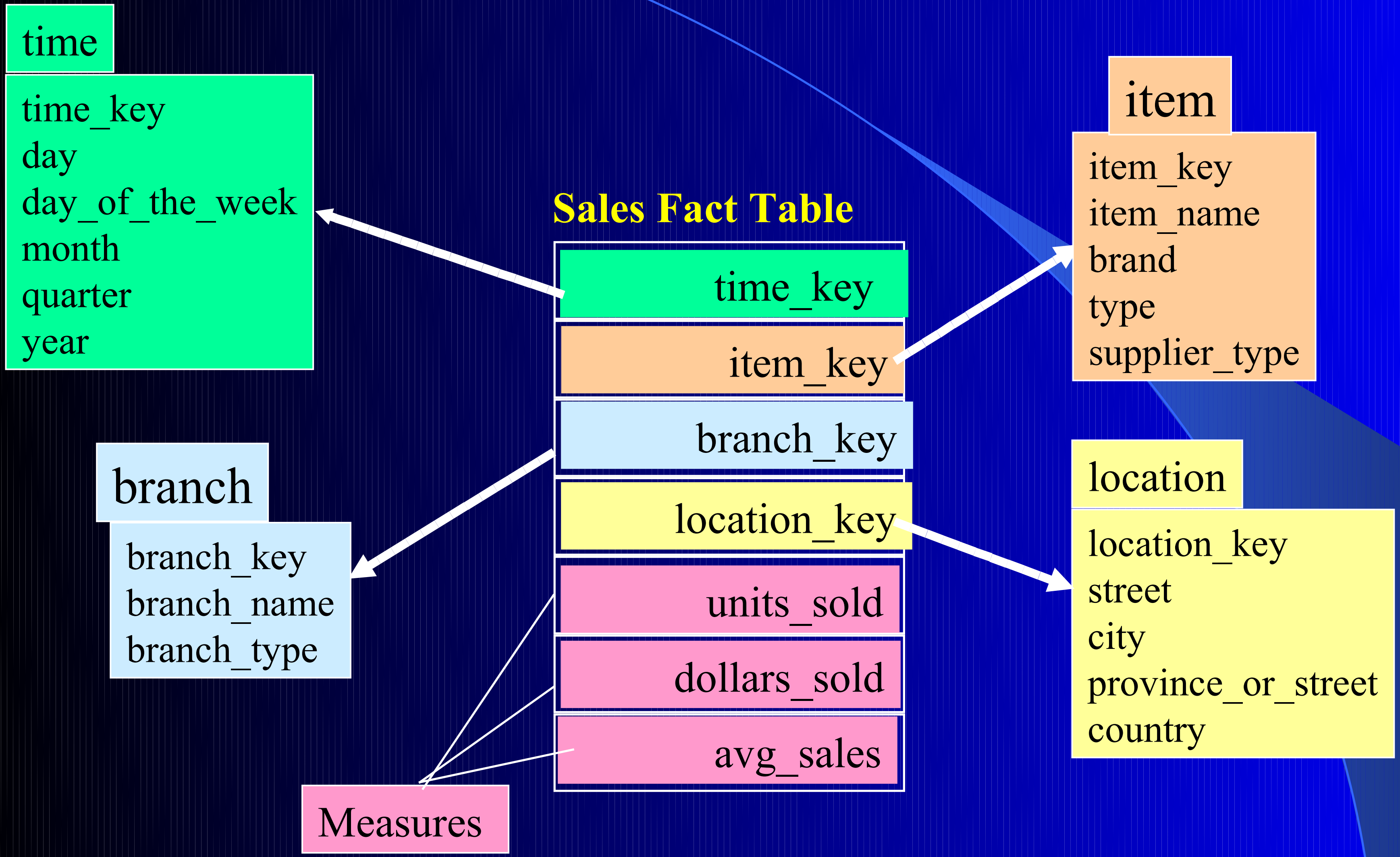
Cube: A Lattice of Cuboids



Conceptual Modeling of Data Warehouses

- Modeling data warehouses: dimensions & measures
  - **Star schema:** A fact table in the middle connected to a set of dimension tables
  - **Snowflake schema:** A refinement of star schema where some dimensional hierarchy is normalized into a set of smaller dimension tables, forming a shape similar to snowflake
  - **Fact constellations:** **Multiple fact tables share dimension tables**, viewed as a collection of stars, therefore called galaxy schema or fact constellation

Example of Star Schema



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

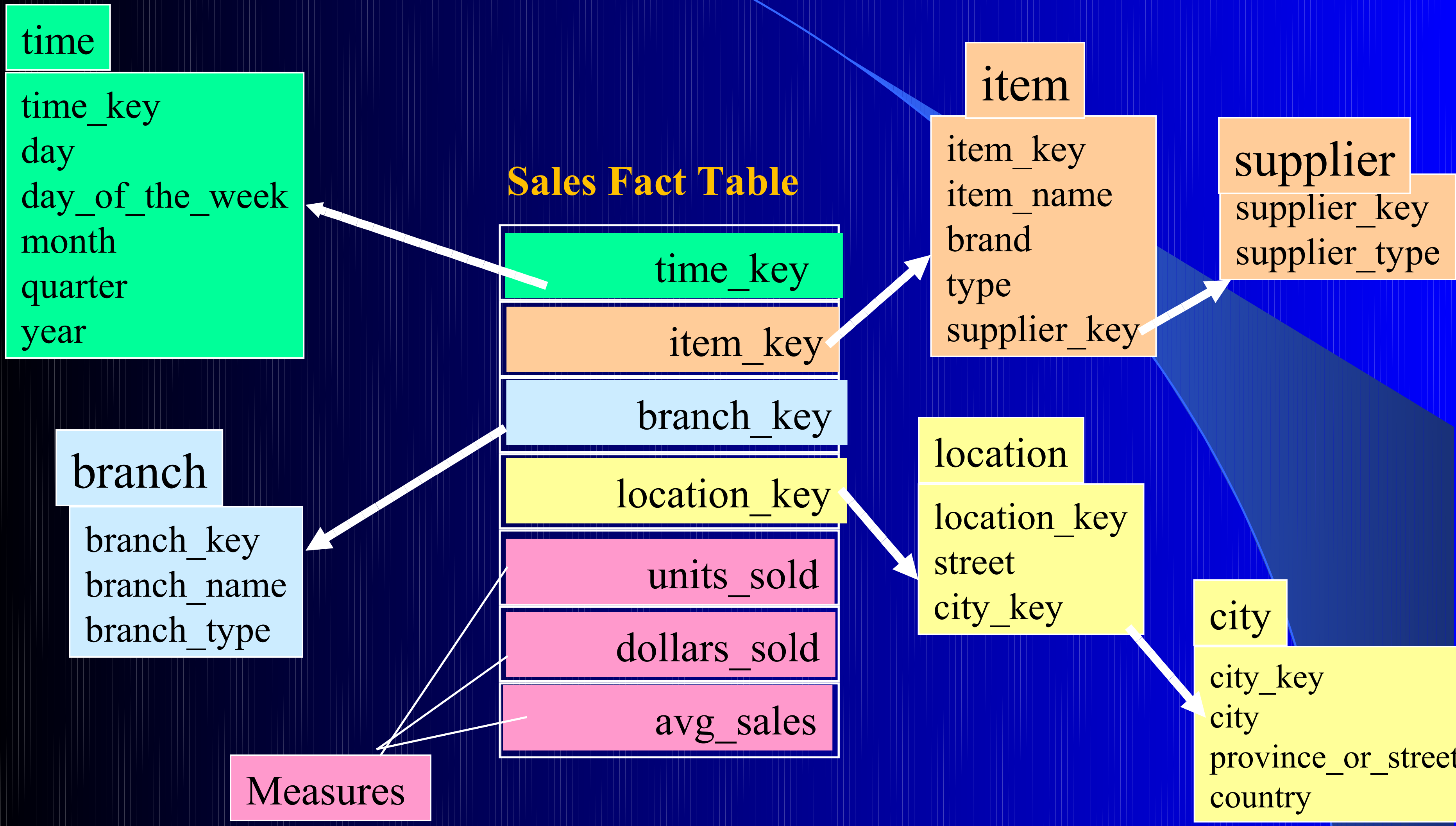
branch_key
branch_name
branch_type

location

location_key
street
city
province_or_street
country

Measures

Example of Snowflake 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

Measures

item

item_key
item_name
brand
type
supplier_key

supplier

supplier_key
supplier_type

branch

branch_key
branch_name
branch_type

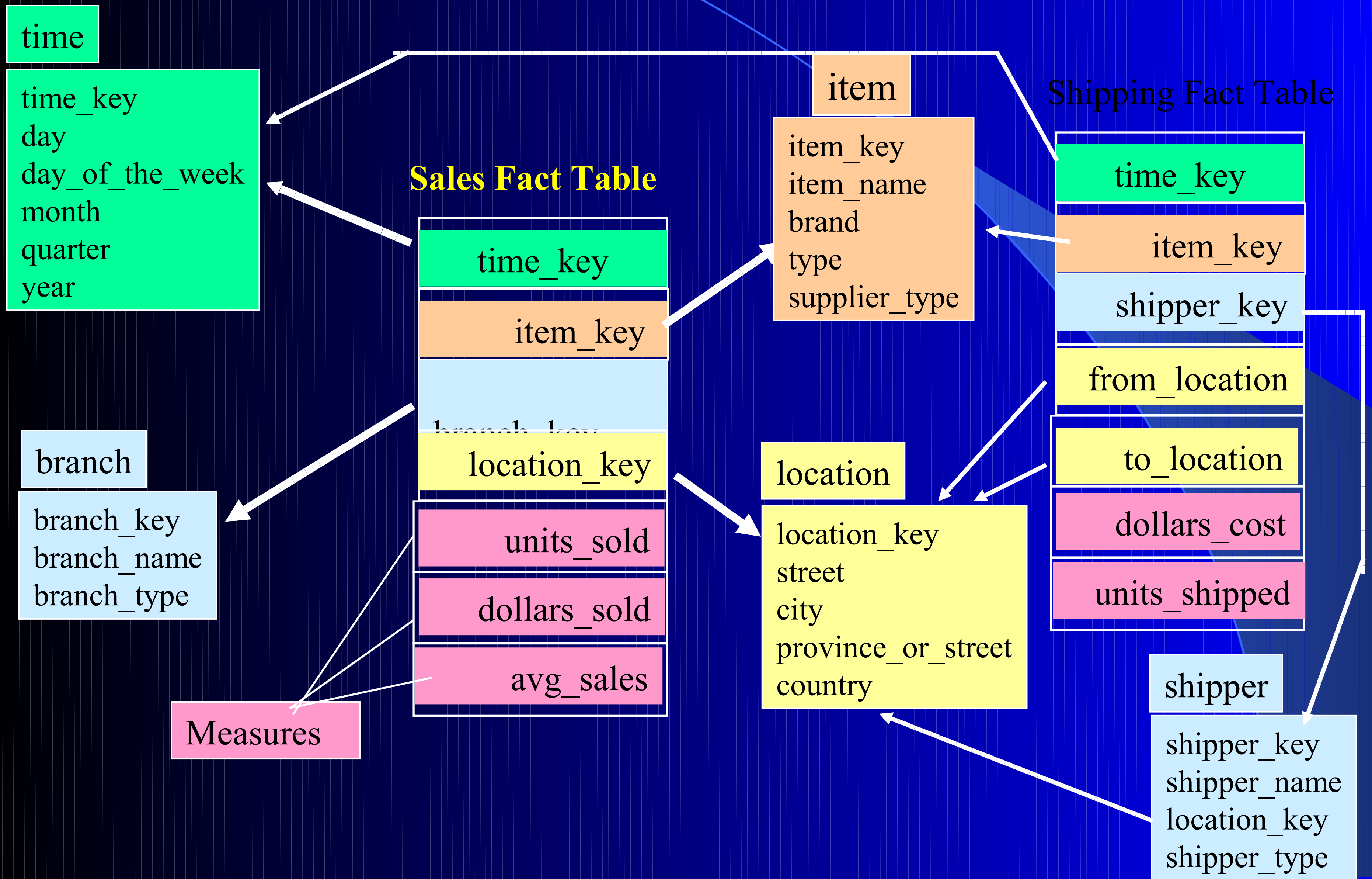
location

location_key
street
city_key

city

city_key
city
province_or_street
country

Example of Fact Constellation



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

Measures

item

item_key
item_name
brand
type
supplier_type

Shipping Fact Table

time_key
item_key
shipper_key
from_location
to_location
dollars_cost
units_shipped

branch

branch_key
branch_name
branch_type

location

location_key
street
city
province_or_street
country

shipper

shipper_key
shipper_name
location_key
shipper_type

Client/Server Computing Model & Data Warehousing

- The fundamental characteristic of client/server computing is distribution of computing resources (e.g. data, compute power) across different computers.
- The idea is to divide applications into logical segments (tasks) so that they are then performed on platforms most appropriate.
- A **client/server database system** increases processing power by separating the database management system from the application; the client as the front-end system handling the user interface and the server as the back-end system accessing the database, which cooperate to run an application.

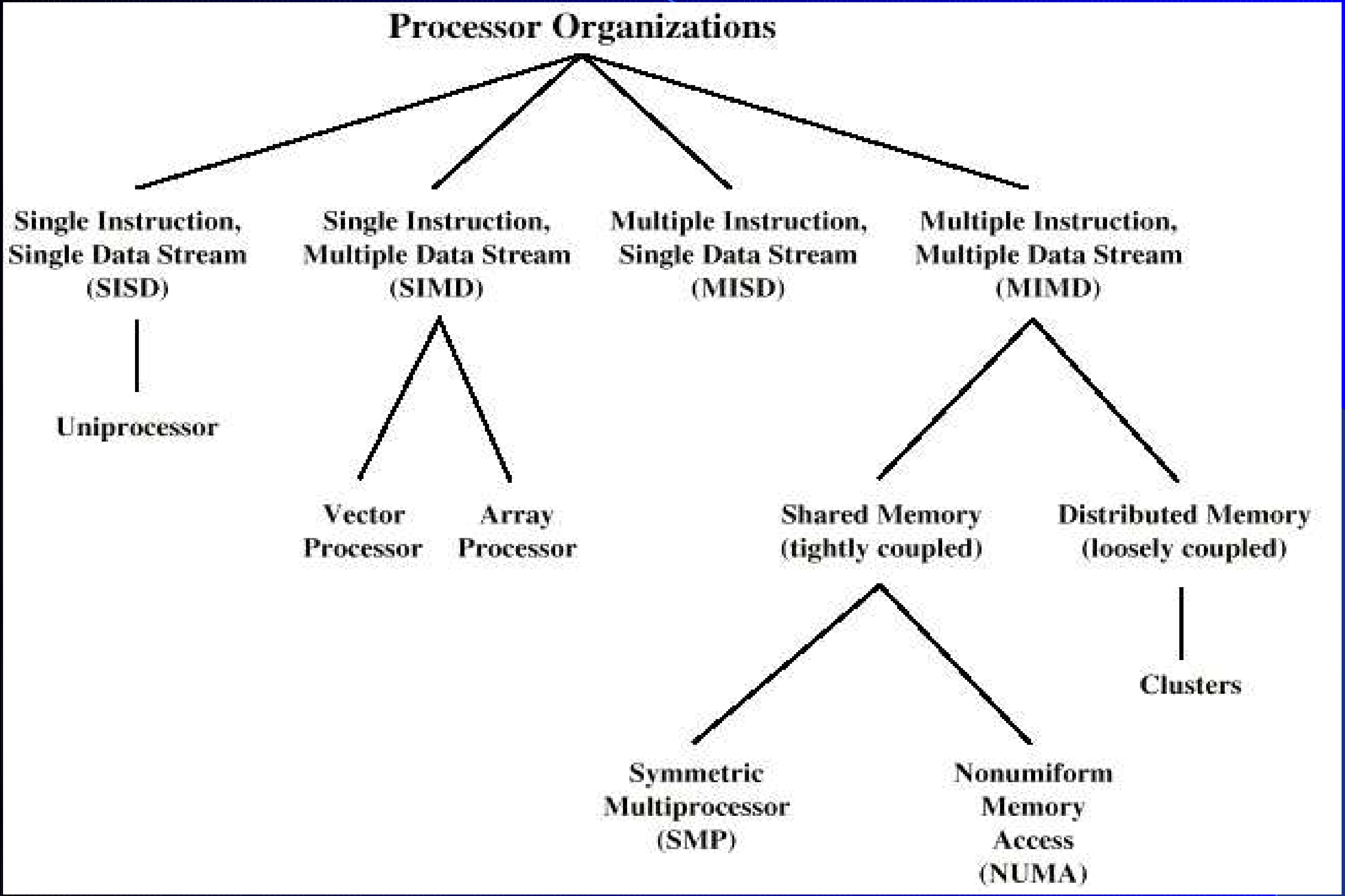
Contd….

- Data Warehousing is a continual process which enables a corporation to assemble operational and other data from a variety of internal and external sources, and transform that data into consistent, high-quality, business information, distribute that information to the points of maximum value within the organizations, and provide easy, flexible and fast access for busy non-technical users.

Reasons for using client/server

- Exploitation of centralised computing power /data capacity
- Scalability
- Performance
- Flexibility (in order to adjust to changing demands)
- GUI on desktop
- Protection of investment, strategic software, strategic data
- Client/server provides an integrated solution.

Parallel Processors & Cluster Systems



Processor Organizations

Single Instruction, Single Data Stream (SISD)

Uniprocessor

Single Instruction, Multiple Data Stream (SIMD)

Vector Processor

Array Processor

Multiple Instruction, Single Data Stream (MISD)

Multiple Instruction, Multiple Data Stream (MIMD)

Shared Memory (tightly coupled)

Symmetric Multiprocessor (SMP)

Nonumiform Memory Access (NUMA)

Distributed Memory (loosely coupled)

Clusters

Loosely Coupled - Clusters

- Collection of independent whole uni-processors or SMPs
  - Usually called nodes
- Interconnected to form a cluster
- Working together as unified resource
  - Illusion of being one machine
- Communication via fixed path or network connections

Cluster Benefits

- Absolute scalability
- Incremental scalability
- High availability
- Superior price/performance

Distributed DBMS implementations

Data Warehousing & Mining

UNIT – II

Syllabus of Unit - II

- DATA Warehousing
- Data Warehousing Components
- Building a Data Warehouse
- Warehouse Database
- Mapping the Data Warehouse to a Multiprocessor Architecture
- DBMS Schemas for Decision Support
- Data Extraction, Cleanup & Transformation Tools
- Metadata.

Data Warehouse

- The Data warehouse is an environment, not a product.
- It is an architectural construct of an information system that provides users with current and historical decision support information that is hard to access or present in traditional operational data store.
- Data warehousing is a blend of technologies and components aimed at effective integration of operation database into an environment that enables strategic use of data.
- These technologies include relational and multi-dimensional database management system, client/ server architecture, meta-data modeling and repositories, graphical user interface etc.

Data Warehousing Components

Data Warehousing Components

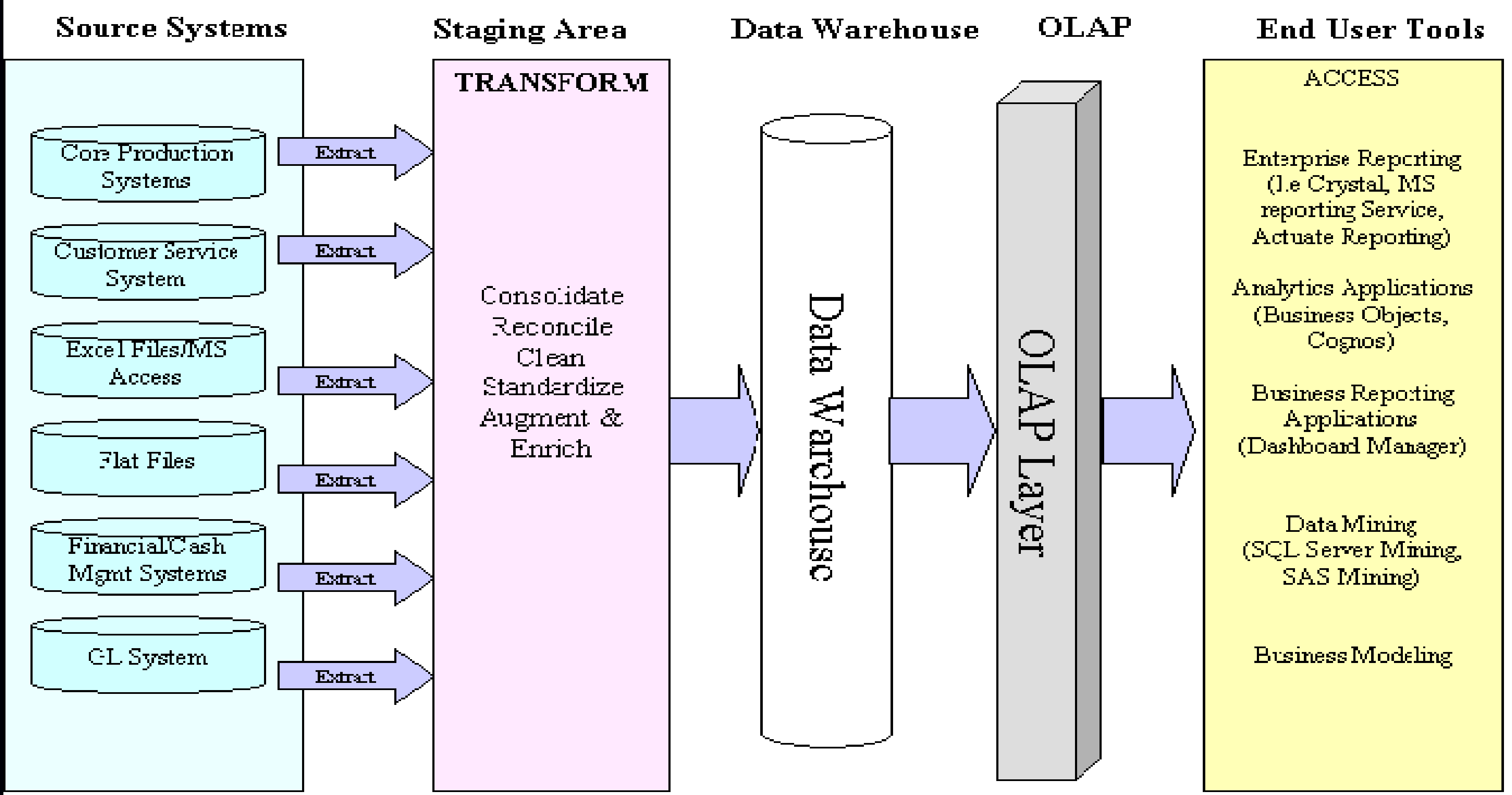
- The data warehouse architecture is based on a relational database management system server that functions as the central repository for informational data. Operational data and processing is completely separated from data warehouse processing. This central information repository is surrounded by a number of key components designed to make the entire environment functional, manageable and accessible by both the operational systems that source data into the warehouse and by end-user query and analysis tools.

Components of Data Warehouse continued…

- There are following **seven** components of a Data Warehouse:
  - **Data Warehouse Database**
  - **Sourcing, Acquisition, Cleanup and Transformation Tools**
  - **Meta Data**
  - **Access (Query) Tools**

    The **query tool** allows executives and other users real-time access to the Data Warehouse database for query generation, result displays, reports and data exports
  - **Data Marts**
  - **Data Warehouse Administration and Management**
  - **Information Delivery System**

Components & Framework



1. Data Warehouse Database

The central data warehouse database is the cornerstone of the data warehousing environment. Certain data warehouse attributes, such as very large database size, ad hoc query processing and the need for flexible user view creation including aggregates, multi-table joins and drill-downs, have become drivers for different technological approaches to the data warehouse database. These approaches include:

- Parallel relational database designs for scalability that include shared-memory, shared disk, or shared-nothing models implemented on various multiprocessor configurations (symmetric multiprocessors or SMP, massively parallel processors or MPP, and/or clusters of uni- or multiprocessors).
- An innovative approach to speed up a traditional RDBMS by using new index structures to bypass relational table scans.
- Multidimensional databases (MDDBs) that are based on proprietary database technology. Multi-dimensional databases are designed to overcome any limitations placed on the warehouse by the nature of the relational data model. MDDBs enable on-line analytical processing (OLAP) tools that architecturally belong to a group of data warehousing components jointly categorized as the data query, reporting, analysis and mining tools.

2. Sourcing, Acquisition, Cleanup and Transformation Tools

The data sourcing, cleanup, transformation and migration tools perform all of the conversions, summarizations, key changes, structural changes and condensations needed to transform disparate data into information that can be used by the decision support tool. They produce the programs and control statements, including the COBOL programs, MVS job-control language (JCL), UNIX scripts, and SQL data definition language (DDL) needed to move data into the data warehouse for multiple operational systems. These tools also maintain the meta data. The functionality includes:

- Removing unwanted data from operational databases
- Converting to common data names and definitions
- Establishing defaults for missing data
- Accommodating source data definition changes

ETL Tools

- **ETL** tools are the equivalent of **schema mappings** in virtual integration, but are more powerful
- **Some of the Well Known ETL Tools**

  The most well known commercial tools are Ab Initio, IBM InfoSphere DataStage, Informatica, Oracle Data Integrator and SAP Data Integrator.

  There are several open source ETL tools, among others:

  Apatar, CloverETL, Pentaho and Talend.
- Arbitrary pieces of code to take data from a source, convert it into data for the warehouse:
  - **Import filters** – read and convert from data sources
  - **Data Transformations** – join, aggregate, filter, convert data
  - **De-duplication** – finds multiple records referring to the same entity, merges them
  - **Profiling** – builds tables, histograms, etc. to summarize data
  - **Quality management** – test against master values, known business rules, constraints, etc.

3. Meta Data

Meta data is data about data that describes the data warehouse. It is used for building, maintaining, managing and using the data warehouse. Meta data can be classified into:

- **Technical meta data**, which contains information about warehouse data for use by warehouse designers and administrators when carrying out warehouse development and management tasks.
- **Business meta data,** which contains information that gives users an easy-to-understand perspective of the information stored in the data warehouse.

4. Access (Query) Tools

Query and Reporting tools can be divided into two groups:

- **Reporting Tools** and **Managed Query Tools**
- ***Reporting tools*** can be further divided into **production reporting** tools and **report writers**.
  - **Production reporting** tools let companies generate regular operational reports or support high-volume batch jobs such as calculating and printing paychecks.
  - **Report writers**, on the other hand, are inexpensive desktop tools designed for end-users.
- ***Managed query tools*** shield end users from the complexities of SQL and database structures by inserting a meta-layer between users and the database. These tools are designed for easy-to-use, point-and-click operations that either accept SQL or generate SQL database queries.

5. Data Mart

- the term data mart means different things to different people. A rigorous definition of this term is a data store that is subsidiary to a data warehouse of integrated data. The data mart is directed at a partition of data (often called a subject area) that is created for the use of a dedicated group of users. These could be classified in two categories:
  - Dependent Data Marts
  - Independent Data Marts

Dependent Data Marts: These types of data marts, data is sourced from the data warehouse, have a high value because no matter how they are deployed and how many different enabling technologies are used, different users are all accessing the information views derived from the single integrated version of the data.

Independent Data Marts: Unfortunately, the misleading statements about the simplicity and low cost of data marts sometimes result in organizations or vendors incorrectly positioning them as an alternative to the data warehouse. This viewpoint defines independent data marts that in fact, represent fragmented point solutions to a range of business problems in the enterprise. This type of implementation should be rarely deployed in the context of an overall technology or applications architecture. Indeed, it is missing the ingredient that is at the heart of the data warehousing concept -- that of data integration.

6. Data Warehouse Administration and Management

Managing data warehouses includes:

1. Security and priority management
2. Monitoring updates from the multiple sources
3. Data quality checks
4. Managing and updating meta data
5. Auditing and reporting data warehouse usage and status
6. Purging data
7. Replicating, sub-setting and distributing data
8. Backup and Recovery and
9. Data warehouse storage management.

7. Information Delivery System

- The information delivery component is used to enable the process of subscribing for data warehouse information and having it delivered to one or more destinations according to some user-specified scheduling algorithm.

- In other words, the information delivery system distributes warehouse-stored data and other information objects to other data warehouses and end-user products such as spreadsheets and local databases.

- Delivery of information may be based on time of day or on the completion of an external event.

- The rationale for the delivery systems component is based on the fact that once the data warehouse is installed and operational, its users don't have to be aware of its location and maintenance.

Building a Data Warehouse

Why a Data Warehouse Application – Business Perspectives

There are several reasons why organizations consider Data Warehousing a critical need. From a business prospective, to strive and succeed in today's highly competetive global environment, business users demand business answers mainly because:

- Decisions need to be made quickly and correctly, using all available data
- Users are business domain experts, not computer professionals
- The amount of data increasing in the data stores, which affects response time and the sheer ability to comprehend its content.
- Competitions is heating up in the areas of business intelligence and added information value.

Building a Data Warehouse

Why a Data Warehouse Application – Technology Perspectives

- There are several technology reasons also for existence of Data Warehousing.
  - First, the Data Warehouse is designed to address the incompatibility of informational and operational transactional systems. These two classes of information systems are designed to satisfy different , often incompatible, requirements.
  - Secondly, the IT infrastructure is changing rapidly, and its capabilities are increasing, as evidenced by the following:
    - The prices of MIPS continues to decline, while the power of processors doubles every 2 years
    - The prices of digital storage is rapidly dropping
    - Network bandwidth is increasing, while the price of high bandwidth is decreasing
    - The workplace is increasingly heterogeneous with respect to both the hardware and software
    - Legacy systems need to, and can, be integrated with new applications

Building a Data Warehouse

1. **Business Considerations (Return on Investment)**
2. **Design Considerations**
3. **Technical Considerations**
4. **Implementation Considerations**
5. **Integrated Solutions**
6. **Benefits of Data Warehousing**

Building a Data Warehouse Contd..

1. Business Considerations (Return on Investment)

1. Approach

- The **Top-down Approach**, meaning that an organization has developed an enterprise data model, collected enterprise-wide business requirements, and decided to build an enterprise data warehouse with subset data marts.
- The **Bottom-up Approach**, implying that the business priorities resulted in developing individual data marts, which are then integrated into enterprise data warehouse.

1. Organizational Issues

A Data Warehouse, in general, is not truly a technological issue, rather, it should be more concerned with identifying and establishing information requirements, the data sources to fulfill these requirements, and timeliness.

Building a Data Warehouse Contd..

2. Design Consideration

To be a successful, a data warehouse designer must take a holistic approach – consider all data warehouse components as parts of a single complex system and take into the account all possible data stores and all known usage requirements. Failing to do so may easily result in a data warehouse design that is skewed toward a particular business requirement, a particular data sources, or a selected access tool. This is also one of the reasons why a data warehouse is rather difficult to build. The main factors include:

- **Heterogeneity of Data sources, which affects data conversion, quality, timeliness**
- **Use of historical data, while implies that data may be "old".**
- **Tendency of databases to grow very large**

Building a Data Warehouse Contd..

2. **Design Consideration -** In addition to the general considerations, there are several specific points relevant to the data warehouse design:

- Data Content
- Metadata
- Data Distribution

  One of the biggest challenge when designing a data warehouse is the data placement and distribution strategy.
- Tools

  These tools provide facilities for defining the transformation and cleanup rules, data movement (from operational sources to the warehouses, end-user query, reporting, and data analysis.
- Performance consideration

Building a Data Warehouse Contd..

3. Technical Considerations

A number of technical issues are to be considered when designing and implementing a Data Warehouse environment.

1. The Hardware Platform that would house the Data Warehouse for parallel query scalability. (Uni-Processor, Multi-processor, etc)

2. The DBMS that supports the warehouse database

3. The communication infrastructure that connects the warehouse, data marts, operational systems, and end users

4. The hardware platform and software to support the metadata repository

5. The systems management framework that enables centralized management and administration to the entire environment.

Building a Data Warehouse Contd..

4. Implementation Considerations

i. Access Tools

Currently no single tool in the market can handle all possible data warehouse access needs. Therefore, most implementations rely on a suite of tools. Examples of Access types include:

a. Simple Tabular for reporting

b. Ranking

c. Multi-variable Analysis

d. Time Series Analysis

e. Data Visualization, Graphing, Charting and pivoting

f. Complex Textual Search

g. Statistical Analysis

h. AI Techniques for testing of hypothesis, trends discovery, definition, validation of Data Clusters and segments

i. Information Mapping (i.e. mapping of Spatial Data in geographic information systems)

j. Ad-hoc User Specified Queries

k. Pre-defined repeatable queries

l. Interactive drill-down reporting and analysis

m. Complex queries with multiple joins, multi-level subquesries, and sophisticated search criteria.

Building a Data Warehouse Contd..

4. Implementation Considerations

ii. Data Extraction, Cleanup, Transformation, and Migration

As a components of the Data Warehouse architecture, proper attention must be given to Data Extraction, which represents a critical success factor for a data warehouse architecture.

1. The ability to identify data in the data source environments that can be read by conversion tool is important. This additional step may affect the timeliness of data delivery to the warehouse.

2. Support for the flat files. (VSAM, IMS, IDMS) is critical, since bulk of the corporate data is still maintained in this type of data storage.

3. The capability to merge data from multiple data stores is required in many installations.

4. The specification interface to indicate the data to extracted and the conversion criteria is important.

5. The ability to read information from data dictionaries or import information from repository product is desired.

6. The ability to perform data-type and character-set translation is a requirement when moving data moving between incompatible systems.

7. The capability to create summarization, aggregation, and derivation records and fields is very important.

Building a Data Warehouse Contd..

4. Implementation Considerations

iii. Data Placement Strategies

As Data Warehouse grows, there are at least two options for Data Placement. One is to put some of the data in the data warehouse into another storage media (WORM, RAID). Second option is to distribute data in data warehouse across multiple servers. Some criteria must be established for dividing it over the servers – by geography, organization unit, time, function, etc. However, the data is divided, a single source of meta data across the entire organization is required. Hence this configuration requires both corporation-wide and the meta data managed for any given server.

Building a Data Warehouse Contd..

4. Implementation Considerations

iv. Metadata

A frequently occurring problem in Data Warehouse is the problem of communicating to the end user what information resides in the data warehouse and how it can be accessed. The key to providing users and applications with a roadmap to the information stored in the warehouse is the **metadata**. It can define all data elements and their attributes, data sources and timing, and the rules that govern data use and data transformations. Meta data needs to be collected as the warehouse is designed and built.

Building a Data Warehouse Contd..

4. Implementation Considerations

v. User Sophistication Levels

Data Warehousing is relatively new phenomenon, and a certain degree of sophistication is required on the end user's part to effectively use the warehouse. The users can be classified on the basis of their skill level in accessing the warehouse:

1. Casual Users: These users are most comfortable retrieving information from the warehouse in pre-defined formats, and running preexisting queries and reports.

2. Power Users: In their delay activities, these users typically combine predefined queries with some relatively simple and ad-hoc queries that they create themselves. These users need access tools that combine the simplicity of pre-defined queries and reports with a certain degree of flexibility.

3. Experts: These users tend to create their own queries and perform sophisticated analysis on the information they retrieve from the warehouse. These users know the data, tools and database well enough to demand tools that allow for maximum flexibility and adaptability.

Benefits of Data Warehouse

Successfully implemented data warehousing can realize some significance benefits which can be categorized in two categories:

1. Tangible Benefits:

1. Product inventory turnover is improved
2. Costs of product introduction are decreased with improved selection of target markets.
3. More cost effective decision making is enabled by separating (ad-hoc) query processing from running against operational database.
4. Better business intelligence is enabled by increased quality and flexibility of market analysis available through multi-level data structures, which may range from detailed to highly summarized.

2. Intangible Benefits:

1. Improved productivity
2. Reduced redundant processing, support, and software to support overlapping decision support applications
3. Enhanced Customer relations through improved knowledge of individual requirements and trends, through customization, improved communications, and tailored product offerings.
4. Enabling business process reengineering – data warehousing can provide useful insights into work process themselves,

Warehouse Database

- The organizations that embarked on data warehousing development deal with ever increasing amounts of data. Generally speaking, the size of a data warehouse rapidly approaches the point where the search for better performance and scalability becomes a real necessity. This search aims to pursue two goals:
  - ***Speed-up:*** the ability to execute the same request on the same amount data in less time
  - ***Scale-up:*** the ability to obtain the same performance on the same request as the database size increases.

  An additional and important goal is to achieve ***linear*** speed-up and scale-up, doubling the number of processors cuts the response time in half (linear speed-up) or provides the same performance on twice as much data (linear scale-up).

Mapping the Data Warehouse to a Multiprocessor Architecture

- The goals of linear performance and scalability (discussed in previous slide) can be satisfied by parallel hardware architectures, parallel operating systems, and parallel DBMSs. Parallel hardware architectures are based on Multi-processor systems designed as a Shared-memory model (symmetric multiprocessors), Shared-disk model or distributed-memory model (MPP and Clusters of SMPs). Parallelism can be achieved in two different ways:
  - Horizontal Parallelism (Database is partitioned across different disks)
  - Vertical Parallelism (occurs among different tasks – all components query operations i.e. scans, join, sort)
  - Data Partitioning

Database Architectures for Parallel Processing

- Shared-memory Architecture
- Shared Disk Architecture
- Shared-nothing Architecture
- Combined Architecture

Parallel RDBMS Features

- Data Warehouse development requires a good understanding of all architectural components, including the data warehouse DBMS Platform. Understanding the basic architecture of Warehouse database is the first step in evaluating and selecting a product.
- State of the art parallel features the developers and users of the Warehouse should demand from the DBMS vendor:
    - Scope and techniques of Parallel DBMS
        - Queries (Insert/ Update/Delete)
        - DBMS that supports parallel database load, backup, reorganization and recovery is much better positioned for VLDBs.
    - Optimizer Implementation
    - Application Transparency
    - The Parallel environment
    - DBMS Management Tools
    - Price/ Performance

Parallel DBMS Vendors

- **ORACLE** – Oracle supports Parallel Database processing with its add-on **Oracle Parallel Server Option (OPS)** and **Parallel Query Option (PQO)** with Query Coordinator.
- **Informix** – Informix developed its **Dynamic Scalable Architecture (DSA) to support Shared-Memory, Shared-Disk, and Shared-Nothing Models**. Informix OnLine release 8, also known as XPS (eXtended Parallel Server), supports MPP Hardware platforms that include IBM, SP, AT & T, Sun, HP, ICL Goldrush, with sequent, Siemens, Pyramid etc.
- **IBM** – DB2 Parallel Edition **(DB2 PE)**, a Database based on DB2/6000 Server Architecture; latest version is **DB2 Universal Database**.
- **Sybase** – Sybase implemented its parallel DBMS functionality in a product called **SYBASE MPP** (formerly Navigational Server). It was jointly developed by Sybase and NCR (formerly AT&T GIS), and its first release was targeted for the AT&T 3400, 3500 (both SMP) and 3600 (MPP) Platforms.
- Other RDBMS Products **i.** NCR Teradata **ii.** Tandem NonStop SQL/MP
- Specialized Database Products - **i.** Red Brick Systems

  ii. White Cross Systems Inc.

DBMS Schemas for Decision Support

- Data Warehousing projects were forced to choose between a data model and a corresponding database schema that is intuitive for analysis but performs poorly and a model-schema that performs better but is not well suited for analysis.
- As Data Warehousing continued to mature, new approaches to schema design resulted in schemas better suited to business analysis that is so crucial to successful data warehousing.
- The schema methodology that is gaining widespread acceptance for Data Warehousing is the **Star Schema**.

Data Layout for best Access

- The original objective in developing an abstract model known as Relational Model were to address a number of shortcomings of non-relational DBMS and application development.
- The typical requirements for the RDBMS supporting operational systems are based on the need to effectively support a large number of small but simultaneous read and write requests.
- The demand placed on the RDBMS by a Data Warehouse are very different. A data warehouse RDBMS typically needs to process queries that are large, complex, ad-hoc and data intensive.
- Solving modern business problems such as market analysis and financial forecasting requires query-centric database schemas that are array-oriented and multi-dimensional in nature.

Multi-dimensional Data Model

- The Multi-dimensional nature of business questions is reflected in the fact that, for example, marketing managers are no longer satisfied by asking simple one-dimensional questions such as “How much revenue did the new product generate by month, in northeastern division, broken down by user demographic, by sales office, relative to the previous version of the product, compared with the plan?” – a six dimensional question.

STAR SCHEMA

- The Multi-dimensional view of Data that is expressed using relational database semantics is provided by the database schema design called Star Schema.
- The basic premise of Star Schema is that information can be classified into two groups: **facts** and **dimensions**.
- **Facts** are the core Data element being analyzed. For example, units of individual items sold are facts.
- **Dimensions** are attributes about the facts. For example, dimensions are the product types purchased and date of purchase.

Data Extraction, Cleanup & Transformation Tools

- The task of capturing data from a source data system, cleaning and transforming it and then loading the results into a target data system can be carried out either by separate products, or by a single integrated solution. More contemporary integrated solutions can fall into one of the categories described below:
  - Code Generators
  - Database data Replications
  - Rule-driven Dynamic Transformation Engines (Data Mart Builders)

Code Generator

- It creates 3GL/4GL transformation programs based on source and target data definitions, and data transformation and enhancement rules defined by the developer.
- This approach reduces the need for an organization to write its own data capture, transformation, and load programs. These products employ DML Statements to capture a set of the data from source system.
- These are used for data conversion projects, and for building an enterprise-wide data warehouse, when there is a significant amount of data transformation to be done involving a variety of different flat files, non-relational, and relational data sources.

Database Data Replication Tools

- These tools employ database triggers or a recovery log to capture changes to a single data source on one system and apply the changes to a copy of the data source data located on a different system.
- Most replication products do not support the capture of changes to non-relational files and databases, and often do not provide facilities for significant data transformation and enhancement.
- These point-to-point tools are used for disaster recovery and to build an operational data store, a data warehouse, or a data mart when the number of data sources involved are small and a limited amount of data transformation and enhancement is required.

Rule-driven Dynamic Transformation Engines

- They are also known as Data Mart Builders and capture data from a source system at User-defined intervals, transform data, and then send and load the results into a target environment, typically a data mart.
- To date most of the products of this category support only relational data sources, though now this trend have started changing.
- Data to be captured from source system is usually defined using query language statements, and data transformation and enhancement is done on a script or a function logic defined to the tool.
- With most tools in this category, data flows from source systems to target systems through one or more servers, which perform the data transformation and enhancement. These transformation servers can usually be controlled from a single location, making the job of such environment much easier.

Data Warehousing & Mining

UNIT – III

Syllabus of Unit - III

- Business Analysis
- Reporting & Query Tools & Applications
- On line Analytical Processing(OLAP)
- Patterns & Models
- Statistics
- Artificial Intelligence.

Business Analysis

- The principle purpose of Data Warehousing is to provide information to business users for strategic decision making.
- This decision making process is business analysis of the information stored in a data warehouse, and it is enabled by a number of applications, tools, and techniques that can provide various business-focused views to business domain experts.

What is Business Analysis?

- **Business analysis** is the discipline of identifying business needs and determining solutions to business problems. Solutions often include a systems development component, but may also consist of process improvement, organizational change or strategic planning and policy development.
- **The person who carries out this task is called a business analyst or BA**. Those BAs who work solely on developing software systems may be called IT Business Analysts, Technical Business Analysts, Online Business Analysts or Systems Analysts.

- Business analysis as a discipline has a heavy overlap with requirements analysis sometimes also called requirements engineering, but focuses on identifying the changes to an organization that are required for it to achieve strategic goals. These changes include changes to strategies, structures, policies, processes, and information systems. Examples of business analysis includes:
- **Enterprise analysis or company analysis**

  Focuses on understanding the needs of the business as a whole, its strategic direction, and identifying initiatives that will allow a business to meet those strategic goals. It also includes:
- Creating and maintaining the business architecture
- Conducting feasibility studies
- Identifying new business opportunities
- Scoping and defining new business opportunities
- Preparing the business case
- Conducting the initial risk assessment

Contd…

- **Requirements planning and management**

  Involves planning the requirements development process, determining which requirements are the highest priority for implementation, and managing change, Requirements elicitation Describes techniques for collecting requirements from stakeholders in a project. Techniques for requirements elicitation include:
- Brainstorming
- Document analysis
- Focus group
- Interface analysis
- Interviews
- Workshops
- Reverse engineering
- Surveys
- User task analysis
- Process Mapping

Contd….

- **Requirements analysis and documentation**

  Describes how to develop and specify requirements in enough detail to allow them to be successfully implemented by a project team. The major forms of analysis are:
- Architecture analysis
- Business process analysis
- Object-oriented analysis
- Structured analysis
- Requirements documentation can take several forms:
  - Textual
  - Matrix
  - Diagrams
  - Models
- **Requirements communication**

  Describes techniques for ensuring that stakeholders have a shared understanding of the requirements and how they will be implemented. Solution assessment and validation Describes how the business analyst can verify the correctness of a proposed solution, how to support the implementation of a solution, and how to assess possible shortcomings in the implementation.

Business Analysis Techniques

- There are a number of generic business techniques that a Business Analyst will use when facilitating business change. Some of these techniques include:
- **PESTLE**

  This is used to perform an external environmental analysis by examining the many different external factors affecting an organization. The six attributes of **PESTLE**:

  - **P**olitical (Current and potential influences from political pressures) **E**conomic (The local, national and world economy impact)
  - **S**ociological (The ways in which a society can affect an organization) **T**echnological (The effect of new and emerging technology)
  - **L**egal (The effect of national and world legislation) **E**nvironmental (The local, national and world environmental issues)

- **HEPTALYSIS**

  This is used to perform an in-depth analysis of early stage businesses/ventures on seven important categories:

  Market Opportunity Product/Solution Execution Plan Financial Engine

  Human Capital Potential Return Margin of Safety

- **MOST**

  This is used to perform an internal environmental analysis by defining the attributes of MOST to ensure that the project you are working on is aligned to each of the 4 attributes. The four attributes of MOST are:

  Mission (where the business intends to go) **O**bjectives (the key goals which will help achieve the mission)
  Strategies (options for moving forward) **T**actics (how strategies are put into action)

- **SWOT**

This is used to help focus activities into areas of strength and where the greatest opportunities lie. This is used to identify the dangers that take the form of weaknesses and both internal and external threats. The four attributes of SWOT:

Strengths - What are the advantages? What is currently done well? (e.g. key area of best-performing activities of your company)

Weaknesses - What could be improved? What is done badly? (e.g. key area where you are performing poorly)

Opportunities - What good opportunities face the organization? (e.g. key area where your competitors are performing poorly)

Threats - What obstacles does the organization face? (e.g. key area where your competitor will perform well)

CATWOE

This is used to prompt thinking about what the business is trying to achieve. Business perspectives help the business analyst to consider the impact of any proposed solution on the people involved. There are six elements of CATWOE:

Customers - Who are the beneficiaries of the highest level business process and how does the issue affect them?

Actors - Who is involved in the situation, who will be involved in implementing solutions and what will impact their success?

Transformation Process - What processes or systems are affected by the issue?

World View - What is the big picture and what are the wider impacts of the issue?

Owner - Who owns the process or situation being investigated and what role will they play in the solution?

Environmental Constraints - What are the constraints and limitations that will impact the solution and its success?

Role of Business Analyst

- Strategist
- Architect
- Systems analyst

Goal of Business Analysis

Ultimately, business analysis want to achieve the following outcomes:

- Reduce waste
- Create solutions
- Complete projects on time
- Improve efficiency
- Document the right requirements

Reporting & Query Tools & Applications

- The principle purpose of Data Warehousing is to provide information to business users for strategic decision making. These users interact with the data warehouse using front-end tools, or by getting the required information through the information delivery systems.
- Different types of users engage in different types of decision support activities, and therefore require different types of tools.

S.No.	User Type	Activity	Tools
1	Clerk	Simple Retrieval	4GL
2	Executive	Exception Report	EIS
3	Manager	Simple Retrieval	4GL
4	Business Analyst	Complex Analysis	Spreadsheet, OLAP, Data Mining

Contd…..

- There are five categories of decision support tools, although the lines that separate them are quickly blurring:
  - Reporting
  - Managed Queries
  - Executive Information Systems
  - OLAP
  - Data Mining

Reporting Tools

- Reporting Tools can be divided into two categories:
  - **Production Reporting Tools**: These tools let companies generate regular operational reports or support high-volume batch jobs, such as calculating and printing paychecks. Production Reporting Tools include 3GLs such as COBOL, specialized 4GL, such as Information Builders, Inc’s Focus and high-end client/ server tools such as MITTI’s SQR.
  - **Desktop Report Writers**: Report writers are inexpensive desktop tools designed for end users. Product such as Crystal Reports, let users design and run reports without having to rely on the IS Department.
  - In general Report Writers have GUI and Built-in Charting functions. They can Pull Groups of data from a variety of Data sources and integrate them in a single report. Leading Report Writers include Crystal Reports, Acutate Reporting System, IQ Objects, InfoReports. Reports Writers also are beginning to offer Object-Oriented Interfaces for designing and manipulating reports and modules for performing ad-hoc queries and OLAP Analysis.

Managed Query Tools

- Managed Query Tools shield users from the complexities of SQL, and Database Structures by inserting a ***Meta-layer*** between users and the Database.
- **Meta-layer** is the software that provides subject-oriented views of a Database and support-point-and-click creation of SQL.
- Different vendors uses different nomenclature for this Meta-layer like – Universe, Catalog.
- Managed Query Tools have been extremely popular because they make it possible for knowledge workers to access corporate data without IS intervention.
- Most Manages Query Tools have embraced Three-tiered architectures to improve scalability.
- Managed Query Tools are racing to embed support for OLAP and Data Mining features.
- Leading Managed Query Tools are IQ Objects, GQL (by Andyne Computing), Decision Servers (by IBM), ESPERANT (by Speedware), Discoverer/ 2000 (by Oracle Corp.), Information Builder etc.

Executive Information System

- Executive Information System (EIS) Report Writers and Managed Query Tools, they were first deployed on Mainframes.
- EIS tools allow developers to build customized, graphical decision support applications that give managers and executives a high level view of the business and access to external sources such as custom, on-line news feeds.
- EIS Applications highlight exceptions to normal business activity or rules by using color-coded graphics.
- Popular EIS tools include Pilot Software, Lightship, Forest & Trees, Comshare, Commander Decision, Oracle Express Analyzer, SAS/EIS.

OLAP Tools

- OLAP tools provide an intuitive way to view corporate data.
- These tools aggregate data along common business subjects or dimensions and then let users navigate through the hierarchies and dimensions with the click of a mouse button.
- Users can drill down, across, or up levels in each dimension or pivot and swap out dimension to change their view of the data.
- Some tools, such as Essbase and Oracle's Express pre-aggregate data in special multidimensional databases. Other tools work directly against relational data and aggregate data on-fly, such as MicroStrategy, DSS Agent (by Inc.) or Information Advantage, DecisionSuite.
- Desktop OLAP tools include PoerPlay, BrioQuery, Planning Sciences, Gentium, Pablo.

Data Mining Tools

- Data Mining tools are becoming hot commodities because they provide insights into corporate data that aren't easily discerned with managed query or OLAP tools.
- Data Mining tools use a variety of statistical and artificial intelligences (AI) algorithms to analyze the correlation of variables in the data and ferret out interesting patterns and relationships to investigate.
- Some Data Mining tools such as IBM's Intelligent Miner, are expensive and require statisticians to implement and manage. But there is a new breed of tools emerging that promises to take the mystery out of Data Mining. These tools include DataMind, Discovery Server etc.

Need for Applications

- In a Data warehouse environment, users expect easy-to-read reports while others concentrate on the on-screen presentation. These tools are preferred choice of the users of Business applications such as Segment Identification, Demographic Analysis, Territory Management and Customer Mailing Lists.
- As the complexity of the questions grows, these tools may rapidly become inefficient. Thus we need to understand the changing requirements and make the provisions of the same in the applications timely. This requires understanding of business needs which may be any of the following or even others:
  - Simple tabular form reporting
  - Ad-hoc User-specified Queries
  - Predefined repeatable queries
  - Complex queries with multiple joins, multi-level sub-queries, and sophisticated search criteria
  - Ranking
  - Multi-variable Analysis
  - Time Series Analysis
  - Data Visualization, Graphing, Charting and Pivoting
  - Complex Textual Search
  - Statistical Analysis

Need for Applications

- Statistical Analysis
- AI Techniques for Testing of Hypothesis, trend discovery, Definition, and validation of Data Clusters and Segments
- Information Mapping (i.e., mapping of Spatial Data in Geographic Information Systems)
- Interactive Drill-down Reporting and Analysis

Popular applications are:

- Cognos Impromptu
- Power Builder
- Forte – It provides application developers with facilities to develop and partition application to be efficiently placed on the proper platforms of the Three-tiered architecture.
- Cactus and FOCUS Fusion (by Information Builders)

On line Analytical Processing(OLAP)

On-Line Analytical Processing (OLAP) is a category of software technology that 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 functionality is characterized by dynamic Multi-dimensional analysis of consolidated enterprise data supporting end user analytical and navigational activities including:

1.Calculations and modeling applied across dimensions, through hierarchies and/or across members
2.Trend analysis over sequential time periods
3.Slicing subsets for on-screen viewing
4.Drill-down to deeper levels of consolidation
5.Reach-through to underlying detail data
6.Rotation to new dimensional comparisons in the viewing area

OLAP is implemented in a multi-user client/server mode and offers consistently rapid response to queries, regardless of database size and complexity. OLAP helps the user synthesize enterprise information through comparative, personalized viewing, as well as through analysis of historical and projected data in various "what-if" data model scenarios. This is achieved through use of an OLAP Server.

The major OLAP vendor are Hyperion, Cognos, Business Objects, MicroStrategy. The setting up of the environment to perform OLAP analysis would also require substantial investments in time and monetary resources.

OLAP Guidelines

Multidimensionality is at the core of a number of OLAP systems available today. However, the availability of these systems does not eliminate the need to define a methodology of how to select and use the product. Dr. E.F. Ted Codd, underlined some of the Guidelines for the OLAP Applications which now have become a de-facto standards. These are:

- Multidimensional Conceptual View
- Transparency
- Accessibility
- Consistent Reporting Performance
- Client/ Server Architecture
- Generic Dimensionality
- Dynamic Sparse Matrix Handling
- Multiuser Support
- Unrestricted Cross-dimensional Operations
- Intuitive Data Manipulation
- Flexible Reporting
- Unlimited Dimensions and Support

On line Analytical Processing(OLAP)

OLAPs have a different mandate from OLTPs. OLAPs are designed to give an overview analysis of what happened. Hence the data storage (i.e. data modeling) has to be set up differently. The most common method is called the **Star design**.

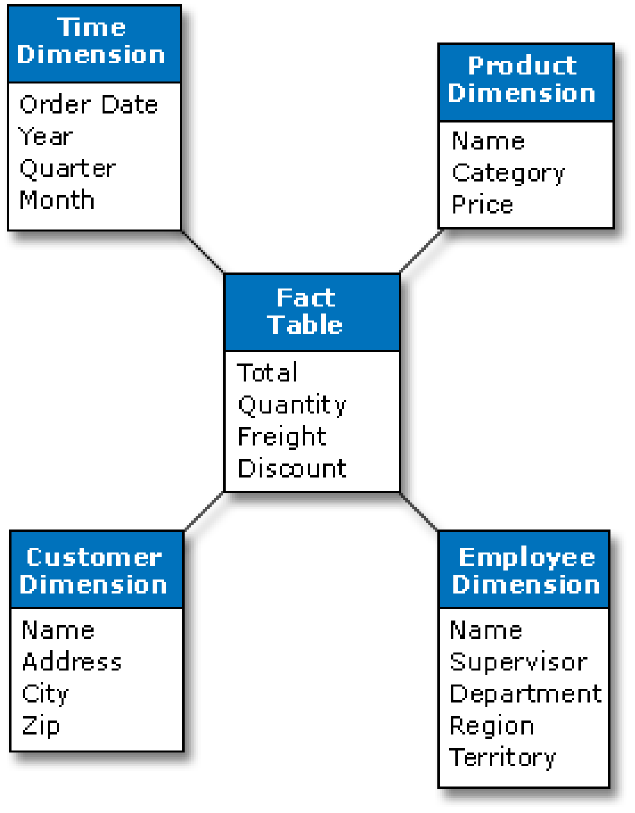
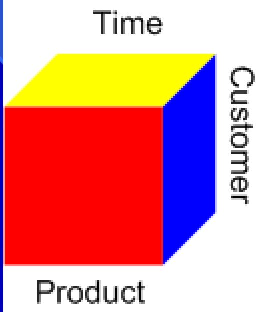


Figure 1. Star Data Model for OLAP

Figure 2. OLAP Cube with Time, Customer and Product Dimensions



To obtain answers, such as the ones above, from a data model OLAP *cubes* are created. OLAP cubes are not strictly cuboids - it is the name given to the process of linking data from the different dimensions. The cubes can be developed along business units such as sales or marketing. Or a giant cube can be formed with all the dimensions.

The central table in an OLAP start data model is called the **fact table**. The surrounding tables are called the **dimensions**. Using the above data model, it is possible to build reports that answer questions on multidimensional requirements.

OLAP can be a valuable and rewarding business tool. Aside from producing reports, OLAP analysis can aid an organization evaluate balanced scorecard targets.

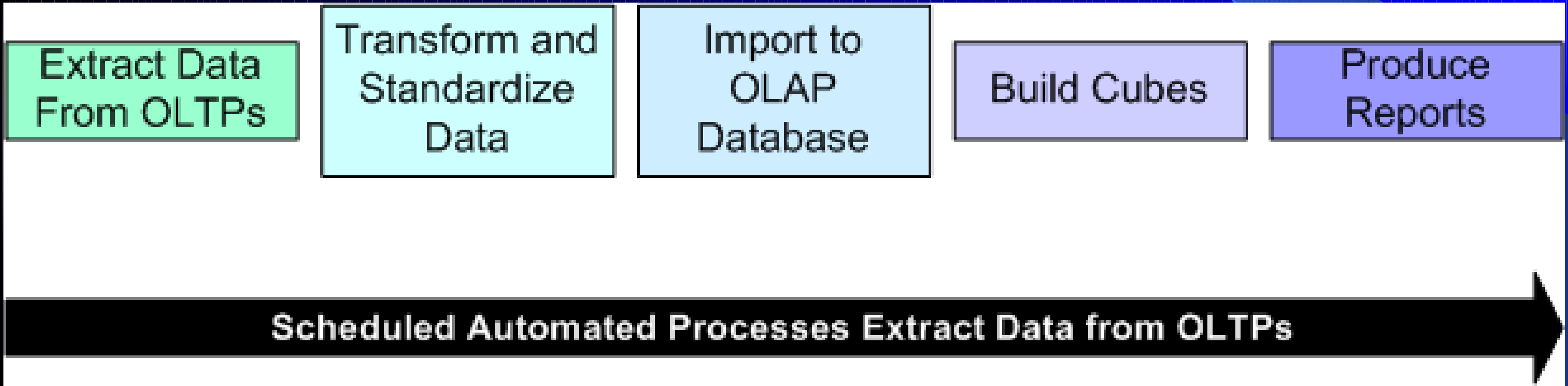

Extract Data From OLTPs
Transform and Standardize Data
Import to OLAP Database
Build Cubes
Produce Reports
Scheduled Automated Processes Extract Data from OLTPs

Figure 3. Steps in the OLAP Creation Process

Need of OLAP Application

- OLAP is an application architecture, not intrinsically a Data Warehouse, OLAP is becoming an architecture that an increasing number of enterprises are implementing to support analytical applications.
- Solving modern business problems such as market analysis and financial forecasting requires query-centric database schemas that are array-oriented and Multi-dimensional in nature.
- These business problems are characterized by the need to retrieve large numbers of records from very large data sets (100s of GBs and even TBs) and summarize them on the fly.
- The multi-dimensional nature of the problems it is designed to address is the key driver for OLAP.

OLAP Contd…

- OLAP tools are based on the concepts of Multi-dimensional databases and allow a sophisticated user to analyze the data using elaborate, multi-dimensioanl, complex views.
- Typical business applications for these tools include product performances and profitability, effectiveness of a sales programme or a marketing campaign, sales forecasting and capacity planning.
- These tools assume that the data is organized in a multidimensional model which is supported by a special multidimensional database (MDDB) or by a Relational Database designed to enable multidimensional properties (e.g. Star Schema).
- Examples of OLAP tools include Axsys, DSS Agent/ DSS Server, Beacon, Metacube, HighGate Project, PowerPlay, Pablo, CrossTargetMedia , FOCUS Fusion, Pilot Decision Support Suite etc.

Patterns & Models

- **Pattern:** An event or combination of events in a database that occurs more often than expected. Typically this means that its actual occurrences is significantly different from what would be expected by random chance.

- **Model:** A Description of original historical database from which it was built that can be successfully applied to new data in order to make predictions about missing values or to make statements about expected values.

- ***Patterns are usually driven from the data and generally reflect the data itself, whereas a model generally reflects a purpose and may not be driven from the data necessarily.***

Basics

- **Database:** The collection of Data that has been collected, on which data analysis will be performed and from which predictive models and exploratory models will be created. This is often called the historical database.
- In machine learning and Data Mining, there is often differentiation between the **Training databases and the Test databases.**
- **Record:** Each record is made up of values for each field that it contains, including the predictor fields and prediction fields.
- **Fields:** Fields correspond to the columns in a relational database and to dimensions.
- **Predictor:** A field that could be used to build a predictive model.
- **Prediction:** The field that will have a value created for it by the predictive model.
- **Value:** Each field has a value **.**

Applications of Models

- Selection
- Acquisition
- Retention
- Extension (Cross Selling)

Statistics

- Statistics is the science of learning from data.
- It includes everything from **planning for the collection of data and subsequent data management to end-of-the-line activities such as drawing inferences from numerical facts called data and presentation of results**.
- Statistics is concerned with one of the most basic of human needs: the need to find out more about the world and how it operates in face of ***variation and uncertainty***. Because of the increasing use of statistics, it has become very important to understand and practice statistical thinking.
- Or, in the words of H. G. Wells: *"Statistical thinking will one day be as necessary for efficient citizenship as the ability to read and write"*.

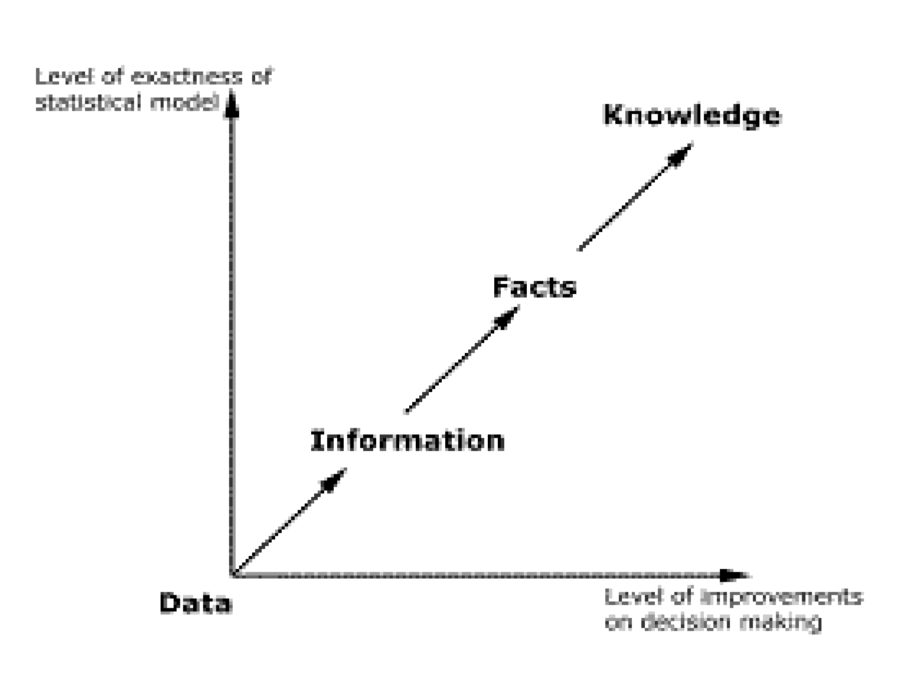
Why Statistics Needed

- **Knowledge** is what we know. **Information** is the communication of knowledge. ***Data*** are known to be crude information and not knowledge by themselves.

- The sequence from data to knowledge is as follows:
  - from data to information (data become information when they become relevant to the decision problem);
  - from information to facts (information becomes facts when the data can support it); and finally,
  - from facts to knowledge (facts become knowledge when they are used in the successful completion of the decision process).

Why Statistics Needed

- Following figure illustrates the statistical thinking process which is **based on data in constructing statistical models for decision making under uncertainties**. That is why we need statistics. Statistics arose from the need to place knowledge on a systematic evidence base. This required a study of the laws of probability, the development of measures of data properties and relationships, and so on.


Level of exactness of statistical model
Knowledge
Facts
Information
Data
Level of improvements on decision making

Significance of Statistics

- Today, businesses deal with the data ranging up to in the order of Terabytes and have to make sense of it and glean the important patterns from it.
- Some of the most frequently used summary statistics include **Max** (maximum value for a given Predictor), **Min** (minimum value for a given Predictor), **Mean** (average value for a given Predictor), **Median** (the value for a given Predictor that divides the databases as nearly as possible into two databases of equal numbers of records), **Mode** (the most common value for the Predictor), **Variance** (the measure of how spread out the values are from the average value)
- Statistics in this process can help greatly by helping us answer several important questions about the data available:
  - What patterns are there in the Database?
  - What is the chance that an event will occur?
  - Which patterns are significant?
  - What is a high-level summary of the data that gives some idea of what is contained in the database?

Some Statistical Concepts

- **Probability:** The notion of probability is a critical concept for statistics and for all data mining techniques.
- **Bayes' Theorem**: It states that if we want to know the probability of event A conditional on event B occurring, it can be calculated as the probability of both events A and B occurring divided by the probability of event B.
- **Independence**: In statistics two events are considered to be independent of each other if the probability of both of them occurring together is equal to the probability of one event multiplied by the probability of other event.
- **Hypothesis Testing:** Hypothesis Testing is a three step process that can be repeated many times until a suitable hypothesis is found:
    - The Data is observed and an understanding is formed how the data was collected and created
    - A Guess about what process created the data is made (that hopefully explains the data). This is called Hypothesis.
    - The Hypothesis is tested against the actual data by assuming that it is correct and then determining how likely it would be observe this particular set of data.

Contd….

- **Contingency Tables:** Contingency Tables are the tables that are used to show the relationship between two categorical predictors or between a Predictor and a Prediction.

- **Chi Square Test:** The Chi Square test is often used to test to see if there is a relationship between two columns of data in a database- may be between a Predictor Column and a Prediction Column or between two predictors.

- **Predictors:** Column(s) based on which predictions are to be made is called a Predictor.

Artificial Intelligence (AI)

- AI is a filed of science and engineering concerned with the computational understanding of what is commonly called intelligent behavior, and with the criterion of artifacts that exhibit such behavior.
- **Expert Systems** are a class of techniques, algorithms and computer programs within the field of Artificial Intelligence which seek to provide expert levels of functionality within well defined domains. These are generally **Rule-based (IF-THEN)**. Applications of Expert systems have wide range from medical diagnosis to large computer configurations. These rule based systems could of two types: Forward Chained and Backward Chained Systems. Popular example are is Xcon (by DEC), Mycin (by Stanford University).
- **Limitations:**
  - **The System is only as smart as a human expert – Since system is not learning from Data directly, rather from knowledge extracted from human experts, any biases or errors in reasoning inherent in the expert's view will be reflected in the system.**
  - **The Systems are very complex**
  - **The Systems are human intensive – The majority of time spent in building these systems is in trying to extract the knowledge from the human experts.**

FUZZY LOGIC

- **FUZZY Logic** is a technique designed to correct the shortcomings of the rule based Expert Systems. The basic idea of Fuzzy Logic is that there is no precise cut off between Sets and Categories and that these boundaries are ***"Fuzzy".***
- Using Fuzzy Logic in a system involves several steps:
  - Step 1: Input Data
  - Step 2: Combining Evidence
  - Step 3: Defuzzification
- One problem with rule-based systems is that they can be somewhat brittle (breakable) in the sense that they break easily when they are bent toward a slightly different problem. For example, there may be a very powerful rule that states " ***If income is high and debt is high, then the loan applicant is a bad risk"*** . The rule itself begs the question of what is "High"? The term high is in the rule rather than a particular number. The high may mean differently to different people, and it may be interpreted to be a very specific cut off value (One income e.g. 100000/- is considered to high but another that is only slightly less is no longer considered to be "High", say 99,999/-). Because there can be such a sharp cut off, some valuable information is lost.

Case - 1

- Consider the problem, for instance, in which “High Income” is defined anything over 10,00,000/- per year and high debt is defined to be when a consumer pays 45% of his gross earning in interest or payout on debt.
- The rule can be interpreted as “If people are wealthy but in a lot of debt, then they should not be given the loan”.
- The fact that words such as “high” have been used to make it easy to understand and interpret the rule.
- The problem is that these words provide continuous information into rigid categories – and mistakes can be made.
- Consider an applicant X whose debt is 55% of his gross annual income, which is 9,99,999/-. In the classic expert system the rule mentioned above would not fire to deny the loan since X’s income falls just barely below the cut off for the definition of what is considered to be high income.
- This is a problem because his debt is exceedingly high. Thus the rule that should have captured X as a bad risk misses. This is an example of brittleness of Classically built expert system.

Data Mining: Concepts &Techniques

Data Preprocessing

By

Dr. Vidushi Singh
Institute of Technolgy & Science, Ghaziabad

Content...

UNIT III - DATA MINING

Overview, Definition & Functionalities,
Data Processing,
Form of Data Preprocessing,
Data Cleaning: Missing Values, Noisy Data,
(Binning, Clustering, Regression, Computer and Human inspection),Inconsistent Data,
Data Integration and Transformation,
Data Reduction:-Data Cube Aggregation, Dimensionality Reduction, Data Compression, Numerosity Reduction

Why preprocess the data?

- Data in the real world is *Dirty*…
  - Incomplete Data: Lacking attribute values, Lacking certain attributes of interest, or containing only aggregate data

    e.g. Occupation=“ ”, year_salary = “13.000”, …
  - Inconsistent Data: Containing discrepancies in codes or names

    e.g. Age=“42” Birthday=“03/07/1997”

    Previous rating “1,2,3”, Present rating “A, B, C”

    Discrepancy between duplicate records
  - Noisy Data: Containing errors or outliers

    e.g. Salary=“-10”, Family=“Unknown”, …

Why data is *dirty*?

- Incomplete data may come from-
  - "Not applicable" data value when collected:
  - Different considerations between *the time when the data was collected* and *when it is analyzed*: Modern life insurance questionnaires would now be: Do you smoke?, Weight?, Do you drink?, …
  - Human/hardware/software problems: forgotten fields…/limited space…/year 2000 problem … etc.
- Noisy data (Incorrect values) may come from-
  - Faulty data collection instruments
  - Human or computer error at data entry
  - Errors in data transmission etc.

Why data is *dirty*?

- □ Inconsistent data may come from-
  - ➢ Integration of different data sources

    e.g. Different customer data, like addresses, telephone numbers; spelling conventions (oe, o”, o), etc.
  - ➢ Functional dependency violation

    e.g. Modify some linked data: Salary changed, while derived values like tax or tax deductions, were not updated
- □ Duplicate records also need data cleaning-
  - ➢ Which one is correct?
  - ➢ Is it really a duplicate record?
  - ➢ Which data to maintain?

    Jan Jansen, Utrecht, 1-1 2008, 10.000, 1, 2, …

    Jan Jansen, Utrecht, 1-1 2008, 11.000, 1, 2, …

Why Data Preprocessing is Important?

- No quality data, no quality mining results!
  - Quality decisions must be based on quality data

    e.g., duplicate or missing data may cause incorrect or even misleading statistics.
  - Data warehouse needs consistent integration of quality data
- Data extraction, cleaning, and transformation comprises the majority of the work of building a data warehouse
  - A very laborious task
  - Legacy data specialist needed
  - Tools and data quality tests to support these tasks

Major Tasks in Data Preprocessing

- **Data cleaning**

  Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies

- **Data integration**

  Integration of multiple databases, data cubes, or files

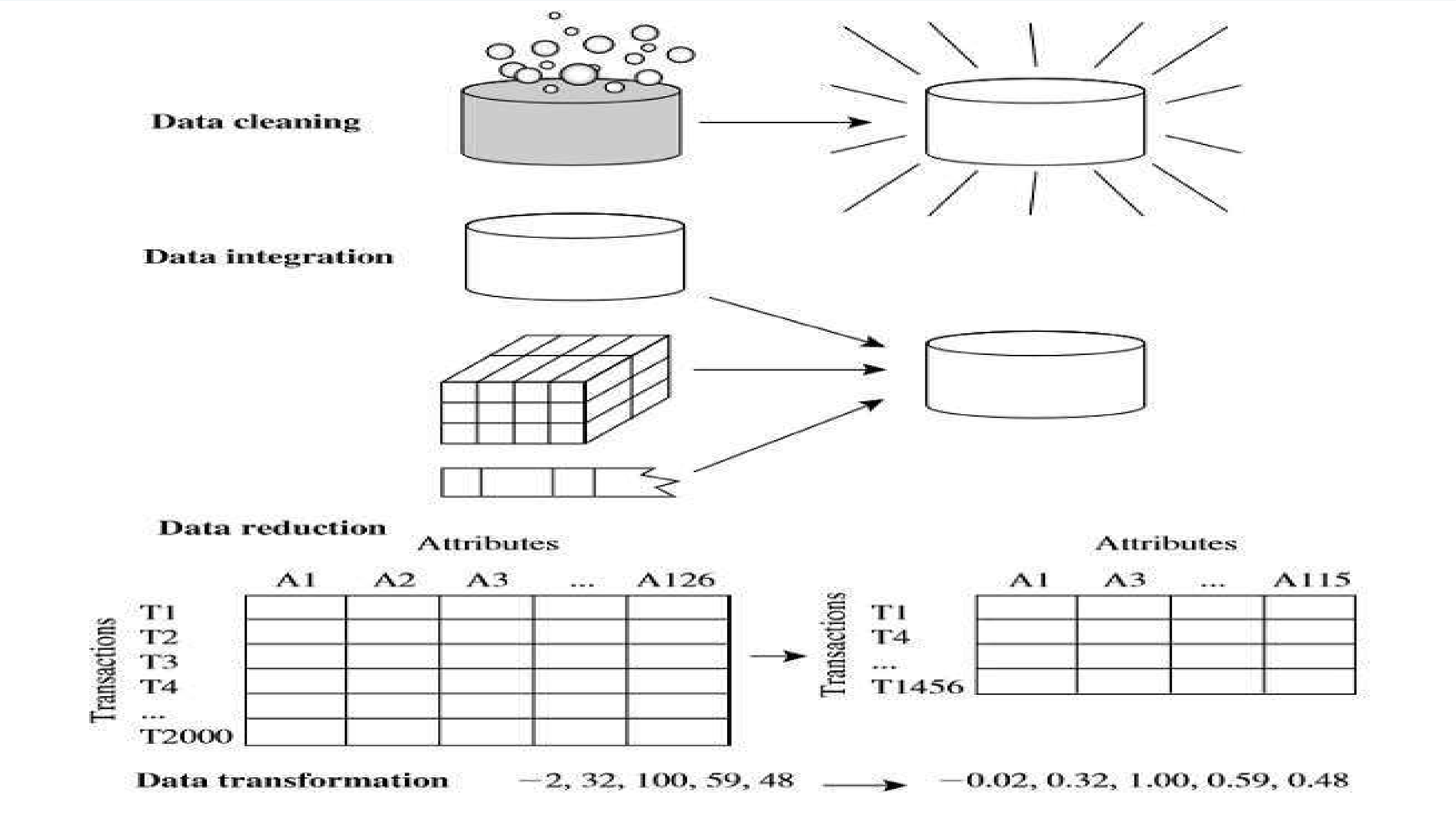
- **Data transformation**

  Normalization and aggregation

- **Data reduction**

  Obtains reduced representation in volume but produces the same or similar analytical results (restriction to useful values, and/or attributes only, etc.)

Forms of Data Preprocessing



Measuring the Central Tendency

- **Mean** (algebraic measure) (sample vs. population):

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad \frac{\sum x}{N}$$

  - Weighted arithmetic mean:

$$\bar{x} = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$$

  - Trimmed mean: chopping extreme values
- **Median** : A holistic measure
  - Middle value if odd number of values; average of the middle two values otherwise
  - Estimated by interpolation (for *grouped data*) if an interval containing the median frequency is known.
- **Mode** : Value that occurs most frequently in the data.

mean ˜ mode 3 (mean ˜ median)

Data Cleaning

- Why Data Cleaning?
  - “Data cleaning is one of the three biggest problems in data warehousing”—Ralph Kimball
  - “Data cleaning is the number one problem in data warehousing”—DCI survey
- Data cleaning tasks
  - Fill in missing values
  - Identify outliers and smooth out noisy data
  - Correct inconsistent data
  - Resolve redundancy caused by data integration

Missing Data

- Data is not always available- many tuples have no recorded value for several attributes, such as customer income in sales data
- Missing data may be due to
  - Equipment malfunction
  - Inconsistent with other recorded data and thus deleted
  - Data not entered due to misunderstanding (left blank)
  - Certain data may not be considered important at the time of entry (left blank)
  - Not registered history or changes of the data
- Missing data may need to be inferred (blanks can prohibit application of statistical or other functions)

How to Handle Missing Data?

- Ignore the tuple: usually done when class label is missing (assuming the tasks in classification—not effective when the percentage of missing values per attribute varies considerably.
- Fill in the missing value manually: tedious + infeasible?
- Use a global constant to fill in the missing value: e.g., “unknown”, a new class?!
- Use the attribute mean to fill in the missing value
- Use the attribute mean for all samples belonging to the same class to fill in the missing value: smarter
- Use the most probable value to fill in the missing value: inference-based such as Bayesian formula or decision tree

Noisy Data

- Noise: Random error or variance in a measured variable
- Incorrect attribute values may be due to
  - Faulty data collection instruments
  - Data entry problems
  - Data transmission problems
  - Technology limitation
  - Inconsistency in naming convention (H. Shree, HShree, H.Shree, H Shree etc.)
- Other data problems which requires data cleaning
  - Duplicate records (omit duplicates)
  - Incomplete data (interpolate, estimate, etc.)
  - Inconsistent data (decide which one is correct …)

How to Handle Noisy Data?

- Binning
  - First sort data and partition into (equal-frequency) bins
  - Then one can smooth by bin means, smooth by bin median, smooth by bin boundaries, etc.
- Regression
  - Smooth by fitting the data into regression functions
- Clustering
  - Detect and remove outliers
- Combined computer and human inspection
  - Detect suspicious values and check by human (e.g., deal with possible outliers)

Binning Methods for Data Smoothing

- Sorted data for price (in dollars): 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34
  - Partition into equal-frequency (equi-depth) bins:

    - Bin 1: 4, 8, 9, 15

    - Bin 2: 21, 21, 24, 25

    - Bin 3: 26, 28, 29, 34
  - Smoothing by bin means:

    - Bin 1: 9, 9, 9, 9

    - Bin 2: 23, 23, 23, 23

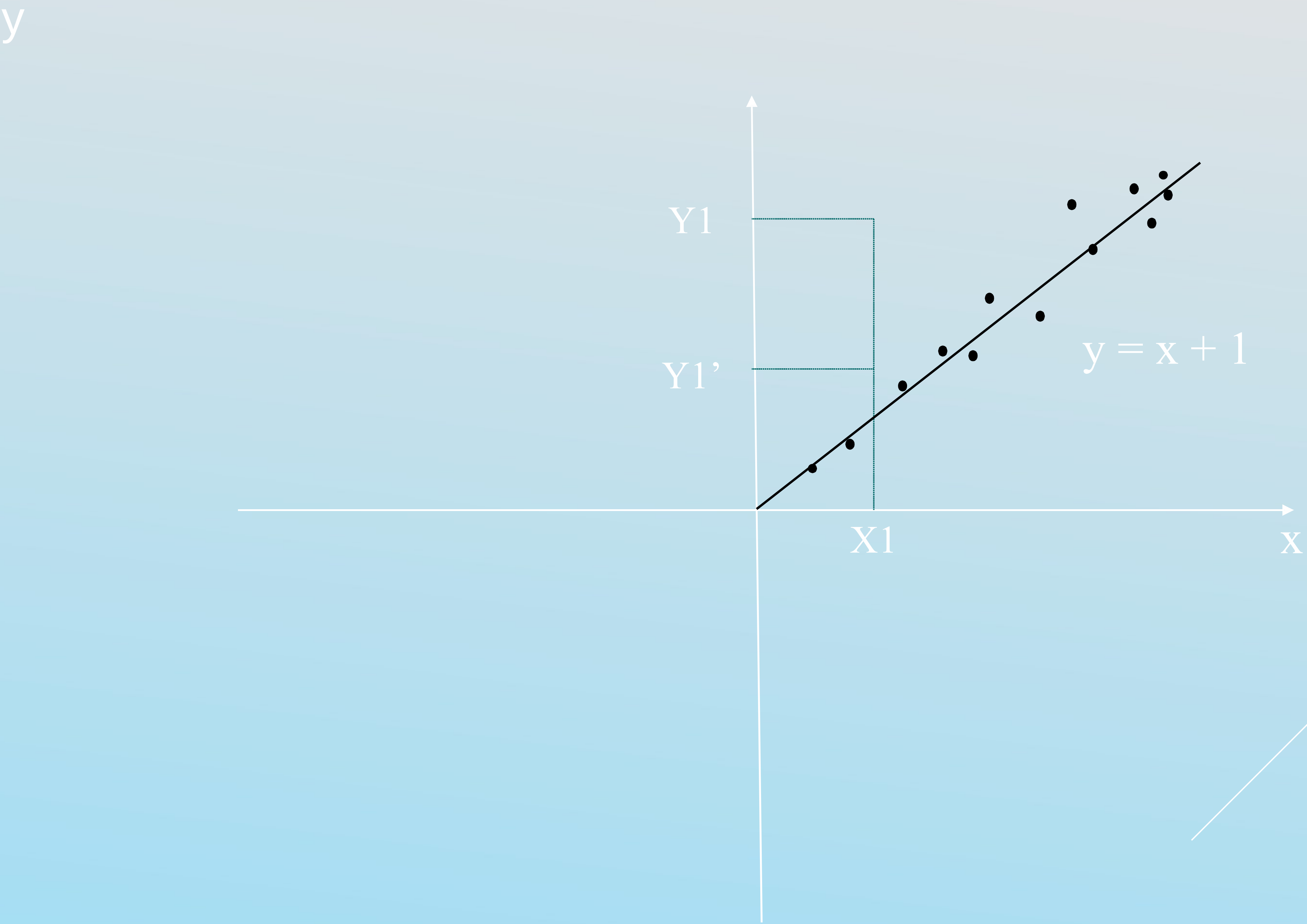
    - Bin 3: 29, 29, 29, 29
  - Smoothing by bin boundaries:

    - Bin 1: 4, 4, 4, 15(boundaries 4 and 15, report closest boundary)

    - Bin 2: 21, 21, 25, 25

    - Bin 3: 26, 26, 26, 34

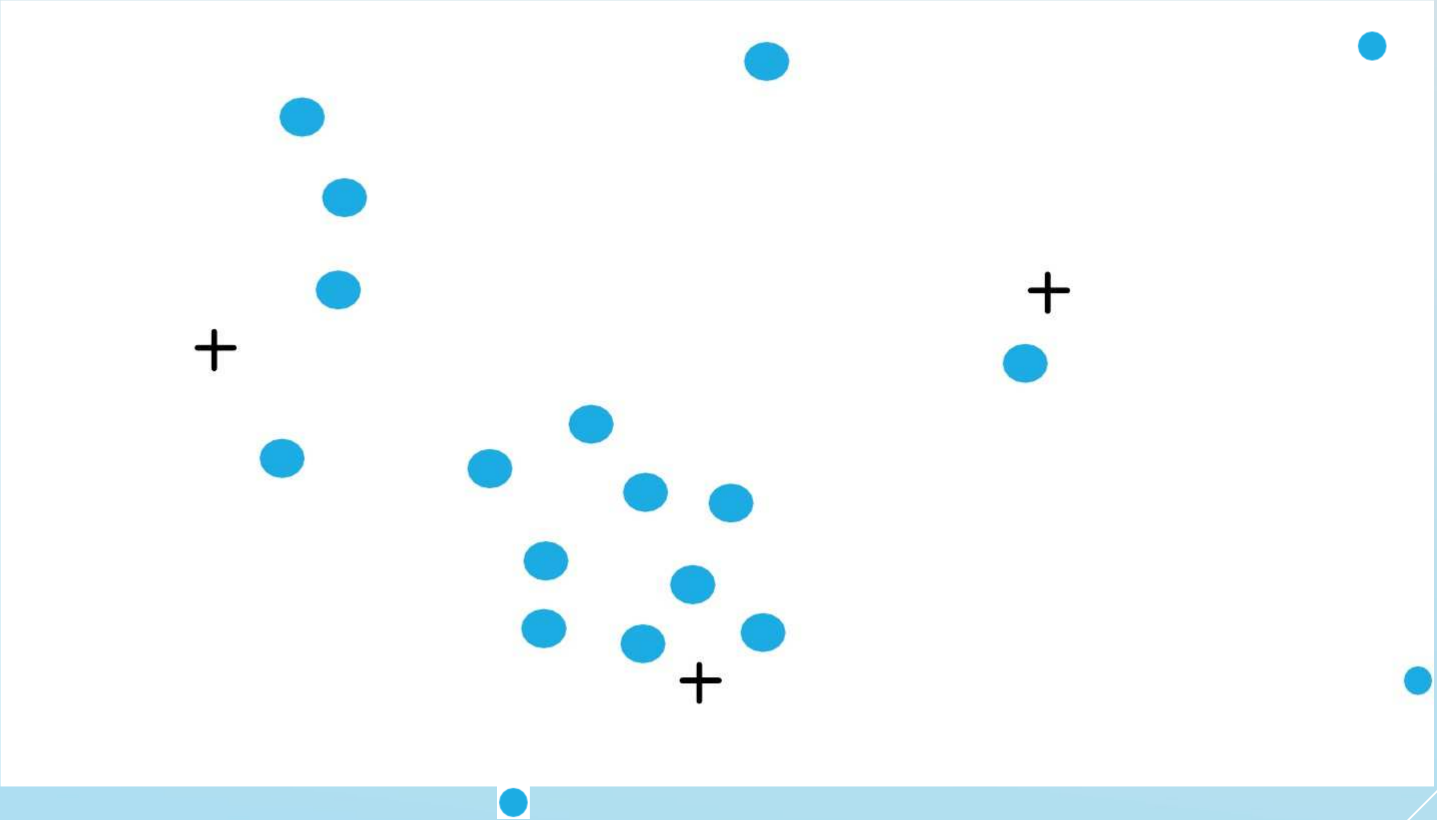
How to handle noisy data: Regression



Data Cleaning as a Process

- **Data discrepancy detection**
  - Use metadata (e.g., domain, range, dependency, distribution)
  - Check field overloading
  - Check uniqueness rule, consecutive rule and null rule
  - Use commercial tools (*Talend Data Quality Tool, Sept. 2008*)
    - **Data scrubbing**: use simple domain knowledge (e.g., postal code, spell-check) to detect errors and make corrections
    - **Data auditing**: by analyzing data to discover rules and relationship to detect violators (e.g., correlation and clustering to find outliers)
- **Data migration and integration**
  - Data migration tools: allow transformations to be specified
  - ETL (Extraction/Transformation/Loading) tools: allow users to specify transformations through a graphical user interface
- **Integration of the two processes**
  - Iterative and interactive (e.g., *Potter's Wheels*)

Handle Noisy Data: Cluster Analysis



Data integration and transformation

- **Data integration**
  - Combines data from multiple sources into a coherent store
- **Schema integration**: e.g., A.cust-id B.cust-#
  - Integrate metadata from different sources
- **Entity identification problem**
  - Identify and use *real world entities* from multiple data sources, e.g., Bill Clinton = William Clinton
- **Detecting and resolving data value conflicts**
  - For the same *real world entity*, attribute values from different sources are different
  - Possible reasons: different representations, different scales, e.g., metric vs. British units

Handling Redundancy in Data Integration

- Redundant data occur often when integration of multiple databases
  - *Object identification*: The same attribute or object may have different names in different databases
  - *Derivable data:* One attribute may be a “derived” attribute in another table, e.g.,

    annual revenue
- Redundant attributes may be able to be detected by *correlation analysis*
- Careful integration of the data from multiple sources may help reduce/avoid

  redundancies and inconsistencies and improve mining speed and quality

Data Transformation

- Smoothing: remove noise from data
- Aggregation: summarization, data cube construction
- Generalization: concept hierarchy climbing
- Normalization: scaled to fall within a small, specified range
    - min-max normalization
    - z-score normalization
    - normalization by decimal scaling
- Attribute/feature construction
    - New attributes constructed from the given ones

Data Transformation: Normalization

- Min-max normalization: to [new_minA, new_maxA]

$$v' \quad \frac{v \;\tilde{}\; min_A}{max_A \;\tilde{}\; min_A}(new\_max_A \;\tilde{}\; new\_min_A) \quad new\_min_A$$

  - Ex. Let income range $12,000 to $98,000 normalized to [0.0, 1.0]. Then $73,000 is mapped to

$$\frac{73{,}600 \;\tilde{}\; 12{,}000}{98{,}000 \;\tilde{}\; 12{,}000}(1.0 \;\tilde{}\; 0) \quad 0 \quad 0.716$$

- Z-score normalization (μ: mean, σ: standard deviation):

$$v' \quad \frac{v \;\tilde{}\; {}_A}{{}_A}$$

  - Ex. Let μ = 54,000, σ = 16,000. Then

$$\frac{73{,}600 \;\tilde{}\; 54{,}000}{16{,}000} \quad 1.225$$

- Normalization by decimal scaling

$$v' \quad \frac{v}{10^j}$$

Where *j* is the smallest integer such that Max(|v'|) < 1

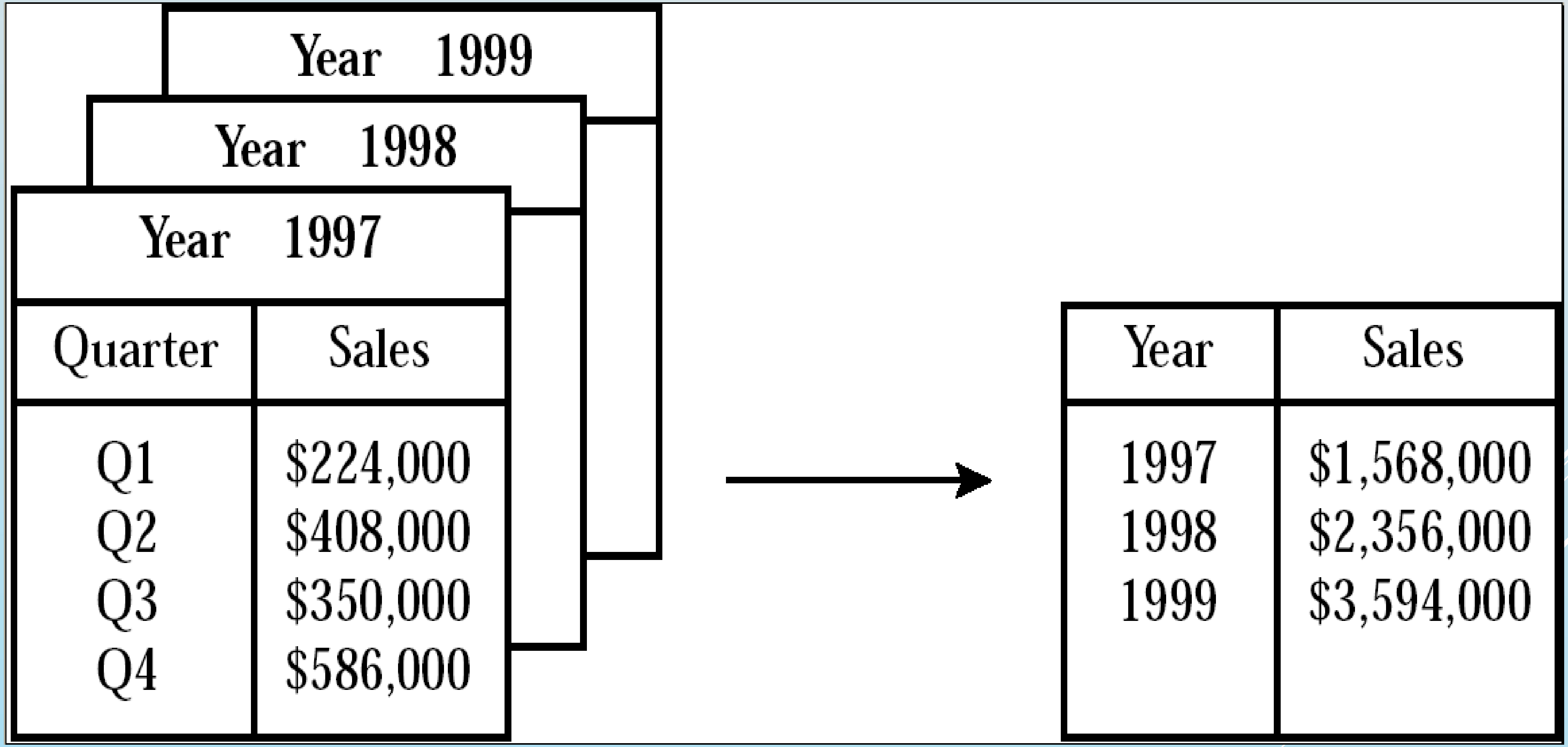
Data Reduction

- Why Data Reduction?
  - A database/data warehouse may store terabytes of data
  - Complex data analysis/mining may take a very long time to run on the complete data set
- Data reduction
  - Obtain a reduced representation of the data set that is much smaller in volume but yet produce the same (or almost the same) analytical results
- Data reduction strategies
  - Data cube aggregation:
  - Dimensionality reduction — e.g., remove unimportant attributes
  - Data Compression
  - Numerosity reduction — e.g., fit data into models

Data Cube Aggregation

- The lowest level of a data cube (base cuboid)
  - The aggregated data for an individual entity of interest
  - E.g., a customer in a phone calling data warehouse
- Multiple levels of aggregation in data cubes
  - Further reduce the size of data to deal with
- Reference appropriate levels
  - Use the smallest (in size) representation which is enough to solve the task
- Queries regarding aggregated information should be answered using the data cube, when possible

Data Cube Aggregation



Year 1999

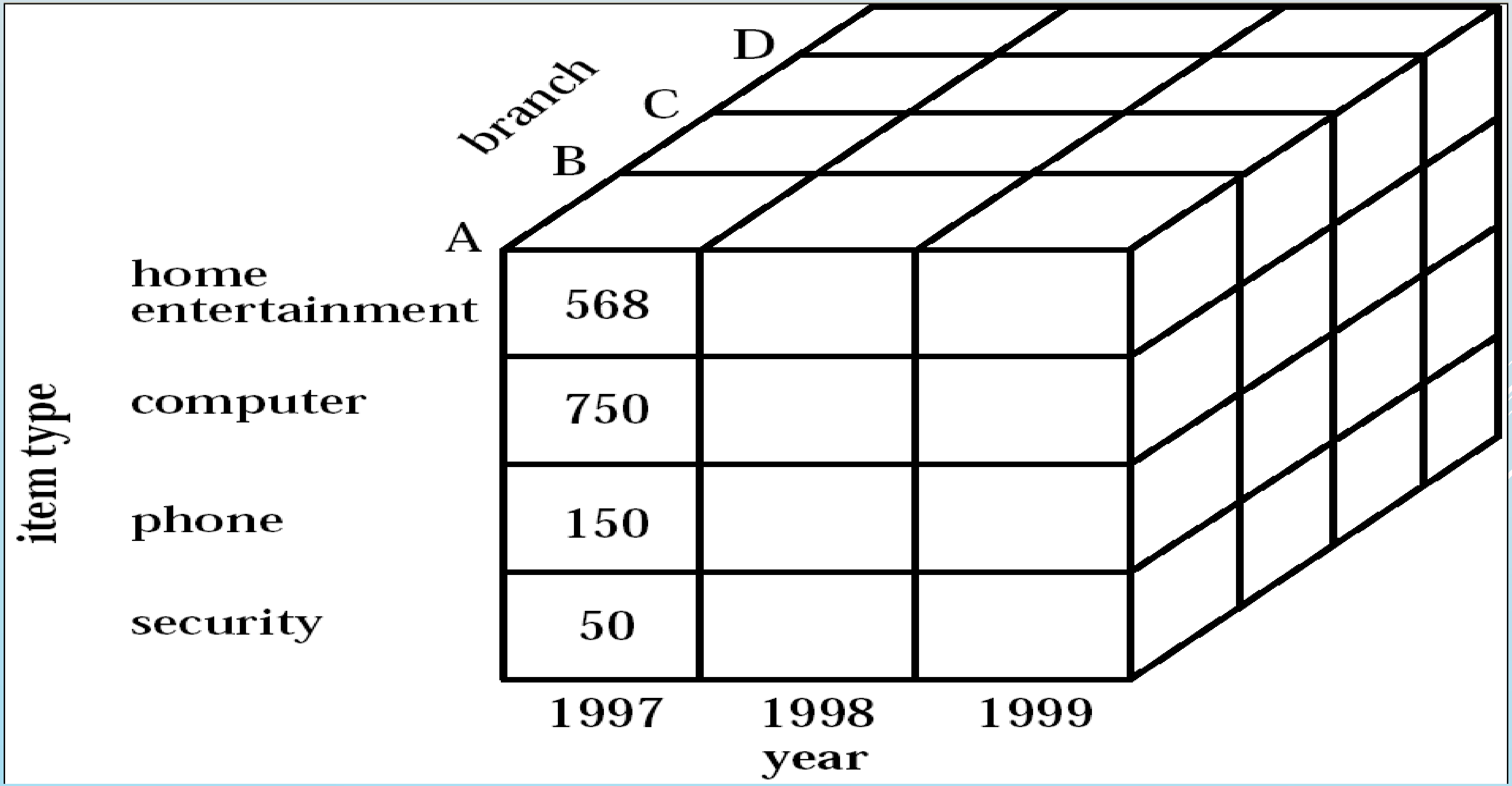
Year 1998

Year 1997

Quarter	Sales
Q1	$224,000
Q2	$408,000
Q3	$350,000
Q4	$586,000

Year	Sales
1997	$1,568,000
1998	$2,356,000
1999	$3,594,000

Data Cube Aggregation



Dimensionality Reduction

Data encoding or transformations are applied to obtain a reduced or compressed representation of original data. Data Reduction can be: lossy and lossless.

- Wavelet Transform
  - Typically lossy
  - When WT is applied to a data vector X, transform it to a numerically different vector X'. Both the data are of same length but X' can be truncated.
- Principal Components Analysis
  - Typically lossy reduction
  - Suppose that the data to be reduced consist of tuples or data vectors described by n attributes or dimensions.

Dimensionality Reduction

- Principal Components Analysis (Continued…)

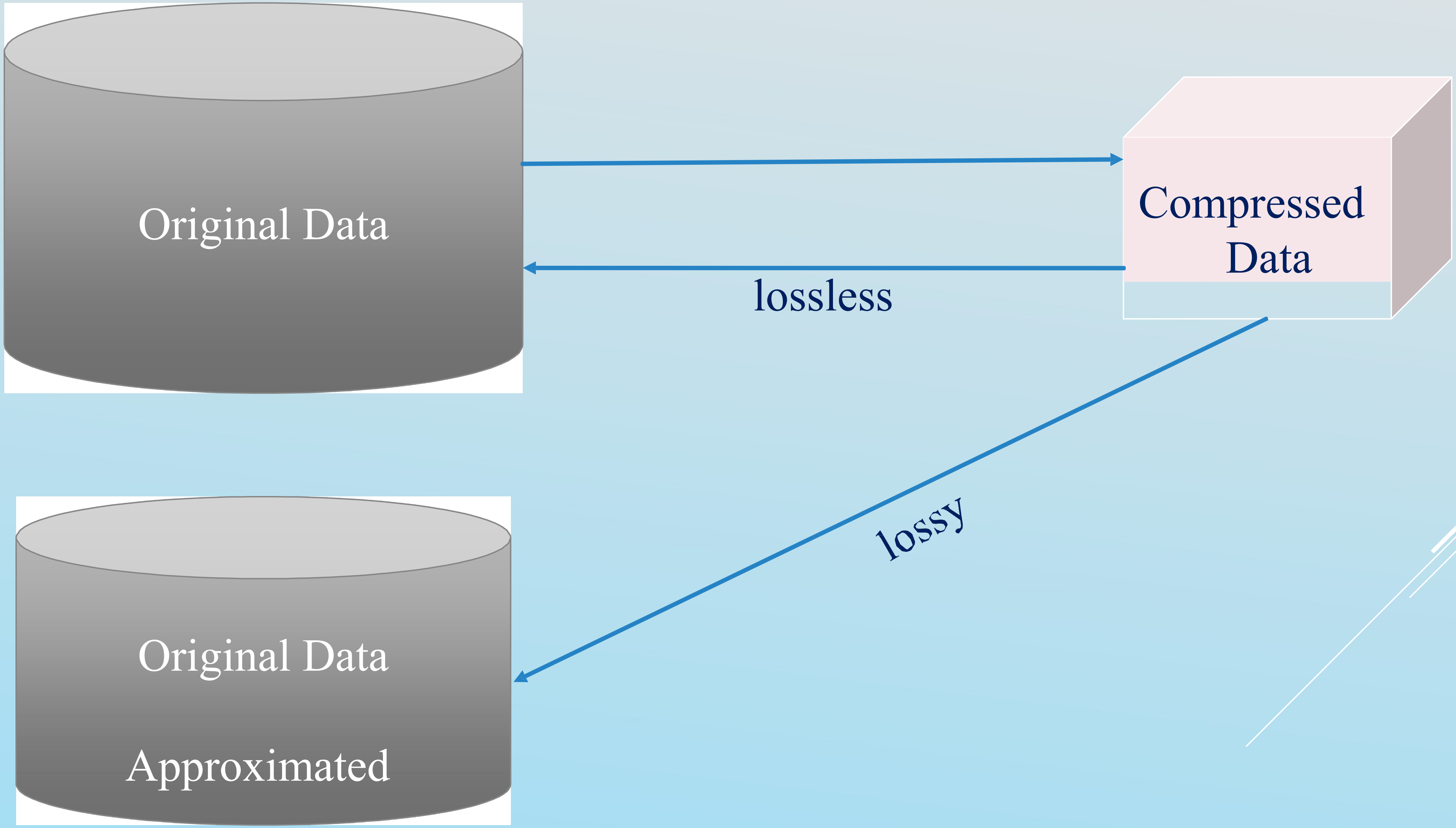
  - PCA searches for k n-dimensional orthogonal vectors that can best be used to represent the data, where k<=n. The original data are thus projected onto a much smaller space, resulting in dimensionality reduction.

Data Compression

- String compression
  - There are extensive theories and well-tuned algorithms
  - Typically lossless
  - But only limited manipulation is possible without expansion

- Audio/video compression
  - Typically lossy compression, with progressive refinement
  - Sometimes small fragments of signal can be reconstructed without reconstructing the whole

Data Compression



Original Data

Compressed Data

lossless

lossy

Original Data Approximated

Numerosity Reduction

“ Can we reduce the data volume by choosing alternative, smaller forms of data representation?” Techniques of numerosity reduction can indeed be applied for this purpose. These techniques may be parametric or non parametric.

- Regression
- Histogram
- Clustering
- Sampling

Regression

- Predict a value of a given continuous valued variable based on the values of other variables, assuming a linear or nonlinear model of dependency.
- Greatly studied in statistics, neural network fields.
- Examples:
  - Predicting sales amounts of new product based on advertising expenditure.
  - Predicting wind velocities as a function of temperature, humidity, air pressure, etc.
- Time series prediction of stock market indices.

Data Reduction Method (1): Regression

- Linear regression: Data are modeled to fit a straight line
  - Often uses the least-square method to fit the line

$$Y = w\,X + b$$

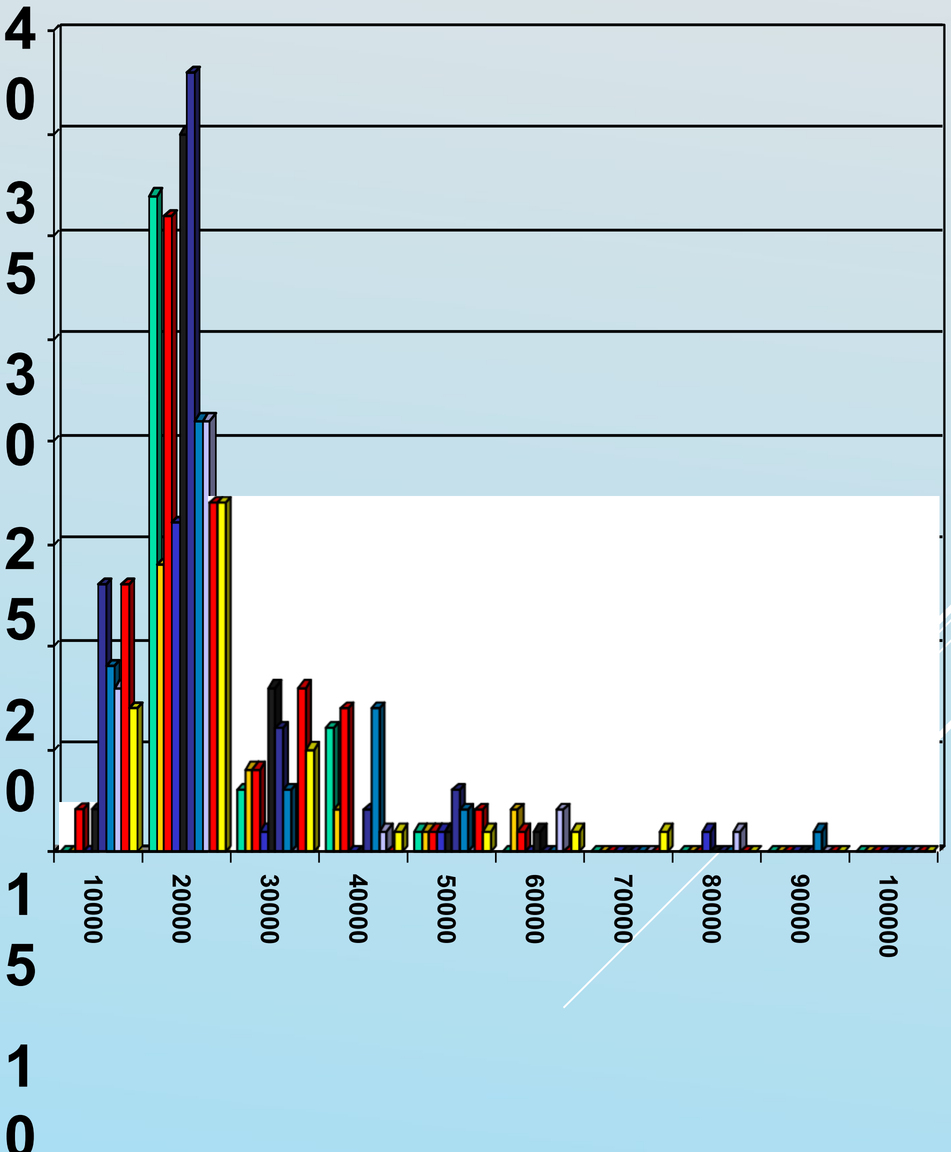
  - Two regression coefficients, *w* and *b,* specify the line and are to be estimated by using the data at hand
  - Using the least squares criterion to the known values of *Y1, Y2, …, X1, X2, ….*
- Multiple regression: Allows a response variable Y to be modeled as a linear function of a multidimensional feature vector

$$Y = b0 + b1\ X1 + b2\ X2.$$

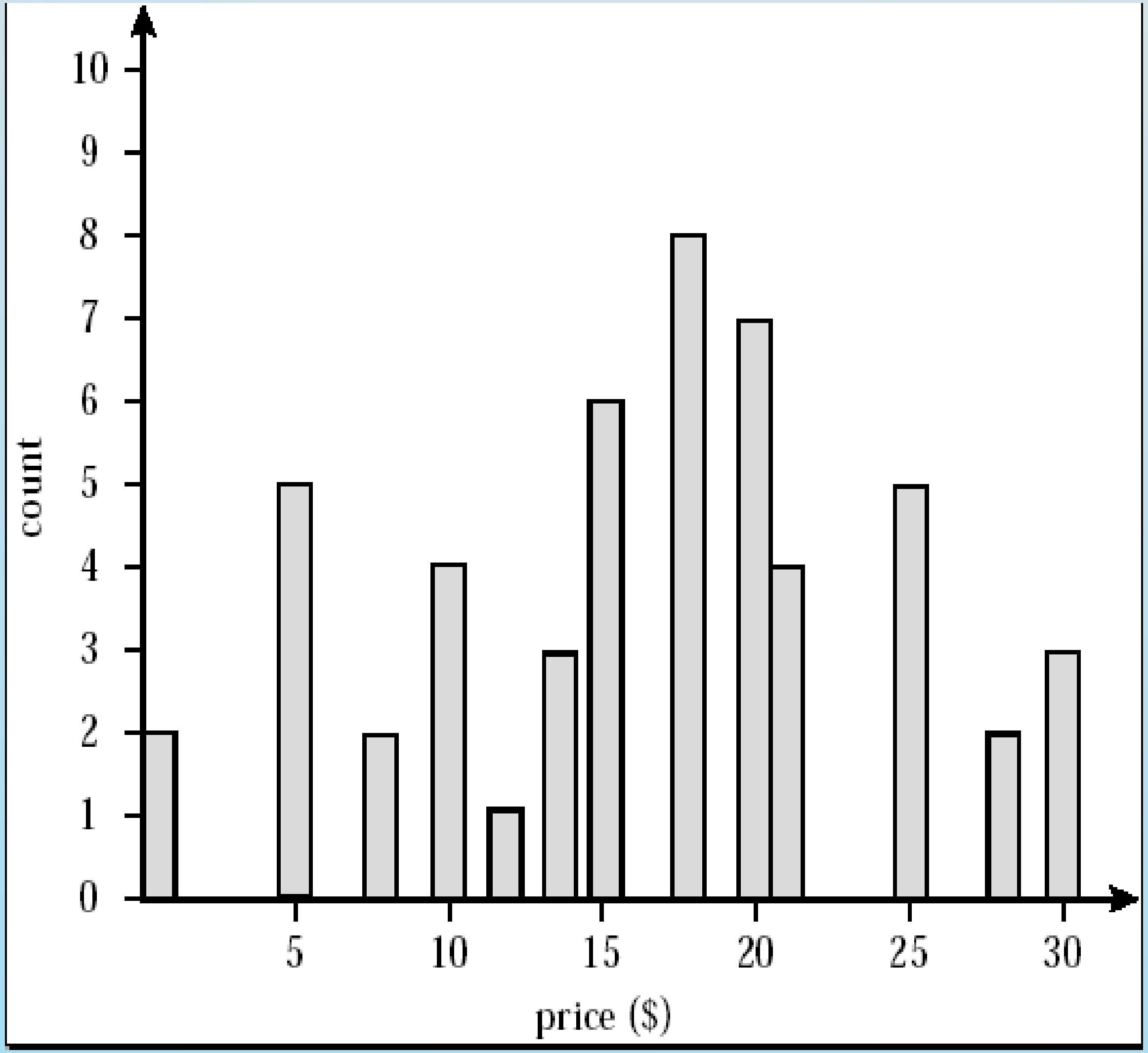
- Many nonlinear functions can be transformed into the above

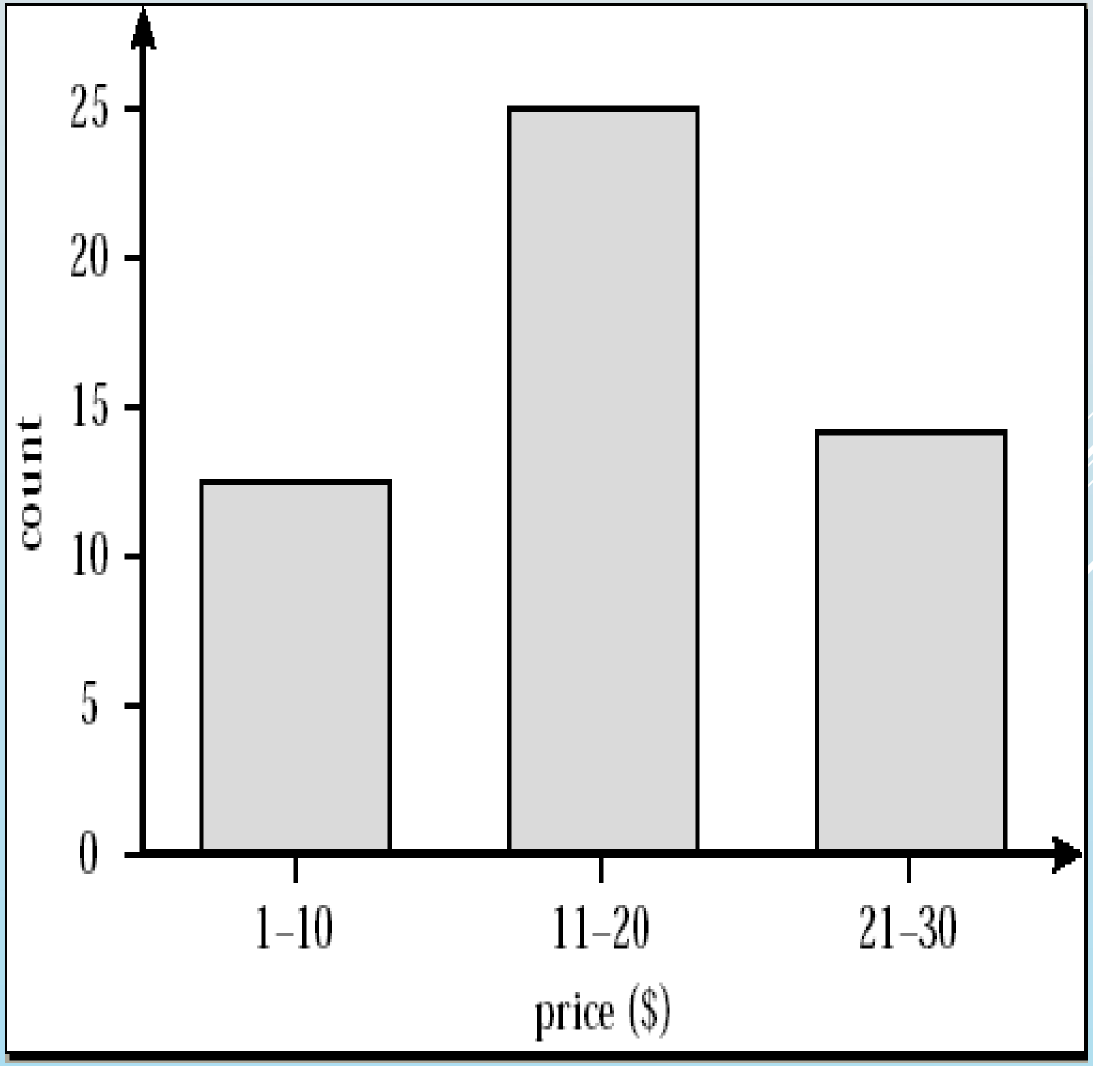
Data Reduction Method (2): Histograms

- Divide data into buckets and store average (sum) for each bucket
- Partitioning rules:
  - Equal-width: equal bucket range
  - Equal-frequency (or equal-depth)
  - V-optimal: with the least *histogram variance* (weighted sum of the original values that each bucket represents)
  - MaxDiff: set bucket boundary between each pair for pairs have the β–1 largest differences



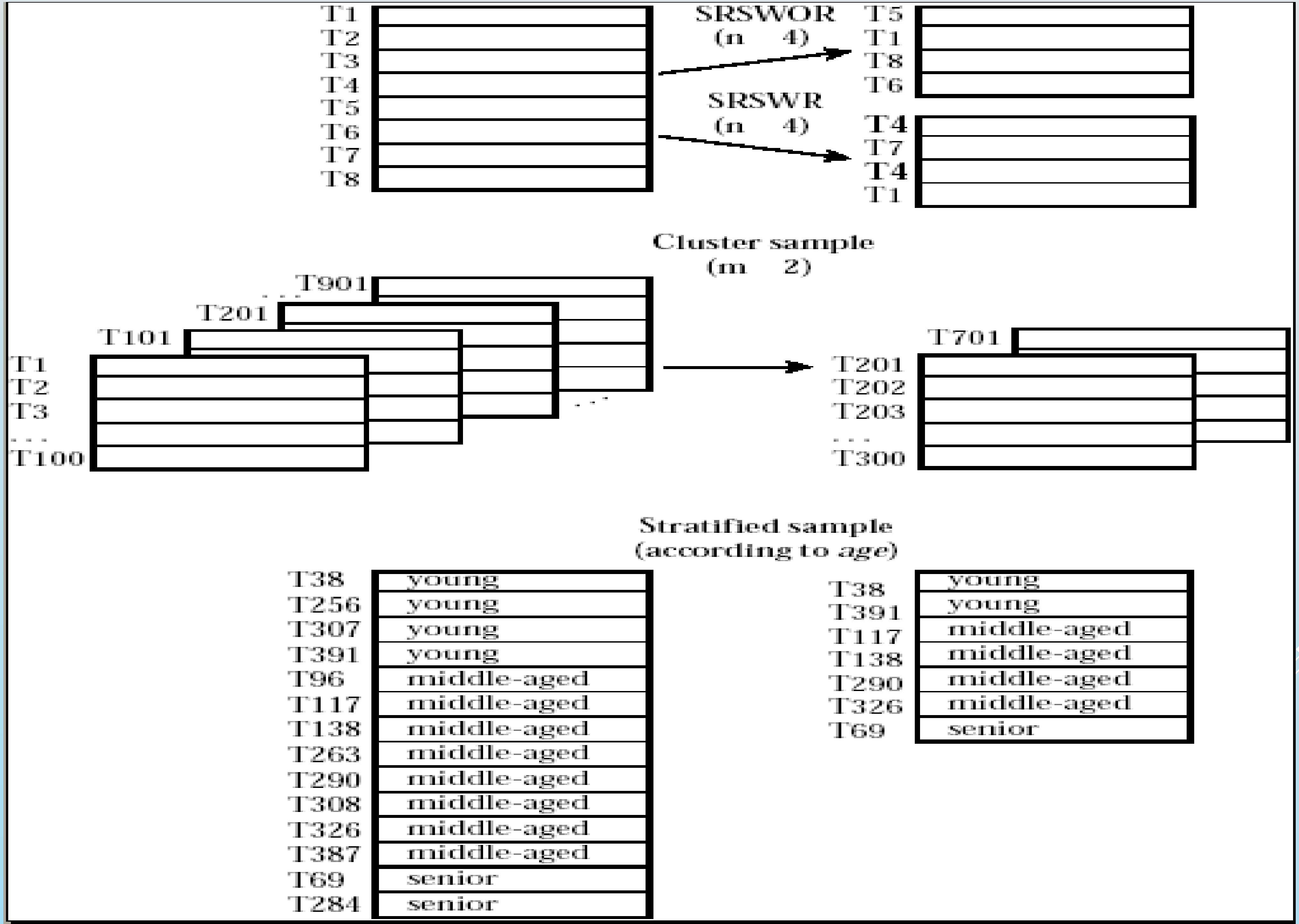
Histograms


count
10
9
8
7
6
5
4
3
2
1
0
5
10
15
20
25
30
price ($)


count
25
20
15
10
5
0
1–10
11–20
21–30
price ($)

Data Reduction Method (4): Sampling

- Sampling: Obtaining a small sample s to represent the whole data set N
- Allow a mining algorithm to run in complexity that is potentially sub-linear to the size of the data
- Choose a representative subset of the data
  - Simple random sampling may have very poor performance in the presence of skew
- Develop adaptive sampling methods
  - Stratified sampling:
    - Approximate the percentage of each class (or subpopulation of interest) in the overall database
    - Used in conjunction with skewed data
- Note: Sampling may not reduce database I/Os (page at a time)



T1
T2
T3
T4
T5
T6
T7
T8

SRSWOR
(n 4)

T5
T1
T8
T6

SRSWR
(n 4)

T4
T7
T4
T1

Cluster sample
(m 2)

T901
. . .
T201
T101
T1
T2
T3
. . .
T100
. . .

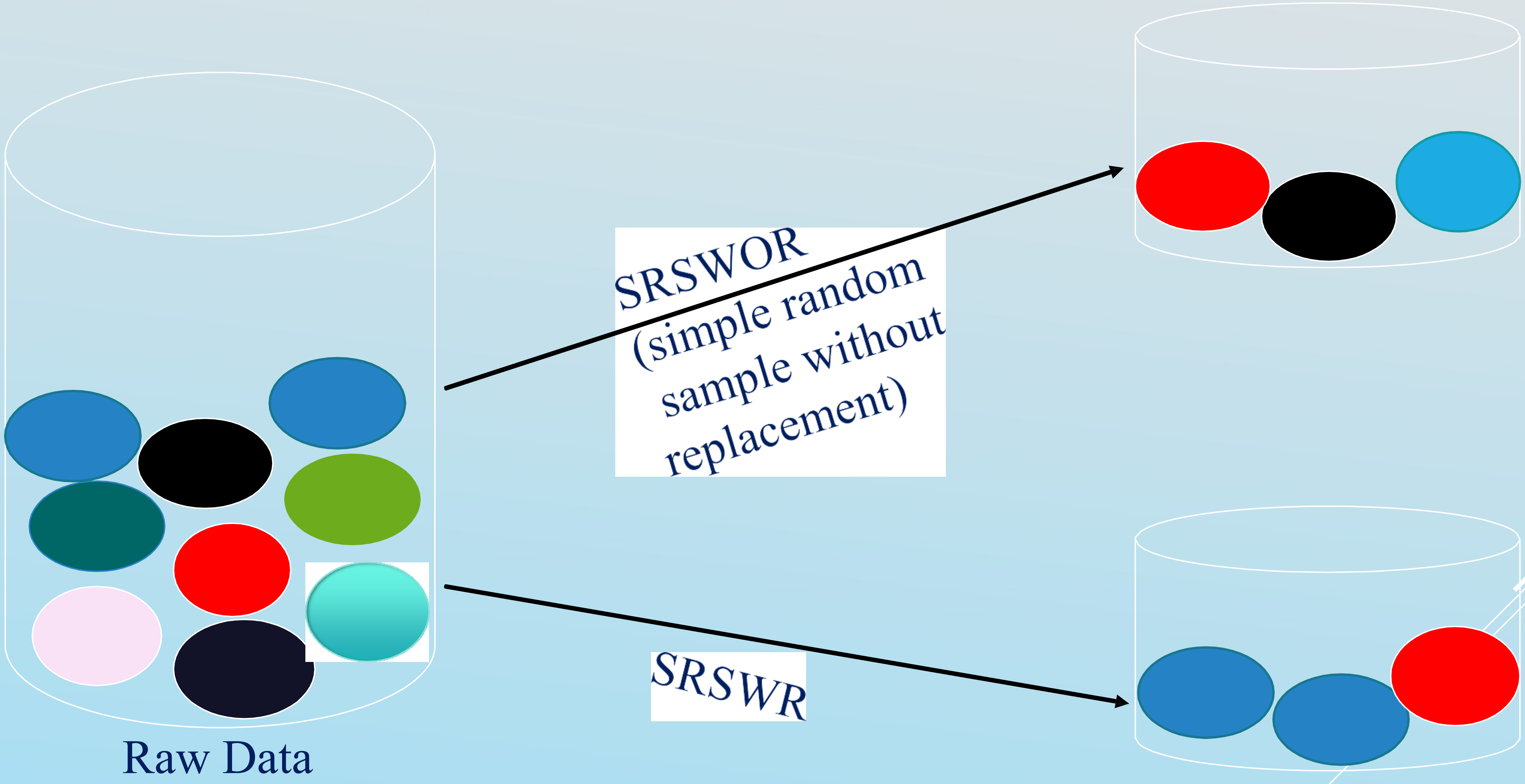
T701
T201
T202
T203
. . .
T300

Stratified sample
(according to *age*)

T38	young
T256	young
T307	young
T391	young
T96	middle-aged
T117	middle-aged
T138	middle-aged
T263	middle-aged
T290	middle-aged
T308	middle-aged
T326	middle-aged
T387	middle-aged
T69	senior
T284	senior

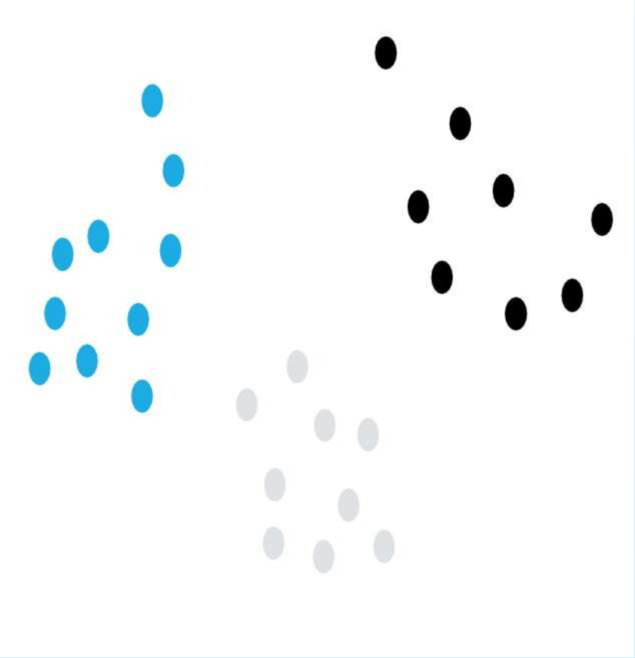
T38	young
T391	young
T117	middle-aged
T138	middle-aged
T290	middle-aged
T326	middle-aged
T69	senior

Sampling: with or without Replacement



Sampling: Cluster or Stratified Sampling

Raw Data Sample



Cluster/Stratified