Unit-1

Syllabus → Overview, Definition, Data Warehousing Components, Building a Data Warehouse, Warehouse Database, Mapping the DataWarehouse to a Multiprocessor Architecture, Difference b/w Database System and DataWarehouse, MultiDimensional Data Model, Data Cubes, Stars, Snow Flakes, Fact Constellations, Concept Hierarchy, Process Architecture, 3 tier Architecture, Data Marting

Datawarehouse

→ "A datawarehouse is a subject-oriented, integrated, time-variant and non-volatile collection of data in support of the management's decision making process."

- Subject-oriented :→ A datawarehouse is organized around major subjects such as customer, supplier, product and sales.

→ Rather than concentrating on the day-to-day operations and transaction processing of an organization, a datawarehouse focuses on the modeling and analysis of data for decision makers.

→ Hence datawarehouses typically provide a simple and and concise view of particular subject issues by excluding data that are not useful in the decision support process.

Scanned by CamScanner

- **Integrated** → A data warehouse is usually constructed by integrating multiple heterogenous sources, such as relational databases, flat files and online transaction records.
  → Data cleaning and data integration techniques are applied to ensure consistency in naming conventions, encoding structures, attribute measures and so on.
- **Time-variant** →
  → Data are stored to provide information from an historic perspective (e.g. past 5-10 years)
- **Non-volatile** →
  → A data warehouse is always a physically separate store of data transformed from the application data found in the operational environment.
  → Due to this separation, a data warehouse doesn't require transaction processing, recovery and concurrency control mechanisms.
  → It usually requires only two operations in data accessing: initial loading of data & access of data.

Datawarehousing Components

There are 7 major components of datawarehouse:—

1) DataWarehouse Database.
2) Sourcing, Acquisition, cleanup & Transformation Tools
3) Meta data
4) Access Tools
5) Data Marts
6) Data Warehouse Administration & Management

Scanned by CamScanner

7) Information Delivery System

1) Data Warehouse Database

→ The central data warehouse database is the cornerstone of the data warehousing environment.

→ This database is almost always implemented on the relational database management system (RDBMS) However this ~~brick~~ kind of implementation is often constrained by the fact that traditional RDBMS products are optimized for transactional database processing

→ 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 technologies approaches to the data warehouse database.

→ These approaches are :-

i) Parallel relational database design for scalability

ii) Multidimensional databases :- They enable 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, Clean up and Transformation Tools

→ The data sourcing, clean up, 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 decision support tool.

Scanned by CamScanner

→ 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.

There are some issues ~~which~~ from which these tools have to deal with :-

1) Database heterogeneity
2) Data heterogeneity

3) Meta data

→ Meta data are data about data. When used in data warehouse, metadata are the data that define warehouse objects.

→ Metadata is like index of data warehouse. It gives us clue where data is ~~used~~ stored ~~to the user~~ in the warehouse.

→ Metadata are created for the data names and definitions of the given warehouse.

→ A metadata repository should contain the following:-

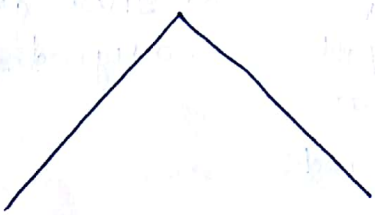
* A description of the data warehouse structure which includes the warehouse schema, view, dimensions, hierarchies and derived data definitions as well as data mart locations and contents.

* Operational metadata
  - It includes data lineage (history of migrated data and the sequence of transformations applied to it), currency of data (active, archived)

Scanned by CamScanner

* Mapping from the operational environment to the data warehouse, which includes source database and their contents
* Data related to system performance
* Business metadata which includes business terms & definitions.

Types of meta data



Operational Metadata or Technical metadata

It contains information about warehouse data for use by warehouse designers and administrators when carrying out warehouse development and management tasks.

It includes :-

→ Information about data sources.
→ Transformation descriptions i.e. the mapping method from operational databases into warehouse.
→ Warehouse object and data structure definitions for data targets.
→ The rules used to perform data cleanup and data enhancement.
→ Data mapping operations when capturing data from source systems and applying it to ~~convert~~ target warehouse database

Business Metadata

→ It contains information that gives users an easy to understand perspective of the information stored in data warehouse.

Its documents contains information about :-

→ Subject areas & information ~~that~~ object type, including queries, reports, images, video and audio clips
→ Internet home page
→ Data history, extract audit trial, usage data
→ Information related to information delivery system.

Scanned by CamScanner

4) Access Tools

Users interact with the data warehousing using front-end tools. Following are some of the tools :-

a) Query and reporting tools

This category is further divided into two groups :-

i) Reporting tools (two types)

Production reporting tools

→ let companies generate regular operational reports or support high volume batch jobs such as calculating and printing paychecks.

Report writers

→ These are desktop tools designed for end users.

ii) Managed query tools

These tools shield end users from the complexities of SQL and database structures by inserting a metalayer between users and database.

Metalayer → is a Software that provide subject oriented views of database.

b) Application Development Tools

→ When the analytical needs of the data warehouse user community exceed the built-in capabilities of query & reporting tools.

→ In that case for such complex queries application development tools are used.

→ Some of these application development tools platforms integrate with popular OLAP tools and can access all major database systems.

→ Ex of Application development environments - PowerBuilder, Visual Basics.

c) OLAP tools

c) Data

→ These tools are based on concepts of multidimensional databases and allow user to analyse data using multidimensional complex views.

Scanned by CamScanner

d) Data mining tools

Data mining is the process of discovering meaningful new correlations, patterns and trends by digging into large amounts of data stored in warehouse using AI & statistical as well as mathematical techniques. Some of the data mining tools are:- Weka, Rapid Miner, Orange etc.

5) Data Marts

→ Datamart is a subset of datawarehouse and is usually oriented to a specific business line or group.
→ The information in data marts pertains to a single department.

6) Datawarehouse Administration & management

Managing datawarehouse includes:-
→ Security and priority management.
→ Monitoring updates from multiple sources.
→ Data quality checks.
→ Managing & updating metadata
→ Auditing and reporting datawarehouse usage & status.
→ Purging data
→ Backup & Recovery

7) Information Delivery System

The Information Delivery System is responsible for delivering the warehouse information to one or more destinations according to specified scheduling algorithms.

Scanned by CamScanner

Building a DataWarehouse

Following are some of points to be considered while building a data warehouse :-

1) Business considerations

On the basis of investment two approaches are followed :-

Top-down approach

→ Organization has developed an enterprise data model, collected enterprisewide business requirements and decided to build an enterprise datawarehouse with subset datamarts.

Bottom-up approach

Business priorities resulted in developing individual datamarts which are then integrated into the enterprise data warehouse.

2) Design Considerations

Following general factors are considered while designing a datawarehouse :-

→ Heterogenity of data sources, which affects data conversion quality and timeliness.
→ Use of historical data, which implies that data may be old.
→ Tendency of databases to grow very large.

In addition to general considerations there are several specific points relevant to datawarehouse design :-

a) Data content

↳ The content & structure of the datawarehouse are reflected in its datamodel
↳ The datamodel is the template that describes how information will be organized within integrated warehouse framework.
** ↳ It identifies major subjects and relationship of the model including keys, attributes & attribute groupings.

Scanned by CamScanner

b) Meta data

↳ A datawarehouse design should ensure that there is a mechanism that populates and maintains the meta data repository and that all access paths to the data warehouse have metadata as an entry point.

c) Data distribution

↳ As database size continues to grow, so it may rapidly outgrow a single server.

↳ So, the warehouse designers should know how the data should be divided across multiple servers.

d) Tools

↳ Designers should consider which tools to be used for transformation, end-user query reporting etc.

e) Performance considerations

↳ Rapid query processing is a highly desired feature that should be designed into datawarehouse.

Nine decisions in the design of a datawarehouse

1. Choosing the subject matter
2. Deciding what a fact table represents.
3. Identifying and conforming the dimensions.
4. Choosing the facts.
5. Storing precalculations in the fact table.
6. Rounding out the dimension tables.
7. Choosing the duration of the database.
8. The need to track slowly changing dimensions.
9. Deciding the query priorities and the query modes.

3) Technical Considerations

A no. of technical issues are to be considered when designing and implementing a datawarehouse environment :-

i) The hardware platform
ii) DBMS that supports the warehouse database.
iii) Communication infrastructure that connects the warehouse, data marts, operational systems and end users. ↳ bandwidth need to be considered.

Scanned by CamScanner

4) Implementation Considerations

Following are the points need to be considered for implementation :-

a) Access Tools

Currently, no single tool on the market can handle all possible data warehouse access needs. Therefore most implementations rely on a suite of tools. The best way to choose this suite includes the definition of different types of access to the data and selecting the best tool for that kind of access. Examples of access types :-

i) Statistical analysis
ii) Multivariable analysis
iii) Data visualization, graphing, charting & pivoting

b) Data extraction, cleanup, transformation and migration

As a component of the datawarehouse architecture, proper attention must be given to data extraction, which represents a critical success factor for a data warehouse architecture. Specifically when implementing data warehouse, several selection criteria that affect the ability to transform, consolidate, integrate and repair the data should be considered.

Mapping DataWarehouse To a Multiprocessor Architecture

↳ The organizations that embarked on data warehousing development deal with ever-increasing amounts of data.

↳ So, the size of data warehouse rapidly approaches the point where the search for better performance and scalability becomes a real necessity.

Scanned by CamScanner

↳ This search is persuing two goals:-

Speed-up → the ability to execute the same request on the same amount of data in less time.

↳ Even when no of processors increase, we get the same performance. Say, there are 5 processors & we get the response in 10 milliseconds with 1 processor as well as with 5 processors. So, speed increased 5 times.

Scale up → the ability to obtain the same performance on the same request as database size increases.

* To achieve these goals we use parallelism.

Parallelism → The use of parallel hardware architecture by implementing multiserver and multithreaded systems is called parallelism.

↳ This is done to handle large no. of client requests efficiently.

Types of Parallelism

1) Horizontal Parallelism →

↳ It means that the database is partitioned across multiple disks and parallel processing occurs within a specific task that is performed concurrently on different processors against different sets of data.

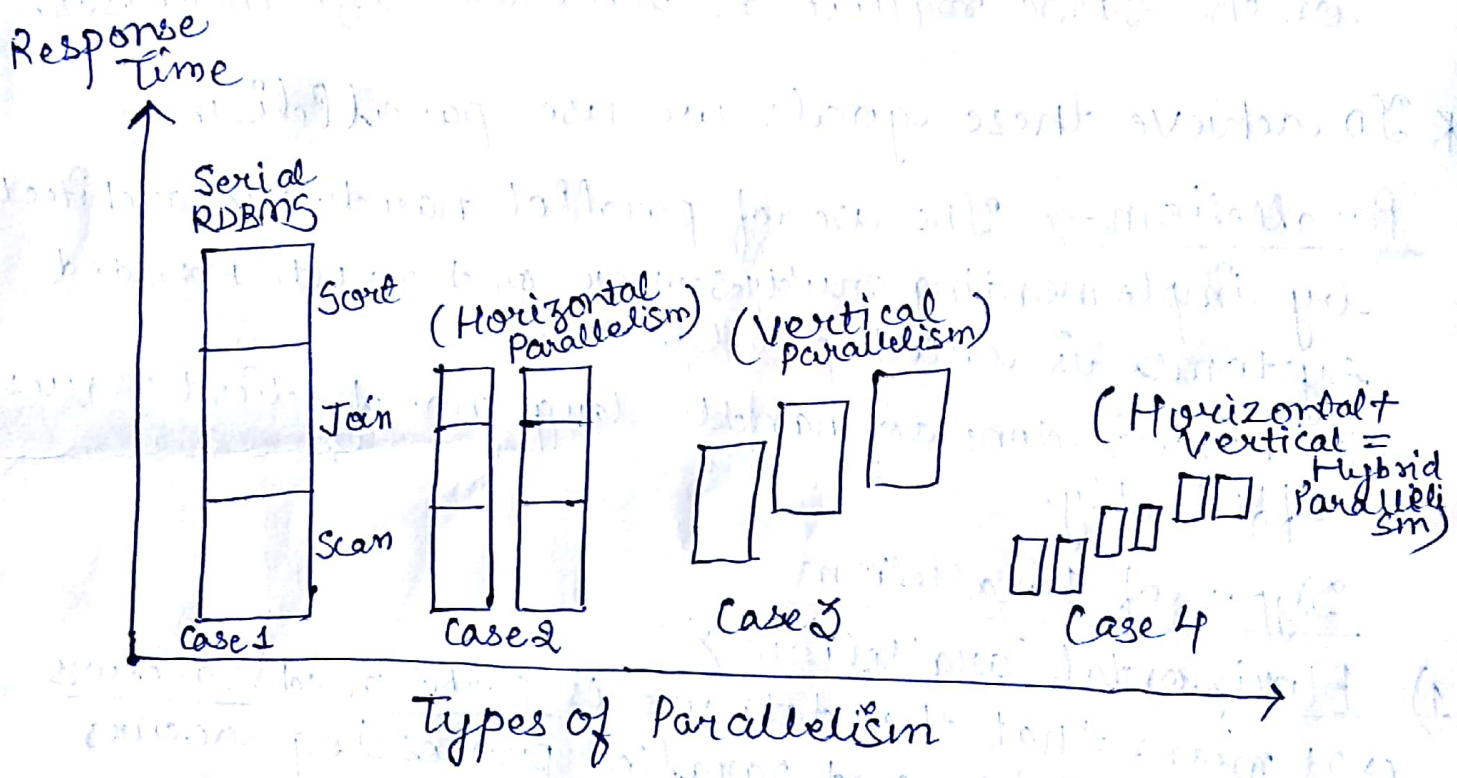
↳ For example a database is partitioned in 5 disks and a search query comes, then that query will work as 5 threads and will search in 5 disks with different data. So, performance increase.

Scanned by CamScanner

2) Vertical Parallelism

↳ This occurs among different tasks - all component query operations are executed in parallel in a pipelined fashion i.e. an O/P from one task becomes an I/P into another task as soon as records becomes available.

Graph showing Response time on different types of Parallelism



→ The given graph shows that hybrid parallelism has minimum response time.

Data Partitioning

→ Data Partitioning is ~~the~~ key requirement for effective parallel execution of database tables across multiple disks so that I/O operations such as read and write can be performed in parallel.

Scanned by CamScanner

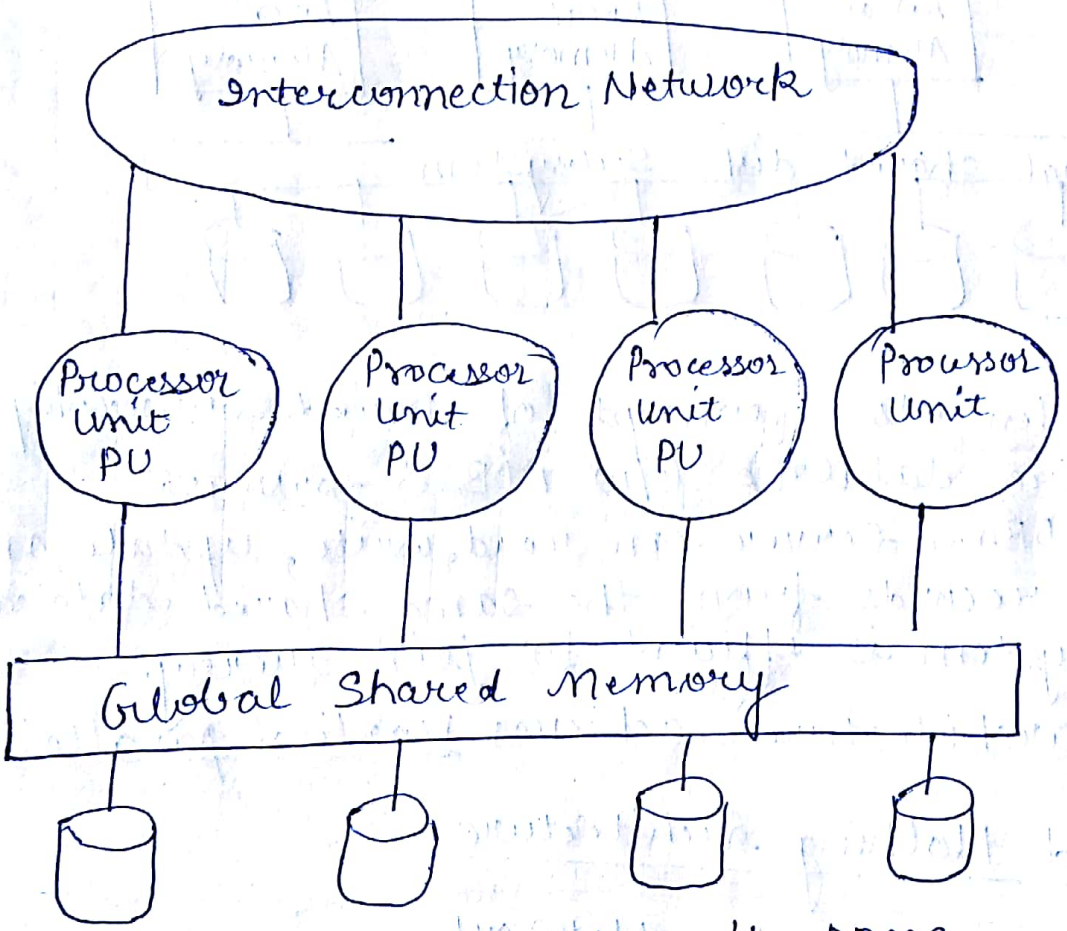
Partitioning can be done randomly or intelligently.

Random Partitioning → includes random data striping across multiple disks on a single server.

Intelligent Partitioning → assumes that DBMS knows where a specific record is located and does not waste time searching for it across all disks.

Database Architectures for Parallel Processing

1) Shared-memory Architecture / Shared everything



→ In shared memory systems, the DBMS assumes that the multiple database components executing SQL statements communicate with each other by exchanging messages and data via shared memory.

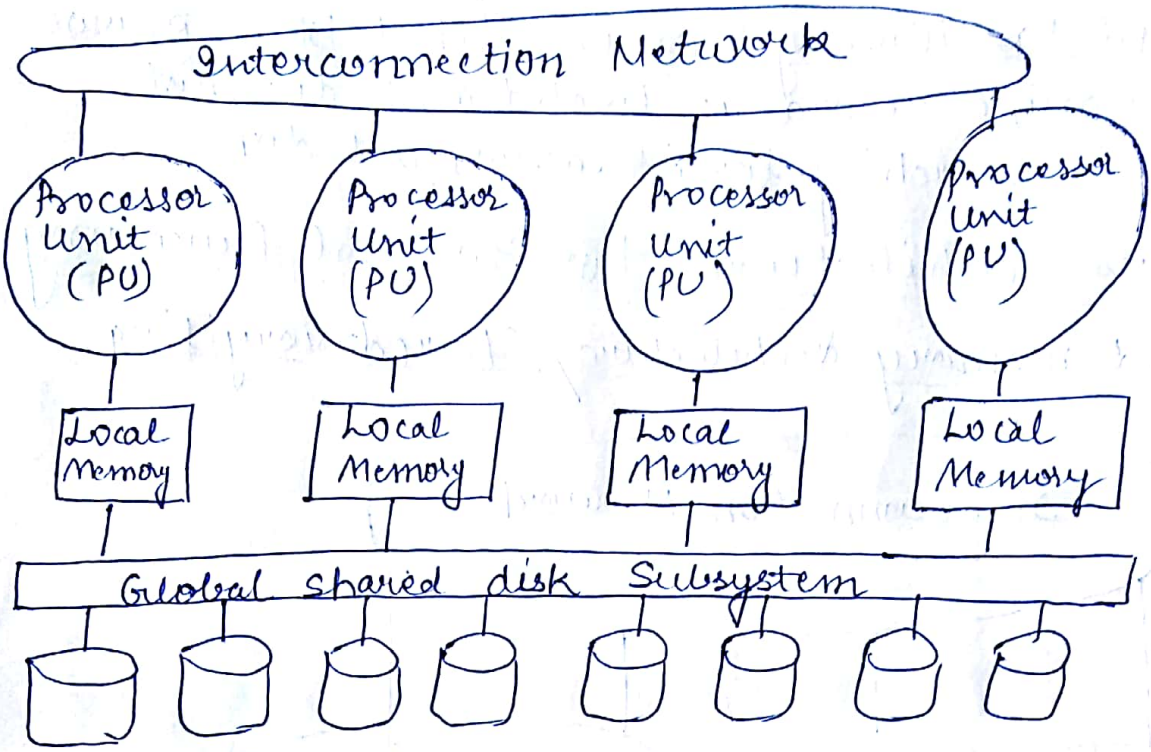
→ All processors have access to all data which is partitioned across local disks.

→ This is a key design approach of thread based implementation, threads provide better resource utilization and faster context switching, thus

Scanned by CamScanner

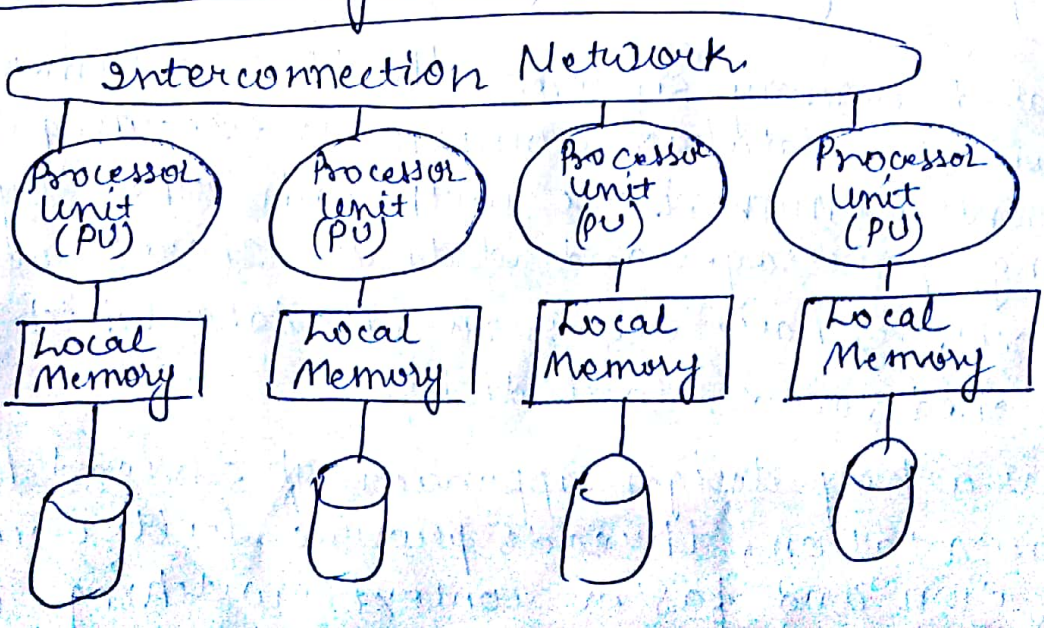
provide better scalability.
↳ This architecture achieves horizontal parallelism.

2) Shared-disk architecture



↳ It implements a concept of shared ownership of the entire database b/w RDBMS servers.
↳ Each RDBMS server can read, write, update and delete records from the same shared database
↳ This system is efficient for join query.
↳ This architecture achieves Vertical parallelism.

3) Shared Nothing Architecture



Scanned by CamScanner

↳ The data is partitioned across all disks, and DBMS is partitioned across multiple machines.
↳ Each processor has its own memory and disk and communicates with other processor by exchanging messages and data over the interconnection network.
↳ This architecture achieves hybrid (H+V) Parallelism.

Difference b/w Database System and DataWarehouse

OLTP → The major task of online operational database system is to perform online transaction and query processing. These systems are called online transaction processing (OLTP) systems.
↳ It involves day-to-day operations.

OLAP → Datawarehouse systems provides data analysis and decision making.
↳ These systems organize and present data in various formats depending upon diverse needs of different users. These systems are known as Online Analytical processing

Feature	OLTP	OLAP
Characteristic	operational processing	informational processing
Orientation	transaction	analysis
User	clerk, DBA	knowledge worker (manager, executive, analyst)
Function	day-to-day operations	long-term informational requirements decision support
DB design	ER based	star/snowflake
Data	current, up-to-date	historic
View	detailed,	summarized, multidimensional
Unit of work	short, simple transaction	complex query
Access	read/write	mostly read.
Focus	data in	information out

Scanned by CamScanner

DB size	GB to high-order GB	≥ TB
Priority	high performance, high availability	high flexibility, end-user autonomy
No of users	thousands	hundreds
No of records accessed	tens	millions

Major distinguishing features of OLTP and OLAP are summarized as follows:-

1) User and system orientation
   - ↳ An OLTP system is customer-oriented and is used for transaction and query processing by clerks, clients and IT professionals.
   - ↳ An OLAP system is market-oriented and is used for data analysis by knowledge workers including managers, executives and analysts.

2) Data contents
   - ↳ An OLTP system manages current data, typically are too detailed to be easily used for decision making.
   - ↳ An OLAP system manages large amount of historic data, provides facilities for summarization & aggregation.
   - ↳ These features make the data easier to use for informed decision making.

3) Database Design
   - ↳ An OLTP system usually adopts an ER data model
   - ↳ An OLAP system usually adopts either a star or snowflake model.

Scanned by CamScanner

4) View → An OLTP system focuses mainly on the current data without referring to historic data or data in different organizations.
↳ OLAP system spans multiple versions of a database schema. Because of their huge volume, OLAP data are stored on multiple storage media.

5) Access Pattern
↳ The access patterns of an OLTP system consist of short atomic transactions.
↳ OLAP systems are mostly read-only operations, consists mainly complex queries.

Why Have a Seperate Data Warehouse??

"Why not perform online analytical processing directly on such databases instead of spending additional time and resources to construct a seperate data warehouse"

Following are the reasons :- —

a) To help promote high performance of both systems.
- Processing OLAP queries in operational databases would substantially degrade the performance of operational tasks.

b) Concurrency control and recovery mechanisms if applied for OLAP operations may reduce the execution of concurrent transactions.

Scanned by CamScanner

MultiDimensional Data Model

↳ Datawarehouse and OLAP tools are based on a multi-dimensional data model. This model views data in the form of data model.

Data Cube

→ A data cube allows data to be modeled and viewed in multiple dimensions.

→ It is defined by dimensions and facts.

Dimensions → Dimensions are the entities with respect to which an organization wants to keep records.

↳ For example, AllElectronics may create a sales datawarehouse in order to keep records of the store's sales with respect to the dimensions time, item, branch and location.

↳ Each dimension may have a table associated with it called dimension table. For example item may contain attributes item_name, brand and type.

Facts → A multidimensional data model is typically organized around a central theme, such as sales. This theme is represented by a fact table.

↳ Facts are numeric measures.

↳ Fact table contains measures or facts as well as keys to each of the related dimension tables.

↳ Measures → For ex. sales of items can be a measure.

Scanned by CamScanner

↳ In datawarehouse the data cube is n-dimensional.

Example

2-D View of Sales Data for AllElectronics According to time & item

location = "Vancouver"

time (quater)	item (type) home entertainment	computer	phone	security
Q_1	605	825	14	400
Q_2	680	952	31	512
Q_3	812	1023	30	502
Q_4	927	1038	38	580

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

location = "Chicago"

time	item home ent.	comp	ph	sec
Q_1	854	882	89	623
Q_2	943	890	64	698
Q_3	1032	924	59	789
Q_4	1129	992	63	870

location = "New York"

time	item home ent	comp	ph	sec
Q_1	1087	968	38	872
Q_2	1130	1024	41	925
Q_3	1002	940	795	58
Q_4	984	978	864	59

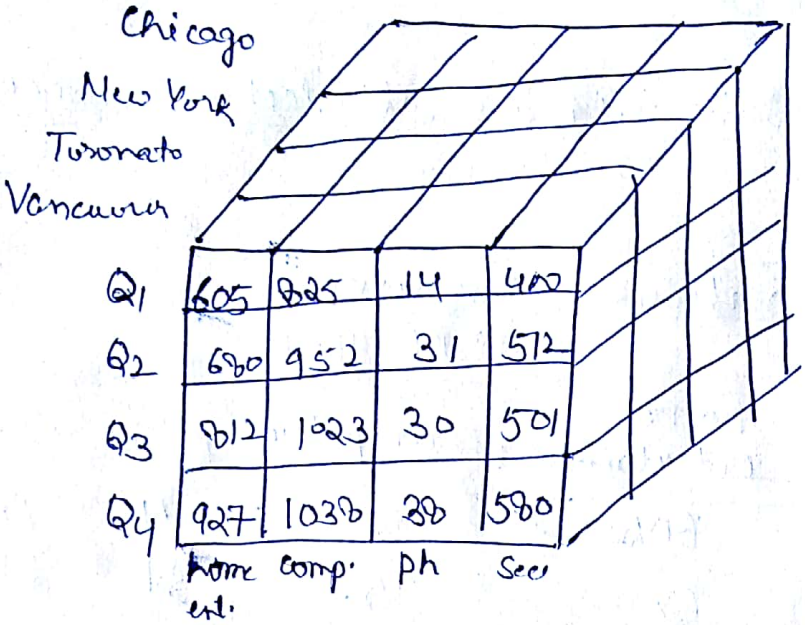
location = "Toronto"

~~time~~	item home ent	comp	ph	sec
	818	746	43	591
	894	769	52	682
	940	795	58	728
	978	864	59	784

location = "Vancouver"

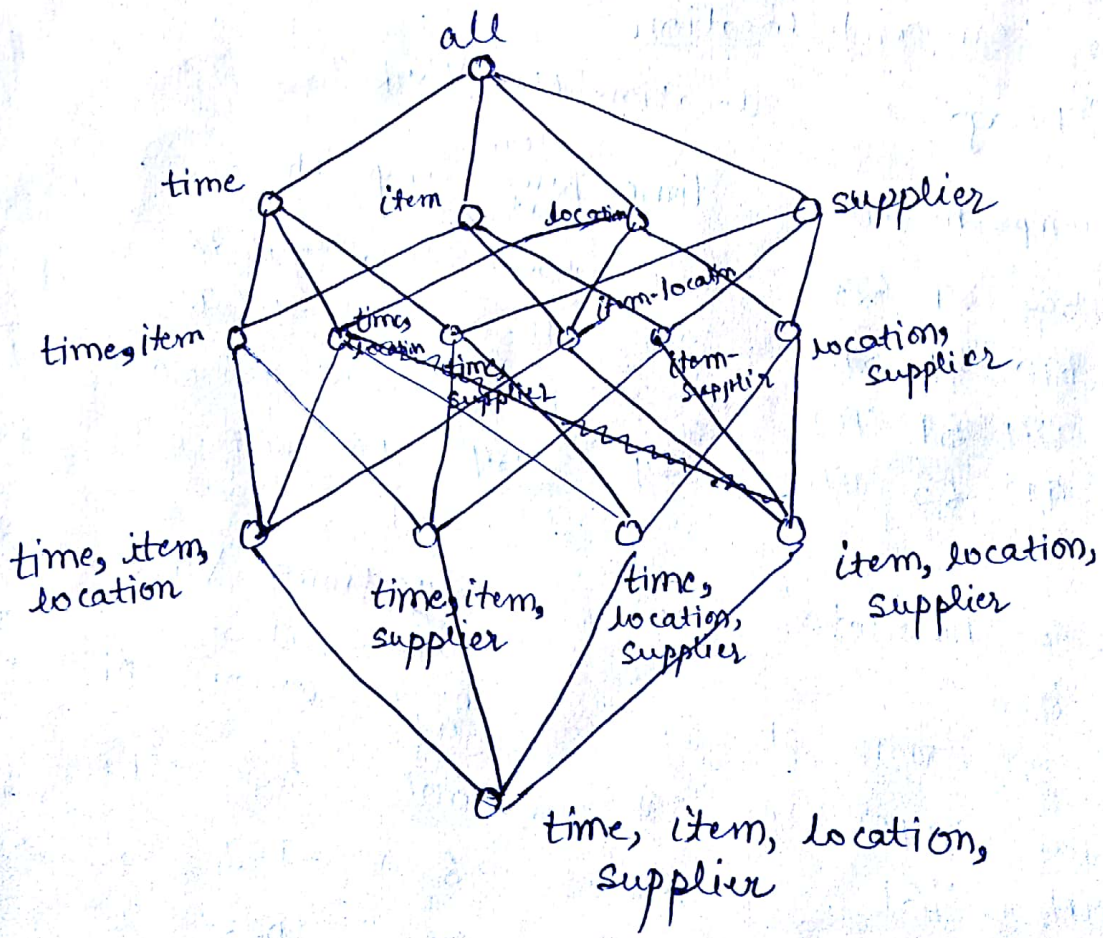
item home ent	comp.	ph	sec
605	825	14	400
680	952	31	512
812	1023	30	501
927	1038	38	580

Scanned by CamScanner



3-D Data Cube representation

4-D data cube representation according to time, item, location & supplier



Scanned by CamScanner

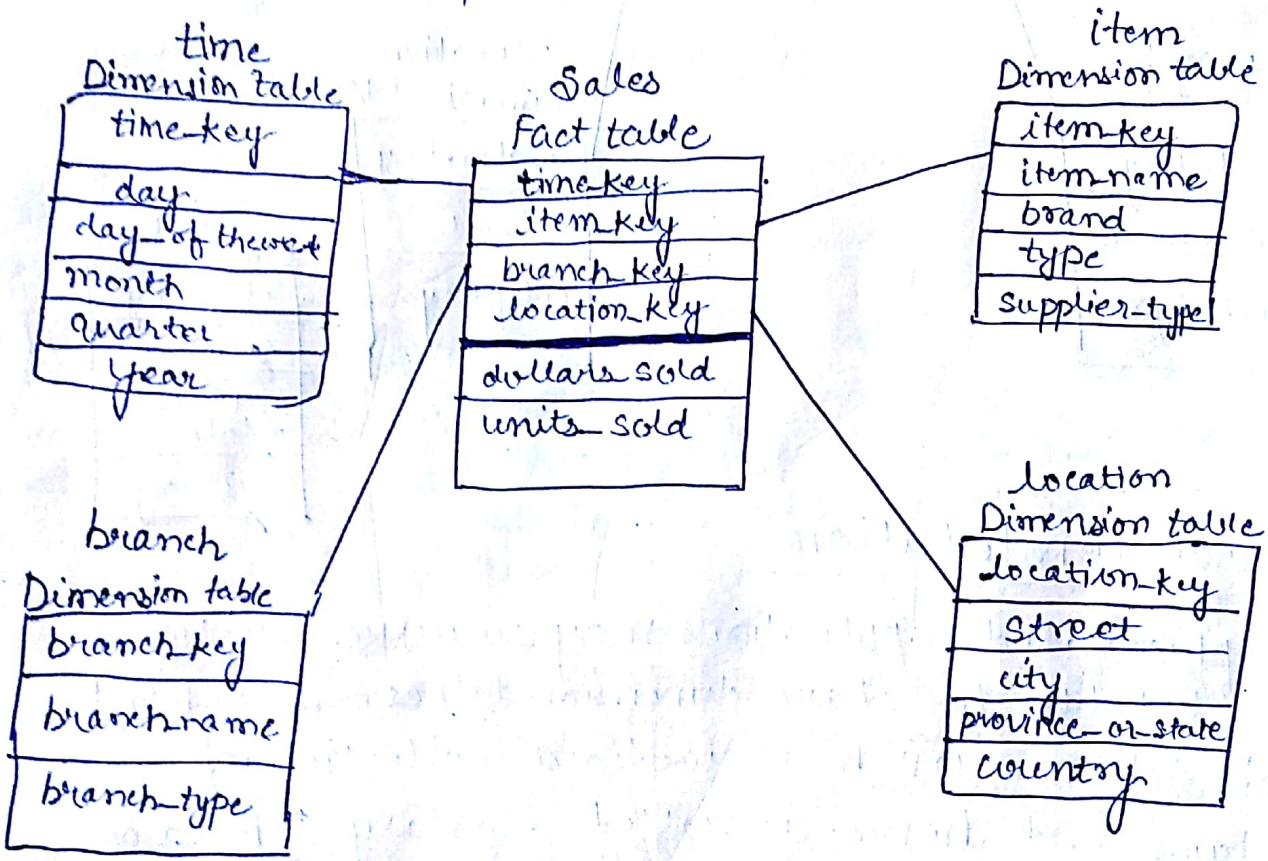
DL Schemas for Multidimensional Data Model

1) Star Schema

↳ In this data warehouse contains
1) a large central table (fact table) containing the bulk of data with no redundancy.
2) Set of smaller attendant tables (dimension tables) one for each dimension.

↳ The schema graph resembles a starburst with the dimension tables displayed in a radial pattern around the central fact table.

Ex→ Star Schema for All Electronics Sales.

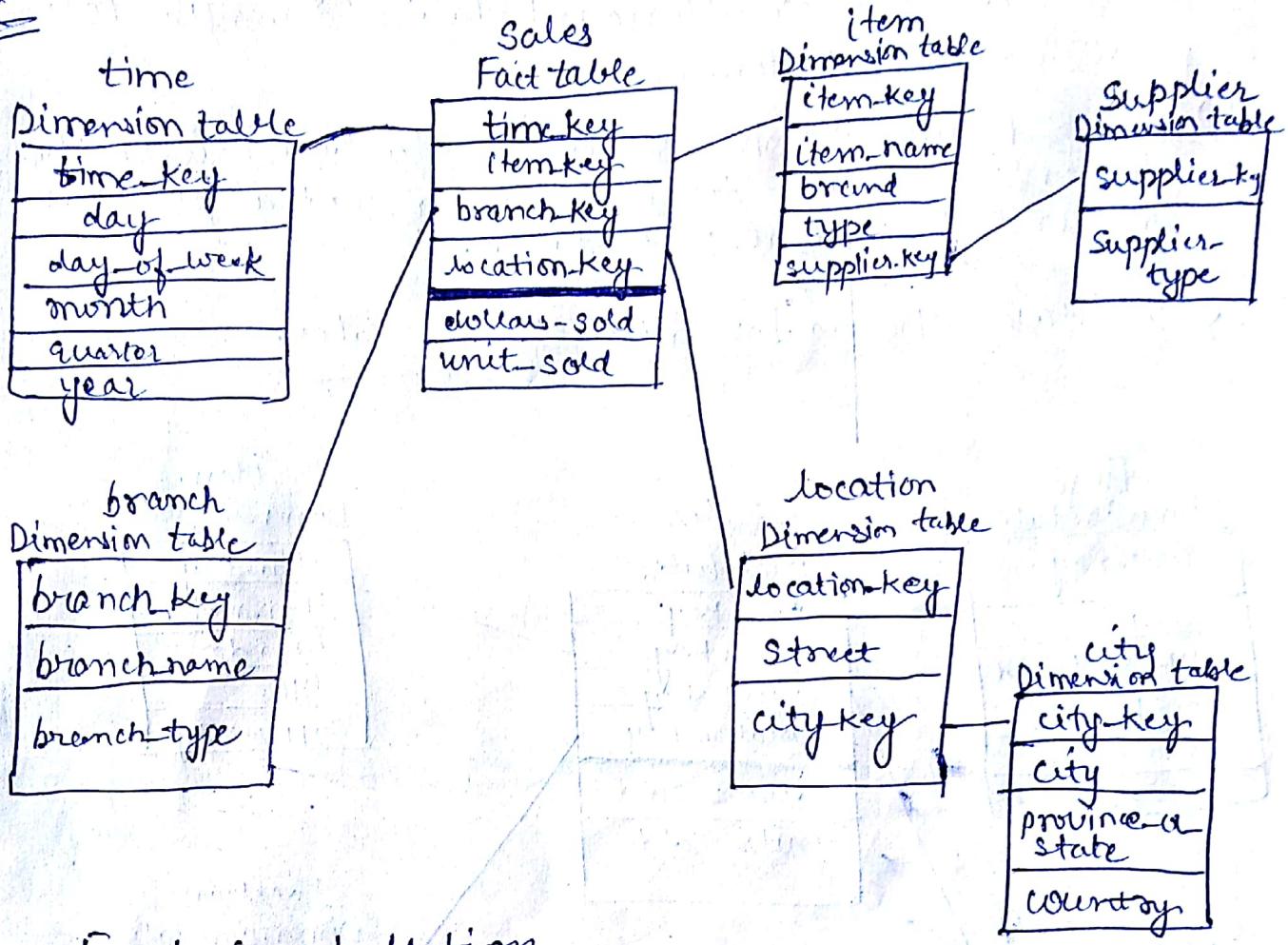


Scanned by CamScanner

2) Snowflake Schema

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

Ex

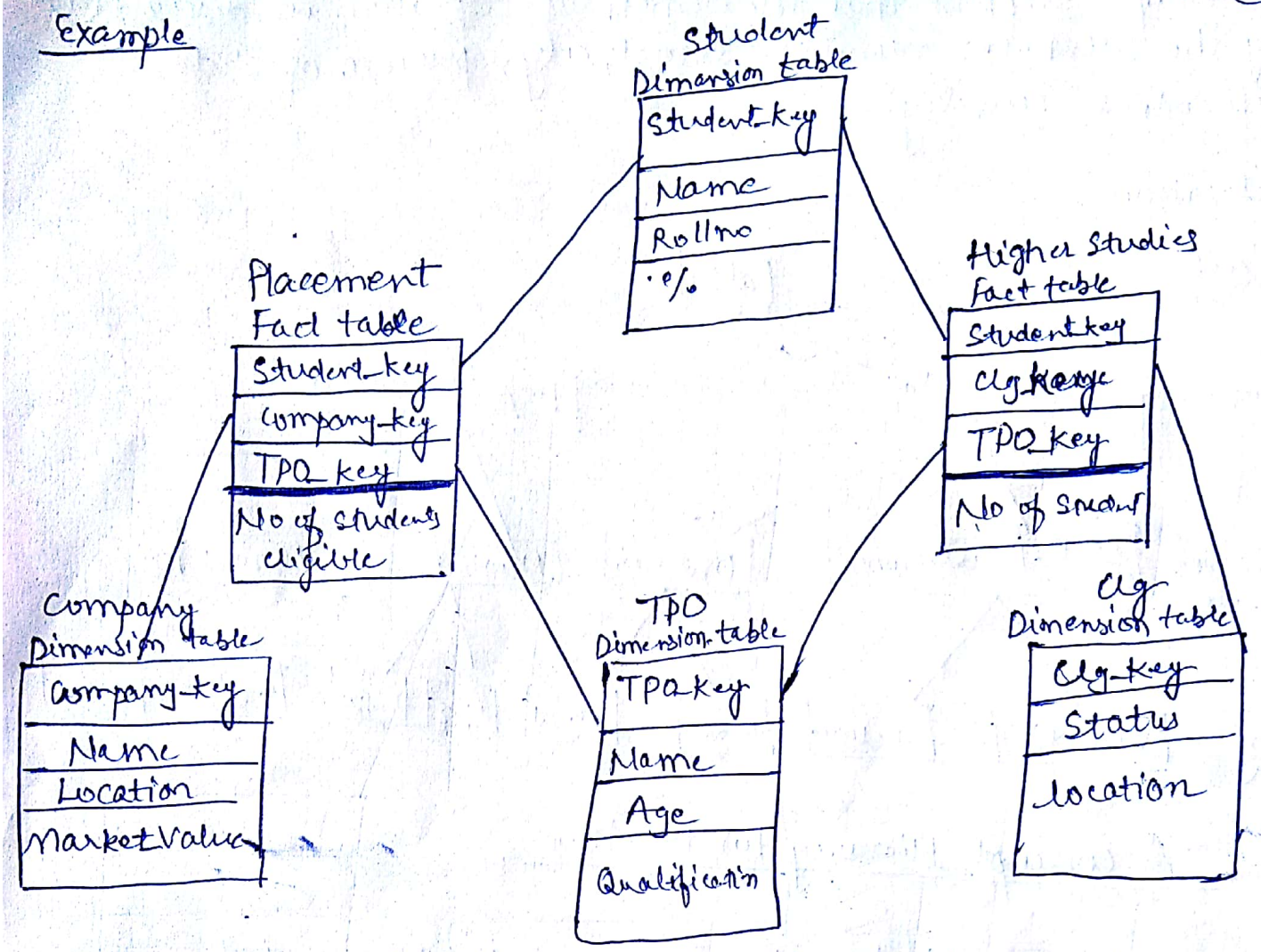


3) Fact Constellation

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

Scanned by CamScanner

Example



Concept Hierarchy

↳ A concept hierarchy defines a sequence of mapping from a set of low level

↳ Consider a concept hierarchy for the dimension location. City values for location include Vancouver, Toronto, New York and Chicago

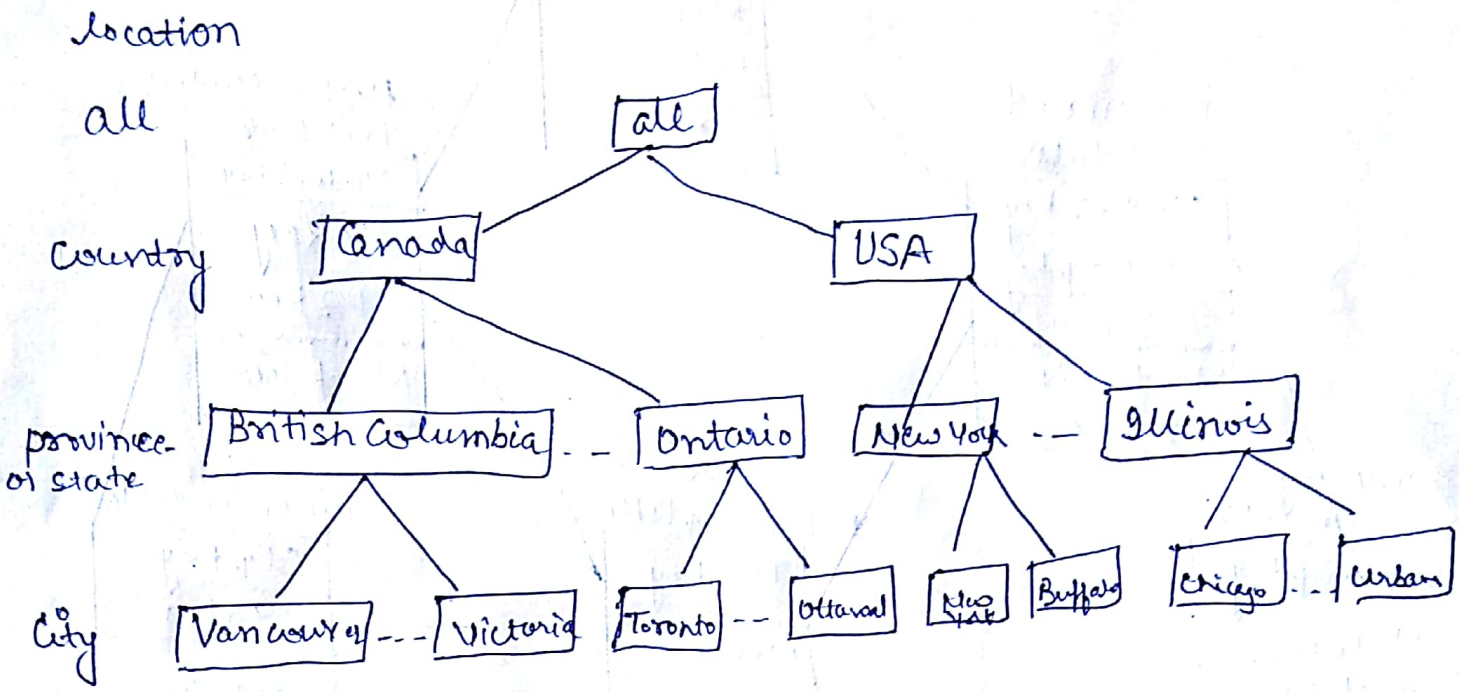
Each city can be mapped to the province or state to which it belongs. For example Vancouver can be mapped to British Columbia and Chicago to Illinois.

The provinces and states can in turn be mapped to the country.

These mappings form a concept hierarchy for dimension location, mapping a set of low-level concepts (i.e. cities) to higher-level, more general concepts (i.e. countries).

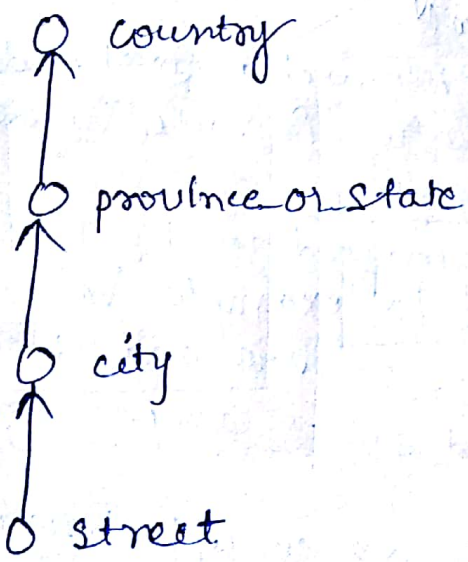
Scanned by CamScanner

For example suppose that the dimension location is described by the attributes number, street, city, province_or_state, zip code & country.



A concept Hierarcy for location

↳ The attributes of location are related by a total order forming a concept hierarchy such as street < city < province_or_state < country.

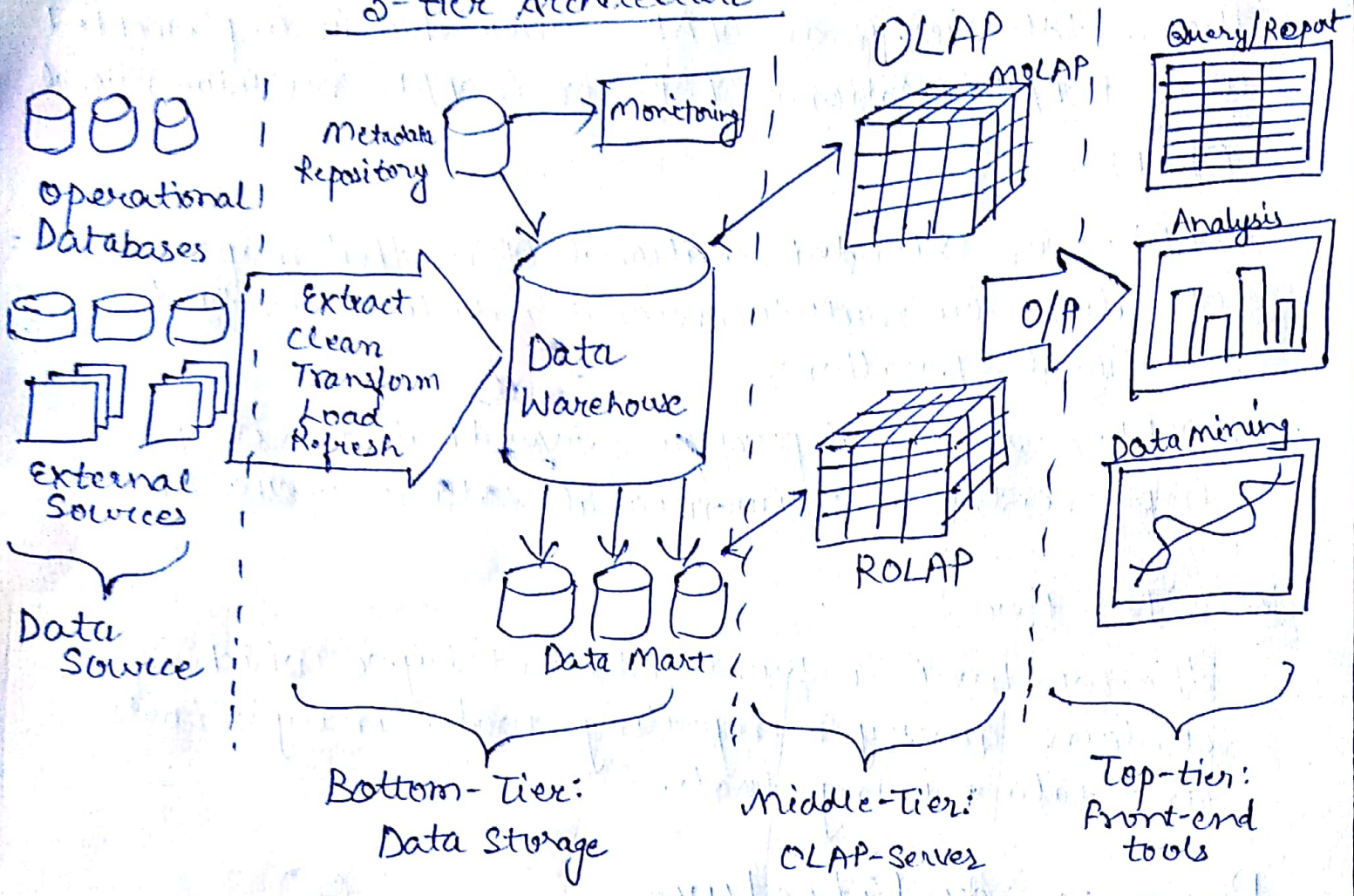


Hierarchical & lattice structures of attributes in warehouse dimensions

Scanned by CamScanner

Process Architecture

3-tier Architecture



Bottom-Tier →

- Bottom-tier is a warehouse database that is a relational database system
- Back-end tools and utilities are used to feed data into the bottom tier from operational databases or other external sources (e.g. customer profile information provided by external consultants).
- These tools perform data extraction, cleaning and transformation (to merge similar data from different sources into a unified format), as well as load & refresh functions to update datawarehouse.
- The data are extracted using Gateway.
- Gateway is supported by DBMS and allows client program to generate SQL code to be executed on Server.
- Example of Gateway is ODBC, JDBC etc.

Scanned by CamScanner

2. Middle Tier

The middle tier is an OLAP server that is implemented using ROLAP (Relational OLAP) or MOLAP (Multidimensional OLAP)

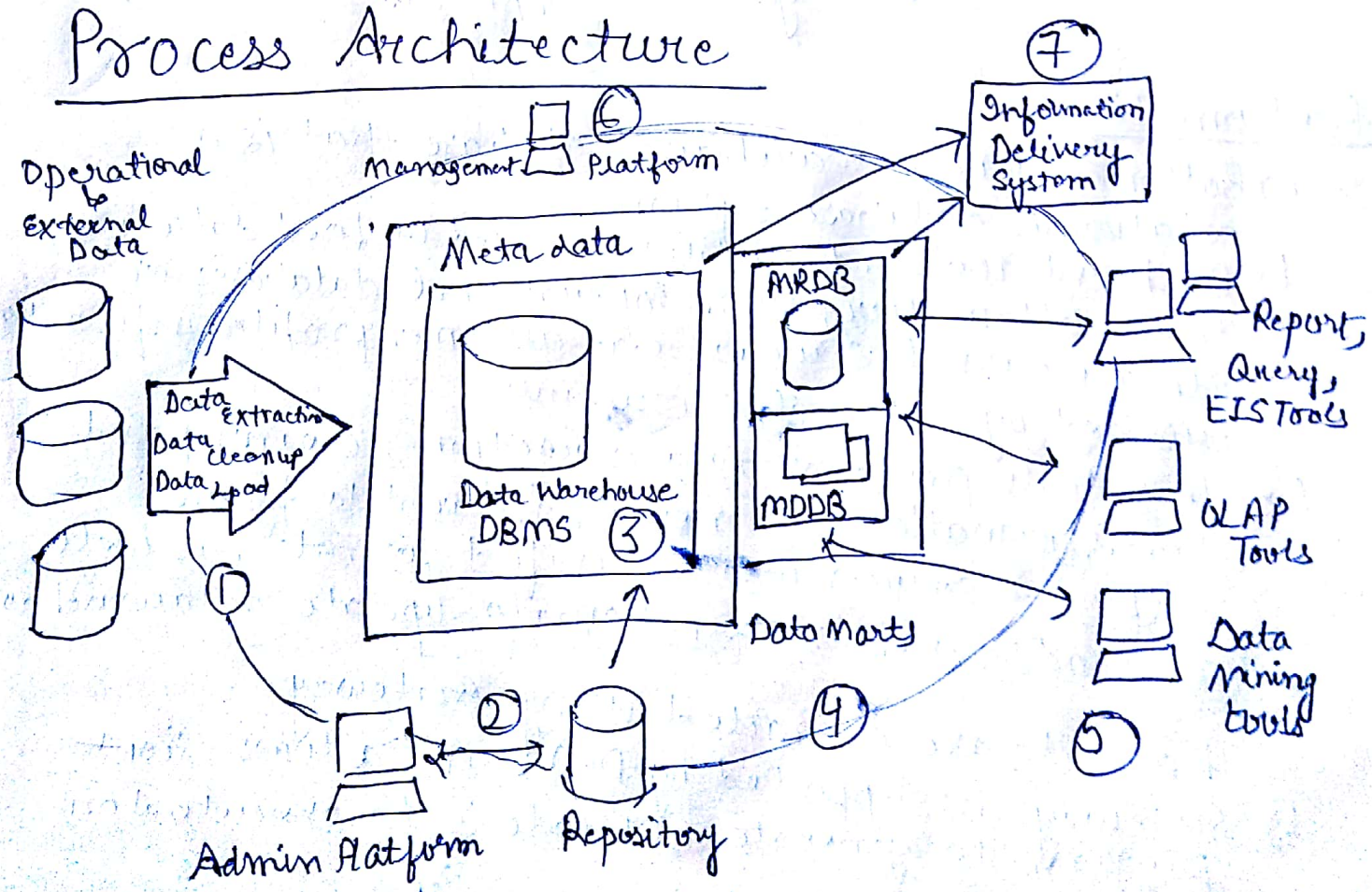
ROLAP → an extended relational DBMS that maps operations on multidimensional data to standard relational operations.

MOLAP → A special purpose server that directly implements multidimensional data & operations.

3. Top-tier

The top-tier is a front-end client layer which contains query & reporting tools, analysis tools and data mining tools.

Process Architecture



Scanned by CamScanner

↳ Data warehouse architecture is based on a relational database management system server that functions as the central repository for informational data

↳ In the data warehouse architecture, operational data & processing is completely seperate from data warehouse processing.

↳ The source data for the warehouse is the operational applications. As the data enters the data warehouse, it is transformed into an integrated structure and format.

↳ The transformation process may involve conversions, summarization, filtering & condensation of data. Because data within the data warehouse contains a large historical component, the data warehouse must be capable of holding and managing large volumes of data as well as different data structures for the same database over time.

EIS → Executive Information Systems

↳ These systems are designed to emphasize with graphical display and easy to use interfaces.

↳ These systems have query analysis, reporting & drill down capabilities.

↳ They are used to monitor business performance.

* About the components already described in the notes

You have to write the same in this topic

Scanned by CamScanner