UNIT – III Data storage and Processing

Data Warehousing:

Organizations are turning to cloud based technologies for data collection, reporting and analysis.

⇓

Data Warehouse → comes as core component of Business Intelligence that enables businesses to enhance their performance.

Data Warehouse → central repository for storing and analyzing information to make better informed decisions

Key characteristics of Data Warehouse:

Subject Oriented:

It provides topic wise information rather than the overall process of a business.

Eg: Analyze your company's sales data, you need to build a data warehouse that concentrate on sales.

Warehouse provide valuable information like

who was your best customer last year?

Who is likely to be your best customer in the coming year?

Integrated:

⇒ DW is developed by integrating data from varied sources into a consistent format.

⇒ Data stored in warehouse is

- Consistent
- Universally acceptable manner in terms of naming, format and coding.

⇒ facilitate effective data analysis

Non Volatile:

⇒ Data once entered in DW ⇒ remains unchanged

⇒ Data ⇒ read only.

⇒ Previous data not erased when new data entered

⇒ helps to analyze what was happened and when.

Time Variant:

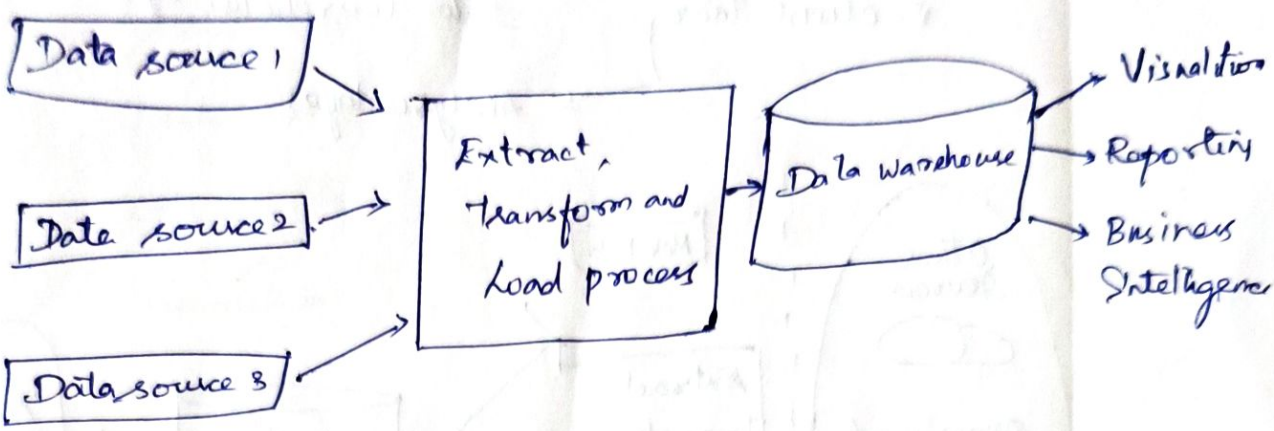
⇒ Data stored in DW is documented with an element of time either explicitly / Implicitly.

⇒ Time variance in DW ⇒ Primary key.

Database Vs Data warehouse:

Data Base	Data Warehouse:
⇒ Supports operational process	Supports analysis and performance reporting.
⇒ Capture and maintain the data.	Explore the data
⇒ current data	Multiple years of history.
⇒ Data is balanced within the scope of this one system	Data must be integrated and balanced from multiple systems
⇒ Data is updated when transaction occurs.	Data is updated on scheduled process
⇒ Data verification occurs when entry done.	Data verification occurs after the fact.
⇒ 100 MB to GB	100 GB to TB.
⇒ ER Based.	Star / snowflake.
⇒ Appln Oriented	Subject Oriented

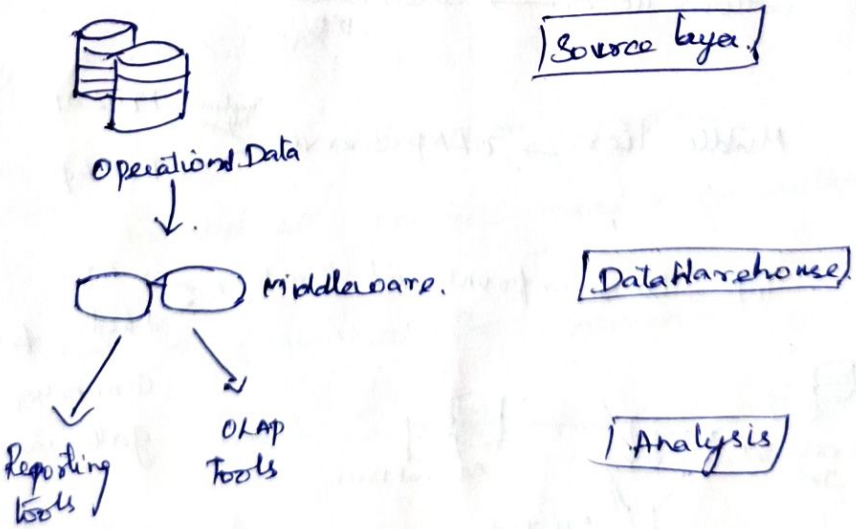
Data Warehouse Architecture



Types of Data Warehouse Architecture.

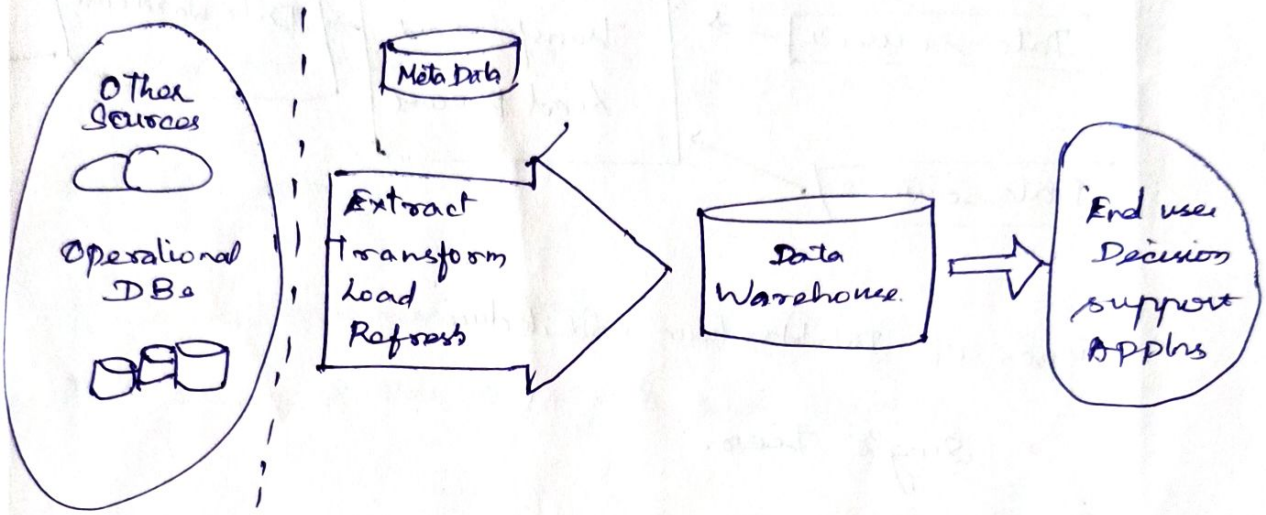
- Single Tier.
- Two-Tier
- Three-Tier

Single Tier. Data Warehouse Architecture



Two tier Architecture

1. Data Tier
   - Source layer.
   - Data staging layer.
   - Data warehouse layer.
2. client Tier
   - Analysis layer.



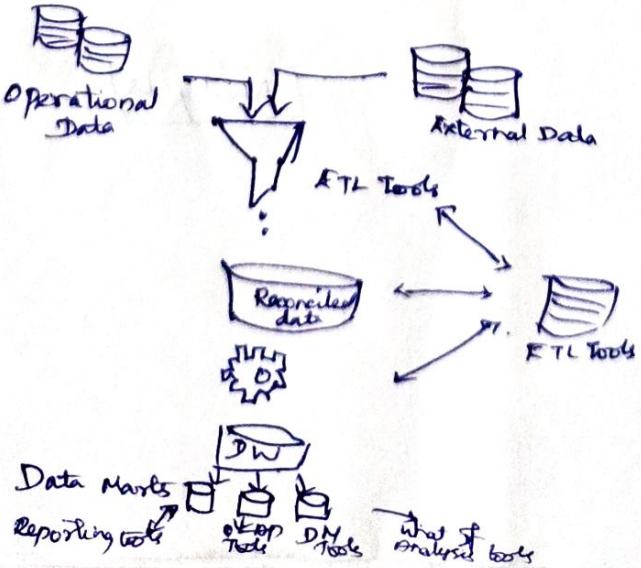
Generic Two Level Architecture

Three tier Data Warehouse Architecture.

Bottom tier → DB → relational DB.

Middle tier → OLAP Server. (Implement by) → MOLAP, ROLAP

Top tier → front end client layer → API
→ tools for. Connecting and gathering data → query tools, Analysis Tool, Datamining tools



Properties of DW Architecture

Security

Administerability

Scalability

Extensibility

Separation

OLAP Vs OLTP:

OLTP → Online Transaction Processing.

Obj ⇒ Processing of data.

⇒ administers the day to day transaction of data under

a 3-Tier Arch. (3NF)

⇒ Each of these transaction involves individual records made up of multiple fields.

⇒ Main ⇒ fast querying processing

Data Integrity in multi access environment

Eg: credit card activity, order entry, ATM transaction.

OLTP Benefits:

* Solves and maintains the challenge of daily transa
  Mgmt.
* Simplifies individual procedure and complex duties
* fast transaction.

OLTP challenges.

+ Transaction are severely affected if OLTP fails
* Enables several users to view and modify the same data simultaneously → results unusual and confusing situatio

OLTP Tools:

→ For transactions → OLTP uses client/server processing to perform mutiple txs.

OLAP → Online Analytical processing

Obj → Analysis of Data for Business decisions

Data Analyst → can get insights into the information on multiple DB and analyze them at a time.

Main emphasis → response time to complex queries.

Eg:

Financial reporting
Trend analysis
Budgeting,
Sales forecasting
other types of planning

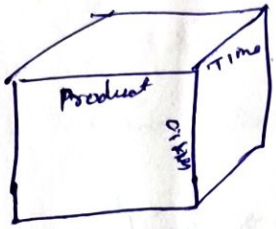
OLAP DB ⇒ multidimensional Data model.

Features ⇒ relational

Navigational

Hierarchical DB.

OLAP cube consists ⇒ multiple types of data.



Benefits:

* Consistency of information
* Security restrictions can be easily applied to diff users.
* adv ⇒ single platform for all corporate analytics needs

OLAP Tools:

1. MOLAP → Multidimensional OLAP → req. Pre computation of data
2. ROLAP → Relational OLAP → doesn't req " " "
3. HOLAP → Hybrid OLAP → decide whether to store data in MOLAP or ROLAP.

MOLAP TOOLS ⇒ IBM cognos, SAS OLAP server, Msft Analysis Services.

ROLAP " ⇒ SAP Netweaver BW, Jedox OLAP server.
Microstrategy Intelligence server.

HOLAP " ⇒ Mondrian OLAP server, Essbase,
SAS OLAP server.

OLAP & OLTP Differences

Data warehousing Tools:

* Amazon Redshift
* Google Bigquery
* Microsoft Azure.
* Postgre SQL.
* Teradata.

Tools like Snowflake, BigQuery and Redshift.

Snowflake.

Why Snowflake?:

* Performance & speed. => multiple virtual warehouse automatic query optimization, cluster Library micro partitions => faster query Process
* User friendly UX → with / without coding - SQL.
* On demand pricing => amt of data used per hour.
* Highly Compatible. → query large dataset, Python, .net, Java, etc.
* Zero Administrative cost => auto scaling, auto suspend. ↳ no hardware / s/w install
* Easy Data sharing. ↳ b/w consumer and providers

Snowflake:

Snowflake is a data warehouse built on top of the cloud infrastructure (AWS, Azure or GCP).

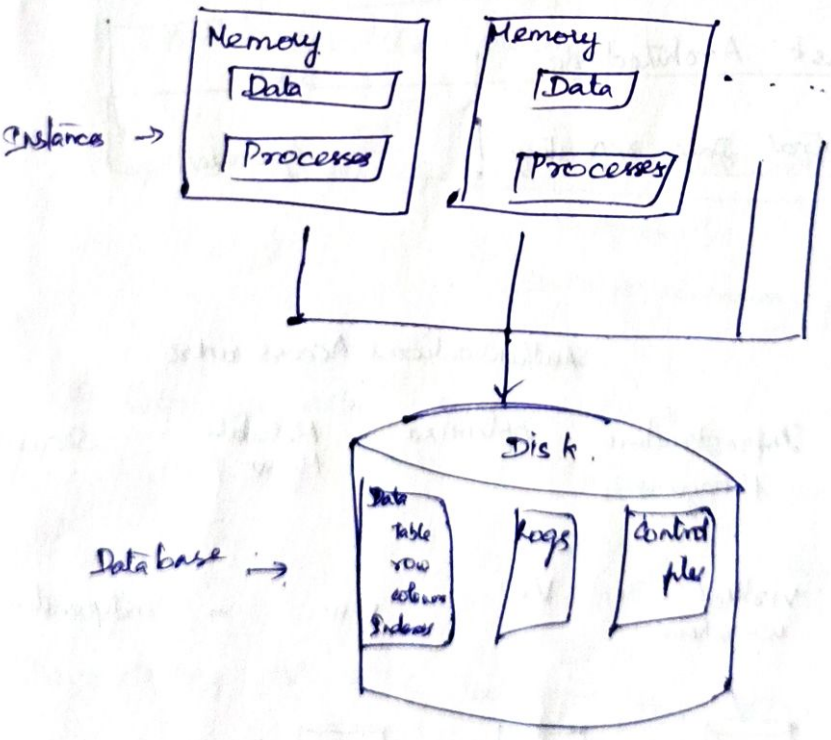
It is a Saas which is ideal for organization that don't want to dedicate resources for setup, maintenance and support of in house servers.

Snowflake → in and for ~~data~~ cloud

Snowflake key features

* Std & Extended SQL support.
* Web base GUI
* Command Line Interface.
* Rich set of client connectors.
  Python node js
  JDBC ODBC.
* Bulk loading & unloading Data.
* Adequate Data protection & security.

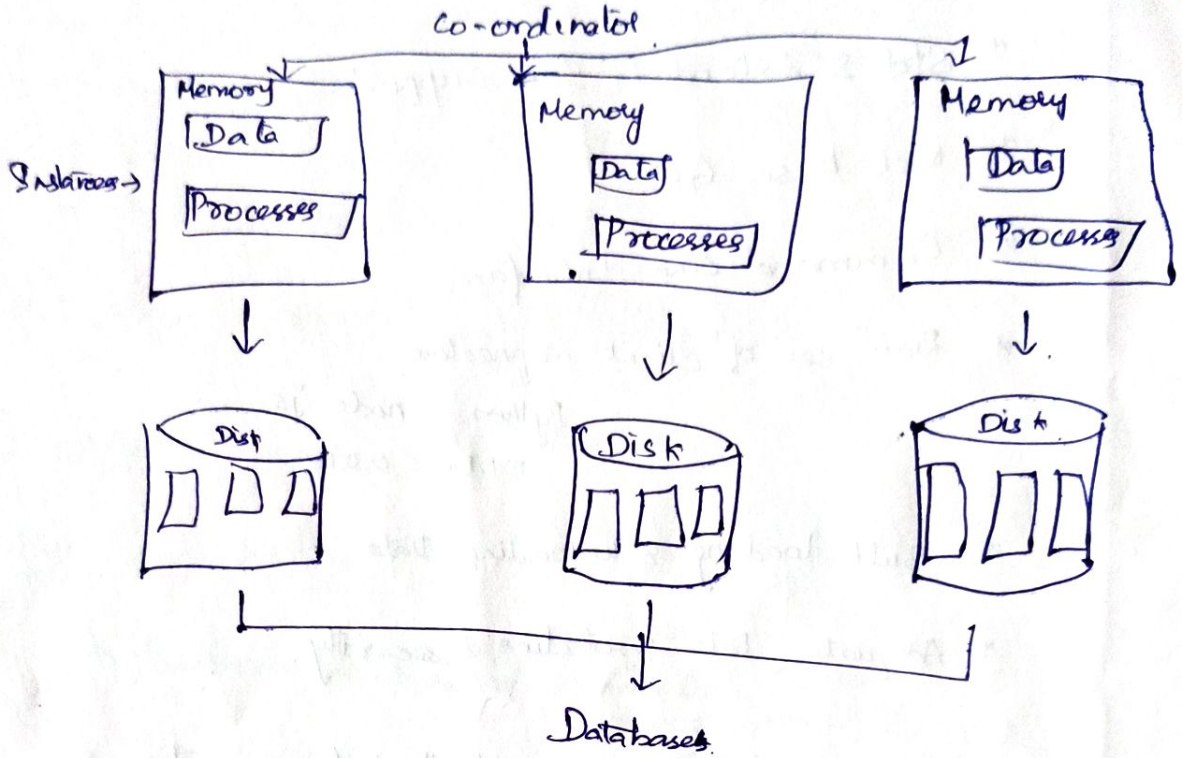
Snowflake Architecture → Distributed computing

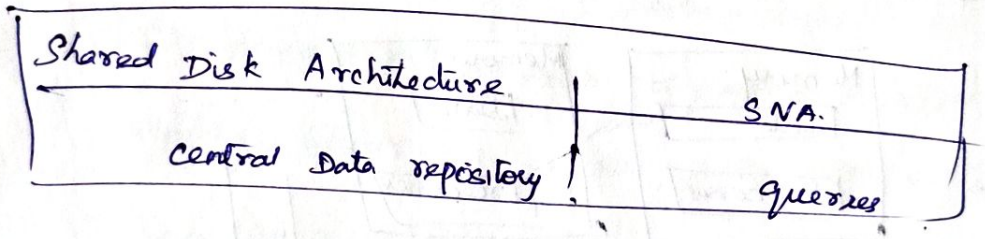


Shared Disk Architecture.

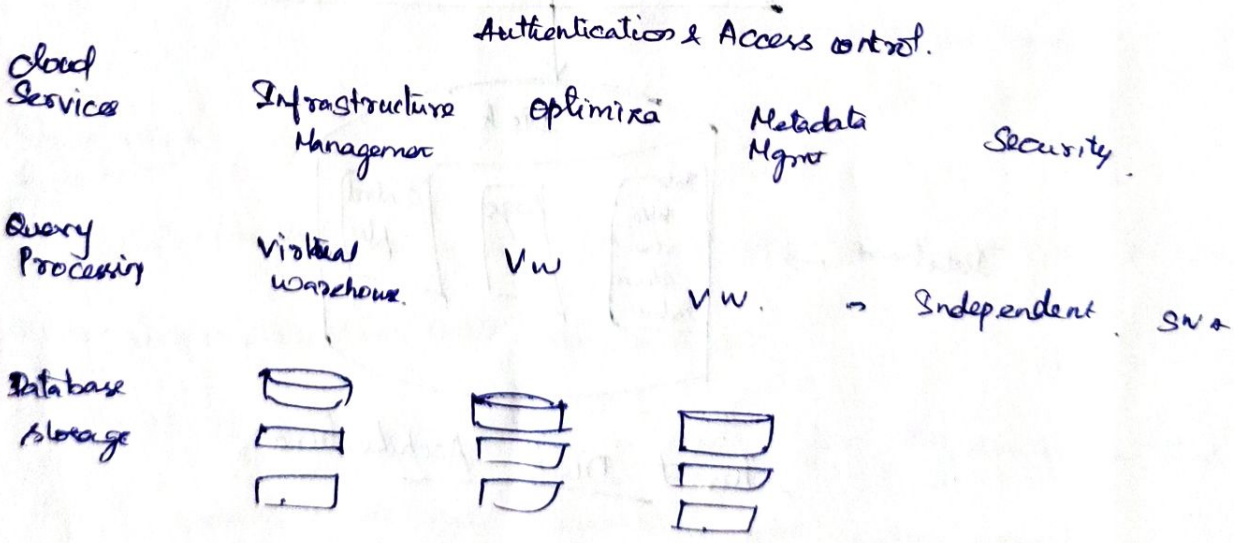
→ robust optimization technique

Shared Nothing Architecture





c.



SNOWFLAKE ARCHITECTURE.

Big Data Basics: Distributed storage using HDFS.

HDFS → Hadoop Distributed File system.

→ designed to store and process big data (structured, unstructured)

→ Core component of Apache Hadoop Ecosystem.
↳ open source framework.

Benefits:

1. Scalable
2. Cost Effective
3. Fast Data access
   Parallel processing
   optimized Data storage.

Excels at providing fault tolerant storage for large dataset
↓ through
Data replication.

Integrates with Apache Spark, Hive, Pig, Flink } ⇒ enabling scalable and efficient data processing

Importance of ETL tools for Big Data processing with HDFS

* streamline the Process of extracting data from various sources, transforming it and loading it into HDFS.

* ETL enables Data cleaning,
  enrichment
  ensure Data consistency and quality.

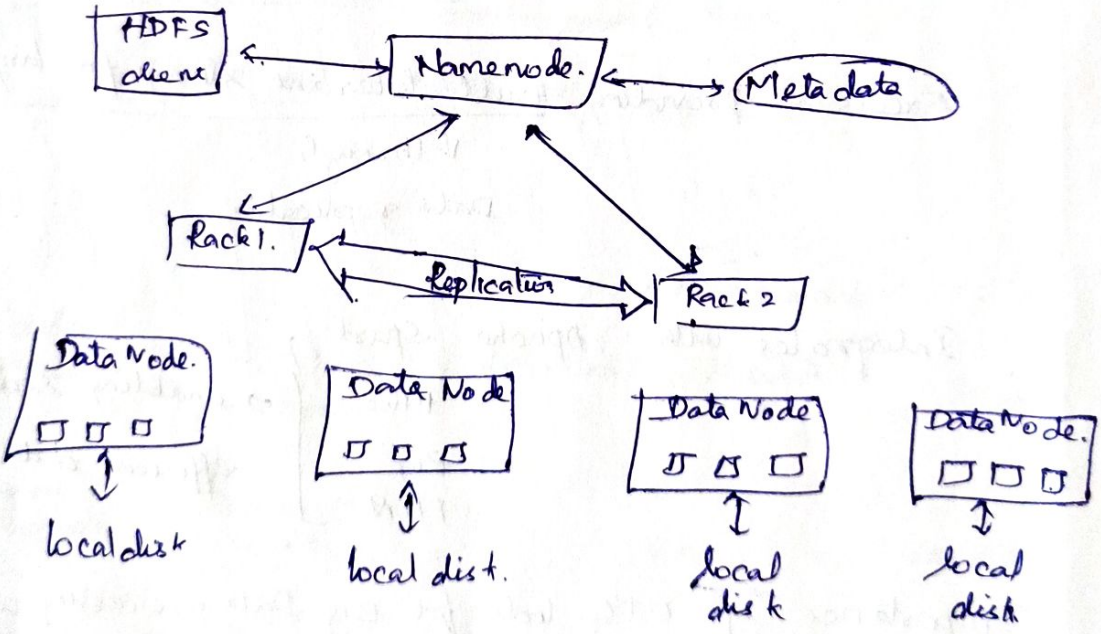
* Provide workflow orchestration capabilities, manage complex data pipelines efficiently.

* offer data governance features, ensuring compliance, security and data lineage tracking.

* Efficiently process and analyze large data.

* empowers informed decision making through insightful data analysis.

HDFS Architecture: → Master slave Architecture



HDFS for Big data processing.

HDFS is essential for.

reliable storage.

Efficient Data processing.

MapReduce Programming Model.

↳ Key component of Apache Hadoop framework.

→ Two stages

↳ Map

↳ Reduce.

MapReduce components.

Map stage:

I/p data is divided into chunks and processed in parallel by multiple map tasks.

Shuffle and sort:

Intermediate key value pairs generated by the map tasks are then sorted and grouped based on their key.

Reduce stage:

Sorted Intermediate key value pairs are processed by multiple reduce task.

Output:

final o/p collected and stored in the HDFS or another designated o/p location.