**Data Base**

A traditional way of storing data into tables, columns, and rows. It helps in data querying and processing easily. It is controlled by DBMS (Data Base Management System).

**Subset:** RDBMS (Relational Data Base Management System)

Companies use DBMS for quick access of the necessary data.

eg,

A) an Airline company wants to figure out their online ticket purchases in a month and process ticket services accordingly.

B) Amazon can access the database to check demands as per market and manage their inventory levels accordingly. Also lead to customer purchases and transactional details.

For transactional integrity the data bases need to have these 4 components:

**1. Atomicity:** Either the transaction takes place at all or not.

**2. Consistency:** Integrity constraints should be maintained for data consistency before and after a transaction.

**3. Isolation:** Multiple transactions can occur concurrently without inconsistency with the database.

**4. Durability:** After completion of a transaction the updates and the modifications of the database are stored securely even though the system failure occurs after.

The Data Base should be **ACID** Compliant.

**Data Mart**

Data Mart is the subset of the Data Warehouse. Business users can access the necessary data from the data warehouse for specific business use cases. It focuses on a specific business area, department, or subject. It contains a smaller volume of data tailored for the needs of a particular user group.

**Types:**

1. **Independent Data Mart:**

The functionalities don’t rely on the data warehouse. It focuses on the one specifics business objective. The data is stored from internal or external data sources.

1. **Dependent Data Mart:**

The data is taken from a data warehouse, a centralized location. When the analytics and reporting are needed, the necessary data is collected.

1. **Hybrid Data Mart:**

Integrates data from external operational sources with an existing data warehouse. This gives higher speed, flexibility and capacity to handle large storage structures.

**Features:**

1. **Subject-Specific**: Concentrates on a single area of a company data, like finance, sales, or marketing.
2. **Simplified**: Smaller and more straightforward compared to a data warehouse, making it quicker to set up and use.
3. **User-Friendly**: Designed to meet the specific needs of end-users, providing quicker access to relevant data.
4. **Cost-Effective**: Requires fewer resources to implement and maintain than a full-scale data warehouse.

**Comparison between Data Mart, Data Warehouse, Data Lake:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Feature** | **Data Mart** | **Data Warehouse** | **Data Lake** |
| **Scope** | Specific business area or department | Entire organization | All data, both structured and unstructured |
| **Size** | Smaller, limited to specific data sets | Large, integrates data from multiple sources | Very large, can store vast amounts of data |
| **Complexity** | Less complex, easier to set up | More complex, integrates diverse data | Less complex, focuses on storage rather than structure |
| **Data Structure** | Structured | Structured | Structured, semi-structured, and unstructured |
| **Usage** | Department-specific analysis and reporting | Organization-wide analytics and reporting | Data exploration, Big Data Analytics, and Machine Learning |
| **Speed of Access** | Faster for specific queries | Optimized for complex queries | May require more processing time for queries |
| **Implementation Time** | Quicker to implement | Takes longer to implement | Can be set up quickly, but requires ongoing management |
| **Cost** | Lower cost due to limited scope | Higher cost due to scale and complexity | Can be cost-effective for storage, but processing and management costs can vary |
| **Example Tools** | AWS Redshift Spectrum, Microsoft Power BI | AWS Redshift, Oracle Exadata | Apache Hadoop, AWS S3, Azure Data Lake |
| **Data Retention** | Historical data relevant to the department | Historical data relevant to the organization | All types of data, historical and real-time |

**Data Leakage**

Types of Datasets:

1. **Train Data.**
2. **Test Data.**

The ML Models are trained based on the Trained Data, but there should not be the overlapping of Train and Test Data. If that occurs, then Data Leakage will occur and leads to failure of the accuracy of the testing Model. If there is Data Leakage, then Model performance hampers the Data Pre-processing Stage in the Feature Engineering Stage.

**Data + ML Algo = Model**

**Causes:**

Common causes include duplicate records and improper handling of training and validation data, which can result in the same information being present in both datasets (train and test data).

**Prevention:**

To avoid data leakage, it's crucial to ensure no overlap between training and validation data and properly preprocess the data to eliminate duplicate or redundant information.