**Data Marts, Data Lakes, ETL, and Data Pipelines**

**Data Warehouse**

A data warehouse works like a multi-purpose storage for different use cases. By the time the data comes into the warehouse, it has already been modeled and structured for a specific purpose, meaning it is analysis ready. As an organization, you would opt for a data warehouse when you have massive amounts of data from your operational systems that needs to be readily available for reporting and analysis.Data warehouses serve as the single source of truth—storing current and historical data that has been cleansed, conformed, and categorized. A data warehouse is a multi-purpose enabler of operational and performance analytics.

A diagram of data warehouse

Description automatically generated

**Data Mart**

A data mart is a sub-section of the data warehouse, built specifically for a particular business

function, purpose, or community of users. The idea is to provide stakeholders data that is most relevant to them, when they need it. For example, the sales or finance teams accessing data for their quarterly reporting and projections. Since a data mart offers analytical capabilities for a restricted area of the data warehouse, it offers isolated security and isolated performance. The most important role of a data mart is business-specific reporting and analytics.

A diagram of a data warehouse

Description automatically generated

**Data Lake**

A Data Lake is a storage repository that can store large amounts of structured, semi-structured,

and unstructured data in their native format, classified and tagged with metadata. So, while a data warehouse stores data processed for a specific need, a data lake is a pool of raw data where each data element is given a unique identifier and is tagged with metatags for further use. You will opt for a data lake if you generate, or have access to, large volumes of data on

an ongoing basis, but don’t want to be restricted to specific or pre-defined use cases. Unlike data warehouses, a data lake would retain all source data, without any exclusions. And the data could include all types of data sources and types. Data lakes are sometimes also used as a staging area of a data warehouse. The most important role of a data lake is in predictive and advanced analytics.

A diagram of data processing

Description automatically generated

**ETL (Extract, Transform, Load)**

Now we come to the process that is at the heart of gaining value from data—the Extract,

Transform, and Load process, or ETL. **ETL is how raw data is converted into analysis-ready data.** It is an automated process in which you gather raw data from identified sources, extract the information that aligns with your reporting and analysis needs, clean, standardize, and transform that data into a format that is usable in the context of your organization; and

load it into a data repository. While ETL is a generic process, the actual job can be very different in usage, utility, and complexity.

1. **Extract**

Extract is the step where data from source locations is collected for transformation. Data extraction could be through:

* Batch processing, meaning source data, is moved in large chunks from the source to the target system at scheduled intervals. Tools for batch processing include Stitch and Blendo.
* Stream processing, which means source data is pulled in real-time from the source and transformed while it is in transit and before it is loaded into the data repository. Tools for stream processing include Apache Samza, Apache Storm, and Apache Kafka.

1. **Transform**

Transform involves the execution of rules and functions that converts raw data into

data that can be used for analysis.

For example, making date formats and units of measurement consistent across all sourced data, removing duplicate data, filtering out data that you do not need, enriching data, for example, splitting full name to first, middle, and last names, establishing key relationships across tables, applying business rules and data validations.

1. **Load**

Load is the step where processed data is transported to a destination system or data repository. It could be:

* Initial loading, that is, populating all the data in the repository,
* Incremental loading, that is, applying ongoing updates and modifications as needed periodically.
* Full refresh, that is, erasing contents of one or more tables and reloading with fresh data.

Load verification, which includes data checks for missing or null values, server performance, and monitoring load failures, are important parts of this process step. It is vital to keep an eye on load failures and ensure the right recovery mechanisms are in place.

A diagram of a data processing process

Description automatically generated

**Data Pipeline**

It’s common to see the terms ETL and data pipelines used interchangeably. And although both move data from source to destination, data pipeline is a broader term that encompasses the entire journey of moving data from one system to another, of which ETL is a subset.

Data pipelines can be architected for batch processing, for streaming data, and a combination

of batch and streaming data. In the case of streaming data, data processing or transformation, happens in a continuous flow.

This is particularly useful for data that needs constant updating, such as data from

a sensor monitoring traffic. A data pipeline is a high performing system that supports both long-running batch queries and smaller interactive queries.

The destination for a data pipeline is typically a data lake, although the data may also be loaded to different target destinations, such as another application or a visualization tool. There are a few data pipeline solutions available, most popular among them being Apache Beam and DataFlow.