Data Engineering — ETL & Pipelines

Title: Fundamentals of Data Engineering: Pipelines and ETL

Data Engineering is the discipline of designing, building, and maintaining the systems and processes that collect, store, and process large volumes of data. It provides the foundation for data science and analytics. A core component of this discipline is the **data pipeline**.

A data pipeline is a series of automated steps that move data from a source system to a destination system. This "destination" is typically a data warehouse or a data lake, where the data can be analyzed. Pipelines ensure that data is reliably and efficiently transported and, if necessary, modified along the way.

The most traditional and well-known type of data pipeline is **ETL**, which stands for **Extract, Transform, and Load**.

1. **Extract:** In this first stage, data is read and pulled from its original source(s). These sources can be diverse, such as relational databases (like MySQL or PostgreSQL), application APIs, text files (CSVs, logs), or streaming data sources (like Apache Kafka).
2. **Transform:** This is often the most complex step. Once extracted, the raw data is cleaned, validated, and converted into a consistent format suitable for analysis. Transformations can include filtering out errors, standardizing date formats, joining data from multiple tables, aggregating values (e.g., calculating daily sales), and masking sensitive information to comply with privacy regulations.
3. **Load:** In the final stage, the newly transformed data is written into the target destination. For decades, this was typically a structured data warehouse (like Teradata, Redshift, or BigQuery) optimized for business intelligence and reporting.

A popular modern variation of this process is **ELT (Extract, Load, Transform)**. With the rise of powerful, scalable cloud data warehouses, it has become feasible to load raw, untransformed data directly into the warehouse first. The "transform" step is then performed *inside* the warehouse using SQL or other tools. This approach allows for greater flexibility, as the raw data is preserved, and different transformations can be applied later as new analytical questions arise.

Key tools used to build and manage these pipelines include workflow orchestration platforms like **Apache Airflow**, data processing engines like **Apache Spark**, and streaming platforms like **Apache Kafka**.