Data Engineering

Jack Thomas

Contents

Chapter 1

Data Engineering

Data engineers work on the processing and storage of data. They develop and maintain systems that ingest raw data and produce high-quality, structured information to support use cases such as analysis and machine learning. These systems are often called data pipelines.

Data pipelines perform **ETL** - Extract, Transform, Load

Extract - Retrieve and collect data from various sources

Transform - Put the data into a suitable format. (Modelling, remove errors, change formats, map same record types to each other, testing and validation)

Load - Send data into a database.

Analysts can access the transformed data using BI tools.

1.1 Data Warehouses

Transactional databases are quick and resilient, but not useful for analysis and complex queries. This is where **data warehouses** come in. A data warehouse consolidates data from multiple sources and is optimised specifically for complex analytics purposes.

Data Warehouse - A central repository that stores data in queryable forms. A relational database optimised for large volumes of data. Typically only contains structured data.

Database vs Data Warehouse

Databases are normalised meaning they have multiple tables (to reduce redundancy). This requires complex queries with mutliple joins. Whereas, data warehouses use few tables and simple queries for improved performance. Also, databases don't typically store historical data whereas data warehouses do. A DW typically

doesn't support as many concurrent users.

On the other hand, a **data lake** stores data raw and unstructured, no pre processing or defined schema. This is typically used in an **ELT** pipeline. The transforming is performed by the data scientists who decide what to do with the data.

An **ELT** pipeline is used typically when as much data as possible has to be ingested. With transformation done later.

Data Flow Challenges

- Data can be corrupted
- Sources may conflict
- Bottlenecks can affect latency
- Duplicate or incorrect data may be generated
- Incompatible data types
- Underestimating data load (Spike in users)

1.2 Big Data

As businesses grow, so does the data they handle. Building scalable systems to deal with large amounts of data is called **big data**.

The 4 V's of Big Data

- Volume - There is a large amount of data
- Variety - Data can be structured or unstructured
- Veracity - Data must be of high quality
- Velocity - Big Data is constantly generated

Most pipelines use **batch processing**. They run periodically and ingest a batch of data that has built up. With big data, new data is generated constantly making batch processing very slow. We can use **streaming processing** to process data in real time.

With multiple data **providers** and multiple data **consumers**, asynchronous communication allows for more efficient flow of information. Here, the data is partitioned into topics and the consumers subscribe to the these topics. Allowing for separate, indepedent flows of information.

Distributed computing is used to store and process large datasets efficiently. Computers are grouped into **clusters** which all perform tasks on their own partition of the data.

Data Migration - Moving data between systems or environments

Data Wrangling - Converting raw data into a useful format for analytics

Data Integration - Collating data from multiple sources

Chapter 2

Data Modelling

Relational database - A database that stores data in tables of rows and columns.

Each table in a RD represents a aspecific object like a 'customers' table for example. They're called relational because entities (tables) can have relationships with others, connecting via a common field (column).

Advantages of Relational Databases

- Flexible - easy to make changes without changing overall structure
- Built-in security
- Easy to run complex SQL queries for analysis - joins and aggregates etc
- ACID compliant

ACID transactions - Atomicity, Consistency, Isolation, Durability

Atomicity - all queries in a transaction must succeed for the transaction to succeed. If one query fails, the entire transaction fails.

Consistency - Referenital integrity. Data kept consistent throughout the tables through the use of foreign keys.

Isolation - Transactions running simultaneously do not interfere with each other.

Durability - Guarantee that any changes made by a committed transaction are not lost.

1st Normal Form

- Atomic Values: Each cell contains a single value.
- There must be a primary key to identify rows.
- No duplicate rows or columns.

2nd Normal Form

- Must be in 1st normal form
- Each non-key field must be fully dependent on the primary key

3rd Normal Form

- Must be in 2nd normal form
- Each non-key field must not be dependent on another.