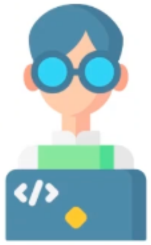
Chapter 1 - DATABASES

Modern Data Project Team

Data Architect
(Design & Architect)



Data Engineer
(Ingestion, Data Pipeline, ETL/ELT)

Analytics Engineer
(Data Modelling, Transformation)

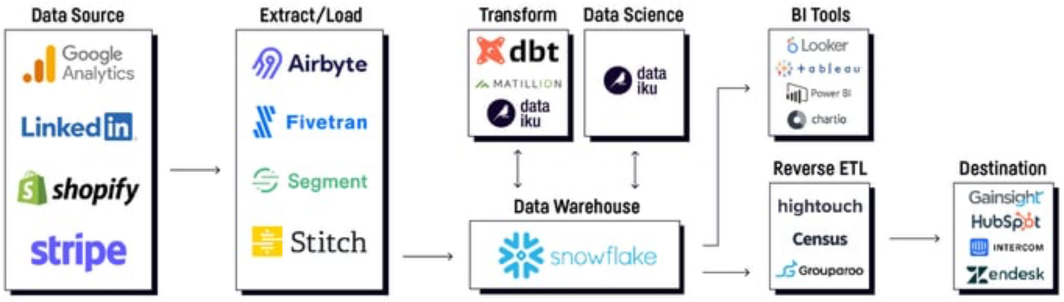
BI Analyst / Data Analyst
(Business Logic)

Project Manager
(Project Governance, Stakeholder Management)

Analytics Engineer Skills

1	SQL Mastery	Data Transformation	Data Warehouse	Data Modelling
2	Data Orchestration	Business Intelligence Tools	Version Control	Communication
3	Programming	CI / CD	Data Engineering Principles	Security & Governance

The Modern Data Stack in the AI Era

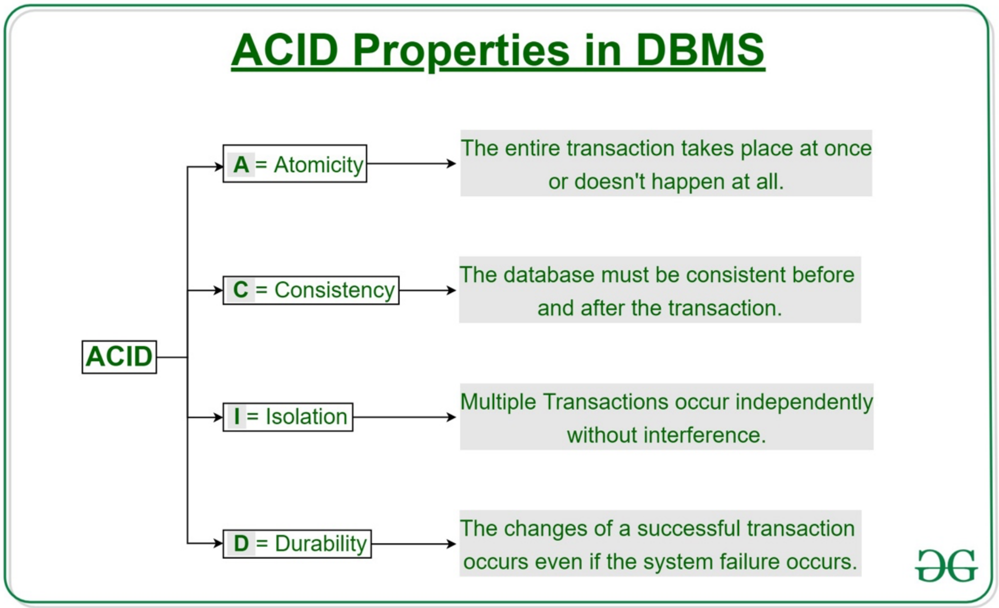


What is a Database?

A. SQL Databases

I. OLTP: Relational Database

- Used for transaction focused tasks, retail applications, CRM
- Row based storage
- Data is structured
- Optimized for insert and update operations
- Required to be available 24/7
- Straightforward queries which return small number of rows
- Highly normalized with many tables
- Minimize data redundancies (no duplicate data)
- Optimized for data collection not for aggregations
- Should not be used for reporting
- OLTP systems are ACID compliant
  - Strong consistency ensuring integrity of the transactions
  - One transaction needs to be successful before another one begins



ACID Properties in DBMS

ACID

A = Atomicity: The entire transaction takes place at once or doesn't happen at all.

C = Consistency: The database must be consistent before and after the transaction.

I = Isolation: Multiple Transactions occur independently without interference.

D = Durability: The changes of a successful transaction occurs even if the system failure occurs.

II. OLAP: Online Analytical Processing

- Efficiently process big data
- Answer analytical queries
- Building blocks of Business Intelligence tools
- Columnar based
- Only needs to read in relevant data
- Data derived from OLTP databases (plus third-party sources)
- Copy of transaction data
- De-normalized with fewer tables (Facts & Dimensions)
- Queries usually have less joins to increase performance and speed
- Insert & Update speed is less important
- Complex aggregations

OLTP vs OLAP Summary (Recreate table)

B. NoSQL Databases

I. "Not Only" SQL

- Non-relational database
- Able to handle different types of data other than RDBMS
- Designed to handle large volume of distributed data
- Suitable for use-cases where fast horizontal scaling is important
- Appropriate for unstructured and semi-structured data
- Usually has simpler schema
- Goal is NoSQL not to replace SQL but to work together
- Many types of NoSQL databases exist for different use cases

I. Key Value Stores

- Simple, only stores key-value pairs
- Retrieves values by associated keys
- Suitable when speed is of most important
- Data is not complex

- Use cases
    - Shopping cart
    - Storing user sessions
    - Game session management
    - API reply stored in cache
    - Product recommendation

II. Document Stores

- Non-relational database designed to store and query JSON-like documents
- Stores each record and data within a single document
- No requirement to create a schema before you load data
- Can scale horizontally very well via sharding
- Common: JSON documents
- Use cases
    - Catalogs
    - Web applications / Ecommerce
    - IoT
    - Realtime Analytics

III. Wide Columns

- Stores data in flexible columns instead of rows
- Highly scalable and able to handle ambiguous and complex data types
- Names and format of the columns can vary across rows in same table
- Not optimized for joins should not be used for:
    - If database requirement changes frequently
    - Ad-hoc query patterns
    - High level of aggregation
- Use cases
    - Real time data / Analytics
    - Time Series
    - Trading data
    - IoT

IV. Graph Databases

- Purpose built database to store and navigate relationships
- Relationships are first-class citizens and it is stored alongside the data in the model
- Data entities are stored in nodes, relationships are stored in edges, information associated to nodes are properties
- Queries are very fast due to relationships not being calculated during query time instead it is stored in the database
- Use cases
  - Recommendation Engines
  - Fraud Detection
  - Social Networks
  - Logistics
  - Metadata Management
  - Natural Language Processing

V. Search Engine Databases

- Database dedicated to search of data in form of web search or full-text search
- Data is stored in JSON document form and is schema-less
- Uses indexes to categorize the similar characteristics among data
- Solves searching of textual content in databases by allowing natural language search
- Use cases
  - Full-text search
  - Time Series Data
  - Logging and Analysis
  - Auto Suggestion / Auto Completing