Database vs Data warehouse vs Data lake vs Delta lake

Database:

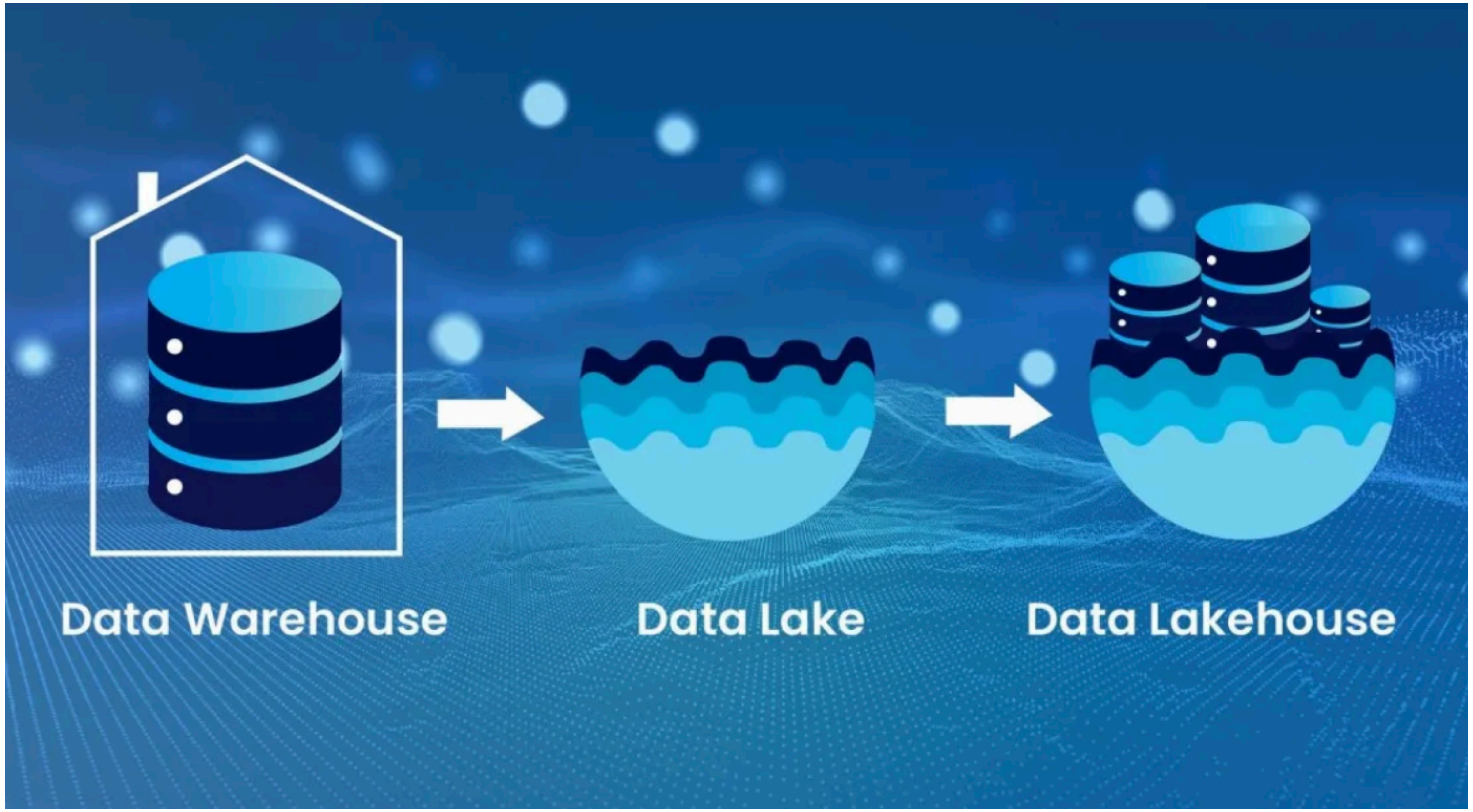
A database is a systematic collection of data or information that is stored electronically. As long as your application needs to store data, you will eventually need a database. Depending on the nature of your data and how they relate to each other, you may choose between a multitude of the following common database families.

- Relational databases: Oracle, MySQL, Microsoft SQL Server, and PostgreSQL
- Document databases: MongoDB and CouchDB
- Key-value databases: Redis and DynamoDB
- Wide-column stores: Cassandra and HBase
- Graph databases: Neo4j and Amazon Neptune

Database characteristics

- Security features to ensure the data can only be accessed by authorized users.
- ACID (Atomicity, Consistency, Isolation, Durability) transactions to ensure data integrity.
- Query languages and APIs to easily interact with the data in the database.
- Indexes to optimize query performance.
- Full-text search.

- Optimizations for mobile devices.
- Flexible deployment topologies to isolate workloads (e.g., analytics workloads) to a specific set of resources.
- On-premises, private cloud, public cloud, hybrid cloud, and/or multi-cloud hosting options.



Data warehouse

A data warehouse is a system that stores highly structured information from various sources. Data warehouses typically store current and historical data from one or more systems. The goal of using a data warehouse is to combine disparate data sources in order to analyze the data, look for insights, and create business intelligence (BI) in the form of reports and dashboards.

You might be wondering, "Is a data warehouse a database?" Yes, a data warehouse is a giant database that is optimized for analytics.

Data warehouse characteristics

Data warehouses store large amounts of current and historical data from various sources. They contain a range of data, from raw ingested data to highly curated, cleansed, filtered, and aggregated data.

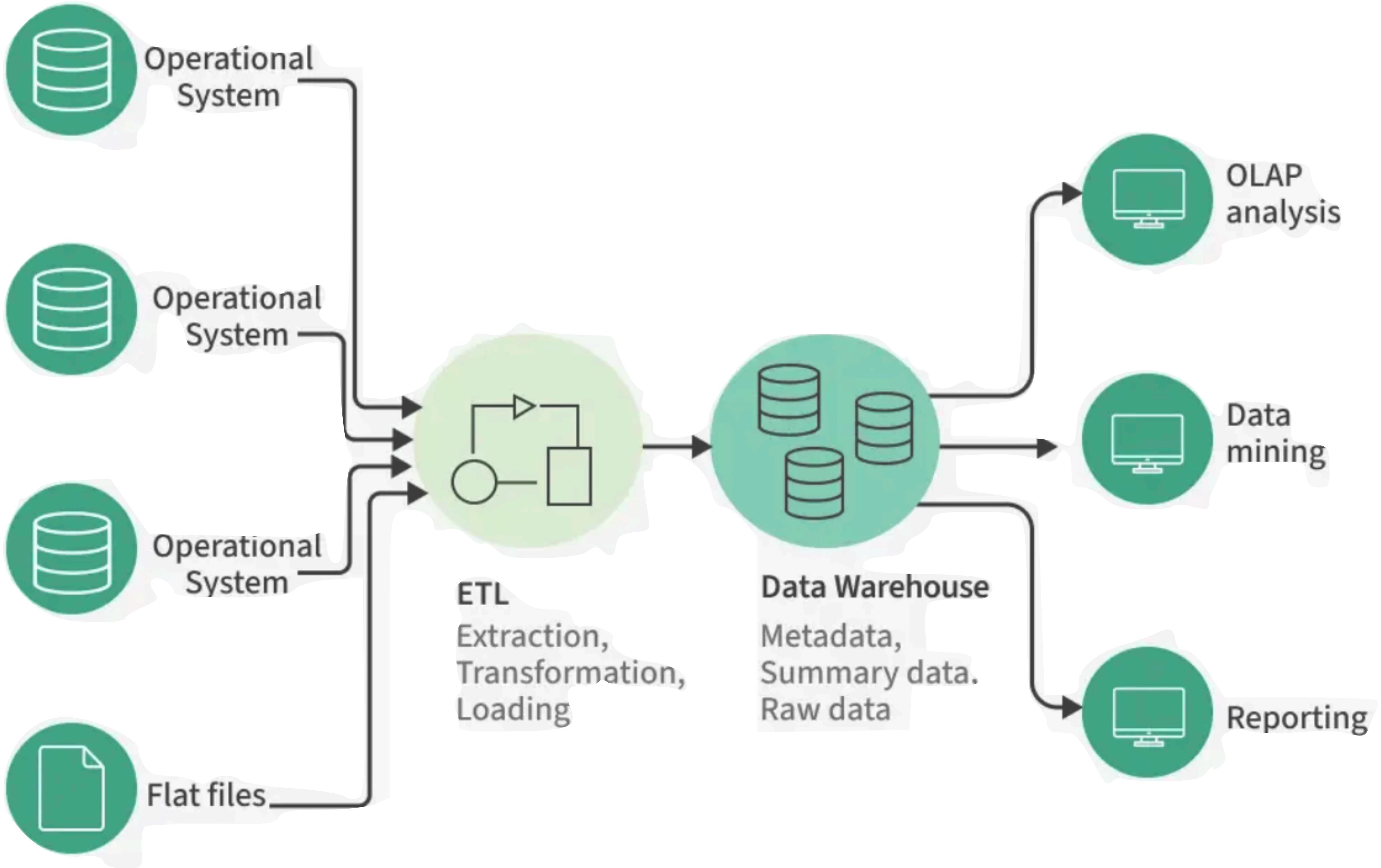
Extract, transform, load (ETL) processes move data from its original source to the data warehouse. The ETL processes move data on a regular schedule (for example, hourly or daily), so data in the data warehouse may not reflect the most up-to-date state of the systems.

Data warehouses typically have a pre-defined and fixed relational schema. Therefore, they work well with structured data. Some data warehouses also support semi-structured data.

Once the data is in the warehouse, business analysts can connect data warehouses with BI tools. These tools allow business analysts and data scientists to explore the data, look for insights, and generate reports for business stakeholders.

Examples of data warehouses include:

- Amazon Redshift.
- Google BigQuery.
- IBM Db2 Warehouse.
- Microsoft Azure Synapse.
- Oracle Autonomous Data Warehouse.
- Snowflake.
- Teradata Vantage.



Data Lake

A data lake is a repository of data from disparate sources that is stored in its original, raw format. Like data warehouses, data lakes store large amounts of current and historical data. What sets data lakes apart is their ability to store data in a variety of formats including JSON, BSON, CSV, TSV, Avro, ORC, and Parquet.

Typically, the primary purpose of a data lake is to analyze the data to gain insights. However, organizations sometimes use data lakes simply for their cheap storage with the idea that the data may be used for analytics in the future.

Data lake characteristics

Data lakes store large amounts of structured, semi-structured, and unstructured data. They can contain everything from relational data to JSON documents to PDFs to audio files.

Data does not need to be transformed in order to be added to the data lake, which means data can be added (or “ingested”) incredibly efficiently without upfront planning.

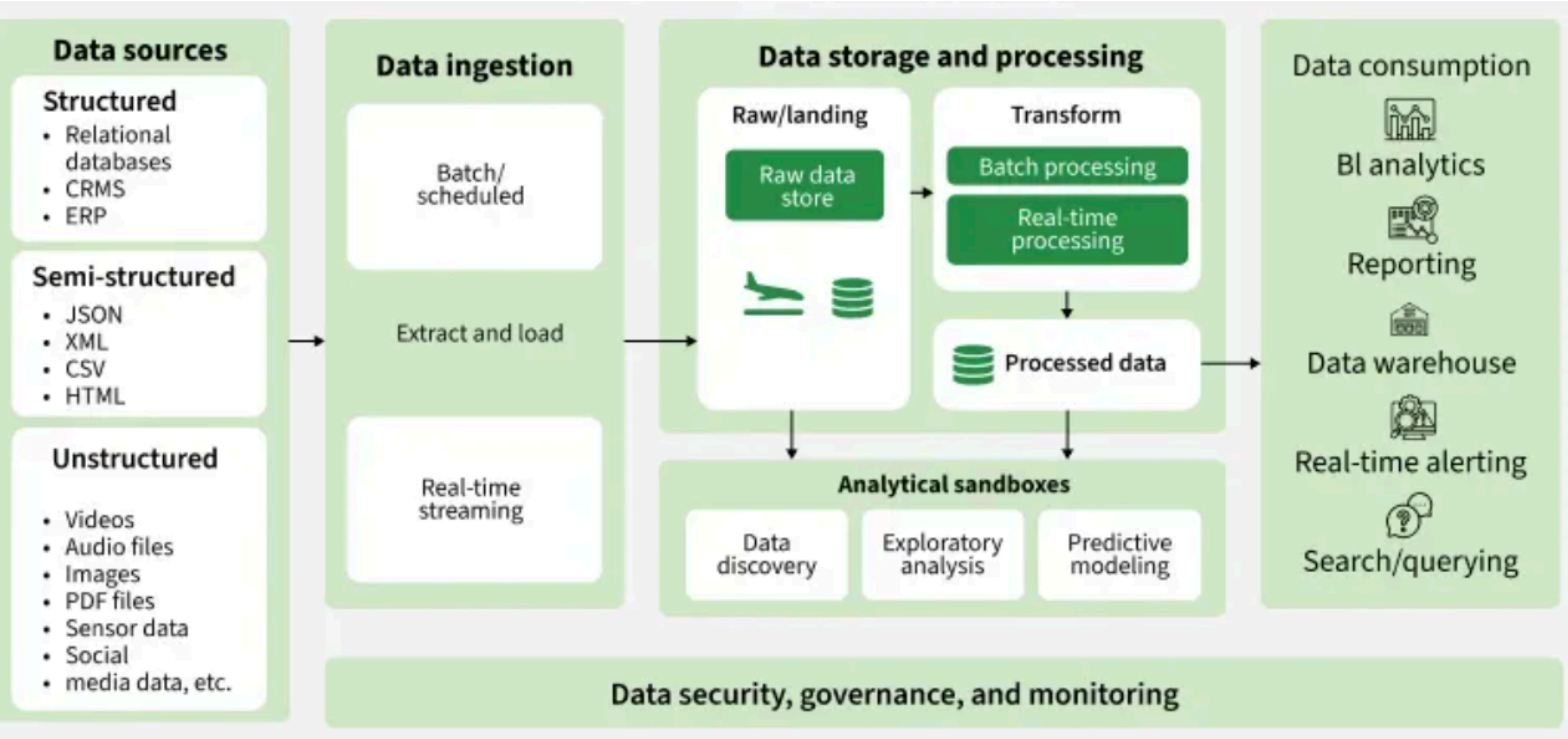
The primary users of a data lake can vary based on the structure of the data. Business analysts will be able to gain insights when the data is more structured. When the data is more unstructured, data analysis will likely require the expertise of developers, data scientists, or data engineers.

The flexible nature of data lakes enables business analysts and data scientists to look for unexpected patterns and insights. The raw nature of the data combined with its volume allows users to solve problems they may not have been aware of when they initially configured the data lake.

Data in data lakes can be processed with a variety of OLAP systems and visualized with BI tools.

The following are examples data lakes:

- AWS S3
- Azure Blob Storage
- Google Cloud Storage



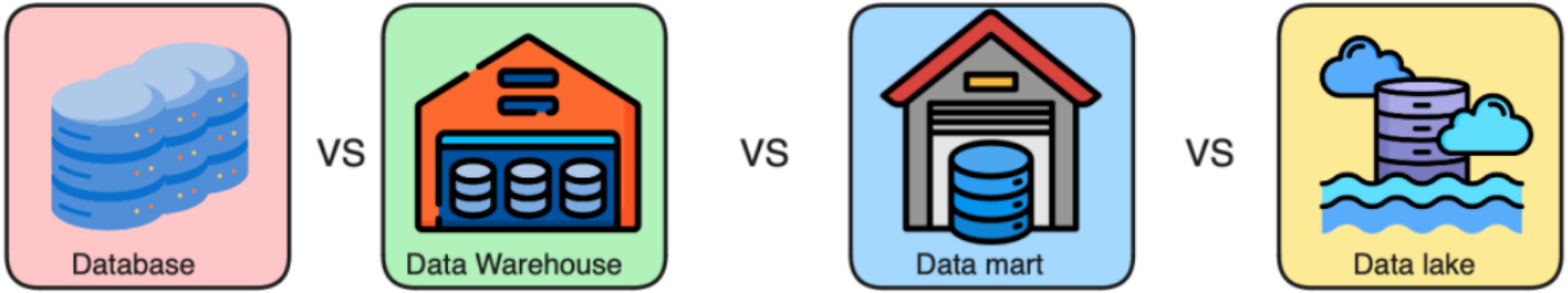
Delta Lake:

Finally, while data lakes are amazing and provide scalability and support for unstructured data (advantages that are missing from the warehouses), the object storage nature of the data lakes makes them lose most of the advantages provided by the warehouses. Despite their capabilities to scale and handle different types of data, Data lakes suffers from the following disadvantages:

- **No Data consistency** — The reads and writes operations are not synchronized nor isolated. This could lead to FileNotFound Errors in a data analytics system, where downstream business analytics processes try to read data from the data lake while it is being written by an ELT process. Data written to a cloud data lake might not be available instantly.
- **No schema enforcement** — Data lakes being object storage, they are not concerned with the structure and schema of data and are happy to store all and any data without performing any checks to make sure that the data is consistent. As a result it is possible to end up with massive amounts of data which are inconsistent or mismatched data types making it hard to be used for analytics.

- **Lack of transactional guarantees** — A typical relational database provides transactional guarantees when data is being written. This means that a database operation either completely succeeds or completely fails, and that any consumer trying to read the data simultaneously doesn’t get any inconsistent or incorrect data because of a database operation failure. Data lakes on the other hand does not provide this guarantees and it is up to the developer to clean up and manually rollback half written incomplete data from any failed jobs

To address these challenges; another layer called delta is implemented on top of the data lake to solve these challenges.
Delta lake is an open source data storage layer implemented on top of the data lake to help bring reliability, ACID transactional guarantees, schema validation and evolution to the cloud-based data lakes.
With delta lake, we get both advantages provided by the legacy data warehouse and the modern data lakes.



Application-specific	Organization-wide, structured data.	Department-specific, structured data.	Organization-wide, any type of data
Structured	Structured	Structured	Structured, semi-structured, unstructured.
Predefined schema	Schema on write	Schema on write (inherited from data warehouse)	Schema on read
Operational applications(OLTP)	Business intelligence, historical analysis(OLAP).	Specific business function analysis	Big data analytics, data exploration.