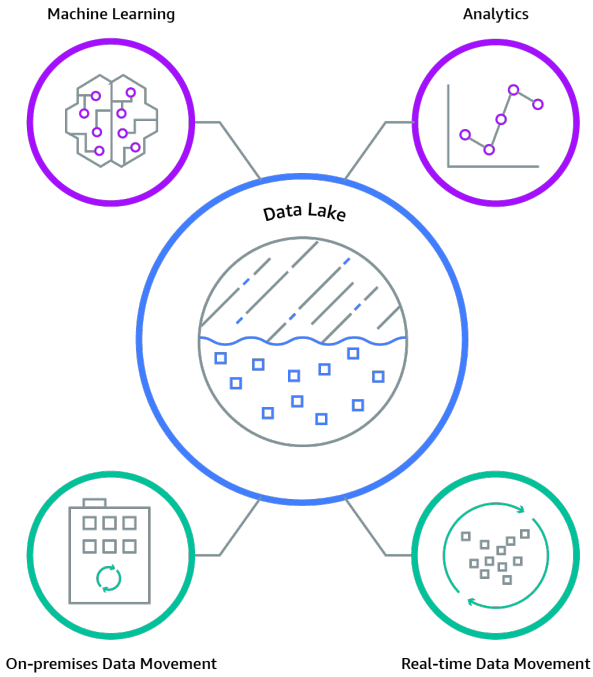
Data Lake:

A data lake is a centralized repository that allows us to store any amount of structured and unstructured data. We can store data without first structuring it, and run various types of analytics, such as dashboards and visualizations, big data processing, real time analytics, and machine learning etc to help make better decisions.



Source: "What is a Data Lake?" by Amazon

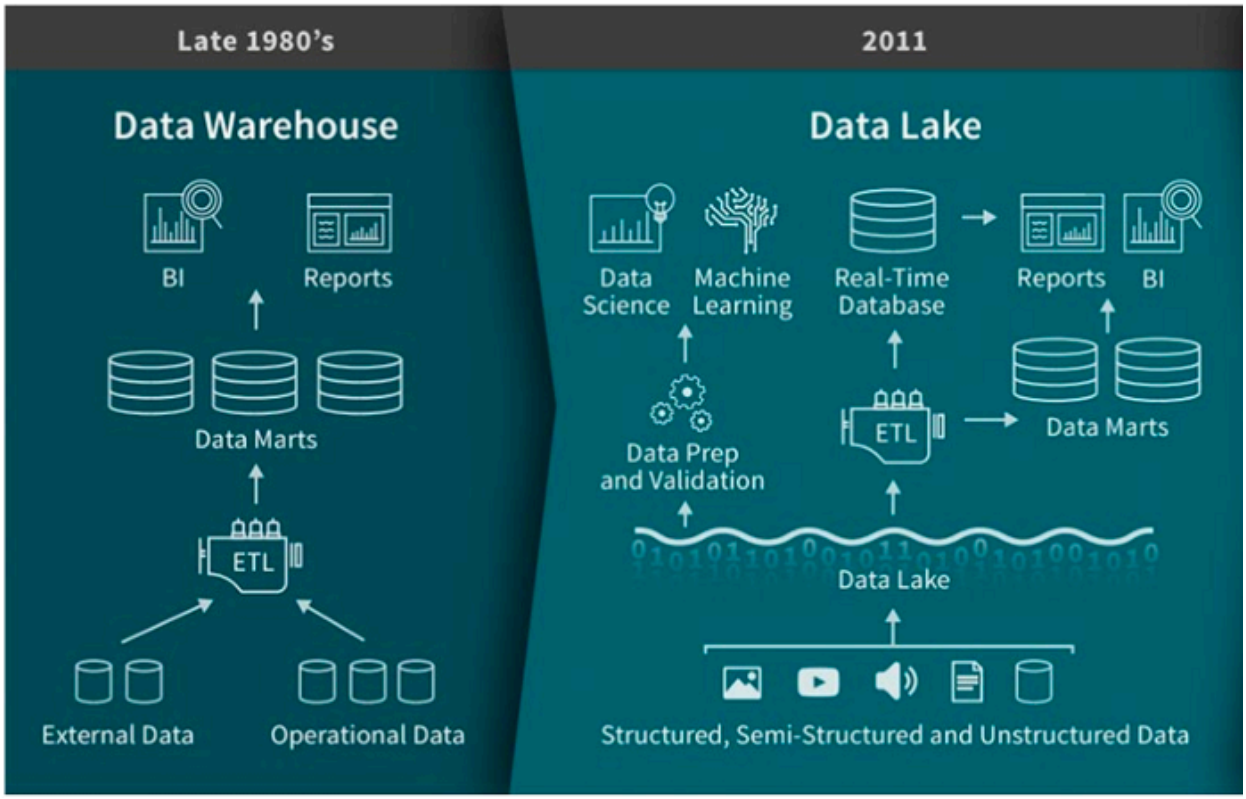
Some of the benefits of a data lake include:

- Eliminates the need for data modeling during data ingestion. We can do this while exploring data for further analysis. As a result, we can filter and model them as needed. However, prior data modeling is required for data warehouse.

- Offers scalability and is relatively inexpensive compared to a traditional data warehouse when we take scalability into account.

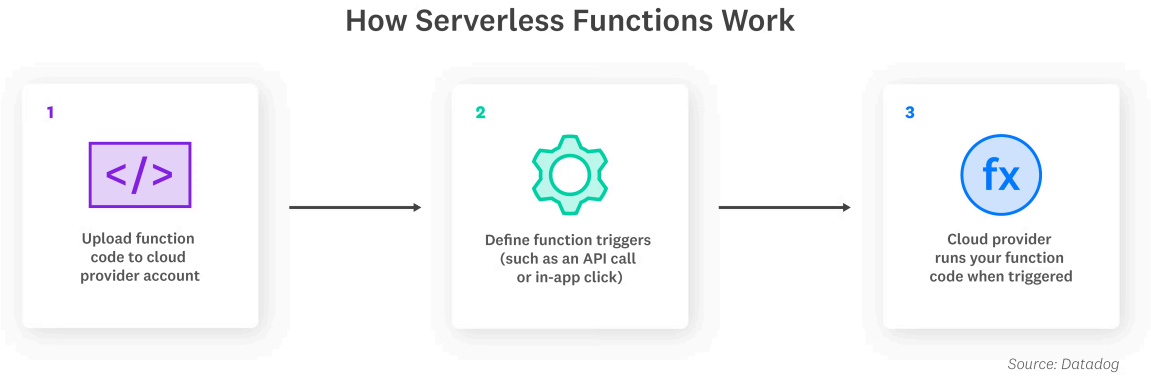
- Capable of storing multi-structured data from various sources. But traditional data warehouse products are schema-based.
- Provides various options and language support for analysis. Traditional data warehouse technology mostly supports SQL, which is appropriate for simple analytics, but we need more ways to analyze data for complex use cases.

Source: "What is a Lakehouse?" by Databricks



Serverless Architecture:

A serverless architecture is a method of developing and deploying applications and services without the need to manage infrastructure. The application still runs on servers, but the cloud provider manages them all. Function as a Service (FaaS) is a popular serverless architecture in which developers write their application code as a set of discrete functions. Each function will perform a specific task when triggered by an event.



Pros of Serverless Architecture:

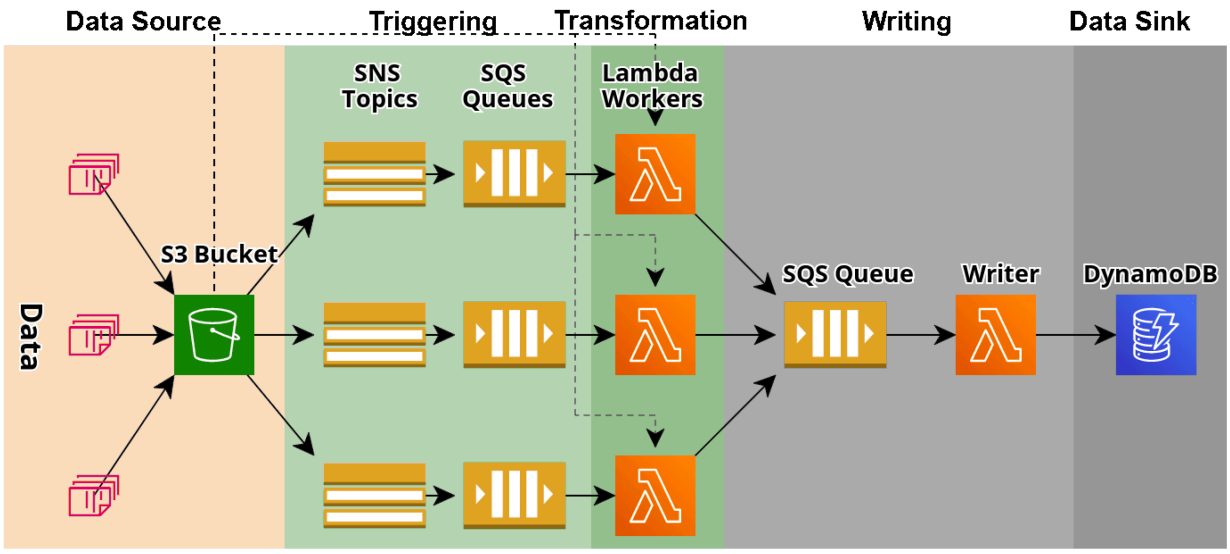
- No server management is necessary
- Charged based on uses which reduce cost
- Can be scaled endlessly and automatically
- Deployment and updates are faster
- Can be run on servers that are closer to end users all over the globe using CDN

Cons of Serverless Architecture:

- Testing and debugging become more challenging
- Vendor lock-in is a risk
- Are not built for long running processes
- May need to boot up which leads to degrade performance

ETL Pipeline Example

An event driven ETL pipeline using serverless AWS services is given below.



It consists of a data source, processing stage and a data sink which uses S3, SNS Topic, SQS Queue, Lambda and DynamoDB.

Amazon S3: An object storage service that offers industry-leading scalability, data availability, security, and performance. In our architecture it is used as a primary source of data

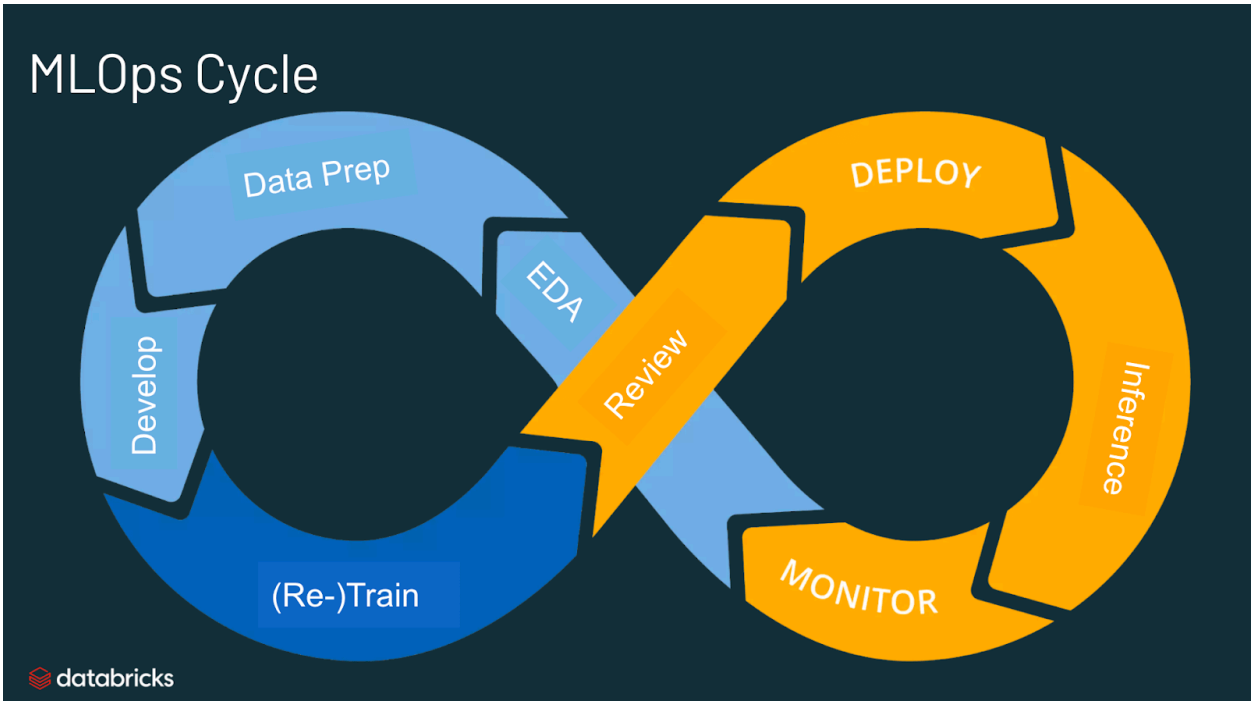
Amazon SNS Topic: A logical access point that acts as a communication channel. We used it to trigger changes in s3.

Amazon SQS Queue: A fully managed message queuing service that enables us to decouple and scale microservices, distributed systems, and serverless applications. We used it to store and retrieve the trigger event.

Amazon DynamoDB: A fully managed, serverless, key-value NoSQL database designed to run high-performance applications at any scale. We used it to store the processed data.

MLOps

MLOps stands for Machine Learning Operations. It focused on streamlining the process of taking machine learning models to production, and then maintaining and monitoring them.



Automating model development and deployment with MLOps means faster go-to-market times and lower operational costs. It helps managers and developers be more agile and strategic in their decisions.