COLLÈGE ROSEMONT

Big Data : Systèmes de gestion de données

Par : Abderrazak Sahraoui

Sommaire

- Base de données
- Entrepôt de données
- Lac de données
- Entrepôt lac de données

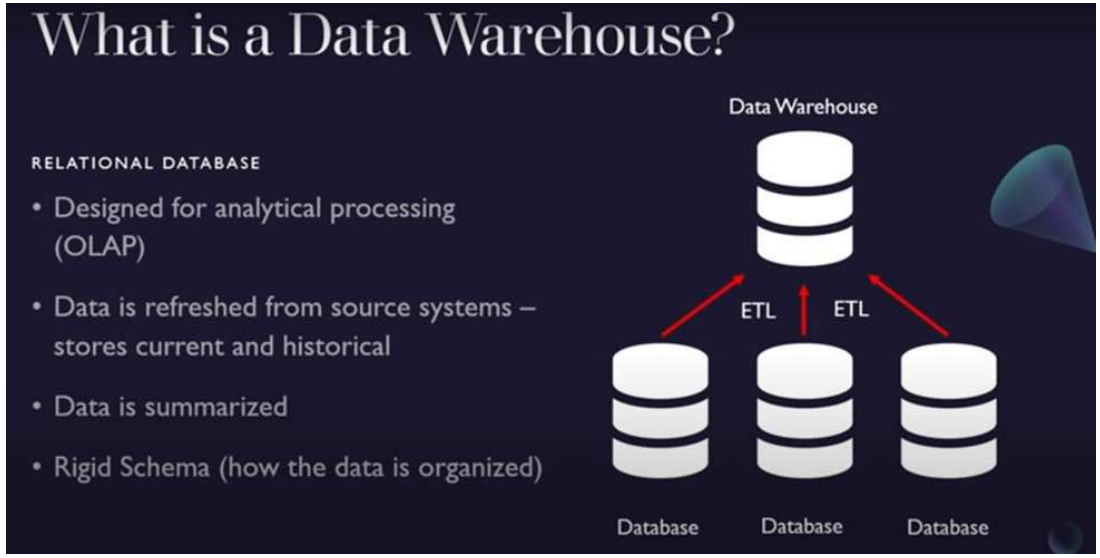
- Quand utiliser une base de donnée vs un entrepôt de données vs lac de données?
- Quel est le rôle de l'architecte dans la construction des systèmes de données?
- Quelle est la différence entre un schéma de base de données et schéma d'un entrepôt?
- Une donnée voyage-t-elle durant son cycle de vie entre BD, ED et LD ?
- Comment déterminer la fraicheur d'une donnée ?
- Quelle est la source de données pour chaque structure ?

Base de données

https://www.youtube.com/watch?v=-bSkREem8dM&ab_channel=AlexTheAnalyst

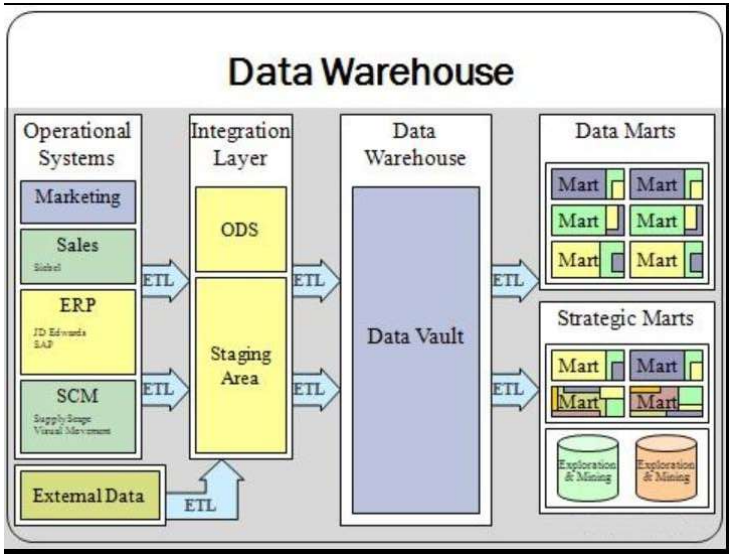
- Une base de données permet d'enregistrer des données provenant d'un processus OLTP (Online Transaction Process). Autrement dit, elle permet de capturer des données fraichement créés par une application.
- Les données sont stockées dans les bases de données relationnelles sous forme de table de plusieurs colonnes et plusieurs lignes.
- Le schéma relationnel d'une table peut être aisément modifié comparé à s structures. Il est possible d'ajouter de nouvelles colonnes à une table ou d'en supprimer ou modifier.

Entrepôt de données



- Un entrepôt de données permet d'entreposer des données provenant de différentes sources de données. Lesquelles données sont destinées à être analysées par un processus OLAP (Online Analytics Process)
- Le but d'un entrepôt de données est de fournir une référence unique pour un ensemble de données pouvant servir dans la prise de décisions au sein de l'entreprise, et d'offrir les outils nécessaires aux processus analytiques BI (Business intelligence ou Informatique décisionnelle).

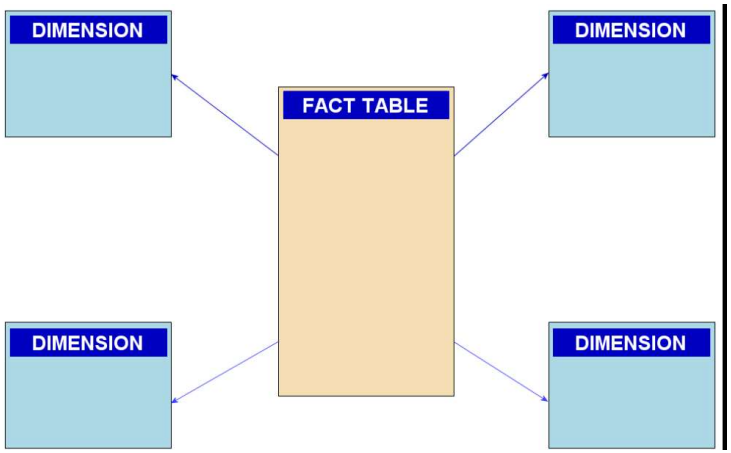
Entrepôt de données



https://fr.wikipedia.org/wiki/Entrep%C3%B4t_de_donn%C3%A9es

- En amont, Les données arrivent à l'entrepôt par le biais d'un processus ETL (Extract, Transforme et Load). Les données sont extraites de sources localisées dans des systèmes transactionnels en production. Les données sont épurées ou transformées par filtrage, codage et certification.
- Les données de l'entrepôt peuvent être conservées sous deux formes :
  - sous forme élémentaire et détaillée.
  - sous forme agrégée selon des axes ou des dimensions d'analyse prévues. il est impossible de retrouver le détail et la profondeur des indicateurs une fois ceux-ci agrégés. (par exemple, si l'on a agrégé les résultats par mois, il ne sera plus possible de faire une analyse par journée).
- En aval, les données peuvent être restituées aux usagers par des outils OLAP de :
  - requêtes ou reporting,
  - cubes ou hypercubes,
  - fouille de données.

Entrepôt de données

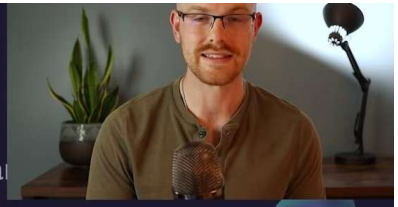


https://fr.wikipedia.org/wiki/%C3%89toile_(mod%C3%A8le_de_donn%C3%A9es)

- Les entrepôts utilisent le modèle dit en étoile ou les tables sont réparties en deux catégories : tables de faits et tables de dimension.
- Chaque modèle en étoile est constitué d'une table centrale de faits contenant les mesures comme montant, quantité, etc. et de plusieurs tables de dimension comme le temps (jour, mois, année) nomenclature (famille de produit, sous-famille, …) segmentation clientèle (sexe, tranche âge,…)
- La jointure dans un modèle en étoile entre table de faits et tables de dimension est facilitée (optimisée) par la présence d'une clé calculée à partir des clés des tables de dimension ce qui facilite l'analytique.
- Le modèle dit en flocon est une variante du modèle en étoile ou les tables de dimensions sont normalisées évitant ainsi le redondance et permettant un gain d'espace de l'ordre de 5 à 10%.

Comparatif

Key Differences



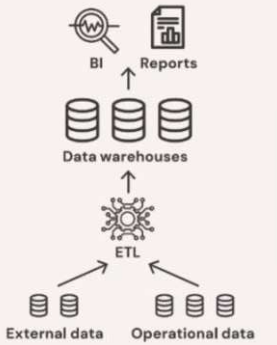
- Databases are designed for Transactions, Data Warehouses a
  analytics and reporting
- Databases data is fresh and detailed, Data Warehouses data is refreshed periodically and is summarized
- Databases work slowly for querying large amounts of data and can slow down transactional processes, Data Warehouses don't interfere with any processes and are generally faster

Comparatif

Caractéristique	Base de données de production	Data warehouses	Datamarts
Opération	gestion courante, production	référentiel, analyse ponctuelle	analyse récurrente, outil de pilotage, support à la décision
Modèle de données	entité/relation	3NF, étoile, flocon	étoile, flocon
Normalisation	fréquente	maximum	rare (redondance d'information)
Données	actuelles, brutes, détaillées	historisées, détaillées	historisées, agrégées
Mise à jour	immédiate, temps réel	souvent différée, périodique	souvent différée, périodique
Niveau de consolidation	faible	faible	élevé
Perception	verticale	transverse	horizontale
Opérations	lectures, insertions, mises à jour, suppressions	lectures, insertions, mises à jour	lectures, insertions, mises à jour, suppressions
Taille	en gigaoctets	en téraoctets	en gigaoctets

Data Warehouse

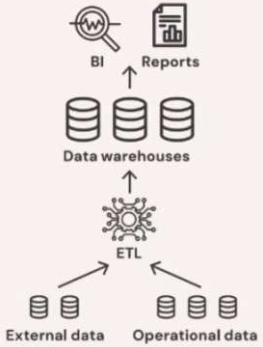
Data warehouse



Pros:

- Business intelligence (BI)
- Analytics
- Structured & clean data
- Predefined schemas

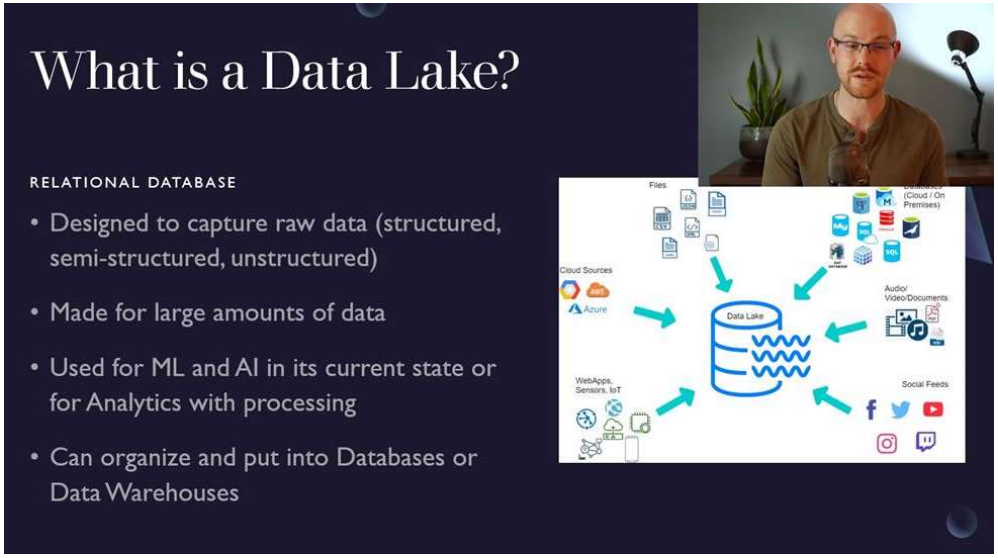
Data warehouse



Cons:

- No support for semi or unstructured data
- Inflexible schemas
- Struggled with volume and velocity upticks
- Long processing time

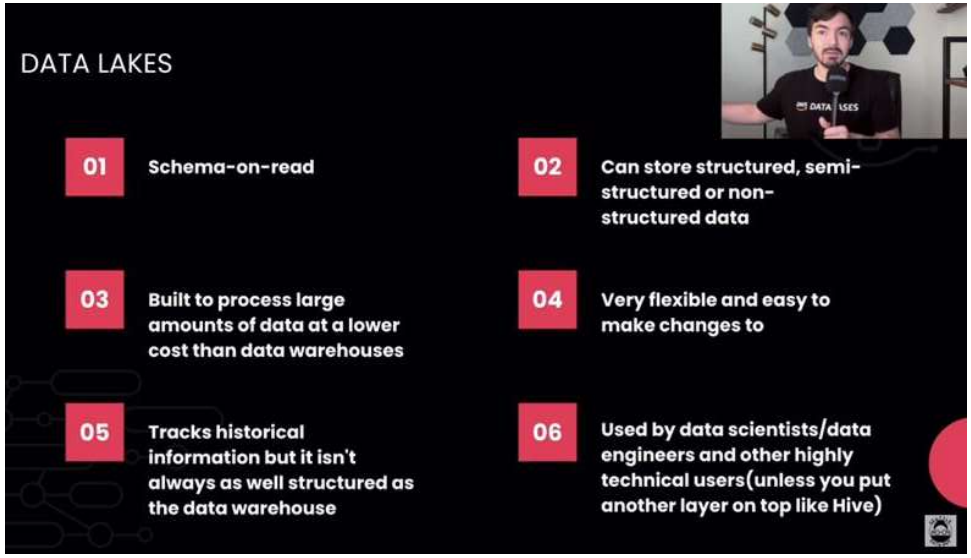
Lac de données



https://fr.wikipedia.org/wiki/Lac_de_donn%C3%A9es

- Un lac de donnée permet le stockage rapide de données massives hétérogènes dans leur format original ou avec peu de transformation.
  - Données structurées issues de bd relationnelles.
  - Données issues de bases NoSQL
  - Données semi-structurées (fichiers CSV, journaux, xml, json,...)
  - Données non structurées (emails, documents, pdf
  - Fichiers blob (images, audio, vidéo)
- Les lacs sont utilisées par des ingénieurs de données et des scientifiques de données pour des applications en apprentissage machine et intelligence artificielle.
- Lorsqu'une donnée arrive au lac, elle se verra dotée d'un identifiant et de balises de métadonnées. Lorsqu'un besoin se présente, le Data Lake est parcouru pour y rechercher des informations pertinentes. L'analyse de ces données permet alors d'apporter de la valeur et de répondre à ce besoin.
- Le stockage se fait en utilisant l'architecture d'un cluster Hadoop.
- Les données peuvent être conservées dans le lac pour un usage ultérieur non prédéterminé.

Lac de données



https://www.youtube.com/watch?v=FxpRL0m9BcA&ab_channel=SeattleDataGuy

- Les avantages des lacs de données sont :
  - la rationalisation du stockage des données,
  - la réduction des coûts de stockage,
  - et la facilitation de l'accès pour l'analyse et la prise de décisions d'une façon globale.
- Les inconvénients sont :
  - la difficulté à conserver un lac de données propre et organisé,
  - la difficulté à organiser et maintenir une gouvernance des données efficace,
  - le temps nécessaire à traiter et analyser les données stockées à l'état brut.
  - L'expertise requise pour rechercher, analyser et traiter les données de manière pertinente et créatrice de valeur, souvent confiées aux Data Scientists
  - la sécurité, la confidentialité et les problématiques liées aux données personnelles et au respect des réglementations.
- Plusieurs environnements fournissent des services complets permettant la gestion d'un lac de données. La plupart d'entre eux sont basés sur la technologie Hadoop et fournissent des installations en local (MapR, Cloudera, Hortonworks) ou dans le Cloud (Microsoft Azure, Google Cloud Platform, Amazon S3)

Lac de données

Data warehouse vs. data lake

	DATA WAREHOUSE	DATA LAKE
DATA TYPES	Structured, processed data from operational databases, applications and transactional systems	Structured, semistructured and unstructured data from sensors, apps, websites, etc.
PURPOSE	Predefined purpose for business intelligence, batch reporting and data visualization	May not have a predefined purpose; typically used for machine learning, deep analysis and discovery
USERS	Data engineers, business analysts, data analysts	Data engineers, data scientists
SCHEMA POSITION	Schema-on-write	Schema-on-read
BENEFITS	Categorized historical data stored in a single repository with ease of access for the end user	Data stored in its native format, allowing flexibility for data scientists to analyze and develop models from diverse data sources

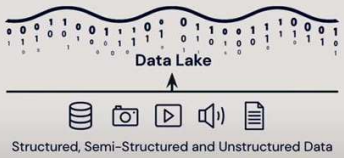
- ....

Lac de données

Pros:

- Flexible data storage
- Streaming support
- Cost efficient in the cloud
- Support for AI and Machine Learning

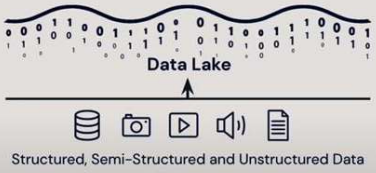
Data Lakes



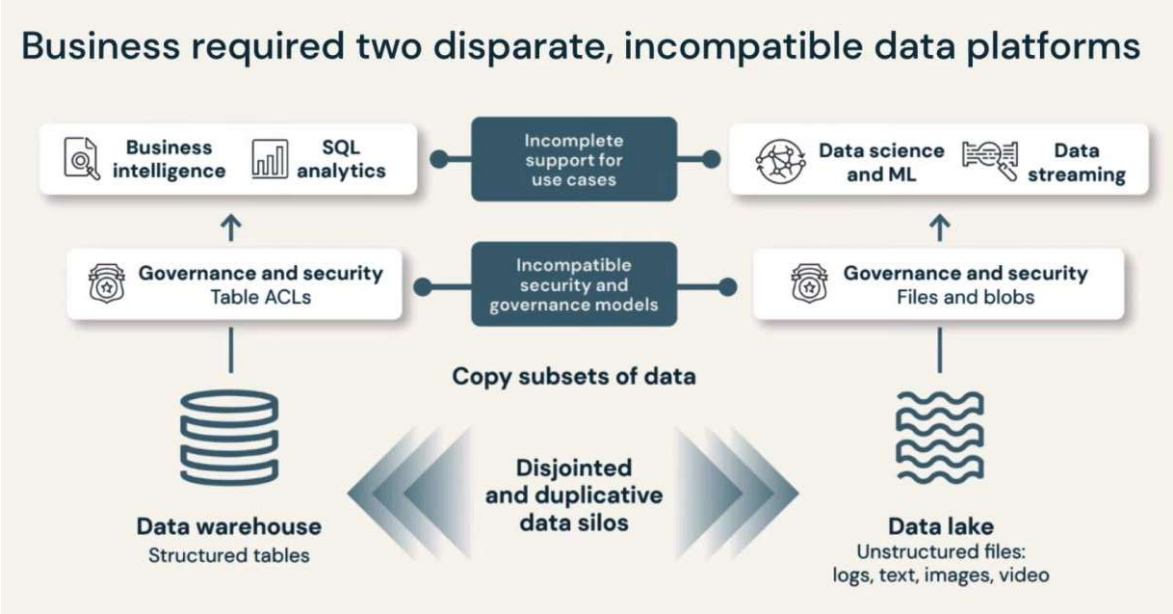
Cons:

- No transactional support
- Poor data reliability
- Slow analysis performance
- Data governance concerns
- Data warehouses still needed

Data Lakes



Entrepôt et lac de données



- Besoin d'une plateforme de données qui combine les avantages des deux solutions et élimine leurs inconvénients.

Databricks Lakehouse

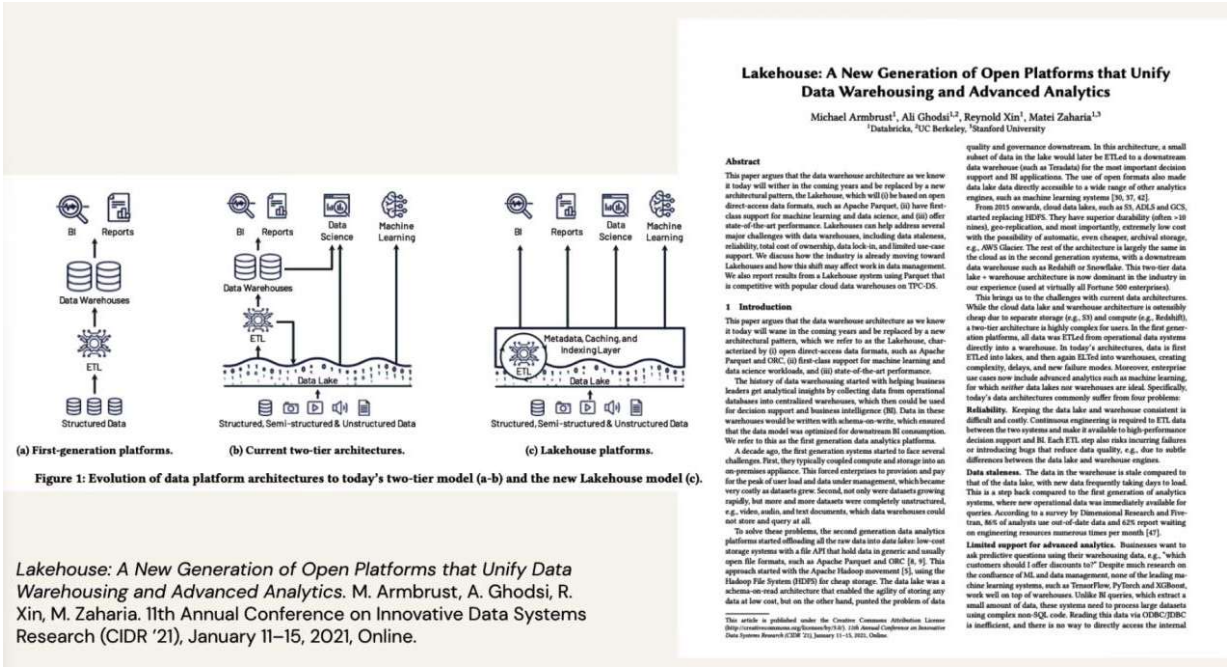


Figure 1: Evolution of data platform architectures to today's two-tier model (a-b) and the new Lakehouse model (c).

Lakehouse: A New Generation of Open Platforms that Unify Data Warehousing and Advanced Analytics. M. Armbrust, A. Ghodsi, R. Xin, M. Zaharia. 11th Annual Conference on Innovative Data Systems Research (CIDR '21), January 11–15, 2021, Online.

Lakehouse: A New Generation of Open Platforms that Unify Data Warehousing and Advanced Analytics

Michael Armbrust[1], Ali Ghodsi[1,2], Reynold Xin[1], Matei Zaharia[1,3]
[1]Databricks, [2]UC Berkeley, [3]Stanford University

Abstract

This paper argues that the data warehouse architecture as we know it today will wither in the coming years and be replaced by a new architectural pattern, the Lakehouse, which will (i) be based on open direct-access data formats, such as Apache Parquet, (ii) have first-class support for machine learning and data science, and (iii) offer state-of-the-art performance. Lakehouses can help address several major challenges with data warehouses, including data staleness, reliability, total cost of ownership, data lock-in, and limited use-case support. We discuss how the industry is already moving toward Lakehouses and how this shift may affect work in data management. We also report results from a Lakehouse system using Parquet that is competitive with popular cloud data warehouses on TPC-DS.

1 Introduction

This paper argues that the data warehouse architecture as we know it today will wane in the coming years and be replaced by a new architectural pattern, which we refer to as the Lakehouse, characterized by (i) open direct-access data formats, such as Apache Parquet and ORC, (ii) first-class support for machine learning and data science workloads, and (iii) state-of-the-art performance.

The history of data warehousing started with helping business leaders get analytical insights by collecting data from operational databases into centralized warehouses, which then could be used for decision support and business intelligence (BI). Data in these warehouses would be written with schema-on-write, which ensured that the data model was optimized for downstream BI consumption. We refer to this as the first generation data analytics platforms.

A decade ago, the first generation systems started to face several challenges. First, they typically coupled compute and storage into an on-premises appliance. This forced enterprises to provision and pay for the peak of user load and data under management, which became very costly as datasets grew. Second, not only were datasets growing rapidly, but more and more datasets were completely unstructured, e.g., video, audio, and text documents, which data warehouses could not store and query at all.

To solve these problems, the second generation data analytics platforms started offloading all the raw data into *data lakes*: low-cost storage systems with a file API that hold data in generic and usually open file formats, such as Apache Parquet and ORC [8, 9]. This approach started with the Apache Hadoop movement [5], using the Hadoop File System (HDFS) for cheap storage. The data lake was a schema-on-read architecture that enabled the agility of storing any data at low cost, but on the other hand, punted the problem of data quality and governance downstream. In this architecture, a small subset of data in the lake would later be ETLed to a downstream data warehouse (such as Teradata) for the most important decision support and BI applications. The use of open formats also made data lake data directly accessible to a wide range of other analytics engines, such as machine learning systems [30, 37, 42].

From 2015 onwards, cloud data lakes, such as S3, ADLS and GCS, started replacing HDFS. They have superior durability (often >10 nines), geo-replication, and most importantly, extremely low cost with the possibility of automatic, even cheaper, archival storage, e.g., AWS Glacier. The rest of the architecture is largely the same in the cloud as in the second generation systems, with a downstream data warehouse such as Redshift or Snowflake. This two-tier data lake + warehouse architecture is now dominant in the industry in our experience (used at virtually all Fortune 500 enterprises).

This brings us to the challenges with current data architectures. While the cloud data lake and warehouse architecture is ostensibly cheap due to separate storage (e.g., S3) and compute (e.g., Redshift), a two-tier architecture is highly complex for users. In the first generation platforms, all data was ETLed from operational data systems directly into a warehouse. In today's architectures, data is first ETLed into lakes, and then again ELTed into warehouses, creating complexity, delays, and new failure modes. Moreover, enterprise use cases now include advanced analytics such as machine learning, for which *neither* data lakes nor warehouses are ideal. Specifically, today's data architectures commonly suffer from four problems:

Reliability. Keeping the data lake and warehouse consistent is difficult and costly. Continuous engineering is required to ETL data between the two systems and make it available to high-performance decision support and BI. Each ETL step also risks incurring failures or introducing bugs that reduce data quality, e.g., due to subtle differences between the data lake and warehouse engines.

Data staleness. The data in the warehouse is stale compared to that of the data lake, with new data frequently taking days to load. This is a step back compared to the first generation of analytics systems, where new operational data was immediately available for queries. According to a survey by Dimensional Research and Fivetran, 86% of analysts use out-of-date data and 62% report waiting on engineering resources numerous times per month [47].

Limited support for advanced analytics. Businesses want to ask predictive questions using their warehousing data, e.g., "which customers should I offer discounts to?" Despite much research on the confluence of ML and data management, none of the leading machine learning systems, such as TensorFlow, PyTorch and XGBoost, work well on top of warehouses. Unlike BI queries, which extract a small amount of data, these systems need to process large datasets using complex non-SQL code. Reading this data via ODBC/JDBC is inefficient, and there is no way to directly access the internal

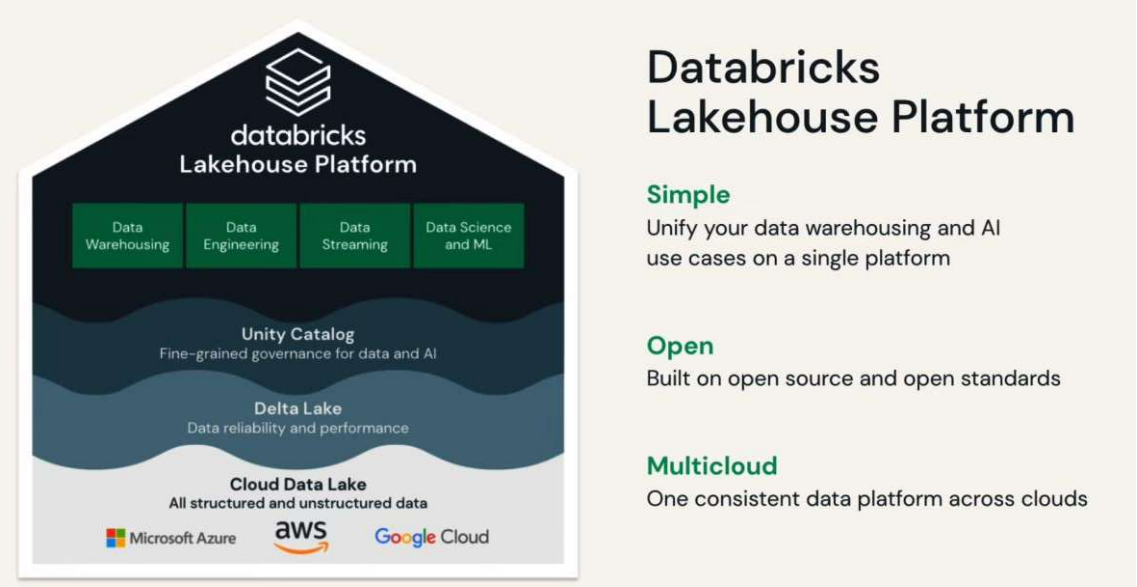
https://www.databricks.com/learn/training/lakehouse-fundamentals-accreditation

Lakehouse : Entrepôt lac de données

https://www.youtube.com/watch?v=Enu-EH7RHHM&ab_channel=IBMTechnology

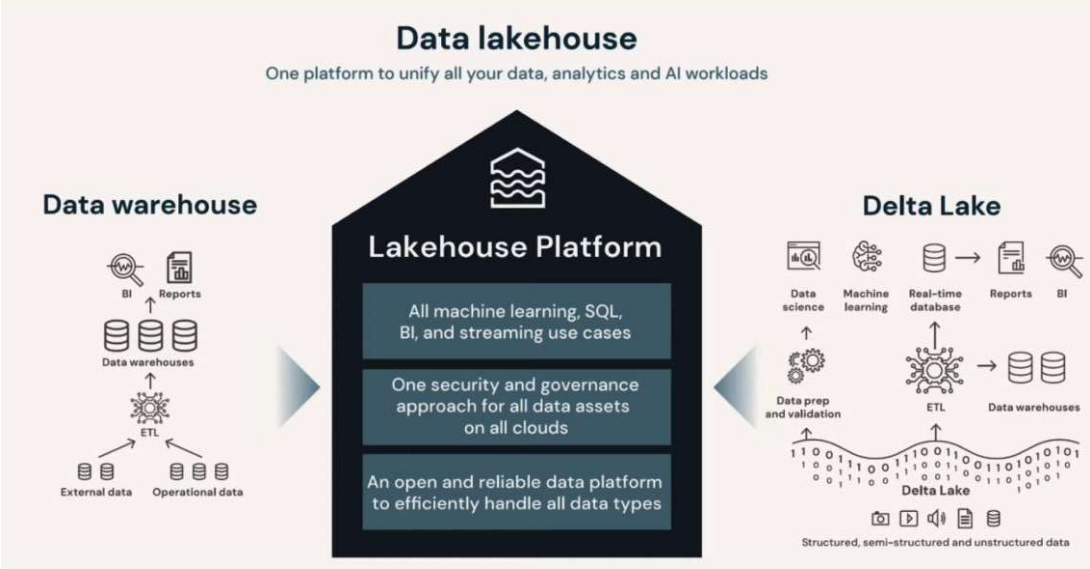
- ....

Databricks Lakehouse



https://www.youtube.com/watch?v=Enu-EH7RHHM&ab_channel=IBMTechnology

Entrepôt lac de données



- Databricks offre une architecture hybride unifiant une plateforme d'entrepôt de données et une plateforme de lac de données.

Entrepôt lac de données