



UNIVERSITE DE CORSE  
Master Informatique  
parcours DFS et DE  
1<sup>ère</sup> année  
2025-2026

**BD partie 3**  
**CH1 – Principes des**  
**DatawareHouses**

Evelyne VITTORI  
[vittori\\_e@univ-corse.fr](mailto:vittori_e@univ-corse.fr)



# Plan du cours



## CH1 – Principes des Datawarehouse

- Objectifs
- Différences avec une BD
- Architectures DW, DL et DLH



## CH2 – Modélisation dimensionnelle

- Concepts de modélisation dimensionnelle
- Schémas en étoile et en flocon



## CH3 – Processus ETL

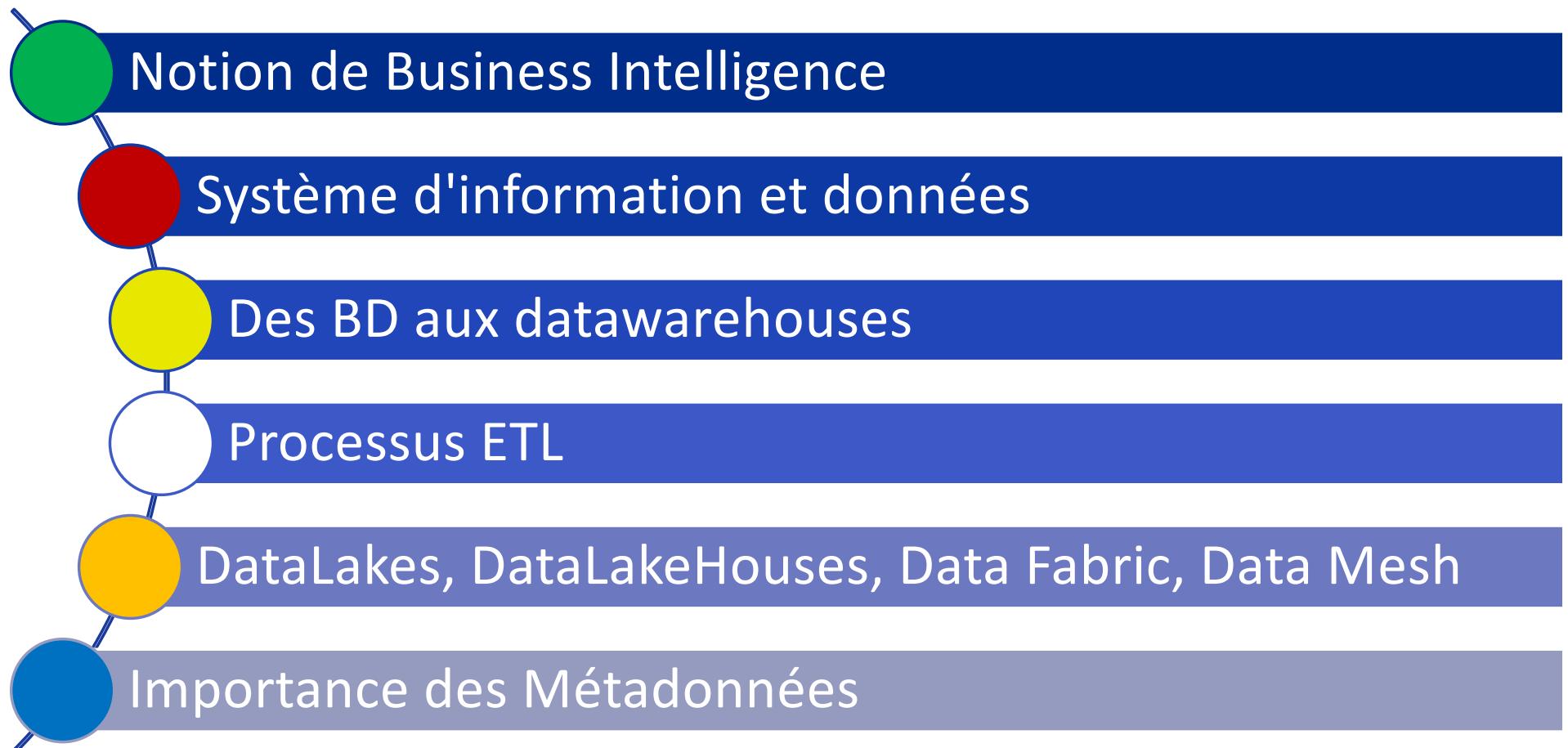
- Définition et rôle d'un processus ETL
- Principaux outils
- Mise en pratique avec PentahoDI



## CH4 – Exploitation d'un DW

- Principes OLAP
- Notion d'hypercube OLAP
- Langage MDX
- Mise en pratique avec IcCube

# CH1 – Principes des Datawarehouse



# 1 – Notion de Business Intelligence



# Notion de Business Intelligence (ou intelligence décisionnelle)

- Objectif : Aider les entreprises à prendre des décisions éclairées
- Quels acteurs ?



Informaticiens et Analystes BI  
*(équipe IT Information Technology)*



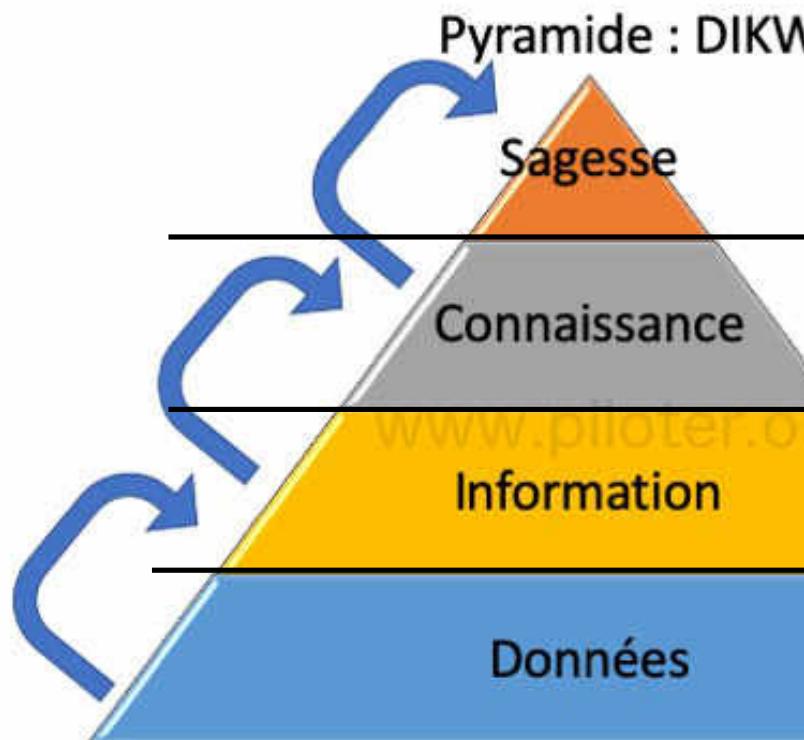
Décideurs  
*(non informaticiens, non statisticiens )*

# Notion de Business Intelligence (ou intelligence décisionnelle)

- Objectif : Aider les entreprises à prendre des décisions éclairées
- Comment?
  - **Collecter** des données de diverses sources
  - **Analyser** les données
    - identifier des tendances, des corrélations, ...
  - Proposer des indicateurs et des **visualisations** : tableaux de bord, statistiques, indicateurs de performance, ....

# Transformer les données en connaissances

"Data Information Knowledge Wisdom"



*Nous allons concentrer nos promotions sur l'électronique en soirée*



Les clients privilégient les achats en soirée pour des promotions sur l'électronique

40% des ventes concernent l'électronique  
Le pic d'achats se situe entre 18h et 20h

10 000 produits vendus répartis sur 5 catégories

# Tendance actuelle : le BI Libre-Service

- Les décideurs et utilisateurs métiers sont autonomes
  - Création de leurs propres rapports, tableaux de bord et analyses de données sans dépendre directement du service informatique (IT)
  - Gain de temps et flexibilité pour répondre rapidement aux besoins d'analyse
- Outils spécifiques intuitifs orientés dataviz
  - PowerBI , Tableau : outils intuitifs et puissants, adaptés aux utilisateurs métier



# BI traditionnel et BI Self-Service

TRADITIONAL BI	SELF-SERVICE BI
Business user gathers requirements for a report/dashboard.	IT team gathers user requests for self-service tool.
User submits request to IT.	Self-service tool is implemented, giving business users access to data.
IT extracts the data and loads it into a data warehouse for analysis.	Business user accesses data directly.
IT creates data model.	Business user prepares data to include.
User approves report or dashboard, or requests changes.	Business user creates data model.



source : <https://www.lemagit.fr/conseil/BI-traditionnelle-ou-BI-en-libre-service-Pourquoi-choisir>

# Pas d'inquiétude pour les data-analystes!

- Rôle toujours nécessaire de l'équipe IT
  - Travail préalable de modélisation des données BI
  - Préparer la couche de présentation
- Besoin de formation des utilisateurs métiers

**« [La BI en libre-service] ne saurait remplacer totalement la BI traditionnelle ».**

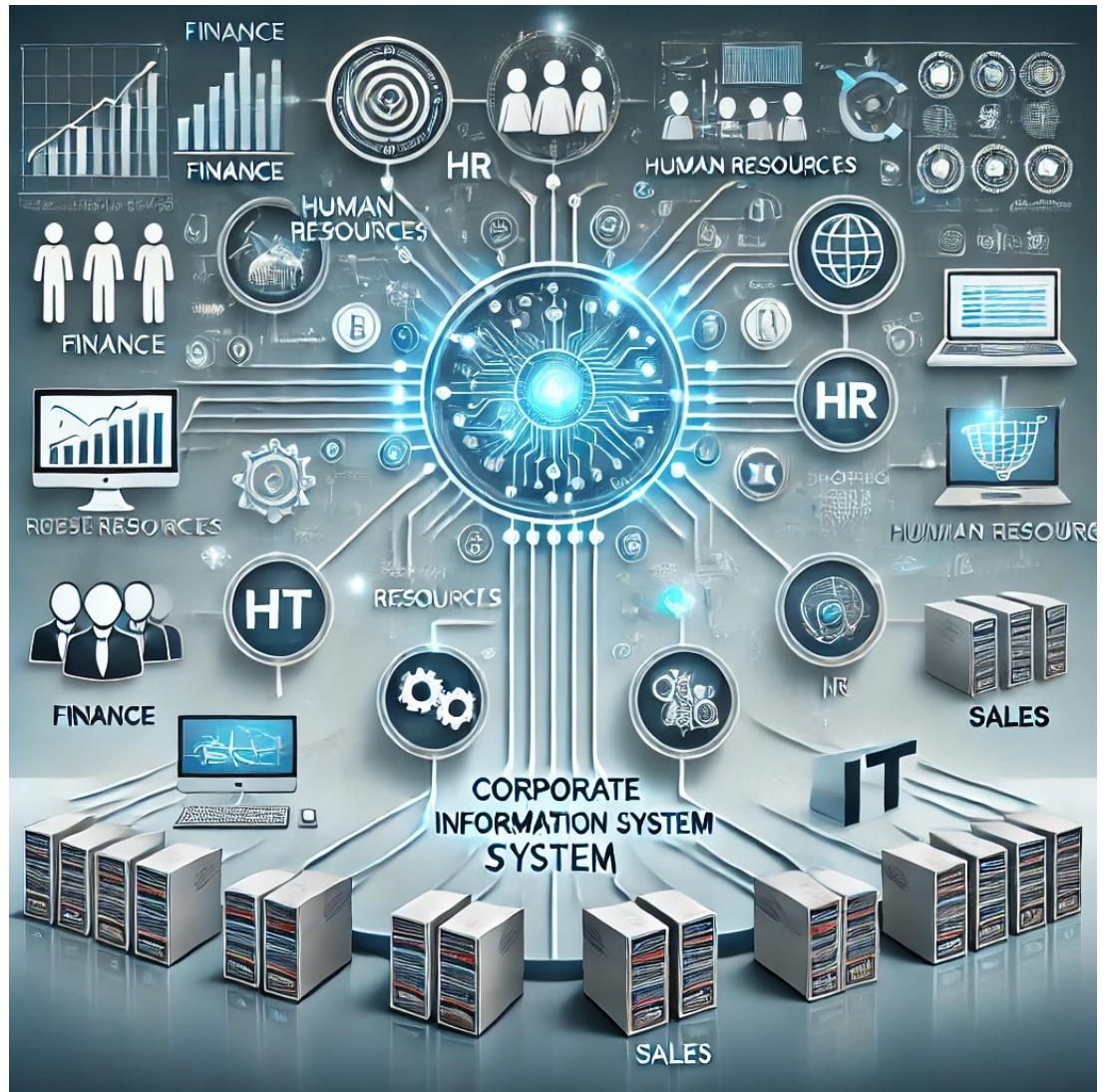
**Igor Ikonnikov**

Directeur de recherches, Info-Tech Research Group

## 2 – Système d'information et données



# Qu'est-ce qu'un Système d'Information ?



# Qu'est-ce que le système d'information SI d'une entreprise ?

- Une notion essentielle : une vision globale non limitée au contexte informatique
- Ensemble des **ressources** et de **processus** qui permettent le fonctionnement et la gestion de l'entreprise

*Le système d'information (SI) est un ensemble organisé de ressources qui permet de collecter, stocker, traiter et distribuer de l'information*

*@wikipédia*

# Notion de Ressource



Ressources  
humaines



Ressources  
matérielles



Ressources  
immatérielles

## Exemple d'une entreprise de vente en ligne



# Notion de processus dans un SI

# Quels processus dans un SI?

## Processus métiers

- Opérations permettant d'atteindre des objectifs opérationnels liés à l'activité principale
- Exemples
  - Gestion des commandes
  - Logistique et livraison
  - Gestion des clients



## Processus décisionnels

- Suivi et gestion des activités des processus métiers pour orienter les décisions stratégiques
- Exemples
  - Indicateurs de performance
  - Analyse de données
  - Prévisions



# Données des processus métiers

## ■ **Données opérationnelles**

- Produites par les activités quotidiennes de l'entreprise
- Utilisées pour le fonctionnement opérationnel

## ■ **Exemples :**

- *Données clients*
- *Commandes*
- *Stock des produits*



Données de  
fonctionnement  
"classique"

# Données des processus décisionnels

## ■ Données analytiques

- Dérivées ou agrégées à partir des données opérationnelles
- Utilisées pour la prise de décision stratégique

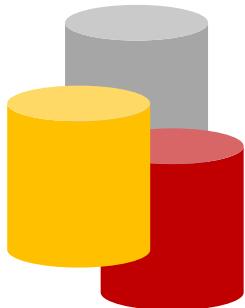
## ■ Exemples :

- *Tendances de vente par région*
- *Performances des produits*
- *Budget annuel*
- *Rapports financiers*

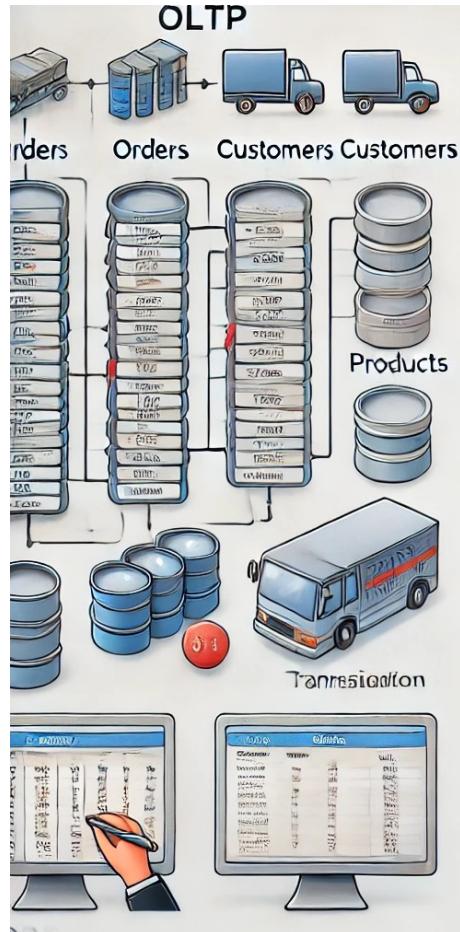


# Deux systèmes distincts au sein du SI

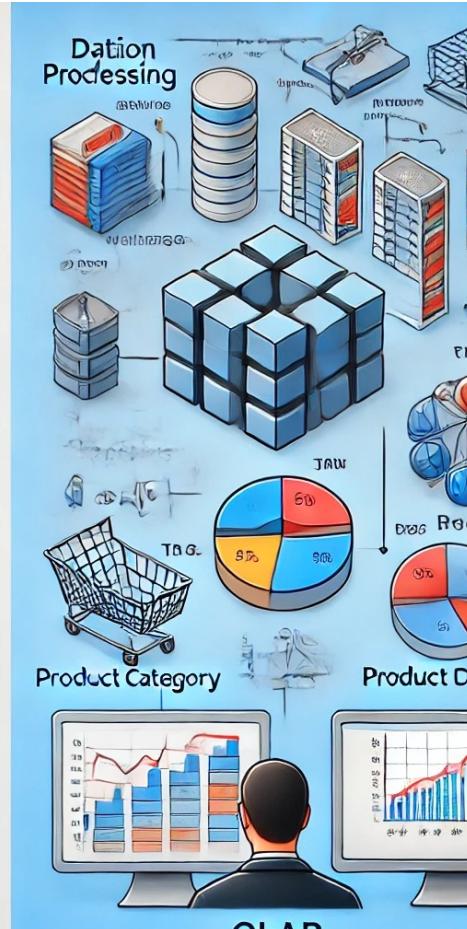
Données opérationnelles



BD opérationnelles "classiques"

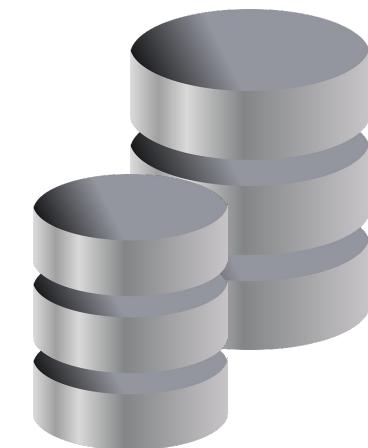


Online Transaction Processing



OnLine Analytical Processing

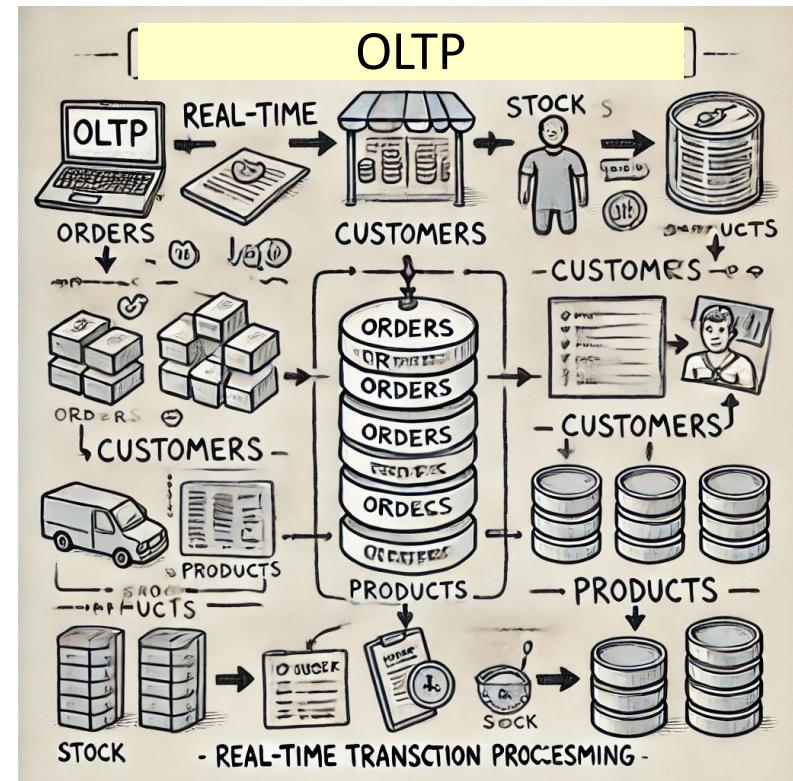
Données analytiques



Entrepôts de données

# OLTP Online Transaction Processing

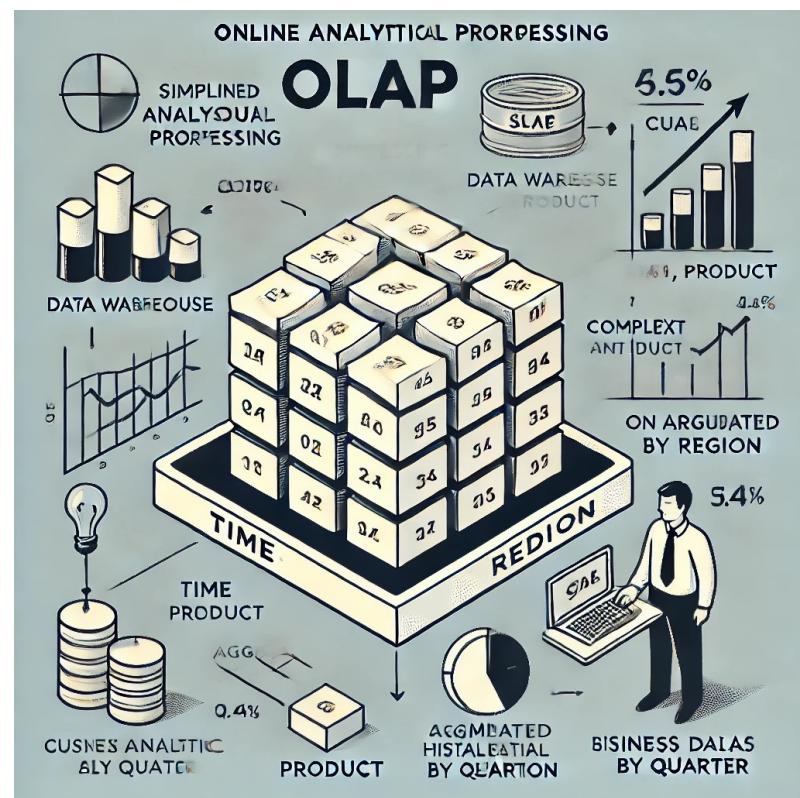
- **Transactions** en temps réel
- Nombreux utilisateurs
- Volume important
- **Mises à jour fréquentes**
- Maintien de l'intégrité
- Réponses **rapides**



Données normalisées -> principalement SGBD relationnels

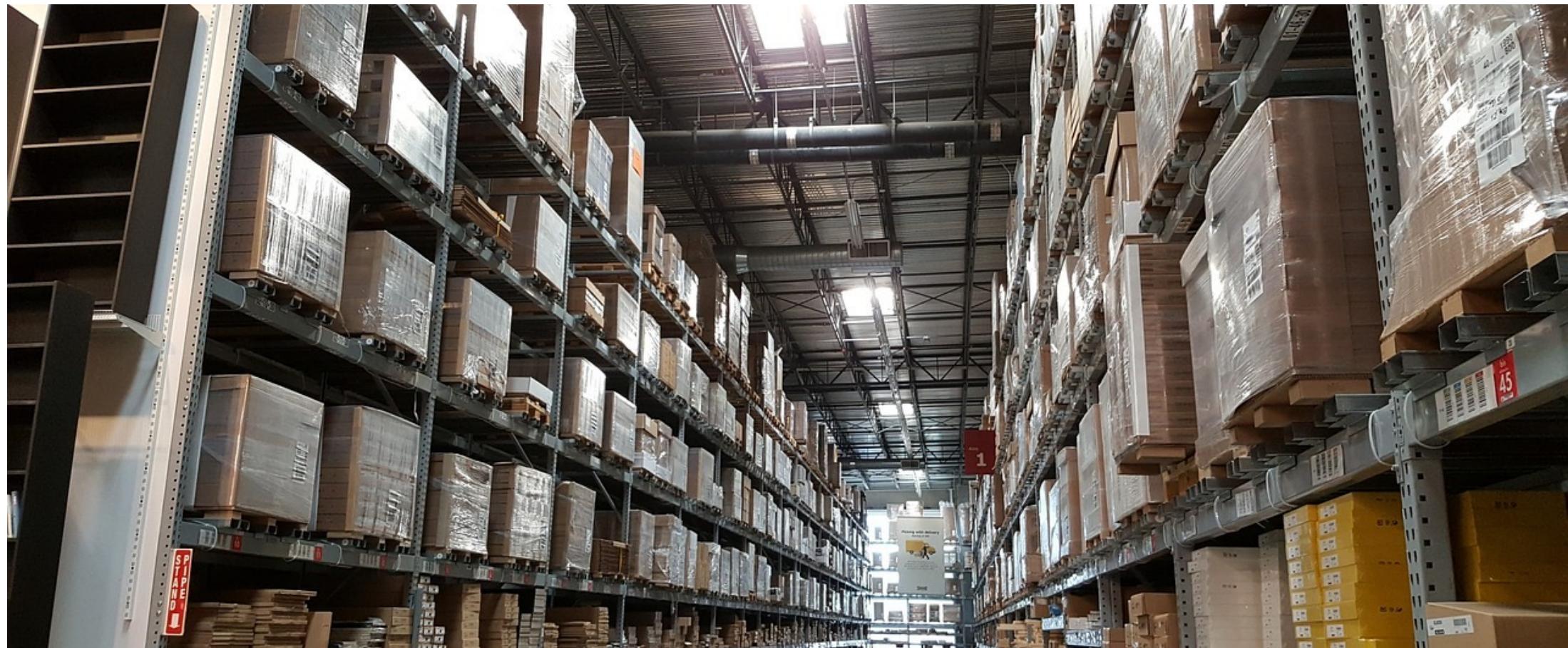
# OLAP OnLine Analytical Processing

- Données **historiques** issues de l'OLTP pour prendre des décisions à long terme
  - Analyse et **requêtes complexes**
  - Modélisation multidimensionnelle
  - Cubes OLAP



Données souvent non normalisées -> dataWareHouse

### 3 - Des BD aux DataWarehouse



# Dawarehouse ou Entrepôt de données DW ou DWH

- "Système de stockage de données intégré, non volatile, **orienté sujet et historisé**, organisé pour le support d'un processus **d'aide à la décision**"

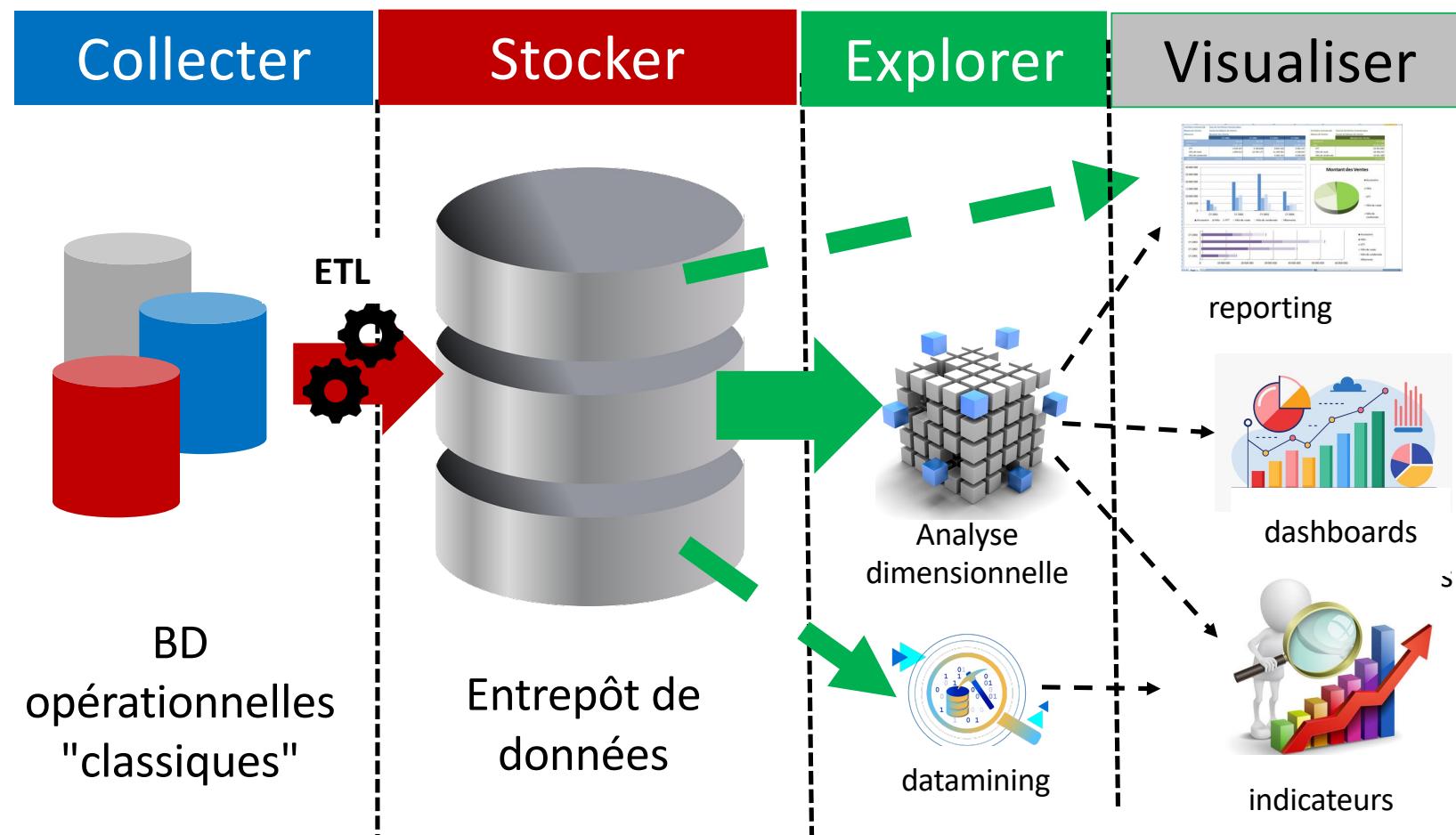


*Bill Inmon : le père des dataware houses*  
source : <https://datascientest.com/bill-inmon-biographie>



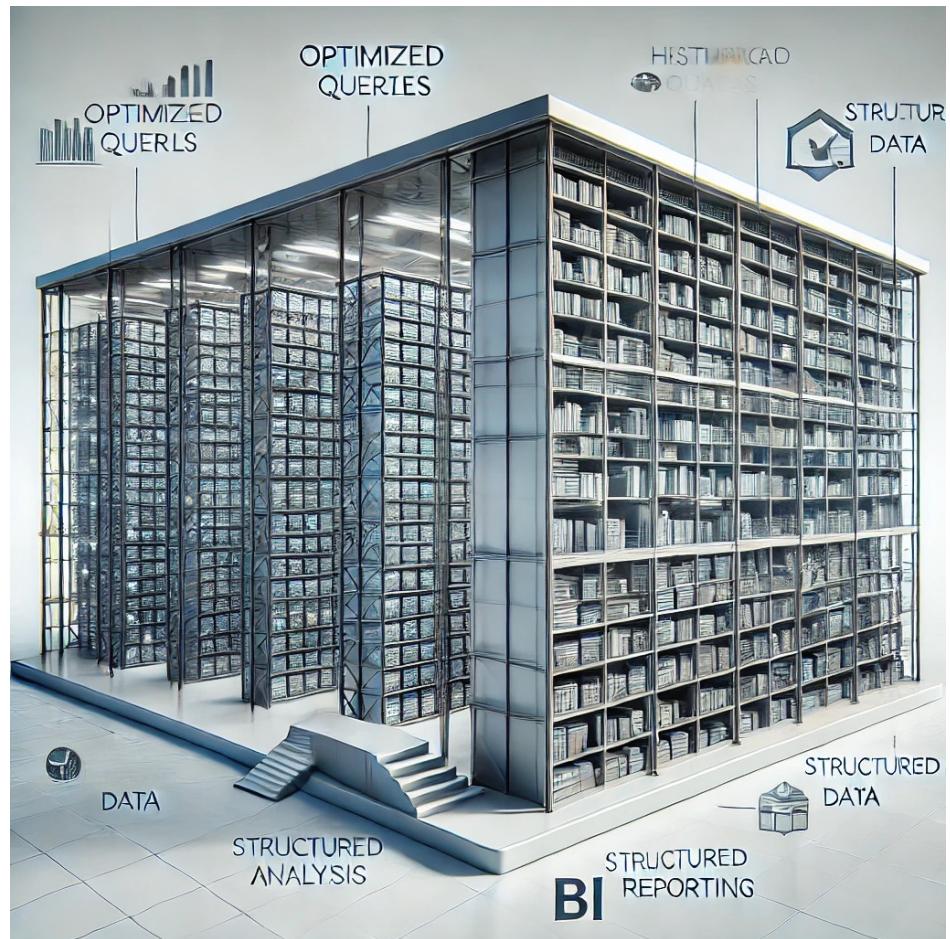
Data warehouse : BD destinée à l'analyse de données

# Processus d'aide à la décision et DataWarehouse



# Caractéristiques d'un DW

- Volume important
- Données orientées **sujet**
  - organisées par thèmes
  - référentiel unique
- Données **non volatiles**
  - ni modifications ni suppressions
- Données **chronologiques**
  - conservation de l'historique



# Quelles différences entre une BD et un datawareHouse?

## BD "classique"

- Conçue pour les transactions courantes
- Données actuelles et en temps réel
- Optimisée pour les **écritures**

## DW

- Conçu pour les requêtes et l'analyse
- Données historiques et agrégées
- Optimisé pour les **lectures**

BD dite "opérationnelle"

BD dite «analytique»

# Pourquoi les données d'un DW sont-elles en général dénormalisées?



# Limites des DW

- Peu adaptés aux décideurs
- Trop vastes et complexes
  - Contiennent trop d'informations
- Coût de traitement et de maintenance élevés
- Temps de réponses importants
- Inertie dans l'évolution

Idée de Datamart



# Notion de Datamart (ou magasin)

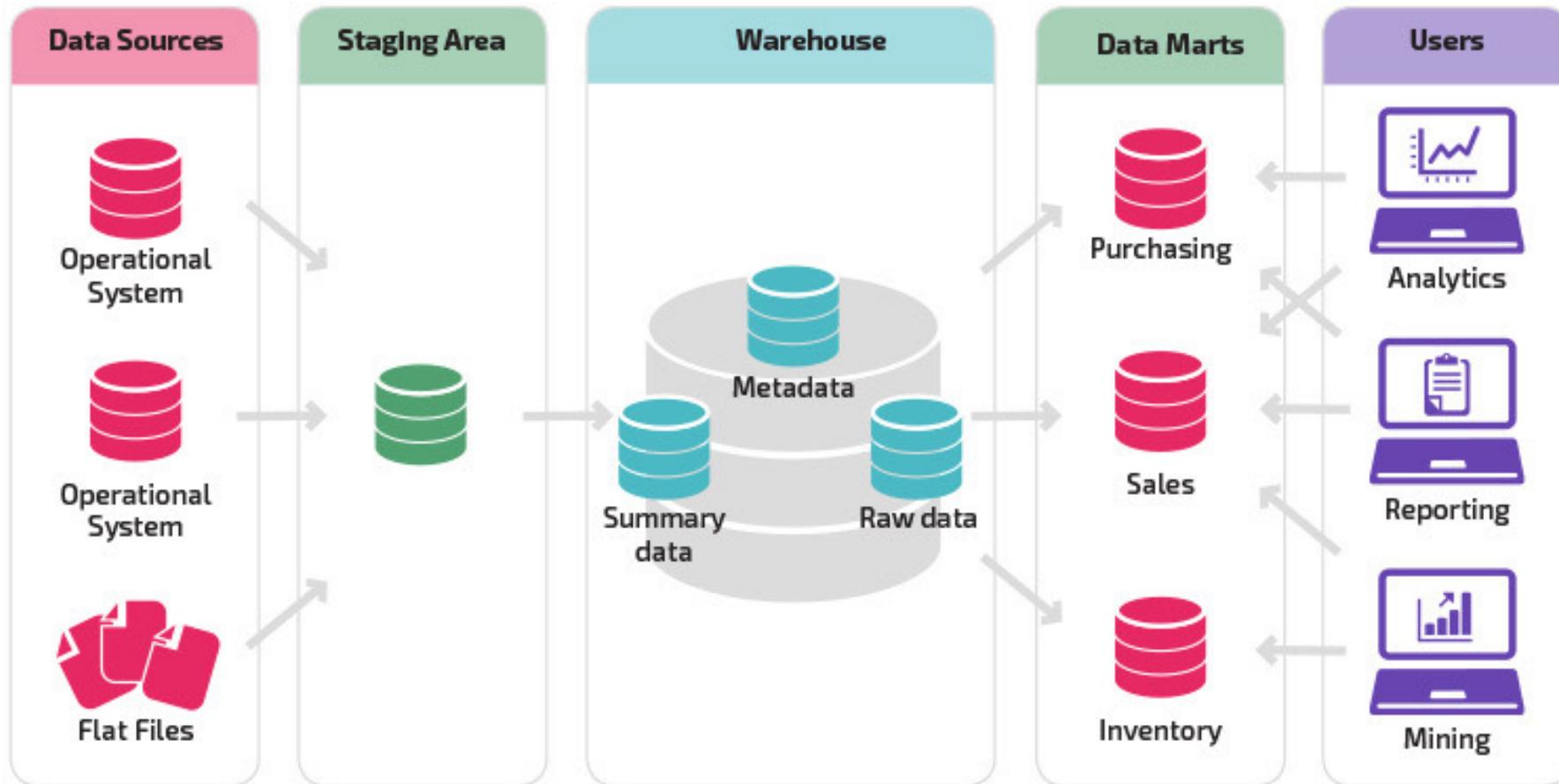
- Sous-ensemble d'un entrepôt
  - adapté à un besoin ou une fonctionnalité spécifique
- Plus facile à gérer, plus petit et moins complexe que l'ensemble de l'entrepôt de données.

Deux possibilités

- Tables internes au DW
- BD distinctes



# DW et Datamarts



source image : <https://panoply.io/data-warehouse-guide/data-mart-vs-data-warehouse/>

# Quel SGBD pour implémenter un DW?

## SGBD Relationnels

- PostgreSQL
- ORACLE
- MySQL / MariaDB
- Microsoft SQL Server

Notre choix  
dans ce cours

## SGBD spécialisés (cloud)

- Google BigQuery
- Amazon Redshift
- Snowflake

## SGBD hybrides (relationnels et NoSQL)

- Vertica
- Azure Synapse Analytics
- Teradata

Pour les très  
grands DW

# SGBD et DW

SGBD	Type	Avantages	Inconvénients	Cas d'utilisation
<b>PostgreSQL</b>	Relationnel open source	Open source, riche en fonctionnalités analytiques, extensible	Performance limitée pour très grands volumes	Petites à moyennes implémentations DW
<b>MySQL / MariaDB</b>	Relationnel open source	Facile à utiliser, faible coût, bon pour les petites bases	Moins performant sur des données volumineuses	Petites implémentations DW ou projets simples
<b>Microsoft SQL Server</b>	Relationnel propriétaire	Intégration BI native, fonctionnalités ETL et reporting	Coût élevé, dépendance à l'écosystème Microsoft	Moyennes à grandes entreprises utilisant BI Microsoft
<b>Oracle Database</b>	Relationnel propriétaire	Haute performance, fonctionnalités avancées (index bitmap, partitionnement)	Coût très élevé, complexité d'administration	Grandes entreprises avec besoins analytiques complexes
<b>Amazon Redshift</b>	Cloud spécialisé DW	Conception en colonnes, scalable, bien intégré à AWS	Coût élevé pour des usages fréquents, dépendant d'AWS	Organisations utilisant AWS pour des analyses volumineuses
<b>Google BigQuery</b>	Cloud spécialisé DW	Serverless, pay-as-you-go, rapide pour grandes analyses	Coût élevé pour des usages fréquents, dépendant de GCP	Analyses massives nécessitant une scalabilité serverless
<b>Snowflake</b>	Cloud spécialisé DW	Scalabilité flexible, interface intuitive, performances élevées	Coût par utilisation, dépendance au cloud	Entreprises cherchant un DW cloud performant et flexible

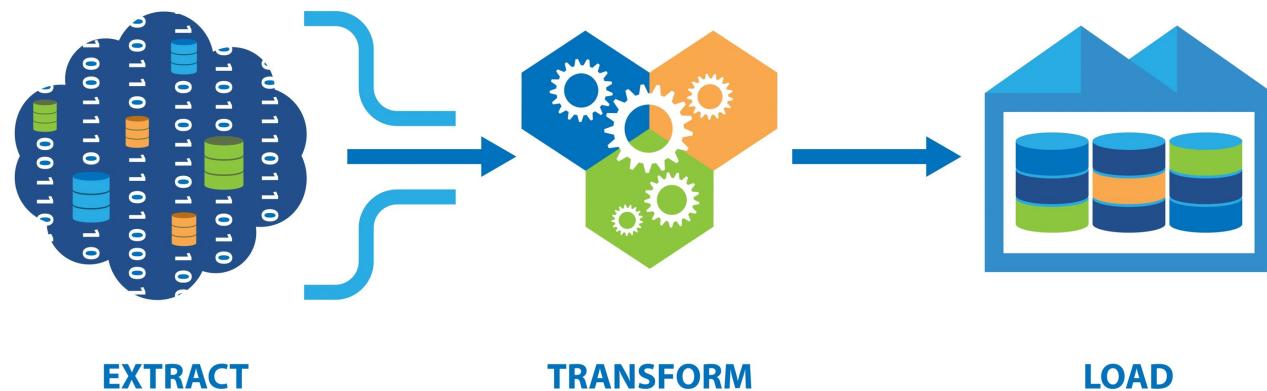
## 4 - Processus ETL (Extract Transform Load)

Comment alimenter un DW?



# Quel est le rôle du processus ETL?

- Méthode utilisée pour intégrer des données provenant de sources multiples dans un entrepôt de données ou un datamart
- Alimenter le DW de manière automatisée
- Trois étapes principales :



source image : <https://www.datachannel.co/blogs/what-is-etl-and-how-the-etl-process-works>

# Processus ETL

Extraction



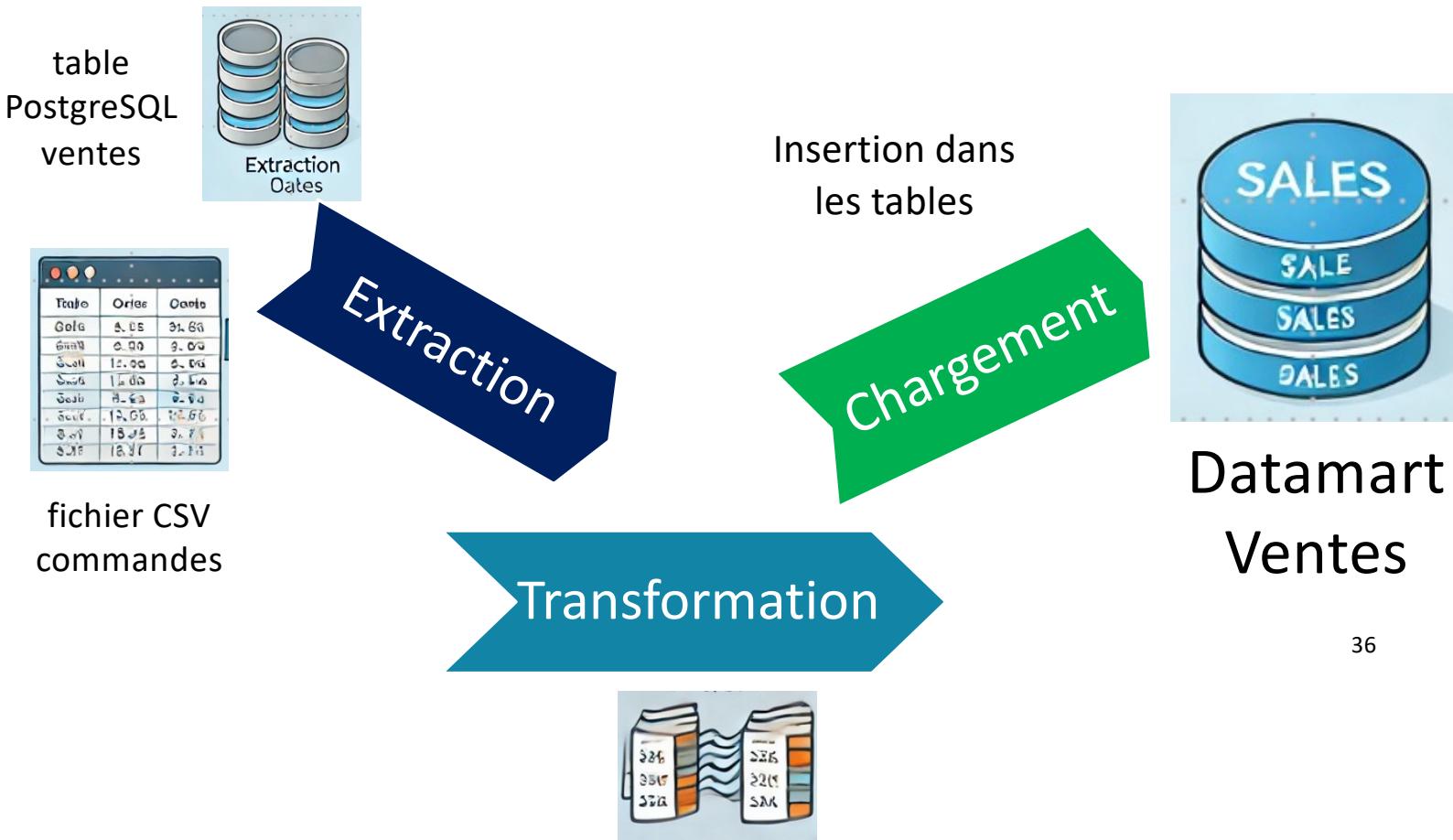
Transformation



Chargement



# Exemple de flux simple ETL pour alimenter un datamart de ventes



# Exemple de transformation table



table PostgreSQL  
ventes



id_vente	id_client	produit	quantite	prix_unitaire	date_vente
1	101	Livre	2	20.00	2024-01-15
2	102	Stylo	5	1.50	2024-01-16
3	103	Cahier	3	5.00	2024-01-17

- ✓ **Calcul du montant total\_vente**  
 $quantite * prix\_unitaire$
- ✓ **Formatage de la date**  
ajout de la colonne mois\_vente

id_vente	id_client	produit	quantite	prix_unitaire	total_vente	mois_vente
1	101	Livre	2	20.00	40.00	Janvier
2	102	Stylo	5	1.50	7.50	Janvier
3	103	Cahier	3	5.00	15.00	Janvier

# Exemple de transformation fichier csv

Produit	Quantité	Prix unitaire
Ordinateur	1	800.00
Clavier	2	50.00
Souris	3	45.00
Ecran	1	200.00
Clavier	1	25.00
Total	5	1075.00

fichier CSV commandes

id_commande	client_email	produit	quantite	prix_total	date_commande
1001	client1@mail.com	Ordinateur	1	800.00	2024-02-01
1002	client2@mail .com	Clavier	2	50.00	2024-02-02
1003	client3@mail.com	Souris	3	45.00	2024-02-02
1004	client1 @mail.com	Ecran	1	200.00	2024-02-03
1005	client2@mail..com	Clavier	1	25.00	2024-02-03



- ✓ **Formatage de la date :** Ajout d'une colonne trimestre
- ✓ **Nettoyage des emails :**
  - Suppression des espaces indésirables.
  - Correction des erreurs de format (..com)

id_commande	client_email	produit	quantite	prix_total	trimestre
1001	client1@mail.com	Ordinateur	1	800.00	T1 2024
1002	client2@mail.com	Clavier	2	50.00	T1 2024
1003	client3@mail.com	Souris	3	45.00	T1 2024
1004	client1@mail.com	Ecran	1	200.00	T1 2024
1005	client2@mail.com	Clavier	1	25.00	T1 2024

# Principaux outils ETL



- **Pentaho Data Integration (Kettle)**
  - Open source, facile à configurer.
  - Gestion visuelle des flux ETL
  - Prise en charge des bases de données, fichiers plats, et systèmes Big Data



- **Talend**
  - Outil ETL open source avec version entreprise payante
  - Connecteurs multiples pour bases de données, API, cloud, etc
  - Interface utilisateur intuitive avec des composants drag-and-drop

- **Informatica PowerCenter :**
  - Leader du marché ETL
  - Solutions pour grandes entreprises
  - Supporte les environnements cloud et sur site, très flexible pour des scénarios complexes



- **Apache Nifi :**
  - Open source, dédié au traitement en temps réel
  - Parfait pour gérer des flux de données continus et des environnements Big Data.

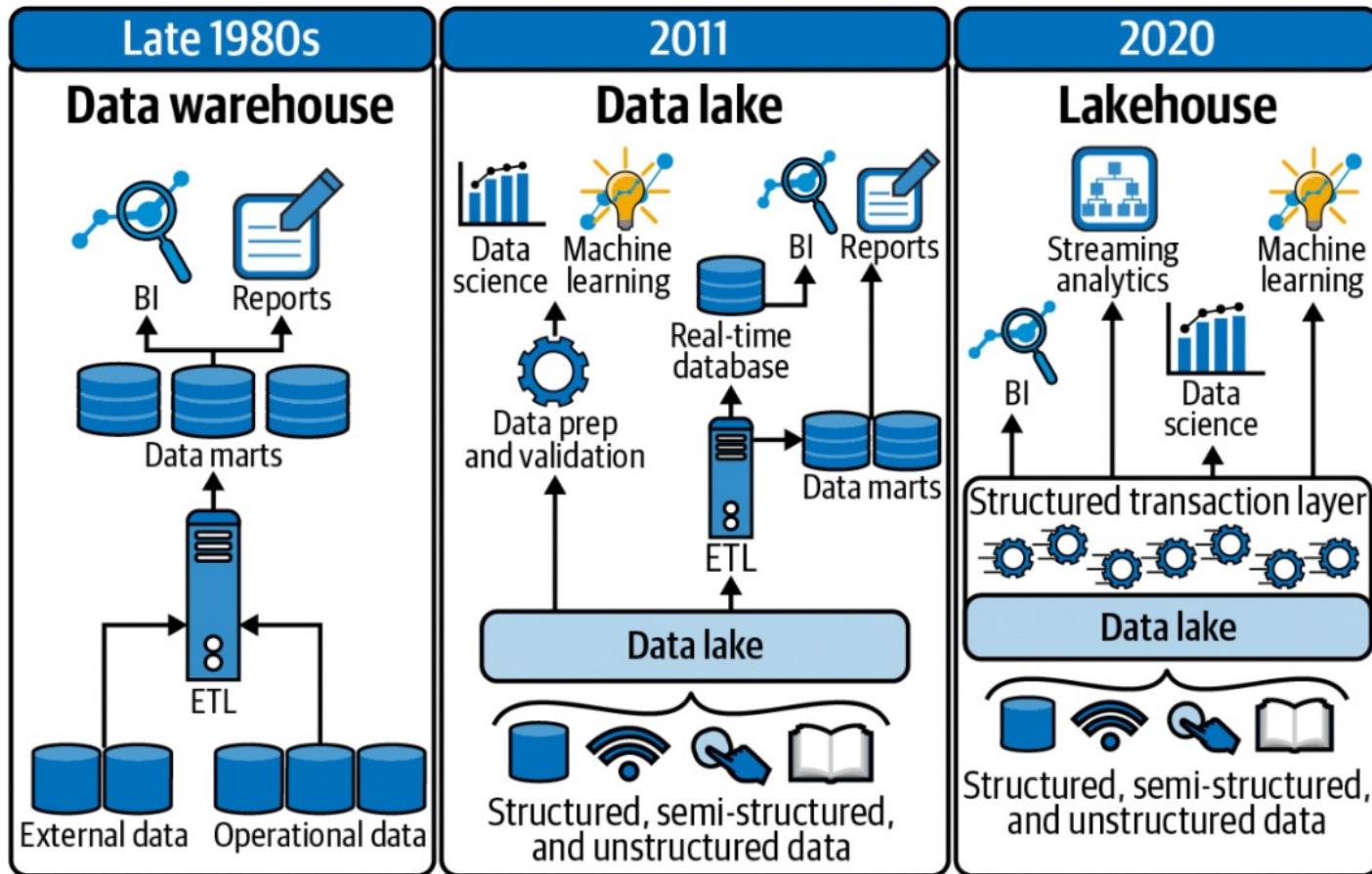


Notre choix dans ce cours

# 5 – Datalakes et DataLakeHouse



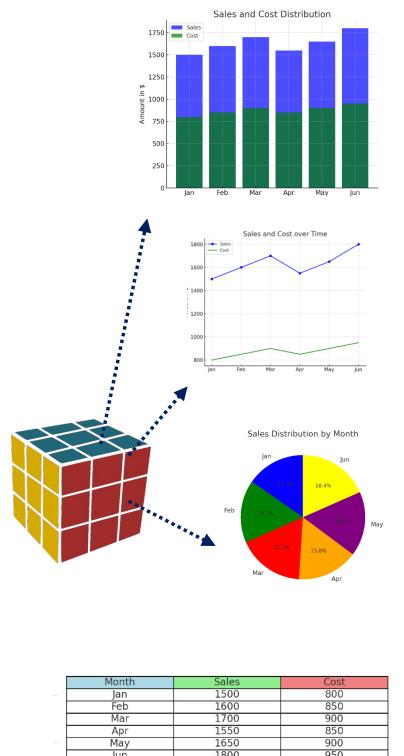
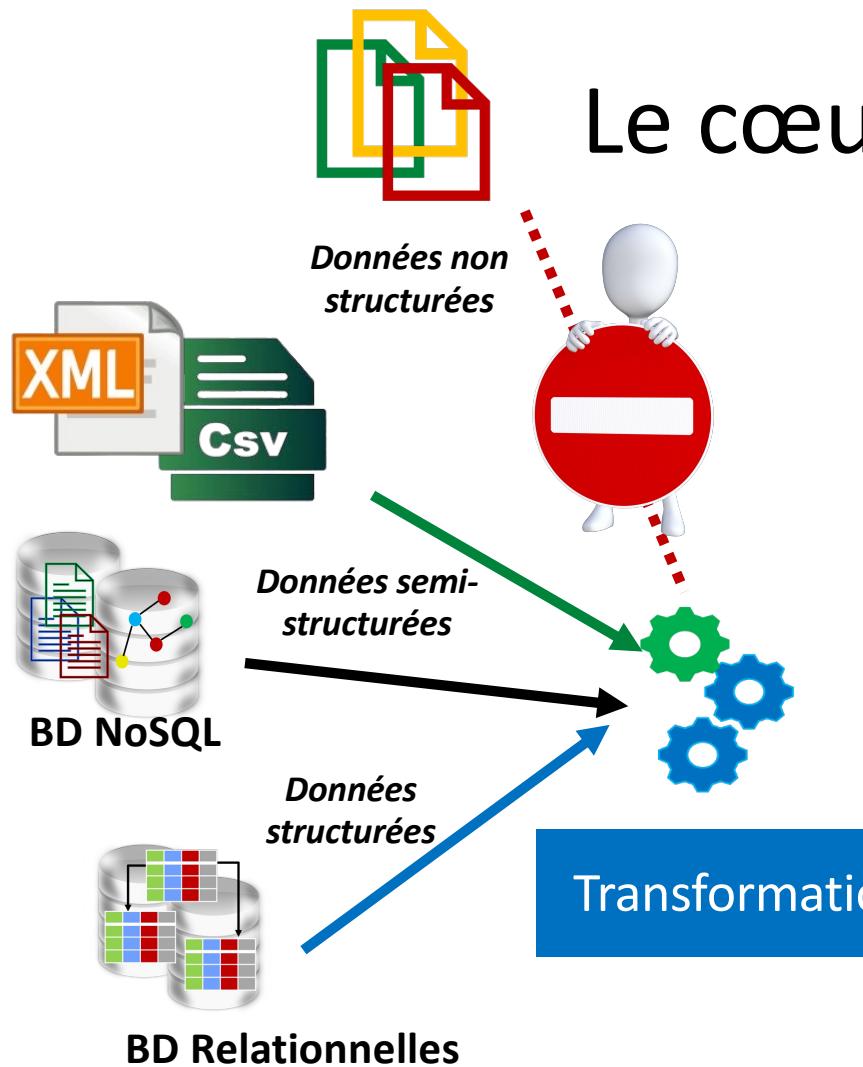
# Evolution : des DW au Data Lakehouses



source image : <https://www.linkedin.com/pulse/lakehouse-convergence-data-warehousing-science-dr-mahendra/>

# DataWareHouse

## Le cœur de l'analyse décisionnelle



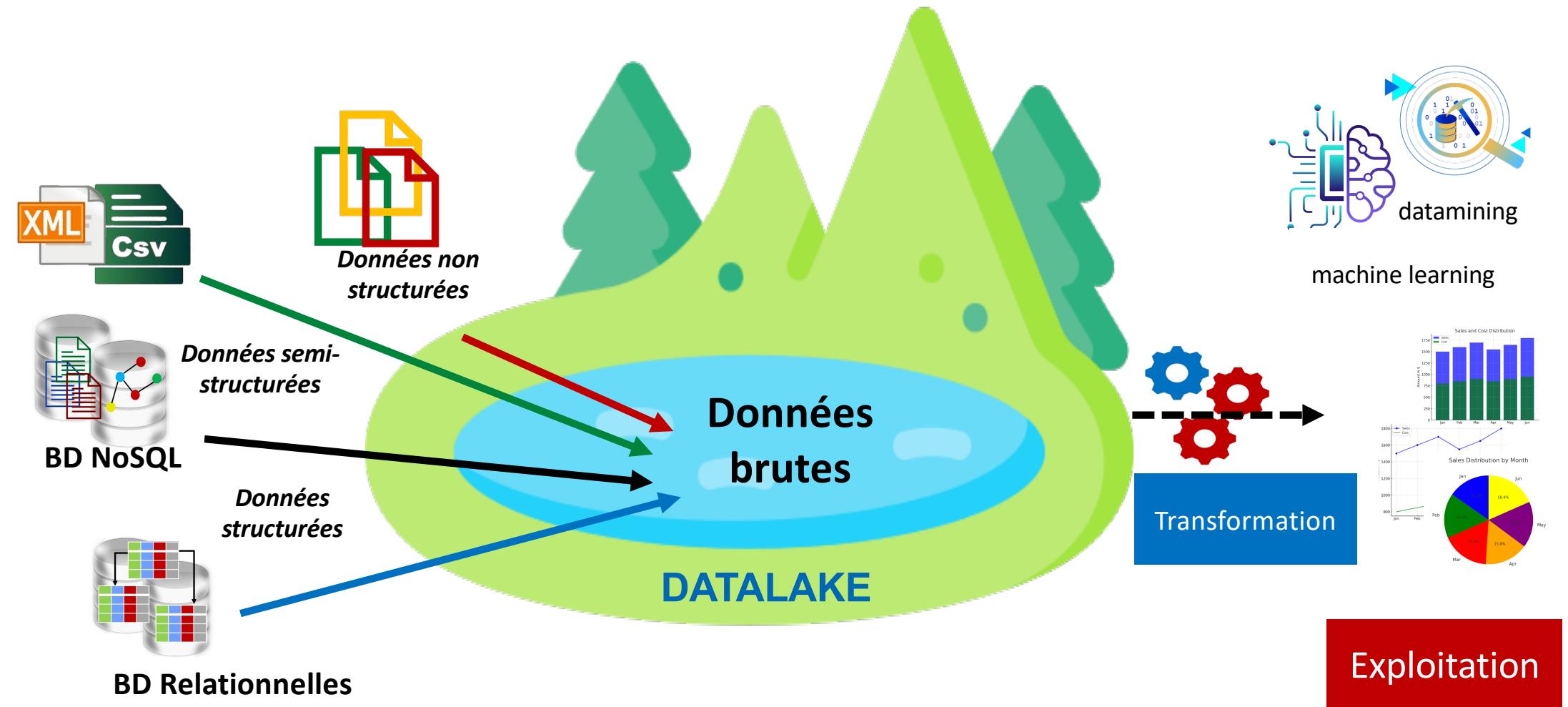
Exploitation

# Pourquoi les data warehouses ne suffisent plus à l'ère du big data ?

- Schéma rigide
- Réservé aux données structurées  
OU semi-structurées (après  
transformation)
- Connaissances préalables des  
besoins d'analyse



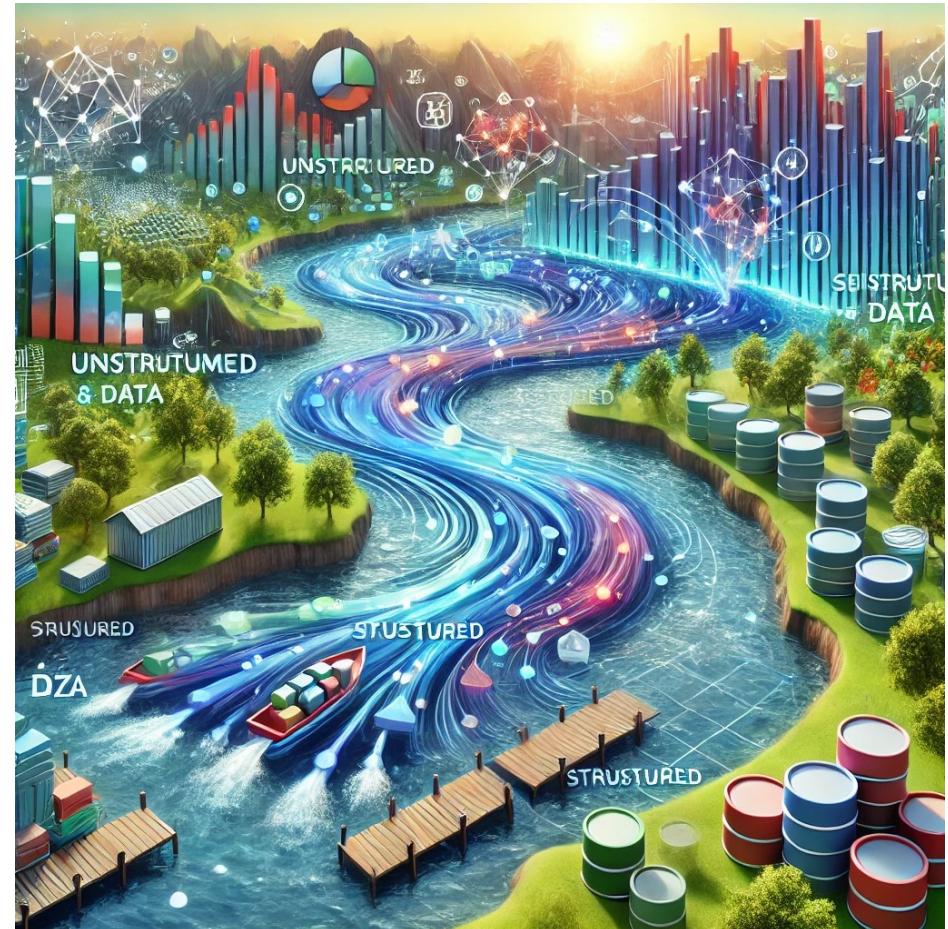
# DataLakes : Le grand saut vers la flexibilité



# Pourquoi les datalakes séduisent tant ?



- Tout type de données
  - Flexibilité
- Format Natif
  - Faible coût
- Grands volumes
- Adaptés à la data science et au machine learning



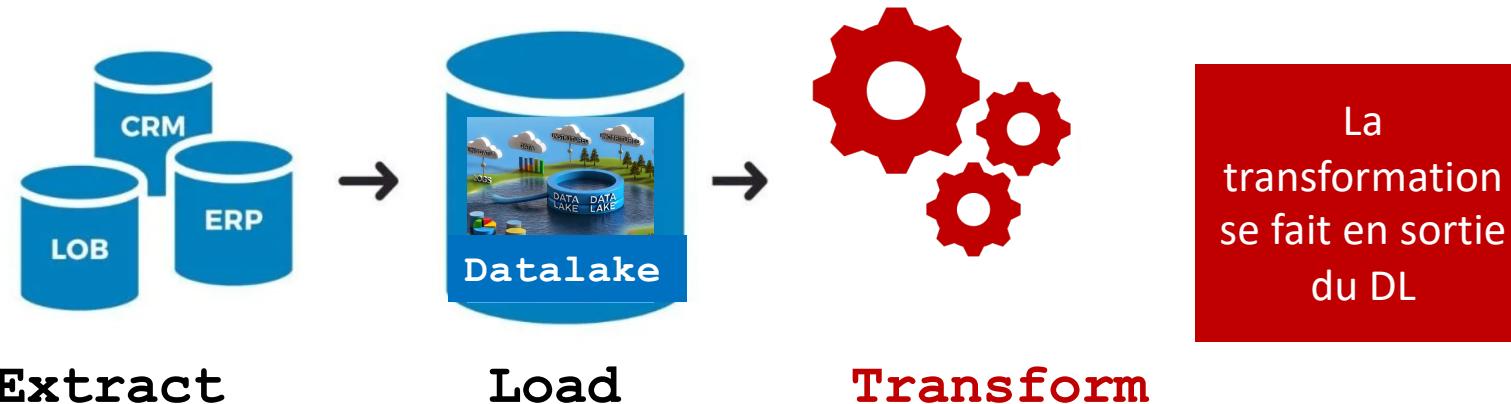
# DataLake : ELT versus ETL



**Extract**

**Transform**

**Load**



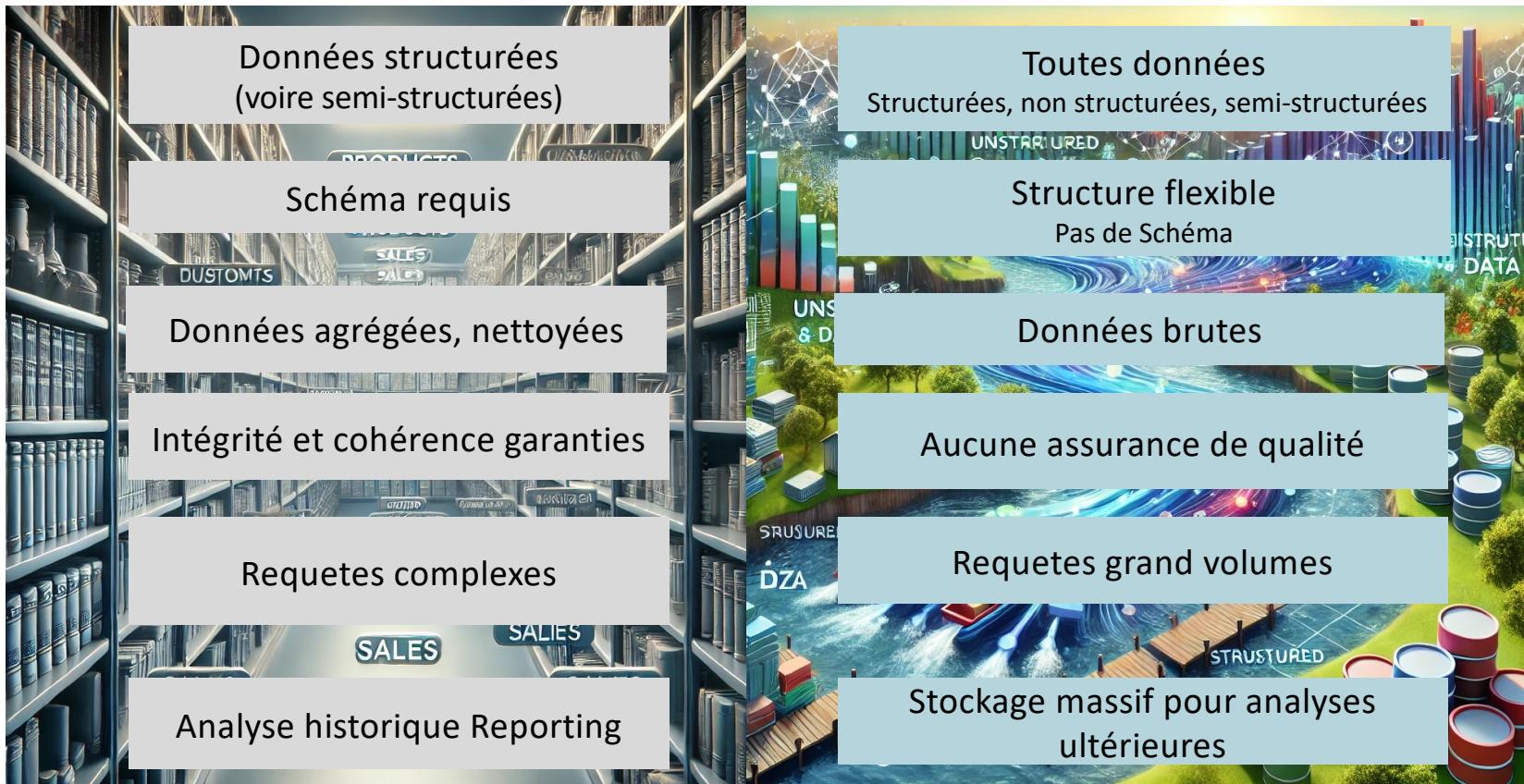
**Extract**

**Load**

**Transform**

source image : <https://blog.bismart.com/en/etl-or-elt-differences-use-cases>

# DataWareHouse versus DataLake

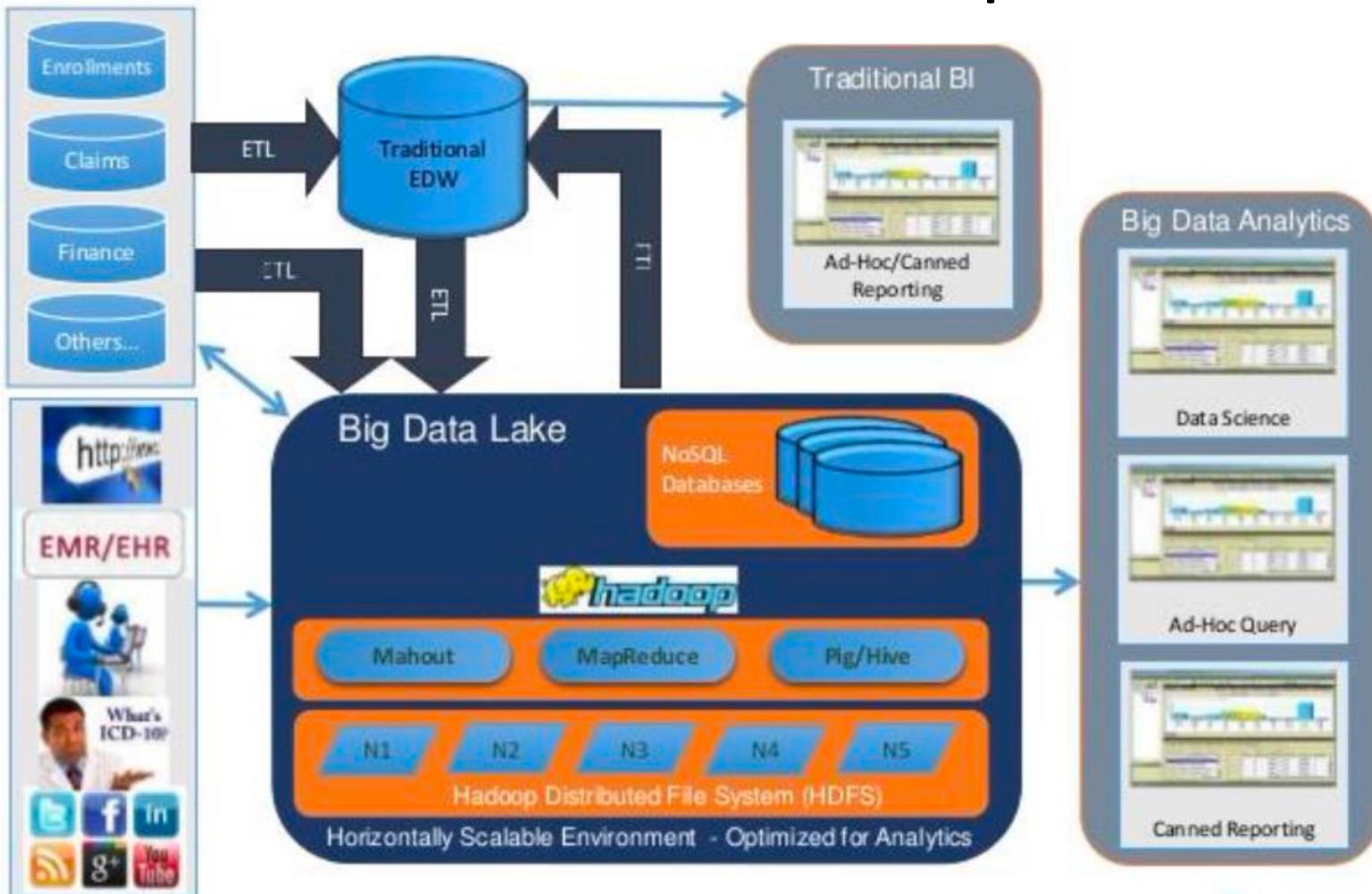


# Datalakes : Toute liberté a un prix ...

- Insuffisances de **Qualité** des données
- Problèmes de **Cohérence** des données
- **Difficultés** induites par l'absence de schémas
  - Limitations pour le machine learning  
Données brutes mal préparées
  - Complexité de l'analyse de données brutes  
Transformations complexes nécessaires



# DW et DL : Complémentaires !

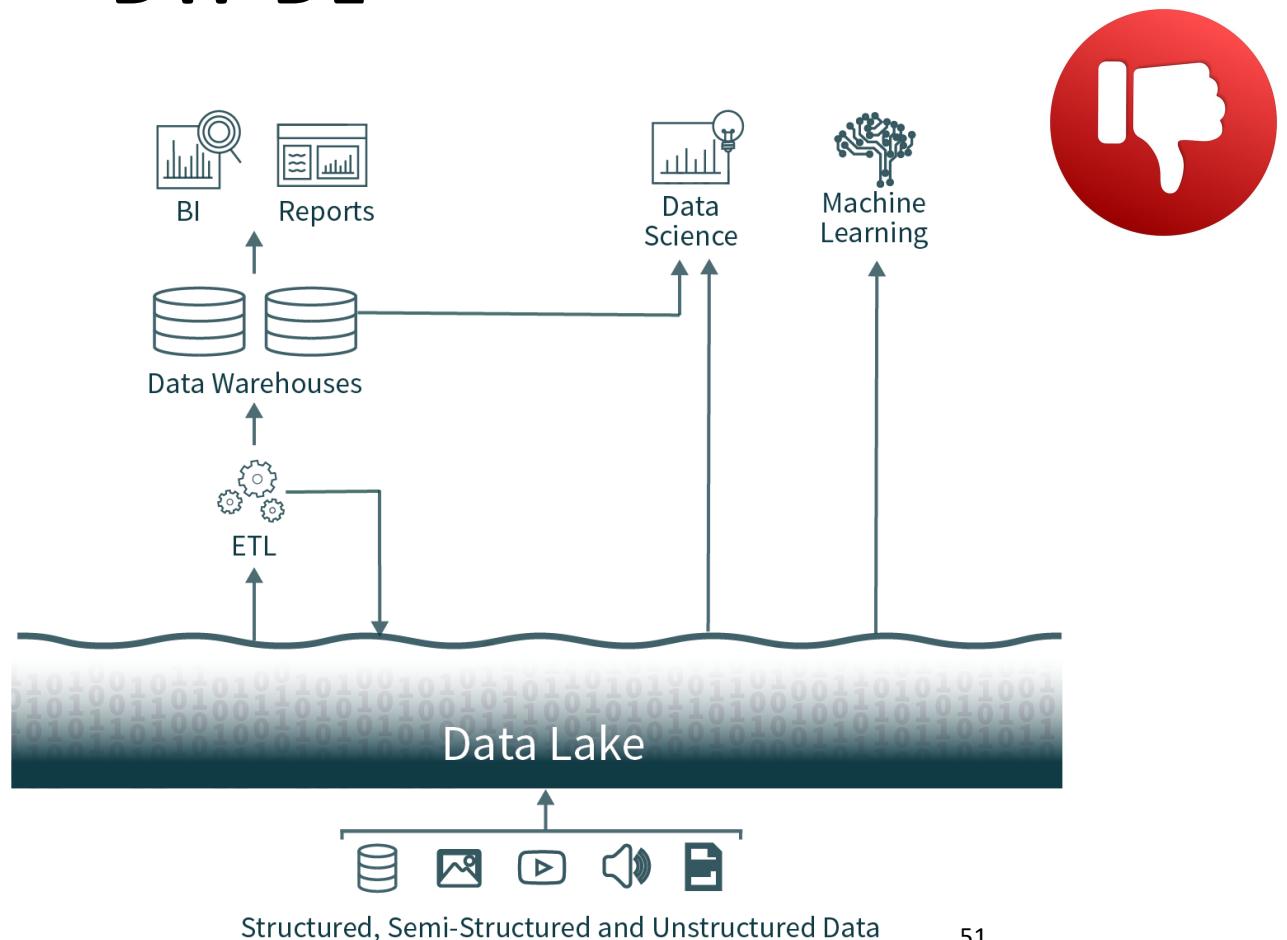


# Quel SGBD pour les DW et DL?

Critères	SGBD pour un DW	SGBD pour un Datalake
Modèle de données	<b>Relationnel</b> , modélisation en étoile ou flocon	Flexible : <b>NoSQL</b> , fichiers (JSON, Parquet, ORC), colonnes
Stockage	Structuré	Structuré, semi-structuré, non structuré
Exemples de SGBD	PostgreSQL, MySQL, Oracle, SQL Server, Snowflake	Hadoop (HDFS), Apache Hive, Apache HBase, Amazon S3
Performances	Optimisé pour les requêtes complexes et agrégations	Optimisé pour l'ingestion massive de données hétérogènes
Mode d'accès	OLAP (requêtes analytiques)	Accès brut ou traitement batch/stream (via Spark, Presto)
Échelle de données	Moins adapté aux pétaoctets, plus efficace pour les téraoctets	Évolutif horizontalement pour des volumes massifs
Langages de requêtes	SQL (standard ou analytique, ex. : OLAP SQL)	SQL-like (HiveQL), NoSQL, API pour frameworks (Spark, etc.)
Schéma	Schéma fixe (rigide, défini au préalable)	Schéma flexible (défini à la lecture ou schema-on-read)
Gestion des métadonnées	Centralisée, organisée (catalogues, dictionnaires)	Diversifiée, souvent gérée par des outils externes
Coût de maintenance	Élevé (optimisation, gestion des index, etc.)	Modéré (gestion des infrastructures et pipelines)
Cas d'usage typique	Reporting, tableaux de bord BI, analyses descriptives	Analyse exploratoire, Machine Learning, analyses Big Data

# Les contraintes des architectures à 2 niveaux DW-DL

- Multiples ETL
- Maintenances régulières très lourdes
- Données souvent obsolètes



source : <https://www.databricks.com/blog/2021/02/04/how-data-lakehouses-solve-common-issues-with-data-warehouses.html>

# L'étape suivante : Les Data LakeHouse

Une solution hybride pour combiner  
les avantages des DW et des DL



Cloud Service providers  
Stockage distribué



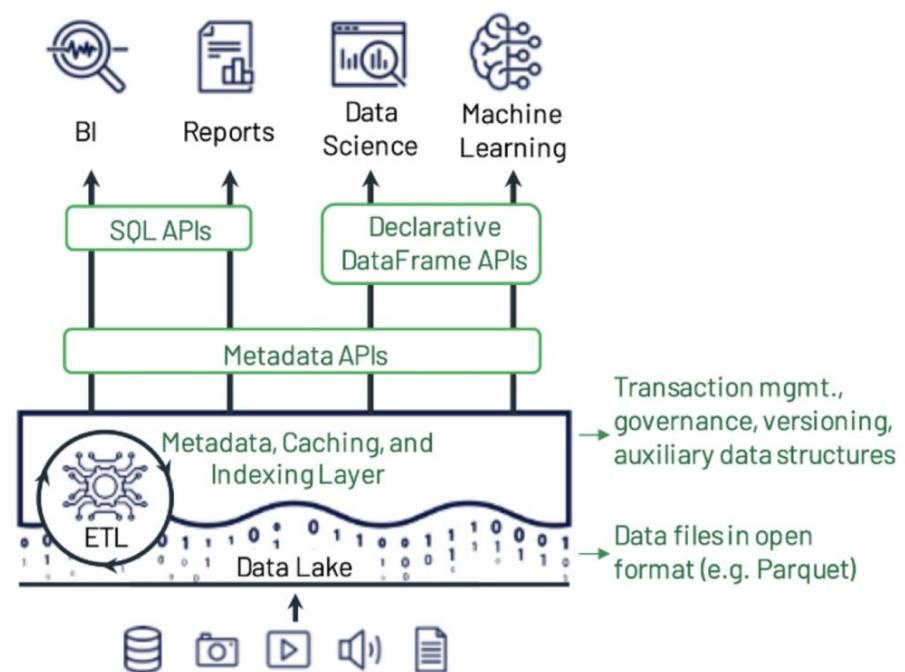
**Flexibilité et scalabilité**  
Données massives et  
variées en format natif

## Structure et performances

Données structurées  
pour des analyses  
complexes avec  
assurance d'intégrité

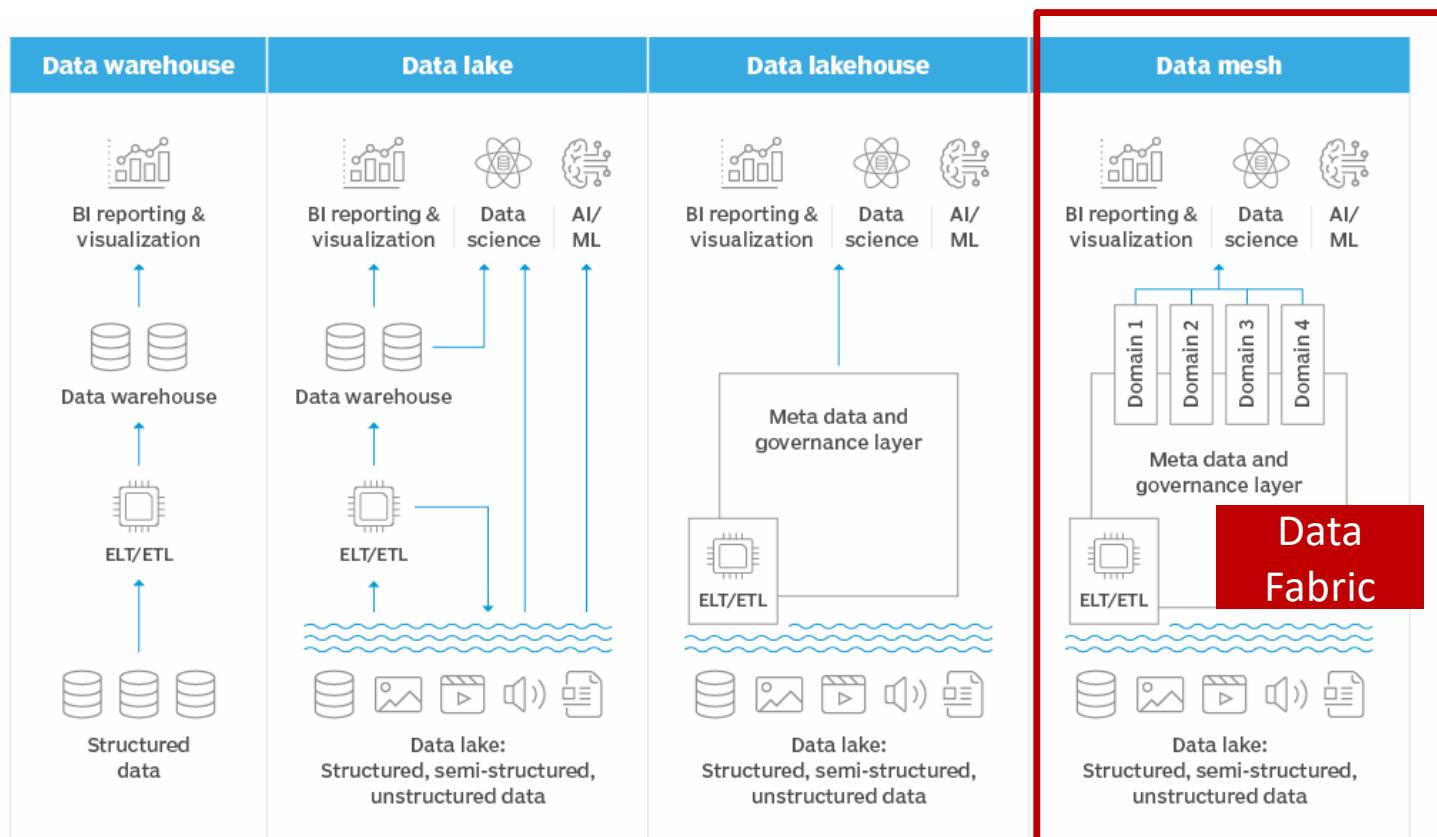
# Exemple d'architecture datalakehouse : Deltalake

- Une couche de métadonnées au dessus d'un datalake
- Ajout de transactions acid



source : <https://www.purestorage.com/fr/knowledge/what-is-delta-lake.html>

# Mais les architectures évoluent encore ...

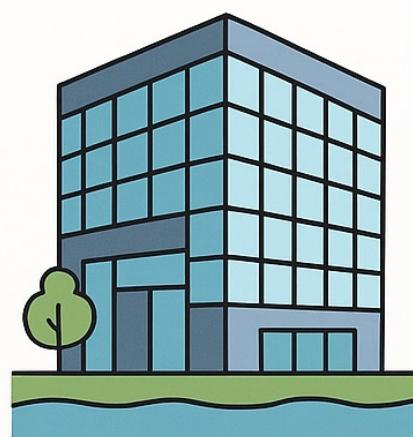


## Pourquoi passer au Data Mesh ?

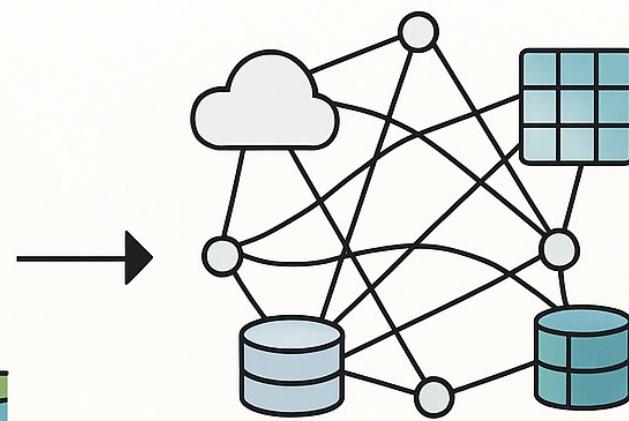
- Explosion des données et usages
- Saturation des équipes centrales
- Cloisonnement entre technique et métier
- Difficultés de gouvernance des données (sécurité, qualité, RGPD, ...)
- Besoin d'agilité et de responsabilisation



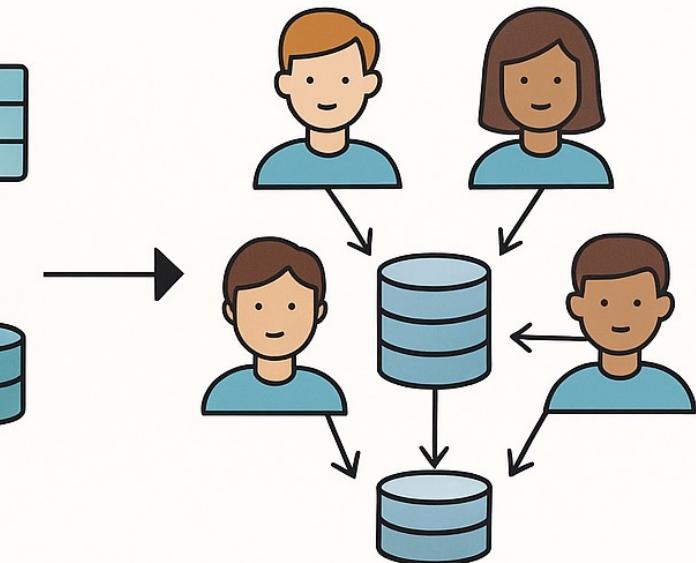
# De la centralisation à la fédération intelligente



**Data Lakehouse**  
cœur analytique  
stockage massif  
structuré et performant



**Data Fabric**  
couche d'intégration  
relie toutes les sources de  
données (cloud, applications  
lacs, entrepôts)



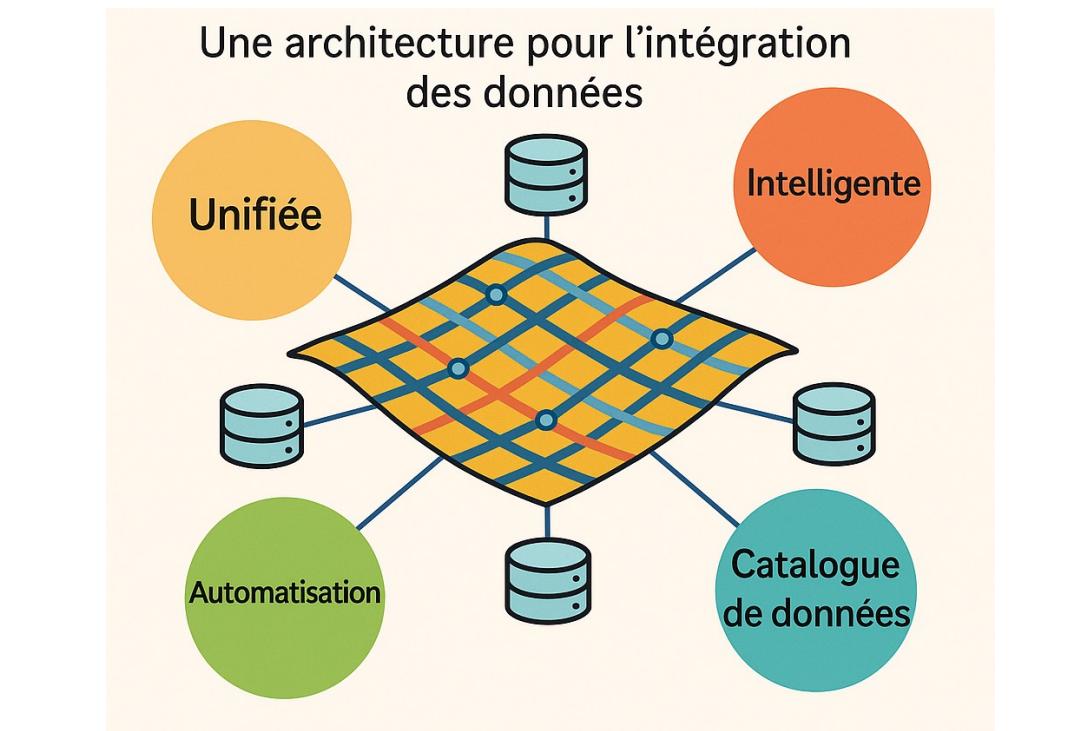
**Data Mesh**  
modèle organisationnel  
chaque domaine devient  
responsable de ses  
**"data products"**

Data Fabric apporte l'infrastructure technique unifiée

# Data Fabric : le réseau invisible qui relie toutes les données

- Connecte les sources dispersées sans les déplacer
- Gère automatiquement et optimise (en s'appuyant sur des outils d'IA) :
  - Les métadonnées globales
  - La qualité
  - La sécurité et la conformité RGPD

 Ex: *Un analyste RH peut accéder à des données logistiques sans copier de fichiers*

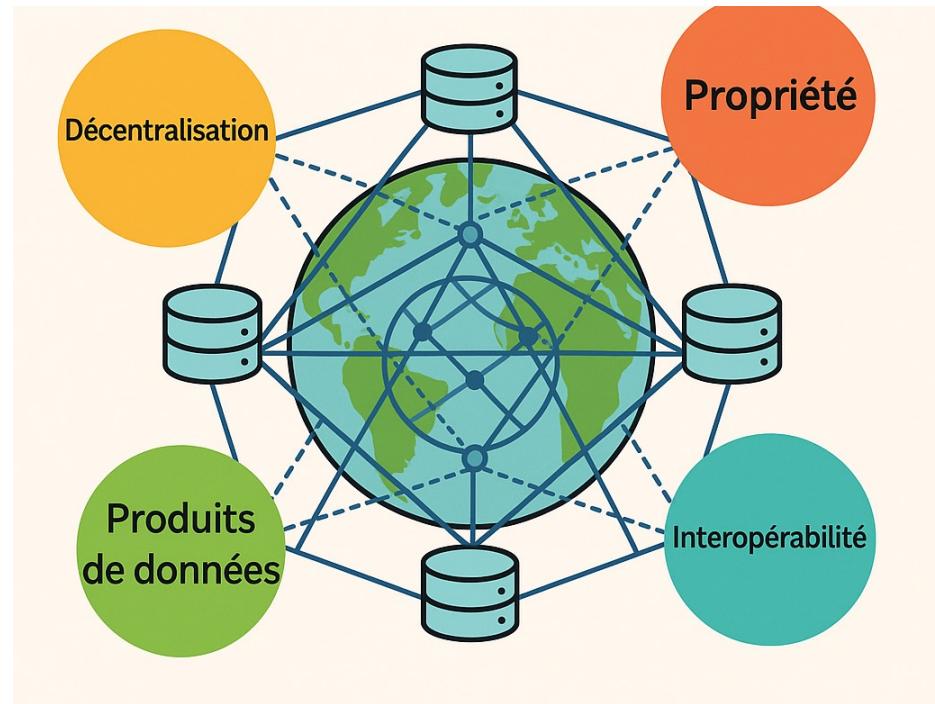


JUL 17 Le concept de *Data Fabric* a été introduit par Gartner en 2019 et est devenu opérationnel dans les grandes entreprises à partir de 2022–2023, avec des déploiements industriels portés par IBM, Informatica et Talend

# Les Data Mesh : chacun ses données !



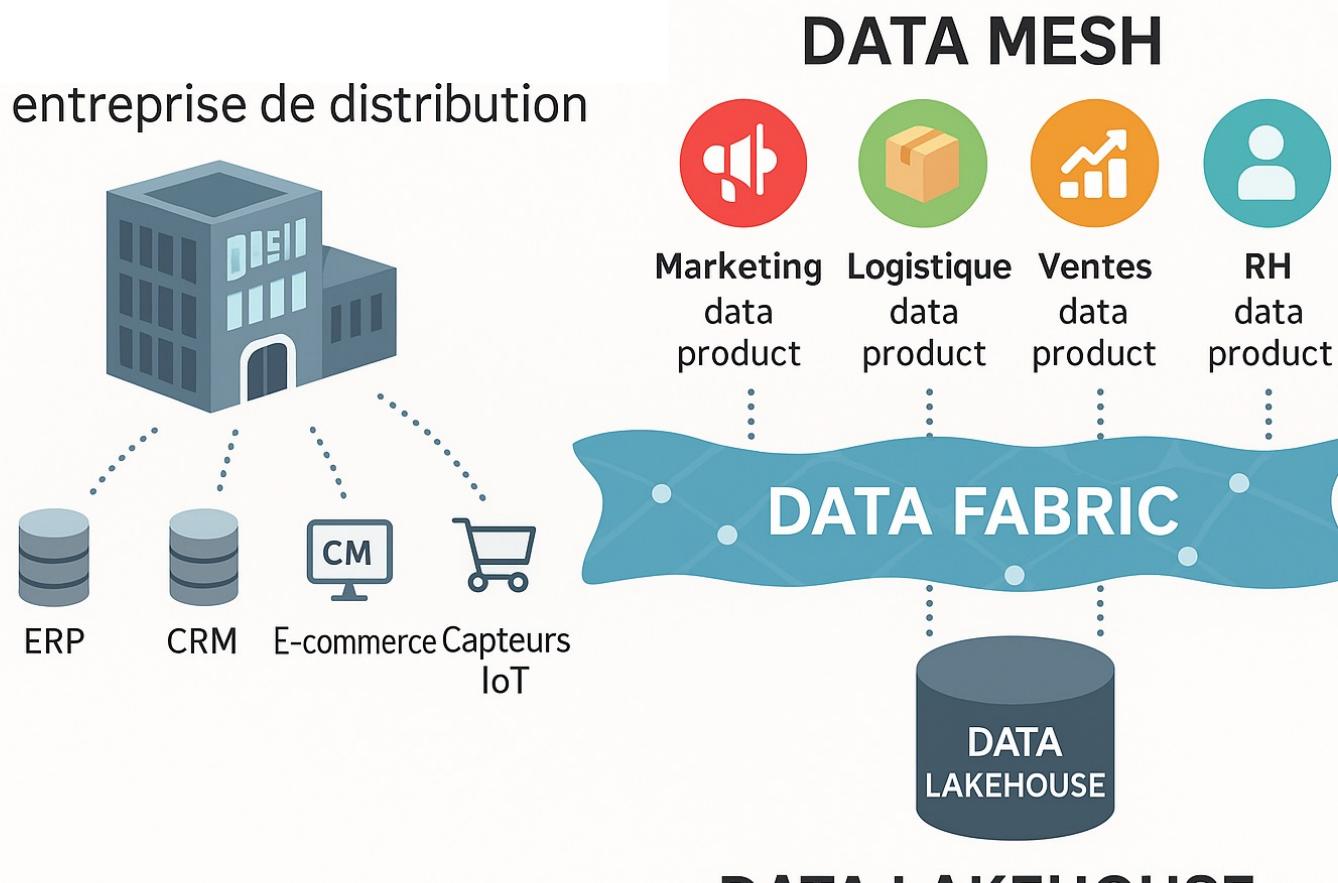
- Chaque Département devient propriétaire de ses données et les publie comme un **produit**
  - Le Data Fabric assure la circulation et la cohérence technique
  - Une équipe centrale définit les standards d'interopérabilité et de sécurité
- Ex: *Le service logistique publie ses données de livraison sous forme d'API interne, et le marketing les consomme directement pour prédire les ruptures de stock..*



## Avantages

Scalabilité organisationnelle  
Réactivité accrue  
Qualité et responsabilité locales renforcées

# Exemple d'architecture DataFabric/Datameshs



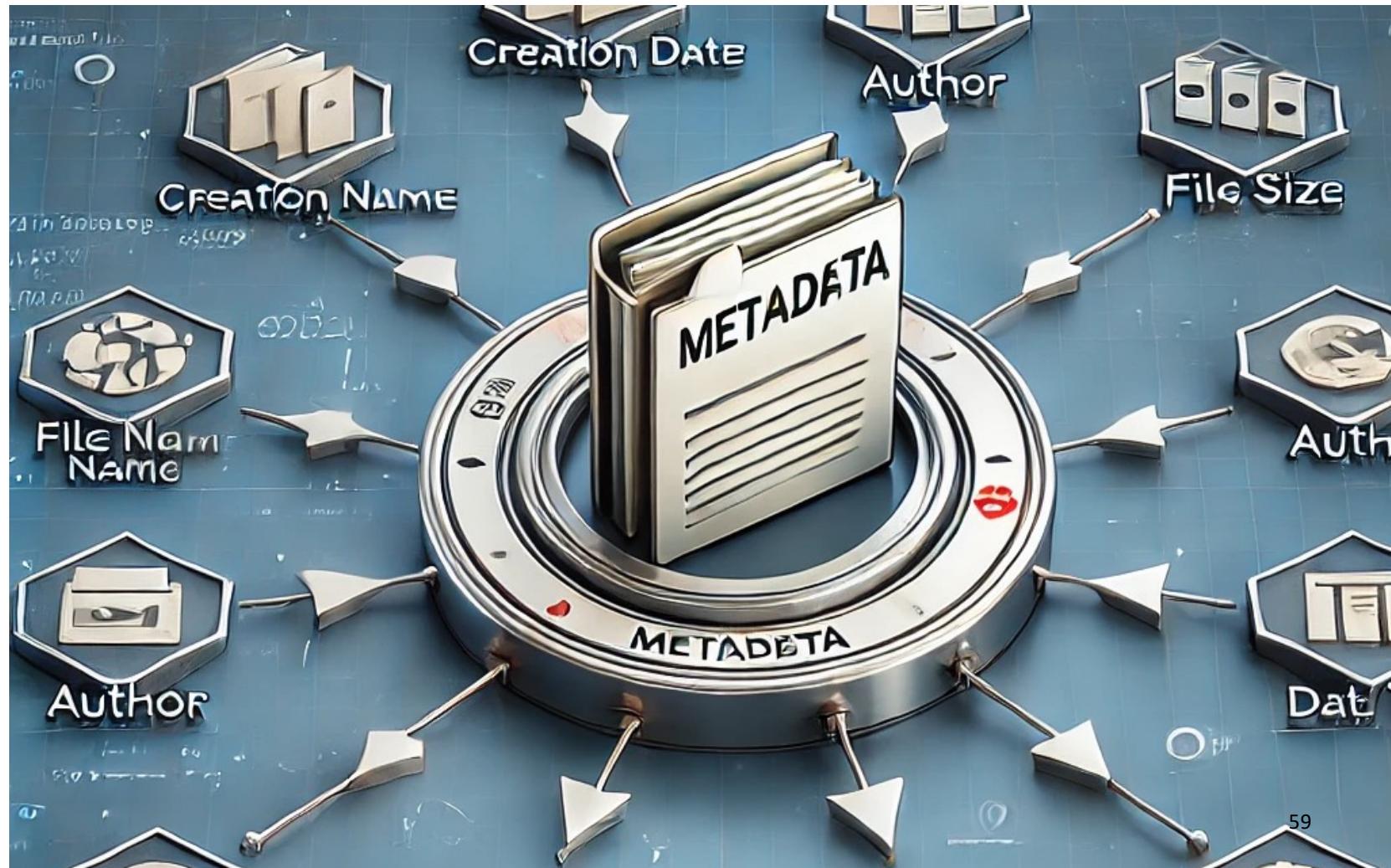
Les Data Mesh ne remplacent pas le Data Lakehouse ; ils changent la façon dont l'organisation pense et gère la donnée



Pour plus de détails :

<https://www.data-eclosion.com/fr/definition-data-mesh/>

# 6 – Importance des Métadonnées



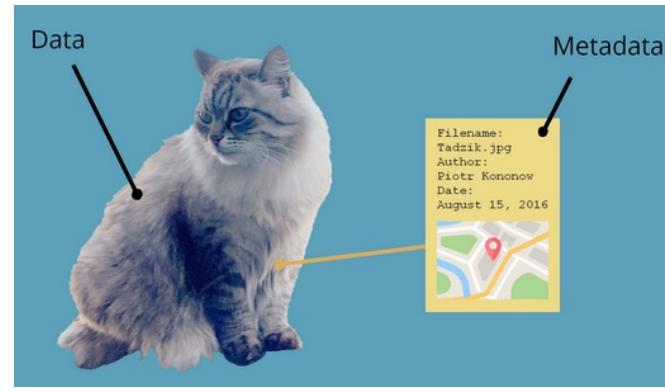
# Notion de métadonnées

- Données décrivant d'autres données.
  - Informations sur le contexte, la structure et le contenu des données
- Types de métadonnées
  - **Descriptives**

Décrivent le contenu  
(ex. *titre, auteur, date*)
  - **Structurelles**

Indiquent comment les données sont organisées  
(ex. *relations entre tables*)
  - **Administratives**

Fournissent des informations de gestion  
(ex. *version, droits d'accès*)



# Rôle des métadonnées dans les DW

- Gestion des sources
  - Identifient les origines des données  
(base relationnelle, fichier CSV, etc.)
- Données non structurées
  - Stockage de leurs métadonnées
- Transformation des données
  - Décrivent les règles de nettoyage, d'agrégation et de chargement (ETL)
- Structures des tables (faits, dimensions, indices)
- Optimisation des requêtes
  - Facilitent l'interprétation des données pour des outils d'analyse



- ✓ Gouvernance des données
- ✓ Compréhension des rapports analytiques
- ✓ Gestion efficace des processus ETL

# Métadonnées dans les Data Lakes

- Métadonnées **essentielles** car les DL stockent les données dans leur format brut
  - Description des données  
(type, origine, date de collecte)
- MD nécessaires pour :
  - Aider les utilisateurs à trouver et comprendre les données pertinentes
  - Faciliter les pipelines de transformation et d'analyse
  - Assurer une traçabilité des données pour la conformité réglementaire



# Métadonnées dans les DataLakehouses

- Combinaison des métadonnées des DW (structurées) et des DL (non structurées)
  - Un DLH gère des données hybrides (brutes et transformées)
- MD nécessaires pour :
  - Catalogue centralisé pour rechercher des données
  - Gestion des versions : suivi des modifications
  - Optimisation des performances
    - Partitionnement basé sur les métadonnées
    - Optimisation des requêtes grâce à des index et statistiques



# Conclusion



# Pourquoi étudier les DW à l'ère des DL et DLH?

- Les DW sont encore largement utilisés dans de nombreuses entreprises.
  - Lesdatalakes et datalakehouses ne remplacent pas totalement les data warehouses



Les architectures hybrides nécessitent des compétences en DW

# Pourquoi étudier les DW à l'ère des DL et DLH?

- Pour comprendre et exploiter les DL et DLH, plusieurs compétences clés sont indispensables :
  - savoir **organiser les données** en fonction des objectifs analytiques
  - savoir utiliser la **logique multidimensionnelle** pour explorer les données de manière interactive et découvrir des tendances ou des anomalies
  - comprendre la **structuration des données** : choisir entre données brutes ou structurées en fonction des besoins



# Pourquoi étudier les DW à l'ère des DL et DLH?

- Maîtriser les **concepts d'ETL** est essentiel dans le contexte des DL et DLH
  - comprendre le nettoyage, la transformation et l'intégration des données
- Développer une **approche critique** des architectures modernes nécessite des compétences en DW