



# Système d'information décisionnels

Master 1 : Data science



# Référentiels

- ❖ The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling – Ralph Kimball & Margy Ross
- ❖ Building the Data Warehouse – W. H. Inmon
- ❖ ETL with Azure Cookbook: Practical Recipes for Building Modern ETL Solutions – Christian Côté, José Mendes, Matt How
- ❖ Data Quality: The Accuracy Dimension – Jack Olson
- ❖ Business Intelligence Guidebook: From Data Integration to Analytics – Rick Sherman
- ❖ Multidimensional Databases and Data Warehousing – Christian S. Jensen, Torben Bach Pedersen, Christian Thomsen
- ❖ Storytelling with Data: A Data Visualization Guide for Business Professionals – Cole Nussbaumer Knaflic
- ❖ Information Dashboard Design – Stephen Few

# Organisation

- ❖ 6 séances de cours et travaux dirigés
  - A partir du 03 Février 2024
- ❖ 1 Projet en groupe
- ❖ 1 examen final

# Plan global

- ❖ **Introduction aux Systèmes d'Information Décisionnels**
- ❖ Architecture des Systèmes d'Information Décisionnels
- ❖ Modélisation multidimensionnelle
- ❖ Extraction, Transformation et Chargement
- ❖ Analyse et Exploration des Données
- ❖ Rappel : Data Mining et Machine Learning appliqué au SID
- ❖ Tendances et Innovations en SID

# Introduction aux SID : Définition

- ❖ Un **Système d'Information Décisionnel (SID)** est un ensemble de processus, technologies et outils permettant de **collecter**, **stocker**, transformation, **analyser** des données brutes issues de sources de données **et restituer** ces données afin d'aider les décideurs à prendre des décisions stratégiques et opérationnelles.
- ❖ Elle vise à :
  - Mesurer les **indicateurs** (mesure, faits ou métriques)
  - Restituer ces indicateurs selon des **axes** d'analyse (ou **dimensions**)
- ❖ **Composants d'un SID** : Un SID repose sur plusieurs **composants clés** :
  - **Entrepôt de données (Data Warehouse)** → Stockage centralisé des données historiques.
  - **ETL (Extract, Transform, Load)** → Processus de transformation et d'intégration des données.
  - **OLAP (Online Analytical Processing)** → Exploration multidimensionnelle des données.
  - **Reporting et BI (Business Intelligence)** → Visualisation et analyse des données.
  - **Data Mining et Machine Learning** → Analyse prédictive et détection de tendances.

# Caractéristiques d'un SID

- ❖ Agrégation et historisation des données
- ❖ Optimisation pour l'analyse (plutôt que les transactions courantes)
- ❖ Prise en charge des volumes massifs de données
- ❖ Fourniture de tableaux de bord et d'indicateurs de performance (KPIs)

## **KPI : Key Performance Indicator**

- ❖ ICP : Indicateur Clé de Performance
- ❖ Mesure (ou ensemble de mesures) centrée sur un aspect critique de la performance globale de l'organisation
  - Financier ROI
  - Externe (sur les clients/fournisseurs) ou interne (sur un service)
- ❖ Exemples : Satisfaction client, taux d'attrition (churn), taux de conversion, qualité de service, qualité de fabrication

# Rôles et importances des SID

Les SID permettent de :

- ❖ **Centraliser les données** provenant de multiples sources (ERP, CRM, bases transactionnelles, fichiers logs, etc.).
- ❖ **Fournir une vision cohérente** et intégrée de l'activité d'une organisation.
- ❖ **Analyser et explorer les données** via des tableaux de bord interactifs et des outils OLAP.
- ❖ **Améliorer la prise de décision** en proposant des tendances, prévisions et insights stratégiques.
- ❖ **Faciliter la planification et l'optimisation** des processus métier.

## Importance des SID

- ❖ **Compétitivité accrue** : Les entreprises peuvent prendre des décisions basées sur des données factuelles plutôt que sur de simples intuitions.
- ❖ **Réduction des coûts** : Identification des inefficacités et optimisation des ressources.
- ❖ **Personnalisation des services** : Grâce à l'analyse de données, les organisations peuvent adapter leurs offres aux besoins des clients.
- ❖ **Conformité réglementaire** : Gestion et traçabilité des données pour respecter les normes comme APDP.

# Origine et évolution des SID

## ❖ Naissance des SID (années 1970-1980)

- Les **premiers entrepôts de données** sont créés pour centraliser les informations des entreprises.
- Apparition des **premiers outils de reporting** basés sur les bases de données relationnelles.

## ❖ Développement de la Business Intelligence (années 1990)

- Ralph **Kimball** et Bill **Inmon** définissent les **modèles de Data Warehousing**.
- Apparition des **cubes OLAP** pour l'analyse multidimensionnelle. Deux nouveaux concepts :
  - **Faits** : mesures définies ) partir des données
  - **Dimensions** : axes d'analyses des faits
  - **4 étapes dans la modélisation** :
    - Choix du processus à modéliser
    - Définition de la granularité du processus
    - Choix des dimensions
    - Identification des faits.
- Développement des outils de **Reporting et de BI** (ex. BusinessObjects, Cognos).



# Origine et évolution des SID

## ❖ Évolution vers le Big Data et le Cloud (années 2000-2010)

- L'augmentation des **volumes de données** pousse au développement de solutions **scalables**.
- L'essor du **Big Data** et des **Data Lakes** avec Hadoop et Spark.
- Les solutions **Cloud (AWS Redshift, Google BigQuery, Snowflake)** remplacent les entrepôts traditionnels.

## ❖ L'ère de l'IA et de l'analyse prédictive (2015-présent)

- Intégration du **Machine Learning** et de l'**IA** pour la prédiction.
- Concepts modernes comme **le Data Mesh et le Data Fabric** émergent.
- Développement d'outils de **BI self-service** (Power BI, Tableau).

## ❖ **Tendance actuelle** : les SID deviennent plus flexibles, automatisés et interconnectés avec l'IA et le Cloud.

# Centralisation des données de l'entreprise 1/2

- ❖ **Problème des données dispersées** : Les entreprises utilisent de nombreux systèmes différents pour gérer leurs opérations :
  - **ERP** pour la gestion financière et des ressources
  - **CRM** pour le suivi des clients
  - **Systèmes transactionnels** pour la gestion des commandes
  - **Logs et capteurs IoT** pour la surveillance des processus
- ❖ **Sans un SID**, les données sont stockées de manière disparate, rendant leur exploitation difficile.
- ❖ **Comment un SID centralise les données ?**
  - Il intègre les données de plusieurs sources via des processus **ETL (Extract, Transform, Load)**
  - Il les stocke dans un **Data Warehouse** ou un **Data Lake**
  - Il garantit une **vision unique et cohérente des données**

# Centralisation des données de l'entreprise 2/2

## ❖ **Avantages de la centralisation des données**

- **Amélioration de la qualité des données** : Éviter les incohérences et redondances
- **Gain de temps** : Les analystes n'ont plus besoin de chercher les données dans plusieurs systèmes
- **Sécurité et gouvernance** : Meilleure gestion des accès et conformité réglementaire (ex : APDP)

## ❖ **Exemple** : Une multinationale unifie ses données financières, RH et marketing dans un SID, permettant une vision globale des performances.

# Aide à la prise de décision basée sur les données

## ❖ **Décision basée sur les données vs Intuition**

- **Décision intuitive** : Basée sur l'expérience et les suppositions
- **Décision basée sur les données** : Appuyée sur des analyses objectives et des KPIs

## ❖ **Un SID** permet aux entreprises de prendre des décisions éclairées en s'appuyant sur des faits et non sur des suppositions.

## ❖ **Processus de prise de décision avec un SID**




- **Collecte des données** : Récupération des données à partir de différentes sources
- **Transformation et stockage** : Nettoyage et organisation des données dans un entrepôt de données
- **Analyse et modélisation** : Exploration des données avec des outils BI et OLAP
- **Visualisation et interprétation** : Tableaux de bord et rapports pour guider les décisions

## ❖ **Exemples**

- **Optimisation des stocks** : Ajuster ses commandes en fonction des ventes historiques
- **Segmentation client** : Personnalisation des offres en fonction du comportement de ses clients
- **Gestion des risques** : Une compagnie d'assurance évalue les risques grâce à l'analyse des données des sinistres
- Un hôpital utilise un SID pour analyser les admissions et optimiser la gestion des lits.

# Production de rapports et d'analyses avancées

## ❖ Rapports générés par un SID

-  **Rapports opérationnels** : Suivi des KPI quotidiens (ventes, performances)
-  **Rapports analytiques** : Étude approfondie des tendances sur plusieurs mois
-  **Rapports prévisionnels** : Projection des performances futures

## ❖ Technologies utilisées

- **BI (Business Intelligence)** : Power BI, Tableau, Apache Superset
- **OLAP (Online Analytical Processing)** : Analyse multidimensionnelle
- **SQL analytique** : Agrégation, groupements et analyses temporelles

## ❖ Exemples

- **Tableaux de bord financiers** : Suivi en temps réel des revenus et des coûts
- **Rapports de performance RH** : Analyse du turnover et de la productivité des employés
- **Visualisation des tendances marketing** : Évaluation des campagnes publicitaires
- Un groupe hôtelier utilise un SID pour visualiser l'occupation des chambres en temps réel et ajuster ses prix.

# Identification des tendances et prévisions

## ❖ Pourquoi analyser les tendances ?

- **Détecter des opportunités** (ex : nouveaux marchés, segments de clients)
- **Anticiper les évolutions du marché** (ex : montée en puissance d'un produit)
- **Prévenir les risques** (ex : baisse des ventes, churn client)

## ❖ Outils et techniques pour l'identification des tendances

- **Data Mining** : Extraction de patterns cachés dans les données
- **Machine Learning** : Prédiction basée sur l'apprentissage des modèles
- **Séries temporelles** : Analyse des tendances sur le temps (ex : prévision de ventes)

## ❖ Exemples

- **Prévisions des ventes** : Une chaîne de supermarchés ajuste ses stocks en fonction des tendances saisonnières
- **Détection des fraudes** : Une banque identifie des comportements suspects grâce à l'analyse des transactions
- **Prédiction du taux de churn** : Une entreprise anticipe les clients à risque de résiliation
- Netflix utilise un SID avancé pour recommander des films en fonction des préférences et des tendances de visionnage.

# Concepts fondamentaux : Données, Décisions, SI

Un **Système d'Information Décisionnel** repose sur trois piliers :

## ❖ Les Données

- **Définition** : Ensemble d'informations collectées à partir de diverses sources.
- **Sources de données** : ERP, CRM, logs, bases transactionnelles, IoT, Open Data.
- **Types de données** :
  - **Données structurées** (bases SQL, fichiers CSV).
  - **Données semi-structurées** (JSON, XML).
  - **Données non structurées** (vidéos, images, emails).

## ❖ La Décision

- Prise de décision basée sur les données : décision descriptive, diagnostique, prédictive et prescriptive.
- **Exemples** :
  - Un retailer ajuste ses stocks en fonction des tendances de vente.
  - Une banque détecte des fraudes grâce à l'analyse des transactions.

# Concepts fondamentaux : Données, Décisions, SI

## ❖ Le Système d'Information

➤ **Définition** : Un ensemble organisé de ressources permettant de traiter l'information.

➤ **Composants** :

■ **Matériel** : Serveurs, bases de données.

■ **Logiciel** : Outils BI, bases de données, ETL.

■ **Réseaux** : Cloud, Data Centers.

■ **Utilisateurs** : Analystes, Data Engineers, décideurs.

❖ Un **SID** combine ces trois éléments pour fournir une intelligence métier efficace.



# Définition des SI transactionnels OLTP

- ❖ Un **SI Transactionnel (OLTP - OnLine Transaction Processing)** est un système informatique conçu pour **gérer les transactions courantes** d'une entreprise en temps réel.
- ❖ Il est utilisé pour stocker, mettre à jour et récupérer rapidement des informations transactionnelles.

## Caractéristiques principales d'un OLTP

- ❖ **Traitement en temps réel** : Répond aux requêtes instantanément (millisecondes à secondes)
- ❖ **Manipulation de données détaillées et opérationnelles** (ventes, commandes, paiements, stocks...)
- ❖ **Forte fréquence des transactions** (ajout, mise à jour, suppression)
- ❖ **Maintien de la cohérence des données** via des **transactions ACID** (Atomicité, Cohérence, Isolation, Durabilité)
- ❖ **Stockage normalisé** pour minimiser la redondance et optimiser l'espace
- ❖ **Exemples typiques de transactions OLTP**
  - **Effectuer un achat en ligne** : Mise à jour du stock en temps réel
  - **Effectuer un virement bancaire** : Débit/crédit instantané des comptes
  - **Réserver une chambre d'hôtel** : Vérification de la disponibilité et confirmation immédiate
  - **Créer un ticket de support client** : Enregistrement instantané dans un CRM

# Objectifs et fonctionnement des SI transactionnels

## Objectifs principaux des SI transactionnels

- ❖ Assurer une gestion efficace des opérations courantes
- ❖ Garantir l'intégrité et la fiabilité des données
- ❖ Optimiser la rapidité des transactions
- ❖ Permettre la scalabilité pour un grand nombre d'utilisateurs simultanés

## Fonctionnement d'un OLTP

- ❖ **Saisie des données** : L'utilisateur effectue une transaction (achat, enregistrement d'un client, mise à jour d'une commande).
- ❖ **Traitement de la transaction** : Le système vérifie l'intégrité des données et exécute la transaction en respectant les règles métiers.
- ❖ **Mise à jour de la base de données** : La transaction est stockée et validée immédiatement via les principes **ACID**.
- ❖ **Restitution des données** : L'utilisateur reçoit une confirmation immédiate de la transaction.
- ❖ **Un OLTP** repose sur une base de données relationnelle optimisée pour les requêtes rapides et fréquentes.

# SI transactionnel vs SI décisionnel

Critère	OLTP (Systèmes Transactionnels)	OLAP (Systèmes Décisionnels)
Objectif	Gérer les transactions opérationnelles(commandes, paiements, gestion RH, etc.)	Analyser les données pour la prise de décision
Type de données	Opérationnelles, détaillées et récentes(ex: achat client)	Agrégées et historiques(ex: tendances de ventes)
Optimisation	Rapide pour INSERT, UPDATE, DELETE	Rapide pour SELECT et agrégations
Volume	Faible à moyen (ex: 10 millions de lignes)	Élevé (ex: plusieurs milliards de lignes)
Structure	Normalisée (évite la redondance)	Dénormalisée (optimisée pour l'analyse)
Temps de réponse	Très rapide (millisecondes)	Rapide mais dépend du volume de données
Stockage	Données brutes et détaillées	Données agrégées et historisées
Exemples	ERP, CRM, gestion des commandes, Systèmes de facturation, bases clients	Data Warehouse, Business Intelligence, Tableaux de bord, rapports de performance

# SI transactionnels : ERP et CRM

- ❖ **ERP (Enterprise Resource Planning)** : Les **ERP** sont des systèmes intégrés qui permettent de **gérer les processus métier** d'une organisation en temps réel.
  - **Exemples de solutions ERP :**
    - **SAP ERP** – Gestion des ressources humaines, finances, logistique
    - **Oracle ERP Cloud** – Gestion financière et supply chain
    - **Microsoft Dynamics 365** – Gestion de la relation client et des opérations
  - **Cas d'usage** : Une entreprise utilise SAP pour **gérer les commandes clients, les stocks et la comptabilité** en temps réel.
- ❖ **CRM (Customer Relationship Management)** : Les **CRM** sont des systèmes permettant de **gérer les interactions avec les clients** et d'optimiser la relation commerciale.
  - **Exemples de solutions CRM :**
    - **Salesforce** – Gestion des ventes et des leads
    - **HubSpot CRM** – Marketing automation et service client
    - **Microsoft Dynamics CRM** – Gestion de la relation client pour les grandes entreprises
  - **Cas d'usage** : Une entreprise utilise Salesforce pour **suivre les opportunités commerciales et automatiser le service client**.

# SI transactionnels : BD relationnelles classiques

- ❖ **Bases de données relationnelles classiques** : Les systèmes OLTP sont souvent construits sur des bases relationnelles **optimisées pour les transactions rapides**.
  - **Exemples de bases de données OLTP** :
  - **MySQL, PostgreSQL** – Bases open-source performantes
  - **Microsoft SQL Server** – Base d'entreprise pour les ERP/CRM
  - **Oracle Database** – Solution robuste pour les grandes entreprises
  - **Cas d'usage** :
    - Un site e-commerce utilise **MySQL** pour **stocker les commandes clients et gérer les paiements en temps réel**.

# Définition des SI Décisionnels OLAP

- ❖ Un **Système d'Information Décisionnel (OLAP - OnLine Analytical Processing)** est un système conçu pour **analyser de grandes quantités de données**, permettant aux entreprises de prendre des **décisions stratégiques** basées sur des tendances et des modèles.
- ❖ Contrairement aux SI transactionnels (**OLTP**), les SI décisionnels sont optimisés pour la lecture et l'analyse des données, plutôt que pour les transactions courantes.

## Caractéristiques principales d'un OLAP

- ❖ **Traitement analytique multidimensionnel** : Exploration des données sous différents angles (temps, produit, région...).
- ❖ **Agrégation des données** : Stockage des informations sous forme **historisée et agrégée**.
- ❖ **Optimisation pour l'analyse** : Temps de réponse rapide pour des requêtes complexes.
- ❖ **Utilisation de cubes OLAP et de bases dénormalisées** : Accélération des calculs analytiques.
- ❖ **Exemples typiques de requêtes OLAP**
  - **Analyse des ventes par région et période** : Quel produit s'est le mieux vendu au cours des 6 derniers mois ?
  - **Étude de la rentabilité des clients** : Quels clients génèrent le plus de CA sur 5 ans ?
  - **Prévision des tendances de consommation** : Comment les ventes évoluent-elles en fonction des saisons ?

# Objectifs et fonctionnement des SI décisionnels

## Objectifs principaux des SI décisionnels

- ❖ **Faciliter l'analyse de données massives** pour prendre des décisions stratégiques
- ❖ **Permettre des requêtes complexes sur des données agrégées et historiques**
- ❖ **Offrir une interface intuitive** aux analystes et aux décideurs (tableaux de bord, visualisations)
- ❖ **Optimiser les performances** en pré-calculant les résultats via des cubes OLAP

## Fonctionnement d'un OLAP

- ❖ **Collecte et intégration des données** : Extraction des données depuis les SI transactionnels (OLTP, ERP, CRM...).
- ❖ **Transformation et stockage** : Nettoyage et agrégation dans un **Data Warehouse** ou un **Data Lake**.
- ❖ **Modélisation multidimensionnelle** : Création de **cubes OLAP** permettant des analyses rapides.
- ❖ **Exploration et analyse** : Requêtes analytiques, rapports et tableaux de bord interactifs.
- ❖ **Un OLAP repose sur un entrepôt de données centralisé optimisé pour l'analyse.**

# Les composants d'un SID

## ❖ Objectifs de cette section

- Comprendre les **principaux composants d'un SID**
- Identifier leurs **rôles et interactions** dans le système décisionnel
- Explorer des **exemples concrets et outils utilisés**

## ❖ Un **Système d'Information Décisionnel (SID)** est une architecture **composée de plusieurs éléments** qui travaillent ensemble pour transformer des données brutes en **informations exploitables pour la prise de décision**.

## ❖ Les 5 composants essentiels d'un SID sont :

- **Entrepôt de données (Data Warehouse)**
- **Processus ETL (Extract, Transform, Load)**
- **OLAP (Online Analytical Processing)**
- **Outils BI (Business Intelligence)**
- **Data Mining et Machine Learning**

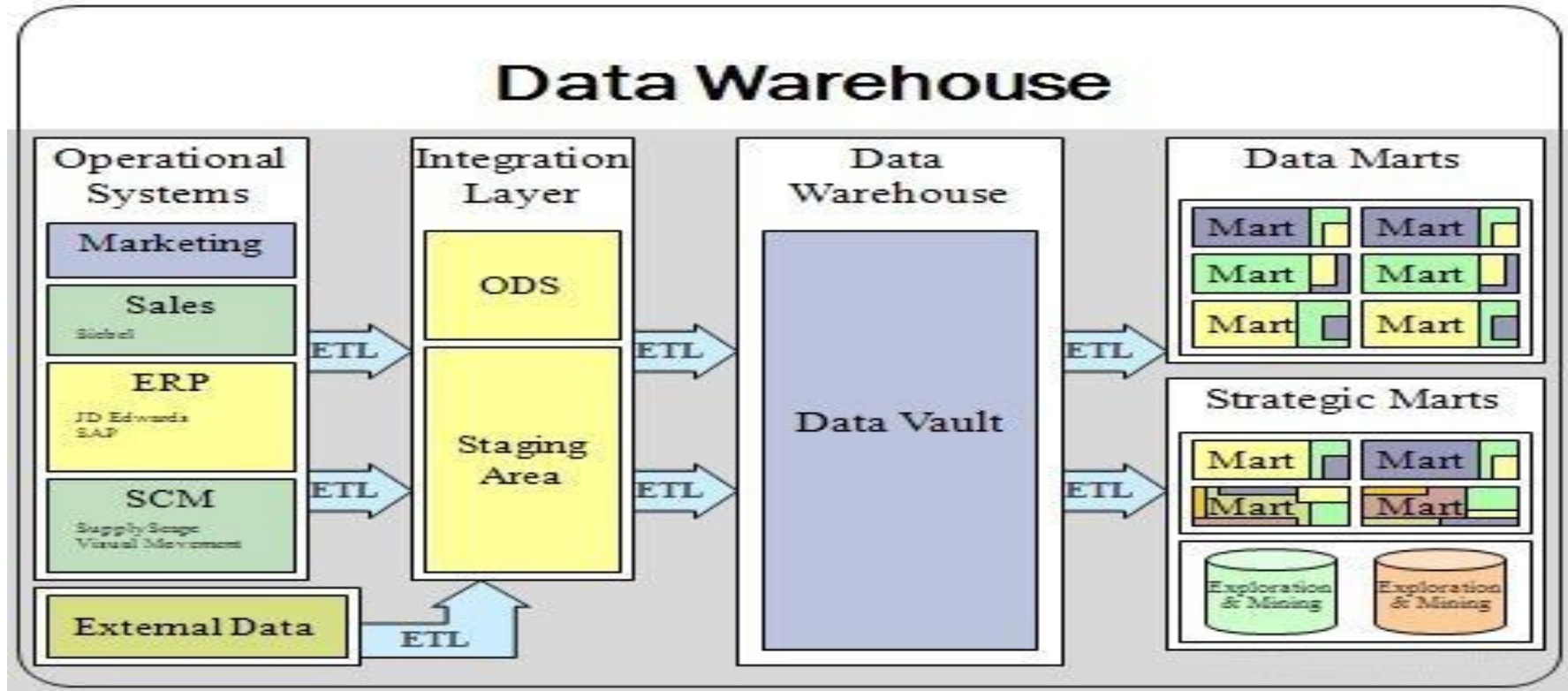


# Entrepôt de données : Data Warehouse

- ❖ **Définition** : Un **Data Warehouse** est une **base de données centralisée** utilisée pour **stocker des données historiques** provenant de diverses sources de l'entreprise (ERP, CRM, SI transactionnels, IoT...) et **analyser ces grandes quantités de données**.
- ❖ **Objectifs** :
  - **Stocker et historiser** de grandes quantités de données
  - **Fournir une source unique de vérité** pour l'analyse
  - **Faciliter les requêtes analytiques** et les tableaux de bord
- ❖ **Exemples** : Amazon Redshift, Google BigQuery, Snowflake (solutions cloud), Microsoft SQL Server Analysis Services (SSAS), Oracle Exadata, IBM Db2 Warehouse
- ❖ **Différence entre une base transactionnelle et un Data Warehouse**

Critère	Base Transactionnelle (OLTP)	Data Warehouse (OLAP)
Objectif	Exécuter des transactions rapides	Analyser et agréger des données
Données	Opérationnelles, récentes	Historiques et agrégées
Structure	Normalisée (évite la redondance)	Dénormalisée (optimisée pour l'analyse)

# Schéma classique d'un Data Warehouse



# ETL : Extract, Transform et Load

- ❖ **Définition** : Un **ETL (Extract, Transform, Load)** est un **processus d'intégration de données** permettant d'extraire des données de sources variées, de les transformer et de les charger dans un **entrepôt de données**.
- ❖ **Objectifs** :
  - **Intégrer et nettoyer les données** provenant de différentes sources
  - **Transformer les formats de données** pour les rendre exploitables
  - **Charger les données dans un Data Warehouse** pour l'analyse

## Processus ETL en 3 étapes

- ❖ **Extract (Extraction)** :
  - Récupération des données depuis les ERP, CRM, fichiers plats, API, ...
  - Définir la fréquence de récupération des données.
  - Évaluer la qualité des données : sources sûres ou non.
  - Intégrité du système indispensable : panne, problème sur des sources, ...
- ❖ **Transform (Transformation)** : Nettoyage, enrichissement et standardisation des données.
  - Nettoyage des données (valeurs erronées, données manquantes, ...)
  - Connaissance des schémas de toutes les sources

# ETL : Extract, Transform et Load

## Processus ETL en 3 étapes

### ❖ Transform (Transformation)

- Liaison entre les sources de données
- Recherche des données incohérentes, les calculs et les agrégats pour la plupart.

### ❖ Load (Chargement) : Insertion dans le Data Warehouse pour analyse

- Transfert des données vers le Data Warehouse
- Création et/ou chargement des tables de dimensions et des faits
- Indexation des données pour optimisation des requêtes
- Alerte des utilisateurs sur la mise à jour

# Les outils ETL

- ❖ Les outils ETL permettent de récupérer des données à partir de différentes sources, de les nettoyer, de les transformer et de les charger dans un entrepôt de données.

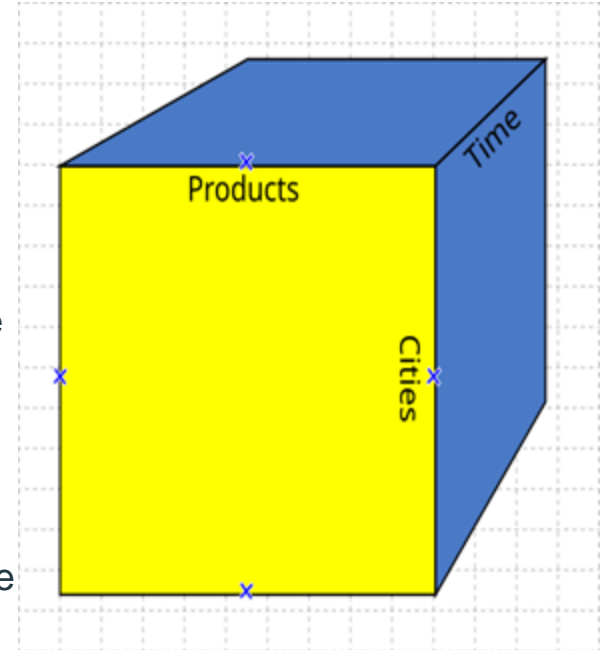
Outil	Fonctionnalités clés	Points forts
<b>Talend Open Studio</b>	Extraction, transformation et chargement de données, automatisation des flux	Open-source, flexible, compatible avec divers formats de données
<b>Apache NiFi</b>	Orchestration et automatisation des flux de données en temps réel	Adapté aux architectures distribuées, gestion de streaming
<b>Informatica PowerCenter</b>	Plateforme leader pour l'intégration de données	Haute performance, support avancé des transformations complexes

# Cube OLAP : OnLine Analytical Processing

- ❖ **Définition : Les Cubes OLAP** est une technologie qui permet **d'explorer et d'analyser les données multidimensionnelles** de manière rapide et interactive en les organisant sous plusieurs axes (temps, produit, région...)..
- ❖ **Objectifs :**
  - Permettre l'analyse des données sous plusieurs angles (axes multidimensionnels)
  - Optimiser les calculs analytiques complexes
  - Fournir des tableaux de bord interactifs
- ❖ **Modélisation multidimensionnelle**
  - **Faits** (métriques chiffrées) : Ex : ventes, bénéfices
  - **Dimensions** (axes d'analyse) : Ex : temps, produit, région
- ❖ **Types de cubes OLAP :**
  - **ROLAP (Relational OLAP)** – Analyse sur des bases relationnelles (ex: PostgreSQL, Oracle)
  - **MOLAP (Multidimensional OLAP)** – Pré-calcul des agrégats pour des performances optimales (ex: Microsoft SSAS, IBM Cognos)
  - **HOLAP (Hybrid OLAP)** – Combine les avantages des ROLAP et MOLAP
- ❖ **Exemples de technologies OLAP (Microsoft SSAS, IBM Cognos Analytics, SAP BW (Business Warehouse)) :** Une banque utilise **MOLAP** pour **analyser les transactions frauduleuses sur plusieurs années en un clic.**

# Cube OLAP

- ❖ Un fait (une valeur) décrit par plusieurs dimensions (plus de 3 en général)
- ❖ Opérations possibles :
- ❖ **Slicing** : choix d'une modalité d'une dimension et une cube avec moins de dimension comme résultat (ex: projection de colonne)
- ❖ **Dicing** : choix d'un sous-ensemble de modalités d'une dimension. Résultat : cube de mêmes dimensions (ex : sélection de ligne)
- ❖ **Drill-down/Roll-up** : parcours dans une hiérarchie. Nécessite une fonction d'agrégation facile à calculer pour passer d'un niveau à l'autre
  - Drill-down : plongée dans un sous-élément
  - Roll-up : remontée sur le parent
- ❖ **Pivot** : Rotation du cube pour modifier la présentation et permet de donner une autre vision des mêmes données.



# Les outils BI (Business Intelligence)

- ❖ **Définition :** Les outils **BI** permettent aux entreprises d'**explorer**, de **visualiser** et d'**analyser** les **données** à travers des **rapports interactifs** et des **tableaux de bord**.
- ❖ **Objectifs :**
  - Transformer les données brutes en insights exploitables
  - Créer des indicateurs clés de performance (KPI)
  - Faciliter la prise de décision grâce à des visualisations dynamiques
- ❖ **Fonctionnalités des outils BI**
  - Rapports et tableaux de bord interactifs
  - Analyse ad hoc (exploration des données en temps réel)
  - Prévisions et tendances
- ❖ **Exemples d'outils BI :**
  - **Power BI** (Microsoft) – Solution populaire pour l'analyse et le reporting
  - **Tableau** – Outil BI avancé pour la visualisation des données
  - **Apache Superset** – Alternative open-source pour la BI
- ❖ **Cas d'usage :** Un directeur commercial utilise **Tableau** pour **suivre les performances des ventes en temps réel** et **ajuster ses stratégies**.



# Data Mining et Machine Learning

- ❖ **Définition** : Le **Data Mining** et le **Machine Learning** permettent d'extraire des modèles cachés et des tendances dans les données grâce à des **algorithmes avancés**.
- ❖ **Objectifs** :
  - Découvrir des relations et des modèles cachés dans les données
  - Faire des prévisions et recommandations
  - Automatiser l'analyse des données
- ❖ **Principaux algorithmes utilisés** : Clustering, Classification, Séries temporelles
- ❖ **Exemples d'outils et bibliothèques ML** :
  - Scikit-learn (Python) – Algorithmes de Machine Learning
  - TensorFlow & PyTorch – Deep Learning

# Panorama des outils et technologies

## Les composants d'un SID

- ❖ **Entrepôt de données** (Data Warehouse)
- ❖ **ETL** (Extract, Transform, Load)
- ❖ **OLAP** (Online Analytical Processing)
- ❖ **Outils BI** (Business Intelligence)
- ❖ **Data Mining et Machine Learning**

## Outils populaires par catégorie

- ❖ **Outils ETL (Extraction, Transformation, Chargement)**
  - **Talend Open Studio** (open-source)
  - **Apache NiFi** (gestion des flux de données en temps réel)
  - **Informatica PowerCenter** (leader industriel)
- ❖ **Bases de données décisionnelles & Data Warehouses**
  - **Amazon Redshift, Google BigQuery, Snowflake** (solutions cloud)
  - **Microsoft SQL Server Analysis Services (SSAS)**
  - **PostgreSQL avec extensions OLAP**

## ❖ Outils OLAP et Reporting

- **Power BI** – Tableaux de bord interactifs
- **Tableau** – Visualisation avancée des données
- **Apache Superset** – Alternative open-source

## ❖ Technologies Big Data et Cloud appliquées aux SID

- **Hadoop, Apache Spark** – Analyse et traitement de données massives
- **Data Lakes** (Azure Data Lake, AWS S3)

# Plan global

- ❖ Introduction aux Systèmes d'Information Décisionnels
- ❖ **Architecture des Systèmes d'Information Décisionnels**
- ❖ Modélisation et Conception des Entrepôts de Données
- ❖ Extraction, Transformation et Chargement
- ❖ Analyse et Exploration des Données
- ❖ Rappel : Data Mining et Machine Learning appliqué au SID
- ❖ Tendances et Innovations en SID

# Architecture des Systèmes d'Information Décisionnels

# Modèles Architecture (Inmon vs Kimball)

## Objectifs pédagogiques :

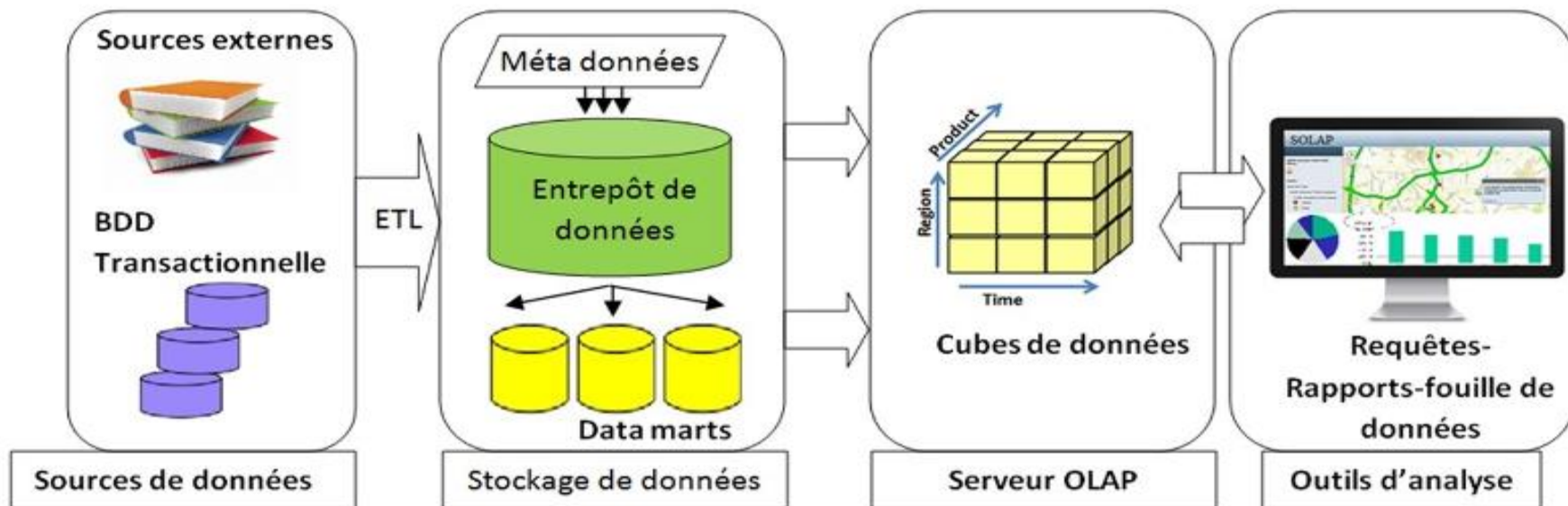
- ❖ Comprendre les sources de données
- ❖ Comprendre les deux principales approches d'architecture des Data Warehouses
- ❖ Différencier le modèle **Top-Down** (Inmon) et **Bottom-Up** (Kimball)
- ❖ Identifier les avantages et inconvénients de chaque modèle

**Présentation des Approches de Data Warehousing : SID** reposent sur des **entrepôts de données (Data Warehouses)**. Deux écoles majeures définissent leur **modélisation et architecture** :

Approche	Auteur	Stratégie	Utilisation
<b>Top-Down</b>	<b>Bill Inmon</b>	Data Warehouse centralisé avant Data Marts	Entreprises avec besoins analytiques globaux
<b>Bottom-Up</b>	<b>Ralph Kimball</b>	Data Marts indépendants avant intégration globale	Entreprises avec besoins analytiques locaux rapides

# Source de données

- ❖ Les systèmes décisionnels peuvent avoir besoin de données issues de différentes sources et de différents formats de stockage.
- ❖ Exemples les fichiers texte, les rapports, les fichiers de base de données



# Modèle de Bill Inmon (Approche Top-Down)

## ❖ Principe :

- Le **Data Warehouse (DW)** est **normalisé** en amont (3e forme normale)
- Il est la **source unique de vérité**
- Les **Data Marts** sont ensuite **dé-normalisés** pour des besoins spécifiques

## ❖ Architecture du Modèle Inmon :

- **Sources de données opérationnelles** (ERP, CRM, fichiers, etc.)
- **ETL** → Extraction, transformation, normalisation, chargement
- **Data Warehouse centralisé** (entrepôt structuré en schéma relationnel)
- **Data Marts** spécialisés pour répondre aux besoins des métiers

## ❖ Avantages du modèle Inmon :

- **Qualité et cohérence des données** grâce à un entrepôt unique
- **Flexibilité analytique** : supporte des analyses globales
- **Meilleure intégrité et évolutivité**

## ❖ Inconvénients du modèle Inmon :

- **Déploiement plus long** et plus complexe
- **Coût initial élevé** dû à la centralisation des données
- **Courbe d'apprentissage** plus difficile pour les analystes métier

# Modèle de Ralph Kimball (Approche Bottom-Up)

## ❖ Principe :

- Les **Data Marts** sont conçus en premier, organisés par domaine métier
- Ils utilisent des modèles **d'étoile** (Star Schema) ou **de flocon** (Snowflake Schema)
- Une éventuelle consolidation en **Data Warehouse** vient après

## ❖ Architecture du Modèle Kimball :

- **Sources de données opérationnelles**
- **ETL** → Extraction et transformation ciblée pour chaque domaine
- **Data Marts** spécialisés selon les besoins métiers
- **OLAP / Reporting** directement sur les Data Marts

## ❖ Avantages du modèle Kimball :

- **Implémentation plus rapide** et moins coûteuse
- **Optimisé pour les requêtes analytiques**
- **Meilleur support des besoins spécifiques des métiers**

## ❖ Inconvénients du modèle Kimball :

- **Risque d'incohérence** entre plusieurs Data Marts
- **Moins de flexibilité** pour des analyses globales
- **Difficulté d'intégration des nouveaux besoins métiers**



# Comparaison entre les deux modèles

- ❖ **Inmon** pour une **vision stratégique globale** et une **meilleure gouvernance**
- ❖ **Kimball** pour une **implémentation plus rapide** et une **meilleure réponse aux besoins métiers**
- ❖ **Approche hybride** : Beaucoup d'entreprises combinent les deux modèles

Critère	Modèle Inmon (Top-Down)	Modèle Kimball (Bottom-Up)
Structure	Data Warehouse centralisé	Plusieurs Data Marts indépendants
Complexité	Conception plus complexe	Plus simple et rapide à déployer
Flexibilité	Facilement adaptable	Plus rigide aux évolutions
Performance	Performant pour les analyses globales	Rapide pour des besoins spécifiques
Coût initial	Élevé (besoin d'une infrastructure robuste)	Plus faible (déploiement progressif)
Qualité des données	Forte intégrité et standardisation	Risque d'incohérence entre Data Marts
Adaptation au Big Data	Moins flexible (besoin d'une refonte)	Plus adaptable aux nouvelles technologies

# Data Lake et hybridation avec les SID traditionnels

- ❖ **Définition :** Un **Data Lake** est un **référentiel de stockage centralisé** permettant de conserver de **grandes quantités de données** dans leur **format natif**, qu'elles soient **structurées, semi-structurées ou non structurées**.
- ❖ **Caractéristiques principales :**
  - **Stockage brut** sans transformation immédiate
  - **Scalabilité élevée** (Big Data, stockage distribué)
  - **Flexibilité** : supporte différents formats de fichiers
  - **Exploration avancée** : Machine Learning (ML), Intelligence Artificielle (IA)
- ❖ **Sources de données pouvant alimenter un Data Lake :**
  - **IoT (Internet des objets)** → Données capteurs, logs machines
  - **ERP, CRM** → Données structurées des systèmes transactionnels
  - **Logs système et réseaux sociaux** → Semi-structuré (JSON, XML)
  - **Fichiers multimédias** → Vidéos, images, sons

# Comparaison Data Warehouse vs Data Lake

Critère	Data Warehouse	Data Lake
Nature des données	Structurées	Structurées, semi-structurées et non structurées
Modèle de stockage	Modèles relationnels (modèles en étoile, flocon)	Stockage brut en fichiers
Performance des requêtes	Très optimisé pour l'analytique	Performant avec technologies Big Data (Spark, Presto)
Coût	Coût élevé (infrastructure SQL)	Moins cher (stockage objet type S3, HDFS)
Cas d'usage	BI, reporting, KPI financiers	Data Science, Machine Learning, logs, IoT

# Modèle hybrides : Le Data Lakehouse

- ❖ Le **Data Lakehouse** : Permet aux entreprises modernes de **combiner les avantages** du Data Warehouse (optimisation analytique) et du Data Lake (scalabilité et flexibilité).
- ❖ **Définition** : Le **Data Lakehouse** est une **architecture hybride** qui combine :
  - La **gestion des données brutes** du Data Lake
  - L'**optimisation analytique** d'un Data Warehouse
  - Un **moteur transactionnel** pour assurer la qualité des données
- ❖ **Exemples de technologies Data Lakehouse**

Technologie	Description
<b>Delta Lake (Databricks)</b>	Ajoute des fonctionnalités ACID et de versioning au Data Lake
<b>Apache Iceberg</b>	Format optimisé pour requêter efficacement des fichiers stockés dans un Data Lake
<b>Snowflake</b>	Plateforme Cloud qui combine Data Warehousing et Data Lake

# Cloud & SID

- ❖ **Cloud & SID** : Le Cloud offre des **solutions évolutives** et **flexibles** pour la gestion des entrepôts de données et l'analytique, permettant de s'affranchir des limites des infrastructures traditionnelles.
  - **Avantages du Cloud pour les SID** : Scalabilité, Haute disponibilité, Flexibilité des coûts "pay-as-you-go: paiement à l'usage", Facilité d'intégration et Sécurité et conformité
  - **Solutions Cloud SID** : Amazon Redshift, Google BigQuery, Azure Synapse Analytics, Snowflake
- ❖ **Interopérabilité avec les SI existants** : Les entreprises fonctionnent souvent avec des **systèmes existants (on-premise)** et cherchent à migrer **progressivement** vers le Cloud.
- ❖ **Scénarios d'interopérabilité** :
  - Approche hybride : Maintenir des données critiques **on-premise** et transférer certaines analyses vers le Cloud
  - Migration progressive : Déplacer les bases de données progressivement vers le Cloud pour éviter un basculement brutal
  - Cloud natif : Adopter une architecture 100% Cloud dès le départ

# Plan global

- ❖ Introduction aux Systèmes d'Information Décisionnels
- ❖ Architecture des Systèmes d'Information Décisionnels
- ❖ **Modélisation multidimensionnelle**
- ❖ Extraction, Transformation et Chargement
- ❖ Analyse et Exploration des Données
- ❖ Rappel : Data Mining et Machine Learning appliqué au SID
- ❖ Tendances et Innovations en SID

# Modélisation multidimensionnelle

# Modélisation multidimensionnelle

## ❖ Objectifs

- Comprendre les concepts fondamentaux de la **modélisation dimensionnelle**
- Comparer les modèles **en étoile et en flocon**
- Analyser les **avantages et inconvénients** de chaque modèle
- Étudier des **cas d'application concrets** et mettre en pratique

## ❖ Introduction à la modélisation dimensionnelle : La modélisation dimensionnelle est une **technique de conception des bases de données décisionnelles** utilisée dans les **entrepôts de données (Data Warehouses)** et les **systèmes OLAP**.

## ❖ La Modélisation multidimensionnelle d'après (Teste, 2000), consiste à «considérer un sujet analysé comme un point dans un espace à plusieurs dimensions. Les données sont organisées de manière à mettre en évidence le sujet analysé et les différentes perspectives de l'analyse».

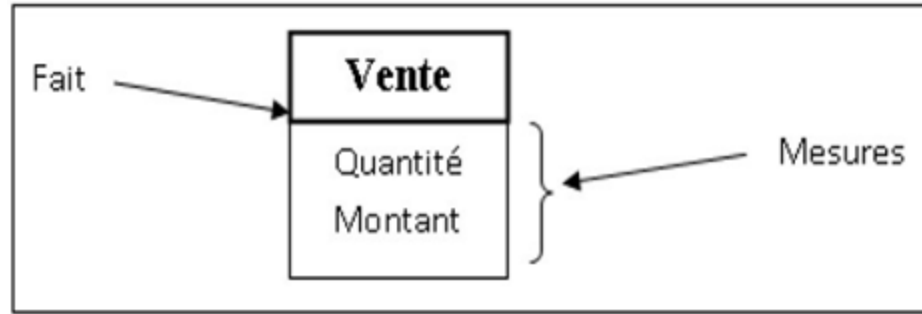


# Modélisation conceptuelle : Fait

- ❖ La modélisation conceptuelle fait référence aux concepts définis dans les travaux de Ravat (2005) et qui sont utilisés dans différents types de schémas de données (Ravat et al. 2005; Naoum, 2006; Kimball, 1996).
- ❖ **Concepts de base** : Un ensemble de concepts sont utilisés pour concevoir un modèle multidimensionnel de données.
  - **Table de faits** : Contient des **données transactionnelles** et des **mesures** à analyser
  - **Table de dimensions** : Contient les **attributs descriptifs** pour contextualiser les faits
  - **Clés primaires et étrangères** : Les tables de faits possèdent des **clés étrangères** pointant vers les **dimensions**
- ❖ **Fait** : C'est le concept qui modélise le sujet de l'analyse.
  - C'est un événement d'intérêts pour une entreprise
  - Une ligne dans la table des faits
  - **Tables de Faits**
    - Il s'est produit quelque chose
    - Il s'est produit autre chose
    - Il s'est encore produit quelque chose

# Modélisation conceptuelle : Mesure

- ❖ **Mesure** : C'est la valeur d'attribut quantitatif ou qualitatif qui évalue un fait. C'est un indicateur basé sur les données et de valeur numérique du fait. Voir figure ci-dessous.



- Mesures de type “**flux**” : Mesures qui peuvent être sommées selon toutes les dimensions (e.g. "montant de vente");
- Mesures de type “**stock**” : Mesures qui ne peuvent pas être sommées selon certaines dimensions (e.g. "population");
- Mesures de type “**valeur par unité**” : Mesures qui ne peuvent jamais être sommées (e.g. "température").

# Modélisation conceptuelle : Dimension

- ❖ **Dimension** : Ensemble de paramètres qui peuvent faire varier les mesures. Elle modélise une perspective de l'analyse. Voir figure ci-dessous.

- Axe d'analyse des faits
- Contexte de chaque fait.

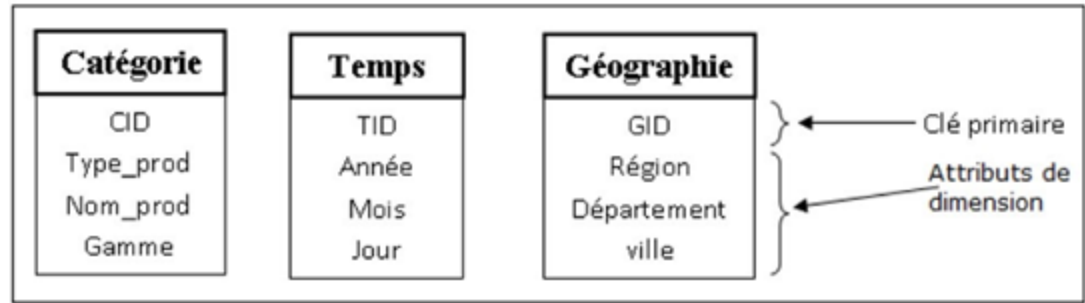


Table de Faits		
Quand	Où	Quoi
Hier	Ici	Il s'est produit quelque chose
Hier	Là	Il s'est produit autre chose
Aujourd'hui	Ici	Il s'est encore produit quelque chose

# Modélisation conceptuelle : Agrégation des mesures

- ❖ **Agrégation des mesures** : C'est une opération qui permet de calculer différents indicateurs d'analyse à différents niveaux de détail. Elle utilise trois éléments clefs du modèle multidimensionnel à savoir la mesure, la fonction d'agrégation et la hiérarchie de dimension et dépend des conditions d'agrégabilité. Nous distinguons trois conditions d'agrégabilité (Lenz et Shoshani, 1997),
  - **Disjonction** : pour permettre d'éviter le comptage en double des valeurs de mesure.
  - **Complétude** : pour le but d'éviter le problème d'agrégats incomplets.
  - **Comptabilité de type** : cette condition vérifie que les natures des trois éléments, mesure, dimension et fonction d'agrégation, sont compatibles.
- ❖ **Cube de données (Hypercube)** : La notion du cube de données (hypercube) a été proposée par (Gray et al., 1997) et est défini comme « un ensemble de données organisées selon des dimensions. On appelle mesure la valeur contenue dans une cellule du cube, associée aux valeurs prises sur les dimensions composant le cube ».

# Modélisation conceptuelle : Hiérarchie

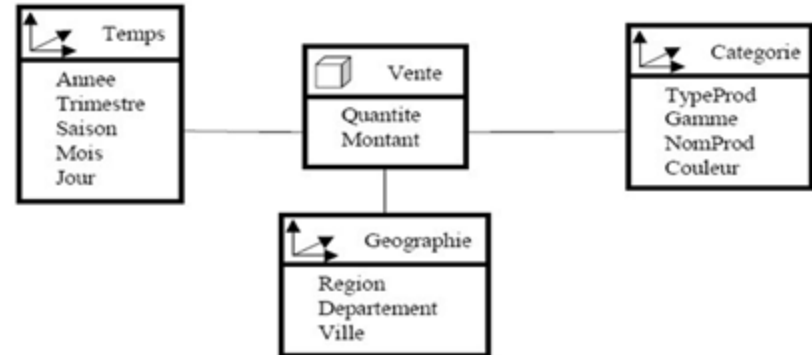
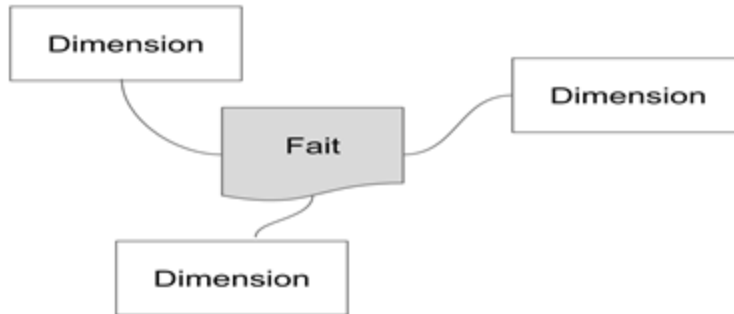
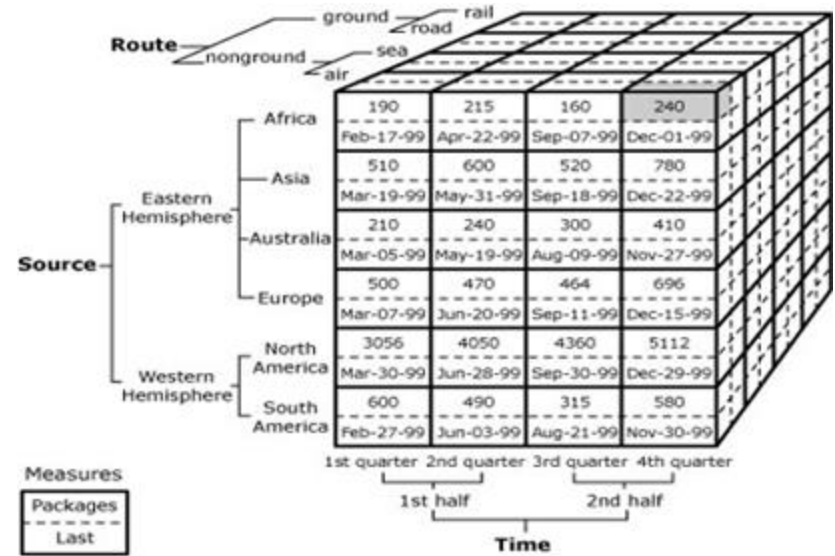
- ❖ **Hiérarchie** : Une hiérarchie permet de structurer les dimension en plusieurs niveaux de granularité. D'après Mazón et Al, les hiérarchies peuvent être classées en deux catégories :
  - Hiérarchies régulières : qui satisfont les conditions de la complétude et la disjonction. Ici nous distinguons :
    - **Hiérarchies strictes** : spécifient pour chaque membre feuille un seul chemin d'agrégation vers le membre racine de la hiérarchie.
    - **Hiérarchies onto** : Tout membre non feuille possède au moins un membre fils et tous les membres feuille se trouvent au même niveau qui est le niveau d'agrégation feuille de la hiérarchie.
    - **Hiérarchies covering** : une hiérarchie est dite covering si elle ne présente pas de raccourcis ou de sauts dans les liens d'agrégation.
  - **Hiérarchies irrégulières** : ces hiérarchies ne satisfont pas l'une ou les deux conditions d'agrégabilité citées ci-dessus (disjonction et complétude).

# Les différents modèles

- ❖ **Schéma en étoile** : C'est une seule table de faits et plusieurs tables de dimension sont directement rattachées à la table des faits.
- ❖ **Schéma en flocon de neige** : Il consiste à faire la normalisation des dimensions et contient la notion de hiérarchie des dimensions
- ❖ **Schéma en constellation des faits** : Nous avons plusieurs tables de faits et les tables de dimensions sont reliées à une ou plusieurs tables de faits
- ❖ **Modèle mixte** : Cette technique consiste à fusionner plusieurs modèles en étoile et en flocon de neige.

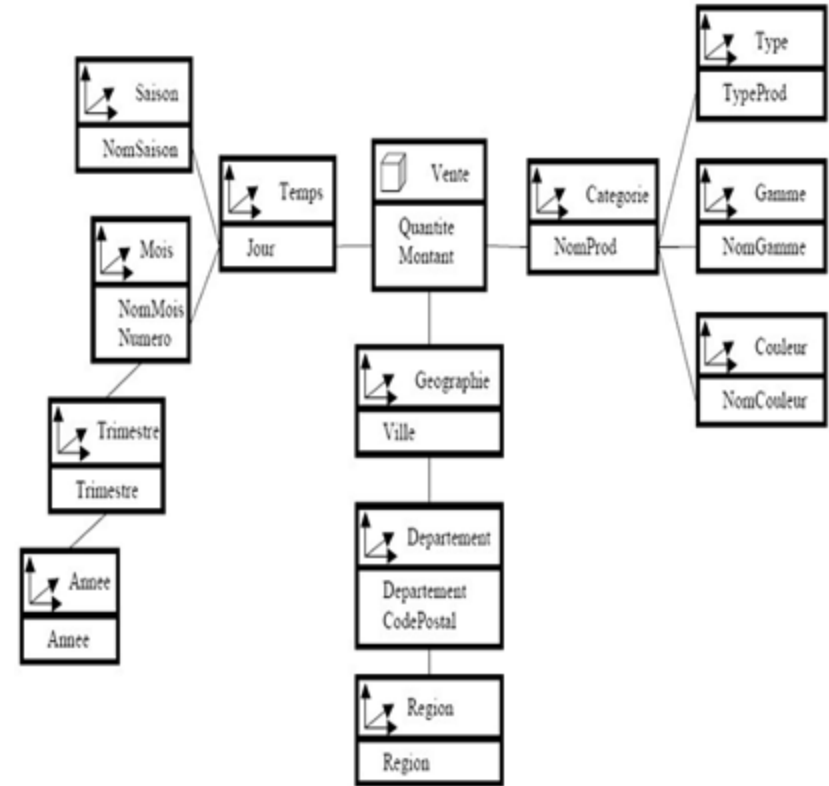
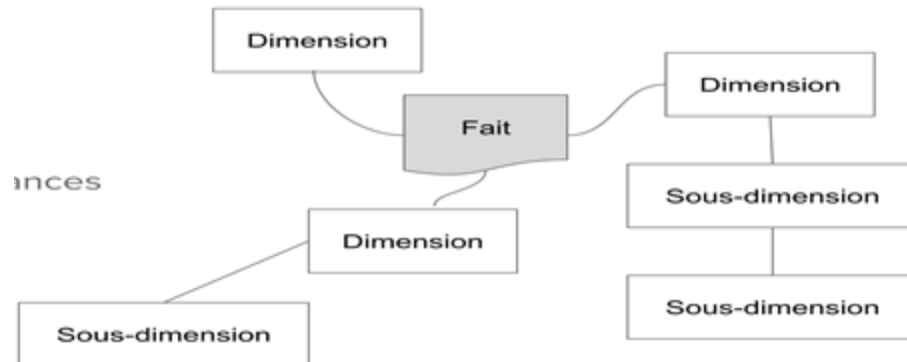
# Modèle en étoile : Star

- ❖ Le schéma est constitué du fait central et des dimensions représentées de manière dénormalisée
- ❖ Une seule table de faits
- ❖ Plusieurs dimensions dénormalisées
- ❖ Relation 1,n
- ❖ Granularité identiques entre les faits et dimensions : Mois vs. Semaine incompatibles
- ❖ Data mart



# Modèle en flocon : snowflake

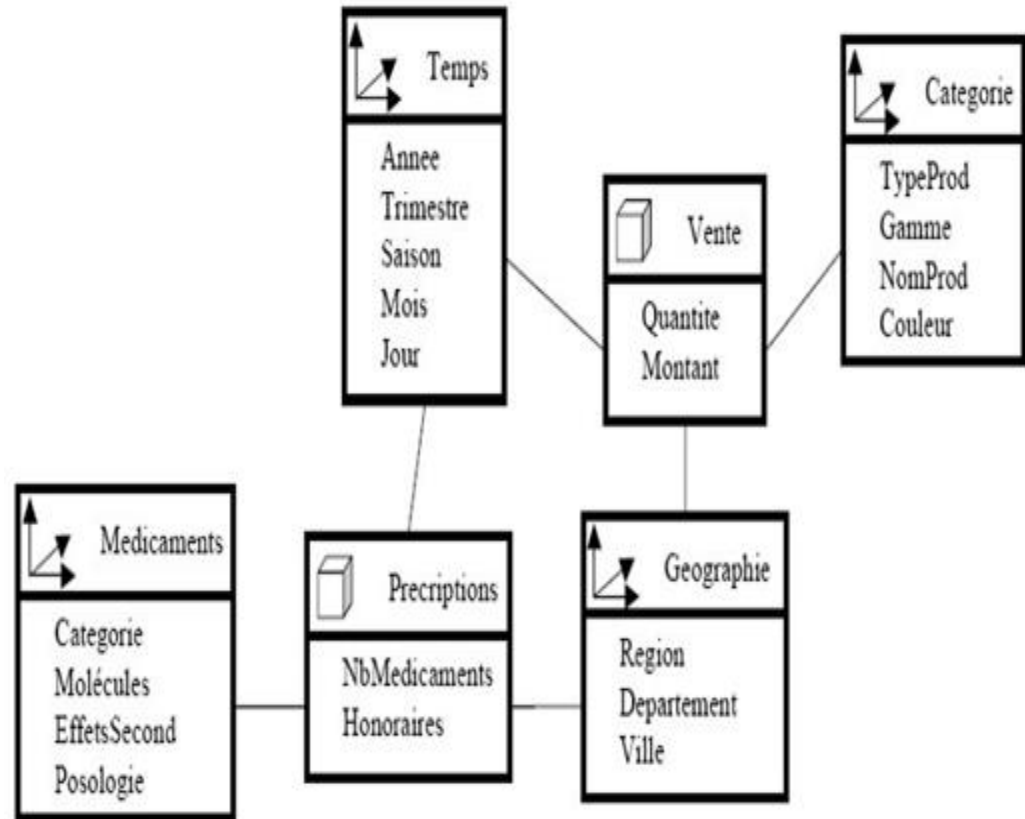
- ❖ Le schéma la table de faits est conservée et les dimensions sont divisées conformément à sa hiérarchie. Ce modèle permet d'éviter le problème de redondance qu'on peut trouver dans le modèle en étoile
- ❖ Une seule table de faits
- ❖ Plusieurs dimensions, avec des sous-dimensions, normalisées
- ❖ Hiérarchie des dimensions





# Modèle en constellation

- ❖ Ce modèle consiste à fusionner plusieurs modèles en étoile. Il correspond donc à plusieurs tables de faits qui partagent des dimensions communes.
- ❖ Plusieurs tables de faits
- ❖ Plusieurs dimensions avec certaines dimensions communes



# Modèle mixte

- ❖ Ce modèle consiste à fusionner plusieurs modèles en étoile et en flocon de neige

# Cas pratique : Modélisation d'un entrepôt de données

- ❖ Cas : Une chaîne de magasins souhaite analyser les **ventes**, les **clients** et les **produits** vendus.
- ❖ **Étape 1 : Identifier la table de faits**
  - Table **Faits\_Ventes** contenant : **ID\_Produit**, **ID\_Client**, **Date\_Achat**, **Montant\_Vente**
- ❖ **Étape 2 : Identifier les dimensions**
  - **Dimension\_Temps** : Année, Mois, Jour
  - **Dimension\_Client** : Nom, Adresse, Âge
  - **Dimension\_Produit** : Nom, Catégorie
- ❖ **Étape 3 : Choisir la modélisation**
  - **Modèle en étoile** : Chaque dimension est directement reliée à la table de faits
  - **Modèle en flocon** : Catégorie de produit est séparée dans une table distincte
- ❖ **Exercice :**
  - Construire un **schéma en étoile** pour le modèle de ventes
  - Transformer ce modèle en **schéma en flocon**
  - Rédiger des **requêtes SQL analytiques** pour récupérer le chiffre d'affaires mensuel

# Outils

Data warehousing : PostgreSQL + extensions OLAP (Citus, Greenplum, TimescaleDB) / ClickHouse /

ETL : Kettle (Pentaho Data Integration)

Reporting : Apache Superset / Metabase