



**WEBFORCE**  
BE THE CHANGE



# RÉSUMÉ THÉORIQUE – FILIÈRE INTELLIGENCE ARTIFICIELLE

## M105 - Appréhender la modélisation des données



60 heures



# SOMMAIRE

## 01 – INTRODUIRE LE CONTEXTE DE LA MODÉLISATION DES DONNÉES

1. Introduire la modélisation des données
2. Introduire le domaine du business intelligence

## 02 – II. APPREHENDER LE MODELE DIMENSIONNEL

1. Maitriser les faits
2. Maitriser les dimensions
3. Appréhender les dimensions à évolution lente



# MODALITÉS PÉDAGOGIQUES



**WEBFORCE**  
BE THE CHANGE



1

## LE GUIDE DE SOUTIEN

Il contient le résumé théorique et le manuel des travaux pratiques



2

## LA VERSION PDF

Une version PDF est mise en ligne sur l'espace apprenant et formateur de la plateforme WebForce Life



3

## DES CONTENUS TÉLÉCHARGEABLES

Les fiches de résumés ou des exercices sont téléchargeables sur WebForce Life



4

## DU CONTENU INTERACTIF

Vous disposez de contenus interactifs sous forme d'exercices et de cours à utiliser sur WebForce Life



5

## DES RESSOURCES EN LIGNES

Les ressources sont consultables en synchrone et en asynchrone pour s'adapter au rythme de l'apprentissage



## PARTIE 1

# INTRODUIRE LE CONTEXTE DE LA MODÉLISATION DES DONNÉES

Dans ce module, vous allez :

- Introduire la modélisation des données
- Définir le modèle dimensionnel



15 heures



# CHAPITRE 1

## INTRODUIRE LA MODÉLISATION DES DONNÉES

**Ce que vous allez apprendre dans ce chapitre :**

- Définir la modélisation des données
- Connaître les types de modèles des données



**05 heures**

# CHAPITRE 1

## INTRODUIRE LA MODÉLISATION DES DONNÉES

### 1. Définitions

### 2. Types

- Modèle relationnel
- Modèle entité-relation (ERM)
- Modèle dimensionnel
- Modèle en réseau



# 1 - INTRODUIRE LA MODÉLISATION DES DONNÉES

## Définitions



### Modélisation des données

La modélisation des données est le processus de création d'un modèle qui représente la structure, les règles et les relations des données d'un système ou d'une organisation. Il s'agit d'une étape importante dans la conception d'une base de données et dans le développement d'applications logicielles qui utilisent des données.

Le processus de modélisation des données implique la définition des entités, des attributs et des relations entre les entités, ainsi que des règles et des contraintes qui s'appliquent aux données. Le résultat de la modélisation des données est généralement un diagramme qui représente graphiquement la structure des données.

Le modèle de données permet de comprendre comment les données sont stockées, organisées et utilisées dans un système ou une organisation. Il peut être utilisé pour faciliter la communication entre les développeurs, les utilisateurs et les administrateurs de la base de données, et pour garantir la cohérence et la qualité des données.

La modélisation des données est une étape importante dans la conception de systèmes d'information et de bases de données, ainsi que dans la gestion et l'analyse des données. Elle est utilisée dans de nombreux domaines, y compris l'informatique, la gestion de projet, la science des données et l'ingénierie logicielle.

# CHAPITRE 1

## INTRODUIRE LA MODÉLISATION DES DONNÉES

### 1. Définitions

### 2. Types

- Modèle relationnel
- Modèle entité-relation (ERM)
- Modèle dimensionnel
- Modèle en réseau





### Vue générale

- Il existe plusieurs types de modèles des données, dont voici les principaux :

#### Modèle relationnel

- Ce modèle utilise des tables pour stocker les données et des relations entre les tables pour représenter les relations entre les données.

#### Modèle entité-relation (ERM)

- C'est le type le plus courant de modélisation des données. Il utilise des entités (objets ou concepts) et des relations pour représenter la structure et les relations des données.

#### Modèle dimensionnel

- c'est un autre modèle couramment utilisé pour la modélisation des Data Warehouse (entrepôts de données). Il utilise des dimensions (caractéristiques) et des mesures (valeurs numériques) pour représenter les données.

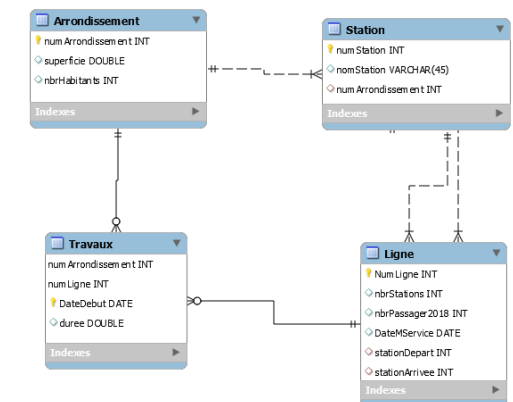
#### Modèle en réseau

- C'est un modèle qui organise les données sous forme de graphes connectés, où les enregistrements sont liés les uns aux autres par des relations complexes. Dans ce modèle, les données sont représentées par des nœuds et les relations entre les données sont représentées par des arêtes.

- Ces différents types de modélisation des données peuvent être utilisés pour différents types de systèmes et de bases de données, en fonction des besoins spécifiques de chaque projet.

## Modèle relationnel

- Le modèle relationnel est une approche de modélisation de données qui est basée sur la théorie des ensembles mathématiques et qui repose sur l'utilisation de tables et des relations pour organiser et structurer les données.
- Dans un modèle relationnel, les données sont représentées sous forme de tables. Chaque table ayant une série de colonnes qui représentent les attributs ou les caractéristiques des données et des lignes qui représentent les enregistrements ou les instances des données. Les tables sont reliées les unes aux autres à travers des clés primaires et étrangères, qui permettent d'établir des relations entre les différentes tables et donc de lier les données entre elles.
- Le modèle relationnel est devenu très populaire dans les années 1970 grâce à la publication des travaux de Edgar F. Codd. Il est aujourd'hui largement utilisé dans la conception de systèmes de gestion de bases de données relationnelles (SGBDR).
- L'avantage du modèle relationnel est qu'il permet de structurer les données de manière logique et cohérente, ce qui facilite leur manipulation et leur accès par les utilisateurs et les applications. De plus, la normalisation des données permet d'assurer l'intégrité et la qualité des données stockées dans la base de données.



### Modèle relationnel

- Le modèle relationnel peut parfois être rigide et ne pas convenir à toutes les situations. Par exemple, lorsque les données sont très complexes et nécessitent une modélisation plus flexible, d'autres approches de modélisation peuvent être préférables, comme le modèle objet-relationnel ou le NoSQL.
- Voici quelques exemples de bases de données relationnelles :



Oracle  
Database



MySQL



Microsoft SQL  
Server



PostgreSQL



IBM DB2

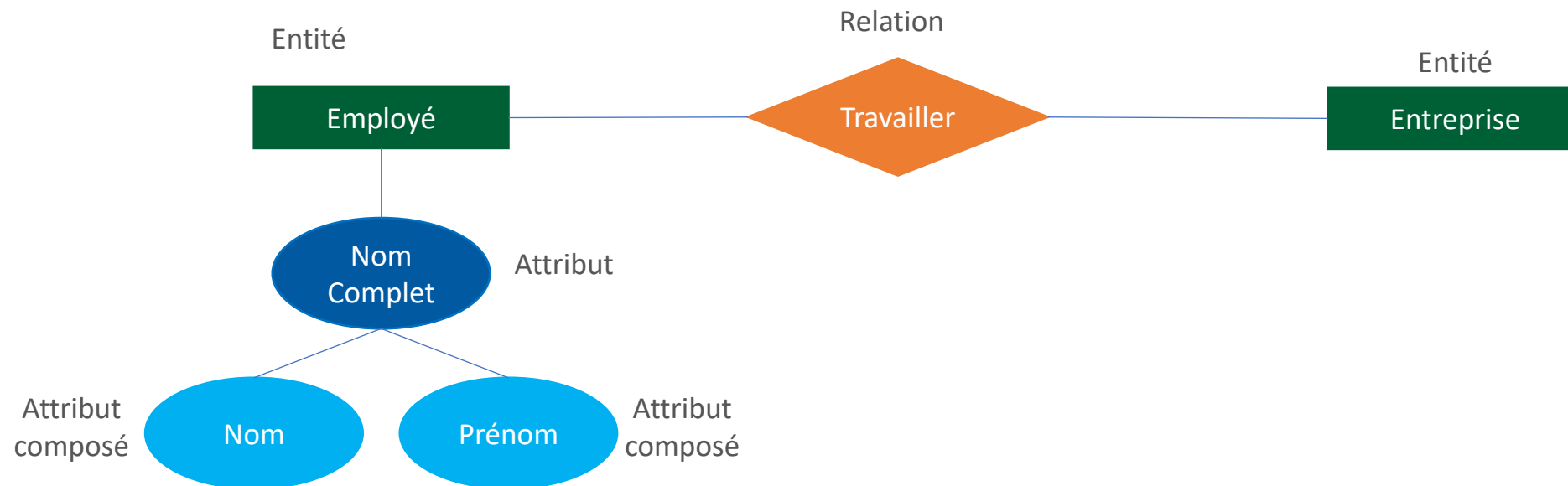


MariaDB

- Ces bases de données sont utilisées dans de nombreux domaines d'application, tels que la gestion de la relation client, la gestion des stocks, la finance, la santé, l'éducation, etc.

### Modèle entité-relation (ERM)

- Le modèle entité-relation (ERM) est un modèle de données conceptuel utilisé pour représenter les entités (objets, concepts, personnes, etc.) d'un système ainsi que les relations entre ces entités.
- Il se base sur la notion d'entités, qui sont des objets distincts et identifiables, et les relations qui existent entre ces entités.
- Le modèle ERM utilise des diagrammes entité-relation pour illustrer les entités, leurs attributs (caractéristiques) et les connexions entre elles, permettant de décrire la structure et les règles de gestion des données d'un système de manière graphique et intuitive.



### Modèle entité-relation (ERM)

- Voici les principales différences entre le modèle relationnel et l'ERM

#### Structure des données

- Le modèle entité-relation utilise des entités pour représenter les objets ou les concepts, tandis que le modèle relationnel utilise des tables pour stocker les données.

#### Relations

- Le modèle entité-relation utilise des relations pour connecter les entités et représenter les associations entre les objets ou les concepts, tandis que le modèle relationnel utilise des clés étrangères pour relier les tables et représenter les relations entre les données.

#### Abstraction

- Le modèle entité-relation est plus abstrait et peut être utilisé pour représenter des concepts ou des idées plus générales, tandis que le modèle relationnel est plus concret et est utilisé pour stocker des données spécifiques.

#### Diagrammes

- Le modèle entité-relation est généralement représenté par un diagramme entité-relation, qui montre les entités et les relations, tandis que le modèle relationnel peut être représenté par un schéma relationnel, qui montre les tables et les relations entre les tables.

- En général, le modèle entité-relation est utilisé pour concevoir le schéma conceptuel d'une base de données, tandis que le modèle relationnel est utilisé pour implémenter le schéma logique de la base de données en utilisant des tables et des clés étrangères. Le modèle entité-relation est donc plus abstrait et conceptuel, tandis que le modèle relationnel est plus concret et technique.

### Modèle entité-relation (ERM)

- Voici quelques exemples de bases de données ERM populaires:



Oracle  
Database



MySQL



Microsoft SQL  
Server



PostgreSQL



IBM DB2



Teradata

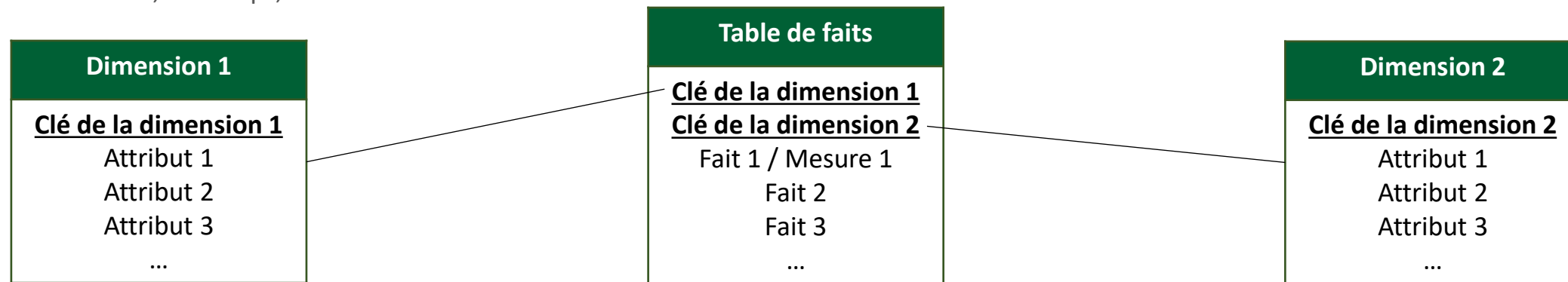


MariaDB

- Ces bases de données sont utilisées dans de nombreux domaines, notamment dans les entreprises, les organisations gouvernementales et les sites web pour stocker des données structurées.

### Modèle dimensionnel

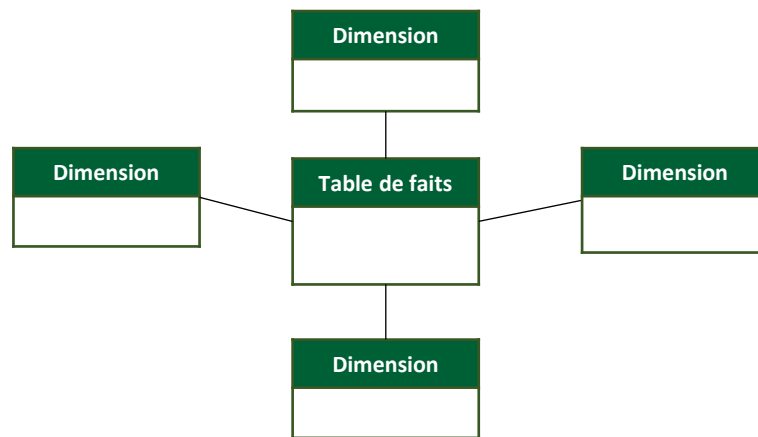
- Un modèle dimensionnel est un modèle de données utilisé en informatique décisionnelle pour organiser les données en fonction de leur signification dans un contexte métier. Il est spécifiquement conçu pour prendre en compte les aspects de temps et de mesure dans les données.
- Le modèle dimensionnel est basé sur deux types de tables : les tables de dimensions et les tables de faits. Les tables de dimensions contiennent les données de référence qui décrivent les caractéristiques des objets métier, tels que des clients, des produits, des emplacements, des temps, etc. Les tables de faits contiennent les mesures numériques qui représentent les performances, les quantités, les montants, les temps, etc.



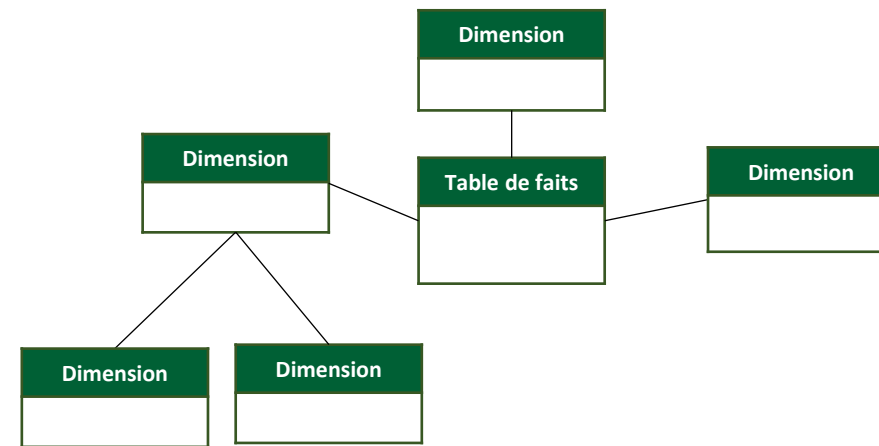
**Remarque :** On va détailler ce modèle dans ce cours

### Modèle dimensionnel

- Le modèle dimensionnel utilise également des schémas en étoile ou en flocon, qui sont des structures de tables organisées autour d'une table centrale de faits, entourée de tables de dimensions. Cette organisation permet d'optimiser les requêtes et les analyses sur les données en minimisant le nombre de jointures nécessaires.
- Le modèle dimensionnel est largement utilisé dans les entrepôts de données et les systèmes décisionnels, car il permet une analyse rapide et facile des données métier. Il est considéré comme une alternative efficace au modèle relationnel traditionnel pour l'analyse de données complexes et volumineuses.



schémas en étoile



schémas en flocon

**Remarque :** On va détailler ce modèle dans ce cours



### Modèle dimensionnel

- Plusieurs exemples de base de données peuvent utiliser le modèle dimensionnel, à savoir :

#### Les ventes

- Cette base de données comprend des informations sur les ventes de produits ou de services, ainsi que sur les clients qui les ont achetés. Les dimensions pourraient inclure les clients, les produits, les emplacements, les dates et les promotions. Les faits/mesures pourraient inclure les ventes brutes, les remises, les coûts et les marges bénéficiaires.

#### Les ressources humaines

- Cette base de données comprend des informations sur les employés et leur performance. Les dimensions pourraient inclure les employés, les postes, les départements, les emplacements et les périodes. Les mesures pourraient inclure le salaire, les heures travaillées, les absences et les performances.

#### Les finances

- Cette base de données comprend des informations sur les transactions financières, les comptes et les portefeuilles. Les dimensions pourraient inclure les clients, les comptes, les dates, les devises et les types de transactions. Les mesures pourraient inclure les soldes, les revenus, les dépenses, les bénéfices et les pertes.

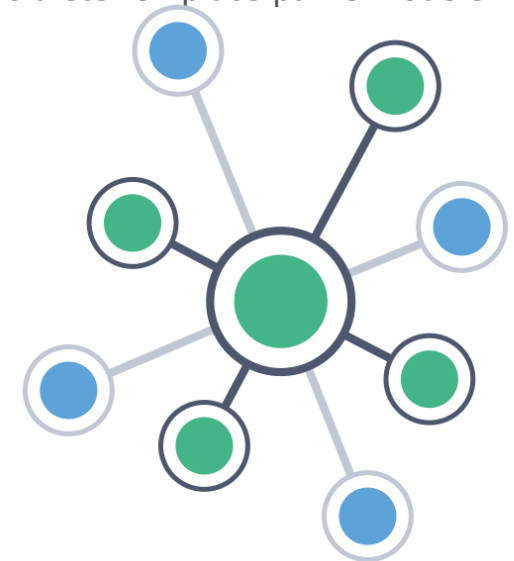
#### Les voyages

- Cette base de données comprend des informations sur les voyages, les destinations et les réservations. Les dimensions pourraient inclure les clients, les destinations, les hôtels, les compagnies aériennes et les dates. Les mesures pourraient inclure les coûts, les réservations, les annulations et les revenus.

- Ces exemples sont tous des bases de données dimensionnelles car elles sont organisées autour de dimensions clés et de mesures numériques pour permettre une analyse facile et efficace.

### Modèle en réseau

- Le modèle en réseau est un modèle de données hiérarchique qui a été développé dans les années 1960. Dans ce modèle, les données sont représentées comme des enregistrements reliés entre eux par des relations hiérarchiques. Les enregistrements sont organisés en ensembles appelés "nœuds", et chaque nœud est connecté à d'autres nœuds par des "liens". Les liens peuvent être de plusieurs types, tels que les "liens propriétaires" qui relient un nœud à ses enregistrements associés, ou les "liens non-propriétaires" qui relient un nœud à d'autres nœuds.
- Contrairement au modèle relationnel, qui utilise des tables pour stocker les données, le modèle en réseau utilise des enregistrements interconnectés pour représenter les données. Le modèle en réseau était populaire dans les années 1970, mais a été remplacé par le modèle relationnel dans les années 1980.
- Le modèle en réseau est toujours utilisé dans certains systèmes de gestion de bases de données, tels que les bases de données orientées graphes. Cependant, ces systèmes ont évolué pour prendre en charge des fonctionnalités supplémentaires qui n'étaient pas possibles dans le modèle en réseau traditionnel.



### Modèle en réseau

- Il peut encore être utilisé dans certaines situations spécifiques, telles que :

#### Stockage et analyse de données hiérarchiques

- Il est bien adapté à la modélisation de données hiérarchiques, où les éléments sont organisés en une structure d'arbre.

#### Recherche de chemin

- Il est également utile pour les applications qui impliquent la recherche de chemins entre les enregistrements de données. Les bases de données basées sur le modèle en réseau permettent de naviguer facilement entre les enregistrements en suivant les relations.

#### Stockage d'informations complexes

- Il peut être utilisé pour stocker des informations complexes telles que des schémas de données et des catalogues.

#### Modélisation des réseaux sociaux

- Il est également utilisé dans les bases de données de réseaux sociaux pour stocker des informations sur les relations entre les individus et les groupes.

- Cependant, le modèle relationnel est généralement plus populaire et plus adapté aux besoins de la plupart des applications de base de données modernes en raison de sa simplicité, de sa flexibilité et de sa capacité à gérer efficacement les données relationnelles.

## CHAPITRE 2

# INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Ce que vous allez apprendre dans ce chapitre :

- Introduire l'informatique décisionnelle
- Présenter un Data Warehouse
- Connaitre l'architecture d'un Data Warehouse
- Savoir les différents types des bases de données
- Avoir une idée générale sur l'ODS (Operational Data Storage)
- Introduire le modèle dimensionnel



10 heures

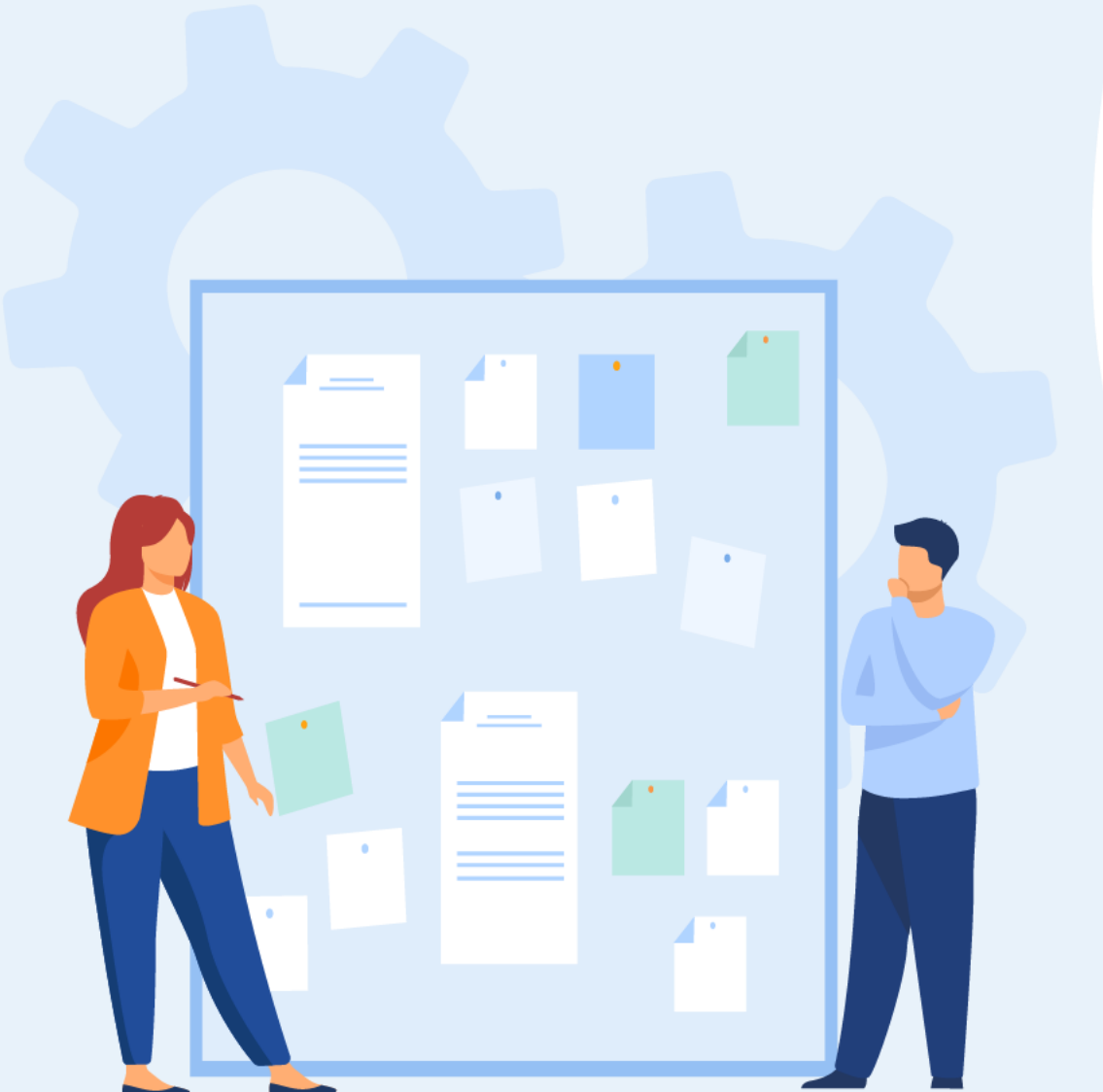


## CHAPITRE 2

# INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE



1. Introduction à l'informatique décisionnelle
2. Présentation générale d'un Data Warehouse
3. Architecture d'un Data Warehouse
4. Types des bases de données
5. Data Warehouse vs ODS (Operational Data Storage)
6. Introduction au Modèle dimensionnel



## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Introduction à l'informatique décisionnelle



#### Notions de base

- L'existence de différentes bases de données du même type et de différents types, ainsi que la présence de plusieurs progiciels intégrés a multiplié la quantité de données à traiter au sein des entités.
- L'analyse de cette grande quantité de données est une phase incontournable dans le processus de la gestion des projets décisionnels.
- La gestion et l'analyse des données massives (big data), afin d'extraire des informations utiles dans la prise de décision, n'est pas une tâche évidente. C'est pourquoi, plusieurs entreprises en quête de croissance s'appuient de plus en plus sur les outils de l'informatique décisionnelle.
- L'instauration d'un **Système d'Information Décisionnel (SID)** semble être comme un défi majeur à relever pour les entreprises.
- **Qu'est-ce qu'un système d'information décisionnel ?**



### Un Système d'Information Décisionnel (SID)



- **Un Système d'Information Décisionnel (SID)** est un système qui repose sur l'informatique décisionnel (ou Business Intelligence) et est adressé aux responsables des entreprises.
- C'est un ensemble des moyens, des outils et des méthodes qui permettent de collecter, consolider, stocker, modéliser, agréger et restituer **les données**, matérielles ou immatérielles, qui peuvent provenir de différentes sources hétérogènes, en vue d'offrir une **aide à la décision**.
- Les sources de données peuvent être des bases de données relationnelles, des fichiers, des services web, etc.. Ces données peuvent être stockées dans **un entrepôt de données (ou Data Warehouse)**.



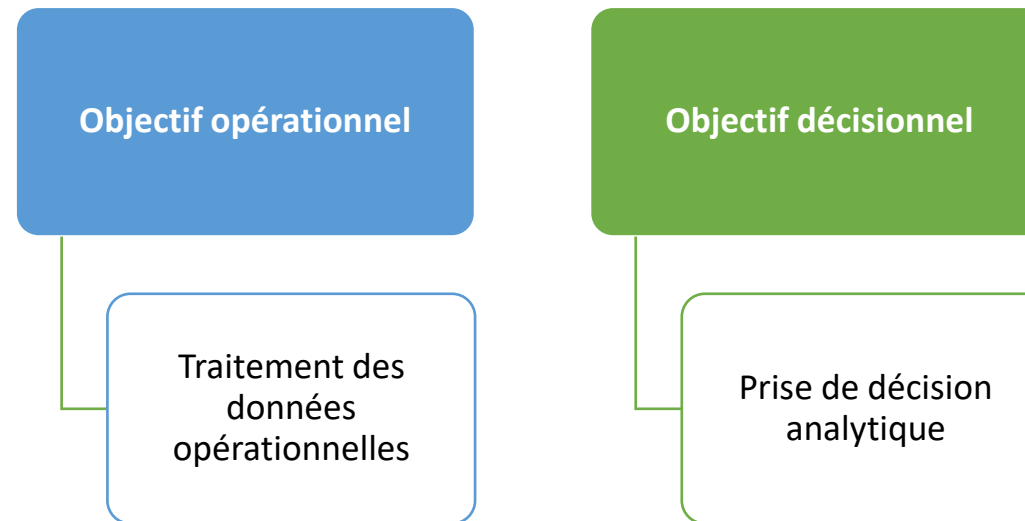
## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Introduction à l'informatique décisionnelle



#### Objectifs et exigences

L'informatique décisionnelle (ou Business Intelligence) à généralement 2 principaux objectifs :





#### Objectifs et exigences : Objectif opérationnel

- **L'objectif opérationnel** (Traitement des données opérationnelles) consiste à utiliser les données dans l'objectif de garantir le bon fonctionnement de l'entreprise, par :
  - La réception des requêtes
  - La réaction envers des demandes
  - L'alimentation du stock
- Le sauvegarde/la conservation opérationnel se réfère à l'OLTP (Online Transactional Processing) qui correspond au traitement transactionnel en ligne.
- Dans la partie **traitement des données opérationnelles**, on est amené à :
  - Traiter, généralement, un seul enregistrement à la fois
  - Enregistrer ou modifier des données
  - Se concentrer seulement sur les données actuelles, sans tenir compte de l'historique

#### Objectifs et exigences : Objectif décisionnel

- **L'objectif décisionnel** (Prise de décision analytique) consiste à utiliser les données dans l'objectif de prendre les bonnes décisions pour le futur et comprendre le fonctionnement de l'entreprise.
- Il consiste à évaluer la performance pour une bonne prise de décision. Réaliser cet objectif revient à répondre à des questions comme :

Quelle est le meilleur produit vendue par l'entreprise ?

Combien de ventes a réalisé l'entreprise cette année par rapport à l'année dernière ?

Comment l'entreprise peut-elle évoluer ?

- Le traitement analytique se réfère à l'OLAP (Online Analytical Processing) qui correspond à l'analyse analytique en ligne.
- Dans la partie **prise de décision analytique**, on est censé à :
  - Analyser et récupérer des millions d'enregistrements en même temps
  - Manipuler des requêtes rapides
  - Donner un sens/un contexte aux données, en les analysant les données au cours du temps (historique) en différents contextes
- Le Data Warehouse vient pour s'adresser aux besoins en données analytiques. Le Data Warehouse est donc un entrepôt de données utilisé pour le reporting (génération des rapports) et l'analyse des données.

## CHAPITRE 2

# INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

1. Introduction à l'informatique décisionnelle
- 2. Présentation générale d'un Data Warehouse**
3. Architecture d'un Data Warehouse
4. Types des bases de données
5. Data Warehouse vs ODS (Operational Data Storage)
6. Introduction au Modèle dimensionnel

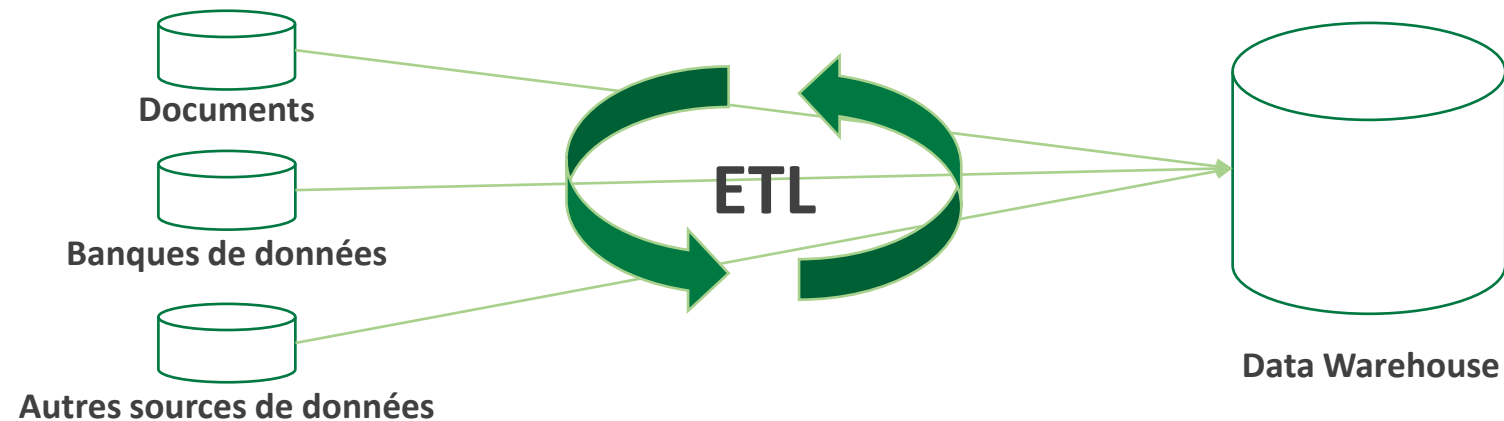


## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Présentation générale d'un Data Warehouse

#### Data Warehouse

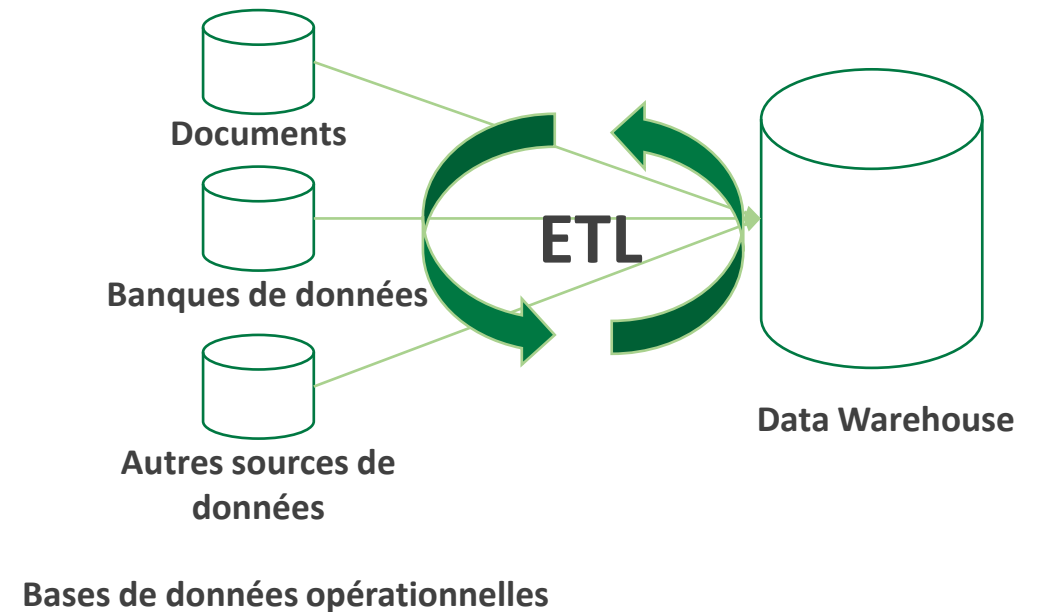
- Un Data Warehouse est une base de données optimisée et utilisée pour les objectifs analytiques. Il est caractérisé par un ensemble de points :
  - Facile à utiliser** : elle doit être très simple à comprendre et à manipuler par les utilisateurs afin de pouvoir analyser les données.
  - Performante dans les requêtes rapides** : Récupérer et traiter une grande quantité de données très rapidement.
  - Analyse de données optimale et facile.**
- Un Data Warehouse est composé de trois composants :



Bases de données opérationnelles

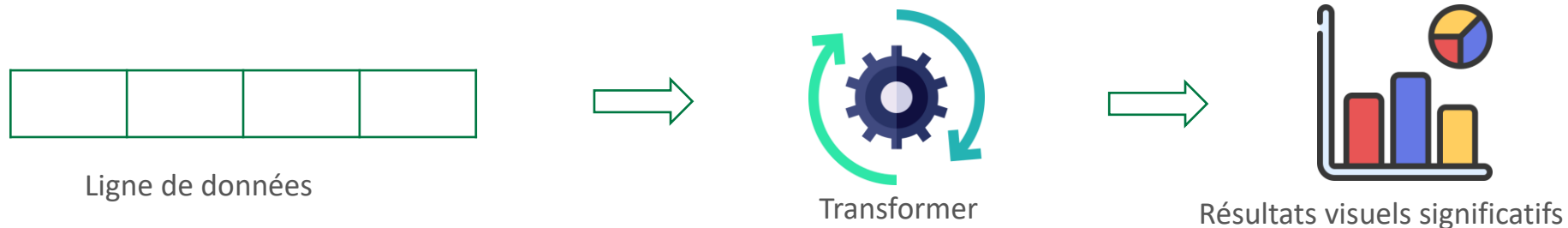
### Data Warehouse

- Les bases de données opérationnelles représentent les différentes sources de données ayant différents formats et structures. Toutes ces sources sont regroupées dans un local centralisé qui est le Data Warehouse.
- Le processus de regroupement et d'enregistrement des différentes sources de données est appelé ETL (Extract Transform, Load) ou Extraire, transformer et charger :
  - L'extraction consiste à extraire les données des différentes sources existants.
  - La transformation consiste à transformer les données de telle sorte elles auront la même structure pour pouvoir les traiter.
  - Le chargement consiste à charger ces données résultantes dans le Data Warehouse.



### Business Intelligence

- Un Business Intelligence (BI) est l'ensemble des stratégies, procédures, technologies et infrastructures qui permettent de donner un sens aux données analysées.
- Pour analyser les données il faut les collecter, les gérer et les sauvegarder afin de créer des rapports (reporting), visualiser les données (Data visualization), explorer les données (data mining) ou fournir des analyses prédictives (predictive analytics)
- En général, le BI permet de trouver une ligne de données, transformer cette dernière pour avoir des résultats visuels significatifs, afin de bien comprendre le contexte et prendre les bonnes décisions dans le futur.



- Le Data Warehouse est un composant très important dans le processus du BI, puisqu'il est l'entrepôt de toutes les données nécessaires qui sont déjà transformées et structurées.

### Data Lake

- Comme le Data Warehouse, un data Lake est aussi une base de données centralisée, mais il ne peut pas remplacer le Data Warehouse, car ils sont différents dans les points suivants :

Data Lake	Data Warehouse
On stocke une ligne de données comme elle est sans traitement	On stocke les données traitées via le processus ETL
Les données sont différentes (fichiers CSV et JSON, images, vidéos, etc.) et leur volume est très grand (Big Data). On utilise différentes technologies	Les données ont la même structure
Les données ne sont pas structurées	Les données sont structurées
Les cas d'utilisation ne sont pas prédéfinis au préalable	Le but est déjà défini et le Data Warehouse est prêt à être utilisé
Il est utilisé par les data scientists	Il est utilisé par les BI et IT
La qualité des données n'est pas assurée	La qualité des données est assurée

- Un Data Warehouse et un data Lake peuvent exister tous les deux dans un même système.

## CHAPITRE 2

# INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

1. Introduction à l'informatique décisionnelle
2. Présentation générale d'un Data Warehouse
- 3. Architecture d'un Data Warehouse**
4. Types des bases de données
5. Data Warehouse vs ODS (Operational Data Storage)
6. Introduction au Modèle dimensionnel



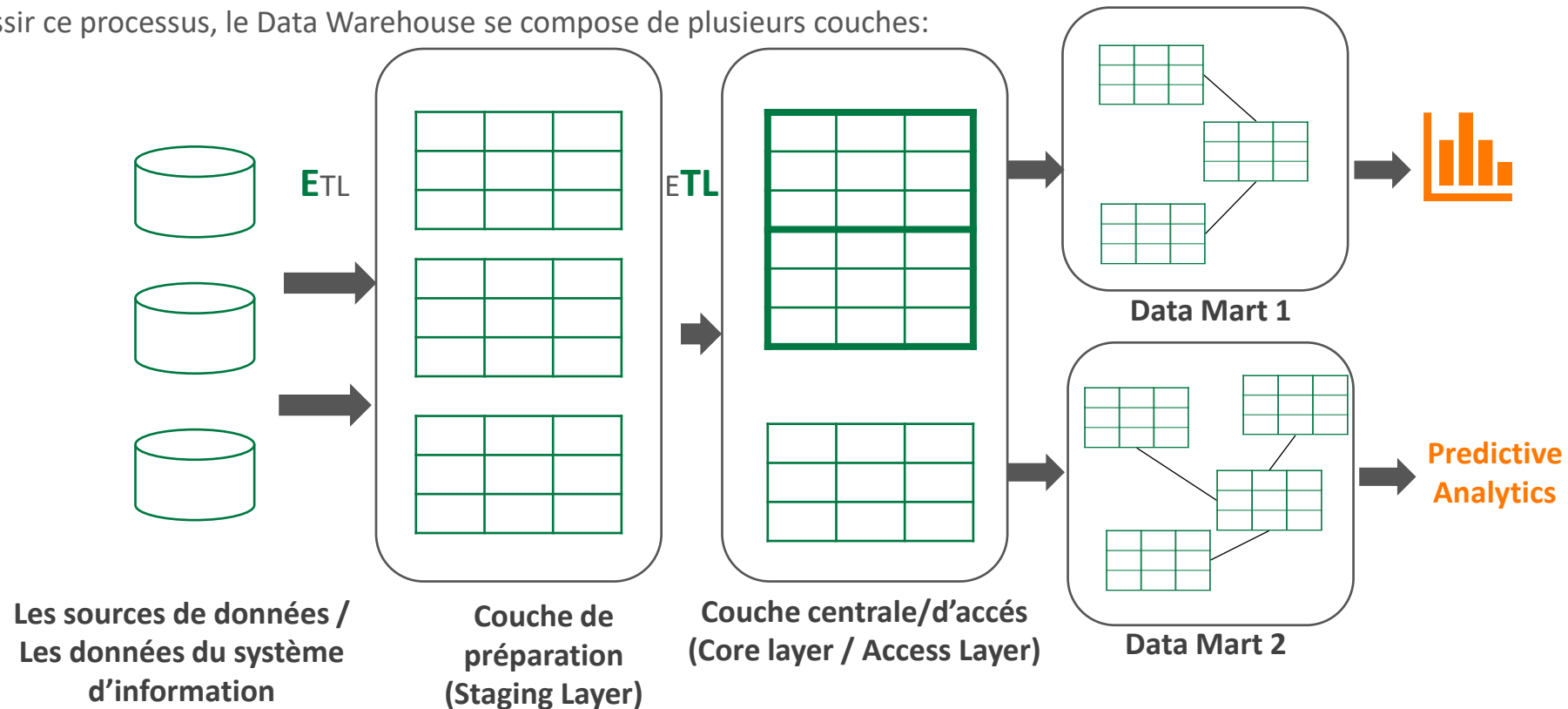


## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Architecture d'un Data Warehouse

#### Les couches d'un Data Warehouse

- Comme on a vu dans sa définition, on dispose d'un ensemble de sources de données qui sont traitées par le processus ETL et enregistrées dans le Data Warehouse.
- Afin de réussir ce processus, le Data Warehouse se compose de plusieurs couches:

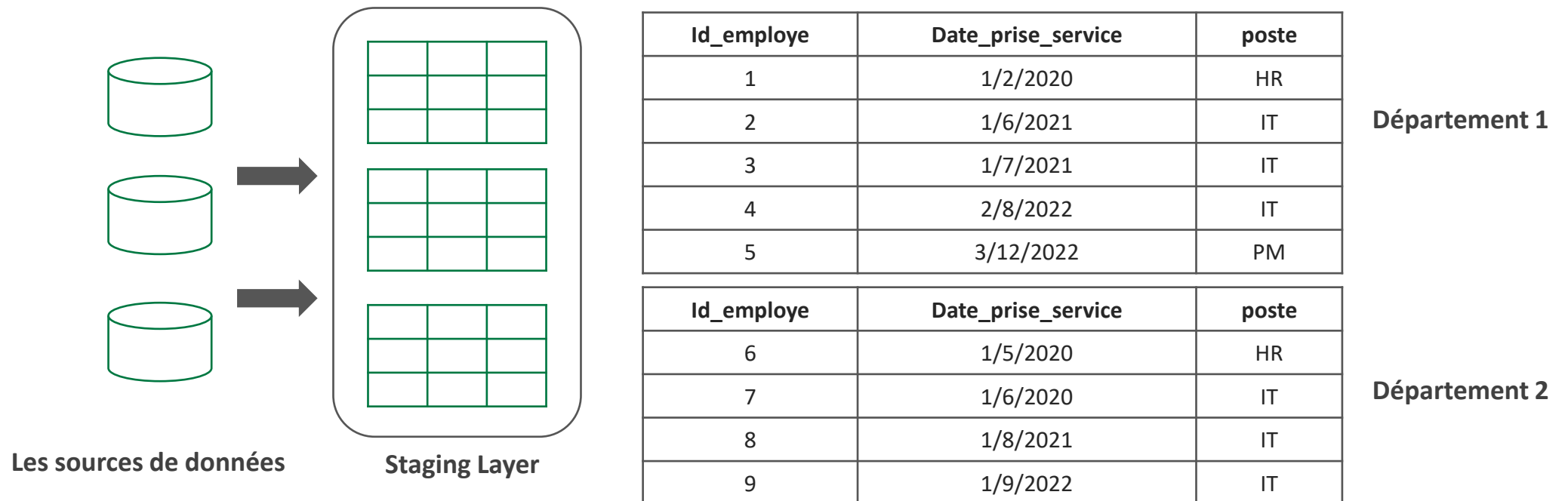


### Les couches d'un Data Warehouse : Staging Layer (couche de préparation)

- On utilise le processus ETL pour extraire les données des différentes sources dans la première couche de préparation (Staging Layer).
- Cette couche permet juste d'extraire les tables comme elles sont, sans faire aucune transformation majeure.

Exemple :

- On extrait les tableaux des employés qui existent dans deux différents départements.




### Les couches d'un Data Warehouse : Staging Layer

- Si les tableaux extraits se ressemblent, on peut faire quelques petites transformations, comme la combinaison, tout en adaptant les intitulés des colonnes et des enregistrements si nécessaire.
- Dans certains cas on peut ajouter une autre couche juste après la couche de préparation afin de nettoyer les données, toujours via l'ETL.
- Dans l'exemple précédent, on peut combiner les deux tables en un seul.



#### Les couches d'un Data Warehouse : Staging Layer

- La question qui se pose par rapport à cette couche : **Pourquoi on utilise la couche de préparation au lieu de traiter les données existants directement à partir de leurs sources?** 
- L'application directe des requêtes sur les données des systèmes opérationnels, qui tournent d'une manière permanente, peut causer un ralentissement ou un arrêt du système. Cette couche permet d'interroger les systèmes le moins de temps possible, pour un accès rapide en lecture, une extraction des données et leur enregistrement dans les tables appropriées.
- Les systèmes opérationnels contiennent différents formats de données (bases de données, fichiers JSON, fichiers CSV et d'autres types de données). La couche de préparation permet de déplacer tous ces données dans une base de données relationnelle, de telle sorte à ne voir que des tables.

### Les couches d'un Data Warehouse : Staging Layer

- Afin de comprendre le fonctionnement de la couche de préparation, on va prendre un exemple pratique.
- On suppose avoir une table de ventes comme données.
  1. On lit et extrait rapidement les données à partir des systèmes opérationnels
  2. On applique les étapes de transformation et chargement dans le Data Warehouse
  3. On tronque (nettoyage) la couche de préparation après chaque cycle. C'est une couche intermédiaire

id	date	produit
1	1/6/2022	Produit 1
2	1/6/2022	Produit 2
3	1/6/2022	Produit 3
4	1/6/2022	Produit 4
5	1/6/2022	Produit 5

Sources des données



id	date	produit
1	1/6/2022	Produit 1
2	1/6/2022	Produit 2
3	1/6/2022	Produit 3
4	1/6/2022	Produit 4
5	1/6/2022	Produit 5

Lecture est extraction des données  
dans la couche de préparation  
temporaire

### Les couches d'un Data Warehouse : Staging Layer

- On suppose qu'après quelques jours il y aura des données additionnelles dans la table des ventes (enregistrements 6 et 7). On a besoin de savoir quelles sont les nouvelles données ajoutées. C'est pourquoi on doit avoir une **colonne delta** qui permet vérifier si les données sont nouvelles. Elle peut être la colonne **id** si c'est un nombre auto incrémental ou ascendant.
- Dans notre exemple, on sait que le dernier id était 5, donc tous les enregistrements ayant un id supérieur à soit sont nouveaux.
- Dans le cas où les valeurs de l'id ne sont pas ascendantes, on peut utiliser la colonne date comme une colonne delta.
- Une fois l'extraction est faite, on peut appliquer les autres étapes et ajouter les nouvelles valeurs au Data Warehouse.

id	date	produit
1	1/6/2022	Produit 1
2	1/6/2022	Produit 2
3	1/6/2022	Produit 3
4	1/6/2022	Produit 4
5	1/6/2022	Produit 5
6	1/7/2022	Produit 6
7	1/7/2022	Produit 7

Sources des données



id	date	produit
6	1/7/2022	Produit 6
7	1/7/2022	Produit 7

Lecture et extraction des données

### Les couches d'un Data Warehouse : Staging Layer

- Les transformations faites sur les données extraites peuvent être problématiques et causer quelques erreurs d'où la nécessité de revenir en arrière et commencer dès le départ puisque la couche de préparation est temporaire.
- On peut donc avoir une autre alternative qui est une couche de préparation persistante, qui n'est jamais tronquée. Au lieu de recommencer à partir des systèmes opérationnels qu'on ne veut pas interroger régulièrement, on peut revenir à cette couche facilement.
- L'utilisation d'une couche de préparation persistante n'est pas toujours conseillée.

Sources des données

id	date	produit
1	1/6/2022	Produit 1
2	1/6/2022	Produit 2
3	1/6/2022	Produit 3
4	1/6/2022	Produit 4
5	1/6/2022	Produit 5
6	1/7/2022	Produit 6
7	1/7/2022	Produit 7



id	date	produit
1	1/6/2022	Produit 1
2	1/6/2022	Produit 2
3	1/6/2022	Produit 3
4	1/6/2022	Produit 4
5	1/6/2022	Produit 5
6	1/7/2022	Produit 6
7	1/7/2022	Produit 7

Lecture est extraction des données  
dans la couche intermédiaire  
**persistante**

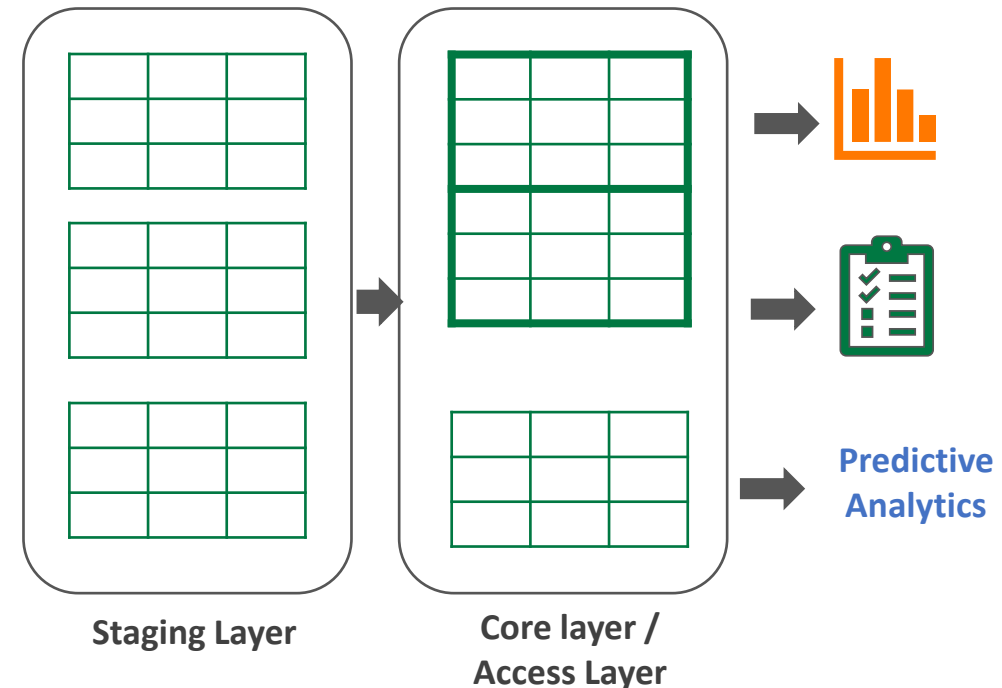
## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Architecture d'un Data Warehouse

#### Les couches d'un Data Warehouse : Core Layer

- On utilise toujours l'ETL pour copier les données, de la couche de préparation à la couche centrale, qui est considérée aussi comme le Data Warehouse lui-même, pour intégrer les transformations nécessaires.
- Cette couche est généralement utilisée par les utilisateurs ou les applications afin de générer des rapports via le data mining\* ou l'analyse prédictive.

\* Le data mining est le processus d'exploration et d'analyse de grandes quantités de données pour découvrir des motifs, des relations et des tendances cachées, permettant ainsi de prendre des décisions éclairées et de générer des connaissances précieuses.



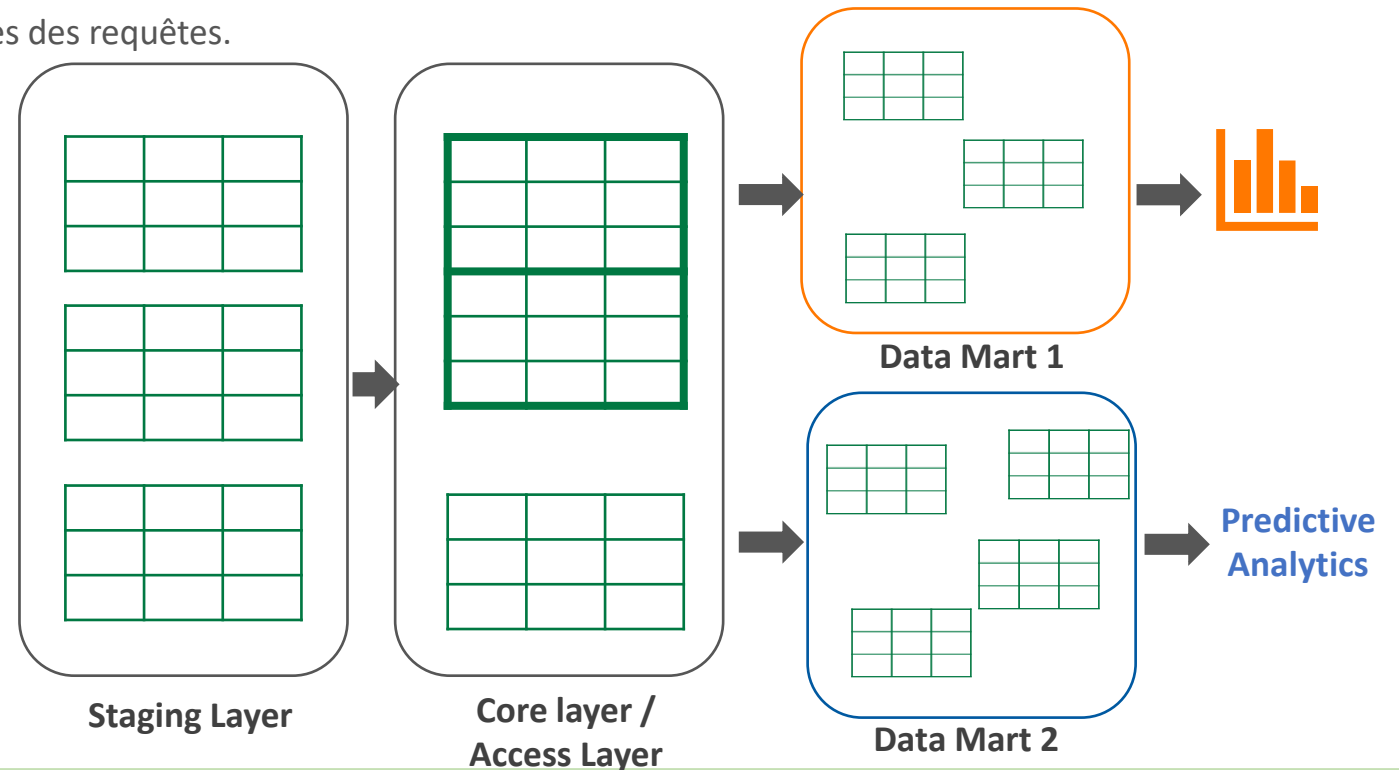


## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Architecture d'un Data Warehouse

#### Les couches d'un Data Warehouse : Data Mart

- Lorsqu'on dispose d'un grand Data Warehouse, qui est constitué d'un très grand nombre de tables et différents cas d'utilisation, on inclut les **data marts** comme couche entre la couche centrale et les résultats.
- Un data mart prend simplement un ensemble de tables du Data Warehouse (du noyau) qui sont pertinentes pour un cas d'utilisation très spécifique. Cela permet à augmenter les performances des requêtes.



**Remarque :** les data marts ne sont pas toujours nécessaires

#### Les couches d'un Data Warehouse : Data Mart

- Un data mart est un sous-ensemble d'un data warehouse (la couche centrale).
- Les données dans un data mart sont modélisées avec un **modèle dimensionnel**. Même dans un Data Warehouse (le noyau) les données peuvent être modélisées avec un modèle dimensionnel.
- Un data mart peut être, à son tour aussi, agrégé, comme un Data Warehouse.
- Les data marts augmentent la convivialité des données puisque on se focalise sur les données pertinentes pour un cas d'utilisation donné.
- Ils peuvent être utilisés pour augmenter la performance puisqu'ils utilisent les modèles dimensionnels, donc on peut utiliser une technologie spécifique à ce modèle :
  - Power BI : In-memory databases
  - Dimensional cubes, etc.

**Remarque :** L'utilisation des data marts est optionnelle. Cela dépend des cas d'utilisation

## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Architecture d'un Data Warehouse



#### Les couches d'un Data Warehouse : Data Mart

- Différents outils :
  - Visualisation des données (par exemple Power BI)
  - L'analyse prédictive avec d'autres outils
  - Etc.
- Différents départements, avec différents cas d'utilisation :
  - Département des ventes
  - Département de la finance
  - Département de commerce
  - Département de Management
  - Etc.
- Différentes régions

## CHAPITRE 2

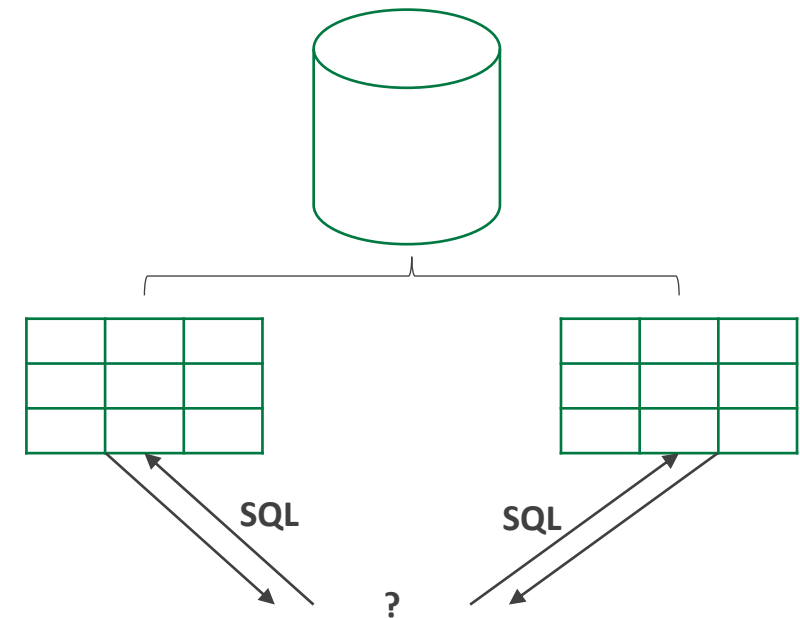
# INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

1. Introduction à l'informatique décisionnelle
2. Présentation générale d'un Data Warehouse
3. Architecture d'un Data Warehouse
- 4. Types des bases de données**
5. Data Warehouse vs ODS (Operational Data Storage)
6. Introduction au Modèle dimensionnel



#### Bases de données relationnelles : Rappel

- Une base de données relationnelles est une simple base de données où les données sont stockées dans des tables. Les données sont structurées sous forme de colonnes et de lignes.
- On utilise le langage SQL pour interroger une base de données relationnelle
  - SELECT pour l'affichage
  - UPDATE pour la modification
  - DELETE pour la suppression
  - INSERT pour l'ajout
- Dans les bases de données relationnelles, on identifie les données dans une table par des clés primaires uniques.
- Les tables sont mises en relations via des clés étrangères.
- Afin de combiner plusieurs tables, on utilise les jointures JOIN.



## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Types des bases de données



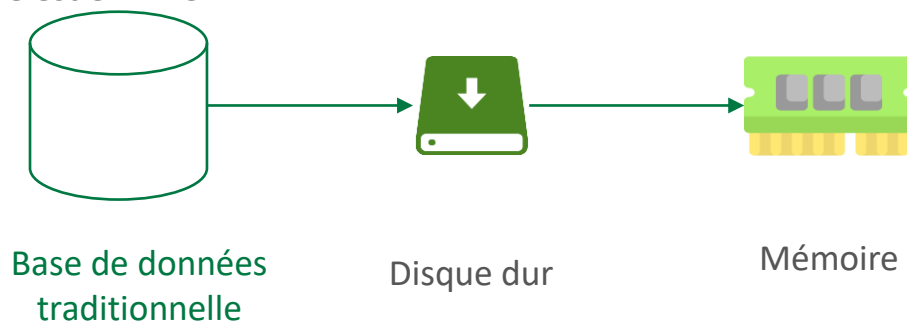
#### Bases de données relationnelles : Rappel

- Les bases de données sont généralement implémentées dans un système d'exploitation (en local). De ce fait on n'interroge qu'une seule au même temps.
- Puisque les tables sont liées par des relations, on peut interroger plusieurs tables, en utilisant l'OLAP\* (Online Analytical Processing).
- En plus des bases de données relationnelles, il y a d'autres types comme les bases de données en mémoire (In-Memory Databases)

\* ça sera expliqué par la suite

#### Bases de données en mémoire

- Les bases de données en mémoire (In-Memory Databases) sont hautement optimisées pour la performance des requêtes.
- Elles sont utilisées pour des fins analytiques ou n'importe quel cas d'utilisation se servant d'un grand nombre de requêtes, et demandant des réponses rapides.
- Elles sont généralement utilisées dans les data marts.
- Cette technologie est indépendante de la structure des données (relationnelle ou non-relationnelle). Elle a des solutions pour les deux cas.
- Les données sont généralement stockées dans un disque dur. Lorsque les données sont interrogées par une requête, ces dernières sont chargées en mémoire avec un temps de réponse. Durant cette étape, on a un temps d'attente pour que le résultat de la requête soit retourné, ce qui n'est pas optimal, si on demande une performance élevée.
- Les bases de données en mémoire n'utilisent pas le disque. Toutes les données sont stockées dans la mémoire et donc le temps de réponse du disque est éliminé.



### Bases de données en mémoire

- Comme pour les bases de données relationnelles, les bases de données en mémoire disposent d'un ensemble de technologies, algorithmes et méthodes utilisés :
  - Stockage par colonnes (columnar storage)
  - Parallélisation des requêtes (parallel query plans)
  - Etc.
- Les bases de données en mémoire font face à plusieurs challenges :

**Durabilité** : On risque de perdre toutes les données si la base de données perd l'alimentation ou se redémarre. Afin de garder la durabilité, on doit ajouter d'autres technologies comme les clichés (snapshots) ou les images qui représentent un état spécifique de la base de données, afin d'y retourner en cas de perte des données.

**Coût** : Le stockage d'une grande quantité de données dans la mémoire est très coûteux.

Même les bases de données traditionnelles essaient d'optimiser l'utilisation du disque dur.

Si on veut adapter ce type de bases de données pour les data marts, on ne doit charger que les données pertinentes pour un cas d'utilisation très spécifique.



### Bases de données en mémoire : exemples



SAP HANA



MS SQL Server In-Memory Tables



Oracle In-Memory



Amazon MemoryDB  
(Services Cloud)

- Tous ces produits peuvent être utilisés dans les data marts si on cherche à avoir des requêtes de bonnes performances.
- En plus des bases de données en mémoire, les **cubes** peuvent être aussi utilisés dans les data marts pour une bonne performance.

## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Types des bases de données

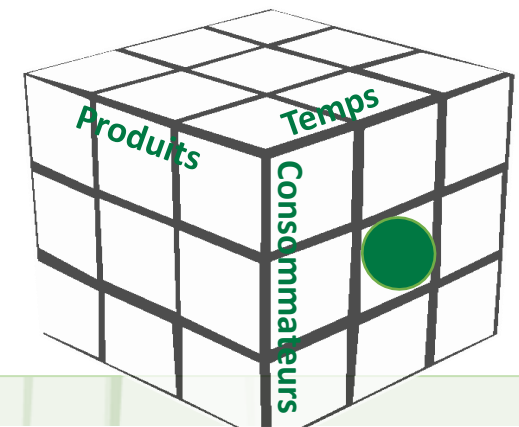


#### Cubes OLAP

- Les cubes OLAP (OnLine Analytical Processing) sont aussi une méthode alternative pour augmenter la performance des data marts.
- Les Data Warehouses traditionnels se basent sur les bases de données relationnelles (ROLAP : Relational OLAP).
- Dans un cube, les données ne sont pas organisées dans des tables liées avec de relations, mais avec les dimensions. On ne parle pas d'une base de données relationnelle. On parle des bases de données multidimensionnelles (MOLAP : Multidimensional OLAP).
- Dans les bases de données multidimensionnelles, les données sont organisées sous forme de tableaux multidimensionnels, c'est-à-dire des tableaux de tableaux (des cubes). Un cube organise les données.
- Les cubes s'utilisent pour des fins analytiques avec une bonne performance. Ils peuvent être utilisés dans différentes solutions BI.

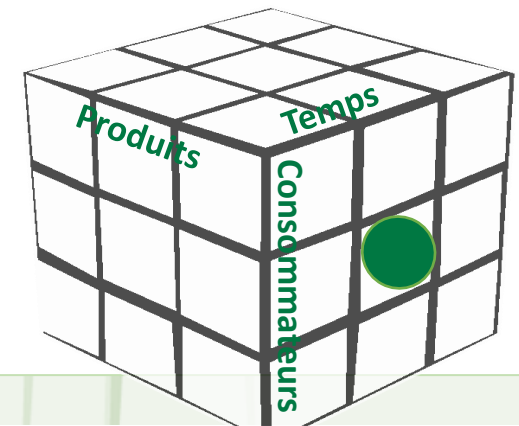
### Cubes OLAP : Structure des données

- Prenons l'exemple d'analyse multidimensionnelles des données des ventes. L'exemple présenté se compose de 3 dimensions, juste pour voir visuellement la représentation des données, mais on peut avoir plus de 3 dimensions.
- Dans cet exemple, on a 3 dimensions : produits, temps et consommateurs et on veut analyser les ventes. On peut utiliser ces cellules pour diviser les données. On peut utiliser par exemple l'intersection pour un consommateur dans un certain temps, afin de récupérer un point spécifique de donnée calculée qui représente la quantité de ventes pour ce cas de figure.
- Un des avantages des cubes est qu'ils offrent des valeurs précalculées (agrégées).
- Contrairement aux bases de données relationnelles, le langage SQL ne peut pas être utilisé pour les cubes, c'est le langage MDX (Multi Dimensional eXpression) qui s'utilise dans ce cas. C'est un langage de requête développé par Microsoft qui est le plus utilisé pour les cubes.



### Cubes OLAP : Recommandations

- Ces cubes multi dimensionnels doivent être construits pour un cas d'utilisation spécifique, c'est pourquoi on les utilise dans les data marts. Les cubes sont bénéfiques lorsqu'idéalement on n'a pas plusieurs dimensions.
- Les cubes sont très pratiques pour les requêtes interactives avec des hiérarchies.
- L'utilisation des cubes dans les data marts est optionnelle. On peut tout simplement utiliser les bases de données relationnelles, mais tout en les organisant afin d'avoir une bonne performance.
- Parmi les outils utilisés pour l'organisation des bases de données relationnelles, on cite le « schéma en étoile ».
- Toujours dans le but d'optimiser les performances, elles existent des méthodes alternatives du stockage des données :
  - Modélisation tabulaire (SSAS)
  - ROLAP
  - Stockage en colonnes
  - Traitement parallèle
- Toutes ces technologies peuvent être utilisées dans les data marts afin d'augmenter la performance.



## CHAPITRE 2

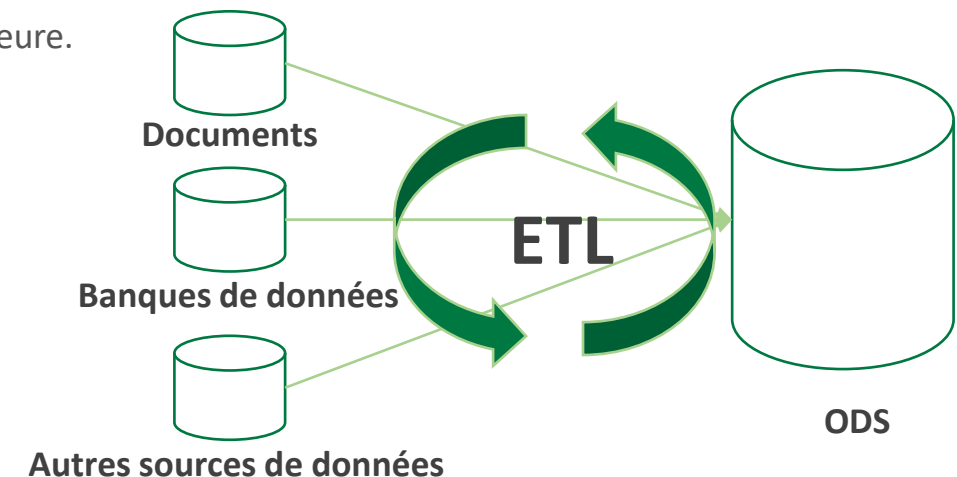
# INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

1. Introduction à l'informatique décisionnelle
2. Présentation générale d'un Data Warehouse
3. Architecture d'un Data Warehouse
4. Types des bases de données
- 5. Data Warehouse vs ODS (Operational Data Storage)**
6. Introduction au Modèle dimensionnel



### ODS (Operational Data Storage)

- La différence entre un ODS et un Data Warehouse n'est pas claire, car l'architecture de l'ODS ressemble à celle du Data Warehouse.
- Comme pour un Data Warehouse, dans un ODS on dispose d'un ensemble de données de différents types qu'on veut intégrer dans une seule base de données (ODS) en utilisant toujours l'ETL.
- L'ODS est utilisé pour des prises de décision opérationnelles, c'est ce qui rend le processus différent par rapport à un Data Warehouse, puisqu'il n'est pas utilisé pour des prises de décisions analytique ou stratégique.
- Puisque le seul cas d'utilisation d'un ODS est l'opérationnel, il n'a pas besoin d'une longue historique. L'état actuel des données peut être suffisant. Idéalement, cet état actuel doit être retourné immédiatement en temps réel à partir des systèmes sources, contrairement à un Data Warehouse où les données peuvent être mises à jour une fois par jour ou par heure.

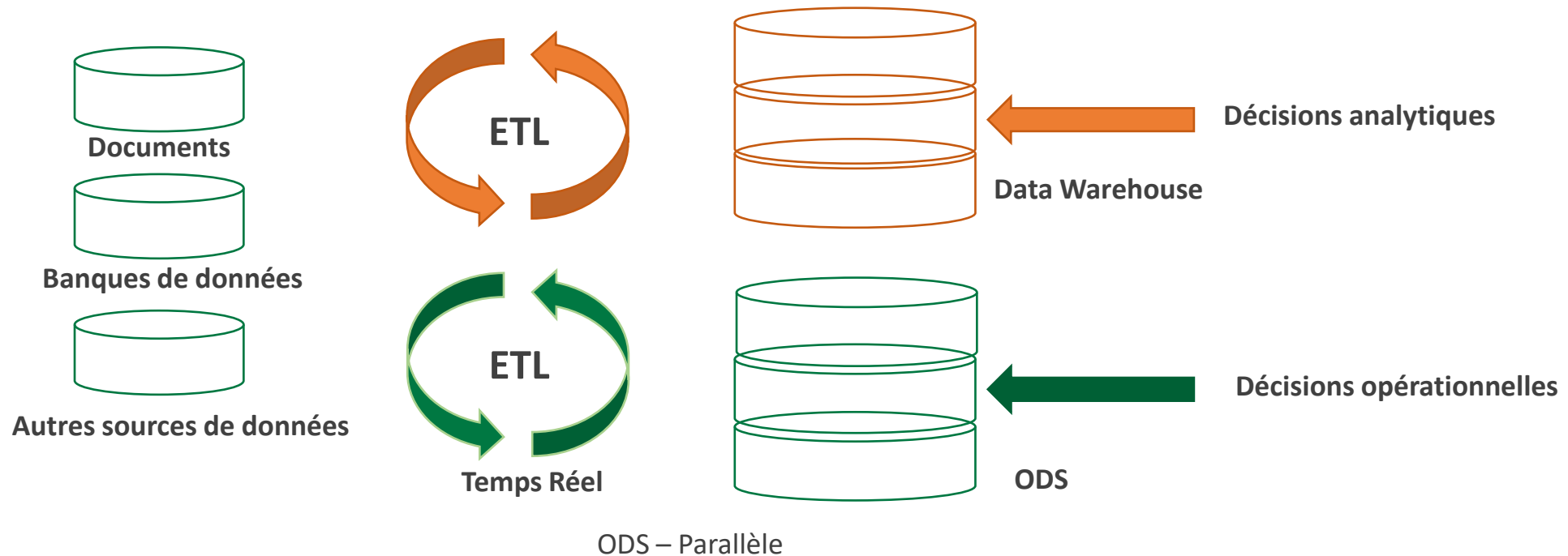


## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Data Warehouse vs ODS (Operational Data Storage)

#### ODS (Operational Data Storage)

- Si on veut décider si on peut donner à un consommateur un crédit ou non. Dans ce cas on a besoin de consulter ses données comme elles sont dans ces systèmes opérationnels dans un temps réel afin de prendre des décisions plus précises. De plus on n'a pas besoin d'une longue historique. L'utilisation d'un ODS peut être adéquate.
- On peut avoir en plus d'un Data Warehouse, un ODS **en parallèle** dans une même structure, avec un autre ETL (temps réel).

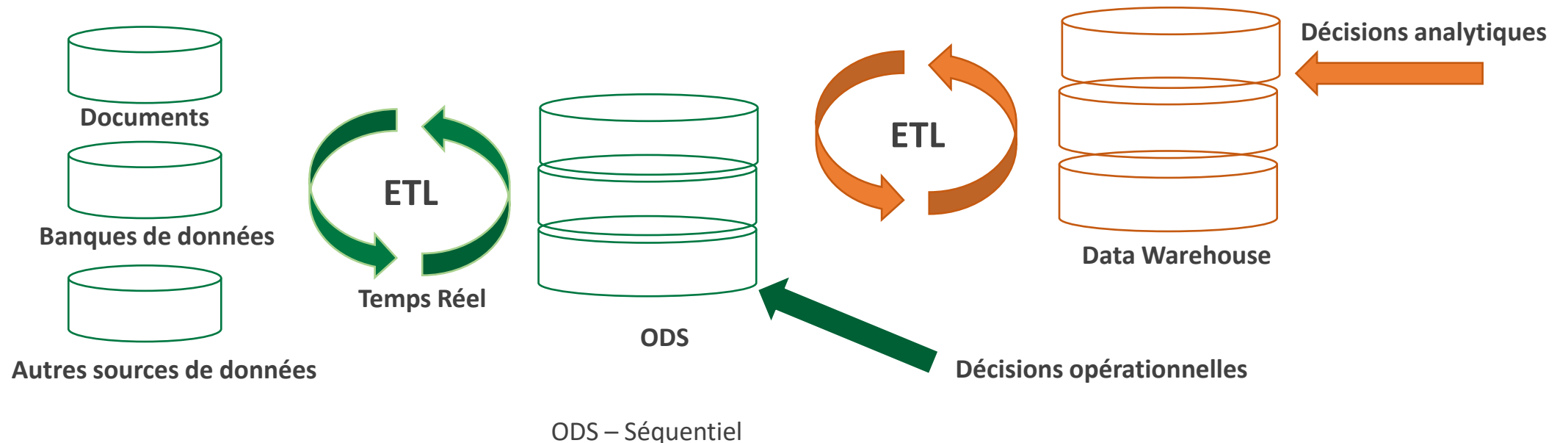


## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Data Warehouse vs ODS (Operational Data Storage)

#### ODS (Operational Data Storage)

- Afin de mettre l'ODS et le Data Warehouse dans un même organisme, une autre solution est possible. On peut les mettre d'une manière séquentielle au lieu de les mettre en parallèle.
- Dans ce cas, on garde toujours l'ODS, mais on ajoute une couche de l'ETL pour le Data Warehouse au dessus de l'ODS, puisque le premier ETL a déjà fait le grand travail.
- Cet ODS peut être vu comme la source de données, ou la couche de préparation du Data Warehouse.





## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Data Warehouse vs ODS (Operational Data Storage)



#### ODS (Operational Data Storage)

- L'ODS a devenu moins pertinent, surtout à cause des nouvelles performances du matériel (ETL et bases de données plus rapides). Les données peuvent être chargées très rapidement sans avoir besoin d'un ODS.
- L'apparition des technologies Big Data a aussi baissé la pertinence des ODS, puisqu'elles ont rendu le traitement d'une très grande quantité de données plus rapide.

## CHAPITRE 2

# INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

1. Introduction à l'informatique décisionnelle
2. Présentation générale d'un Data Warehouse
3. Architecture d'un Data Warehouse
4. Types des bases de données
5. Data Warehouse vs ODS (Operational Data Storage)
- 6. Introduction au Modèle dimensionnel**



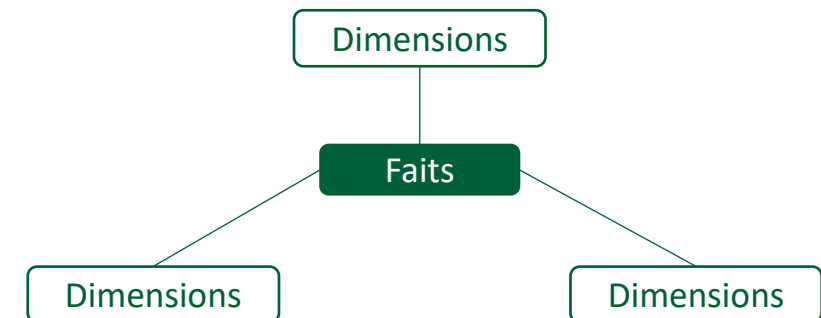
## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Introduction au Modèle dimensionnel



#### Définition

- Un modèle dimensionnel fait référence à des méthodes permettant l'organisation des données d'une manière spécifique. Ceci est utilisé généralement dans un Data Warehouse, puisque ce dernier a des exigences spécifiques sur les données (facilité d'utilisation, performance, ...).
- Dans un modèle dimensionnel, toutes les données sont organisées sous forme de **faits** et de **dimensions**.
- Un **fait** représente tout ce qui est généralement mesuré (par exemple un gain qui peut être cumulé).
- Les **dimensions** donnent un contexte supplémentaire à ces mesures (par exemple : un mois, une période, une catégorie du produit, etc.). Avec ces dimensions, on peut retourner les faits (les mesures) dans un contexte avec des résultats significatifs (par exemple : analyser les gains par année ou par catégorie).
- Un fait est généralement modélisé par une table au milieu entouré par un ensemble de dimensions regroupées autour de ce fait. Ces différentes dimensions peuvent être utilisées pour analyser les données et les mesures dans la table de fait.



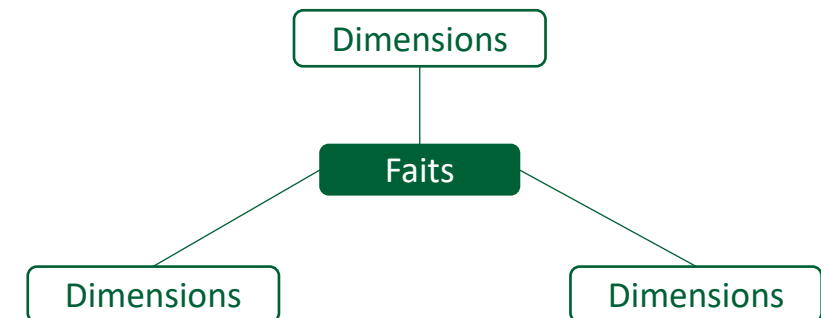
## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Introduction au Modèle dimensionnel



#### Utilité

- Récupération rapide des données, cela est lié à la fois aux performances et à la convivialité
- La modélisation dimensionnelle est utilisée dans les Data Warehouse, les cubes OLAP, etc.

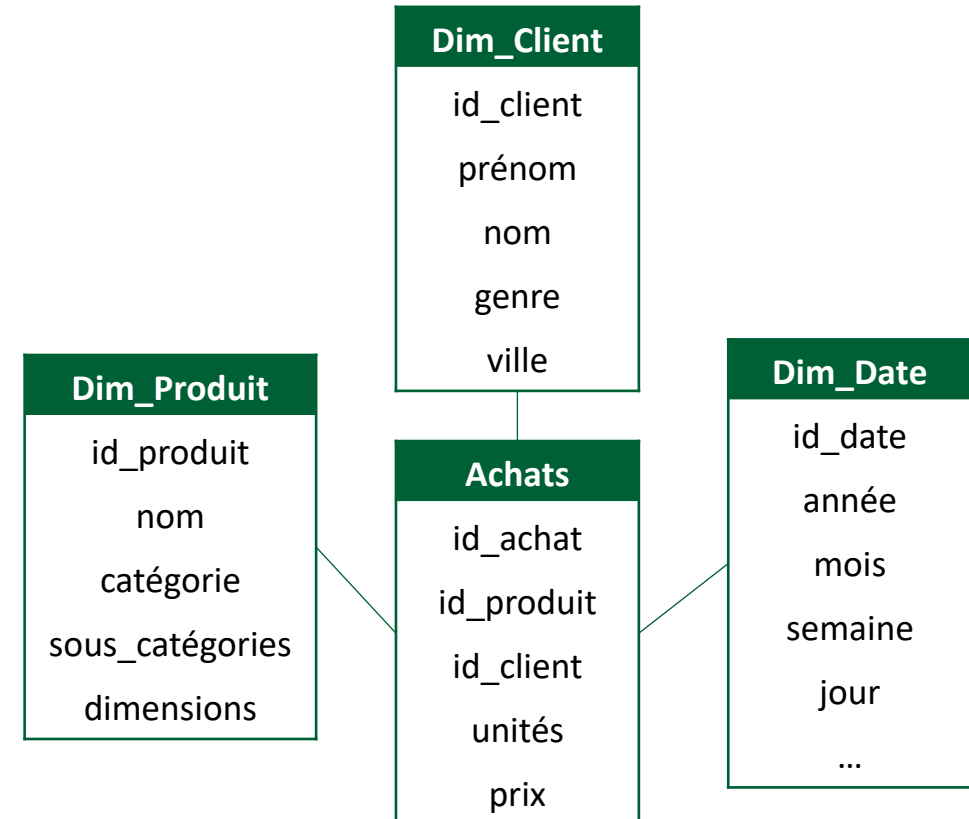


### Table de faits

- Comme nous avons vu dans un schéma en étoile, on dispose d'une table de faits au milieu un ensemble de tables de dimensions qui entourent ce dernier.
- La table de fait est la base d'un Data Warehouse (par exemple), parce qu'elle contient les mesures clés d'une entité.
- Ces faits sont les résultats qu'on veut généralement agréger et analyser par les dimensions dont on dispose.

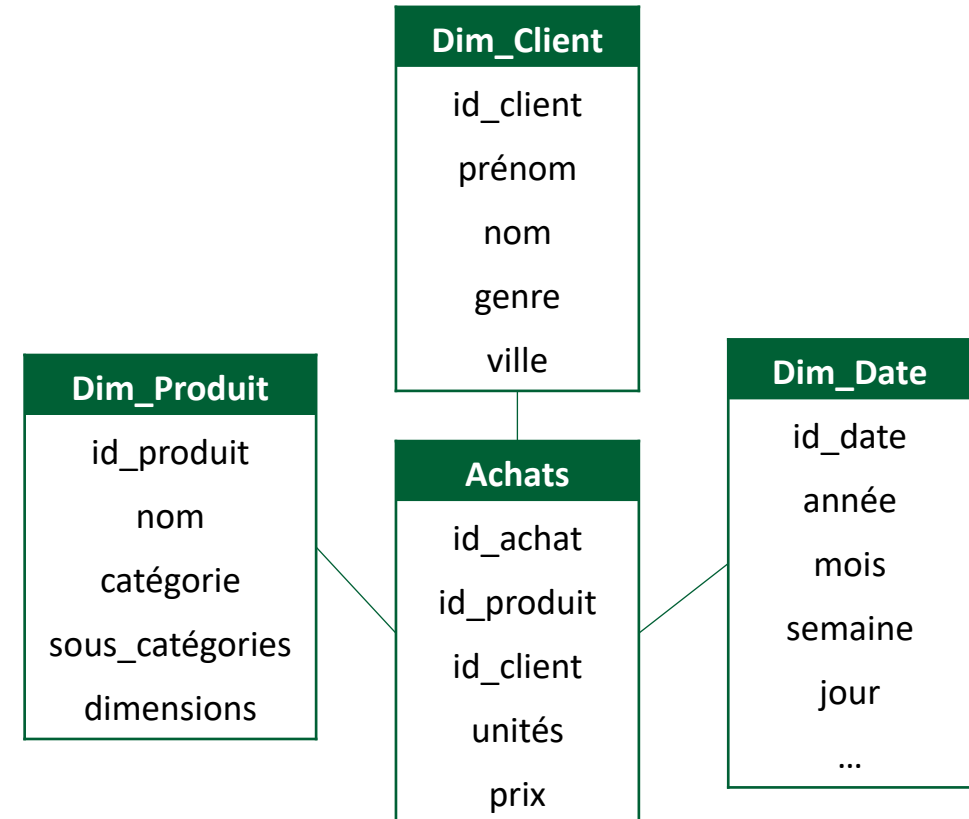
### Exemple :

- La table des achats représente la table de fait, tandis que les tables de dimensions, dans cet exemple, sont les produits, les clients et la date.



#### Table de faits : Caractéristiques

- Les faits sont généralement additives, c'est-à-dire qu'on peut les additionner. On peut ajouter des nombres tout en gardant un sens aux résultats (le nombre d'unités par exemple est calculable).
- Contrairement aux dimensions, les faits ne sont pas descriptifs, ils sont mesurables.
- Un fait peut être événementiel ou transactionnel (exemple : Un achat peut être traité comme une transaction ou un autre événement qui se produit dans un temps spécifique).
- On trouve des fois le temps ou la date comme une colonne incluse dans la table de faits. Ces données ne sont pas des faits eux même, mais elles sont incluses dans la table de faits.



### Table de faits

- Une table de fait se compose d'une clé primaire (comme dans les bases de données relationnelles), multiples clés étrangères qui font références aux dimensions et les faits eux-mêmes qui sont les mesures clés.
- Une table de faits est définie par ce qu'on appelle le **grain**. Le grain est le niveau le plus atomique d'un fait.

### Exemple

- Prenons la table suivante, on remarque qu'on a un profit pour chaque région et date.
- On a un profit dans une ligne pour une région spécifique dans un temps spécifique. Ceci donc est le niveau le plus atomique. C'est le grain de cette table.

id	id_date	id_region	profit (DH)
1	20092022	1	200
2	20092022	2	150
3	20092022	10	900
4	20092022	4	300
5	20092022	3	250

## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Introduction au Modèle dimensionnel



#### Table de dimensions

- Les dimensions servent à catégoriser les faits afin d'obtenir un contexte significatif à nos mesures. Si par exemple, on n'a qu'un nombre total des unités vendues, on n'a pas vraiment des résultats significatifs.
- Le caractère de ces dimensions est plus **descriptif**. On ne peut pas le mesurer mais on peut l'utiliser pour décrire un fait (un produit, une catégorie, ...). Ceci aide à supporter les faits et **analyser**, **filtrer** les groupes et **labéliser** les données.
- Afin de distinguer entre les faits et les dimensions, voici les caractéristiques communes des dimensions :

Les tables de faits sont agrégées (calculables : +, -, Etc.) et donc numériques, tandis que les dimensions peuvent être numériques sauf qu'ils ne peuvent plus être agrégées.

Leur caractère est descriptif, contrairement à celui des faits qui est mesurable.

Dans la table de faits, on a toujours des mesures qui changent (des valeurs), par contre aux dimensions qui sont plus statiques.



### Table de dimensions

- Comme une table de faits, la table de dimension se compose aussi d'une clé primaire et les dimensions. Elle peut même comporter des clés étrangères. Cela devient important lorsqu'on parle **d'un schéma en flocons**.
- Les dimensions peuvent représenter des personnes (employés, consommateurs, managers), des produits, des lieux (régions, villes, adresses), des temps, etc.

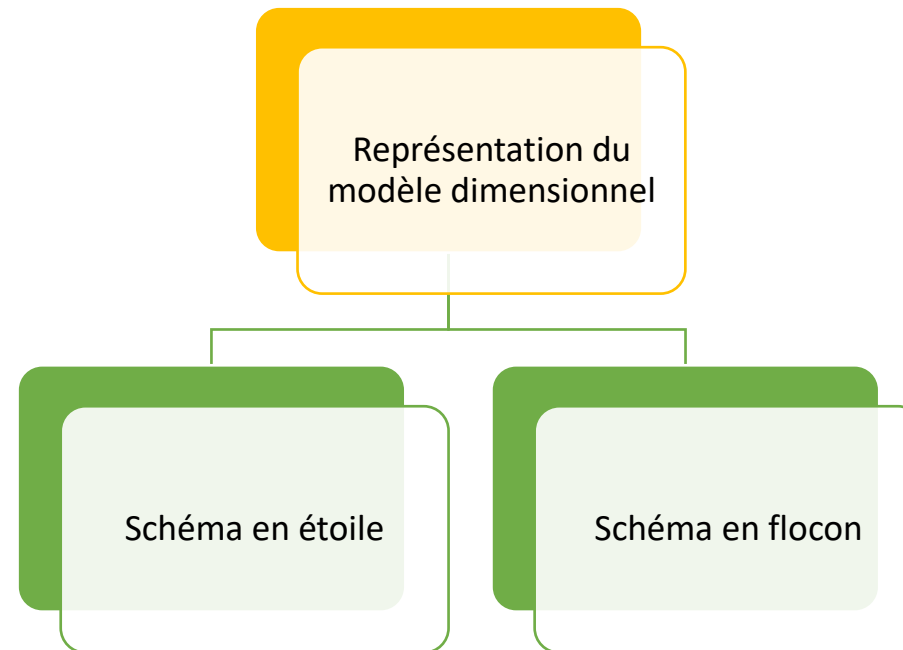
### Exemple

- Prenons la table des consommateurs suivante comme une table de dimension. On a id\_consommateur comme une clé primaire et les différents dimensions (prénom\_consommateur, nom\_consommateur et email\_consommateur).
- Contrairement aux faits, les dimensions peuvent être changées lentement (de temps à autre).

id_consommateur	prénom_consommateur	nom_consommateur	email_consommateur
1	Malak	Grini	mgrini@gmail.com
2	Sofia	Falahi	falahiS@gmail.com
3	Karim	Fassi	KarimFassi@gmail.com
4	Khalid	Chouaib	Chouaib111@gmail.com
5	Fatima Zahra	Tlemsani	FZTlemsani@gmail.com

### Table de dimensions

- Comme nous avons vu dans la première représentation de la modélisation dimensionnelle, les faits et les dimensions sont présentées sous forme d'une étoile. On parle du schéma en étoile.
- En plus du schéma en étoile le modèle dimensionnel peut être représenté par un autre schéma appelée « **schéma en flocon** »



### Schéma en étoile

- Le schéma en étoile est le schéma le plus important dans un Data Warehouse, et plus précisément dans les data marts.
- Comme on a déjà vu dans un modèle dimensionnel, on arrange et structure les données sous formes de faits et de dimensions. Si on reprend le même exemple des achats, on a la table des achats qui contient tous les faits importants, et on crée des relations avec les dimensions (afin de joindre cette table avec les tables de dimensions), en utilisant les clés étrangères et primaires.
- Comme dans le modèle relationnel, on parle de différents types d'associations (relations) (1:n, 1:1, etc.). Dans cet exemple, et généralement entre une table de fait et une autre de dimension, on a une relation 1:n, 1 coté dimensions et n coté faits. Chaque produit (dimension) peut faire l'objet de plusieurs achats, tandis qu'un achat ne se fait que sur un seul produit.

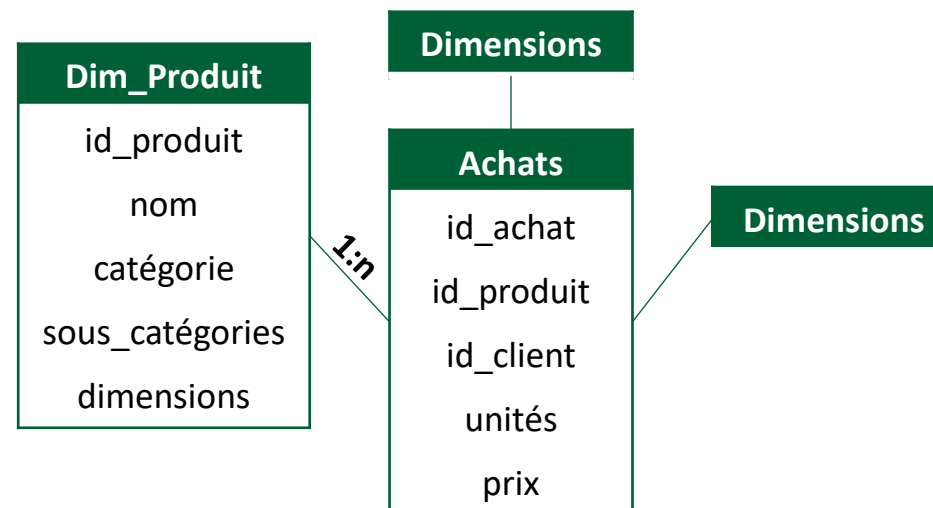
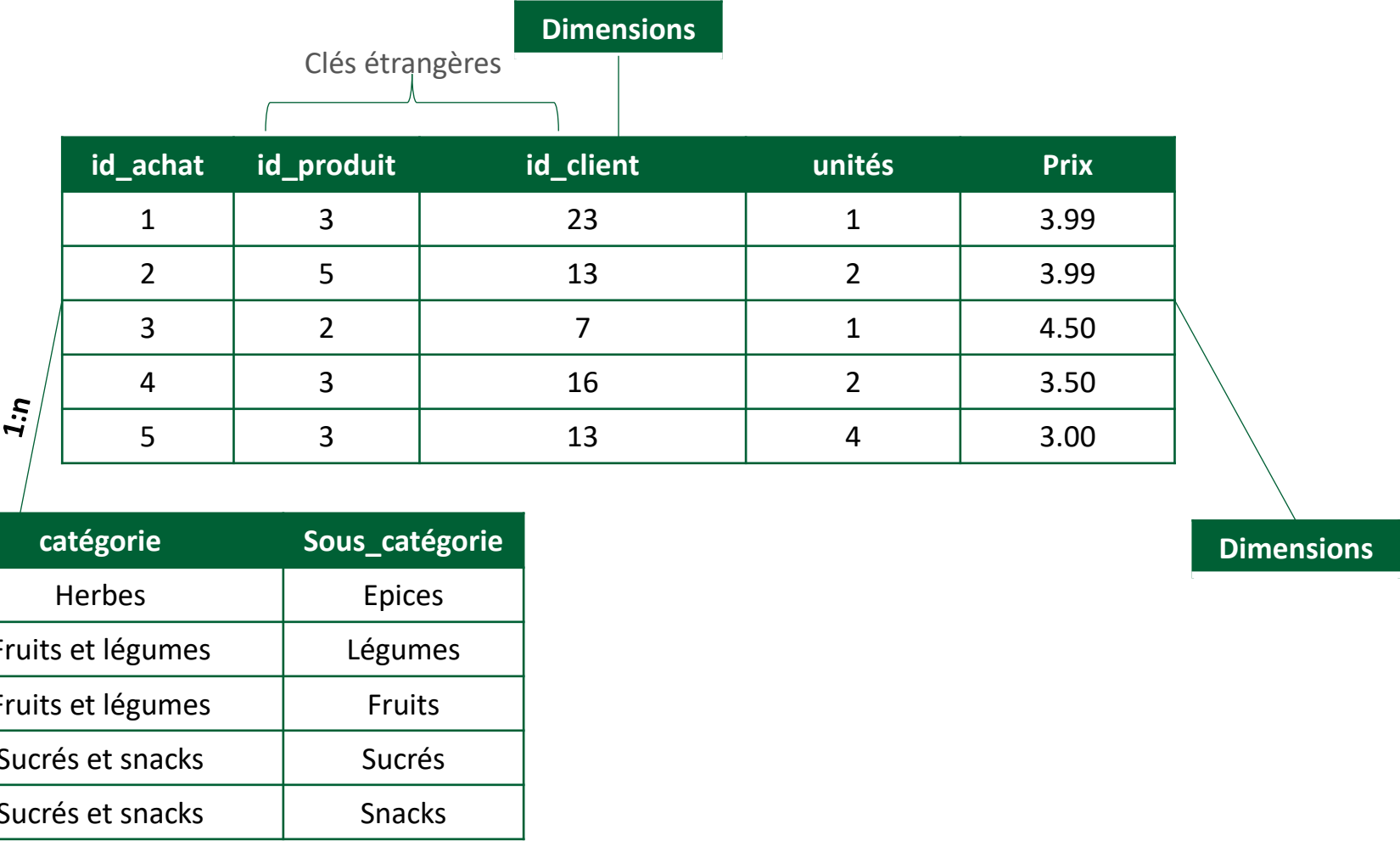


Schéma en étoile



### Schéma en étoile

- Comme on n'a, dans un schéma en étoile, qu'un seul niveau hiérarchique, on a donc une seule connexion entre une table de dimension et une autre de faits (on ne trouve pas une autre connexion partant de la même table de dimensions) . Cela peut engendrer une redondance de données, parce que la colonne catégorie par exemple doit appartenir un autre niveau d'hiérarchie.
- On verra par la suite, le schéma en flocon qui est une alternative qui peut réduire cette redondance. Cette réduction de la redondance des données est appelée « normalisation ». C'est une technique mathématique. Dans le cas d'un schéma en étoile, la base de données est plus au moins **dénormalisée**.
- La normalisation est importante dans certains cas, mais pas vraiment idéale pour récupérer les données et avoir une bonne lecture de nos opérations, surtout lorsqu'on dispose de plusieurs tables et requêtes. Dans ce cas, on peut accepter cette redondance des données si elle répond à ce qu'on veut faire avec ces données.

id_produit	nom	catégorie	Sous_catégorie
1	Chili	Herbes	Epices
2	Ail	Fruits et légumes	Légumes
3	Banane	Fruits et légumes	Fruits
4	Chocolat	Sucrés et snacks	Sucrés
5	Chips	Sucrés et snacks	Snacks

## 2 - INTRODUIRE LE DOMAINE DU BUSINESS INTELLIGENCE

### Introduction au Modèle dimensionnel



#### Schéma en étoile : Remarques

- On peut trouver, dans certains cas, plusieurs tables de faits. Mais la situation la plus commune et idéale est d'avoir une seule table de faits.
- Dans le cas de plusieurs tables de faits, la même table de dimensions peut être pertinente pour plus qu'une seule table de faits (plusieurs connexions).
- Le schéma en étoile est applicable pour des besoins très spécifiques (un ensemble de requêtes bien précises).

### Schéma en flocon

- Théoriquement, le schéma en étoile est un cas spécifique d'un schéma en flocon. Cette dernière est le concept le plus général, parce qu'il permet plusieurs niveaux d'hierarchies. Un schéma en étoile st un schéma en flocon avec un seul niveau d'hierarchie.
- Prenons le même exemple afin de comprendre à quoi ressemble un schéma en flocon. On a remarqué une redondance de données au niveau de la catégorie du produit qu'on a accepté dans le schéma en étoile pour la raison de la visibilité et la bonne lecture.

id_produit	nom	catégorie	Sous_catégorie
1	Chili	Herbes	Epices
2	Ail	Fruits et légumes	Légumes
3	Banane	Fruits et légumes	Fruits
4	Chocolat	Sucrés et snacks	Sucrés
5	Chips	Sucrés et snacks	Snacks

id_achat	id_produit	id_client	unités	Prix
1	3	23	1	3.99
2	5	13	2	3.99
...	...	...	...	...

### Schéma en flocon

- Mais si on veut utiliser un schéma en flocon, cette redondance peut être réduite, en ajoutant une table de catégorie dans un niveau inférieur (2 ème niveau de hiérarchie) et en gardant seulement l'identifiant de la catégorie (clé étrangère). Ceci permet de prendre moins d'espaces dans le disque.
- Contrairement au schéma en étoile, le schéma en flocon est plus normalisé.

id_achat	id_produit	id_client	unités	Prix
1	3	23	1	3.99
2	5	13	2	3.99
...	...	...	...	...

Faits

id_produit	nom	catégorie	Sous_catégorie
1	Chili	1	Epices
2	Ail	2	Légumes
3	Banane	2	Fruits
4	Chocolat	3	Sucrés
5	Chips	3	Snacks

Dimensions

Dimensions

id_produit	catégorie
1	Herbes
2	Fruits et légumes
3	Sucrés et snacks



### Schéma en étoile vs Schéma en flocon

- Le tableau suivant résume les avantages et les inconvénients d'un schéma en flocon par rapport au schéma en étoile.

Avantages	Inconvénients
Moins d'espace de stockage	Beaucoup plus compliqué par rapport au schéma en étoile (plus de tables qui peuvent être plus compliquées à comprendre)
Moins de redondance → facile à maintenir et à modifier et donc moins de données endommagées	Besoins de beaucoup de jointures afin de récupérer une information (des requêtes SQL plus complexes)
Résoudre quelques ralentissements (lors des mises à jour des données)	Data Marts / Data Warehouse moins performants à cause des jointures

- Un schéma en flocon n'est pas utilisé dans les data marts, puisqu'on a besoin des requêtes rapides. On utilise généralement un schéma en étoile dans la mesure du possible.
- Le schéma en flocon est utilisé lorsqu'on rencontre des difficultés dans la maintenance ou la mise à jour des données, ou lorsque le coût du stockage est un vrai challenge, chose qui est très rare.