



**Université Internationale
de Casablanca**

LAUREATE INTERNATIONAL UNIVERSITIES

Introduction aux bases de données

Dr. Mohammed BOUTABIA
Université Internationale de
Casablanca

Des données y'en a partout

- Au niveau personnel
 - Des relevés de banques, de cartes de crédit
 - Des carnets d'adresses
 - La consommation du téléphone
 - Des papiers utiles
 - Des horaires et disponibilités de transport
 - => Important de les stocker mais pas obligatoire
- Au niveau de l'entreprise
 - Fiche de paye des salariés
 - Liste des fournisseurs
 - Liste des clients
 - Produits en stock
 - => Données très importantes et même stratégiques
- **Il est impératif de stocker ces informations en lieu sûr et pouvoir les consulter à n'importe quel moment**

Systeme d'information (SI)

- Un **systeme d'information** (SI) est un ensemble organisé de ressources (matériels, logiciels, personnel, données et procédures) qui permet de regrouper, de classier, de traiter et de diffuser de l'information sur un environnement donné => concept plus générale qu'une simple base de données
- Données
 - Sur des « objets » (produit, personne, facture, etc.)
 - Sur des événements (vente, embauche, etc.)

Donnée

- Élément fondamental qui sert de base à un système d'information
- Elle est considérée comme un objet qui entre dans le système d'information où il sera soumis à un ou plusieurs traitements pour répondre aux besoins des utilisateurs du système
- Peut correspondre à
 - Un élément de donnée
 - Une structure de données

Élément de donnée

- La plus petite unité porteuse d'une signification pour les utilisateurs du système
- Exemples :
 - Numéro d'assurance sociale
 - Prix d'un article
 - Description d'un produit
- Souvent représenté par un type de donnée élémentaire
 - Chaîne de caractères
 - Numérique (entier, réel)

Structure de données

- Ensemble d'éléments de données
- Peut inclure :
 - Des éléments de données
 - D'autres structures de données
- Ex: Facture
 - Numéro de facture
 - Prix total
 - Client
 - Numéro
 - Nom

Base de données

- **Définition** : une base de données est un ensemble structuré de données représentant un univers donnée (1) enregistrées sur des supports accessibles par l'ordinateur (2) pour satisfaire simultanément plusieurs utilisateurs (3) de manière sélective (4) en un temps opportun (5).
 - (1) : Organisation et description de données
 - (2) : Stockage sur disque
 - (3) : Partage des données
 - (4) : Confidentialité
 - (5) : Performance
- Exemples
 - Annuaire téléphonique
 - Dictionnaire
 - Horaires de trains
- N'est pas une base de données
 - Roman -> Non structuré

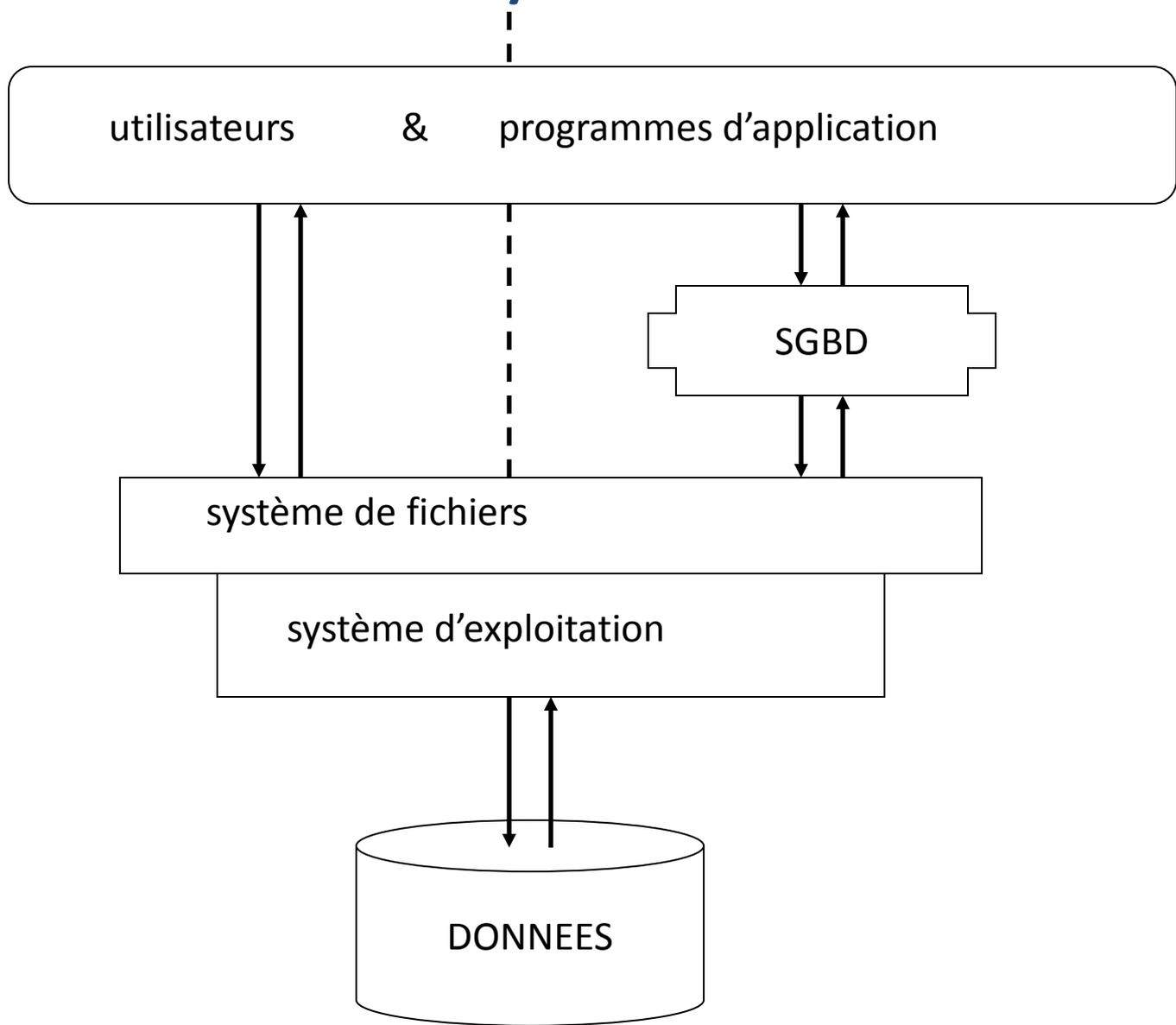
Systeme de gestion de bases de données (SGBD)

- Ensemble de programmes qui gèrent un ensemble de données et qui permettent à plusieurs utilisateurs de stocker des données, de les modifier, de les supprimer et d'extraire des informations à partir de cette base de données
- Protège les utilisateurs de la base de données des détails du niveau matériel

Historique

- 1950-1960
 - Des fichiers séquentiels, du ‘batch’
- 1960 – 1970
 - Le début des bases de données hiérarchiques
- 1970 – 1980
 - La naissance du modèle relationnel
- Début des années 90
 - Sql, l’aide à la décision
- Fin des années 90
 - Croissance du volume des données, Internet, modèle multi tiers

SGF / SGBD

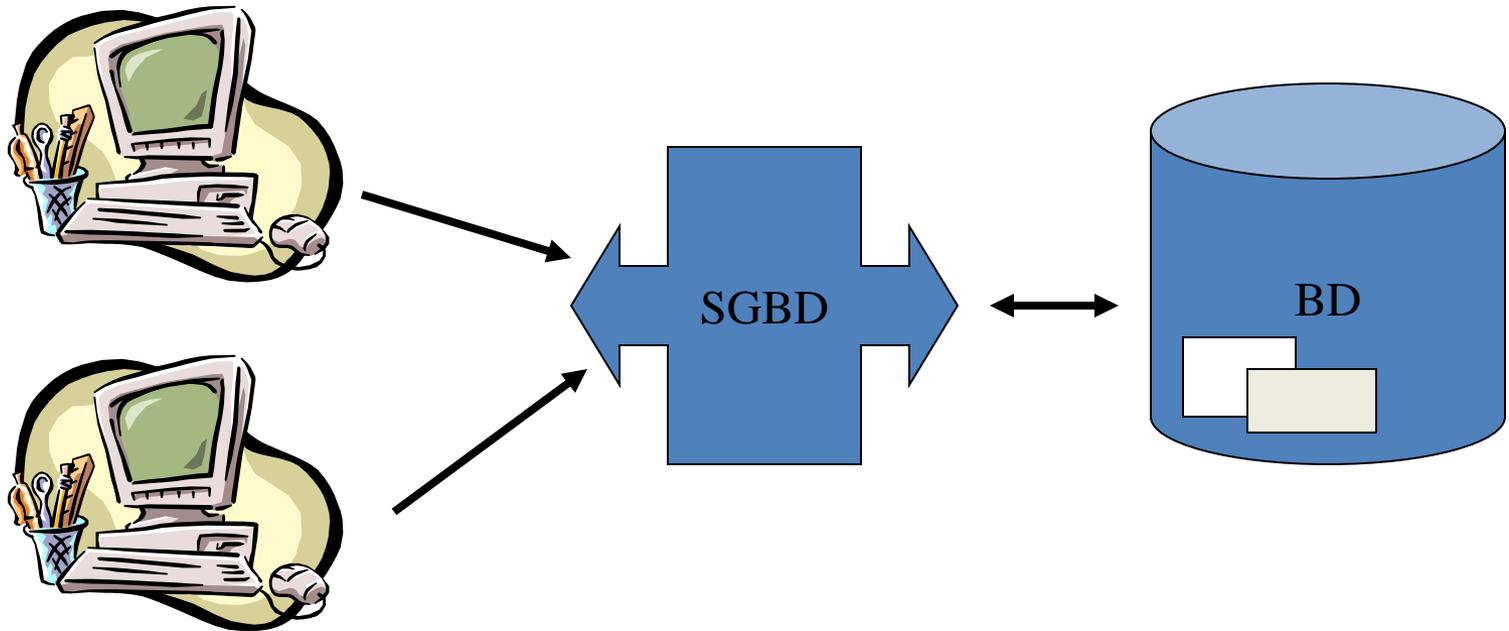


Les limites de l'utilisation des fichiers

- De telles applications sont
 - rigides, (difficile à modifier)
 - longues et coûteuse à mettre en œuvre
 - connaissance technique approfondies
- Les données associées sont :
 - mal définies et mal désignées,
 - redondantes
 - peu accessibles de manière ponctuelle
 - Manque de contrôle d'accès

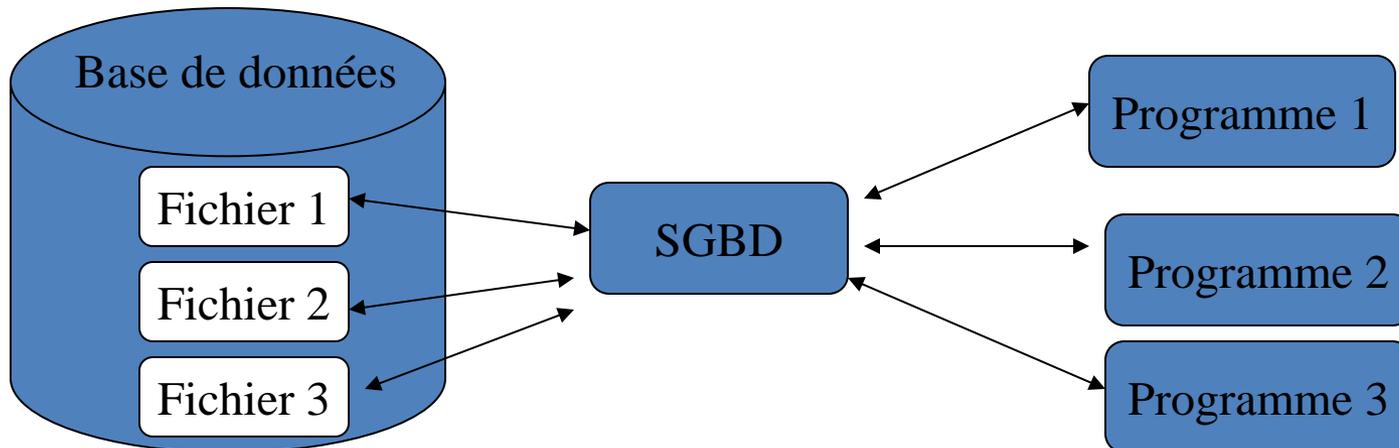
SGBD

Le SGBD permet d'interagir avec une BD est Système de Gestion de Base de Données (SGBD)



SGBD

- Un SGBD est un intermédiaire entre les utilisateurs et les fichiers physiques
- Un SGBD facilite:
 - la gestion de données, avec une représentation intuitive simple sous forme de table par exemple
 - la manipulation de données. On peut insérer, modifier les données et les structures sans modifier les programmes qui manipulent la base de données

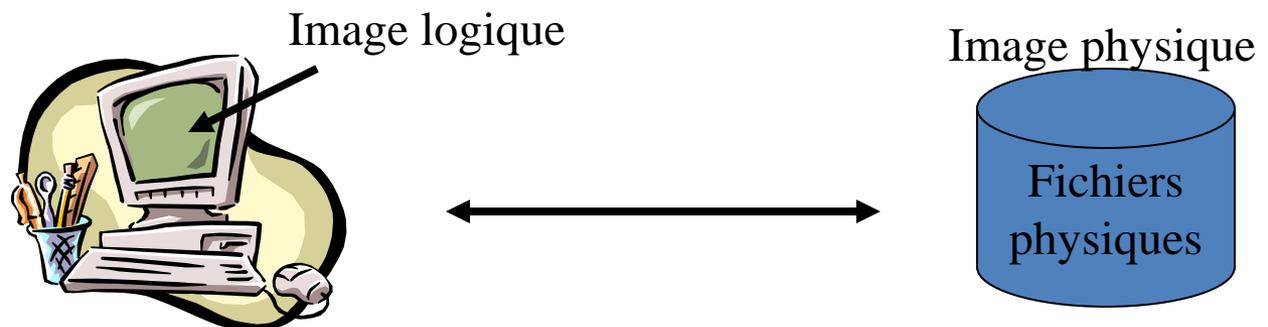


SGBD pour quel objectifs?

- Indépendance physique
- Manipulations des données par des non informaticiens
- Efficacité des accès aux données
- Cohérence des données (intégrité)
- Non redondance des données
- Partageabilité des données
- Sécurité des données
- Résistance aux pannes

Objectifs des SGBD

- Indépendance physique : ne plus se soucier de la manière dont les fichiers sont stockés sur le disque => image logique de la base de données, (ou aussi description logique ou conceptuelle). On peut insérer, supprimer, modifier des données directement sur l'image logique. Le SGBD va s'occuper de faire le travail sur les fichiers physiques.



Objectifs des SGBD

- Manipulations des données par des non informaticiens. Il faut pouvoir accéder aux données sans savoir programmer ce qui signifie des langages « quasi naturels ».
- Efficacité des accès aux données : Ces langages doivent permettre d'obtenir des réponses aux interrogations en un temps raisonnable. Il doivent donc être optimisés (minimiser le nombre d'accès disques).

Objectifs des SGBD

- Cohérence des données. Les données sont soumises à un certain nombre de contrainte d'intégrité qui définissent un état cohérent de la base. Elles doivent pouvoir être exprimées simplement et vérifiées automatiquement à chaque insertion, modification ou suppression de données, par exemple:
 - l'âge d'une personne supérieur à zéro
 - Le solde d'un compte est toujours positive

Dés que l'on essaie de saisir une valeur qui ne respecte pas cette contrainte, le SGBD le refuse.

Objectifs des SGBD

- Non redondance des données : Afin d'éviter les problèmes lors des mises à jour, chaque donnée ne doit être présente qu'une seule fois dans la base.
- Partageabilité des données : Il s'agit de permettre à plusieurs utilisateurs d'accéder aux mêmes données au même moment. Si ce problème est simple à résoudre quand il s'agit uniquement d'interrogations et quand on est dans un contexte mono-utilisateur, ce n'est pas le cas quand il s'agit de modifications dans un contexte multi-utilisateurs:
 - Permettre à deux (ou plus) utilisateurs de modifier la même donnée « en même temps »;
 - Assurer un résultat d'interrogation cohérent pour un utilisateur consultant une table pendant qu'un autre la modifie.
 - Gestion des accès concurrents

Objectifs des SGBD

- Sécurité des données. Les données doivent pouvoir être protégées contre les accès non autorisés. Pour cela, il faut pouvoir associer à chaque utilisateur des droits d'accès aux données.
- Résistance aux pannes Les pannes se produisent de temps en temps. Il faut pouvoir, lorsque l'une d'elles arrive, récupérer une base dans un état sain => fichiers de journalisation

Accès concurrents

- Les données peuvent être manipulées **simultanément** par plusieurs utilisateurs.
- L'objectif est de mettre à la disposition du plus grand nombre d'utilisateur un ensemble **intègre** (cohérent) de données
- Ex: transactions bancaires simultanées
- La solution pour les accès concurrents consiste à **verrouiller** les données momentanément (pendant la durée de la transaction). Les autres transactions sont alors en **attente**.

Trois Fonctions d'un SGBD

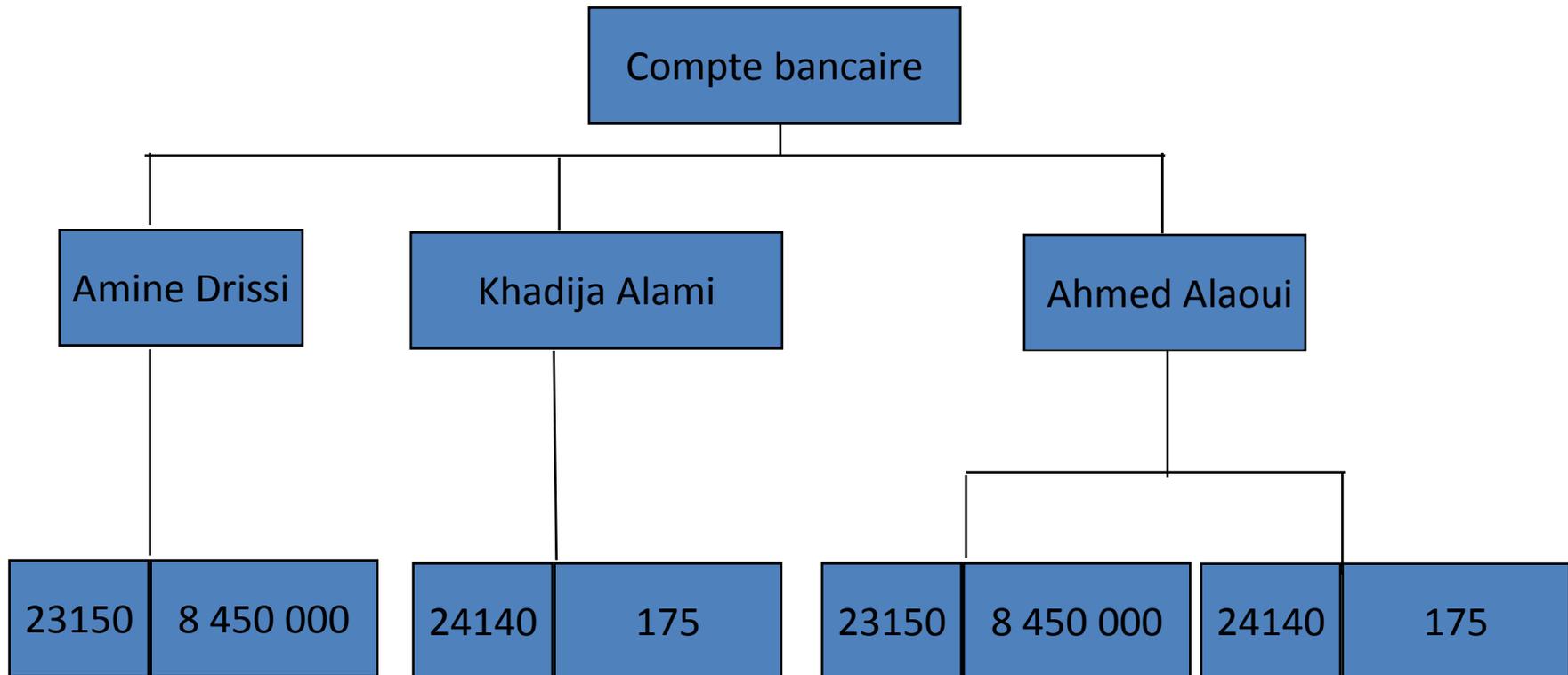
- Description des données : codification structuration, grâce à un Langage de Description de Données (LDD)
- Manipulation et restitution des données (insertion, mise à jour, interrogation)
 - Mise en œuvre à l'aide d'un Langage de Manipulation de Données (LMD)
 - S.Q.L. (Structures Query Langage) : Langage standard
- Contrôle et administration (partage, intégrité, confidentialité, sécurité)

Types de SGBD

- Hiérarchique
- Réseau
- Relationnel
- Objet

SGBD hiérarchique

- Structure arborescente

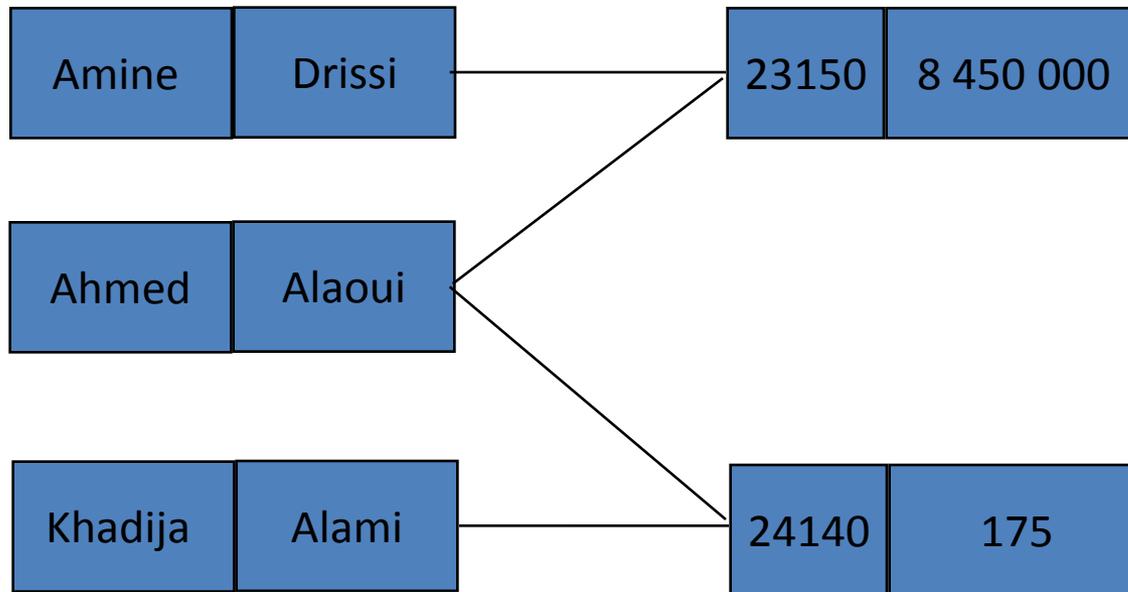


SGBD hiérarchique

- Caractéristiques
 - Accès vertical (haut vers bas et bas vers haut)
 - Les chemins hiérarchiques sont implémentés par des pointeurs
- Problèmes
 - **Redondance des données** : une modification du solde d'un seul compte nécessite plusieurs mises à jour
 - **Coûteux en espace mémoire**
 - **Accès aux informations parfois difficiles et/ou coûteux** : clients possédant un compte donné

SGBD réseau

- Composé d'enregistrements et de liens



SGBD réseau

- Implémentation avec des pointeurs
- Avantages
 - Redondance éliminée, moins d'espace mémoire
 - Manipulations plus faciles
- Inconvénients
 - Accès séquentiel, donc lent
 - Difficile à programmer et à maintenir
- Les SGBD hiérarchiques et réseaux sont aujourd'hui considérés comme obsolètes

SGBD relationnel (SGBDR)

- Constituée de tables représentant des objets ou des associations entre objets

TABLE CLIENT

NOCLIENT	NOM	PRENOM
KOI01	Drissi	Amine
DEL01	Alami	Khadija
KOI02	Alaoui	Ahmed

TABLE COMPTE

NOCOMPTE	SOLDE
23150	8 450 000
24140	175

TABLE COMPTE-CLIENT

NOCOMPTE	NOCLIENT
23150	KOI01
23150	KOI02
24140	DEL01
24140	KOI02

SGBD relationnel (SGBDR)

- Pas de pointeurs
- Une table est composée de
 - Colonnes
 - Portent un nom unique à l'intérieur d'une table
 - Peuvent faire partie de plusieurs tables (pour les lier)
 - Rangées ou lignes
 - Identifiées par une clé unique (**clé primaire**) composée d'une ou plusieurs colonnes

Clé primaire

- La valeur de la clé primaire doit être
 - Unique dans chaque table
 - De longueur fixe
 - De longueur raisonnable
 - Non modifiable
 - Comment choisir une clé primaire
 - Numéro d'assurance sociale ?
 - Nom + Prénom ?
 - Numéro de téléphone ?
- => Identifiant dans le monde réel ou attribué par l'administrateur de la BD

SGBD relationnel (SGBDR)

- Lien logique de chaque valeur d'une rangée à la clé primaire => relation de dépendance
- Exemple

TABLE CLIENT

NOCLIENT	NOM	PRENOM	VILLE	CODE POSTAL
DEL01	Alami	Khadija	Marrakech	40000

- Le Prénom Khadija dépend de la clé DEL01 => correct
- Le code postal dépend de la ville, donc d'un autre attribut que la clé primaire => faux
- Ville et code postale doit figurer dans une autre table

SGBD relationnel (SGBDR)

- Lien logique entre les tables
 - Une rangée d'une table peut avoir un lien logique nommé **relation** avec une ou plusieurs rangées d'une ou de plusieurs autres tables
 - Relation représentée par une **clé étrangère**
- Clé étrangère
 - Constituée d'une ou de plusieurs colonnes
 - Représente la relation entre deux rangées
 - Correspond toujours à une clé primaire
 - Ex : NOCLIENT est la clé primaire de la TABLE CLIENT et une clé étrangère de la table COMPTE-CLIENT

SGBD relationnel (SGBDR)

- Avantages
 - Redondance minimisée : seules les clés primaires sont dupliquées
 - Facilité de programmation et d'exploitation
 - Structure basée sur la logique mathématique
 - Possède son propre langage normalisé : SQL
 - Portabilité
- Quelques SGBD (relationnels du marché)
 - Micro : ACCESS, SQL Server, Dbase, PostgreSQL, MySQL, ...
 - Gros système : DB2, ORACLE, SYBASE, ...

SQL (Structured Query Language)

- Ensemble de commandes permettant de créer et d'exploiter une base de données relationnelle
- Assume toutes les principales fonctions d'un SGBD => création de table, insertion, mise à jour, suppression, requêtes
- Peut être intégré dans des applications écrites en différents langages (C, C++, ASP, PHP, etc.)

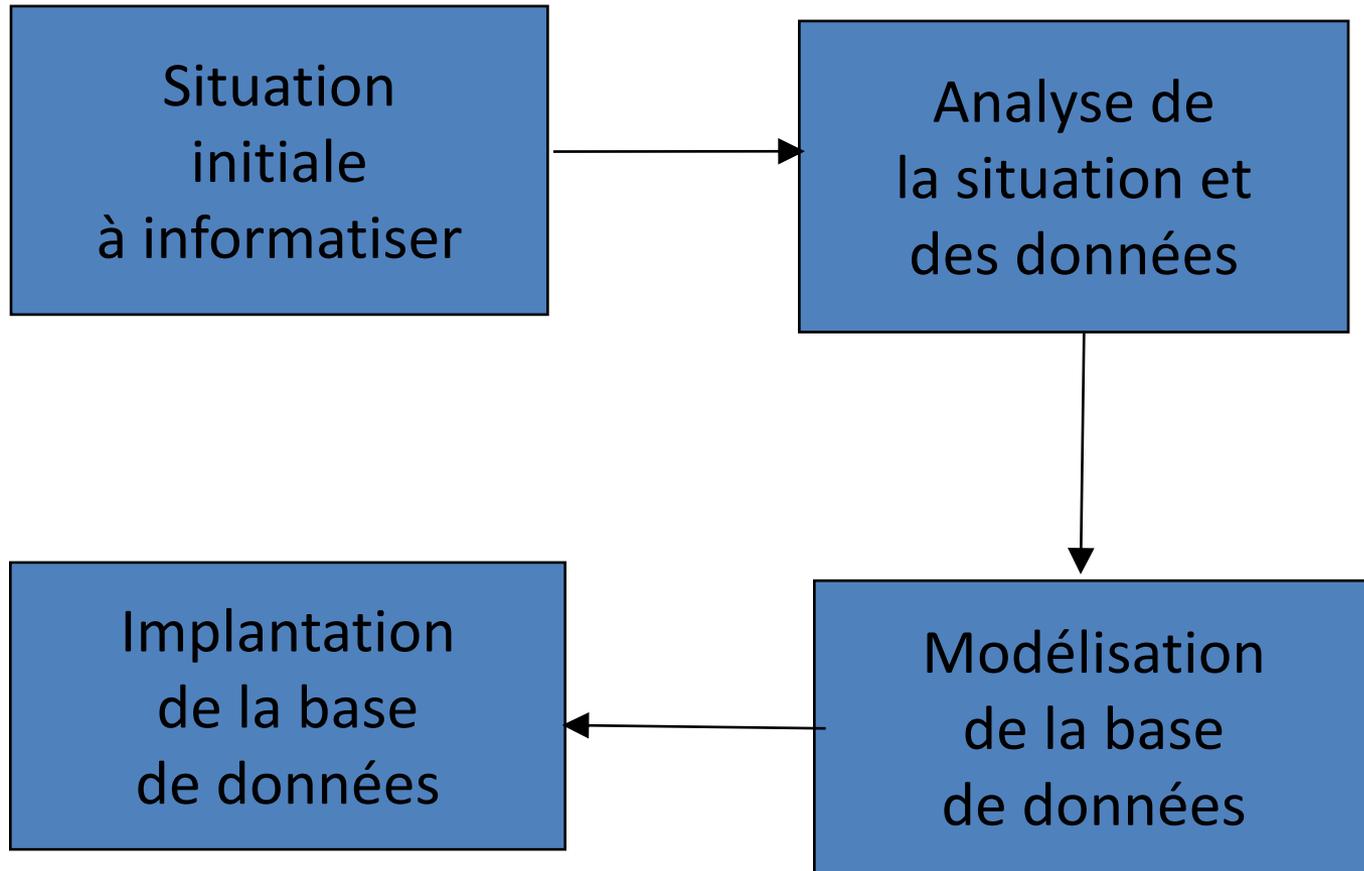
Les intervenants dans un SGBD

- Utilisateurs
 - Utilisent les applications conçues par les concepteurs
 - Entrent, modifient, suppriment et consultent les données
- Concepteurs
 - De bases de données : conçoivent les modèles de données et créent la structure de la base de données
 - D'applications : développent des programmes qui gèrent les bases de données
- Administrateurs (DBA)
 - Gèrent le système, les utilisateurs (opérateurs et concepteurs), les objets et les privilèges

Etapes de construction d'une BD

- La mise en place d'une Base de données passe par plusieurs étapes :
 - Analyse du domaine d'activité: définition des besoins en informations et recensement des données utiles pour le domaine d'activité.
 - Modélisation (Représentation) des Données du domaine.
 - Construction du schéma de la Base de Données.

Étapes de construction d'une BD



Base de données = image organisationnel de l'entreprise

Analyse du domaine

- Il s'agit d'une étape fondamentale qu'on trouve dans tout projet d'informatisation et en particulier celui de la construction d'une Base de Données.
- Cette étape consiste à :
 - Comprendre le domaine d'activité.
 - Préciser les périmètres fonctionnels et organisationnels.
 - Fixer les acteurs utilisateurs de la BD.
 - Recenser les données, les traitements et les règles qui les régissent.

Analyse du domaine

- Exemple d'une bibliothèque:
 - Des objets: exemple OUVRAGES, ETUDIANTS
 - Des propriétés des objets (attributs) : exemple Titre de OUVRAGES
 - Des liens entre les objets : un OUVRAGE peut être emprunté par un ETUDIANT
 - Des contraintes : le nombre d'exemplaires d'un OUVRAGE est supérieur à zéro
- Cette étape aboutit à l'établissement de deux catalogues (listes):
 - Un dictionnaire des données.
 - Un catalogue des règles de gestion.

Modélisation des données

- Cette étape consiste à :
 - L'obtention du schéma de la base de données.
 - Selon un formalisme spécifique (UML, Merise)
- le schéma conceptuel est la base pour la structure de la base de donnée

Implémentation de la BD

- Il s'agit d'implémenter sur ordinateur, le schéma de la base de données, obtenu dans l'étape précédente.
- La construction de la base de données se fait à l'aide d'un SGBD.
- Une fois construite la base pourra être exploitée par les utilisateurs.

Opérations sur les BD

- Deux catégories d'opérations sur les BD
 - Les opérations de définition de la structure de la BD (squelette de la BD) => réalisées une fois pour toute à partir du schéma conceptuel
 - Opérations de manipulation de données => exploitation de la base de données

Définition de la structure

- Les informations relatives à un domaine donné concernent plusieurs sujets (produit, client, commande...)
- Chaque sujet est caractérisé par un certain nombre d'attributs
 - Exemple: produit
 - Code produit
 - Désignation
 - prix

Définition de la structure

- Chaque sujet sera représenté dans la base par une table
- La table regroupe les informations sur le sujet sous forme de champs
- Définir la table consiste à
 - Attribuer un nom à la table
 - Définir les champs (colonne)
- Définir le champs d'une tabel consiste à
 - Attribuer un nom au champs
 - Définir les propriétés du chapms

Définition de la structure

- Parmi les propriétés d'un champs
 - Le type: domaine des valeurs du champs
 - Taille: nombre de caractère du champs
 - Format: monétaire, pourcentage
 - Valeur par défaut

Modification de la structure

- La maintenance de la BD peut mener à des modification de la structure de la BD
- On peut modifier la structure de la BD par
 - L'ajout, suppression ou modification de la structure de la table
- La structure de la table est modifiée par
 - L'ajout, suppression ou la modification des propriétés d'un champs

Manipulation du contenu

- La manipulation du contenu consiste à
 - Ajouter un enregistrement (une ligne) qui correspond à une instance du sujet, exemple: ajout du produit café au catalogue des produit, ajout d'un nouveau client
 - Modifier les champs d'un enregistrement, exemple: modifier le prix d'un produit
 - Supprimer un enregistrement
 - Rechercher des informations, quels sont les produits commandés par les clients de casablanca?

Les contraintes d'intégrité

définition

- La contrainte d'intégrité est une condition qui porte sur le contenu d'une table de la base de données
- La contrainte d'intégrité est implémenté à partir des règles de gestion du domaine d'activité
- Le SGBD veille au respect en permanence de ces contraintes d'intégrité

exemples

- Un article doit être identifié de manière unique (une référence par article)
- L'âge minimal des employés est 18 ans
- Le prix de vente des articles est supérieur à un centime
- Une commande est passé par un seul client

Types de contraintes d'intégrité

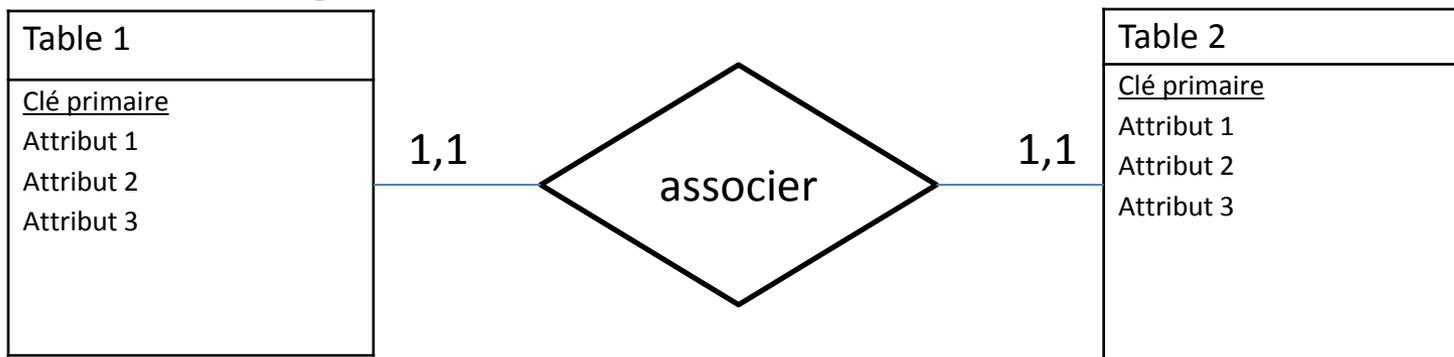
- Unicité de la clé primaire
 - Une clé primaire est un ensemble minimal de champs dont la valeur est unique dans toute la table
 - Deux lignes différentes de la même table ont des clés primaires différentes
 - Code article, CIN, CNE...
- Contrainte de domaine
 - C'est une condition qui porte sur les valeurs d'un champ
 - Restreindre les valeurs d'un champ à un sous ensemble donné
 - Exemple: le nombre des étudiants dans une classe est compris entre 20 et 30 étudiants

Types de contraintes d'intégrité

- Relation d'intégrité référentielle:
 - C'est une condition qui porte sur la relation entre les lignes d'une table avec les lignes d'une autre table
 - À cet effet la clé primaire d'une table va migrer vers l'autre table pour mettre en œuvre cette relation

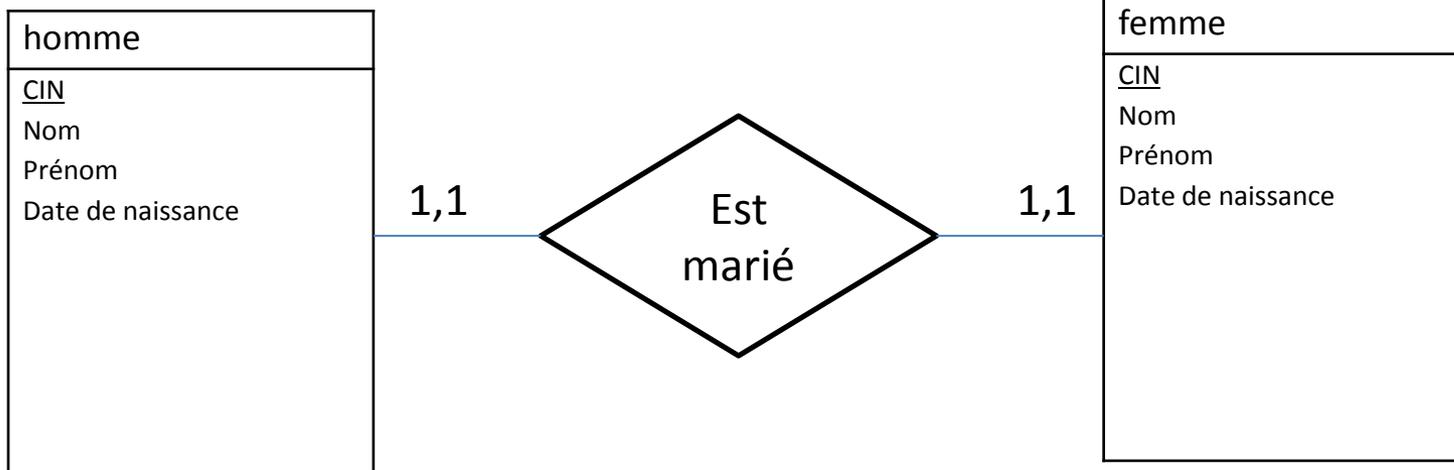
Relation un à un

- La règle de gestion est
 - Un élément de la table 1 est en relation avec un et un seul élément de la table 2
 - Un élément de la table 2 est en relation avec un et un seul élément de la table 1
- Implémentation dans la BD => clé primaire d'une des deux tables migre vers l'autre table, la table qui voit sa clé migrer est appelée table référençante et la table qui reçoit la clé étrangère est appelée table référencée
- Condition sur la clé étrangère:
 - même type de données
 - même longueur
 - les appellations peuvent être différentes
 - La Clé étrangère doit être indexé sans doublons



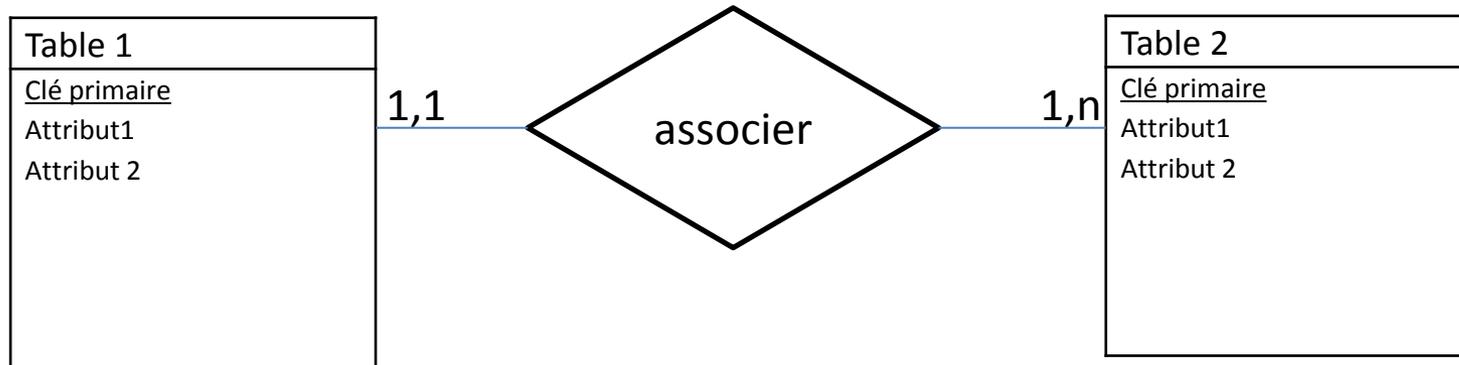
exemple

- La règle de gestion est :
 - Un homme est marié à une et une femme
 - Une femme est mariée à un et un homme
- Implémentation dans la BD => CIN de homme migre vers la table femme
- Exemple:



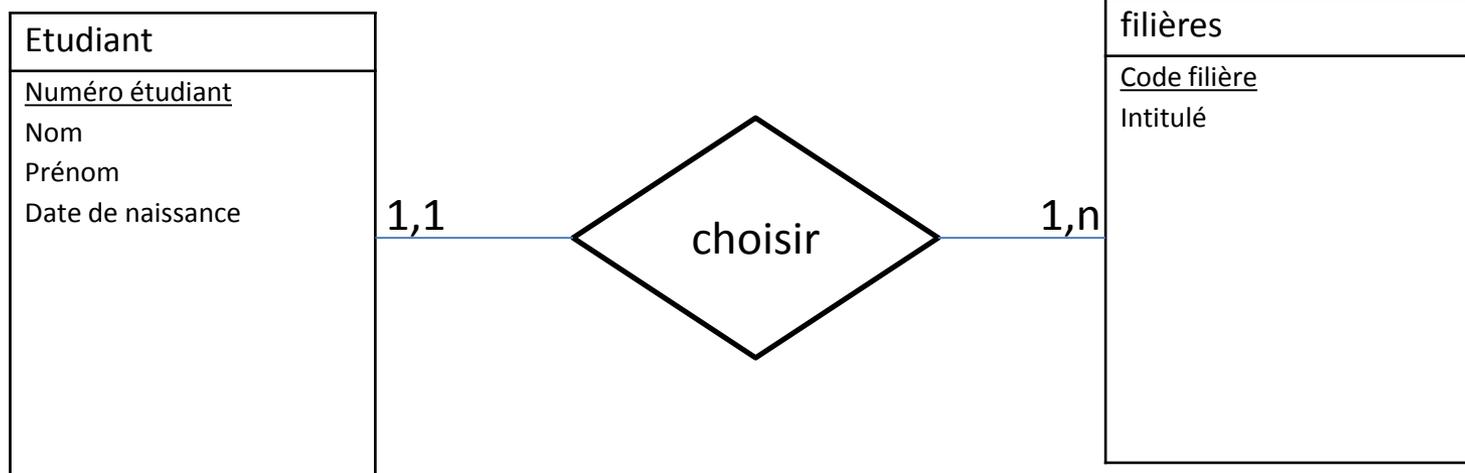
Relation un à plusieurs

- La règle de gestion:
 - Un élément de la table 1 est en relation avec un et un élément de la table 2
 - Un élément de la table 2 est en relation avec un ou plusieurs éléments de la table 1
- Implémentation dans la BD => la clé primaire de la table 2 migre vers la table 1
- Condition sur la clé étrangère
 - même type de données
 - même longueur
 - les appellations peuvent être différentes)
 - La clé étrangère doit être indexée avec doublons



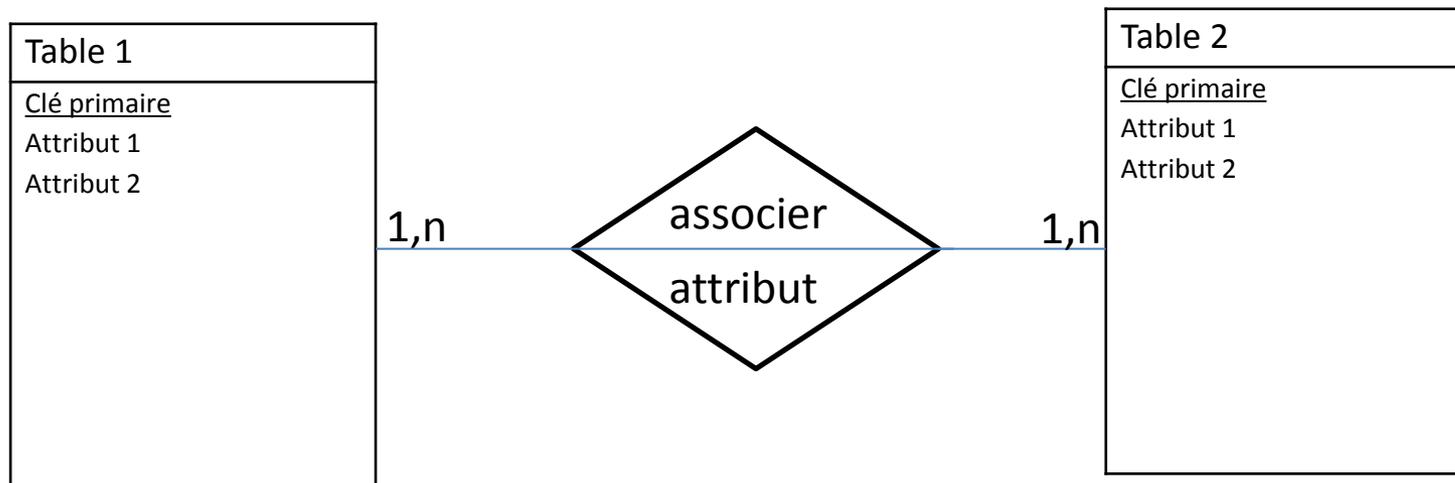
exemple

- La règle de gestion:
 - Un étudiant choisi une et une filière
 - Une filière est choisi par un ou plusieurs étudiant
- Implémentation dans la BD => la clé primaire de la table filière migre vers la table étudiant
- Exemple:



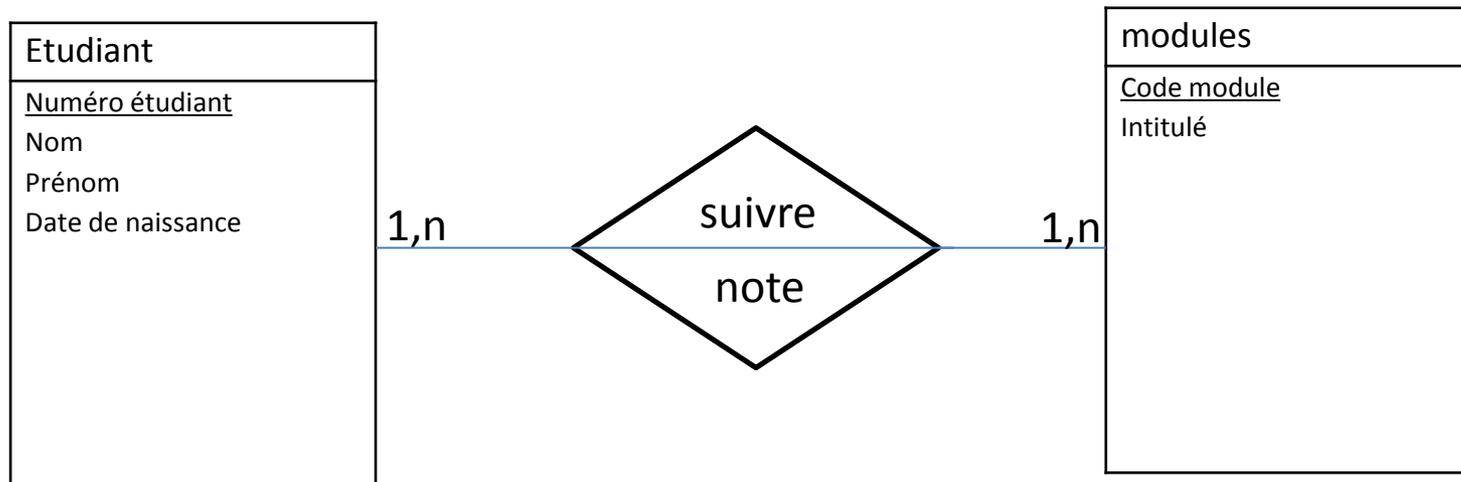
Relations plusieurs à plusieurs

- La règle de gestion:
 - Un élément de la table 1 est lié à un ou plusieurs éléments de la table 2
 - Un élément de la table 2 est lié à un ou plusieurs éléments de la table 1
- Implémentation dans la BD => création d'une nouvelle table qui a comme identifiant les deux clés primaires des deux tables, les attribut de l'association deviennent les champs de la table créée



exemple

- La règle de gestion:
 - Un étudiant suit plusieurs modules
 - Un modules est suivi par plusieurs étudiants
- Implémentation dans la BD => création d'une nouvelle table qui a comme clé primaire les deux clés primaires des tables étudiant et modules, la note est un champ de la table



Les requêtes

- Une requêtes est un ordre envoyé à la base de données dans le but d'extraire, mettre à jour ou supprimer certaines données selon quelques critères
- Une requête peut être basée sur une ou plusieurs tables
- Les bases de données relationnelle utilise le langage SQL pour exécuter les requêtes
- Il y a aussi un mode graphique pour les débutants

Types de requêtes

- **Requête sélection** : Affiche les enregistrements qui répondent aux critères demandés. C'est le type que vous allez utiliser le plus souvent. Il affiche seulement les enregistrements qui répondent aux critères sélectionnés.
- **Requête analyse croisée** : Affiche un tableau de synthèse croisé selon les champs demandés.
- **Requête nouvelle table**: Création d'une nouvelle table avec les champs demandés selon les critères demandés.
- **Requête mise à jour**: Mise à jour des enregistrements selon les critères demandés.
- **Requête ajout**: Ajoute des enregistrements d'une autre table selon les critères demandés.
- **Requête suppression**: Élimine des enregistrements selon les critères demandés. Il est possible d'effacer en même temps un groupe d'enregistrements qui répondent aux critères demandés.

Les étapes pour la création d'une requête

1. Choisir la ou les tables et les requêtes nécessaires
2. Choisir le type de requête
3. Choisir le ou les champs nécessaires
4. Déterminer si les champs ont besoin d'être triés
5. Cacher les champs au besoin
6. Déterminer les critères de sélection
7. Exécuter la requête