

Cours Fodements des Bases de Données

Dr.Syrine Belguith

INSTITUT SUPERIEUR D'INFORMATIQUE ET DE MATHEMATIQUE DE MONASTIR

March 7, 2025

Fondements des bases de données

Objectifs:

Au bout de ce module, l'étudiant doit être capable :

1. Appréhender les notions de BD dans un système d'information.
2. Posséder une connaissance approfondie des divers modèles de bases de données (de l'hierarchique au NoSQL en passant par le relationnel-objet).
3. Savoir modéliser une BD avec un modèle conceptuel (EA, UML)
4. Maitriser le passage du modèle conceptuel au modèle logique
5. Apprendre à comprendre le modèle relationnel et son algèbre qui constitue la base des relations.
6. Se servir d'un outil de modélisation (PowerDesigner, Toad, Erwin, DB-Main, etc.)
7. Maitriser l'essentiel du Standard SQL

Plan

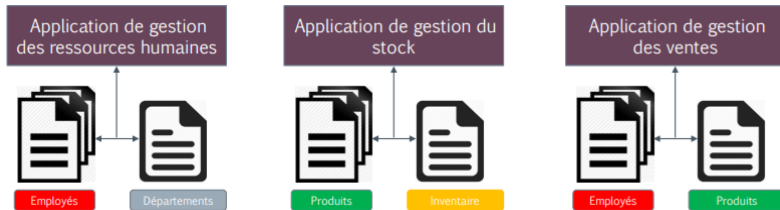
1. Introduction aux bases de données
2. Modélisation conceptuelle des BD (EA ET UML)
3. Modélisation logique (Modèle relationnel et son Algèbre)
4. Normalisation et fromes normales
5. Le Langage SQL

Introduction aux bases de données

1. Notion de bases de données
2. Architecture et rôle des Systèmes de Gestion de Bases de Données (SGBD)
3. Modèle de données et typologie des SGBD
4. Cycle de développement d'une BD

Notion de bases de données

Approche classique: Les données sont décrites dans les programmes dans des fichiers



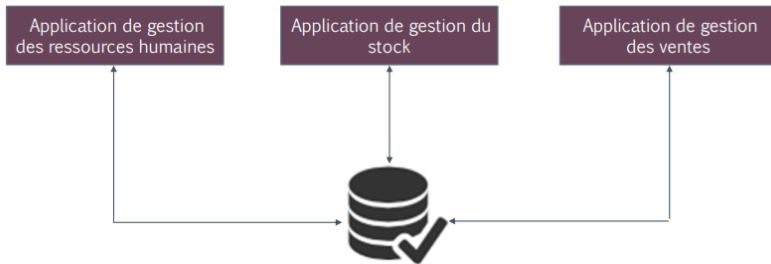
Notion de bases de données

Limites approche classique:

- La difficulté de créer et de maintenir les données dans les fichiers – > Le coût élevé...
 - différentes technologies de programmation
 - différents formats de fichiers et de données
- Redondance de certaines informations
- Inconsistance potentielle entre les données
- Accroissement inutile : du nombre des fichiers, de leurs tailles, du temps d'accès à l'information.
- Complexité de protection et d'accès concurrents aux données

Notion de bases de données

Stocker les données dans une base de données centralisée



Notion de bases de données

Base de données :

- collection de données
- enregistrées (sur un support adressable),
- utilisées par des programmes
- structurées indépendante d'une application,
- cohérentes,
- accessibles par plusieurs utilisateurs

Notion de bases de données

Objectifs des bases de données:

- Organiser les données indépendamment des programmes : Organiser les informations en fonction de leur nature et des liens réels qui existent entre elles
- Partager l'information : partage d'un ensemble unique d'informations par plusieurs utilisateurs / applications
 - Besoin de gérer l'accès concurrent et l'intégrité de la BD
- Réduire les redondances :
 - Un seul fichier du personnel pour l'application paye/comptabilité et l'application ressources humaines

Système de Gestion de Bases de Données (SGBD)

SGBD

ensemble d'outils logiciels permettant la gestion et la manipulation de bases de données.

Le SGBD cache la complexité des opérations et offre une vue synthétique sur le contenu.

Les objectifs d'un SGBD sont :

- Définition des données
- Manipulation des données
- Sécurité et intégrité des données
- Gestion des transactions et des accès concurrents

Fonctions des SGBD

- **Définition des données** : Élaborer une description pour le stockage des données
 - > **Modélisation logique**
 - > **Langage de définition des données (LDD)**

Basé sur **une description des données** : Décrire les données de l'application sans faire référence à une solution informatique particulière

Fonctions des SGBD

- **Manipulation des données** : Créer au fur et à mesure les données.
Modifier si besoin et éventuellement supprimer toute donnée déjà rentrée.
 - Mise à jour des données (insertion, suppression, modification)
 - > **Langage de manipulation des données** (LMD)
- **Interrogation des données** : Répondre à toute demande d'information portant sur les données contenues dans la base.
 - **Recherche des données** (interrogation de la BD)

Fonctions des SGBD

- **Contrôle des données** : Il faut pouvoir exprimer toutes les règles qui contraignent les valeurs pouvant être enregistrées de façon à éviter toute erreur qui peut être détectée.
 - Contraintes d'intégrité
 - Droits d'accès
 - Gestion des transactions et de la sécurité
 - > Langage de contrôle des données (LCD)
- **Garantie de confidentialité** : Toute information doit pouvoir être protégée contre l'accès par des utilisateurs non autorisés

Fonctions des SGBD

Un SGBD est un ensemble d'outils logiciels permettant de:

- Décrire , Manipuler, Consulter les données, définir des Contraintes d'Intégrité

— — — > en assurant

- la confidentialité, le partage des données, l'intégrité, la sécurité,

SGBD : Typologie

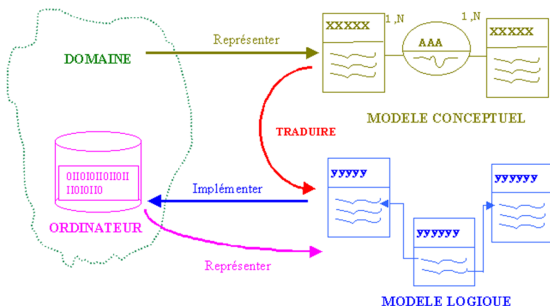
Modèle de représentation de données, Classe de SGBD:

- Hiérarchique, réseau
- Relationnel
- Orienté Objets
- NoSQL

Les méthodologies de conception de BD

Les étapes de conception d'une BD :

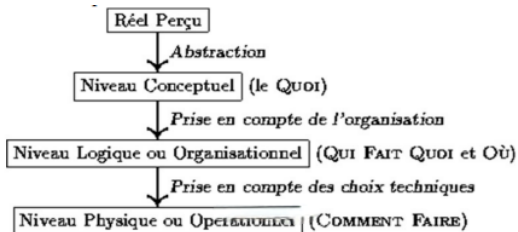
1. Analyse de la situation existante et des besoins
2. Création d'une série de modèles qui permettent de représenter tous les aspects importants
3. A partir des modèles, implémentation d'une base de données



Les méthodologies de conception de BD: Merise

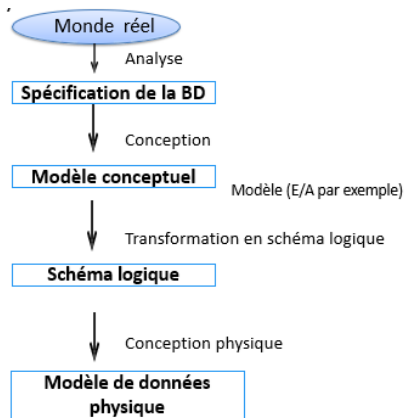
Une méthode d'analyse informatique : Merise

1. Approche globale menée parallèlement sur les données et les traitements.
2. Description en trois niveaux :



Les méthodologies de conception de BD

Les étapes de conception d'une BD :



Etape 1: Analyse de monde réel

- > Avant de commencer la conception de la BD, interviewer les utilisateurs de la BD pour prendre connaissance de leurs besoins :
- caractéristiques des informations,
- relations entre les informations,
- caractéristiques du système informatique qui accueillera la BD

Etape 2: Spécification de la BD

Après avoir étudié le monde réel, il est nécessaire de rédiger **une spécification détaillée** qui décrit les diverses **entités** du monde réel, leurs **caractéristiques** et les **liens** qui les relient.

Etape 2: Spécification de la BD

Exemple 1: Réservation des chambres d'Hôtel

Etape 2: Spécification de la BD

Exemple 1: Réservation des chambres d'Hôtel

Hôtel est caractérisé par son nom, adresse, ville, et téléphone.

Chambre : Contient les détails des chambres (numéro, type, prix par nuit, état).

Un hôtel possède plusieurs chambres.

Client : nom, prénom, email, téléphone).

Réservation : Gère les réservations (client, chambre, dates, prix total).

Un client peut effectuer plusieurs réservations.

Etape 2: Spécification de la BD

Exemples

- **Agence de location de voiture:**

Proposer une spécification textuelle décrivant les principaux objets d'une agence de location de voitures et les interactions entre ces objets

Etape 2: Spécification de la BD

Une voiture est identifiée par son numéro d'immatriculation, sa couleur, la date de sa mise en service et le prix de location unitaire.

- **La marque**, la carrosserie, les cylindrés, le carburant, la puissance, le type de boîte et le nombre de rapports sont des éléments qui distinguent un modèle.
- **Une voiture** fait partie d'un modèle.
- **Un client** est identifié par un numéro de CIN, son nom et prénom, son adresse, sa date de naissance et le numéro de son permis de conduire.

Le client a la possibilité de louer une ou plusieurs voitures.

- Le numéro de transaction, la date de location et la date de retour sont attribués à chaque location effectuée.

Etape 3: Modèle conceptuel

Passage du monde réel au monde conceptuel via un modèle (exemple : le modèle entité/association).

- Il permet d'établir une représentation claire des données du Système d'Information.
- Définit les dépendances fonctionnelles de ces données entre elles.
- Doit être indépendant de toute implantation (de toute machine)
- Géré par le concepteur de la BD
- Un modèle conceptuel de données est indépendant de l'état de l'art technologique.

Etape 4: Schéma logique

Passage du modèle conceptuel à un modèle de bases de données (relationnel, objet, ...), en vue de l'implantation sur machine.

- Géré par le concepteur de la BD
- ! Il ne faut pas confondre le modèle conceptuel (E/A par ex.) avec le modèle logique (relationnel par ex.).
- Le niveau logique est une REPRESENTATION du système tel qu'il sera implémenté dans des ordinateurs.
- ! Il ne faut pas confondre le modèle logique (relationnel par ex.) avec son implémentation physique en machine (avec Access ou Oracle par ex.)

Etape 5: Modèle de données physique

- Réaliser des enregistrements en accord avec le modèle de données logique.
- Comment sont stockées les données sur les supports
- physiques. niveau d'implémentation : relatif à la mémoire physique
- Géré par le SGBD

Synthèse

Les avantages d'un SGBD par rapport à un système de gestion de fichiers classique:

- Une grande flexibilité
- L'intégrité et la cohérence des données.
- Une gestion simple des grands ensembles de données.

Les étapes de conception d'une BD :

1. Analyse de la situation existante et des besoins
2. Création d'une série de modèles qui permettent de représenter tous les aspects importants
3. A partir des modèles, implémentation d'une base de données

MODÉLISATION CONCEPTUELLE DES BD (EA et UML)

Modélisation des données: Modèle Entité-Association

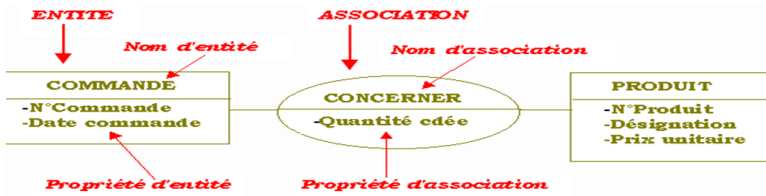
- **Niveau conceptuel de Merise** : Modèle de Conception de Données (**MCD**)
- Le MCD est basé sur la Conception **Entité-Association**, qui est un modèle d'analyse du monde réel.
- Conçu par Peter Chen dans les années 1970 afin de fournir une notation unifiée
- Il fournit une **description graphique** pour représenter des modèles de données sous la forme de diagrammes contenant des **entités** et des **associations**.

Modélisation des données: Modèle Entité-Association

Modèle E/A : Description des données en termes d'objets, liens et propriétés.

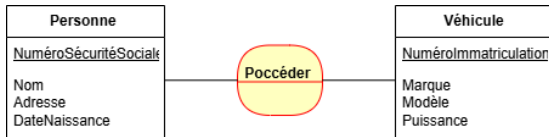
- objet → entité
- liens → association
- propriété → attribut

Exemple: Diagramme ENTITE-ASSOCIATION (**Gestion de commande**)



Le Modèle Entité-Association: Exemple

Exemple d'un Diagramme Entité-Association pour la **Gestion des Cartes Grises** :



Cette représentation graphique des données est appelée :

- **Diagramme** ENTITE/ASSOCIATION (E/A) ou
- **SCHEMA** CONCEPTUEL DE DONNEES (SCD) ou
- **MODELE** CONCEPTUEL DE DONNEES (MCD)

Le modèle Entité-Association, les concepts de base: entité (I)

Entité : objet du monde réel avec une existence indépendante, une entité (ou type d'entité) est une chose (concrète ou abstraite) qui existe et est distinguable des autres entités.

Exemples :

- Le crayon qui se trouve sur ma table (Entité concrète)
- Le Projet de Fin d'Etude PFE (Entité abstraite)

Entité: Exemples d'Entités:

- Une entité de **type personne** est définie par son numéro de sécurité sociale, son nom, son prénom et sa date de naissance
- Une entité de **type commande** est définie par son numéro et sa date de commande

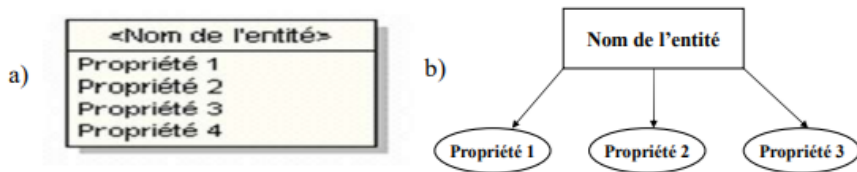
Le modèle Entité-Association, les concepts de base : entité (I)

- **Occurrence d'une entité** est un élément particulier correspondant à l'entité et associé à un élément du réel.
- Dans l'exemple de BD de Gestion des Cartes Grises, on dit **l'entité Personne** et on dit que **Amine Ben ahmed est une occurrence** de l'entité Personne .
- Chaque entité a **des propriétés** (attributs) particulières qui la décrivent,
- Une entité donnée a **des valeurs** pour chacun de ses attributs (ex: nom et prénom du client)
- **Un domaine de valeur** est associé à chaque attribut simple.

Le modèle Entité-Association, les concepts de base : entité (I)

Une entité est défini par un nom et une liste d'attributs.

Représentation graphique des entités :



Contraintes sur les attributs :

Éléментарité : Tous les attributs sont simples ou **atomiques**, c.a.d. ils ne peuvent pas être exprimés par les autres attributs.

Unicité: Identifiant

Nom	Prenom	Date Naissance
ali	ben foullen	05/11/91
ahmed	ben fllen	07/11/1993
ali	ben foullen	05/11/91

Problème: Comment distinguer les ali?

Le numéro de client est un identifiant clés. Un identifiant caractérise de façon unique une entité

Unicité: Identifiant

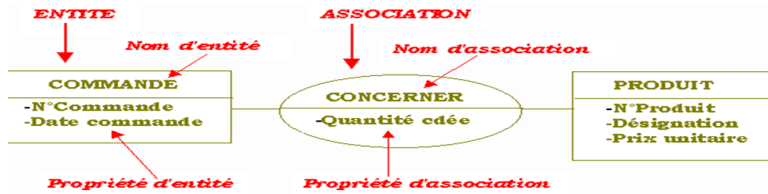
Un identifiant caractérise de façon unique une entité

- Cas 1 : L'identifiant est formé d'un ou plusieurs attributs de l'entité à identifier.
 - Le Numéro d'immatriculation est la clé de l'entité Véhicule.
 - Les Nom et prénom sont les attributs clé de l'entité Auteur.
- Cas 2 : on ajoute à l'entité un attribut artificiel, un numéro arbitraire dont l'unicité est garantie (par la BD)

Par convention, l'identifiant d'entité est souligné

Le modèle Entité-Association, les concepts de base: association (I)

Une **association** (ou type d'association) représente un lien quelconque entre différents entités

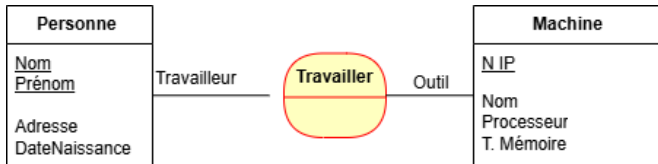


Une **occurrence** d'une association est un élément particulier de l'association constitué d'une et une seule occurrence des objets participants à l'association

Le modèle Entité-Association, les concepts de base: association (I)

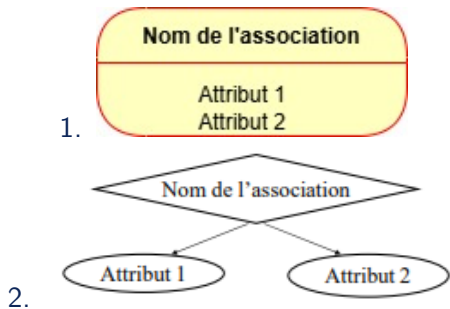
Ahmed travaille sur la machine Asus

- L'association **TRAVAILLE** relie la personne **Ahmed** à la machine Asus.
- Ahmed: joue **le rôle de travailleur** dans l'association TRAVAILLE
- La machine **Asus**: joue le rôle de l'**outil** de travail dans l'association TRAVAILLE



Le modèle Entité-Association, les concepts de base: association (I)

Représentation graphique des associations :



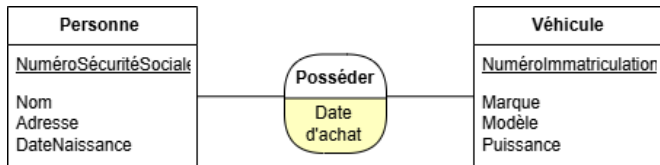
Le modèle Entité-Association, les concepts de base: association (I)

Une association peut avoir ses propres attributs.

Exemples :

Commande(Client, Produit, **Quantité**)

Posséder(Personne, Véhicule, **Date d'achat**):



Le modèle Entité-Association, les concepts de base: association (II)

Le degré d'une association est le nombre d'entités y participant.

Exemple :

Ecriture(Auteur, article) degré = 2
Cours(Salle, Groupe, Prof) degré = 3

Types d'associations:

- Association réflexive
- Association Binaire
- Association ternaire / n-aire

Le modèle Entité-Associations: associations, cardinalités

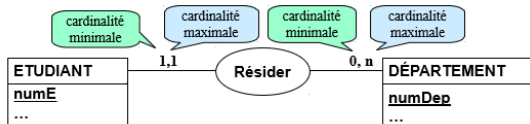
Contraintes sur les associations : "contraintes structurelles"

Des cardinalités précisent la participation de l'entité à l'association

- La **cardinalité minimale** peut être de 0 ou de 1
- La **cardinalité maximale** peut être de 1 ou de n

Un étudiant réside dans un et un seul département (1,1)

Un département a pour résident aucun ou plusieurs étudiants (0, n)

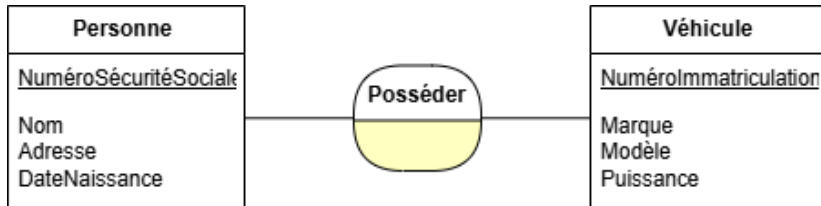


Le modèle Entité-Associations: associations, cardinalités

Exemple2: La Gestion des Cartes Grises

Quelle est la Cardinalité ?

0,1 / 1,1 / 0,n / 1,n / n,m

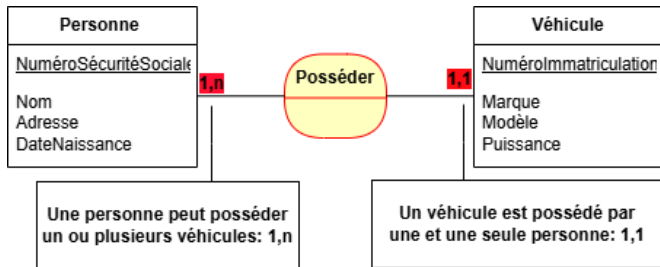


Le modèle Entité-Associations: associations, cardinalités

Exemple2: La Gestion des Cartes Grises

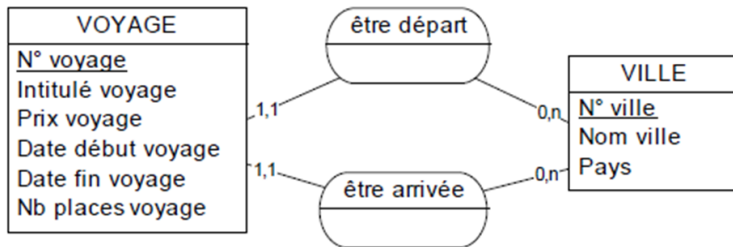
Quelle est la Cardinalité ?

0,1 / 1,1 / 0,n / 1,n / n,m



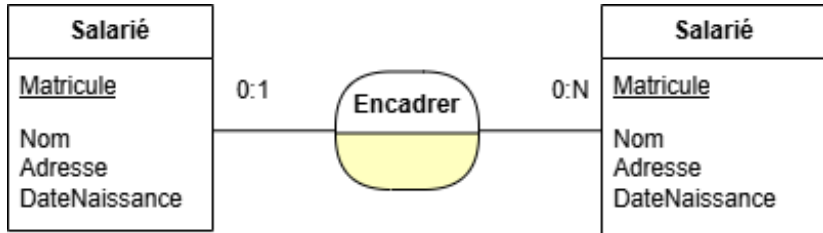
Le modèle Entité-Association: les associations, suite

On peut avoir plusieurs associations définies sur **deux entités**.



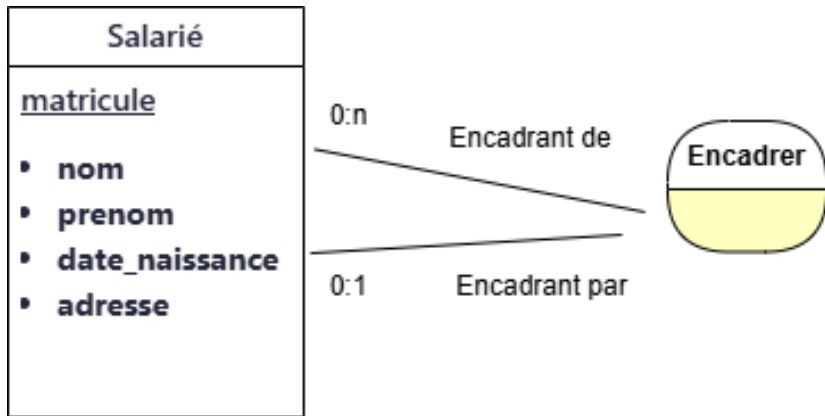
Le modèle Entité-Association, Association réflexive

Dans une usine, un salarié peut encadrer ses collègues.

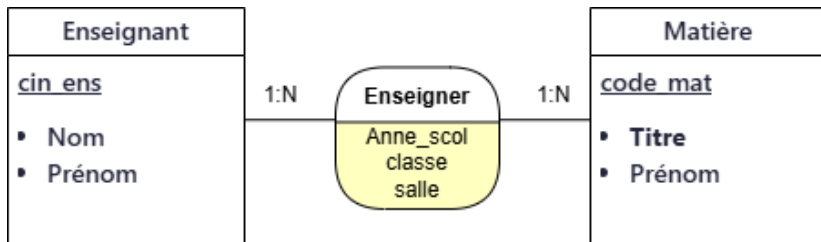


Le modèle Entité-Association, Association réflexive

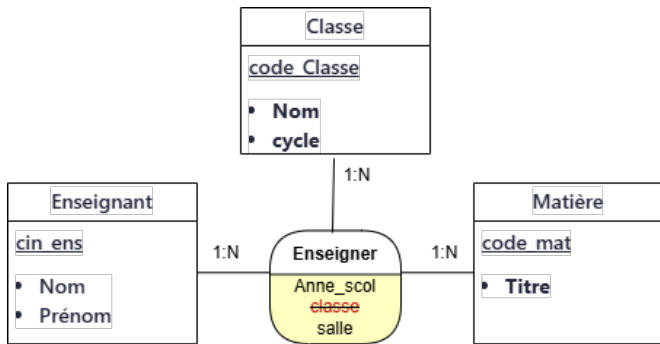
Une association réflexive (ou unaire) est une association qui relie une entité à elle même.



Le modèle Entité-Association, Association binaire/ternaire

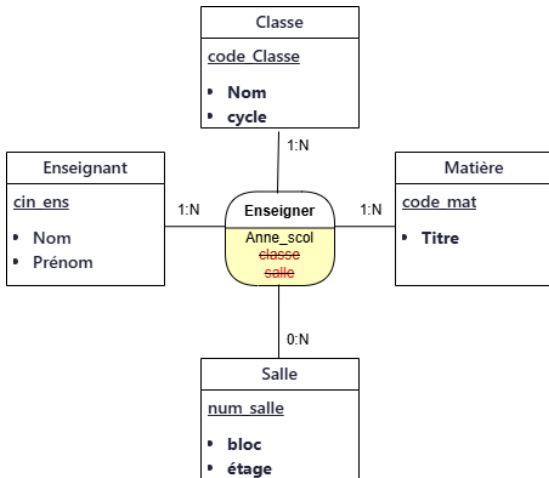


Le modèle Entité-Association, Association binaire/ternaire

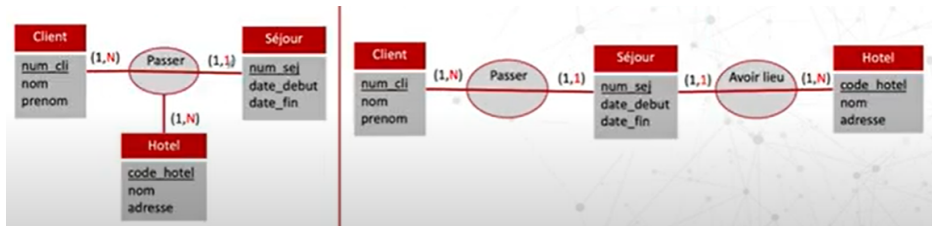


Ce modèle MCD permet de connaître pour chaque année scolaire, quelle matière est enseignée pour quelle classe et par quel enseignant.

Le modèle Entité-Association, Association Binaire/n-aire



Le modèle Entité-Association, Fausse Association ternaire

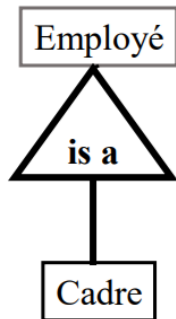


Elle est considérée **non optimisée** toute relation ternaire dont une entité possède une cardinalité max à 1.

Dans ce cas, il faut décomposer le relation ternaire en 2 relations binaires."

Le modèle Entité-Association, Concept généralisation/spécialisation

- Un lien, orienté, d'une **E spécialisé** (sous-type ou spécifique) vers une **E générique** (sur-type).
- Un sous-type entité hérite de tous les attributs de son sur-type.
- A toute occurrence d'une E sous-type correspond une occurrence de l'E sur-type.

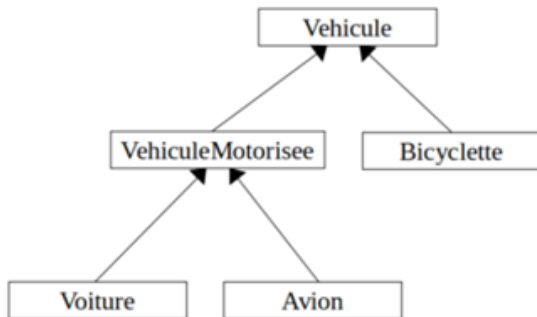


Le modèle Entité-Association: Concept généralisation/spécialisation

Les attributs communs au TE générique et aux TE spécifiques ne sont décrits, dans le schéma, que comme attributs du TE générique.

- Les E spécifiques peuvent avoir des attributs propres.
- Un sous-type entité participe aussi à toutes les associations auxquelles participe son sur-type.
- Un sous-type peut être sur-type d'un autre type.

Le modèle E-A, Concept généralisation/spécialisation: Exemple



Le modèle Entité-Association: Remarques

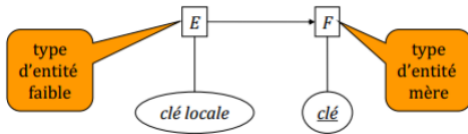
- Un attribut peut être composé hiérarchiquement de plusieurs autres attributs

Exemple : attribut composite "adresse"

- numéro
- rue
- no-appt
- ville
- pays
- code postal

- Un attribut peut être monovalué ou multivalué Exemple : l'âge des enfants d'un employé
- La valeur d'un attribut peut être dérivée d'une ou plusieurs autres valeurs d'attributs. Exemple : l'âge d'une personne peut être dérivé de la date du jour et de celle de sa naissance

Le modèle Entité-Association: Entité faible

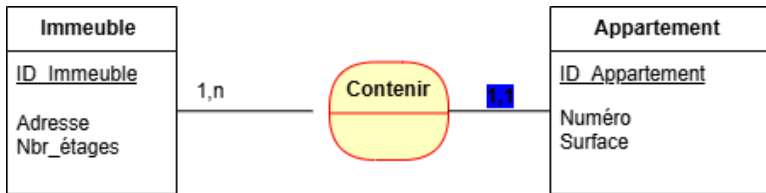


Une entité faible est une entité dépendante d'une autre entité de cardinalité max de 1.

- Une entité faible est une entité qui n'existe que comme composante d'une autre entité : son entité mère.
- Lorsqu'elle celle-ci est supprimée, elle doit l'être aussi.
- Une entité faible a une clé locale qui l'identifie parmi les entités faibles du même type de son entité mère.
- La clé d'une entité faible est donc la concaténation de sa clé locale et de la clé de son entité mère.

Le modèle Entité-Association: Entité faible

- L'entité Appartement est l'entité faible.
- L'entité Immeuble est l'entité dominante.
- L'association Contenir est appelé identifiante.
- La clé de l'entité faible Appartement est composée de sa clé partielle (qui est l'attribut *ID_Appartement*) plus la clé (*ID_Immeuble*) de l'entité dominante Immeuble.



La cardinalité de l'entité faible au sein de l'association identifiante est toujours (1,1)

Le modèle E/A: LA BASE GESTION DU PERSONNEL

Dans une base de données dédiée à la gestion du personnel, plusieurs aspects doivent être pris en compte :

1. Affectation administrative

Un employé appartient à un département.

Un employé peut avoir un supérieur hiérarchique.

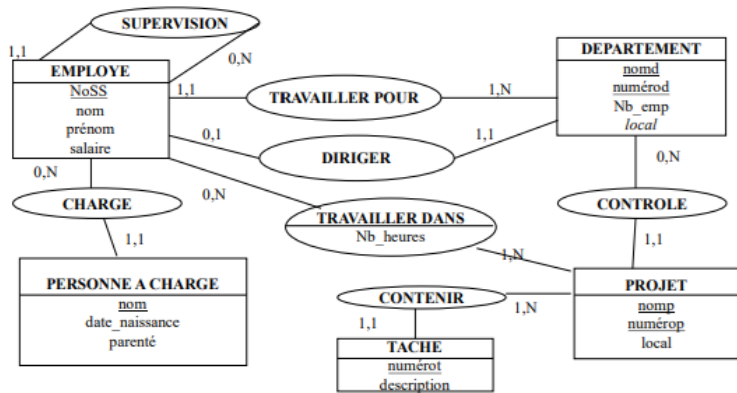
2. Charge de travail

Un employé peut travailler sur plusieurs projets.

3. Plan social (personnes à charge)

Un employé peut avoir des personnes à charge (enfants, conjoint, etc.).

Le modèle E/A: LA BASE GESTION DU PERSONNEL



SUPERVISION est réflexive sur EMPLOYE. Elle matérialise le fait qu'un EMPLOYE a un chef direct et un seul et qu'un EMPLOYE peut être le chef direct.

Le modèle Entité-Association: Synthèse

Face à un problème de modélisation **Entité-Association (E-A)**, procéder comme suit :

- Recueil des besoins
- Identifier les entités en utilisant leurs identifiants (clés)
- Établir les associations entre les entités
- Établir les cardinalités
- Lister les attributs des entités et des associations
- Éliminer les synonymes (plusieurs signifiants pour un même signifié, exemple : *Libellé Article*, *Nom Produit*)
 - C'est une tâche difficile à détecter.
 - Il faut trancher en choisissant un des deux termes.

Il faut garder à l'esprit que le modèle doit être exhaustif (c-à-d contenir toutes les informations nécessaires) et éviter toute redondance.

LE MODÈLE RELATIONNEL ET SON ALGÈBRE (partie 1)

Introduction: Le niveau logique

- **Le niveau logique** est le **lien** entre le **niveau conceptuel** et **l'implémentation** effective de l'application.
- Description de la structure de la base de données (en terme de tables, de lignes et de colonnes) obtenue en utilisant un modèle conceptuel
- Encore aujourd'hui dominant le modèle relationnel est un fondement indispensable à la conception de bases de données.

LE MODELE RELATIONNEL

Le modèle relationnel a été introduit par Codd , en 1970 au laboratoire de recherche d'IBM de San José.

- Il s'agit d'un modèle simple et puissant à la base de la majorité des bases de données, encore aujourd'hui.

Le modèle relationnel inclut des concepts pour la **description** des données :

- **Les relations ou tables**

- Les domaines de valeurs
- Les clés primaires

- il existe des cases dont les valeurs doivent être uniques et non nulles

- **Les clés étrangères**

- il existe des cases qui doivent prendre une valeur existante dans les cases d'une autre table

Les concepts de bases: Les relations ou tables

Une Base de Données (BD) est un ensemble d'informations que l'on peut représenter, intuitivement, sous forme de lignes et de colonnes qui sont regroupées en tables (ou **relations**).

On parle alors de **BD Relationnelle**, du **MODELE RELATIONNEL** de DONNEES.

Table1 / Relation1

Table3 / Relation3

Table2 / Relation2

Une ligne
ou
un tuple ou n-uplet
ou
un **enregistrement**

Clé primaire : colonne(s) où les valeurs en double ne sont pas admises

Clé étrangère : colonne(s) dont les valeurs font référence à une clé primaire

Les concepts de bases: Schéma de Table

Schéma de Table

Un schéma de table est constitué du nom de la table suivi de la liste de ses colonnes et de la définition de leurs domaines.

Exemples

Le schéma de la table CLIENTS est: **CLIENTS** (**CODECLI** : NUMERIQUE, **NOMCLI** : TEXTE, **TELCLI** : NUMERIQUE, **ADRCLI** : TEXTE)

Les concepts de bases: Domaine

Domaine

Un domaine est un ensemble de valeurs caractérisé par un nom.

Exemples

Le domaine MARQUESVOITURES = {'Peugeot', 'Citroën'}.

Le domaine MODELESVOITURES = {206, 307, 406, 'C2', 'C3'}.

Les concepts de bases: Produit Cartésien

Produit Cartésien

En mathématiques, le produit cartésien de deux ensembles X et Y est l'ensemble de tous les couples dont la première composante appartient à X et la seconde à Y .

Exemples

$$D_1 = \{A, B, C\}$$

$$D_2 = \{1, 2, 3\}$$

$$\begin{aligned} D_1 \times D_2 = \{ & \langle A, 1 \rangle, \langle A, 2 \rangle, \langle A, 3 \rangle, \\ & \langle B, 1 \rangle, \langle B, 2 \rangle, \langle B, 3 \rangle, \\ & \langle C, 1 \rangle, \langle C, 2 \rangle, \langle C, 3 \rangle \} \end{aligned}$$

Concept de bases: Relation

Relation

Une Relation est un sous ensemble du produit cartésien de domaines.

Une relation est désignée par un nom

Exemples

la relation Etudiant est un sous-ensemble de : Nom X Prénom X No-Carte X Sexe
où Nom, Prénom, No-Carte et Sexe sont des domaines.

Concept de bases: Relation

Une **relation** est une table dans laquelle chaque colonne correspond à un domaine.

Un attribut prend ses valeurs dans un DOMAINE (soit l'ensemble de ses valeurs possibles)

The diagram shows a table with four columns: CODE_CLIENT, NOM, AD_RUE, and AD_VILLE. The first column contains three values: AXV508, BWY405, and KAC567. The second column contains three values: DUPONT, DURAND, and DUVAL. The third column contains three values: 15, rue du Canada, 6, rue Méchin, and 25 rue Blanche. The fourth column contains three values: CAEN, CHERBOURG, and ALENCON. A red circle highlights the first column, with an arrow pointing to it from the word 'ATTRIBUT' in red. A red rectangle highlights the entire table, with an arrow pointing to it from the word 'N-UPLET' in red. A red arrow points to the first row of data, with the text 'Valeurs d'attribut' in red.

CODE_CLIENT	NOM	AD_RUE	AD_VILLE
AXV508	DUPONT	15, rue du Canada	CAEN
BWY405	DURAND	6, rue Méchin	CHERBOURG
KAC567	DUVAL	25 rue Blanche	ALENCON

Une relation respecte les règles suivantes :

- Elle ne contient pas deux lignes identiques (qui ont les mêmes valeurs).
- Elle ne contient pas deux colonnes identiques (qui portent le même nom).
- Chacune de ses cases est **atomique** : chaque case (intersection d'une ligne et d'une colonne) ne contient qu'une seule valeur à la fois (jamais une liste de valeurs).

Concept de bases: Schéma de Relation

Le schéma d'une relation spécifie le nom de la relation ainsi que le nom et le type de chaque attribut.

$$R(A_1 : D_1, A_2 : D_2, \dots, A_n : D_n)$$

Remarque : Le plus souvent, lorsqu'on définit le schéma d'une relation, les domaines sont implicites et découlent du nom de l'attribut. On omet donc de les présenter.

- **Schéma simplifié :**

ETUDIANT(Uid, Nom, Date_naiss, Groupe)

- **Schéma détaillé avec types de données :**

ETUDIANT(Uid:integer, Nom:string, Date_naiss:date, Groupe:char(2))

Modèle Logique des Données (MLD)

Une BD relationnelle est une BD dont le schéma est un **ensemble** de schémas de relations et dont les occurrences sont des tuples ou n-uplets (lignes des tables) de ces relations.

- Les entités et les associations du modèle E/A sont représentées exclusivement par des relations (des tables).
 - **Une entité** est représentée par sa liste d'attributs.
Notation : Schéma de la relation CLIENT
CLIENT (Numclient, Nom, Adresse, Solde)
 - **Une association** est représentée par la liste des clés des entités qu'elle associe et ses propres attributs.
Notation : Schéma de la relation Facturer
Facturer (#Numclient, #Numfacture, Période)

Passage du MCD au MLD

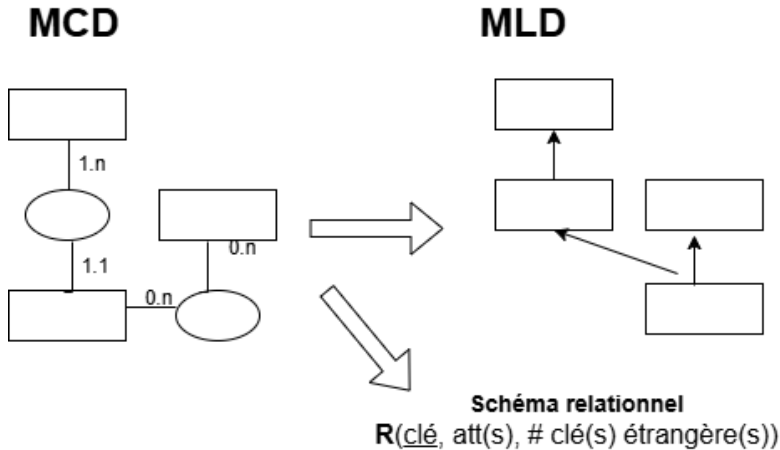
Rappel :

Le modèle Entité-Association est basé sur trois concepts : entité, association et attribut.

Comparaison modèles E/A et relationnel

Concept	Description
entité	relation
occurrence d'une entité	nuplet
attribut simple	attribut atomique
attribut composite	simulé par un ensemble d'attributs
entité faible	relation
association	relation
occurrence d'une association	nuplet

Passage du MCD au MLD



Règles de Passage

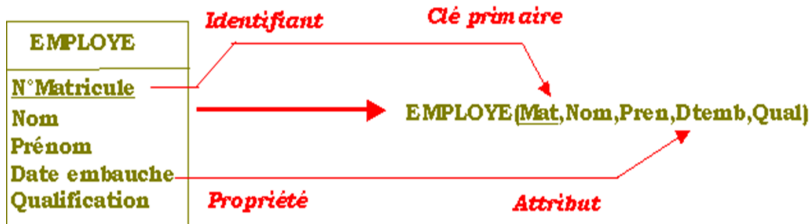
Il existe différentes règles pour déduire un MLD à partir d'un MCD:

- Règles classiques
- Cas particuliers
- Transformation de l'héritage

Règle 1: entités

TOUTE ENTITE DEVIENT UNE RELATION dans laquelle :

- les attributs traduisent les propriétés de l'entité
- la clé primaire traduit l'identifiant de l'entité

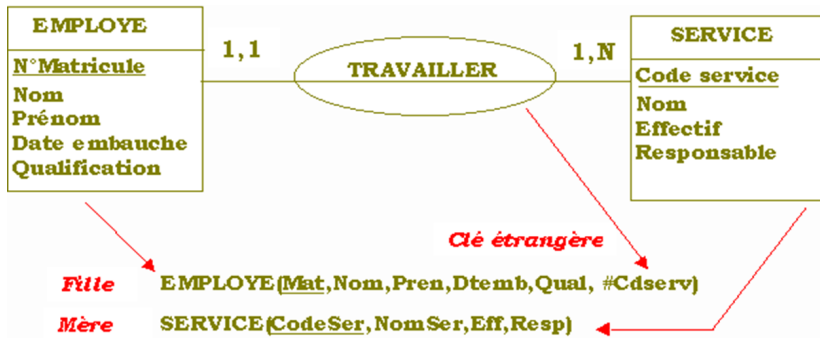


Règle 2: associations 1:n

UNE ASSOCIATION Binaire avec cardinalité 1,1 SE RÉÉCRIT EN :

Portant dans la relation fille la clé primaire de la relation mère.

L'attribut ainsi ajouté s'appelle **clé étrangère**. Symbole : #.

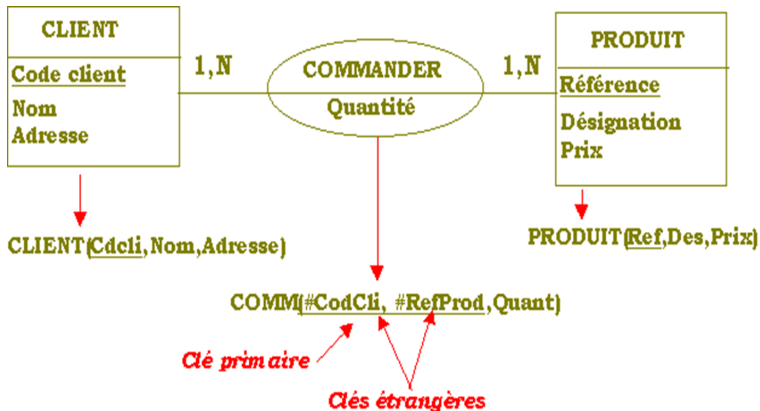


Règle 3: associations n:m

UNE ASSOCIATION BINAIRE AVEC CARDINALITÉ PLUSIEURS À PLUSIEURS SE RÉÉCRIT EN :

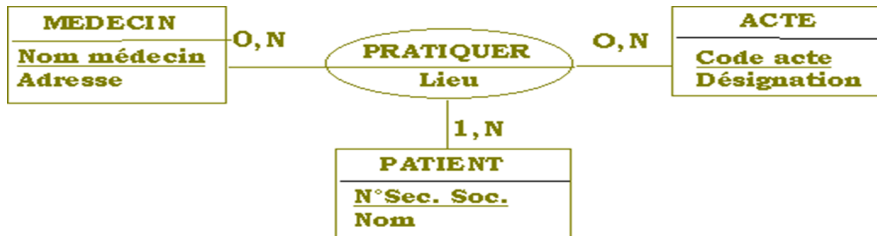
- Créant une relation particulière qui contient comme attributs les identifiants des deux entités associées.
- Ces attributs constituent à eux deux la **clé primaire** de la relation.
 - Ils sont individuellement des **clés étrangères**.
 - Ajoutant la ou les éventuelles propriétés de l'association à cette relation.

Règle 3: associations n:m



Règle 3: associations n:m (suite)

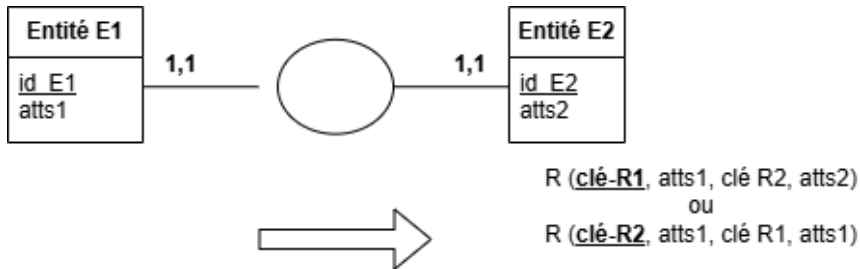
UNE ASSOCIATION DE DIMENSION SUPERIEURE A 2 SE REECRIT SELON LA REGLE 3



MEDECIN(NomMed, Adr)
ACTE(CdActe, Des)
PATIENT(SS, NomPat)
PRATIQUER(#Med, #Acte, #Secu, Lieu)

Cas particuliers (1/5)

Association 1:1 avec des cardinalités 1,1 des deux cotés se traduit par la fusion des deux relations relatives aux deux entités

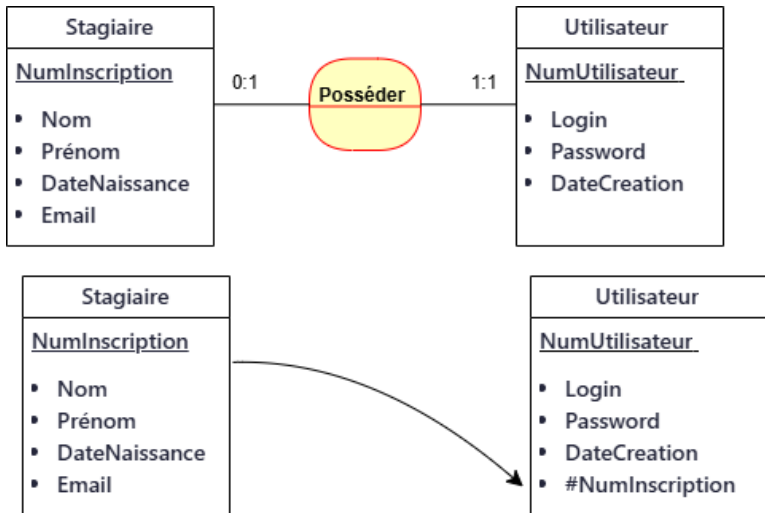


On dit a relation 1,1 est **permanente** → Exp: Une personne aura toujours une carte d'identité, et une carte appartient toujours à une seule personne.

La dépendance est forte → Une carte n'existe pas sans une personne, donc elle n'a pas besoin d'une entité séparée.

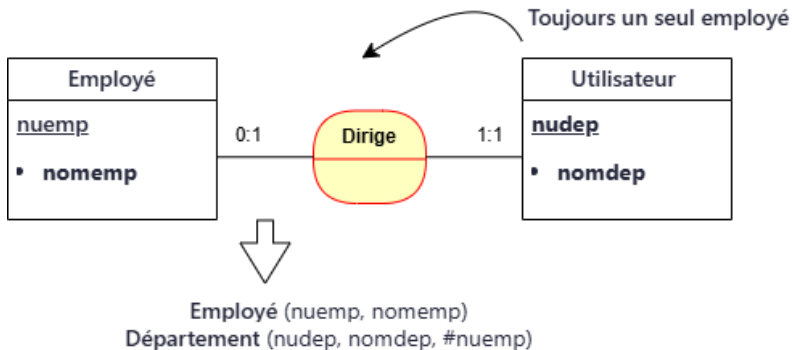
Cas particuliers (2/5)

Association 1:1 avec des cardinalités 1,1 et 0,1



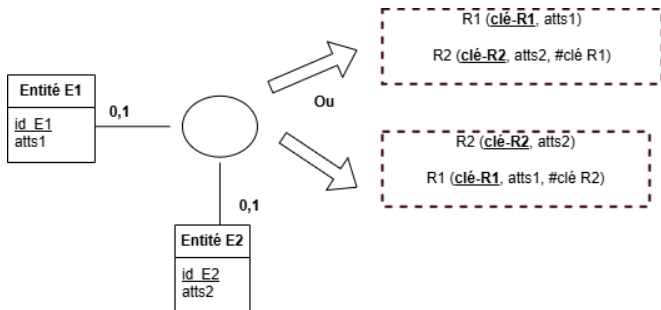
Cas particuliers (2/5)

Association 1:1. Exemple 2



Cas particuliers (3/5)

association 1:1 avec des cardinalités 0,1 des deux cotés 2 solutions au choix:

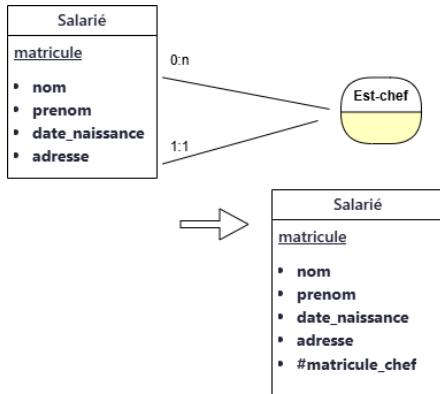


Une instance de l'entité A peut être liée à une instance de l'entité B, mais ce n'est pas obligatoire.

De même, une instance de l'entité B peut être liée à une instance de l'entité A, mais ce n'est pas obligatoire non plus exp:Voiture et Place de Parking).

Cas particuliers (4/5)

association réflexive 1:n



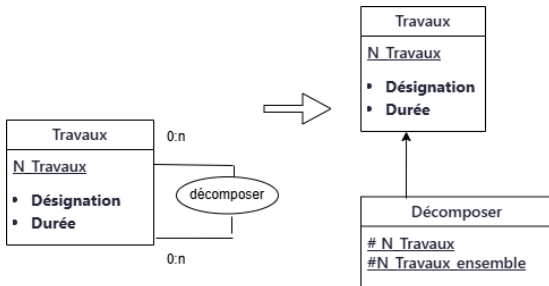
on duplique la clé de la relation (avec un nom différent)

Cas particuliers (5/5)

Association réflexive n:n

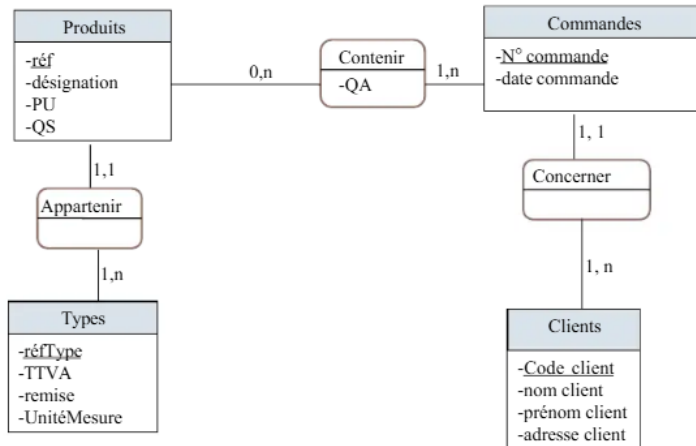
un travail est composé d'un ensemble de travaux

- un travail est un élément d'autre travail



Créer une relation décomposer dont la clé est composé de deux fois la clé de la relation Travaux

Exercices



Traduire ce MCD en LMD?

Exercices

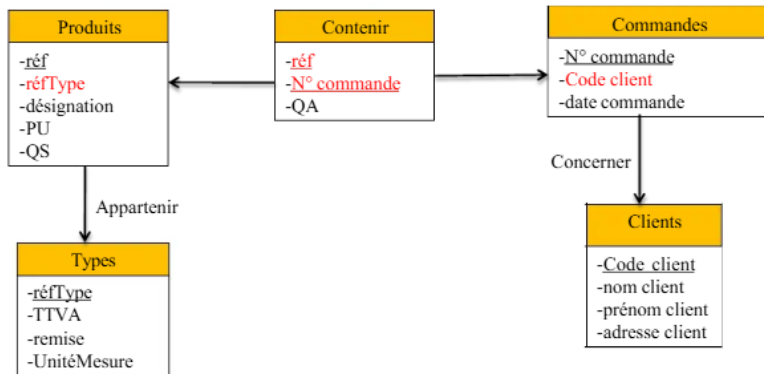


Schéma relationnel:

Produits (réf, réfType, désignation, PU, QS) ;

Contenir (réf, N°commande, QA);

Commandes (N° commande, Code client, date commande);

Clients (Code client, nom client, prénom client, adresse client);

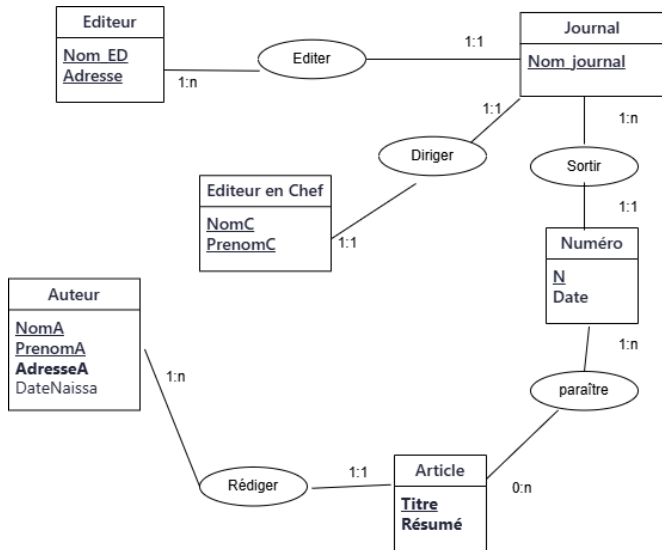
Types (réfType, TTVA, remise, UnitéMesure);

Exercices

Conception d'une base de données destinée à conserver des descriptions d'articles parus dans les journaux.

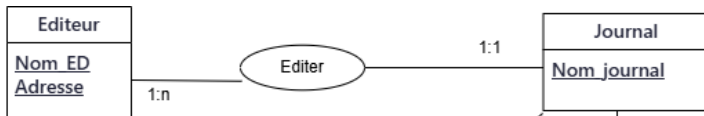
1. Un **éditeur** édite des **journaux**. Il est caractérisé par un **nom** et une **adresse**.
2. Un **journal** est édité par un **éditeur** et publie des **articles** dans ses **numéros**. On conservera le nom du journal et le **nom** de son **rédacteur en chef**.
3. Un **numéro de journal** contient une collection d'articles.
4. **Chaque article** paru dans un **numéro** est signé par un **auteur**. On désire conserver le **titre** et un **résumé** de l'article ainsi que le **nom** de son auteur.
5. Les **auteurs** sont connus par leur **nom**, leur **prénom**, leur **adresse** et leur **date de naissance**.

Exercices



Exercices

Solution: (association avec cardinalités 1:1)



EDITEUR (Nom_Ed, Adresse)

JOURNAL (Nom_J, #Nom_Ed, #Adresse)

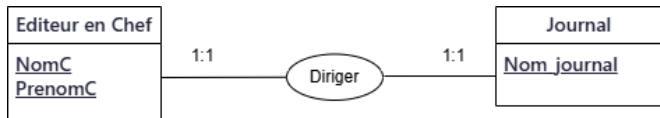
car EDITER est une association avec cardinalités (1,1).

→ **La Clé primaire de JOURNAL** : Nom_J

Clés étrangères : #Nom_Ed, #Adresse

Exercices

Solution: (association avec cardinalités 1:1)

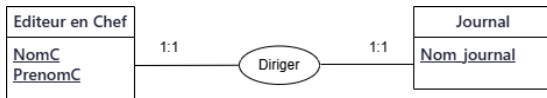


Remarque

- Il existe toujours deux solutions selon que l'on choisit l'une ou l'autre relation pour accueillir la clé étrangère. Selon la cardinalité minimale, un des deux choix peut être plus pertinent.
- Il est parfois possible de choisir de fusionner les deux entités au sein d'une seule relation plutôt que d'opter pour une clé étrangère.

Exercices

Solution: (association avec cardinalités 1:1)



EDITEUR EN CHEF (NomC, PrénomC)

JOURNAL (Nom_J, #Nom_Ed, #Adresse, NomC, PrénomC)

car Diriger est une association avec deux cardinalités (1,1).

Dans notre exercice, on peut garder l'entité **JOURNAL** et créer deux attributs **NomC** et **PrénomC**.

Exercices

Solution: (association avec cardinalités 1:1)

JOURNAL (Nom_J, #Nom_Ed, #Adresse, NomC, PrénomC)

NUMERO (N°, #Nom_J, Date)

AUTEUR (Nom_A, Prénom_A, Adresse_A, Date de naissance)

ARTICLE (Titre, #Nom_A, #Prénom_A, #Adresse_A, Résumé)

Solution: (association avec cardinalités n:n)

ARTICLE (Titre , #Nom_A, #Prénom_A, #Adresse_A, Résumé)

NUMERO (N° , #Nom_J, Date)

Exercices

EDITEUR (**Nom_Ed**, Adresse)

EDITEUR EN CHEF (**NomC**, PrénomC)

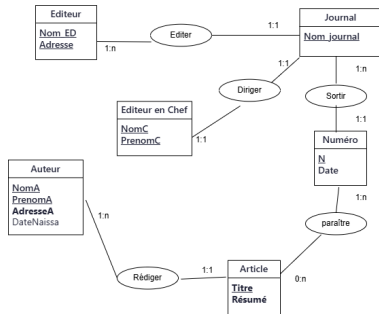
JOURNAL (**Nom_J**, #Nom_Ed, #Adresse,
#NomC, #PrénomC)

AUTEUR (**Nom_A**, Prénom_A,
Adresse_A, Date de naissance)

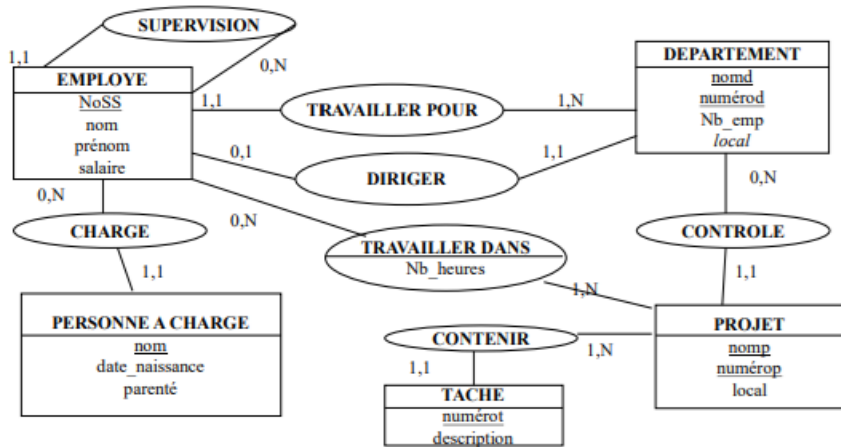
ARTICLE (**Titre**, #Nom_A, #Prénom_A,
#Adresse_A, Résumé)

NUMERO (**N°**, #Nom_J, Date)

PARUTION (#**Titre**, **Nom_A**,
#Prénom_A, #Adresse_A, #N°,
#Nom_J)



Exercices: La Base gestion du personnel



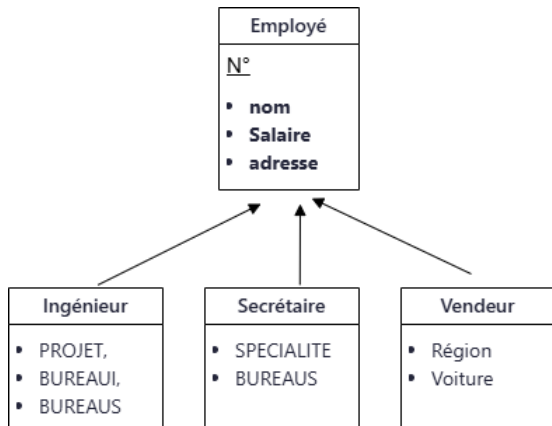
Traduction de l'héritage

Trois solutions possibles :

1. **Conserver uniquement la super-entité** (ex: **Employé**) et utilisation de la valeur **NULL**

EMPLOYEE(N°, NOM, ADRESSE, SALAIRE, PROJET, BUREAU1, BUREAUS, SPECIALITE, REGION, VOITURE)

Par exemple, la valeur de l'attribut **Voiture** est égale à **NULL** pour ingénieur et secrétaire.



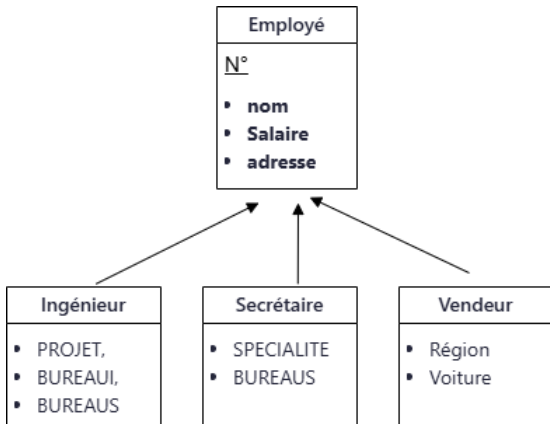
Traduction de l'héritage

Trois solutions possibles :

1. **Conserver uniquement la super-entité** (ex: **Employé**) et utilisation de la valeur **NULL**

EMPLOYEE(N°, NOM, ADRESSE, SALAIRE, PROJET, BUREAU, BUREAUX, SPECIALITE, REGION, VOITURE)

- Utilisable lorsque les spécialisations ne sont pas réellement utilisées
- Pas d'attribut ou peu (utilisation de la valeur null),
- pas d'association

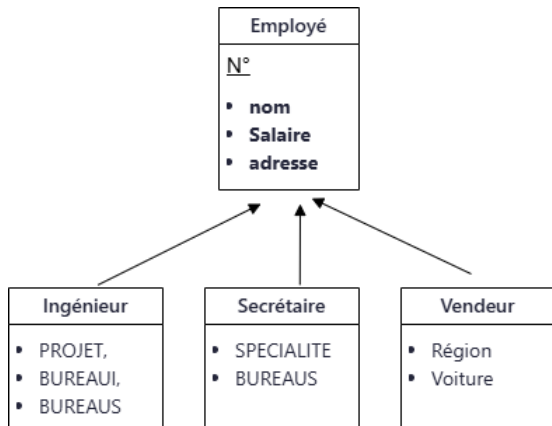


Traduction de l'héritage

Trois solutions possibles :

2. Conserver uniquement les spécialisations

INGENIEUR(N°, NOM, ADRESSE, SALAIRE, PROJET, BUREAU) SECRETAIRE(N°, NOM, ADRESSE, SALAIRE, BUREAU, SPECIALITE)
VENDEUR(N°, NOM, ADRESSE, SALAIRE, REGION, VOITURE)



- Utilisable lorsque la super-entité n'est pas utilisée
- Généralisation/spécialisation totale et
- disjointe

Traduction de l'héritage

Trois solutions possibles :

3. Conserver toutes les entités

Choix 1 : le schéma est factorisé (seule la clé (N°) est partagée)

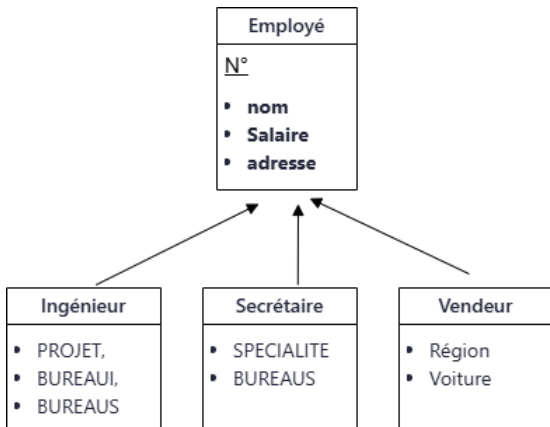
Clés étrangères pour les spécialisations (N°*)

EMPLOYE(N°, NOM, ADRESSE, SALAIRE)

INGENIEUR(N°*, PROJET, BUREAU)

SECRETAIRE(N°*, BUREAU, SPECIALITE)

VENDEUR(N°*, REGION, VOITURE)



Traduction de l'héritage

Trois solutions possibles :

3. Conserver toutes les entités

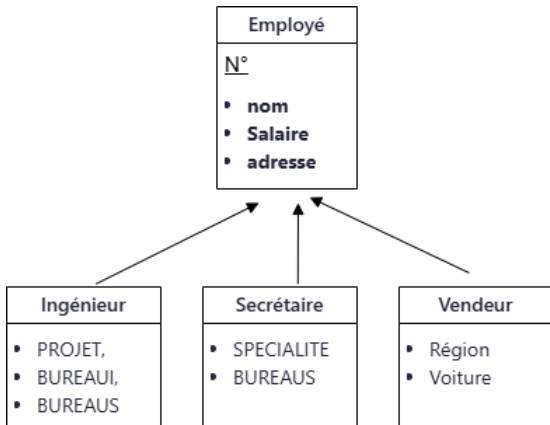
Choix 2 : le schéma est dupliqué

Clés étrangères pour les spécialisations (N°*)

EMPLOYE(N°, NOM, ADRESSE, SALAIRE)
INGENIEUR(N°, NOM, ADRESSE, SALAIRE, PROJET, BUREAU)
SECRETAIRE(N°, NOM, ADRESSE, SALAIRE, BUREAU, SPECIALITE)
VENDEUR(N°, NOM, ADRESSE, SALAIRE, REGION, VOITURE)

Inconvénient

Information redondante



Introduction

Exemple1: gestion des Clients inscrits dans des centres sportifs

NUM_CENTRE	NOM_CENTRE	COUT_INSCRI	NUM_CLIENT	PREN_CLIENT	NOM_CLIENT
103	Gym Center	230,00	002	Omar	SHERIFF
107	Sport Center	260,00	002	Omar	SHERIFF
107	Sport Center	260,00	004	Hedi	MUFTI
107	Sport Center	260,00	007	Syn	KONNERY
107	Sport Center	260,00	008	Woody	HELENE

Problème : Cette table contient des informations redondantes. Elle génère les problèmes de mise à jour suivants :

- **Anomalie de modification** : la modification du coût d'inscription (COUT_INSCRI) pour le centre 107 nécessite la modification de plusieurs lignes au lieu d'une seule.
- **Anomalie de suppression** : la suppression dans la table du client n°002 peut entraîner la perte des informations concernant le centre 103.

Introduction

Multiples modélisations possibles du monde réel

Exemple : BD représentant un Parc Automobile

Modélisation suivant le modèle Entité-Association :

- Peut être modélisée par une seule relation :

Condveh(NSS, Nom, Prénom, Adresse, NVH, Type, Marque, Couleur, Puiss, prix, date)

- Certains choix sont mauvais ...

Introduction

- Redondance des données : Incohérence en mise à jour • Anomalie d'insertion : Comment représenter les personnes sans véhicule
- Anomalie de suppression : Perte d'information

Introduction

Nécessité d'une "bonne" conception



Appliquer des règles

Solution basée sur les Dépendances Fonctionnelles & Normalisation

Definitions

La normalisation est un processus qui permet de s'assurer de la "Bonne conception" du Schéma de la BD (c'est-à-dire de la non redondance de ses données).

L'objectif de la normalisation est de limiter le nombre de redondances et les anomalies lors de la mise à jour de Base de Données.

Definitions

Concepts

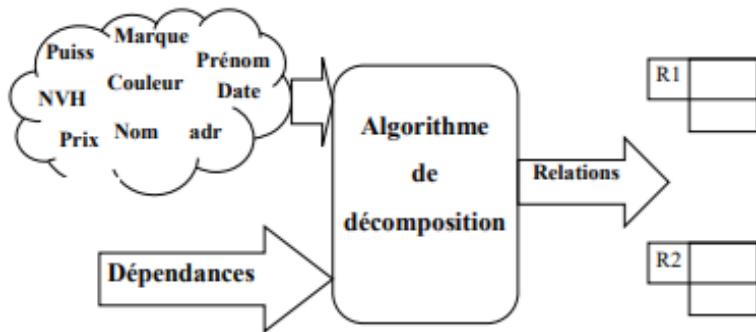
- Les dépendances fonctionnelles traduisent des contraintes sur les données.
- Les formes normales définissent des relations bien conçues.
- Processus de normalisation : Décomposer progressivement les relations jusqu'à obtenir des relations normalisées.

La normalisation

- Consiste à faire éclater une relation en plusieurs relations plus réduites grâce aux formes normales, qui sont basées sur les dépendances fonctionnelles entre attributs.

Algorithmes de décomposition

Relation Universelle



Definitions

References

- H. Garcia Molina, J.D. Ullman et J. Widom: Database Systems - The Complete Book, Prentice Hall, 2002
- R.A. El Masri et S.B. Navathe : Fundamentals of Database Systems, Fourth Edition; Prentice Hall C.J. Date : An introduction to Database Systems; Pearson Education 2004