

## I- BASE DE DONNEES :

### I-1 Définition 01 :

Un ensemble organisé d'informations avec un objectif commun. Peu importe le support utilisé pour rassembler et stocker les données (papier, fichiers, etc.), dès lors que des données sont rassemblées et stockées d'une manière organisée dans un but spécifique, on parle de base de données.

### I-2 Définition 02 :

Une base de données informatisée est un ensemble structuré de données enregistrées sur des supports accessibles par l'ordinateur, représentant des informations du monde réel et pouvant être interrogées et mises à jour par une communauté d'utilisateurs.

### Remarque :

*La gestion et l'accès à une base de données sont assurés par un ensemble de programmes qui constituent le Système de Gestion de Base de Données (SGBD).*

### I- 3 Processus De Conception D'une BD (base de données) :

- Analyse de la situation existante et des besoins (clarification)
- Création d'un modèle conceptuel qui permet de représenter tous les aspects importants du problème
- Traduction du modèle conceptuel en modèle logique (et normalisation de ce modèle logique)
- Implémentation d'une base de données dans un SGBD, à partir du modèle logique (et optimisation).

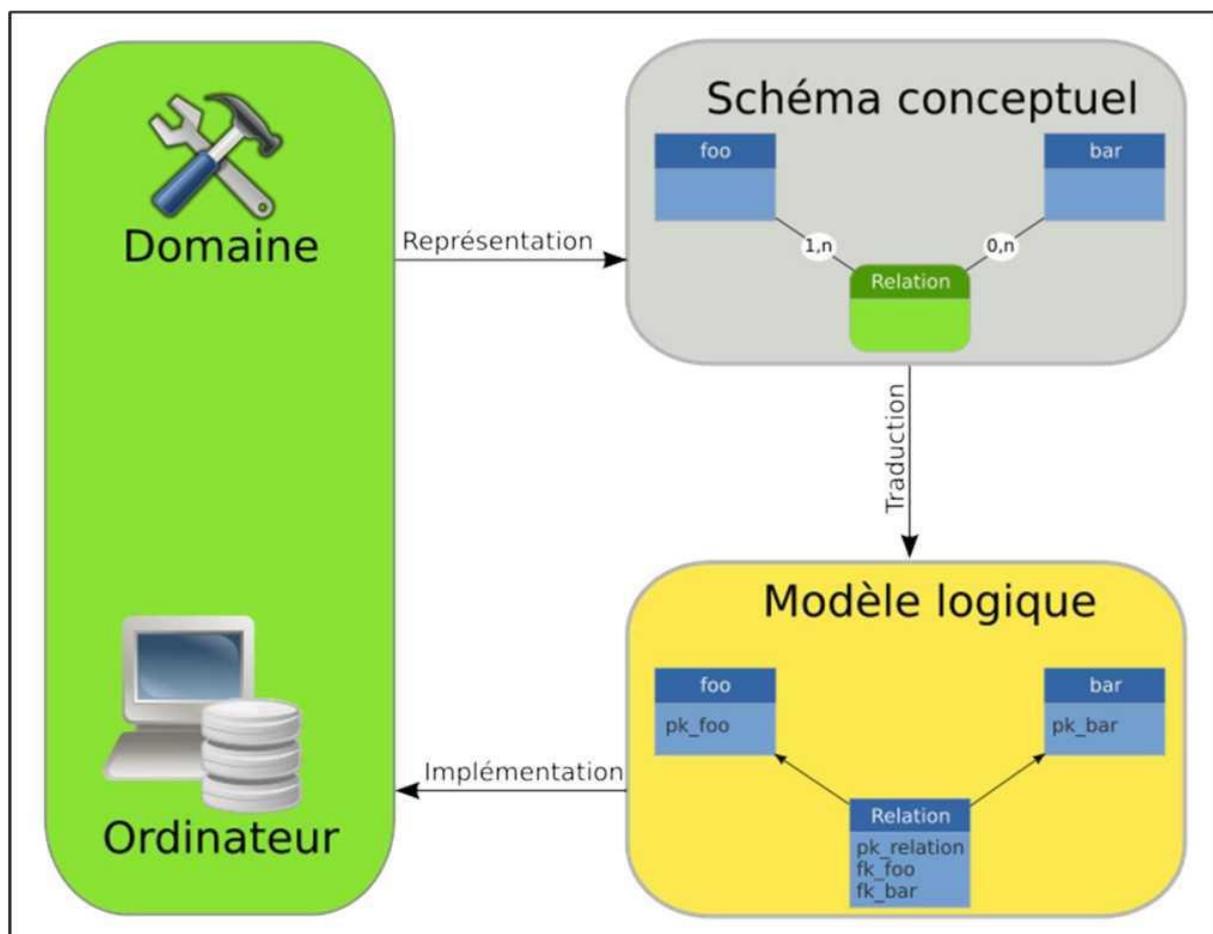
## II- Système De Gestion De Base De Données (SGBD) :

La gestion et l'accès à une base de données sont assurés par un ensemble de programmes qui constituent le Système de gestion de base de données (SGBD). Une fois la base de données spécifiée, on peut y insérer des données, les récupérer, les modifier et les détruire. C'est ce qu'on appelle manipuler les données.

### II-1 OBJECTIFS DU SGBD :

- **Indépendance physique** : La façon dont les données sont définies doit être indépendante des structures de stockage utilisées.

- **Indépendance logique** : Un même ensemble de données peut être vu différemment par des utilisateurs différents. Toutes ces visions personnelles des données doivent être intégrées dans une vision globale.
- **Accès aux données** : L'accès aux données se fait par l'intermédiaire d'un Langage de Manipulation de Données (LMD). Il est crucial que ce langage permette d'obtenir des réponses aux requêtes en un temps « raisonnable ». Le LMD doit donc être optimisé, minimiser le nombre d'accès disques, et tout cela de façon totalement transparente pour l'utilisateur.
- **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.
- **Cohérence des données** : Les données sont soumises à un certain nombre de contraintes 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 des données. Les contraintes d'intégrité sont décrites dans le Langage de Description de Données



## II-2 LE MODÈLE ENTITÉS - ASSOCIATIONS :

- Avant de modéliser un domaine sous une forme directement utilisable par un **GBD**, on passe par des modélisations intermédiaires, le modèle entités-associations constitue l'un des premiers et des plus courants. Ce modèle, permet une description naturelle du monde réel à partir des concepts d'entité et d'association. Ce modèle, utilisé pour la phase de conception, s'inscrit notamment dans le cadre d'une méthode plus générale : Merise.
- **MERISE** (*Méthode d'Étude et de Réalisation Informatique pour les Systèmes d'Entreprise*). Langage de spécification le plus répandu dans la communauté de l'informatique des systèmes d'information. Merise propose une démarche, dite par niveaux. Les trois niveaux de représentation des données sont:
- **Niveau conceptuel** : (MCD) décrit les entités du monde réel, en terme d'objets, de propriétés et de relations, indépendamment de toute technique d'organisation et d'implantation des données.
- **Niveau logique** : (MLD) précise le modèle conceptuel par des choix organisationnels. Il s'agit d'une transcription du MCD dans un formalisme adapté à une implémentation ultérieure.
- **Niveau physique** : (MPD) permet d'établir la manière concrète dont le système sera mis en place (SGBD retenu).

## II-3 ÉLÉMENTS CONSTITUTIFS DU MODÈLE ENTITÉS - ASSOCIATIONS:

La représentation du modèle entités-associations s'appuie sur trois concepts de base:

- **Entité:(Entité -Type):**

Une entité est un objet, une chose concrète ou abstraite qui peut être reconnue distinctement et qui est caractérisée par son unicité. Exemples d'entité : Mohamed, Omar, Livre de Math, Voiture 207 etc.

- **Association :(Association-Type ):**

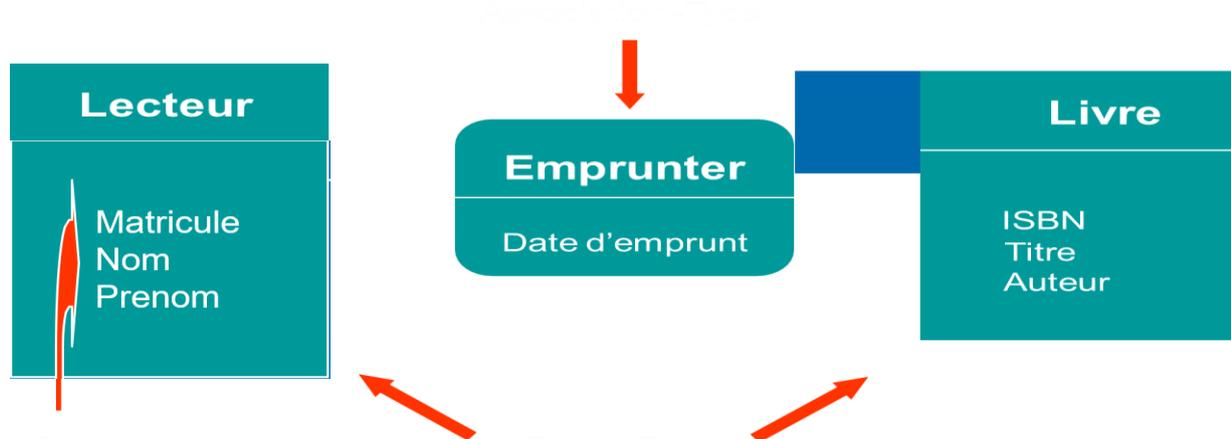
Une association (ou une relation) est un lien entre plusieurs entités. Exemples d'association : l'emprunt par l'étudiant Mohamed du 3 ème exemplaire du livre « Bases de Données ».

- **Propriété :(Valeur):**

Une propriété (ou un attribut ) est une caractéristique associée à un entité-type ou à une association-type. Exemples d'attributs : le nom d'une personne, le titre d'un livre, la puissance d'une voiture.

- **Cardinalités :**

Les cardinalités des rôles permettent de contraindre les associations par les nombres minimum et maximum de participation de chaque entité du TE (entité-type entité-type) à l'association. Les cardinalités peuvent être notées comme sur le schéma conceptuel ci-dessus min:max ou représentées graphiquement comme indiqué ci-dessous.



## II-4 RÈGLES :

- Un attribut ne peut en aucun cas être partagé par plusieurs entités-type ou associations-type.
- Un attribut est une donnée élémentaire, ce qui exclut des données calculées ou dérivées.
- Une entité-type et ses attributs doivent être cohérents entre eux (i.e. ne traiter que d'un seul sujet).

Par exemple, si le modèle doit comporter des informations relatives à des parcelles et à leur propriétaire, ces informations ne doivent pas coexister au sein d'une même entité-type. Il est préférable de mettre les informations relatives aux parcelles dans une entité-type Parcelle et les informations relatives aux propriétaires dans une entité-type Propriétaire.

## Règles De Bonne Formation D'un Modèle Entités-Associations

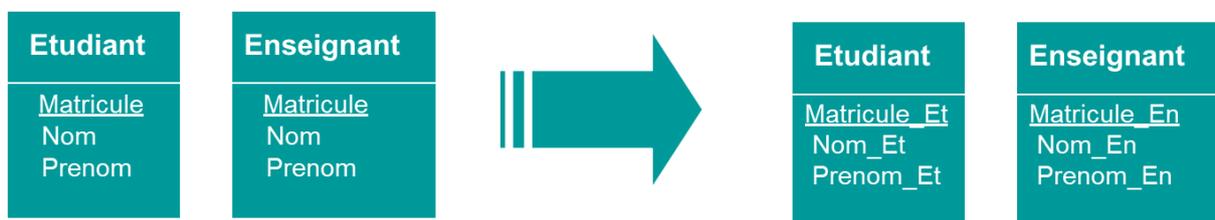
- La bonne formation d'un modèle entités-associations permet d'éviter une grande partie des sources d'incohérences et de redondance. Pour être bien formé, un modèle entités-associations doit respecter certaines règles et les entités et les associations doivent être normalisées. L'objectif des principes exposés dans cette section est d'aider le concepteur à obtenir un diagramme entités-associations bien formé, **ces principes ne doivent pas être interprétés comme des lois**

### A- Règles portant sur les noms

**R1** : Dans un modèle entités-associations, le nom d'une entité, d'une association ou d'un attribut doit être unique.

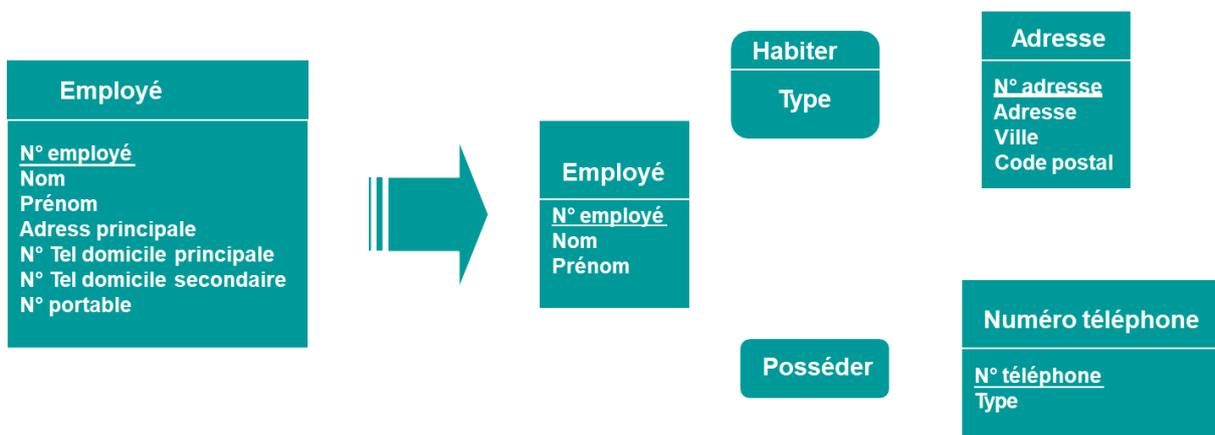


Si les deux entités doivent être séparées, on change les noms des propriétés

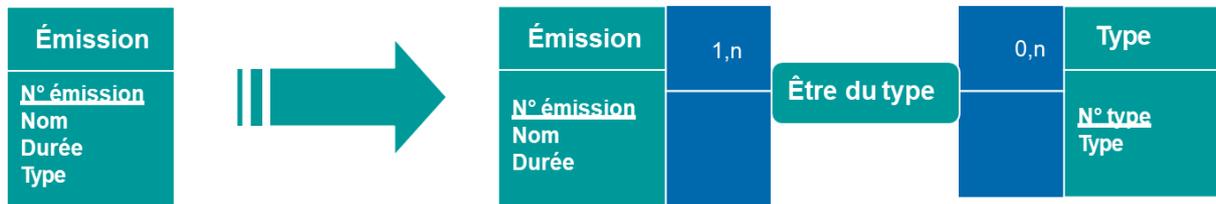


### B- Règles de normalisation des attributs

**R2** : Il faut remplacer un attribut multiple en une association et une entité supplémentaire



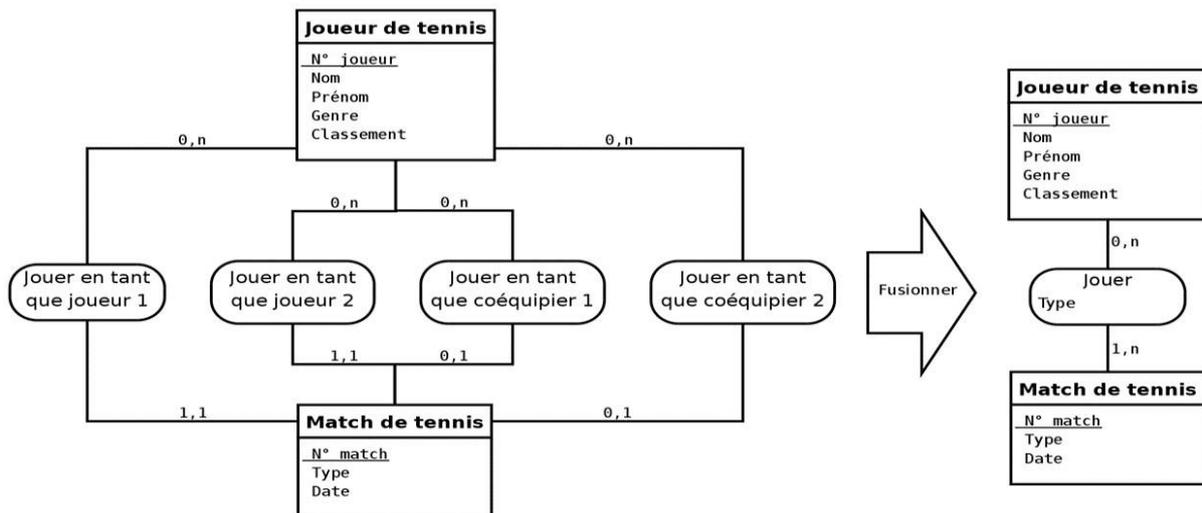
C- **R3** : Un attribut correspondant à un type énuméré est généralement remplacé par une entité.



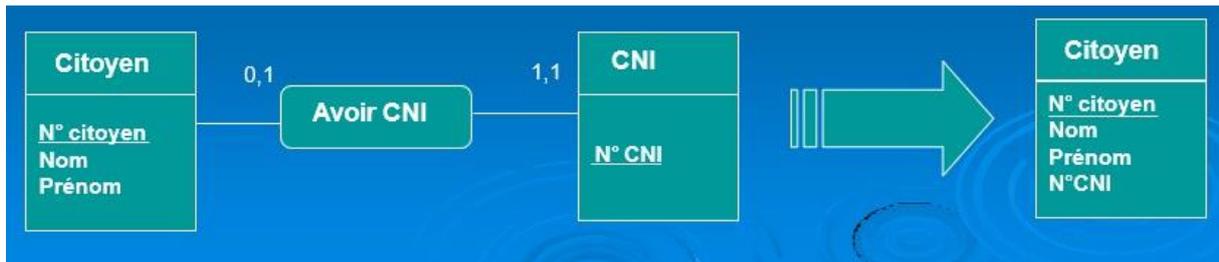
D- **R4** : Il faut factoriser les entités quand c'est possible. La spécialisation de l'entité obtenue peut se traduire par l'introduction d'un attribut supplémentaire dont l'ensemble des valeurs possibles est l'ensemble des noms des entités factorisées.



E- **R5** : Il faut factoriser les associations quand c'est possible



F- **R6** : Lorsque les cardinalités d'une association sont toutes 1,1 c'est que l'association n'a pas lieu d'être.



**Remarque :** même si toutes ses cardinalités maximales sont de 1, il est parfois préférable de ne pas supprimer l'association



### III- LE MODÈLE LOGIQUE DE DONNÉES (MLD):

**Définition :** Le modèle logique des données est une étape intermédiaire établie à partir du MCD pour avoir une représentation physique des données qui soit adaptée à un système de gestion d'une base de données relationnelles.

#### A. Clé primaire : -----

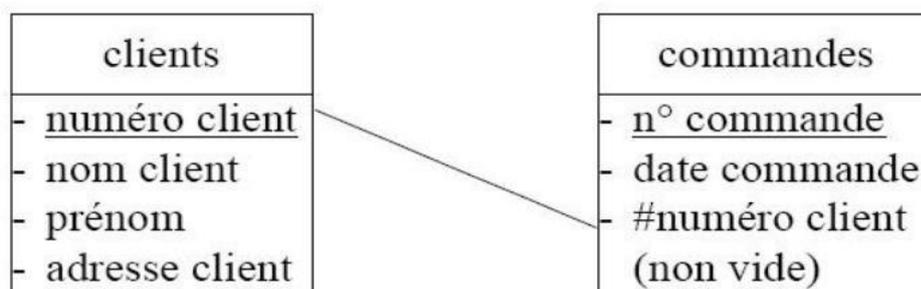
Les lignes d'une table sont uniques  $\Rightarrow$  il existe au moins une colonne qui sert à identifier les lignes : il s'agit de la clé primaire de la table. Propriétés requises :

- La valeur vide (NULL) est interdite
- la valeur de la clé primaire d'une ligne ne devrait pas changer au cours du temps

#### B. Clé étrangère : #####

Une clé étrangère identifie une colonne ou un ensemble de colonnes d'une table comme référant une colonne ou un ensemble de colonnes d'une autre table (la table référencée).

#### Exemple :



#### CONVENTIONS :

- On souligne les clés primaires
- On fait précéder d'un # les clés étrangères Par exemple dans la description des colonnes d'une table : clients (numclient, nom client, prénom, adresse) commandes (numcommande, date commande, # numclient (non vide))

#### REMARQUE :

- Une même table peut avoir plusieurs clés étrangères mais une seule clé primaire (éventuellement composée de plusieurs colonnes)

- Une clé étrangère peut aussi être primaire (dans la même table)
- Une clé étrangère peut être composée (c'est le cas si la clé primaire référencée est composée)
- Implicitement chaque colonne qui compose une clé primaire ne peut pas recevoir la valeur vide (NULL interdit)
- Par contre, si une clé étrangère ne doit pas recevoir la valeur vide, alors il faut le préciser dans la description des colonnes.

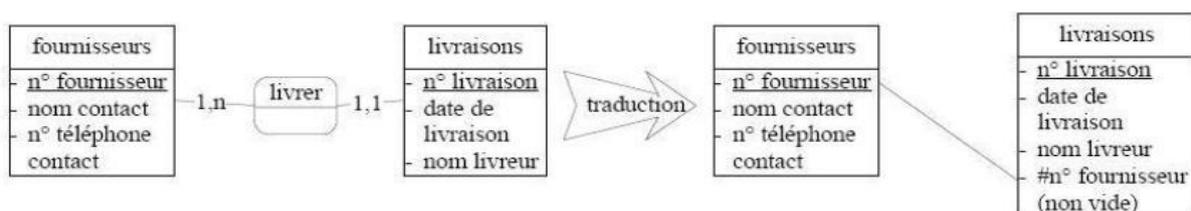
### III-1 Passage Du Modèle Conceptuel De Données (Mcd) Au Modèle Logique De Données (Mld):

Pour traduire un schéma du modèle Entités-Associations vers le Modèle Relationnel, on peut appliquer les règles suivantes :

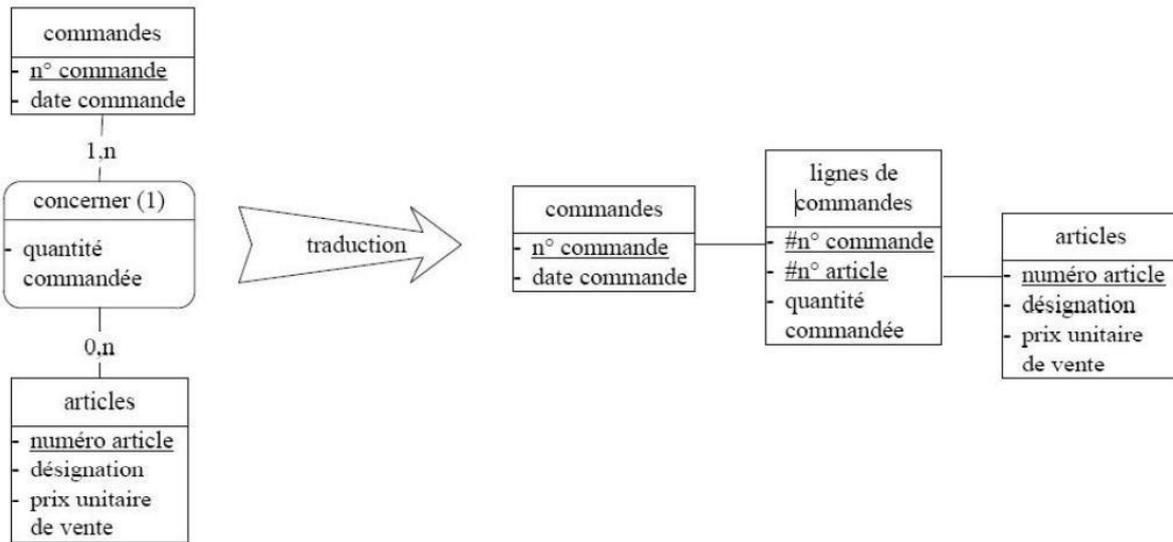
- **Règle 1** : Toute entité devient une table dans laquelle les attributs deviennent les colonnes.

L'identifiant de l'entité constitue alors la clé primaire de la table.

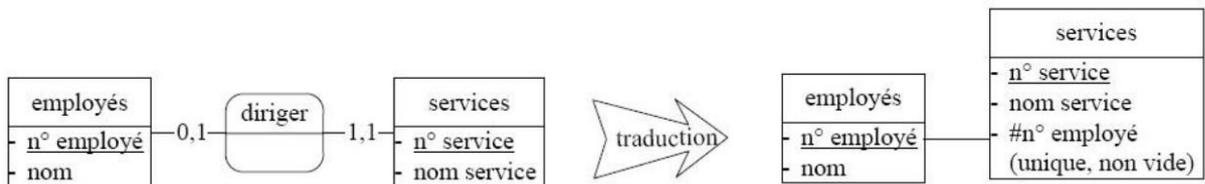
- **Règle 2** : Une association binaire de type 1,n disparaît, au profit d'une clé étrangère dans la table coté 0,1 ou 1,1 qui référence la clé primaire de l'autre table. Cette clé étrangère ne peut pas recevoir la valeur vide si la cardinalité est 1,1



- **Règle 3** : Une association binaire de type **n ,m** devient une table supplémentaire (table de jonction) dont la clé primaire est composée des deux clés étrangères.



- Règle 4 :** Une association binaire de type **1,1** est traduite comme une association binaire de type **1,n** sauf que la clé étrangère se voit imposer une contrainte d'unicité en plus d'une éventuelle contrainte de non vacuité (cette contrainte d'unicité impose à la colonne correspondante de ne prendre que des valeurs distinctes).



- Règle 5 :** Une association non binaire est traduite par une table supplémentaire dont la clé primaire est composée d'autant de clés étrangères que d'entité en association. Les attributs de l'association deviennent les colonnes de cette nouvelle table.

