

# Cours en MERISE MCD & MLD

## I. Présentation de la méthode MERISE

MERISE (Méthode d'Étude et Réalisation Informatique pour les Systèmes de l'Entreprise) est une méthode de conception, de développement des systèmes informatiques, elle vise à recenser la totalité des informations dont l'entreprise a besoin pour assurer tout une partie de ses activités fondamentales, elle est présentée souvent comme une méthode d'analyse informatique.

Elle nous permet de :

- ✚ De maîtriser le développement du système, en répondant au besoin net de l'évolution de l'entreprise.
- ✚ Assurer la continuité du système
- ✚ Faciliter la communication

La méthode MERISE est basée sur la séparation des données et des traitements à effectuer en plusieurs modèles conceptuels et physiques.

**La méthode MERISE a 3 niveaux d'abstraction**

- **Le niveau conceptuel**
- **Le niveau organisationnel**
- **Le niveau physique**

### Le niveau conceptuel :

Il consiste à répondre à la question QUOI ?

Quoi faire, avec quelles données ?

A ce niveau, on ne se préoccupe pas de l'organisation du travail ni du matériel utilisé.

**Les deux modèles sont le Modèle conceptuel des données (MCD) et le Modèle conceptuel des traitements (MCT).**

### Le niveau organisationnel :

Il consiste à répondre à la question QUI ?, OU ?, QUAND ?

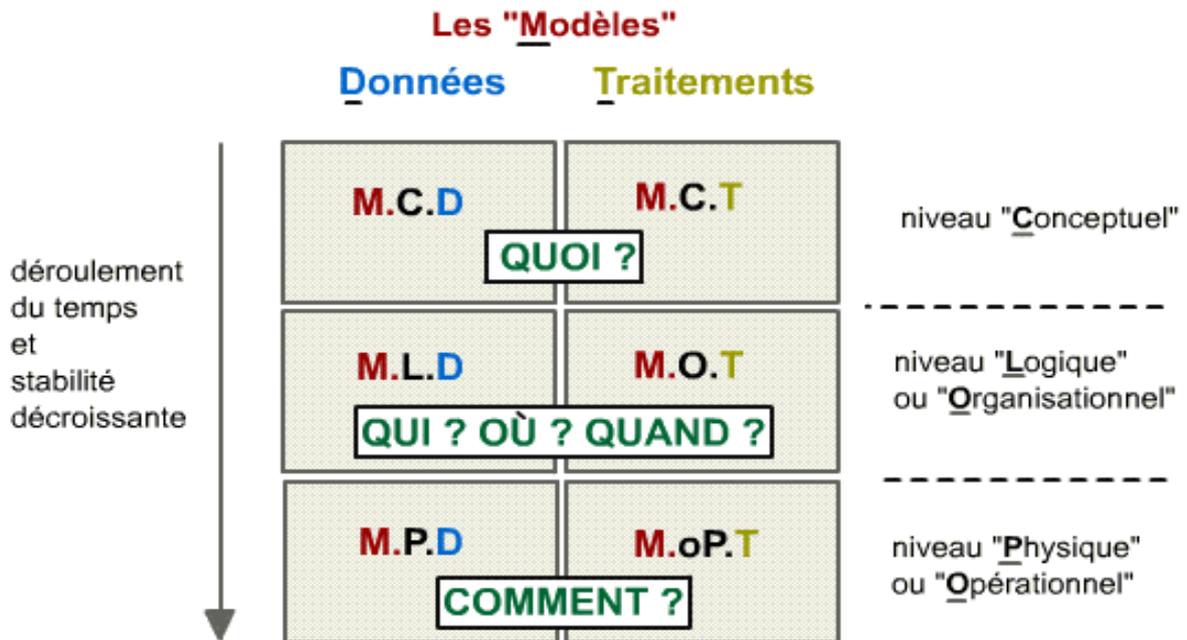
C'est à ce niveau que sont intégrés les critères d'organisation de travail.

On tient compte (ou on propose) des choix d'organisation de travail comme la répartition des traitements entre l'homme et la machine, le mode de fonctionnement (temps réel, temps différé).

**Le modèle de représentation est le modèle organisationnel des traitements.**

Niveau	Données	Traitements	Choix pris en compte
<b>Conceptuel</b>	Modèle Conceptuel des Données (MCD) Signification des informations sans contrainte technique ou économique	Modèle Conceptuel des Traitements (MCD) Activité du domaine sans préciser les ressources ou leur organisation	Choix de gestion Quoi ?
<b>Organisationnel</b>	Modèle Organisationnel des Données (MOD) Signification des informations avec contrainte organisationnelles et économique	Modèle Organisationnel des Traitements (MOT) Fonctionnement du domaine avec les ressources utilisées	Choix d'organisation Qui ?, Ou ?, Quand ?
<b>Physique</b>	Modèle Physique des Données (MPD) Description de la ou des bases de données dans la syntaxe du logiciel SGF ou SGBD	Modèle Physique des Traitements (MPT) Architecture technique des programmes	Choix technique Comment ?

**Dissociation des données et des traitements et l'étude de leurs interactions (Cycle d'abstraction)**



**Exemple :**

Nous allons étudier une partie de la méthode à l'aide d'un exemple.

Le système d'information contient les propriétés figurant sur les factures :

N° Facture : .....				Date : .....	
Nom Client : .....					
Adresse : .....					
Code Produit	Désignation	Quantité	Prix Unitaire	Montant	
.....	.....	.....	.....	.....	
.....	.....	.....	.....	.....	
.....	.....	.....	.....	.....	
				Total : .....	

# Règles de gestion

- 1 : Un client peut obtenir une ou plusieurs factures ou aucune facture.
- 2 : Une facture ne peut être affectée qu'à un seul client.
- 3 : Une facture peut concerner un ou plusieurs produits.
- 4 : Un produit peut être affecté à une ou plusieurs factures ou aucune facture.

## II. Modèle conceptuel des données (MCD)

### IV.1 Définition

Un Modèle Conceptuel de Données est la formalisation de la structure et de la signification des informations décrivant des objets et des associations perçus d'intérêt dans le domaine étudié, en faisant abstraction des solutions et contraintes techniques informatiques d'implantation en base de données.

Le modèle conceptuel de données (MCD) fait référence à tous les objets du système d'information et aux relations entre les objets. Le formalisme utilisé est connu sous le nom de "**Schéma Entité-Relation**". Il se base autour de 3 concepts principaux, les **entités**, les **relations** et les **propriétés**

### IV.2 Entité

Une entité est la représentation d'un élément matériel ou immatériel ayant un rôle dans le système que l'on désire décrire

- ✓ Une entité peut être un acteur : client, usine, produit => pourvue d'une existence intrinsèque.
- ✓ Une entité peut être un flux : commande, livraison => existe par l'intermédiaire d'acteurs.

Une entité est caractérisée par son nom et ses propriétés.

#### Exemple de l'entité CLIENT



Nom_Cli
Prenom_Cli
Adresse_Cli
Ville_Cli
Tele-Cli

### IV.3 Association (ou Relation)

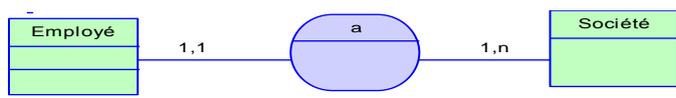
Une relation décrit un lien entre deux ou plusieurs entités. Chaque relation possède un nom, qui est généralement constitué par un verbe à l'infinitif.

Une relation est liée à chacune de ses entités par un lien sur lequel on indique les cardinalités.

Les cardinalités représentent la participation de l'entité concernée à la relation.

Le premier nombre indique la cardinalité minimale, le deuxième la cardinalité maximale.

#### Exemple de relation entre un employeur et une société



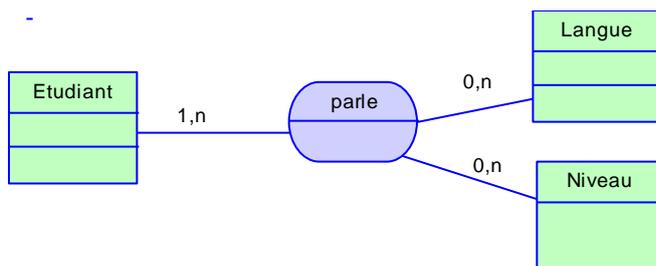
Un employé a une et une seule société. Une société a 1 ou n employés.

#### Exemple de relation entre une commande et un produit



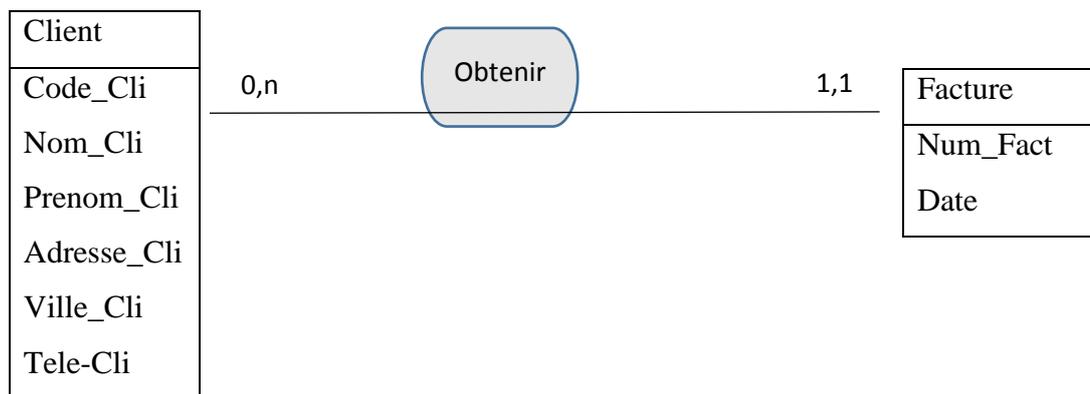
Une commande est composée de 1 ou n produits distincts en certaine quantité. Un produit est présent dans 0 ou n commandes en certaine quantité.

#### Exemple de relation entre un étudiant , une langue et un niveau



Un étudiant parle une ou plusieurs langues avec un niveau. Chaque langue est donc parlée par 0 ou n étudiants avec un niveau. Pour chaque niveau, il y a 0 ou plusieurs étudiants qui parlent une langue.

### Exemple de relation entre un client et une Facture



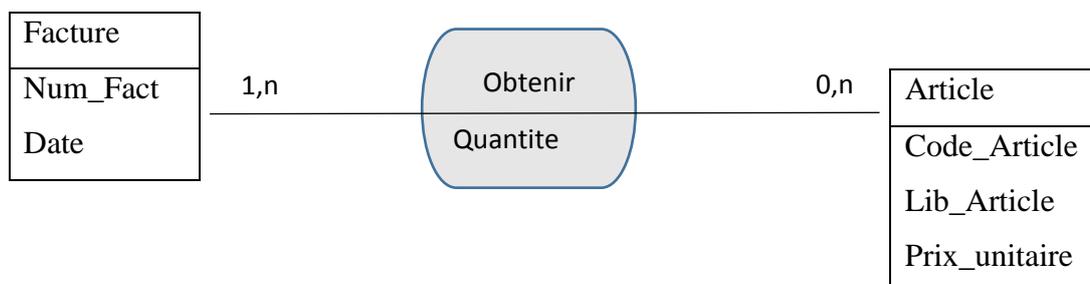
#### Entité « Client » et relation « Obtenir »

- Cardinalité minimale = 0 : chaque client obtient aucune facture.
- Cardinalité maximale = n : chaque client peut obtenir plusieurs (n) factures.

#### Entité « Facture » et relation « Obtenir »

- ✓ Cardinalité minimale = 1 : chaque facture est obtenue par au moins un client.
- ✓ Cardinalité maximale = 1 : chaque facture est obtenue au maximum par un seul client

### Exemple sur la relation « Facture », « Article »



## IV.4 La propriété

Une propriété est une donnée élémentaire qui qualifie l'entité à laquelle elle se rapporte :

- + Chaque propriété prend des valeurs qui sont appelées occurrences de la propriété,
- + Chaque propriété se rattache toujours à une entité.

### Exemple :

Si l'on considère le domaine de gestion des commandes d'une société de vente par correspondance, les données : « **référence article** », « **désignation article** », « **prix unitaire HT** », « **taux de TVA** » sont des propriétés pertinentes pour ce domaine.

Chaque valeur prise par une propriété est appelée **occurrence**. Des occurrences de la rubrique « désignation article » sont par exemple : « cahier », « stylo », « règle », ...

## IV.5 Les identifiants

C'est une propriété (ou ensemble de propriétés) particulière qui permet d'identifier de façon unique une occurrence de l'entité

L'identifiant figure en premier dans la liste des propriétés

Il est souligné

## IV.6 Les Cardinalités

Les cardinalités sont des couples de valeurs que l'on trouve entre chaque entité et ses associations liées

Les cardinalités permettent de caractériser le lien qui existe entre une entité et la relation à laquelle elle est reliée. La cardinalité d'une relation est composé d'un couple comportant une borne maximale et une borne minimale, intervalle dans lequel la cardinalité d'une entité peut prendre sa valeur:

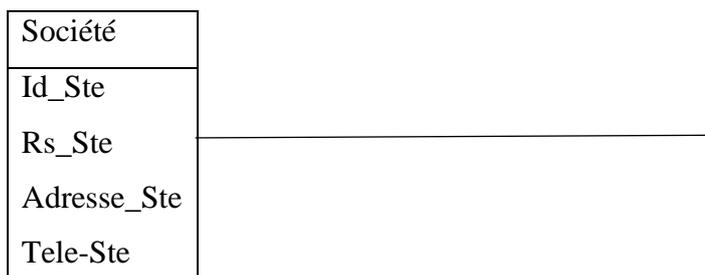
La borne minimale (généralement **0** ou **1**) décrit le nombre minimum de fois qu'une entité peut participer à une relation

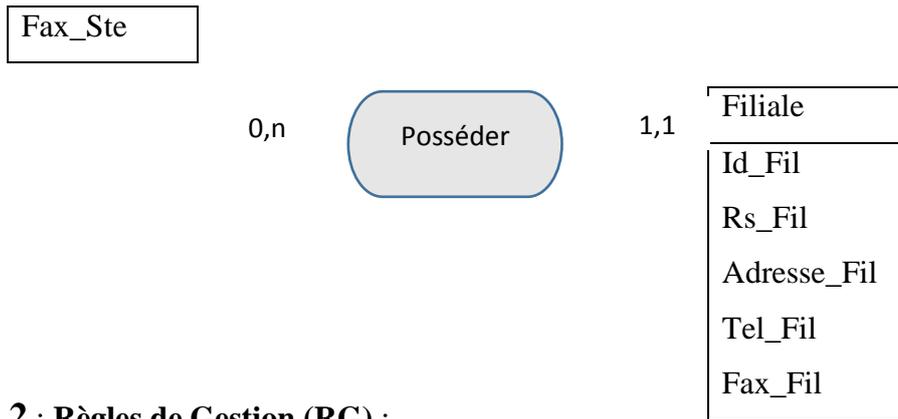
La borne maximale (généralement **1** ou **n**) décrit le nombre maximum de fois qu'une entité peut participer à une relation

### Exemple 1 :

Une société à zéro (cardinalité minimale =0) ou plusieurs filiales (cardinalité maximale = n).

Une filiale appartient à une (cardinalité minimale =1) et une seule société (cardinalité minimale =1)

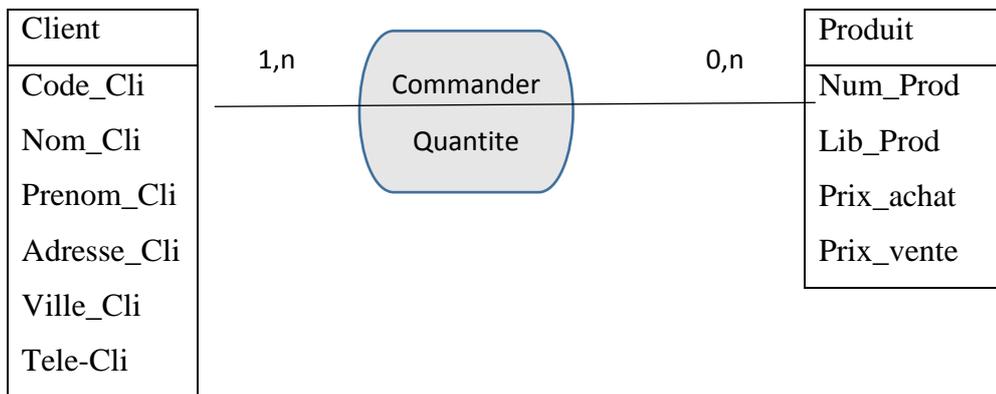




### **Exemple 2 : Règles de Gestion (RG) :**

Un client commande au moins un produit (sous-entendu ou plusieurs).

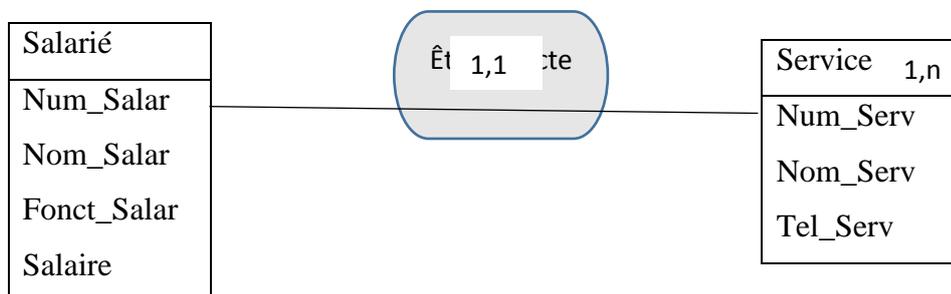
Un produit peut ne pas encore avoir été commandé, comme il peut l'avoir été commandé plusieurs fois.



### **Exemple 3 : Règles de Gestion (RG)**

Un salarié est obligatoirement affecté à 1 et 1 seul service.

Un service pour exister doit avoir au moins un salarié (sous-entendu ou plusieurs).



## **IV.7 Contrainte d'intégrité fonctionnelle**

Une contrainte d'intégrité fonctionnelle entre plusieurs entités exprime que l'un des objets est totalement identifié (**dépendant de**) par une autre entité.

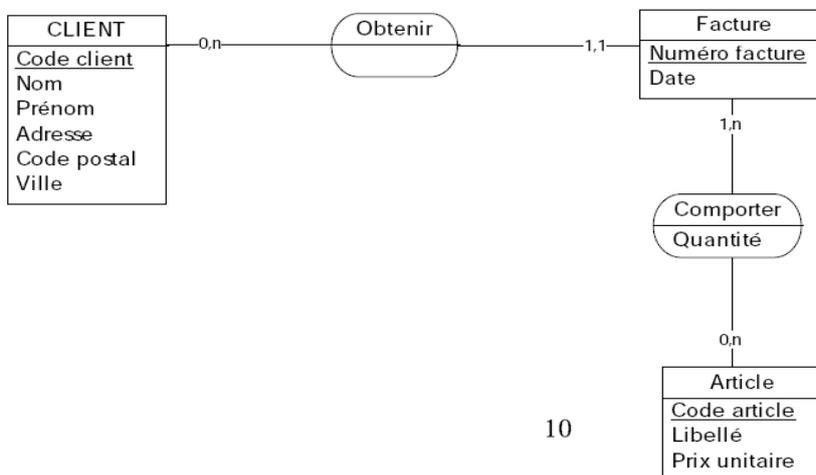
Supprimer le " Père " conduit à supprimer le(s) " Fils " ; on est donc en présence de cardinalité **1,1** du fils vers le père.

Une CIF indique que l'une des entités est totalement déterminée par la connaissance de l'autre.

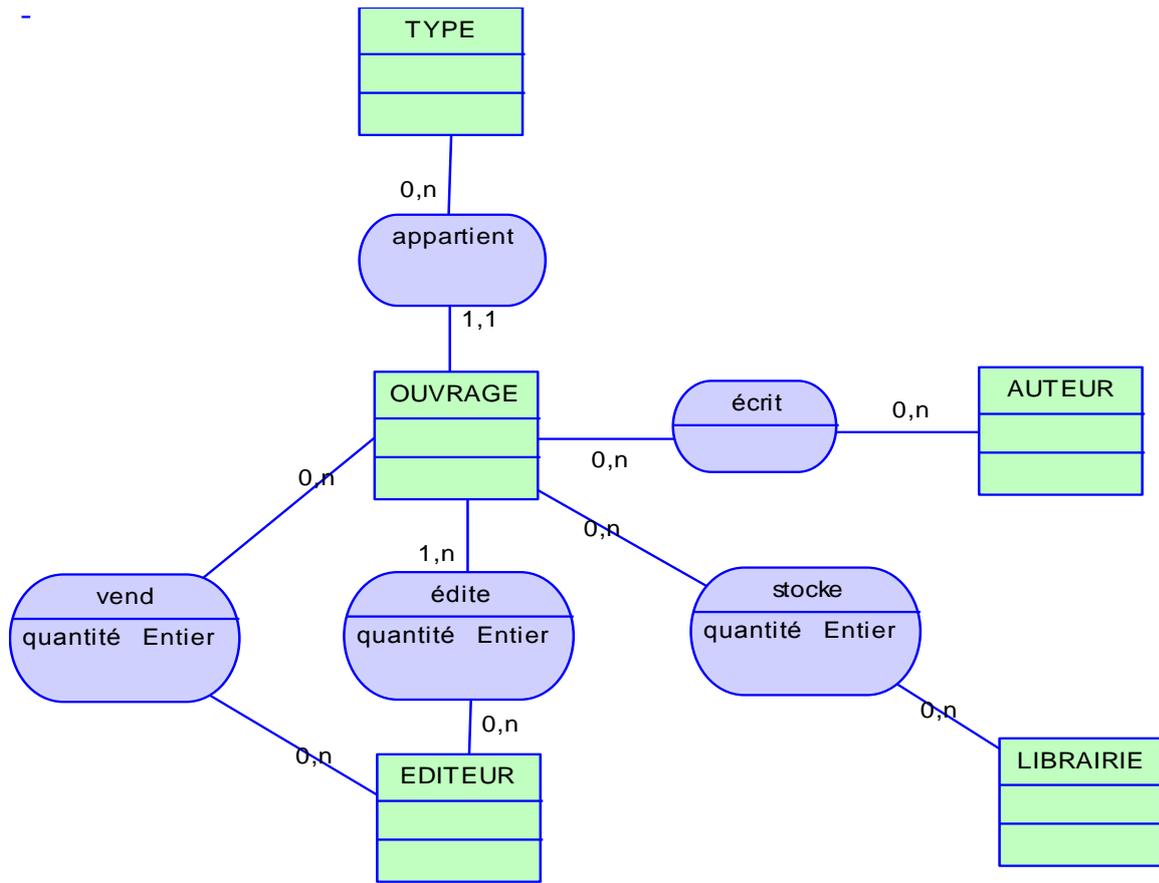
### Exemple:

Connaissant une facture bien précise, on connaît avec certitude le client correspondant.

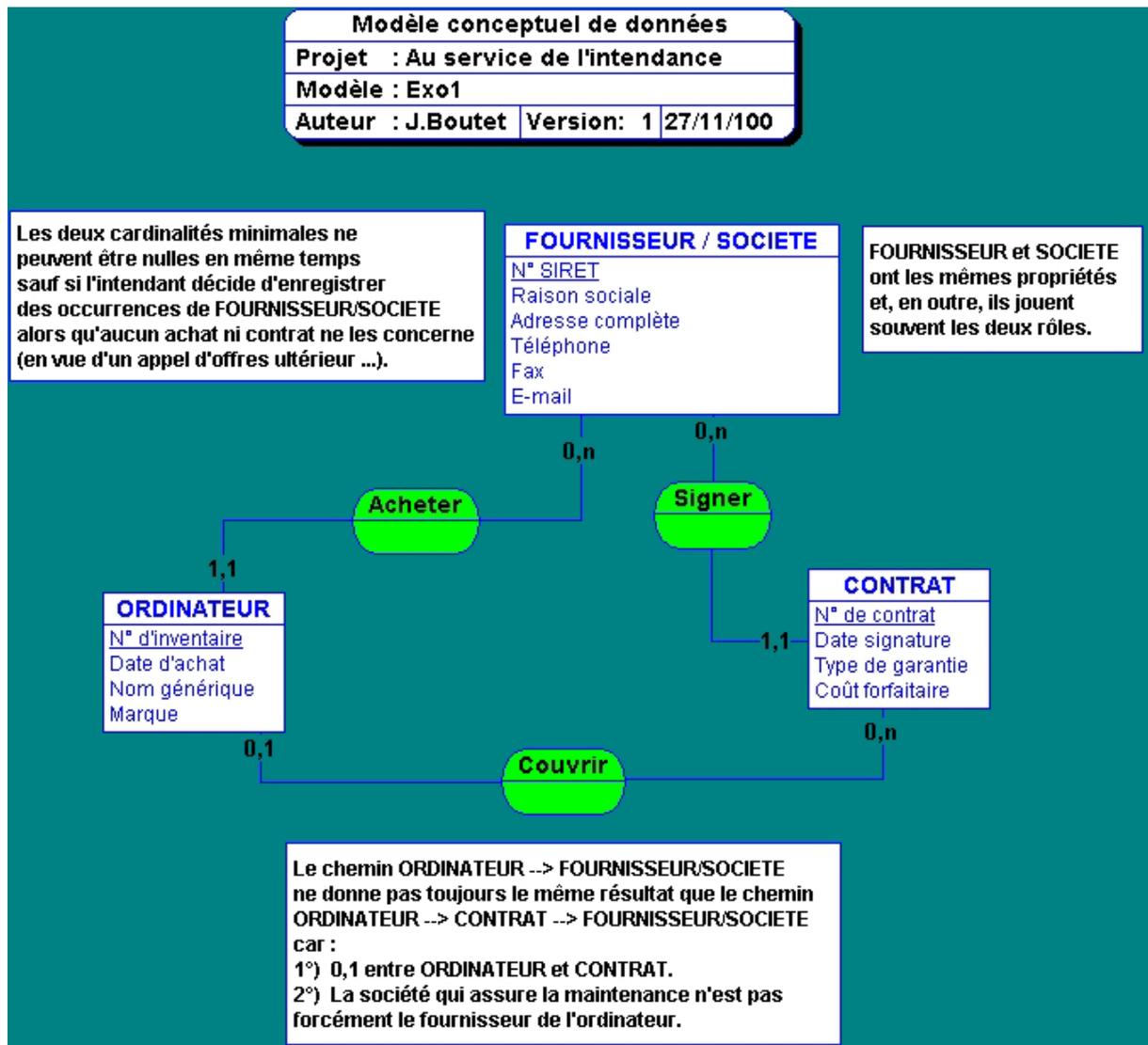
### Exemple1 de Modèle conceptuel de données (MCD)



### Exemple2 : de Modèle conceptuel de données (MCD)



**Exemple 3 : de Modèle conceptuel de données (MCD)**



### Cas Pratique MCD avec des Explications détaillées

Une entreprise **X** vend des véhicules toutes marques qu'elle stocke dans de grands entrepôts. Dans un même entrepôt, nous pouvons trouver plusieurs marques de véhicules, cependant, pour des raisons de logistiques, le gérant de la société **X** a exigé de ses employés qu'une marque ne puisse se trouver que dans un seul entrepôt. Chaque attaché commercial gère son propre portefeuille de clients. L'entreprise **X** souhaite établir des statistiques commerciales sur ses ventes de véhicules : nombre de véhicules achetés par un client, chiffre d'affaire réalisé par une marque, mais aussi sur les marques entreposées dans un entrepôt.

### Démarche de Construction de MCD

**Objet :**

Vente de véhicules toute marque

**Application :**

Statistiques commerciales

**Résultats attendus :**

- ✓ Nombre de véhicules achetés par un client ?
- ✓ Chiffre d'affaire réalisé par une marque ?
- ✓ Quelles sont les marques entreposées dans un entrepôt ?

**Données :**

- ✓ Nom de marque
- ✓ Nom de dépôt
- ✓ Nom du type
- ✓ Puissance fiscale
- ✓ Nom du responsable commercial pour une marque
- ✓ Prix unitaire d'un type de véhicule
- ✓ Adresse de dépôt
- ✓ Nom, adresse du client
- ✓ Quantité d'une vente
- ✓ Date d'une vente
- ✓ Nom de l'attaché commercial
- ✓ Adresse de l'attaché commercial

**Contraintes sur les données :**

- ✓ Un dépôt peut-être multi-marques,
- ✓ Une marque ne se trouve que dans un seul entrepôt,
- ✓ Un attaché gère plusieurs clients,
- ✓ Un client est géré par un seul attaché

**Repérer les entités :**

Les entités sont les objets de gestion essentiels du système d'information.

L'entité est un ensemble dont chaque élément est un élément particulier.

Dans notre exemple, que gère-t-on ?

En première lecture, 3 entités ressortent :

1. Les types de véhicule,
2. Les clients,
3. Les dépôts.

**Attribuer à chaque entité son identifiant et ses propriétés :**

Pour l'entité TYPE DE VEHICULE, nous avons défini une propriété Nom de marque.

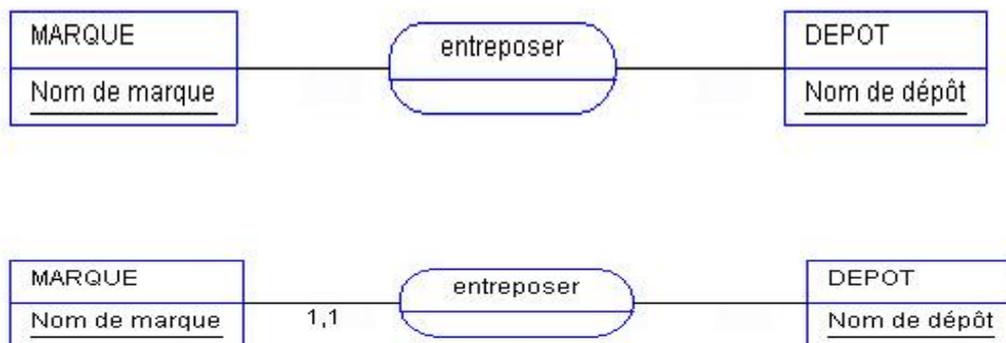
Cependant, le responsable commercial est dépendant de la marque. MARQUE est donc une entité cachée.

**Il convient donc de modéliser ainsi :**

<b>Entité :</b>	<b>MARQUE</b>	<b>TYPE DE VEHICULE</b>
<b>Identifiant :</b>	<i><u>Nom de marque</u></i>	<i><u>Nom du type</u></i>
<b>Propriétés :</b>	<i>Responsable commercial</i>	<i>Prix</i> <i>Puissance fiscale</i>

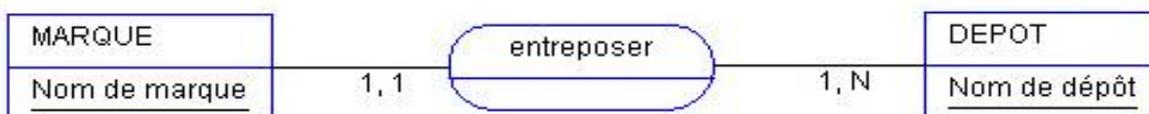
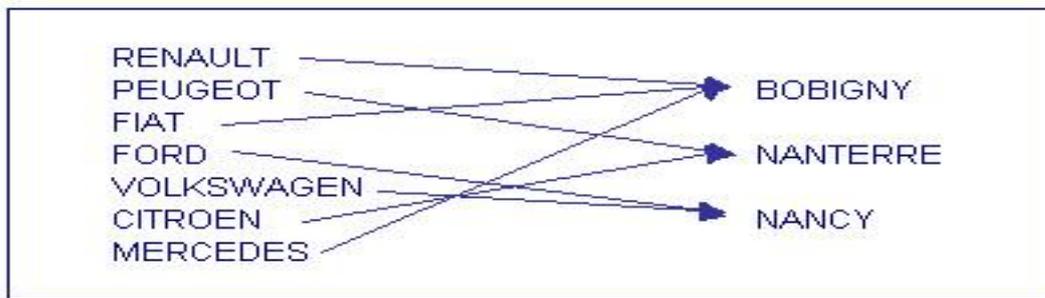
**Définition des relations entre les Entités et cardinalité :**

Relation entre MARQUE et DEPOT :



**Définition des relations entre les Entités et cardinalité :**

Relation entre **MARQUE** et **DEPOT** :



**(1,1)** : Une marque est entreposée dans un seul entrepôt.

**(1, N)** : Dans un entrepôt sont entreposées une ou plusieurs marques.

**Dépendances fonctionnelles entre Entités (DF) :**

La relation qui lie un TYPE DE VEHICULE à une MARQUE est « appartenir ». Il s’agit d’une relation hiérarchique : à un type de véhicule donné ne correspond qu’une seule marque, et à une marque correspond plusieurs type de véhicule. Ici, TYPE DE VEHICULE détermine totalement MARQUE.

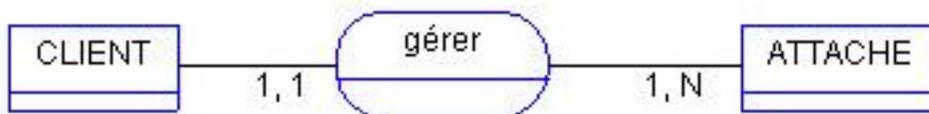
Nous appelons cette relation fonctionnelle (**DF**), et elle se note de la manière suivante :

TYPE DE VEHICULE -> MARQUE

Qu’en est-il de ATTACHE COMMERCIAL et de CLIENT ?

**CLIENT -> ATTACHE COMMERCIAL**

(A un client ne correspond qu’un attaché commercial)

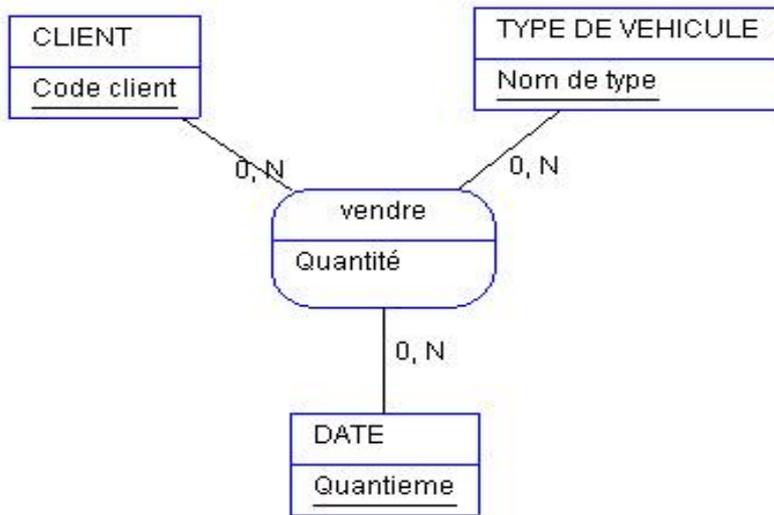


**Relations entre 3 Entités :**

Regardons à présent le problème des quantités élémentaires de ventes.

Cette donnée est une propriété de la relation « vendre », liant CLIENT et TYPE DE VEHICULE : la quantité ne prend de sens que si l’on connaît conjointement le client et le type.

Cependant, pour qu'il n'y ait qu'une seule occurrence de quantité, il est nécessaire d'introduire un « chronomètre » dans notre système d'information :



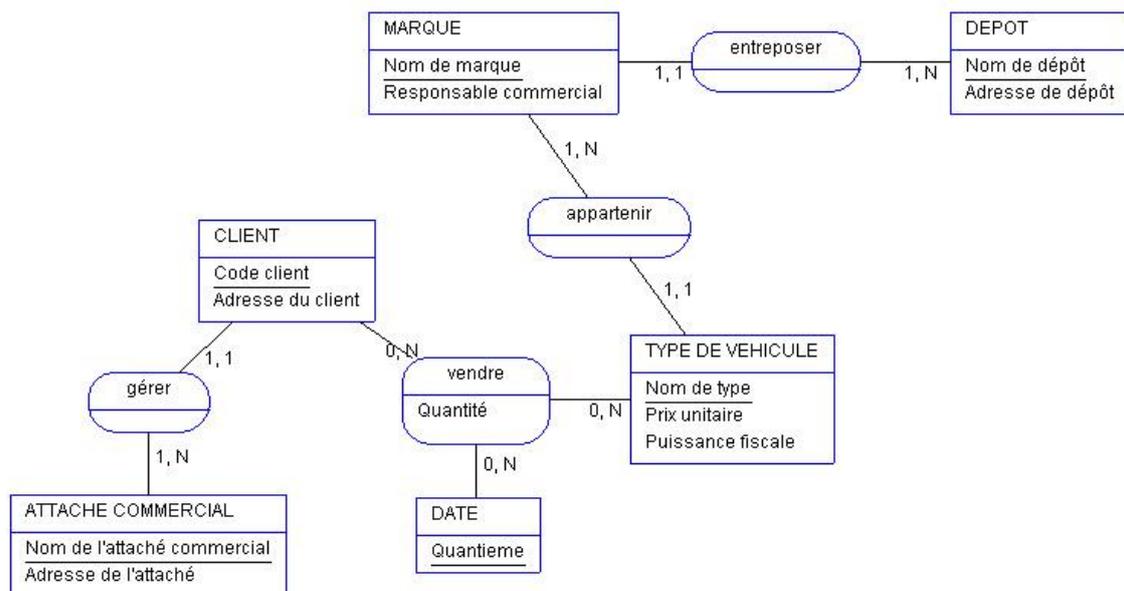
Les cardinalités se lisent alors :

Pour un **type de véhicule**, il peut y avoir de 0 à N relations « vendre », autrement dit il peut y avoir plusieurs ventes pour un même type de véhicule,

Pour un **client**, il peut y avoir 0 à N relations « vendre », en d'autres termes, il peut y avoir plusieurs ventes pour un même client.

A une **date**, il peut y avoir 0 à N relations « vendre ». Ce qui se traduit par : certains jours, il y a plusieurs ventes.

Finalement on obtient le MCD suivant :



### III. Modèle logique de données (MLD)

#### V.1 Définition

Le modèle logique de données (MLD) se base sur un modèle conceptuel des données. Il est composé de tables logiques reliées entre elles par des flèches.

#### V.2 Passage du MCD au MLD

Règles de passage du MCD au MLD :

##### V.2.1 Entité

L'entité est transformée en table. Les propriétés de l'entité deviennent les attributs de la table.

##### V.2.2 Identifiant

L'identifiant de l'entité devient la clé primaire de la table.

La clé primaire permet d'identifier de façon unique un enregistrement dans la table.

Les valeurs de la clé primaire sont donc uniques.

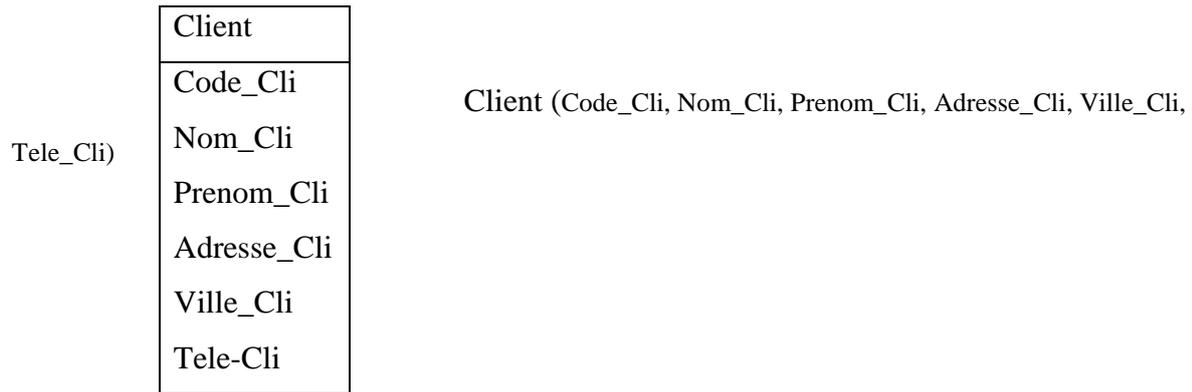
Les valeurs de la clé primaire sont obligatoirement non nulles.

Une table contiendra donc un ensemble d'enregistrements.

Une ligne correspond à un enregistrement.

Une colonne correspond à un champ.

Exemple :

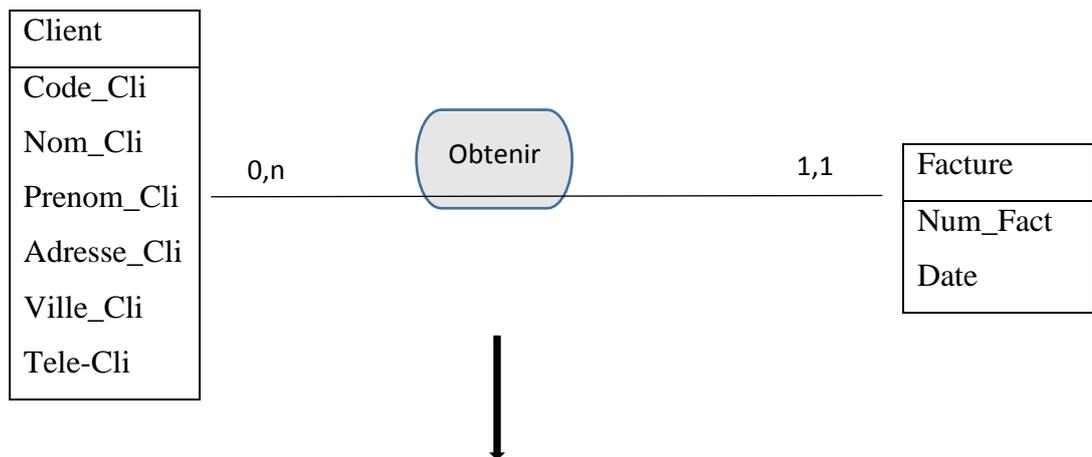


### V.2.3 Relation avec cardinalités (x,n) et (x,1) ou x = 0 ou 1

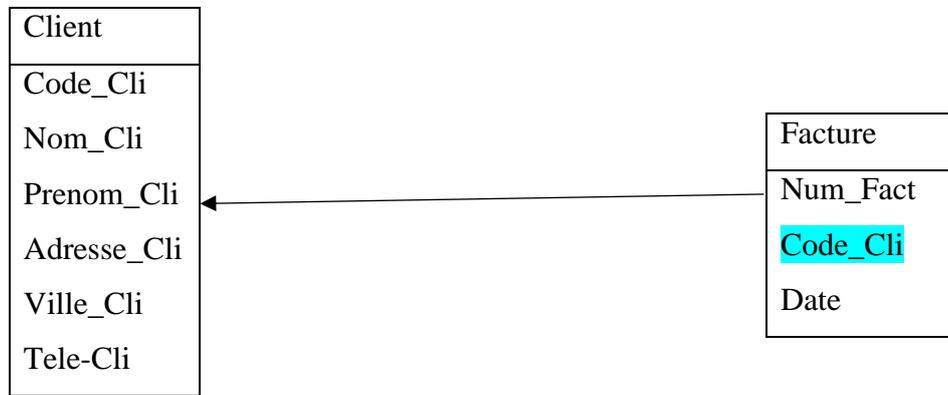
On copie la clé primaire de la table dont la cardinalité est (x,n) dans la table dont la cardinalité est (x,1). L'attribut est appelé clé étrangère. Les deux tables sont liées par une flèche qui pointe de la table à clé étrangère vers la table qui contient la clé primaire correspondante.

#### Exemple sur la relation « Client », « Facture

MCD



MLD



Client (Code\_Cli, Nom\_Cli, Prenom\_Cli, Adresse\_Cli, Ville\_Cli, Tele\_Cli)

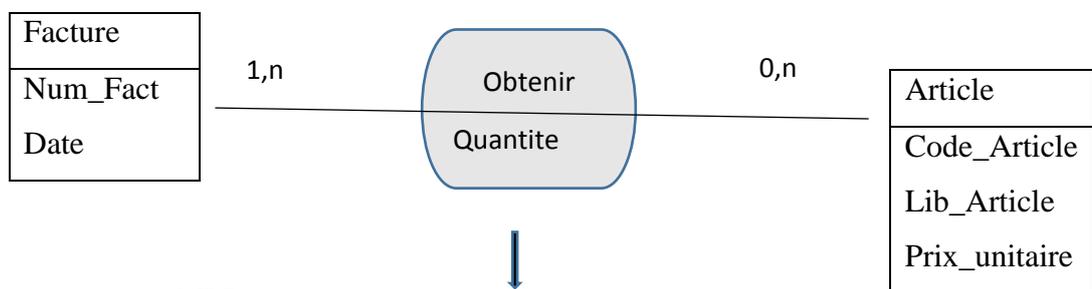
Facture (Num\_Fact, # Code\_Cli, Date)

## V.2.4 Relation avec cardinalités (x,n) et (x,n) ou x = 0 ou 1

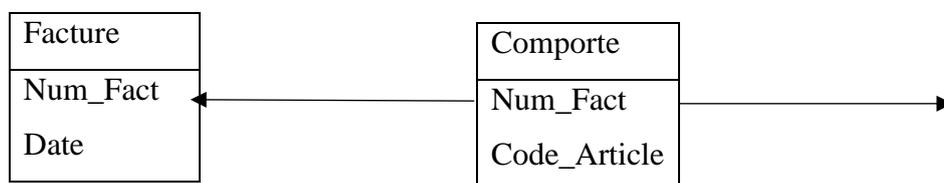
Il y a création d'une table supplémentaire ayant comme clé primaire une clé composée des identifiants des 2 entités. Lorsque la relation contient des propriétés, celles-ci deviennent attributs de la table supplémentaire.

### Exemple sur la relation « Facture », « Article »

MLD



MLD



Quantite

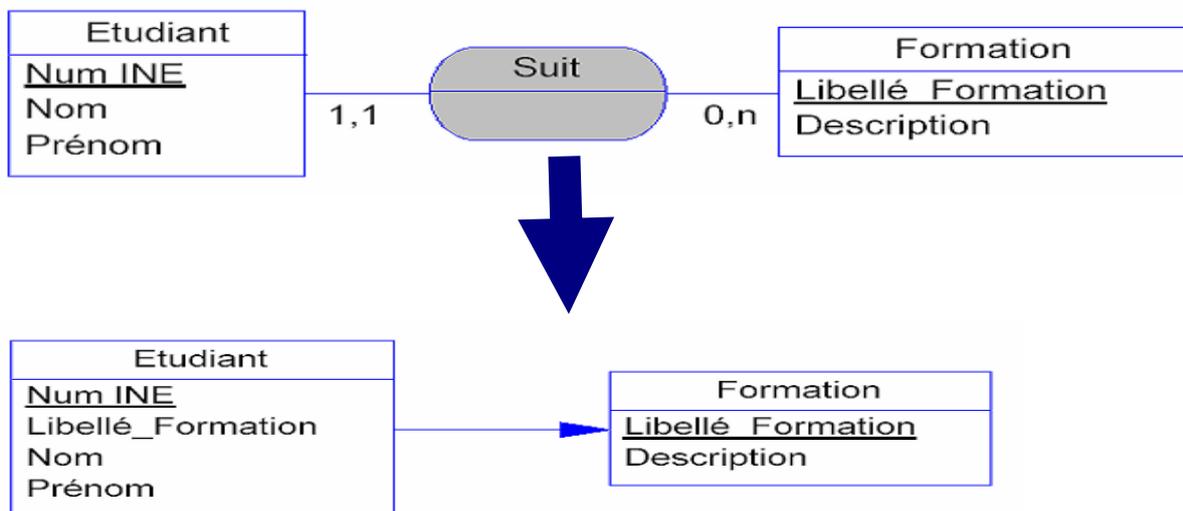
Article
Code_Article
Lib_Article
Prix_unitaire

Facture (Num\_Fact , Date)

Comporte (#Num\_Fact, #Code\_Article, Quantite)

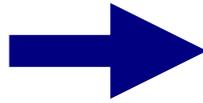
Article(Code\_Article, Lib\_Article, Prix\_Unitaire

### Cas d'un étudiant qui fait une formation

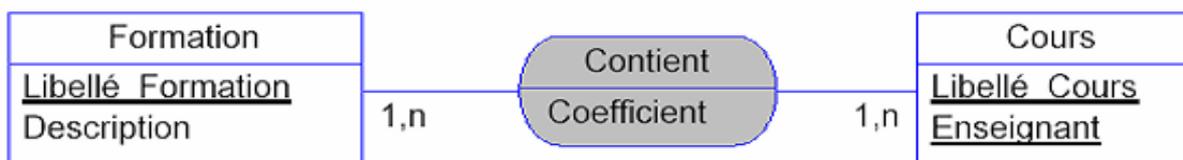


### Cas d'une formation qui se fait dans un département

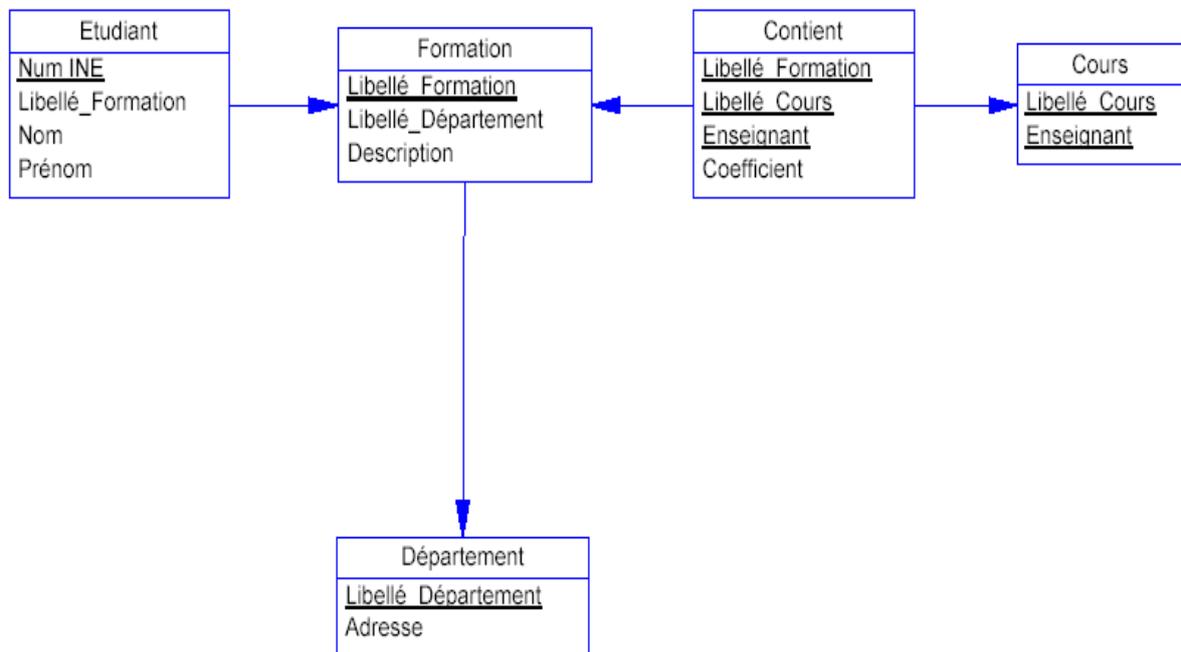




**Cas des cours qui se font dans une formation**

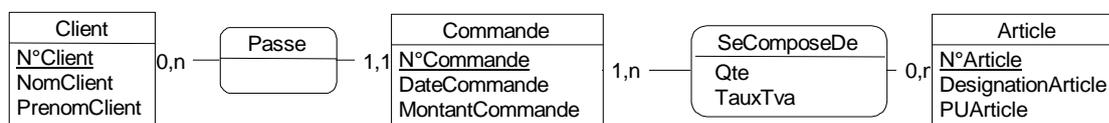


## Exemple récapitulatif N°1



## Exemple récapitulatif N°2

### MCD



## MLD

