

# **Module: Manipulation des bases de données**

Le modèle logique de données  
(MLD)

# Le Modèle Logique de Données Relationnel ( MLDR )

Ce modèle permet de constituer une *base de données* au sens *logique* au moyen de *tables*

## Les Concepts du MLDR

- 1 ) **L'attribut** : C'est le plus petit élément d'information enregistré dans une base de données .  
Il possède un nom et prend des valeurs dans un domaine de valeurs bien déterminé .  
Exemples :

Attribut	Domaine de valeurs
N° Client	Entier naturel
Adresse Client	Alphanumérique
Mode de paiement	Liste alphabétique (Espèces, Chèque , Traite)

- 2 ) **La TABLE** : est un ensemble d'attributs significativement associés ( dont l'association a un sens au niveau du S.I ) .

Représentation d'une relation :     **T ( A1, A2 , A3, ....., An )**     Représentation en intention  
ou Schéma de la TABLE

T	A1	A2	A3	.....	An
tuple 1	.....	.....	.....	.....	.....
tuple 2	valeur	valeur	valeur	.....	Valeur
.....	.....	.....	.....	.....	.....
tuple n	.....	.....	.....	.....	.....

**Représentation en extension**  
( montrant les tuples de la TABLE )

T : Nom de la TABLE

A1, A2 , ....., An : Attributs de la relation

# Le Modèle Logique de Données Relationnel ( suite 1 )

Les clés d'une TABLE: soient 3 TABLES comportant certains attributs communs :

**R1 ( A1 , A2 , A3 , ....., An )**

**R2 ( B1 , B2 , B3 , ....., Bn , A1# )**

**R3 ( A1# , B1# , C1 , C2 , C3 , ....., Cn )**

Les attributs suivants jouent un rôle particulier :

- **A1** dans R1 et **B1** dans R2 sont appelés *clés primaires* : Chacun de ces attributs a été choisi pour identifier de manière discriminante les tuples de sa TABLE .

# Le Modèle Logique de Données Relationnel ( suite 2 )

- **A1# dans R2** est une **clé étrangère** : c'est un attribut défini sur un domaine primaire ( celui de R1 ) mais qui est présent dans une autre relation ( R2 ) dans le but de créer un lien entre les relations R1 et R2 .
- **A1# et B1# dans R3** représentent une **clé primaire composée** :  
C'est un groupe d'attributs définis chacun sur un domaine primaire . Les occurrences de ce groupe ( couples de valeurs de A1# et B1# ) sont utilisées pour identifier de manière discriminante les tuples de la relation R3 .
- **Remarques:** \* une clé primaire ( simple ou composée ) est toujours soulignée dans une relation .  
\* une clé étrangère ( ou externe ) peut être composée comme dans le cas d'une clé primaire  
\* l'attribut ou les attributs constituant une clé étrangère possèdent un nom qui se termine par le symbole #  
\* une relation est toujours identifiée par une clé primaire

)

**4 ) Schéma relationnel :** C'est un ensemble de relations logiques présentant des liens sémantiques . Cet ensemble est destiné à la création d'une base de données physique .

# Le Modèle Logique de Données Relationnel ( suite 3 )

## 5 ) Les Contraintes d'Intégrité :

Elles représentent un ensemble de règles fondamentales dont l'application permet de garantir la cohérence du schéma relationnel d'une base de données .

Ces règles contrôlent la cohérence des valeurs prises par :

- \* **les attributs par rapport à leur domaine de valeurs (contrainte d'intégrité de domaine)**

Exemple : Si l'attribut ' N° Client ' est défini sur un domaine de valeurs numériques , il ne peut pas contenir de lettres .

- \* **les clés primaires des relations ( contraintes d'intégrité de relations )**

L'intégrité de relation concerne les valeurs d'une clé primaire qui doivent être uniques ( pas de doublons ) et non nulles ( toujours spécifiées ) .

- \* **les clés étrangères des relations ( contraintes d'intégrité référentielles )**

L'intégrité référentielle stipule qu'une clé étrangère ne peut prendre que les valeurs définies dans le domaine primaire de la clé primaire à laquelle elle est associée .

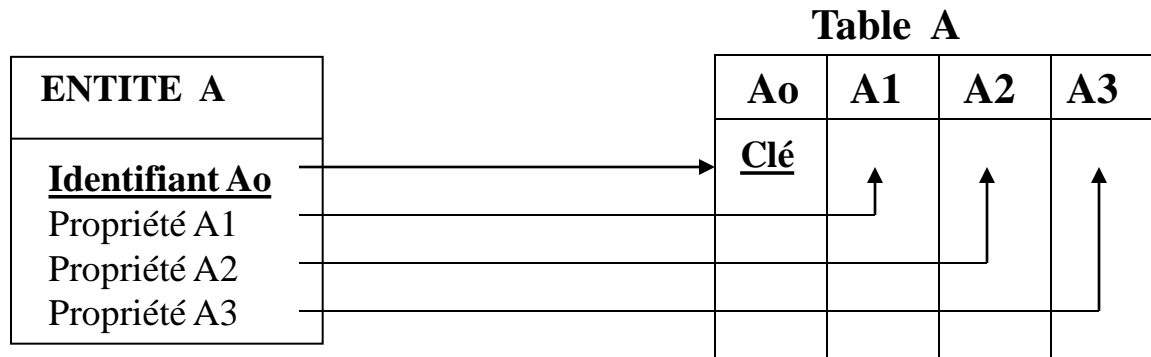
# Construction du Modèle Logique de Données Relationnel

Le **MLDR** est construit à partir du **MCD** en appliquant des *règles de transformation* simples aux entités et aux associations .

Les entités donnent toujours lieu à des relations dans le MLDR .

Les associations , selon leur cardinalités , peuvent ou non donner lieu à des relations .

## 1 ) Transformation des Entités



Une *entité* A du MCD devient la table ) : A ( A0 , A1 , A2 , A3 )

*L'identifiant* A0 de l'entité A devient la *clé primaire* A0 de la Table A .

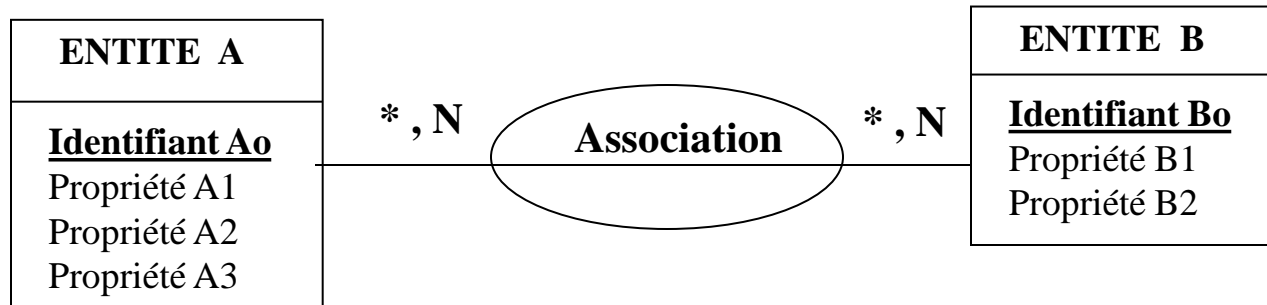
Les *autres propriétés* deviennent les *attributs* de la Table A .

Les *occurrences* de l'entité deviennent les *tuples* de la Table A .

# Construction du Modèle Logique de Données Relationnel ( suite 1 )

## 2 ) Transformation des Associations

### 2.1 ) Association multivaluée plusieurs [ 0, N ou 1, N ] à plusieurs [ 0, N ou 1, N ]



TABLES obtenues :      A , B et C

A ( Ao , A1 , A2 , A3 )

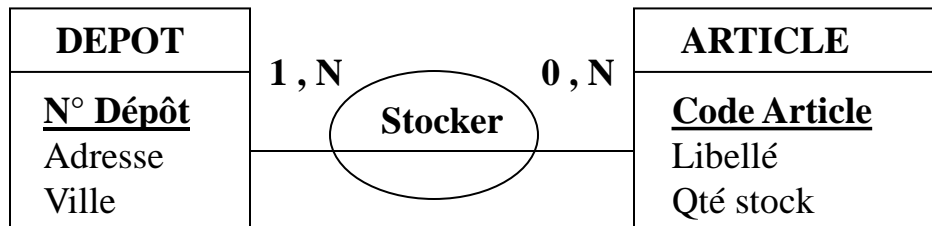
B ( Bo , B1 , B2 )

Cas d'une association non porteuse : C ( Ao# , Bo# )

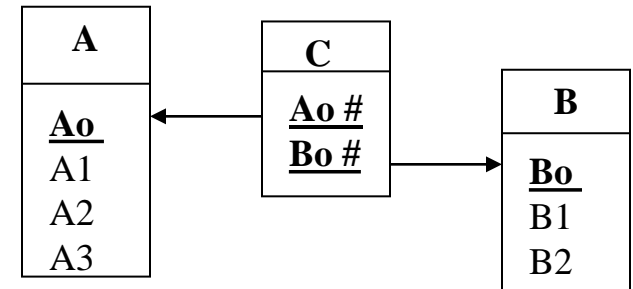
Cas d'une association porteuse des propriétés : C1, C2,...

C ( Ao# , Bo# , C1 , C2 , ... )

Exemple :



Représentation graphique du MLDR



TABLES obtenues :

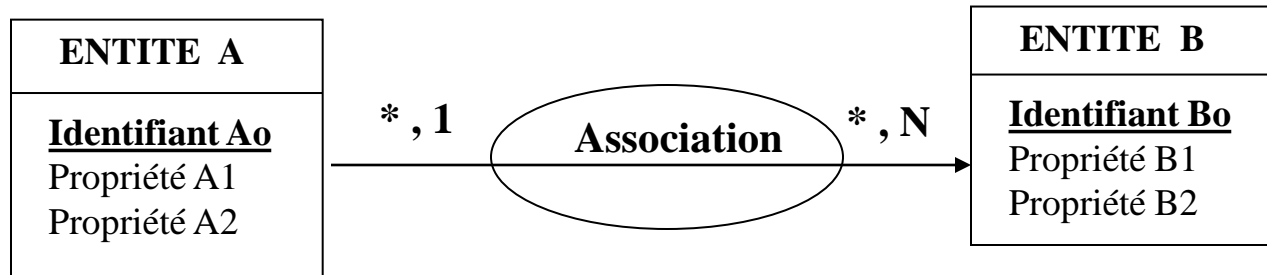
DEPOT ( N° Dépôt , Adresse, Ville )

ARTICLE ( Code Article , Libellé , Qté Stock )

STOCKER ( N° Dépôt # , Code Article # )

# Construction du Modèle Logique de Données Relationnel ( suite 2 )

## 2.2 ) Association hiérarchique Un [ 0, 1 ou 1, 1 ] à Plusieurs [ 0, N ou 1, N ]



Cette association traduit la dépendance fonctionnelle inter-entités : **A0**  $\longrightarrow$  **B0**

L'entité **A** qui émet la dépendance fonctionnelle reçoit au niveau logique l'identifiant de l'autre entité **B**.

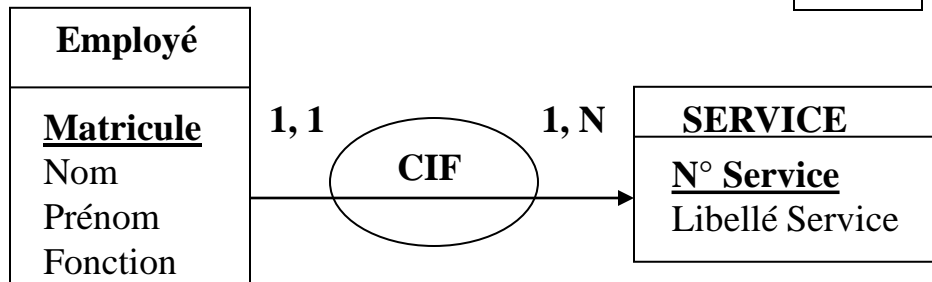
La **clé primaire B0** migre dans la relation **A** comme attribut **clé étrangère ou externe**.

TABLES obtenues : **A , B**

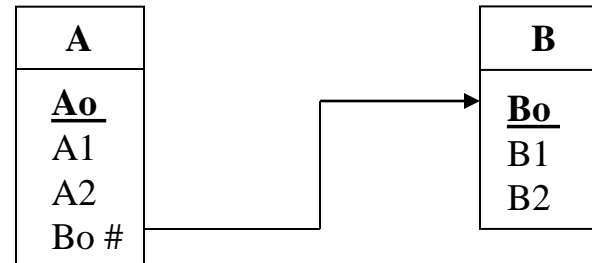
**A** ( A0 , A1 , A2 , Bo# ... )

**B** ( B0 , B1 , B2 , ... )

Exemple :



Représentation graphique du MLDR



TABLES OBTENUES:

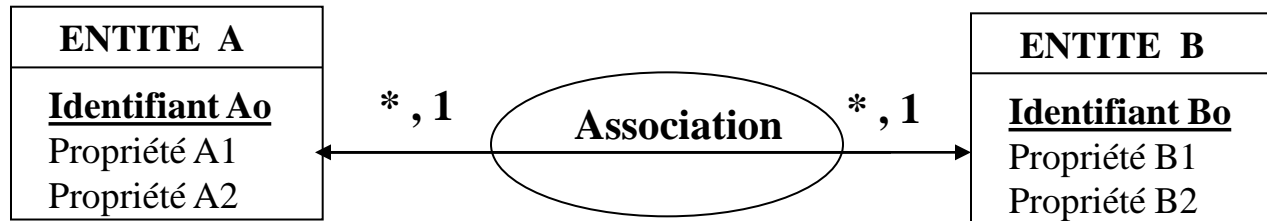
**EMPLOYE** ( Matr. , Nom , Prénom ,  
Fonction , N° Service # )

**SERVICE** ( N° Service , Libellé Service )



# Construction du Modèle Logique de Données Relationnel ( suite 3 )

## 2.3 ) Association hiérarchique Un [ 0, 1 ou 1, 1 ] à Un [ 0, 1 ou 1, 1 ]



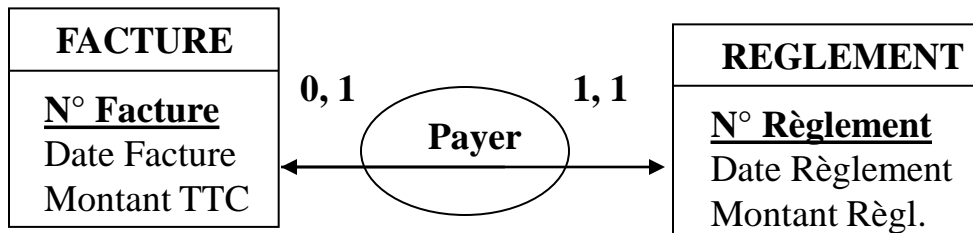
Cette association traduit l'existence de 2 dépendances fonctionnelles inter-entités :  $Ao \longrightarrow Bo$  et  $Bo \longrightarrow Ao$   
La migration de clé peut se faire dans un sens ou l'autre selon les besoins du système d'information .  
Si la cardinalité d'un côté de l'association est **1, 1** ( exemple côté Entité A ), il est conseillé de choisir la migration de la clé primaire **Bo** dans la relation **A** comme **clé étrangère** ( règle équivalente au cas d'une **association hiérarchique** )

TABLES obtenues :      **A , B**

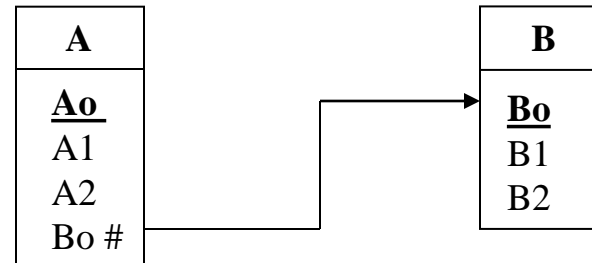
**A ( Ao , A1 , A2 , Bo# ... )**

**B ( Bo , B1 , B2 , ... )**

Exemple :



Représentation graphique du MLDR



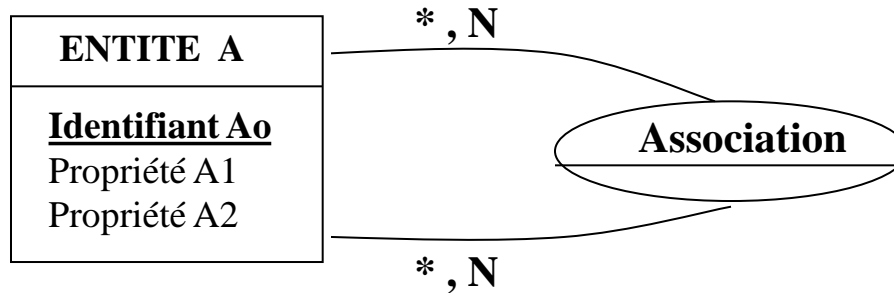
TABLES obtenues :

**FACTURE ( N° Facture , Date Facture ,  
Montant TTC )**

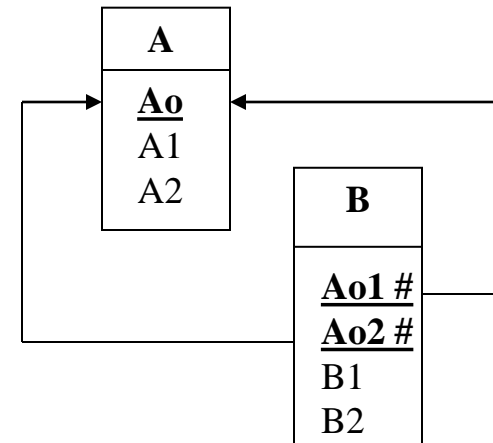
**REGLEMENT ( N° Règl. , Date Règl. ,  
Montant Règl. , N° Facture # )**

# Construction du Modèle Logique de Données Relationnel ( suite 4 )

## 2.4 ) Association réflexive multivaluée Plusieurs [ 0, N ou 1, N ] à Plusieurs [ 0, N ou 1, N ]



Représentation graphique  
du MLDR



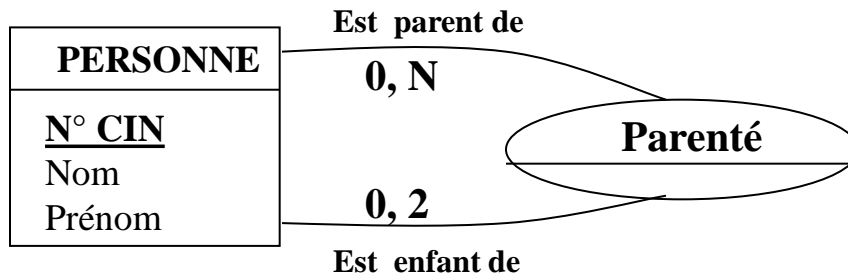
TABLES obtenues :      A , B

A ( Ao , A1 , A2 , ... )

B ( Ao1# , Ao2# ) : Cas d'une assoc. non porteuse

B ( Ao1# , Ao2# , B1 , B2 , ... ) : Cas d'une assoc. porteuse

Exemple :



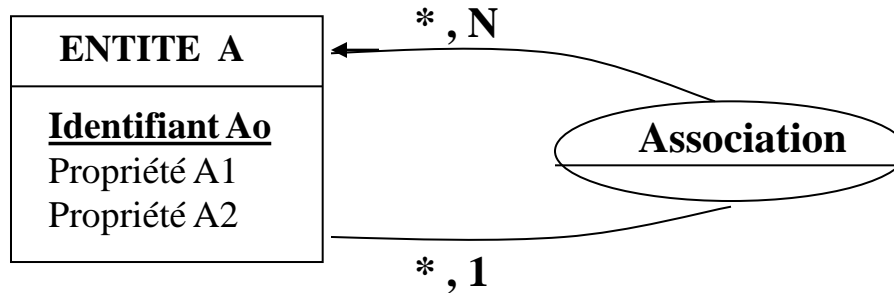
TABLES obtenues :

PERSONNE ( N° CIN , Nom , Prénom )

PARENTE ( N° CIN\_Parent # , N° CIN\_Enfant # )

# Construction du Modèle Logique de Données Relationnel ( suite 5 )

## 2.5 ) Association réflexive hiérarchique Un [ 0, 1 ou 1, 1 ] à Plusieurs [ 0, N ou 1, N ]



Représentation graphique  
du MLDR

A
<u>Ao</u>
A1
A2
Ao' #

TABLE obtenue :      A

A ( Ao , A1 , A2 , ... , Ao' # )

### Exemple :

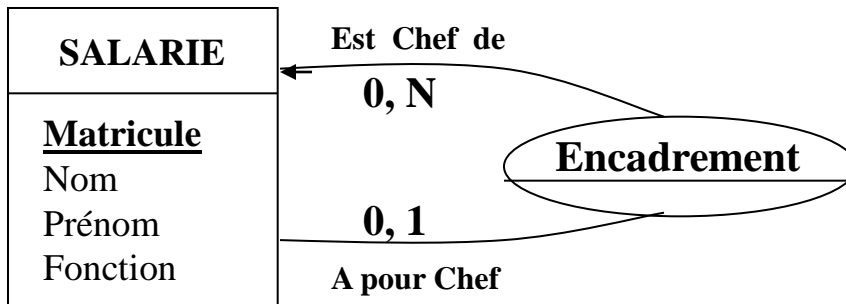


TABLE obtenue :

SALARIE ( Matricule , Nom , Prénom ,  
Fonction , Matricule\_Chef # )