

# Le Modèle Logique de Données

Maintenant que le MCD est établi, on peut le traduire pour différents systèmes logiques et notamment les bases de données relationnelles qui proposent une vision plus concrète pour modéliser la situation.

## 1. Systèmes logiques

Aux tout début de l'informatique, les données étaient stockées dans des fichiers binaires et gérées par des programmes exécutables. Mais la maintenance de ceux-ci (en cas de modification de la structure des données, notamment) était très problématique.

Dans les années 1960 les systèmes de gestion de base de données relationnels (SGBDR ou RDBMS pour Relational Data Base Management System) sont apparus. Dans ces systèmes, les données sont stockées dans des tables et l'information peut être obtenue par une requête formulée dans un langage quasiment naturel (le SQL pour Structured Query Language). Parmi les SGBDR les plus répandus nous trouvons Oracle, MySQL, SQL Server et DB2. Nous exposons ici le modèle logique de données relationnel dans un contexte proche de MySQL.

## 2. Tables, lignes et colonnes

Lorsque des données ont la même structure (comme par exemple, les renseignements relatifs aux clients), on peut les organiser en **table** dans laquelle les **colonnes** décrivent les champs en commun et les lignes contiennent les valeurs de ces champs pour chaque enregistrement :

number	firstname	lastname	country
0012	Joe	Dalton	USA
0013	Jack	Dalton	USA
0014	William	Dalton	USA
0015	Averell	Dalton	USA
7656	Daisy	May	Norway
1342	Lucky	Luke	Belgium

Tab. 3 – Contenu de la table clients, avec en première ligne les intitulés des colonnes

**Convention** : Le nom d'une table est au pluriel

## Clés primaires et clés étrangères

Les lignes d'une table doivent être uniques : il ne peut pas y avoir deux lignes contenant exactement les mêmes valeurs :

number	firstname	lastname	country
0012	Joe	Dalton	USA
0013	Jack	Dalton	USA
0014	William	Dalton	USA
0015	Averell	Dalton	USA
1342	Lucky	Luke	Belgium
7656	Daisy	May	Norway
7656	Daisy	May	Norway

Cela signifie qu'une colonne (au moins) doit servir à les identifier. Il s'agit de la **clé primaire** de la table.

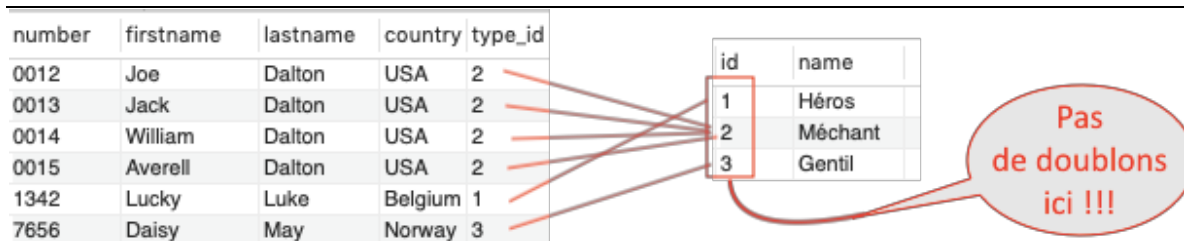
L'absence de valeur dans une clé primaire ne doit pas être autorisée. Autrement dit, la valeur vide (NULL) est interdite dans une colonne qui sert de clé primaire, ce qui n'est pas forcément le cas des autres colonnes, dont certaines peuvent ne pas être renseignées à toutes les lignes.

De plus, la valeur de la clé primaire d'une ligne ne devrait pas, en principe, changer au cours du temps.

**Convention** : La première colonne de toutes nos tables

- Se nomme 'id'
- Est de type 'INT'
- Est autoincrémentale
- Ne peut pas être vide (null)
- Est la clé primaire de la table

Par ailleurs, il se peut qu'une colonne Colonne1 d'une table ne doive contenir que des valeurs prises par la colonne Colonne2 d'une autre table (par exemple, le numéro du client sur une commande doit correspondre à un vrai numéro de client). Par conséquent, la Colonne2 doit être sans doublons (bien souvent il s'agit d'une clé primaire).



On dit alors que la Colonne1 ('type\_id' ci-dessus est **clé étrangère** et qu'elle **référence** la Colonne2 ('id').

**Convention** : nous nommerons une clé étrangère '(table)\_id', où (table) est le nom de la table référencée, au singulier.

Exemple : 'client\_id'

Remarques :

- Une même table peut avoir plusieurs clés étrangères mais une seule clé primaire
- Une clé étrangère peut être nulle, il faut alors le préciser dans la description des colonnes
- Les SGBDR vérifient au coup par coup que chaque clé étrangère ne prend pas de valeurs en dehors de celles déjà prises par la ou les colonne(s) qu'elle référence. Ce mécanisme qui agit lors de l'insertion, de la suppression ou de la mise à jour de lignes dans les tables, garantit ce que l'on appelle l'intégrité référentielle des données.

## Schéma relationnel

Concentrons-nous désormais sur le MLD, que nous réaliserons à l'aide du logiciel MySQL Workbench.

Le MLD appartient au domaine technique. Il n'est destiné qu'à être utilisé par les informaticiens.

**Convention** : Le MLD est rédigé en anglais

On peut représenter les tables d'une base de données relationnelle par un schéma relationnel dans lequel les tables sont appelées relations et les liens entre les clés étrangères et leur clé primaire sont symbolisés par un

connecteur :

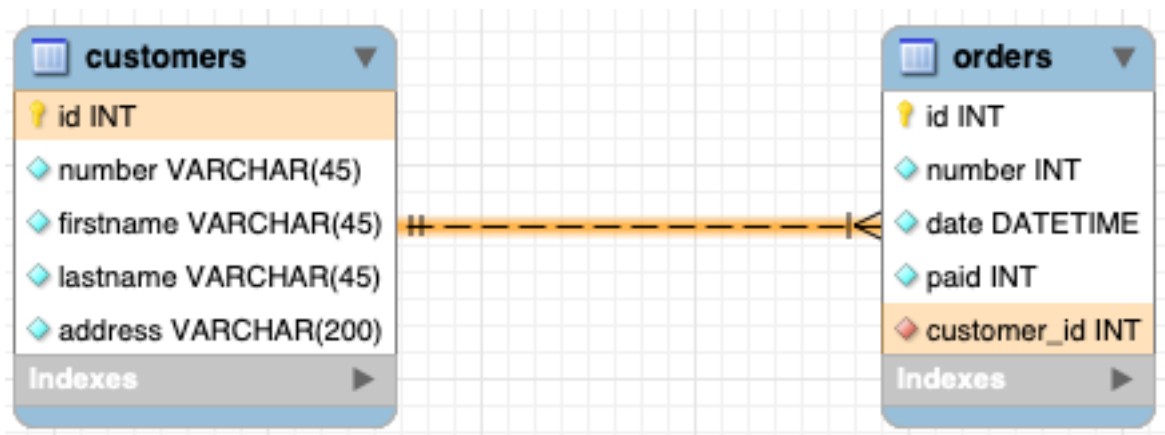


Fig. 26 – Schéma relationnel simple entre deux tables

### 3. Traduction d'un MCD en un MLD

Notations : on dit qu'une association binaire (entre deux entités ou réflexive) est de type :

- 1 : 1 (un à un) si aucune des deux cardinalités maximales n'est n ;
- 1 : n (un à plusieurs) si une des deux cardinalités maximales est n ;
- n : m (plusieurs à plusieurs) si les deux cardinalités maximales sont n.

En fait, un schéma relationnel ne peut pas faire la différence entre 0,n et 1,n. Par contre, il peut la faire entre 0,1 et 1,1.

Pour traduire un MCD en un MLD, il suffit d'appliquer cinq règles.

## Règle 1

Toute entité devient une table dans laquelle les attributs deviennent les colonnes. L'identifiant de l'entité devient un index de type UNIQUE :

The screenshot shows a database management interface. At the top, a dropdown menu for the 'items' table lists its attributes: id INT, brand VARCHAR(45), model VARCHAR(45), price DECIMAL(5,2), and description TEXT(1000). Below this, the 'Indexes' tab is active, showing a table with two indexes: 'PRIMARY' (PRIMARY type) and 'Uniqueltem' (UNIQUE type). The 'Uniqueltem' index is selected, and its details are shown on the right. The 'Index details 'Uniqueltem'' panel lists the columns included in the index: id, brand, model, price, and description. The 'brand' and 'model' columns are checked, indicating they are part of the index. The 'id' column is also checked but has a small 'x' icon next to it, possibly indicating it is not part of the index. The 'price' and 'description' columns are unchecked.

Index	Type
PRIMARY	PRIMARY
Uniqueltem	UNIQUE

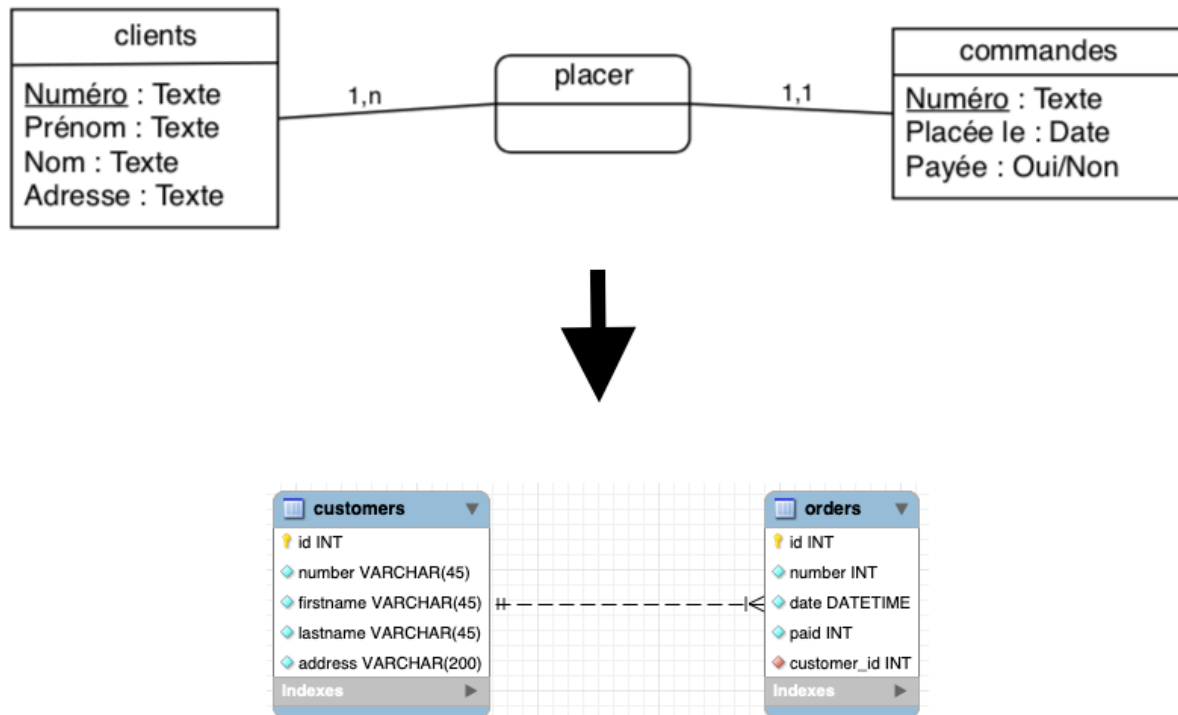
Index Columns	#	Order	Length
<input type="checkbox"/> id		ASC	
<input checked="" type="checkbox"/> brand	1	ASC	
<input checked="" type="checkbox"/> model	2	ASC	
<input type="checkbox"/> price		ASC	
<input type="checkbox"/> description		ASC	

**Convention** : On nommera l'index 'Unique<table>', où <table> est le nom de la table au singulier

## Règle 2

Une association binaire de type 1 : n disparaît, au profit d'une clé étrangère dans la table côté 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.

Exemple :

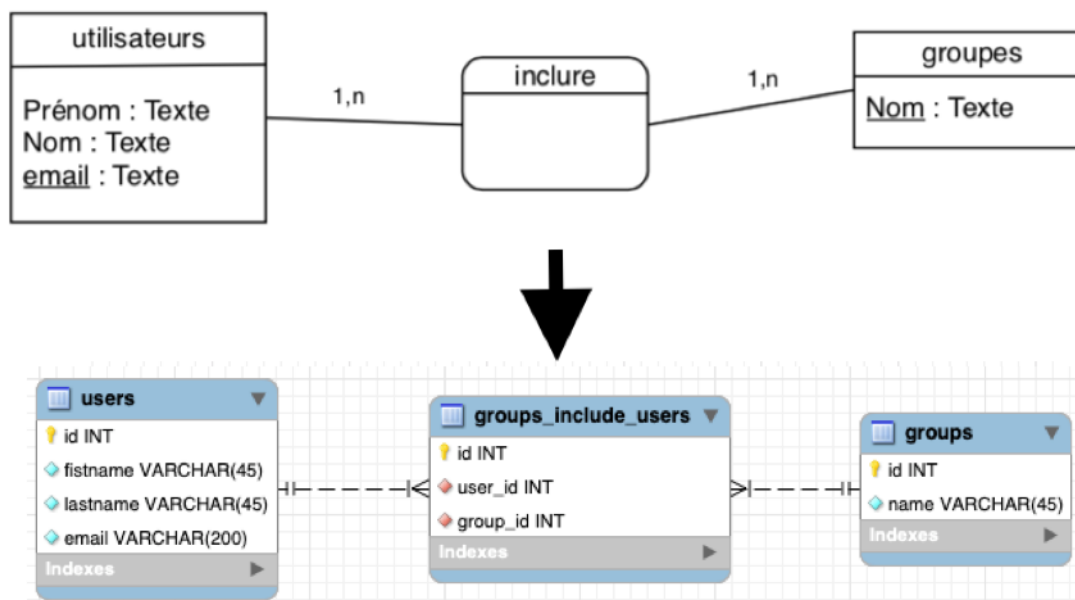


Il ne devrait pas y avoir d'attribut dans une association de type 1 : n, mais s'il en reste, alors ils glissent vers la table côté 1.

## Règle 3

Une association binaire de type  $n : m$  devient une table supplémentaire, appelée **table de jointure**.

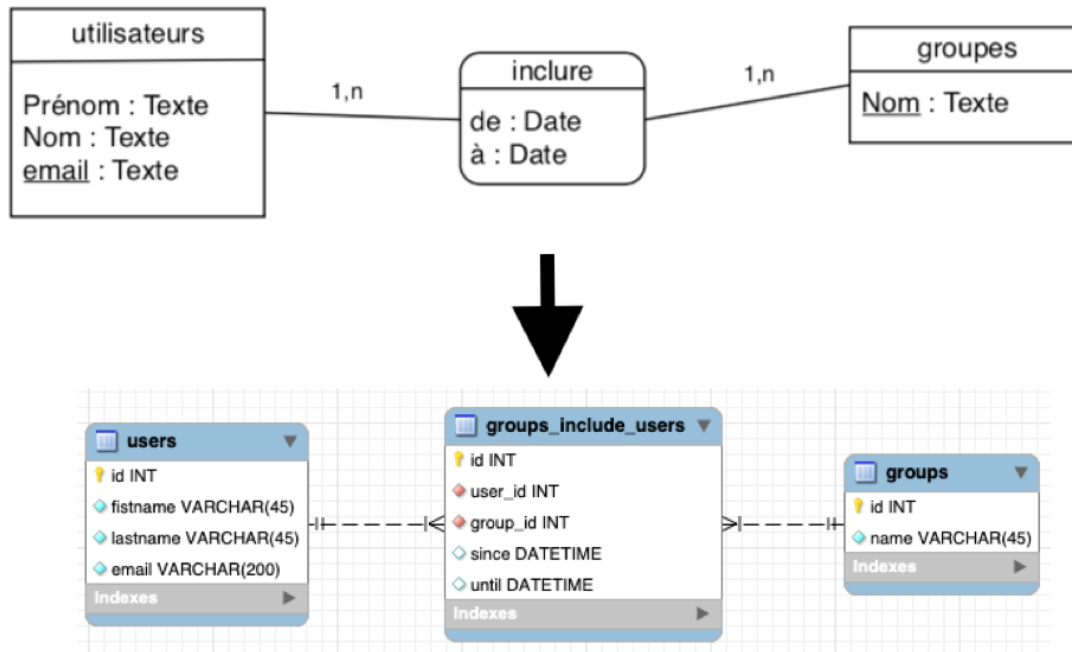
Exemple :



**Convention** : Le nom que l'on donnera à la table de jointure est la traduction de la relation du MCD, incluant les noms des deux tables, comme illustré dans l'exemple ci-dessus.

Si l'association a des attributs, ceux-ci se retrouvent dans la table de jointure.

Exemple :



On est en droit de se demander à quoi sert la colonne 'id' de la table 'belongs' dans le premier exemple. Les enregistrements de cette table ne portent aucune information propre, on ne va donc pas avoir besoin de les rechercher.

La réponse tient en deux points :

- Nous ne créons pas d'exception à notre convention, ce qui la rend plus facile à appliquer
- Cela facilite une possibilité d'évolution intéressante : celle de pouvoir tenir compte de la possibilité qu'un utilisateur rejoigne un groupe, le quitte et y revienne !

Cela se traduirait par un contenu de la table 'belongs' de ce genre :



id	user_id	group_id	since	until
1	4	1	2018-02-01 00:00:00	NULL
2	6	2	2018-03-01 00:00:00	2018-09-18 00:00:00
3	5	2	2018-01-01 00:00:00	2018-04-01 00:00:00
4	5	2	2018-08-10 00:00:00	NULL
5	8	1	2018-05-05 00:00:00	NULL

Dans le cas où nous ne voulons gérer que l'appartenance du moment ou non au groupe, il faut interdire le doublon des valeurs 5 et 2 des champs 'user\_id' et 'group\_id'. Cela se réalisera comme précédemment (Règle 1) avec un index UNIQUE.

## Règle 4

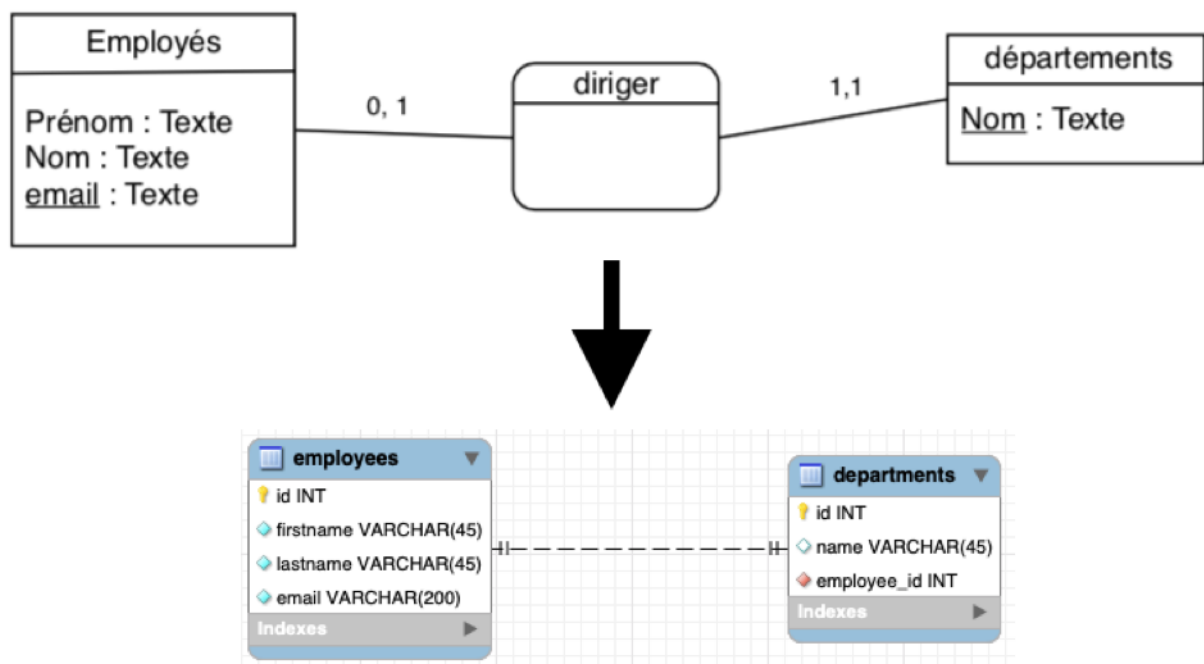
Si on a une association binaire de type 1 : 1 et que l'une des deux entités est impliquée dans une autre association, elle est traduite comme une association binaire de type 1 : n sauf que la clé étrangère :

- Doit être unique (contrainte d'unicité : la colonne correspondante ne peut prendre que des valeurs distinctes)
- Peut être vide (null) si la cardinalité est 0,1 de ce côté.

Si on a un côté de cardinalité 0,1 et l'autre de 1,1, la clé étrangère va du côté 1,1.

Si les deux côtés sont de cardinalité 0,1 ou 1,1 alors la clé étrangère peut être placée indifféremment dans l'une ou l'autre des deux tables.

Exemple :

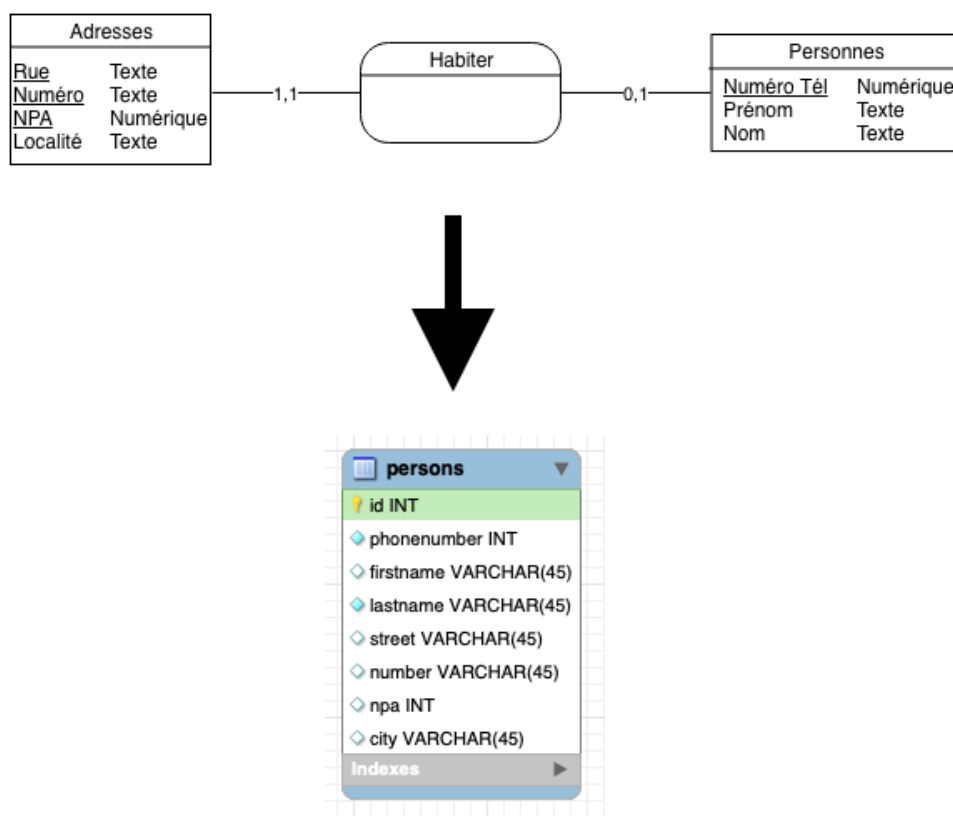


Remarque : dans cet exemple, le champ 'employee\_id' est unique et non nul à cause du 1,1 du côté de l'entité départements.

## Règle 5

Si on a une association binaire de type 1 : 1 et qu'aucune des deux entités n'est impliquée dans une autre association, on fusionne les deux entités dans une seule table.

Exemple :



## 4. Cartouche

Afin d'éviter au maximum toute confusion, un MLD contiendra toujours dans un de ses coins un cartouche contenant :

- Le nom du projet
- Un titre (car le MLD peut ne représenter qu'une partie de l'ensemble)
- Le ou les auteur(s)
- Une indication soit de version soit de date de publication (soit les deux)
- Le MCD (nom et version) dont il est tiré

Exemple :

