

Modèle conceptuel de données (MCD)

[cours YouTube E mds](#)

Généralités

C'est un modèle visuel permettant de schématiser les dépendances fonctionnelles, on le conçoit avant d'obtenir les relations qui permettront de construire la base de données.

Il respecte les règles du modèle relationnel (but : avoir des dépendances fonctionnelles élémentaires et directes respectant la troisième forme normale des relations 3NF). Il ajoute des règles de représentation visuelle.

Intérêt d'un MCD vis-à-vis d'un MLD : beaucoup plus facile à construire et plus facile à lire (aspect visuel marqué)

Les outils d'un MCD

- **Entités :** (coins droits) représentent une relation en 3NF dont la clé primaire n'est composée que d'un seul attribut
- **Associations :** (coins arrondis) représentent une relation en 3NF dont la clé primaire est composée de plusieurs attributs

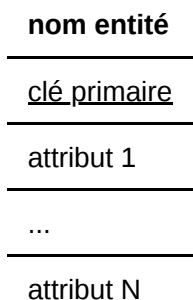
Les autres éléments d'un MCD dérivent de ces deux types d'éléments, entités et associations

Les éléments d'un MCD

Les entités

On la représente dans un rectangle (coins droits donc). Dans la partie haute figure le nom de la relation (= entité), en dessous la clé primaire (=identifiant de l'entité, on la souligne), ensuite la liste des attributs qui sont en DF à partir de la clé primaire

Représentation générique :



A retenir :

- une seule clé primaire, donc un seul attribut souligné
- la clé primaire est écrite en tête de l'entité (sous le nom de celle-ci)
- 0 à N attributs

- Les clés étrangères ne sont pas écrites dans l'entité (elles sont représentées par des liens entre les différentes entités)

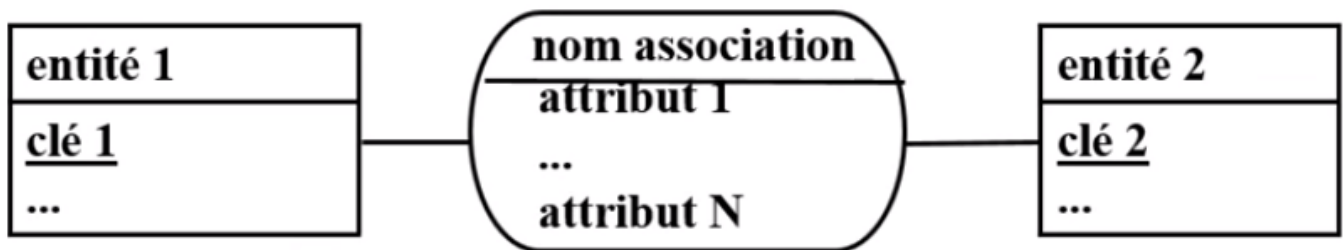
Exemple : relation Article

ARTICLE
<u>n° article</u>
nom_article
prix

Les associations

Une association fait un lien entre deux entités. On la représente par un rectangle aux coins arrondis. En haut, le nom de l'association, en dessous la liste des attributs

Représentation générique :



Exemple : L'association entre les entités Commande et Article se nomme LIGNE_COM et possède un seul attribut qté_commandée



A retenir :

- une association a obligatoirement plusieurs clés primaires
- les clés primaires sont représentées par les liens directs entourant l'association
- il y a au minimum deux liens directs autour de l'association
- 0 à N attributs dans l'association
- les clés étrangères ne sont pas écrites dans l'association directement

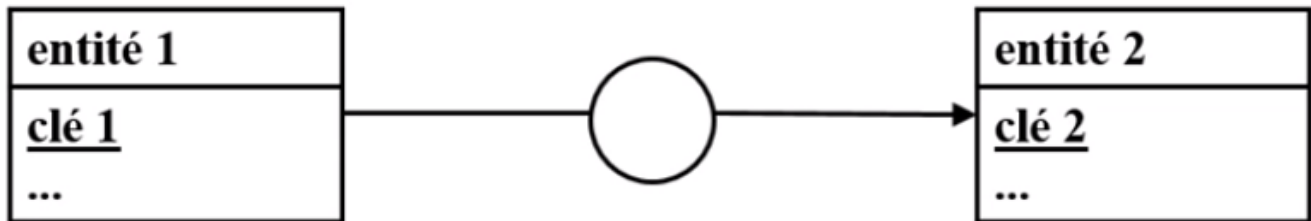
Les DF entre entités

Elle représente une DF entre les clés primaires des deux objets (entités ou associations).

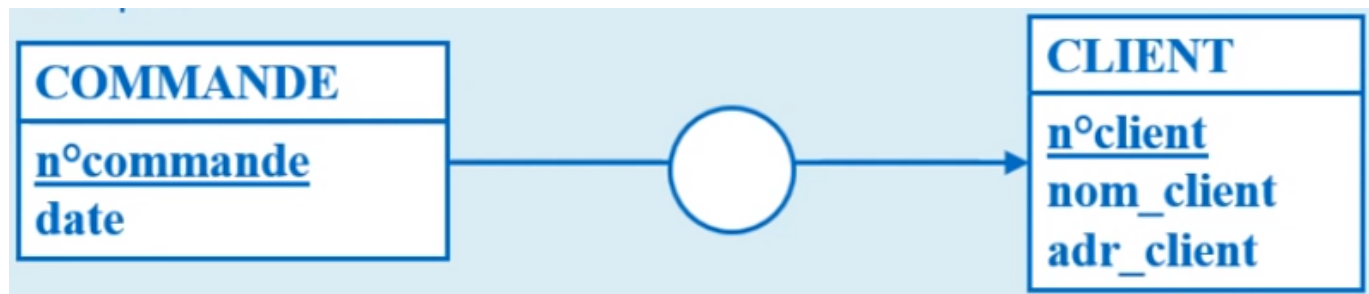
On parle de Contrainte d'Intégrité Fonctionnelle (CIF) quand la DF est forte, c'est-à-dire lorsque nécessairement un objet donne un autre objet.

On la représente par un cercle à cheval sur une flèche entre les deux entités.

Représentation générique : un élément de l'entité 1 correspond à un et un seul élément de l'entité 2



Exemple : une commande est associée à un client et un seul (cardinalité de type 1,1 dans le sens Commande -> Client)

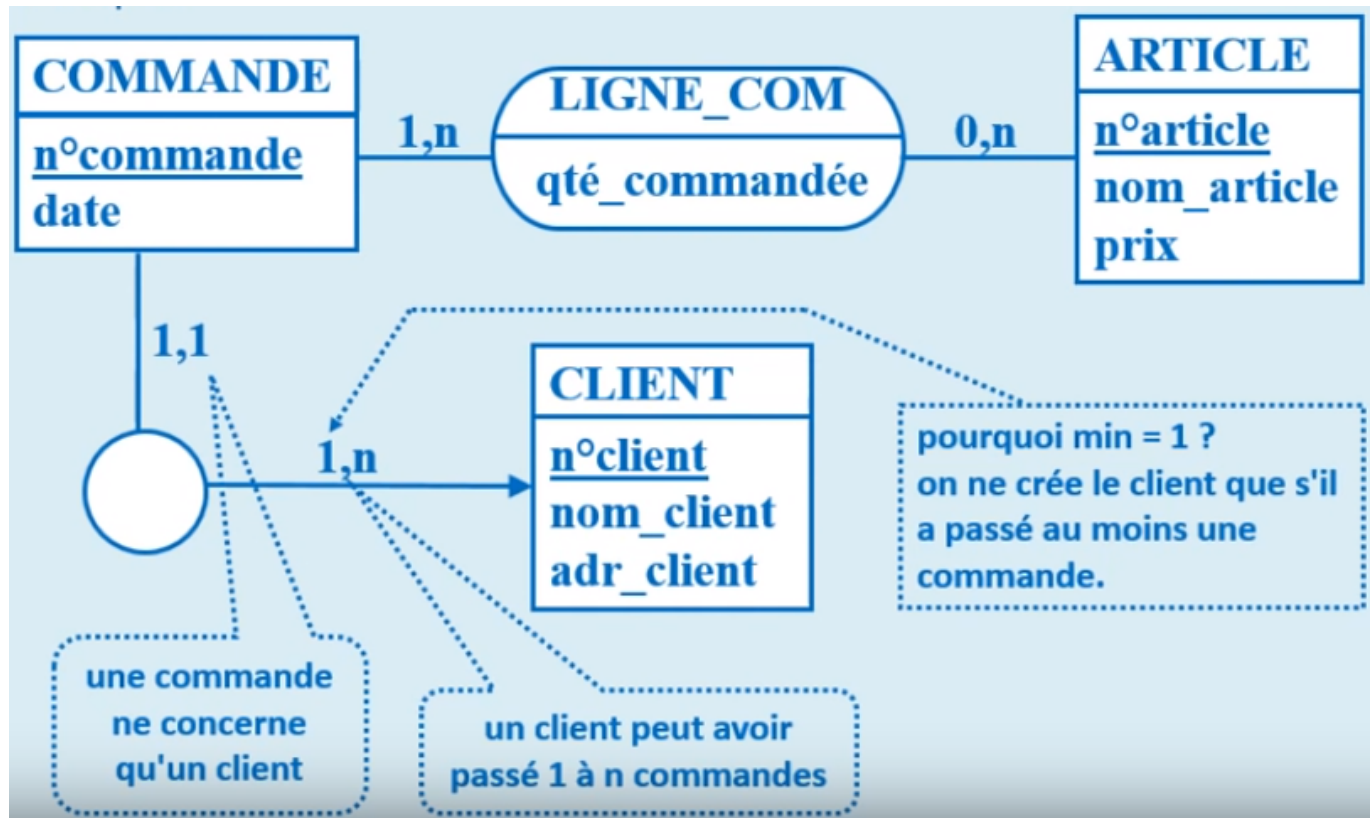


A retenir :

- la flèche n'est pas obligatoire mais apporte du sens au niveau visuel
- la représentation sous forme d'association est acceptée, cependant on notera que cette association n'est pas une relation et ne donnera donc pas une table dans la base de données

Les cardinalités

Une cardinalité permet d'exprimer le nombre minimum et maximum d'occurrences d'une entité par rapport à une autre (entités liées par une association) : nombre_minimum, nombre_maximum. On le note au dessus des flèches portées par les associations.

Exemple :**Les associations vides**

Une CIM (Contrainte d'Intégrité Multiple) est une association non porteuse de données (association vide). Elle ne doit son existence que par un besoin au niveau des traitements.

Exemple : un livre peut avoir plusieurs auteurs. Il sera nécessaire d'éditer un "catalogue" de livres par auteur, on crée alors une association entre les entités Livre et Auteur permettant pour chaque livre, de connaître la liste de ses auteurs, et inversement de connaître tous les livres écrits par un auteur donné

**A retenir :**

- cette association vide ne contient pas d'attribut, elle contiendra une clé primaire n° livre - n° auteur.

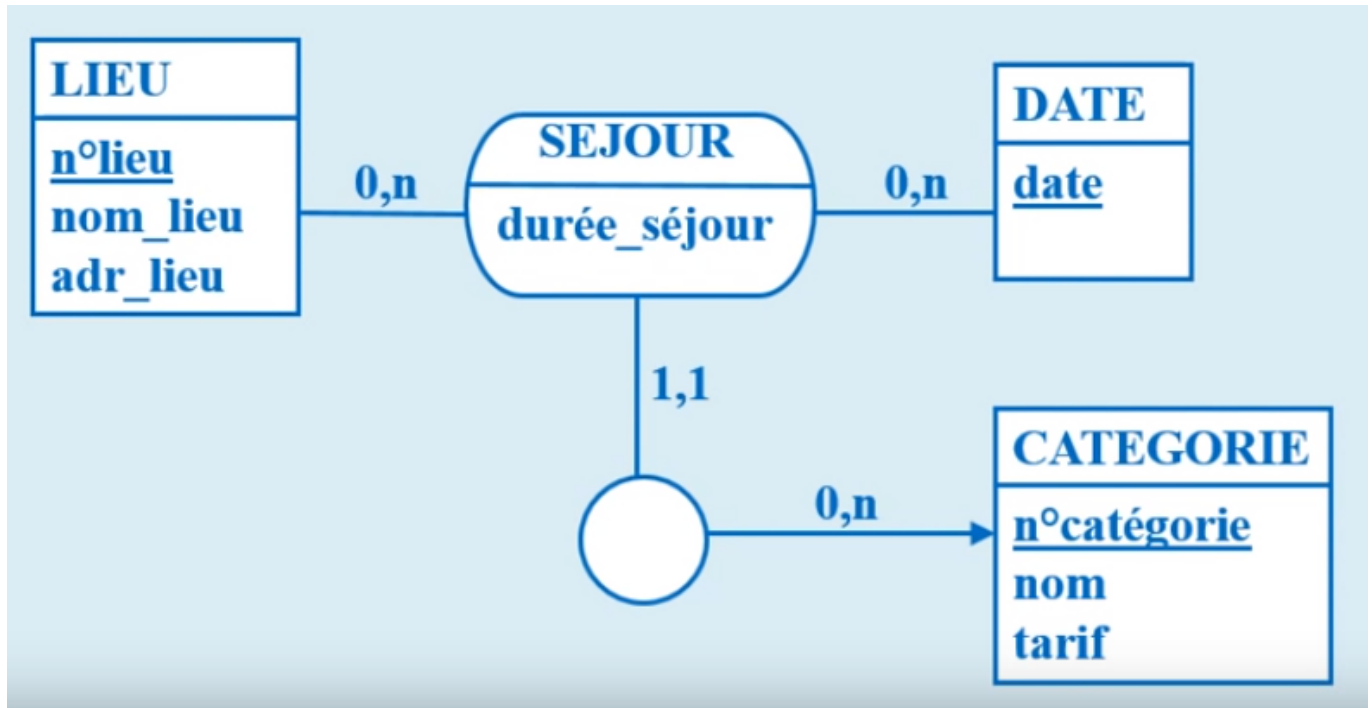
Exemple : l'auteur d'identifiant 8 a écrit le livre d'identifiant 12

n° auteur_livre	n° livre	n° auteur
1	12	8
...

Les CIF sur une association

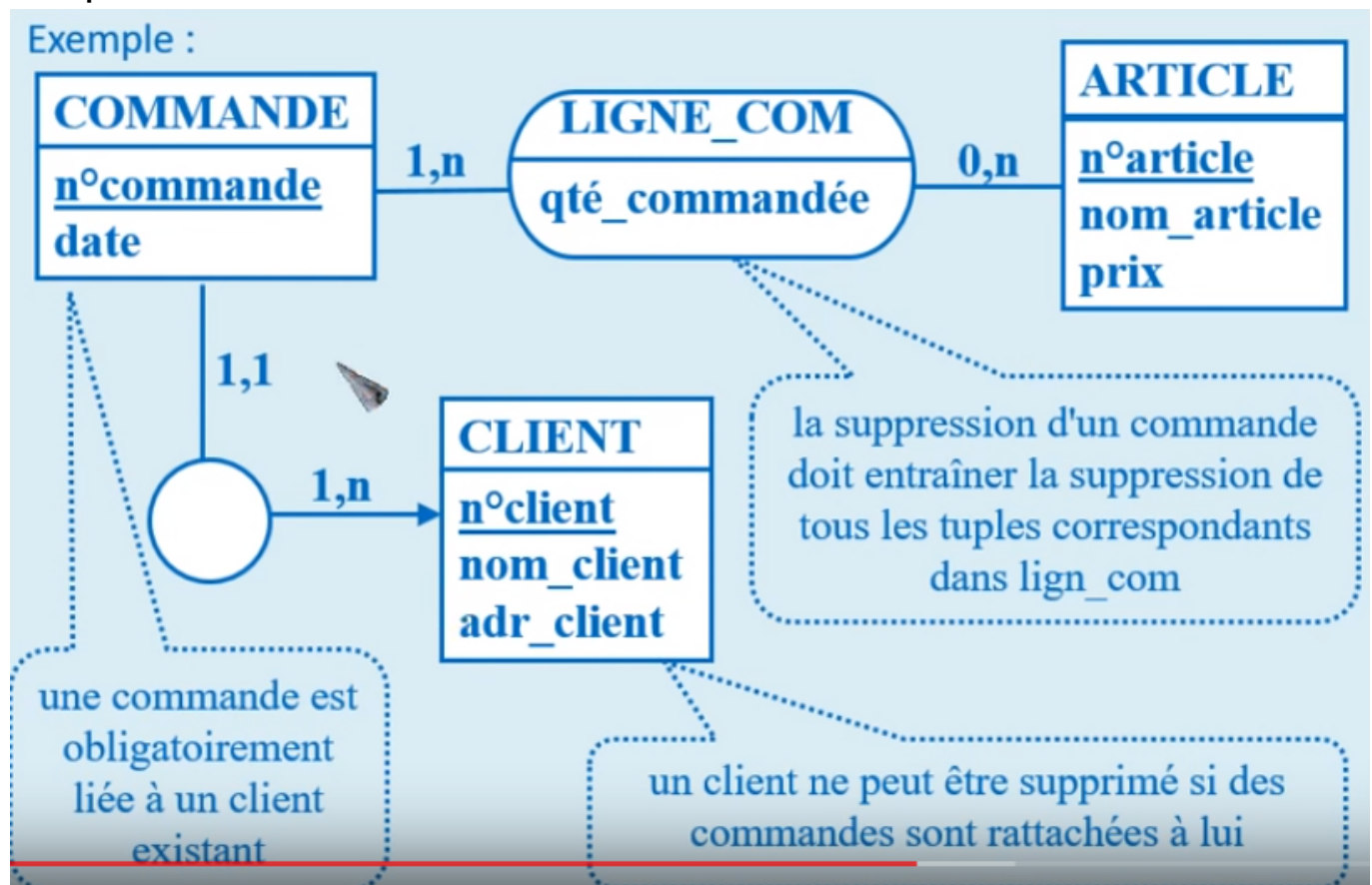
Elles traduisent une DF entre la clé primaire d'une association et la clé primaire d'une entité. Autrement dit, une DF peut partir d'une entité comme ici d'une association.

Exemple : il existe une DF entre la clé primaire de l'association Sejour et celle de l'entité Catégorie (n° categorie)



Notion d'intégrité fonctionnelle

Une intégrité fonctionnelle traduit une dépendance forte entre des objets : elle permet d'éviter les données parasites.

Exemple :

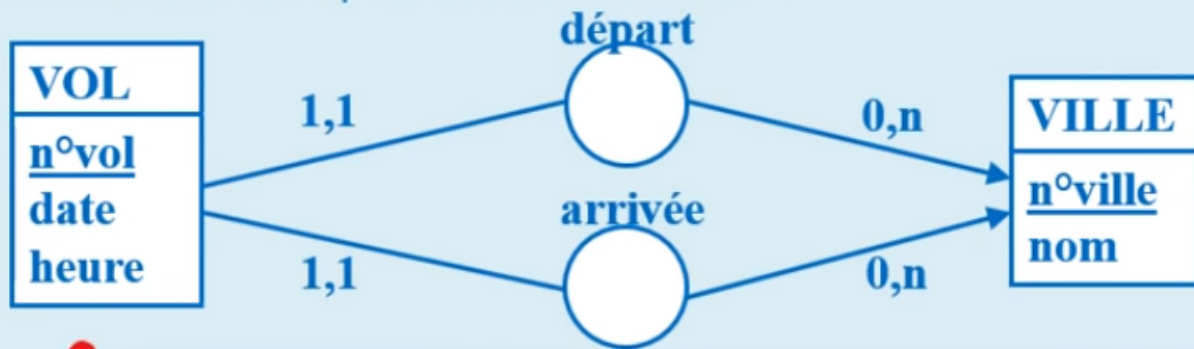
L'historique

La double CIF

Elle représente la nécessité d'obtenir deux informations provenant d'une même entité.

Exemple : un vol possède une ville de départ et une ville d'arrivée (on considère une seule entité Vol afin de ne pas générer de doublons en base de données)

Un vol a une ville de départ et une ville d'arrivée.



Au niveau relationnel, il faut faire attention aux noms.

VILLE (n°ville, nom)

n°ville : clé primaire

VOL (n°vol, date, heure, n°ville_depart, n°ville_arrivee)

n°vol : clé primaire

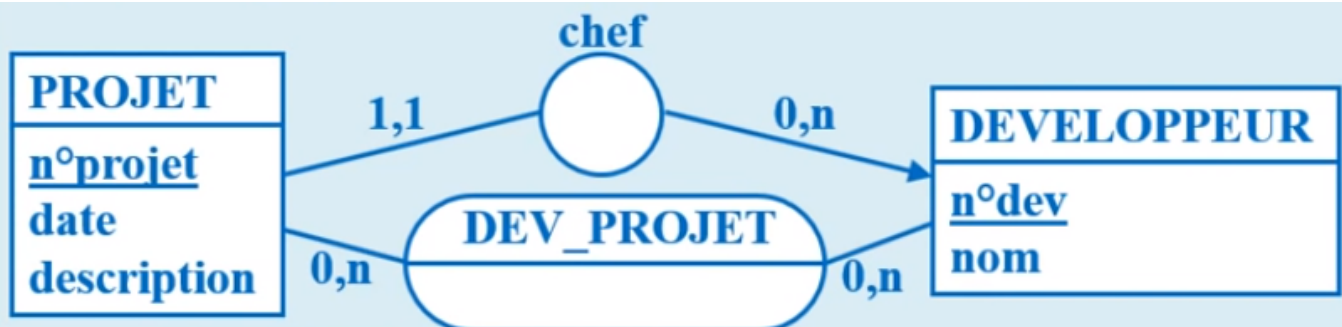
n°ville_depart : clé étrangère en référence à n°ville de VILLE

n°ville_arrivee : clé étrangère en référence à n°ville de VILLE

La CIF + association

On fait ressortir un cas particulier d'une liste.

Exemple : un projet est géré par des développeurs dont l'un est le chef de projet



DEVELOPPEUR (n°dev, nom)

n°dev : clé primaire

PROJET (n°projet, date, description, n°chef)

n°projet : clé primaire

n°chef : clé étrangère en référence à n°dev de DEVELOPPEUR

DEV_PROJET (n°projet, n°dev)

n°projet, n°dev : clé primaire

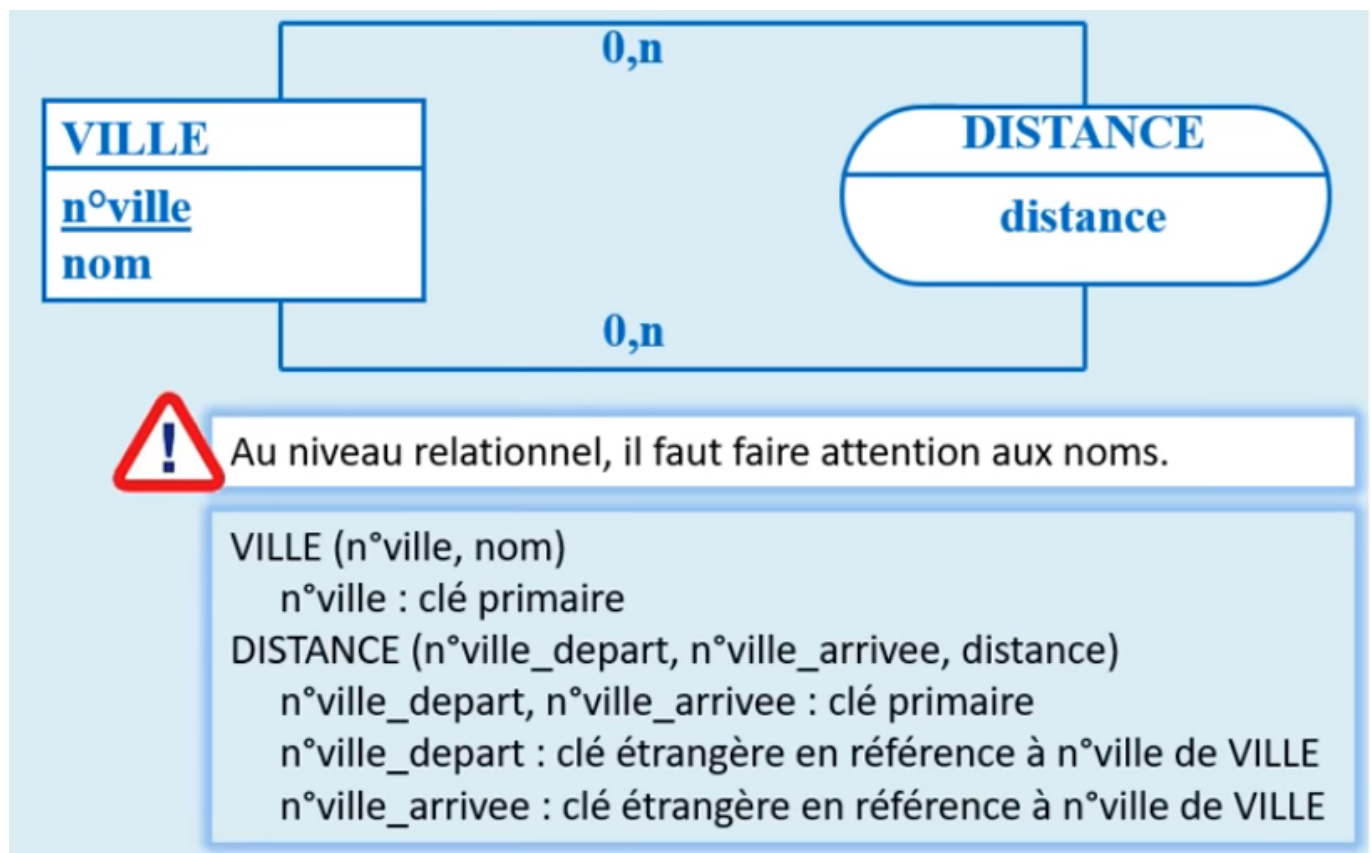
n°projet : clé étrangère en référence à n°projet de PROJET

n°dev : clé étrangère en référence à n°dev de DEVELOPPEUR

L'association réflexive

Elle est nécessaire lorsqu'il faut lier une information avec elle-même.

Exemple : mémoriser les distances entre les villes (entité ville et association distance)



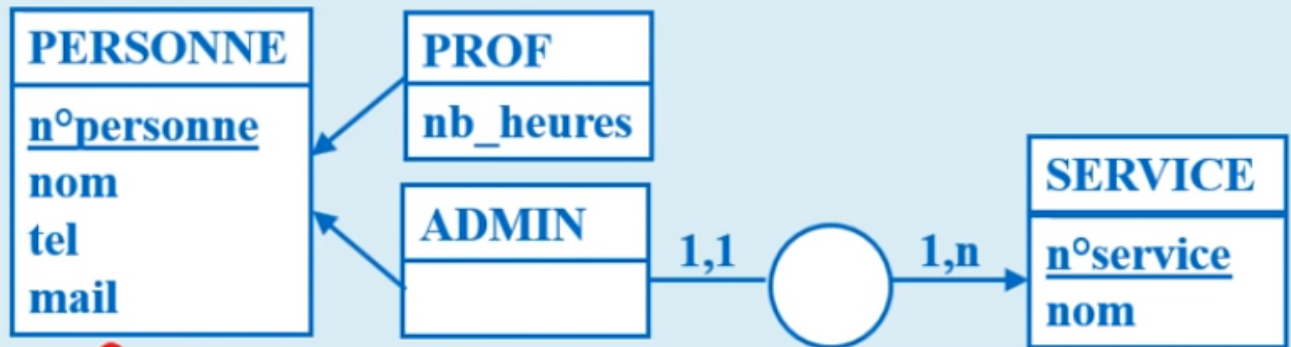
L'héritage

Il intervient pour distinguer des attributs communs et d'autres spécifiques. Elle permet d'éviter que des attributs restent vides dans l'entité principale (entité mère).

Exemple : gérer des personnes (nom, tél, mail) avec le nombre d'heures d'enseignement pour les profs et le service pour les administratifs.

Personne est l'entité mère, Prof et Admin sont des entités filles. Des flèches partent des entités filles vers l'entité mère afin de modéliser cet héritage

⚠ Certains logiciels de conception de MCD représentent l'héritage par un triangle



Au niveau relationnel, on peut changer les noms des clés.

PERSONNE (n°personne, nom, tel, mail)

n°personne : clé primaire

PROF (n°prof, nb_heures)

n°prof : clé primaire

n°prof : clé étrangère en référence à n°personne de PERSONNE

...