

Conception des Systèmes d'Information

Année Universitaire 2022-2023

Filière IRISI 2



Pr. Sara Qassimi

Chapitre 3:

MLD

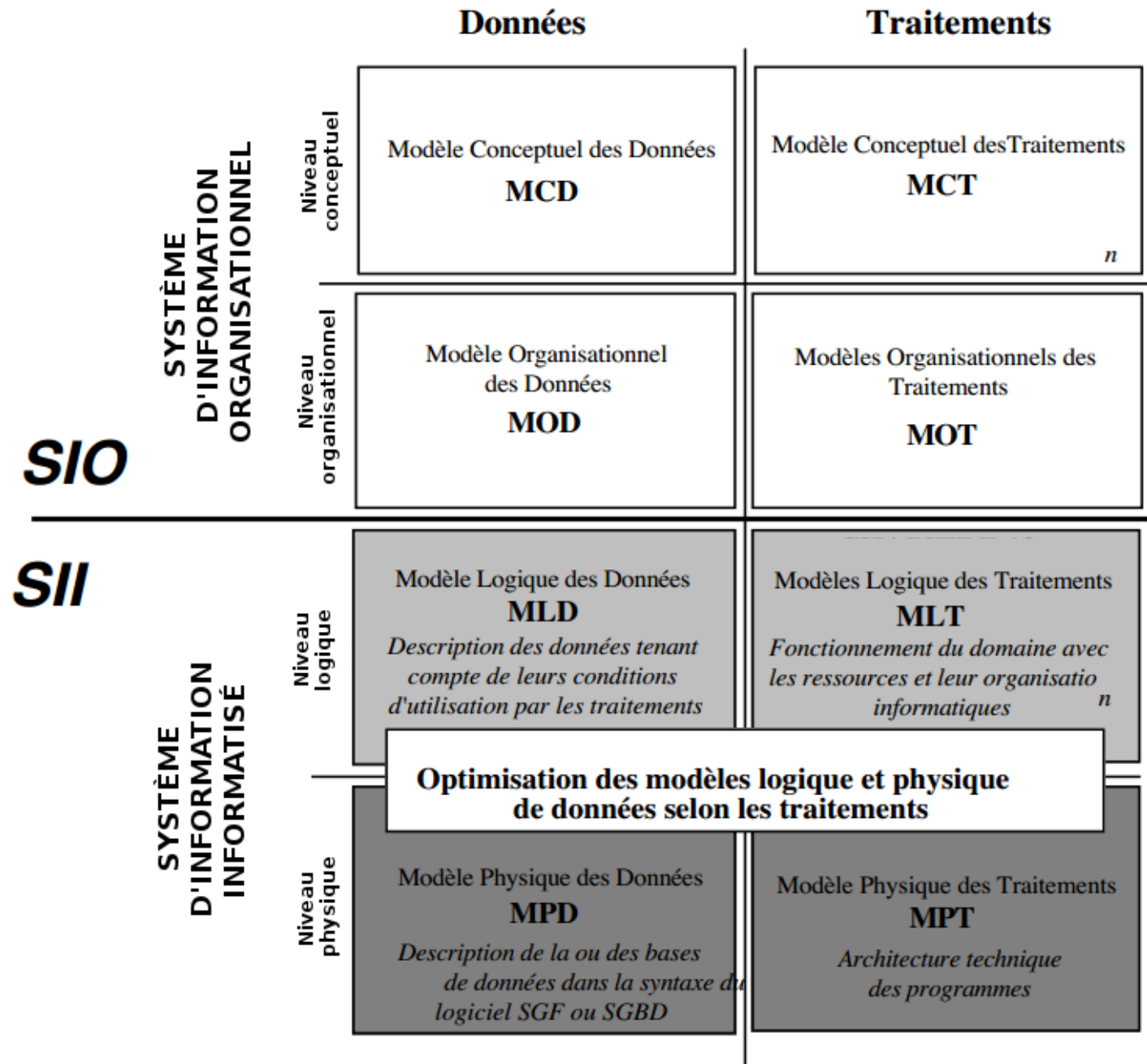
Modèle Logique des Données

L'étude du Système d'Information Informatisé (SII)

Modèle Logique de Données

- **Objectifs:**
 - Base de données relationnelle
 - Modèle Logique de Données (MLD)
 - Passage MCD au MLD
 - Normalisation

La « jungle » des modèles !			
Méthode	MCD	MLD	MPD
Langage	MEA, schéma E-R, UML	MR	SQL



LE CYCLE D'ABSTRACTION		
Niveaux	DONNEES	TRAITEMENTS
CONCEPTUEL	M C D	M C T
QUOI	<i>Modèle conceptuel des données</i> Signification des informations sans contraintes techniques, organisationnelles ou économiques. Modèle entité – association	<i>Modèle conceptuel des traitements</i> Activité du domaine sans préciser les ressources et leur organisation
ORGANISATIONNEL	M O D	M O T
QUI, OU, QUAND	<i>Modèle organisationnel des données</i> Signification des informations avec contraintes organisationnelles et économiques. (Répartition et quantification des données ; droit des utilisateurs)	<i>Modèle organisationnel des traitements</i> Fonctionnement du domaine avec les ressources utilisées et leur organisation (répartition des traitements sur les postes de travail)
LOGIQUE	M L D	M L T
COMMENT	<i>Modèle logique des données</i> Description des données tenant compte de leurs conditions d'utilisation (contraintes d'intégrité, historique, techniques de mémorisation). Modèle relationnel	<i>Modèle logique des traitements</i> Fonctionnement du domaine avec les ressources et leur organisation informatique.
PHYSIQUE	M P D	M P T
COMMENT	<i>Modèle physique des données</i> Description de la (ou des) base(s) de données dans la syntaxe du Système de Gestion des données (SG.Fichiers ou SG Base de Données) Optimisation des traitements (indexation, dénormalisation, triggers).	<i>Modèle physique des traitements</i> Architecture technique des programmes

Merise: niveau Logique

Réponse à la question : **OUI ? QUAND ? OU ?**

- Description du système, indépendamment du logiciel SGBD
- Passage 'automatique' au modèle relationnel

**Modèle Logique de Données
MLD**

- Description indépendante de la machine
- Structuration en procédure

**Modèle Logique de Traitements
MLT**

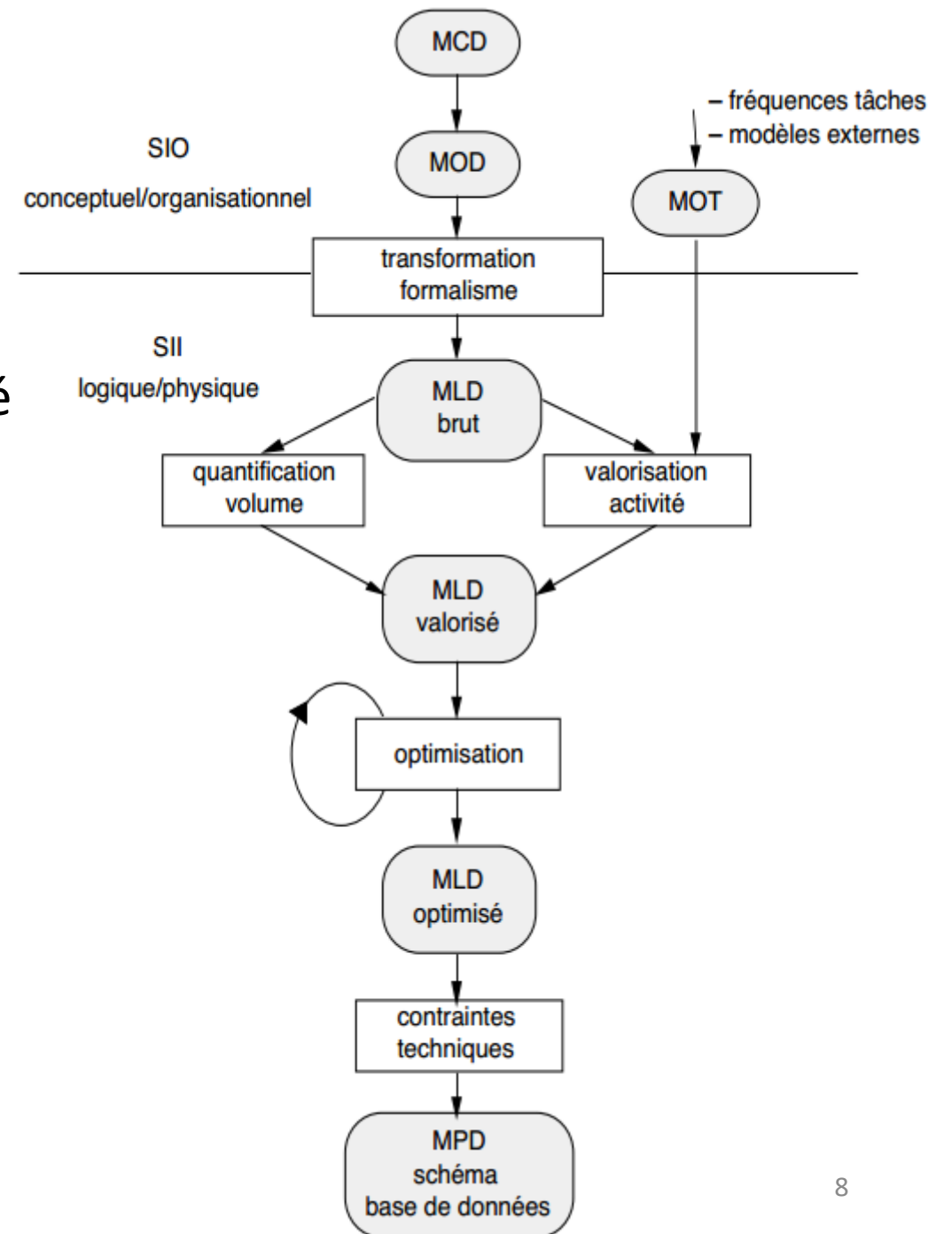
Modèle Relationnel de Données

- Une base de données où l'information est organisée dans des tableaux à deux dimensions appelés des relations ou tables, selon le modèle introduit par Edgar F. Codd en 1970.
- Selon le modèle relationnel, une base de données consiste en une ou plusieurs relations.
- Les lignes de ces relations sont appelées des nuplets ou enregistrements.
- Les colonnes sont appelées des attributs.
- Les logiciels qui permettent de créer, utiliser et maintenir des bases de données relationnelles sont des systèmes de gestion de base de données relationnels (SGBDR).
- Le modèle relationnel présente deux aspects fondamentaux :
 - une algèbre permettant de manipuler des tables ou relations
 - une démarche de conception permettant de définir une collection de relations.

Dans ce chapitre, nous ne présenterons de ce modèle que l'essentiel, nécessaire pour conduire l'informatisation d'un système d'information.

- La modélisation logique des données conduira aux opérations suivantes :

- transformation du MOD, exprimé en formalisme entité-relation, en un MLD exprimé dans un formalisme logique adapté au SGBD envisagé ;
- quantification en volume du modèle logique
- valorisation de l'activité générée par les modèles externes associés aux traitements (tâches du MOT)
- optimisation générale.



Modèle Logique de Données (MLD)

- La modélisation logique des données est une représentation des données, issue de la modélisation conceptuelle puis organisationnelle des données.
- Il est aussi appelé modèle relationnel.
- Appelé aussi:
 - Modèle logique de données relationnelles (MLDR)
 - Modèle relationnel de données (MRD)
 - Modèle relationnel logique de données (MLDR)
- Le MCD ne peut pas être implanté dans une base de données sans modification. Il est obligatoire de transformer ce modèle.
- On dit qu'on effectue un passage du modèle conceptuel de données vers le modèle logique de données.
- Le MLD pourra être implanté dans une base de données relationnelle.

La « jungle » des modèles !			
Méthode	MCD	MLD	MPD
Langage	MEA, schéma E-R, UML	MR	SQL

Les contraintes d'intégrité

- Une contrainte d'intégrité est une assertion qui doit être vérifiée par les valeurs d'attributs de tables constituant une base de données.
- Les deux principaux types de contraintes d'intégrité sont la contrainte d'unicité de valeur, qui nous permettra de définir la clé primaire d'une table, et la contrainte référentielle permettant de relier deux tables.
- Une contrainte référentielle est un lien sémantique défini entre deux tables. Ce lien est réalisé par une duplication de la clé primaire d'une table dans une autre table. Cette clé dupliquée est appelée *clé étrangère*.

Règles de passage du MCD au MLD

- Règle numéro 1 :

a) Une entité du MCD devient une relation, c'est à dire une table.

- Dans un SGBD de type relationnel, une table est structure tabulaire dont chaque ligne correspond aux données d'un objet enregistré (d'où le terme enregistrement) et où chaque colonne correspond à une propriété de cet objet. Une table contiendra donc un ensemble d'enregistrements.
- Une ligne correspond à un enregistrement. Une colonne correspond à un champ.
- La valeur prise par un champ pour un enregistrement donné est située à l'intersection ligne-colonne correspondant à enregistrement-champ. Il n'y a pas de limite théorique au nombre d'enregistrements que peut contenir une table. Par contre, la limite est liée à l'espace de stockage.
- On définit aussi pour la table les notions suivantes :
 - cardinalité (nombre de lignes ou tuples de la table) ;
 - degré (n, nombre de colonnes ou d'attributs de la table).

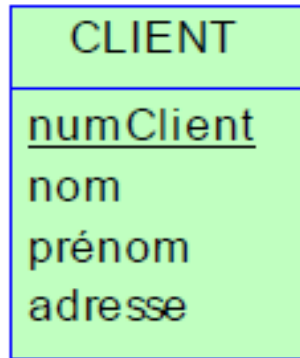
b) Son identifiant devient la clé primaire de la relation.

- 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.
- Dans la plupart des SGBDR, le fait de définir une clé primaire donne lieu automatiquement à la création d'un index.

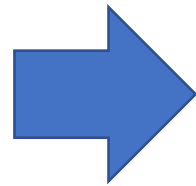
c) Les autres propriétés deviennent les attributs de la relation.

Exemple: Règle numéro 1

Entité



Relation



CLIENT(numClient , nom , prénom , adresse)

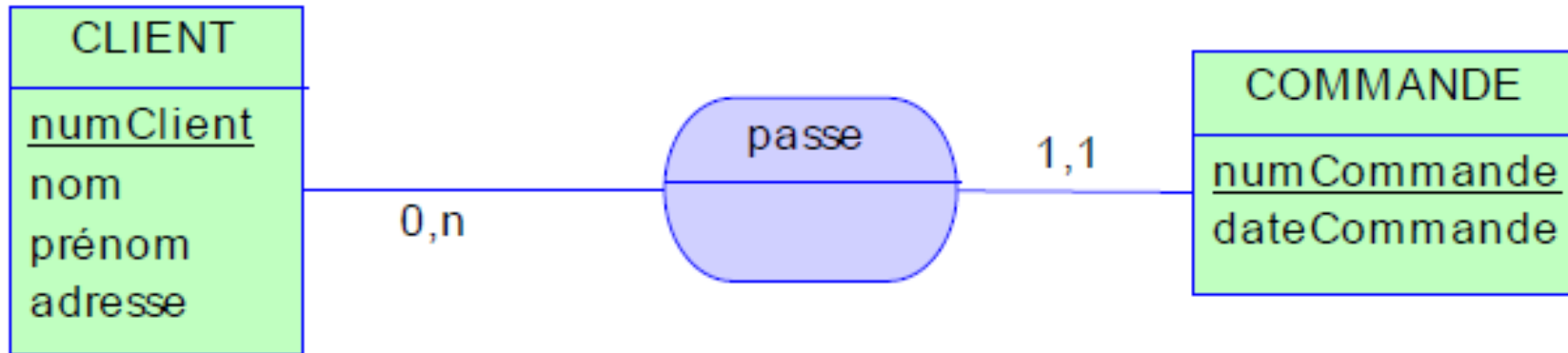
Table : CLIENT

Enregistrement

numClient	nom	prénom	adresse
1	Dupon	Angel	Av Renne, num 108, Nancy
2	Duran	Frederiq	Apprt 4, 10 arrondi, Paris
3	Alami	Ali	Rue105,av yassin, Fes

Règles de passage du MCD au MLD

- Règle numéro 2 :
- Relation binaire (0,n)-(1,1) ou (1,n)-(1,1)
- se traduit par la création d'une clé étrangère dans la relation correspondante à l'entité côté « 1 ».
- Cette clé étrangère référence la clé primaire de la relation correspondant à l'autre entité.



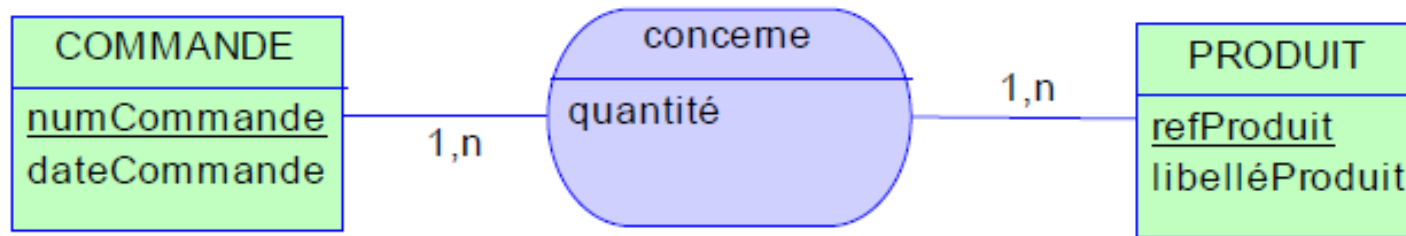
CLIENT(numClient , nom , prénom , adresse)

COMMANDE(numCommande , dateCommande , #numClient)

Règles de passage du MCD au MLD

- **Règle numéro 3:**

- Relation binaire (0,n)-(0,n) ou (1,n)-(1,n) ou (1,n)-(0,n)
- se traduit par la création d'une relation dont la clé primaire est composée des clés étrangères référençant les relations correspondant aux entités liées par l'association.
- Les éventuelles propriétés de l'association deviennent des attributs de la relation.



COMMANDE(numCommande ,dateCommande)
PRODUIT(refProduit, libelleProduit)
CONCERNE(#numCommande , #refProduit , quantité)

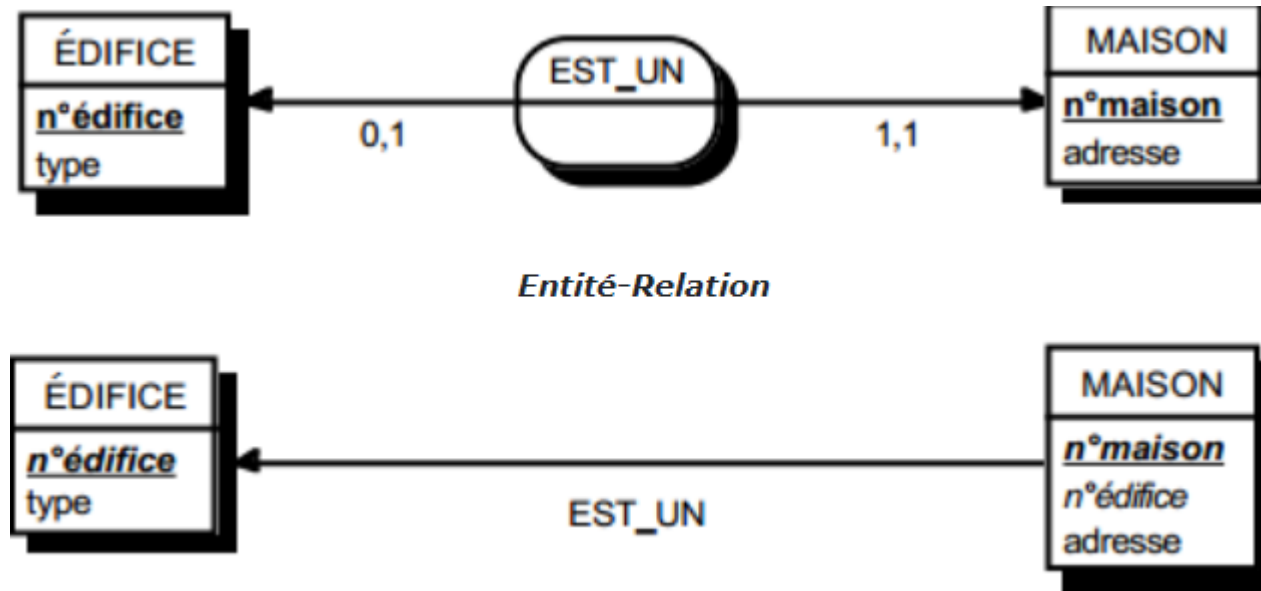
Si le nom du MCD n'est pas significatif, on peut renommer le nom de la table.
Dans notre exemple, plutôt que la table « CONCERNE », on la nommera « LIGNE_DE_COMMANDE ».

LIGNE_DE_COMMANDE (#numCommande , #refProduit , quantité)

Règles de passage du MCD au MLD

- **Règle numéro 4:**

- Relation binaire (0,1)-(1,1)
- C'est en fait une particularisation des cas précédemment traités, correspondant souvent à exprimer des sous-types. On duplique la clé de la table issue de l'entité à cardinalité (0,1) dans la table issue de l'entité à cardinalité (1,1)



- Schémas relationnels associés :

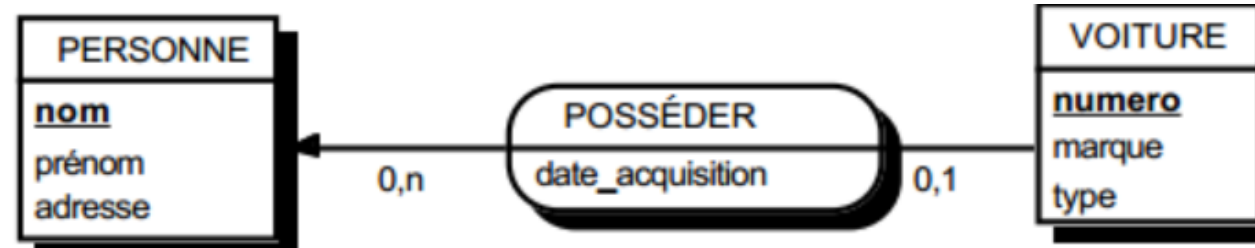
table EDIFICE (n°édifice, type)

table MAISON (n°maison, #n°édifice, adresse)

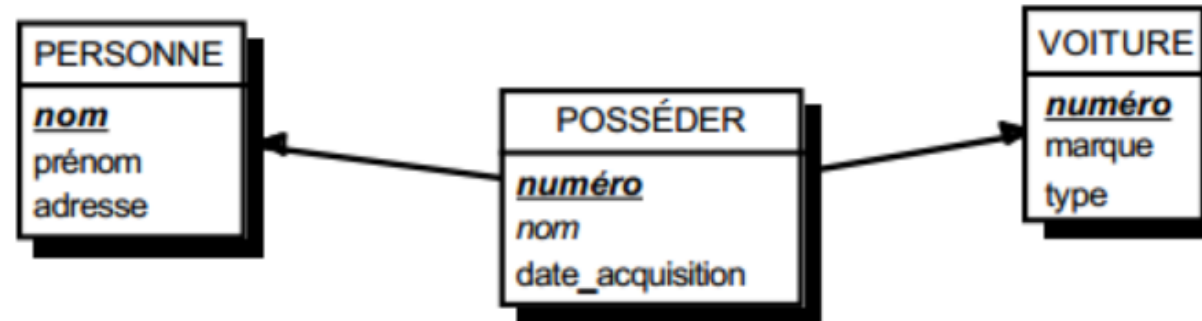
Règles de passage du MCD au MLD

- **Règle numéro 5:**

- Relation binaire (0,n)-(0,1) ou (1,n)-(0,1)
- Deux solutions sont possibles.
- La première solution (solution 1) consiste à créer une table avec comme clé primaire l'identifiant de l'entité à cardinalité (0,1) ; l'identifiant de l'autre entité devenant clé étrangère de cette table. Les éventuelles propriétés de la relation deviennent aussi attributs de la table issue de la relation.



Entité-Relation

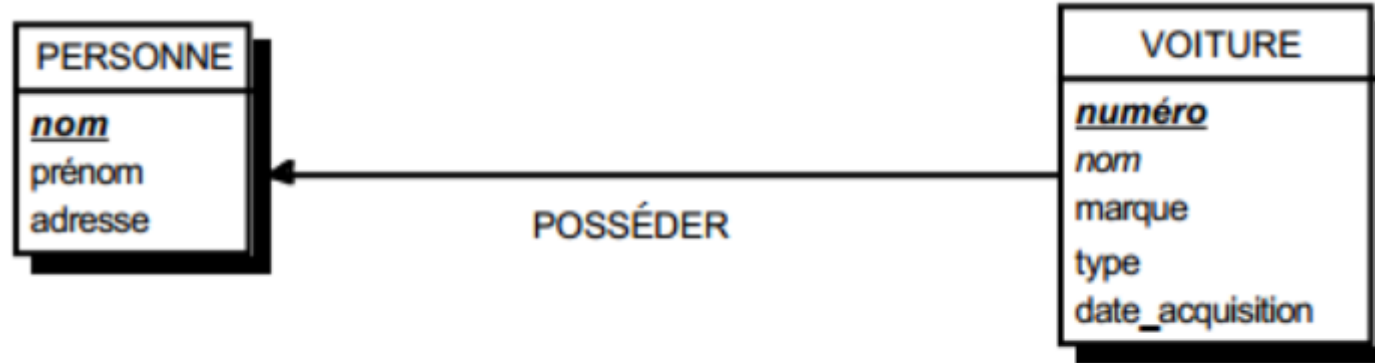


Relationnel dérivé : solution 1

Règles de passage du MCD au MLD

- **Règle numéro 5:**

- Relation binaire (0,n)-(0,1) ou (1,n)-(0,1)
- Deux solutions sont possibles.
- Dans la seconde solution (solution 2), on duplique comme clé étrangère la clé de la table issue de l'entité à cardinalité (0,n) ou (1,n) dans la table issue de l'entité à cardinalité (0,1).



Relationnel dérivé : solution 2

- Remarque : dans la seconde solution, la cardinalité (0,1) posera le problème, pouvant être difficile à gérer selon le SGBD adopté, d'accepter des valeurs nulles sur la clé étrangère.

Règles de passage du MCD au MLD

- **Règle numéro 6:**

- Relation binaire (0,1)-(0,1)

- 4 solutions sont possibles.

- Solution 1 :

table ENTREPRISE (n°entreprise, adresse) ;
table TIERS (n°tiers, type) ;
table CORRESPOND (n°entreprise, #n° tiers).

- Solution 2 :

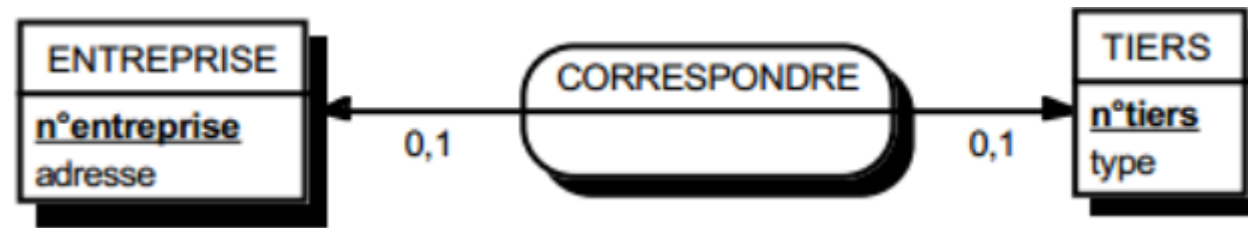
table ENTREPRISE (n°entreprise, adresse) ;
table TIERS (n° tiers, type) ;
table CORRESPOND (#n°entreprise, n° tiers).

- Solution 3 :

table ENTREPRISE (n°entreprise, adresse) ;
table TIERS (n° tiers, #n°entreprise, adresse).

- Solution 4 :

table ENTREPRISE (n° entreprise, #n° tiers, adresse) ;
table TIERS (n° tiers, type).

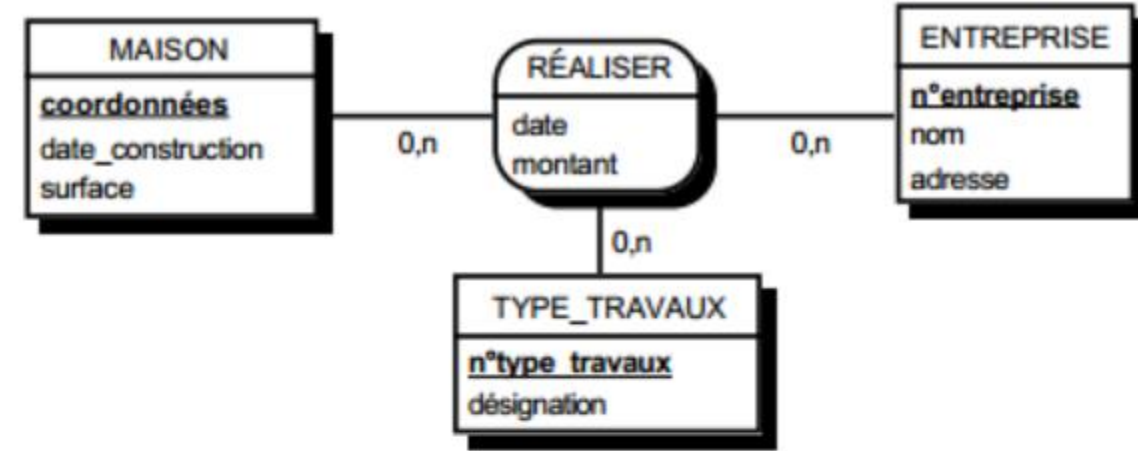


Les solutions 3 et 4, la cardinalité (0,1) posera le problème, pouvant être difficile à gérer selon le SGBD adopté, d'accepter des valeurs nulles sur l'attribut migrant.
On peut considérer ces solutions comme associée à un choix d'optimisation.

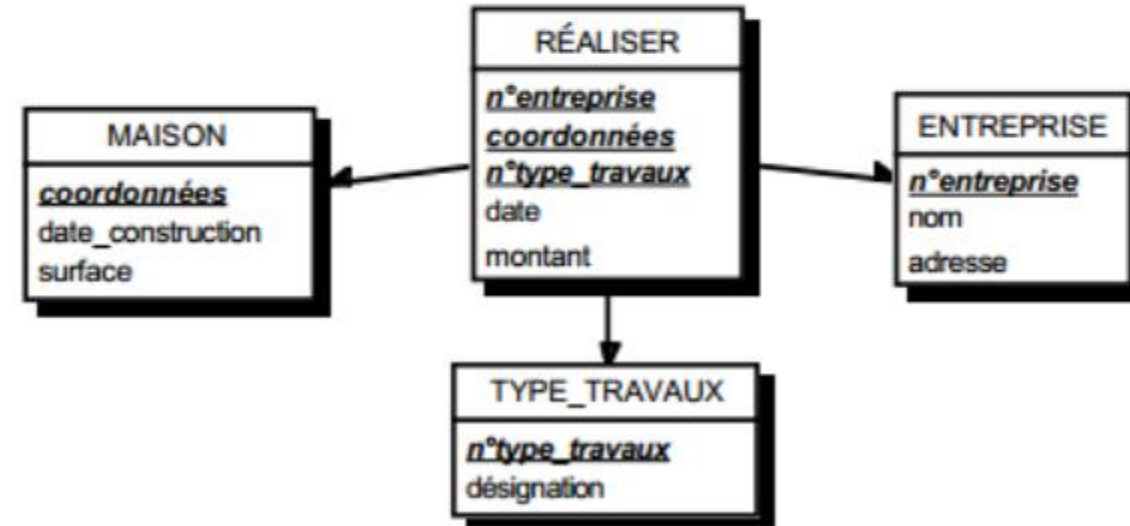
Règles de passage du MCD au MLD

- **Règle numéro 7:**

- Relation ternaire ou supérieure
- La transformée d'une relation ternaire ou supérieure, quelles que soient les cardinalités, consiste à créer une table ayant comme clé une clé composée des identifiants des diverses entités reliées par la relation considérée.
- Remarque : rappelons qu'une relation n-aire munie de cardinalité (1,1) aura été au préalable décomposée, comme nous l'avons déjà indiqué dans la partie précédente.



Entité-Relation

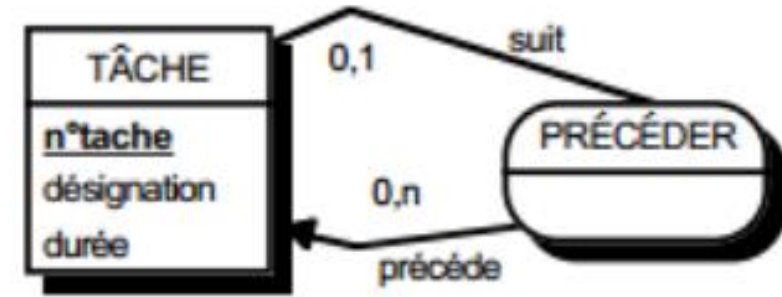


Relationnel dérivé

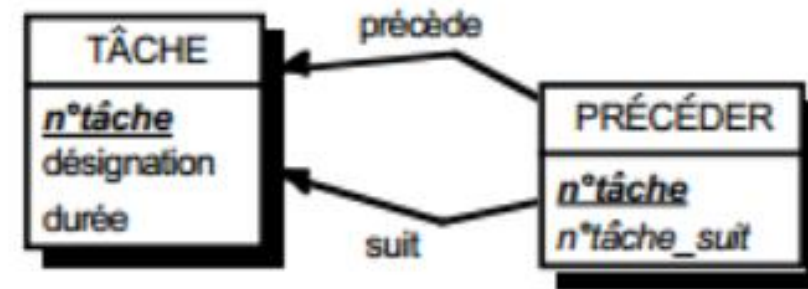
Règles de passage du MCD au MLD

- **Règle numéro 8:**

- Relation binaire réflexive (0,n)-(0,1)
- Dans une première solution, la relation conduit à la création d'une table dans laquelle la clé primaire de la table issue de l'entité se retrouve à la fois comme clé et comme simple attribut.
- On procède à un changement d'appellation de ces attributs dupliqués qui conservent cependant leur domaine de valeurs. Les éventuelles propriétés de cette relation deviennent des attributs de cette table associée.



Entité-Relation



Relationnel dérivé : solution 1

- Solution 1 :

table TACHE (n°tâche, désignation, durée) ;

table PRÉCÉDER (#n°tâche, n°tâche_suit).

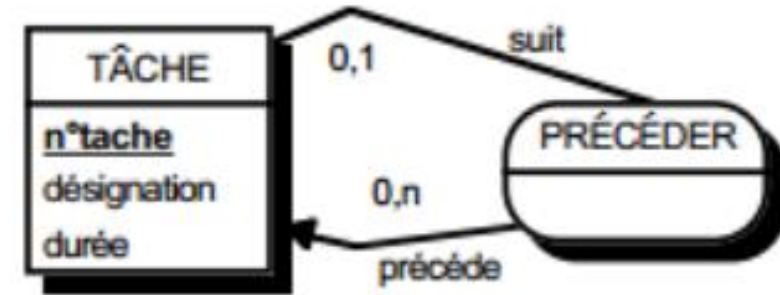
Règles de passage du MCD au MLD

• Règle numéro 8:

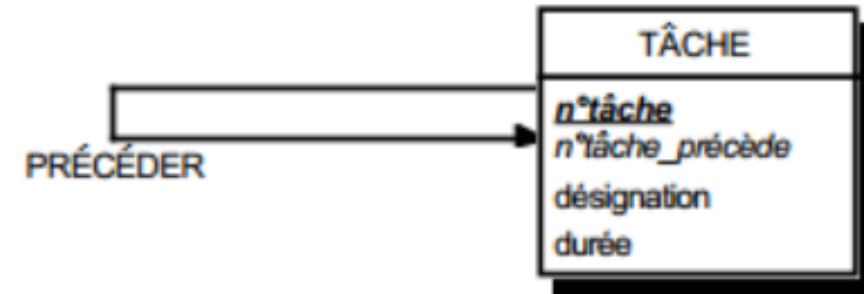
- Relation binaire réflexive (0,n)-(0,1)
- Dans une seconde solution, on duplique la clé de la table issue de l'entité dans la table issue de l'entité à cardinalité (0,1).
- On procède à un changement d'appellation de l'attribut dupliqué qui conserve cependant son domaine de valeurs (n° tâche_précédente). Les éventuelles propriétés de cette relation deviennent des attributs de la table issue de l'entité

- Solution 2 :

table TACHE (n°tâche, #n°tâche_précède, désignation, durée).



Entité-Relation



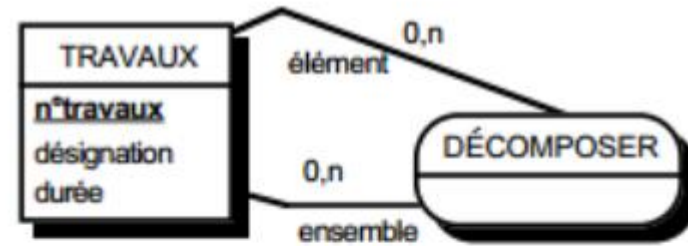
Relationnel dérivé : solution 2

Dans cette seconde solution, la cardinalité (0,1) posera le problème d'accepter des valeurs nulles sur l'attribut migrant. On peut considérer cette seconde solution comme associée à un choix d'optimisation.

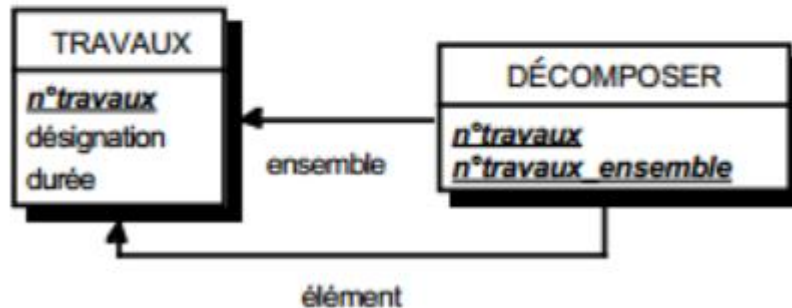
Règles de passage du MCD au MLD

- **Règle numéro 9:**

- Relation binaire réflexive $(*,n)-(*,n)$
- La solution consiste à créer une table de lien ayant comme clé une clé composée de deux fois l'identifiant de l'entité. Les clés étrangères seront qualifiées par rapport aux rôles des pattes de la relation. Les éventuelles propriétés de cette relation deviennent des attributs de cette table issue de la relation



Entité-Relation

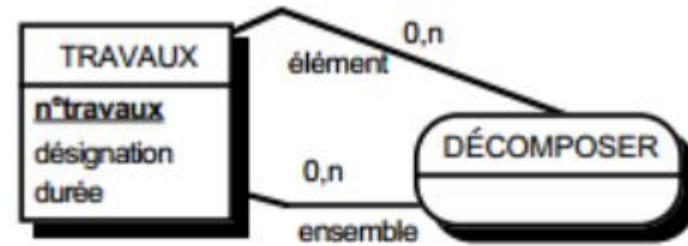


Relationnel dérivé

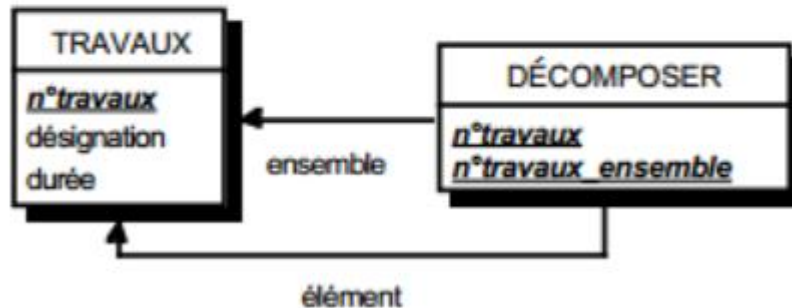
Règles de passage du MCD au MLD

- **Règle numéro 9:**

- Relation binaire réflexive $(*,n)-(*,n)$
- La solution consiste à créer une table de lien ayant comme clé une clé composée de deux fois l'identifiant de l'entité. Les clés étrangères seront qualifiées par rapport aux rôles des pattes de la relation. Les éventuelles propriétés de cette relation deviennent des attributs de cette table issue de la relation



Entité-Relation

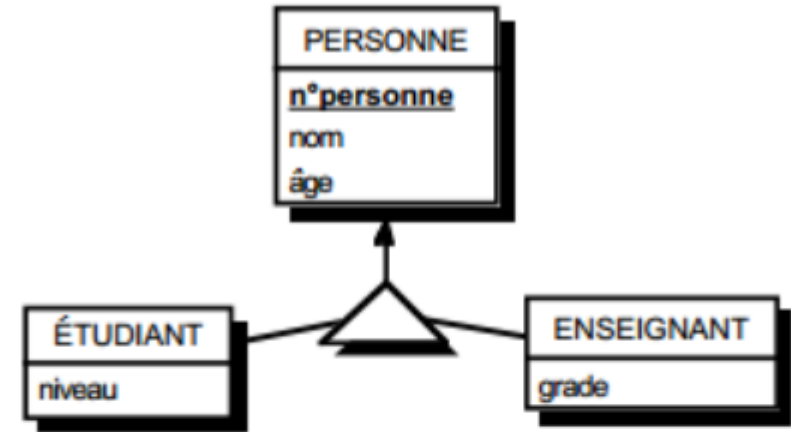


Relationnel dérivé

Règles de passage du MCD au MLD

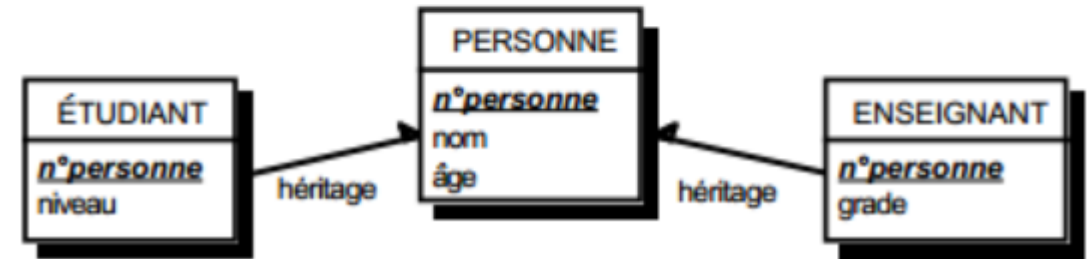
- **Règle numéro 10:**

- Prise en compte des sous-types du MCD
- Plusieurs solutions possibles.
- **Solution 1** : migration de l'identifiant du surtype dans les sous-types



Entité-Relation

- Schémas relationnels associés à la solution 1 :
 - table PERSONNE (n°personne, nom, age) ;
 - table ETUDIANT (#n°personne, niveau) ;
 - table ENSEIGNANT (#n°personne, grade).



Relationnel dérivé

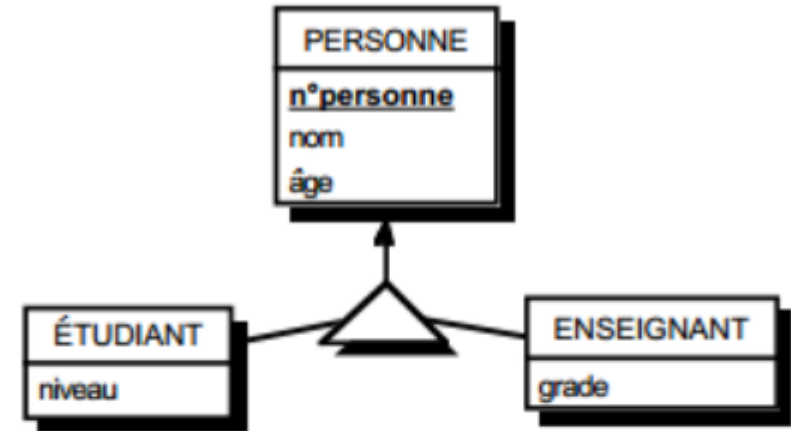
Règles de passage du MCD au MLD

- **Règle numéro 10:**

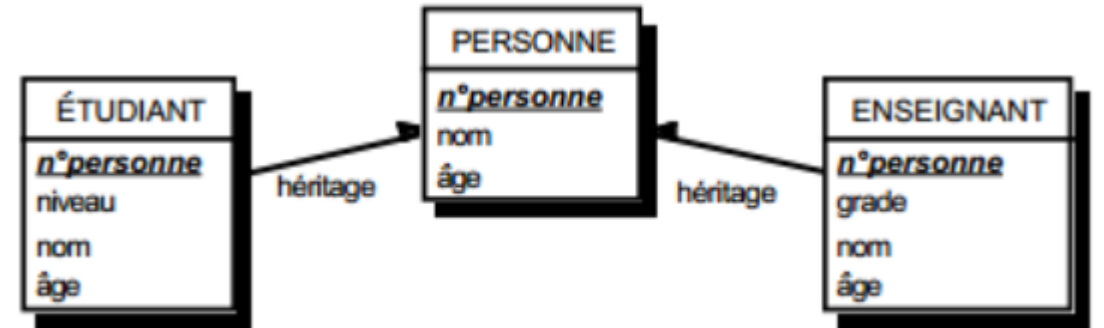
- Prise en compte des sous-types du MCD
- Plusieurs solutions possibles.

- **Solution 2 :**

- Comme dans la solution précédente, on exprime les sous-types par des tables spécifiques.
- Il y a duplication des attributs du surtype dans les sous-types associés, dont la mise à jour simultanée peut être réalisée à travers un mécanisme automatique implémentant l'héritage, par exemple par triggers.



Entité-Relation



Relationnel dérivé

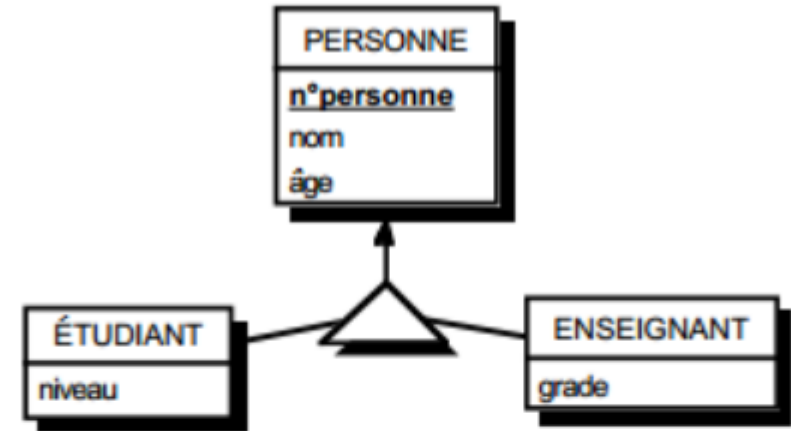
Règles de passage du MCD au MLD

- **Règle numéro 10:**

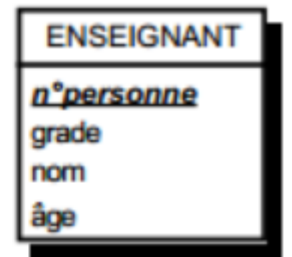
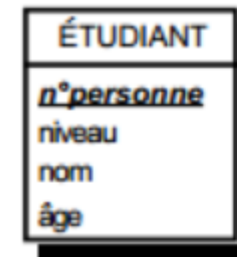
- Prise en compte des sous-types du MCD
- Plusieurs solutions possibles.

- **Solution 3 :**

- on duplique la totalité du contenu du surtype dans les sous-types associés et on supprime le surtype.
- Cette solution n'est pas conseillée dans le cas où il existe, dans le modèle conceptuel (entité-relation), des relations portant sur le surtype



Entité-Relation



Relationnel dérivé

Règles de passage du MCD au MLD

- **Règle numéro 10:**

- Prise en compte des sous-types du MCD
- Plusieurs solutions possibles.

- **Solution 4 :**

- On transfère la totalité des propriétés des sous-types dans la table correspondant au surtype.
- On exprime ensuite les sous-types par des vues relationnelles d'une table PERSONNE globale.
- Pour faciliter l'expression de ces vues, on peut introduire un nouvel attribut « type_personne » dont le domaine est {étudiant, enseignant} :

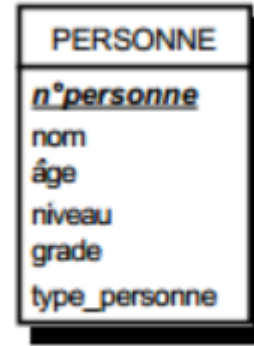
table PERSONNE (n°personne, nom, age, **type_personne**, niveau, grade) ;

- avec comme définition des vues :

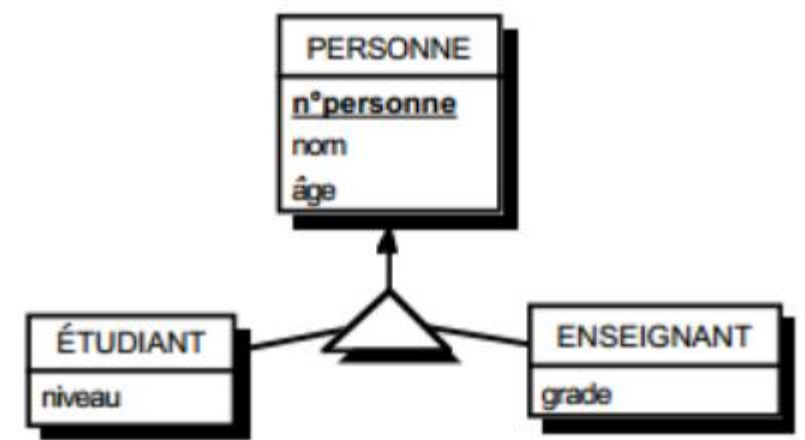
vue PERSONNE_ETUDIANT : vue de PERSONNE (sélection + projection) ;

vue PERSONNE_ENSEIGNANT : vue de PERSONNE (sélection + projection).

```
CREATE VIEW PERSONNE_ETUDIANT AS SELECT n° personne, nom, age, niveau
FROM PERSONNE WHERE PERSONNE.type_personne = 'étudiant';
CREATE VIEW PERSONNE_ENSEIGNANT AS SELECT n° personne, nom, age, grade
FROM PERSONNE WHERE PERSONNE.type_personne = 'enseignant';
```



Relationnel dérivé



Règles de passage du MCD au MLD

- **Règle numéro 11:**

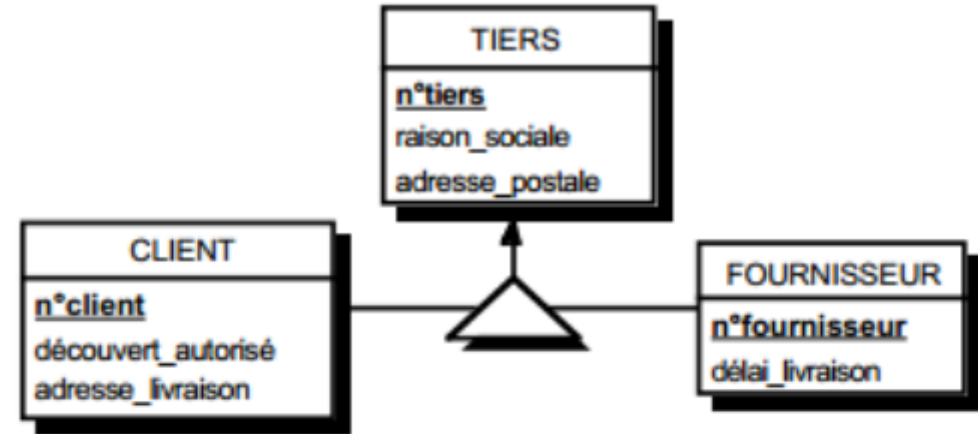
- Prise en compte des sous-types du MCD
- Dans le cas d'une généralisation,
- les sous-types ont leurs propres identifiants. Ainsi, seules les transformations des solutions précédentes 1 et 2 sont possibles

- Schémas relationnels associés à la figure 13.20 :

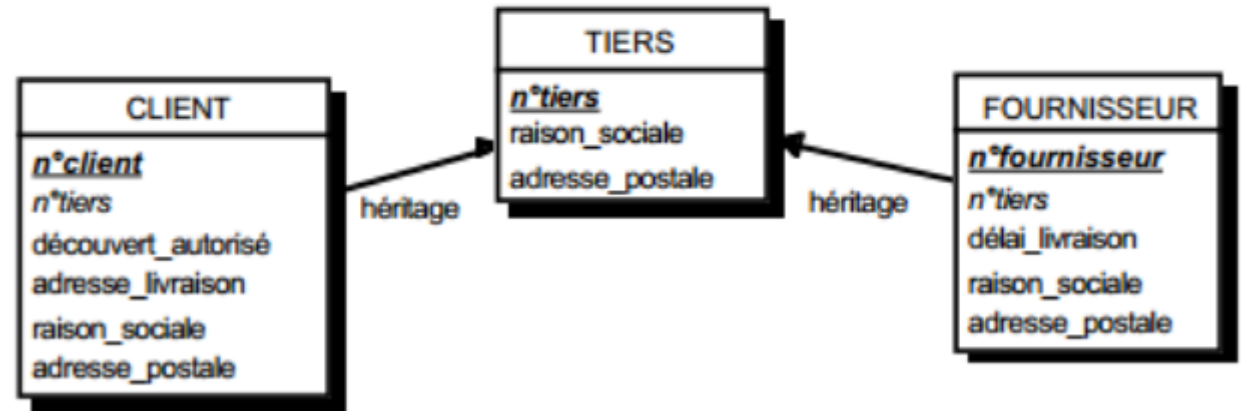
table TIERS (n°tiers, raison_sociale, adresse_postale) ;

table CLIENT (n°client, #n°tiers, découvert_autorisé, adresse_livraison, raison_sociale, adresse_postale) ;

table FOURNISSEUR (n°fournisseur, #n°tiers, délai_livraison, raison_sociale, adresse_postale).



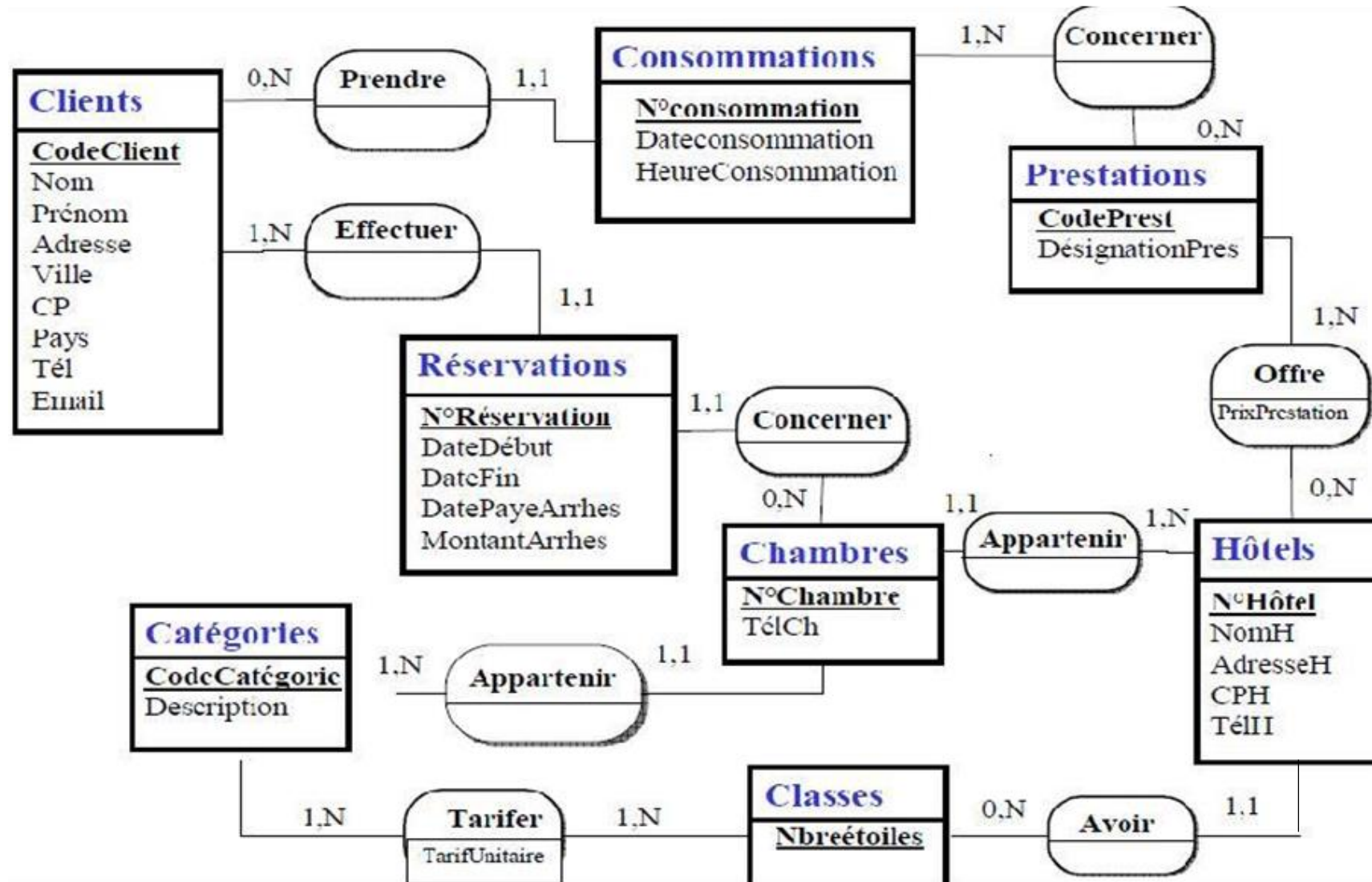
Entité-Relation



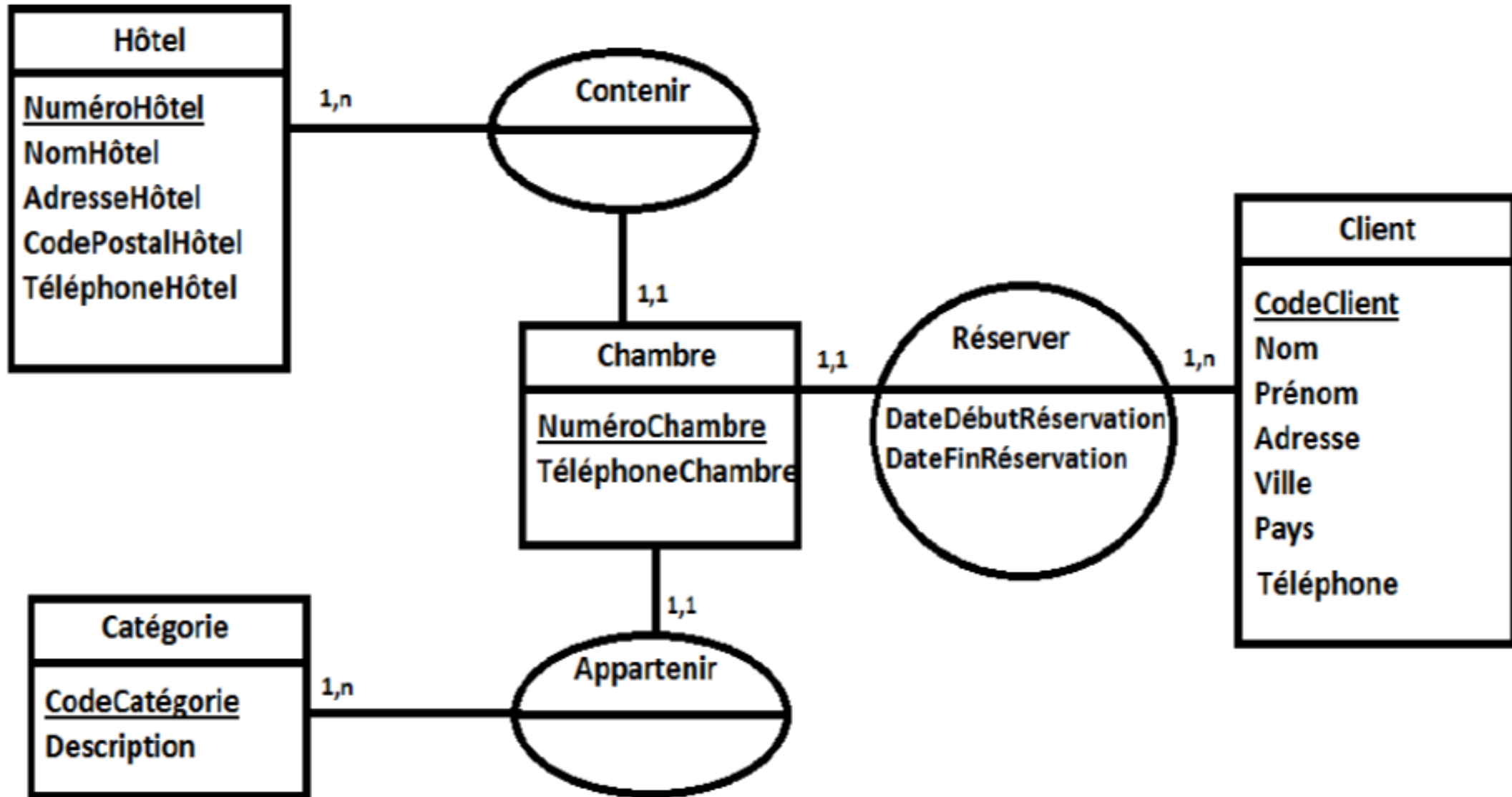
Relationnel dérivé

Atelier: Convertir le MCD en MLD

- Transformez le MCD suivant, qui représente «la gestion d'une chaine Hôtelière» en un modèle relationnel :



Atelier: Convertir le MCD en MLD



Étude de cas de gestion des concours d'apprentis menuisiers

Une école désire gérer la participation de ses apprentis à divers concours de menuiserie. Chaque apprenti est encadré par un tuteur de l'école.

Dans chaque concours, l'apprenti doit réaliser un objet qu'il choisit lui-même. Le jury accorde toujours un nombre de points qui permet d'établir le classement (si 2 apprentis ont le même nombre de points, ils sont ex-æquo).

On désire connaître les concours auxquels ont participé les apprentis, l'objet réalisé, la place et le nombre de points qu'ils ont obtenus.

NB : pour la place, on ne gère que les apprentis de l'école.

Les informations collectées sont : nom de l'apprenti; prénom de l'apprenti; nom du tuteur; prénom du tuteur; nom du concours; lieu du concours; dotation globale du concours ; date du concours; nombre de points obtenus; place obtenue; nom objet réalisé.

Normalisation

- Dans une base de données relationnelle, une forme normale désigne un type de relation particulier entre les entités.
- Le but essentiel de la normalisation est d'éviter les anomalies transactionnelles pouvant découler d'une mauvaise modélisation des données et ainsi éviter un certain nombre de problèmes potentiels tels que les anomalies de lecture, les anomalies d'écriture, la redondance des données et la contre-performance.
- La normalisation des modèles de données permet de vérifier la robustesse de leur conception pour améliorer la modélisation (et donc obtenir une meilleure représentation) et faciliter la mémorisation des données en évitant la redondance et les problèmes sous-jacents de mise à jour ou de cohérence. La normalisation s'applique à toutes les entités et aux relations porteuses de propriétés.

Première forme normale (1FN)

- Une relation est en première forme normale si, et seulement si, tout attribut contient une valeur atomique (non multiples, non composées).
- Par exemple, le pseudo schéma de relation **Personne** n'est pas en première forme normale.

Personne(num-personne, nom, prénom, rue, ville, prénoms-enfants)

Il faut le décomposer en :

Personne(num-personne, nom, prénom, rue, ville)

Prénoms-enfants(#num-personne, # num-prénom)

Prénoms(num-prénom, prénom)

Deuxième forme normale (2FN)

- Une relation est en deuxième forme normale si, et seulement si, elle est en première forme normale et si toutes les dépendances fonctionnelles entre la clé et les autres attributs sont élémentaires.
- Autrement dit, une relation est en deuxième forme normale si, et seulement si, elle est en première forme normale et si tout attribut n'appartenant pas à la clé ne dépend pas que d'une partie de la clé.
- Une relation peut être en deuxième forme normale par rapport à une de ses clés candidates et ne pas l'être par rapport à une autre. Une relation avec une clé primaire réduite à un seul attribut est, par définition, forcément en deuxième forme normale.

Deuxième forme normale (2FN)

- Soit, par exemple, le schéma de relation suivant :

***Commande* (*Num-Commande*, *Num-Client*, *Quantité*, *Nom_Client*, *Date*)**

- Cette relation indique qu'un client (identifié par Num-Client) a passé une commande (identifiée par Num-Commande). Elle est bien en première forme normale.
- Par contre, les attributs ***Quantité***, ***Date*** et ***Nom_Client*** ne dépendent pas de
- ***Num-Commande***, et de ***Num-Client***.

Num-Client, Num-Commande -\-> Quantité, Date, Nom_Client

- Cette relation n'est donc pas en deuxième forme normale. Une solution simple pour la normaliser est de la remplacer par :

***Client* (*Num-Client*, *Nom_Client*)**

***Commande* (*Num-Commande*, *Quantité*, *Date*, *#Num-Client*)**

Troisième forme normale (3FN)

- Une relation est en troisième forme normale si, et seulement si, elle est en deuxième forme normale et si toutes les dépendances fonctionnelles entre la clé et les autres attributs sont élémentaires et directes.
- Autrement dit, une relation est en troisième forme normale si, et seulement si, elle est en deuxième forme normale et si tout attribut n'appartenant pas à la clé ne dépend pas d'un attribut non-clé. Soit, par exemple, le schéma de relation suivant :

CommandeLivre(Num-Commande, #Num-Client, *Titre -Livre*, Auteur, Quantité, Prix)

- Cette relation est bien en deuxième forme normale. Par contre, les attributs **Auteur** et **Prix** dépendent de l'attribut **Titre -Livre**.

***Titre -Livre* > Auteur, Prix**

- La relation n'est donc pas en troisième forme normale. Pour la normaliser, il faut la décomposer de la manière suivante :

CommandeLivre(Num-Commande, #Num-Client, #*Titre -Livre*, Quantité)

Livre(*Titre -Livre*, Auteur, Prix)

Forme normale de BOYCE-CODD

- Une relation est en forme normale de BOYCE-CODD (BCNF) si, et seulement si, elle est en troisième forme normale et si les seules dépendances fonctionnelles élémentaires sont celles dans lesquelles une clé détermine un attribut.
- Cette forme normale permet de renforcer certaines lacunes de la troisième forme normale. Soit, par exemple, le schéma relationnel décrivant l'enseignement d'une matière donnée à une classe par un enseignant :
 - Matière(nom-matière)
 - Classe(num-classe)
 - Enseignant(nom-enseignant)
 - Enseignement(nom-enseignant, num-classe, nom-matière)

Forme normale de BOYCE-CODD

- Supposons, de plus, qu'une matière n'est enseignée qu'une seule fois dans une classe et que par un seul enseignant, et qu'un enseignant n'enseigne qu'une seule matière. Chacune des relations respecte bien la troisième forme normale. Cependant, dans la relation Enseignement, nous avons les dépendances fonctionnelles élémentaires suivantes :

1. nom-matière, num-classe -> nom-enseignant

2. nom-enseignant -> nom-matière

Il existe donc des dépendances fonctionnelles élémentaires dont la source n'est pas la clé de la relation.

Forme normale de BOYCE-CODD

nom-enseignant	num-classe	nom-matière
George	5	Physique
George	6	Physique
George	7	Physique
George	8	Physique
Michael	5	Mathématiques
Michael	6	Mathématiques
Michael	7	Mathématiques
Michael	8	Mathématiques

Exemple de relation présentant une redondance due au non respect de la forme normale de BOYCE-CODD.

Forme normale de BOYCE-CODD

- Le non respect de la forme normale de BOYCE-CODD entraîne une redondance illustrée par la table pour chaque nom-enseignant identifiant un enseignant, il faut répéter le nom-matière identifiant la matière qu'il enseigne.
- Pour normaliser la relation Enseignement, il faut la décomposer pour aboutir au schéma relationnel suivant :
 - Matière(nom-matière)
 - Classe(num-classe)
 - Enseignant(nom-enseignant, #nom-matière)
 - Enseigner(#num-classe, #nom-matière, #nom-enseignant)

Exercices:

- Soit le schéma de BD Relationnel suivant :

COMMANDE(numCom,datCom, numCli, nomCli,adrCli,RédCom)

ARTICLES (numCom, numArt, nomArt, desArt, qtéCom, PUArt, PT, RédAcc)

- RG1 : Une commande est constituée de plusieurs articles.
- RG2 : Un article commandé l'est pour une certaine quantité et pour un prix total (PT) donné.

Ce schéma est-il en forme normale ? Pourquoi ?

Appliquer les 4 règles de normalisation sur le modèle pour vérification

Quantification du MLD

- **Présentation des schémas relationnels associés au modèle logique des données validé**

- Chaque table du MLD validé en ajoutant la nature des attributs ainsi que la taille de chaque attribut.

1. Agent : (# matr : char [6] ; # codgr : char [3] ; # codser : char [20] ; # coddir : char [10] ; nomag : char [20] ; postnom agr : char [20] ; sexe : char [1]).

2. Ouvrage : (# codou : char [6] ; # codray : char [2] ; # codaut : char [6] ; titou : char [50]; pages : char [4]) ; Année d'édition [4] .

3. Rayon : (# codray : char [6] ; Design rayon : char [20] ;

4. Auteur : (# codaut : char [6] ; nomaut : char [30] ;

5. Concerner : (#numempr: char [6] ; codouv: char [6] ;

6. Service : (# codser : char [10] ; Désiser: char [50] ;

7. Direction : (# coddir: char [10] ; Désidir: char [50] ;

8. maison d'édition : (# codmais : char [6] ; nommais: char [20] ;

9. Emprunt : (#numempr: char [6] ; matr: char [6] ; datedebut emprunt : char [10] ; date fin empr : char [10] ;

Quantification du MLD

- **Quantification du volume des données au niveau logique**
- Calcul du volume des tables

Tables	Nombre d'occurrence	Taille	Volume
Agent	1800	121	217800
Ouvrage	16000	64	1024000
Rayon	5	26	130
Auteur	9500	26	247000
Maison d'édition	3000	26	78000
Service	75	60	4500
Direction	26	60	1560
Concerner	11024000	12	132288000
Emprunt	1000	32	32000
Total volume des tables			133892990

Tables	Index	Nombre d'occurrence	Taille	Volume
Agent	# matr	1800	6	10800
	codser	1800	6	10800
	coddir	1800	6	10800
Ouvrage	# codou	16000	6	96000
	codray	16000	6	96000
	codmais	16000	6	96000
	Codaut.	16000	6	96000
Rayon	# codray	5	6	30
Auteur	# codaut	9500	6	57000
Maison d'éd.	# codmais.	3000	6	18000
Service	# codser	75	50	3750
Direction	# coddir	26	50	1300
Concerner	# Num empr.	11024000	6	66144000
	Cod ouvr.	11024000	6	66144000
Emprunt	#Numempr.	1000	6	6000
	matr	1000	6	6000
Volume total des index				132796480

- Calcul du volume des index

Quantification du MLD

- Volume total de la base de données
- Le volume total de la base de données est égal au volume des tables plus le volume des index.

$$\begin{aligned} V.\text{total (bdd)} &= V.\text{total} + V.\text{index} \\ &= (133892990 + 132796480) \times 1,5 \\ &= 400034205 \\ &= 381.50\text{Mo} \end{aligned}$$

1, 5 : coefficient

Il est recommandé de trouver le volume réel de la base de données en utilisant un coefficient qui varie entre 2 et 5 afin de prendre en compte certains imprévus en terme d'espace mémoire.

Etudes de Cas

- Un office HLM gère des cités composées de bâtiments, eux-mêmes composés d'appartements. Ces appartements sont libres ou loués à des locataires. Chaque cité est sous la responsabilité d'un gardien. Chaque appartement appartient à un type de logement (F1 à F6) correspondant à un certain nombre de caractéristiques précises. Des opérations d'entretien (estimation maximal de 10000 par an) peuvent être effectuées sur les appartements par au plus 10 artisans (estimation maximal de 2000 par an) . Chaque année, une estimation de 7000 opérations d'entretien peuvent être effectuées par au moins un artisan.

Les règles de gestion sont les suivantes :

- Une cité n'est gardée que par un seul gardien
- Un bâtiment n'appartient qu'à une seule cité
- Une opération est relative à un seul appartement
- Un appartement , s'il est loué, n'a qu'un seul locataire

Définir : MCD - MOD - MLD