

Les systèmes d'information

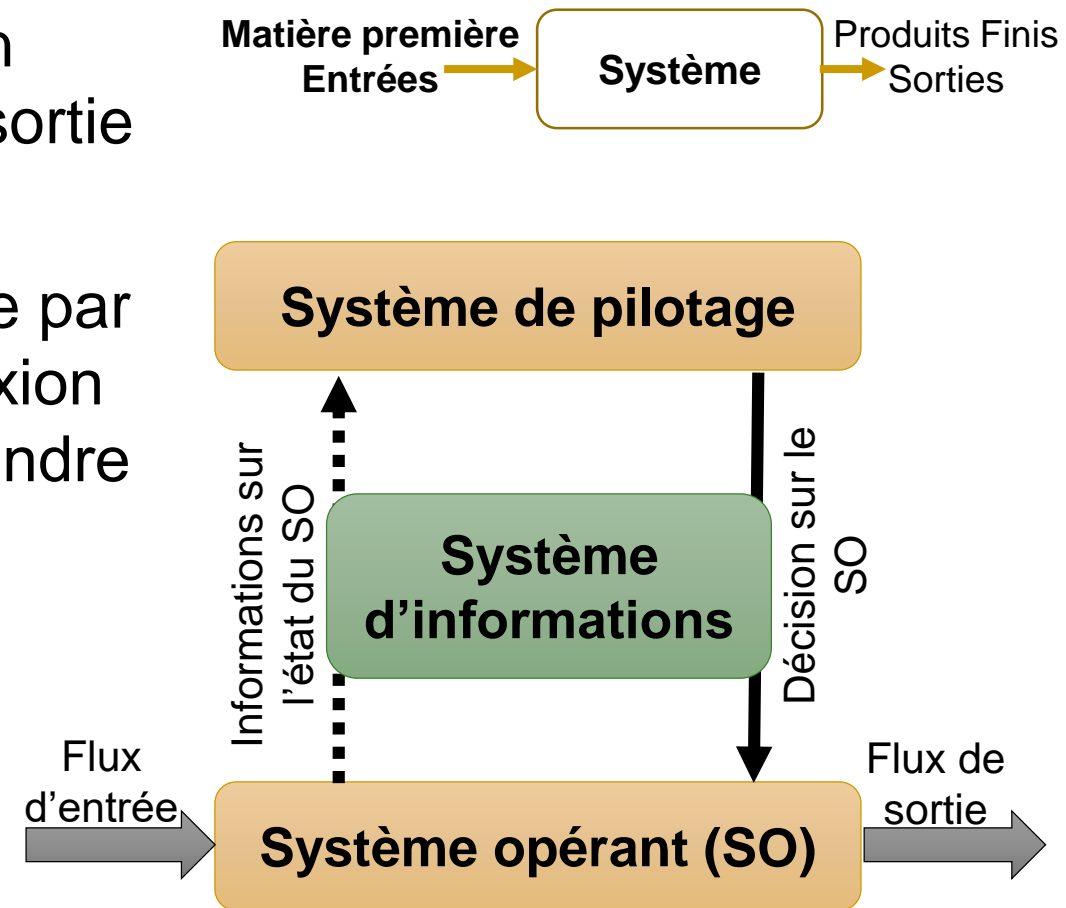
Jamal BAKKAS

Université Cadi Ayyad

Ecole supérieur de Technologie-Safi

Les systèmes de base d'une organisation

- ❑ **Système:** on peut définir un système comme étant un ensemble d'éléments en interaction transformant un flux d'entrée en un flux de sortie
- ❑ **Organisation :** Une organisation (entreprise par exemple) peut être vue comme l'interconnexion de trois systèmes qui collaborent afin d'atteindre les objectifs :
 - ❑ Le système opérant,
 - ❑ Le système d'informations
 - ❑ Et le système de pilotage.



Les 3 systèmes de base d'une organisation

- ❑ **Le système opérant** : appelé aussi système de production. C'est l'ensemble de ressources et moyens matériels ou humains couvrant l'activité de l'entreprise. L'objectif principal de ce système est la production.
- ❑ **Le système de pilotage** : appelé aussi système de commande ou encore de décision. Il se situe à la tête de toute organisation, son objectif est de piloter le système opérant. Il fixe les objectifs et prend les décisions.
 - ❑ Les décisions stratégiques(long terme): définition des politiques ou moyens à mettre en œuvre. Exemple lancement d'un nouveau produit
 - ❑ Les décisions tactiques (moyen terme): utilisation efficace des ressources disponibles pour atteindre les objectifs fixés. Exemple lancer une promotion
 - ❑ Les décisions opérationnelles (court terme): Exemple gestion de stock
- ❑ **Le système d'information** : Il intervient entre les deux autres, ses objectifs principaux:
 - ❑ Transmission au système de pilotage des informations nécessaires à sa mission de pilote.
 - ❑ Transmission au système opérant des directives fournies par le système de pilotage.

Systeme d'informations

❑ Définition:

« Un SI est un ensemble organisé de ressources : matériel, logiciel, personnel, données, procédures... permettant d'acquérir, de traiter, de stocker des informations (sous forme de données, textes, images, sons, etc...) dans et entre des organisations ». R. REIX, 2005.

- ❑ Le SI est le centre nerveux des entreprises.
- ❑ Les flux entrant et sortant d'un SI et constitué exclusivement des informations

❑ Système informatique

Le système informatique est la partie informatique du système d'information, composée de matériels, logiciels, réseaux et procédures d'utilisation.

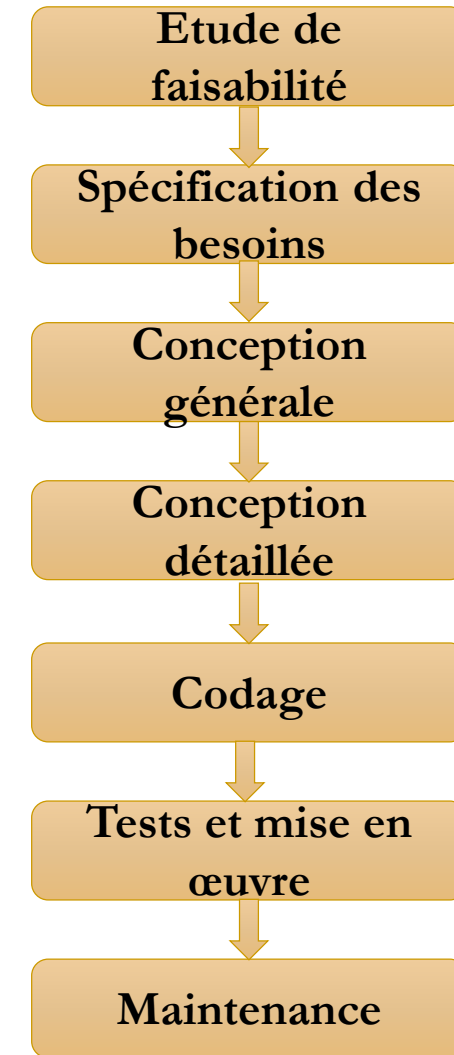
Système d'informations

- ❑ Le système d'information permet:
 - ❑ Stocker l'information (conservation de l'information pour des besoins ultérieurs),
 - ❑ Traiter l'information (rapprochements, calculs, comparaisons),
 - ❑ Diffuser l'information (accès à la mémoire, échange entre acteurs)
 - ❑ Produire les informations légales réclamées par l'environnement
 - ❑ Déclencher les décisions programmées
 - ❑ Fournir des informations aux décideurs pour aider à la prise de décisions non programmées
 - ❑ Coordonner les tâches en assurant les communications au sein du système organisationnel

Cycle de vie d'un logiciel

Pour obtenir un logiciel de qualité, il faut en maîtriser le processus d'élaboration appelé «cycle de vie d'un logiciel ». Ce processus est composé de différentes étapes formelles dont on doit contrôler la succession :

- a. **Etude de faisabilité** : On détermine si le logiciel à développer vaut la peine d'être mis en œuvre suivant le besoin du marché.
- b. **Spécification des besoins fonctionnels** : On détermine les fonctionnalités que doit posséder le logiciel à savoir les exigences des utilisateurs (collecte des exigences) ainsi que les tâches et les structures à informatiser (analyse du domaine).
- c. **Conception générale** : On détermine la structure du système (le modèle du système) à savoir les différentes parties du système et la façon dont les différentes parties du système interagissent entre elles
- d. **Conception détaillée** : On écrit les algorithmes de chaque partie du système
- e. **Codage** : L'implantation des algorithmes.
- f. **Tests et mise en œuvre** : On essaye le logiciel sur des données d'exemple pour s'assurer qu'il fonctionne correctement. Puis on le fournit au client.
- g. **Maintenance** : C'est la mise à jour du logiciel et son amélioration pour assurer de son opérabilité.



La méthode Merise

- ❑ Merise a été très utilisée dans les années 1970 et 1980 pour l'informatisation massive des organisations.
- ❑ Développée par le ministère de l'industrie français, c'est une méthode qui se base sur la décomposition du problème étudié en plusieurs sous problèmes puis la décomposition de chaque sous problème en d'autres sous problèmes jusqu'à obtention des problèmes élémentaires faciles à codifier (fonctionnalités) et ceci afin de vaincre la complexité du problème d'origine.
- ❑ MERISE est une méthode d'analyse et de conception des systèmes d'information basée sur le principe de la séparation des données et des traitements.
- ❑ Elle possède plusieurs modèles qui sont répartis sur 3 niveaux:
 - ❑ Le niveau conceptuel (quoi)
 - ❑ Le niveau Organisationnel (qui quant où)
 - ❑ Le niveau logique (comment)
 - ❑ Le niveau physique

LES NIVEAUX D'ABSTRACTION

La méthode merise propose un modèle pour chaque niveau d'abstraction et pour chaque volet (Données, traitements).

- ❑ **Niveau conceptuel** : Représentation des informations et des traitements sans tenir compte de l'organisation des données ou bien des moyens matériels. On répond à la question **QUOI ?** en termes de données et de traitements.
- ❑ **Niveau organisation** : Définition de l'organisation des ressources humaines et matérielles. On intègre les détails de l'organisation en termes de ressources humaines (postes de travail), de lieu et de temps. On répond aux questions : **QUI** fait **QUOI ? QUAND** et **OU ?**
- ❑ **Niveau logique** : On intègre les moyens et ressources informatiques sans tenir compte de leurs caractéristiques techniques. Décrit :
 - ❑ Le schéma de la base de données (relationnel, hiérarchique ou réseau) : les caractéristiques du mode de gestion des données :
 - ❑ La répartition des Données sur les différentes unités de stockage :
- ❑ **Niveau physique** : On intègre les choix techniques. Définit complètement:
 - ❑ Les fichiers, les programmes
 - ❑ L'implantation physique des données et des traitements
 - ❑ Les ressources à utiliser
 - ❑ Les modalités de fonctionnement

LES NIVEAUX D'ABSTRACTION

Niveau	Données	Traitements	Exemple
Conceptuel	MCD	MCT	Le client effectue une demande de service à la compagnie d'assurance pour assurer sa voiture. Il demande un devis.
Organisationnel	MOD	MOT	Un client effectue une demande au guichet de l'agence de son choix, pour assurer sa voiture. Un agent de service concerné, cherche si le client est fiable (consultation d'un fichier central inter assurances), prend contact par téléphone pour le client (après 12 heures) afin d'examiner plus précisément ses besoins et établir un devis.
Logique	MLD	MLT	Les agences structurent leurs données via le modèle relationnel, la base de données relationnelle est choisie.
Physique	MPD	MPT	Le fichier central inter assurances est accessible par internet. Les agences sont connectées au siège de la compagnie par liaison ADSL sécurisée. Chaque agence dispose de microordinateurs et peut traiter ses données en local grâce au SGBD Oracle.

Les règles de gestion

- ❑ Après le recueil des besoins des futurs utilisateurs de l'application. Il faut établir les règles de gestion des données à conserver. Ce sont des règles propres à l'organisation étudiée
- ❑ **Exemple :** On vous demande de modéliser les données nécessaires à une gestion de projets et on vous dit, entre autres, qu'« *il y a toujours une personne qui est nommée chef de projet pour un projet et que plusieurs personnes travaillent sur un projet* ».
- ❑ **Exemple de questions à se poser:**
 - ❑ est-ce qu'une personne est obligatoirement chef de projet au moins une fois ?
 - ❑ est-ce que toute personne enregistrée dans le système travaille sur au moins un projet ?
 - ❑ est-ce qu'une personne ne peut travailler que sur un seul projet à la fois ?
- ❑ **Règles de gestion à extraire**
 - 1) Un projet est dirigé par une seule personne (nommée « chef de projet » pour ce projet) et une personne peut diriger plusieurs projets (en étant nommée « chef de projet » pour chacun d'eux).
 - 2) Un projet fait travailler de une à plusieurs personnes et une personne peut travailler sur plusieurs projets.

Le dictionnaire des données

- ❑ Le dictionnaire des données est un document qui regroupe toutes les données à conserver dans base de données (et qui figureront donc dans le MCD). Pour chaque donnée, il indique :
 - ❑ **Code mnémonique** : il s'agit d'un libellé désignant une donnée (par exemple «*titre_1*» pour le titre d'un livre)
 - ❑ **Désignation** : il s'agit d'une mention décrivant ce à quoi la donnée correspond (par exemple «*titre du livre*»)
 - ❑ **Nature** : Élémentaire, Calculée, ou Paramètre ;
 - ❑ **Type de donnée** :
 - ❑ **Alphabétique** : lorsque la donnée est uniquement composée de caractères alphabétiques (de 'A' à 'Z' et de 'a' à 'z')
 - ❑ **Numérique** : lorsque la donnée est composée uniquement de nombres (entiers ou réels)
 - ❑ **Alphanumérique** : lorsque la donnée peut être composée à la fois de caractères alphabétiques et numériques
 - ❑ **Date** : lorsque la donnée est une date (au format AAAA-MM-JJ)
 - ❑ **Booléen** : Vrai ou Faux
 - ❑ **Taille** : elle s'exprime en nombre de caractères ou de chiffres
 - ❑ Et parfois des **Remarques** ou **Observations** complémentaires (par exemple si une donnée est supérieure à 0, ...etc).
- ❑ Le dictionnaire des données se fait en deux étapes :
 - 1) Etablissement du dictionnaire complet (toutes les données y sont répertoriés)
 - 2) Etablissement du dictionnaire réduit : les données calculées sont éliminées.

Le dictionnaire des données

Exemple:

Extraire le dictionnaire des données figurant à la facture ci-contre

Restaurant Le Safio

17 Rue Sidi Bouzid - Safi

Tel: 0524625247



Facture

NumCli 156
Nom: JAZI
Prénom: SAMIR
Adresse: 45, Rue agadir
Code postal 20000
Ville Casablanca
Téléphone: 06 65 42.00.45

Date	CodeArticle	Désignation	Qté	Prix	Total
14/07/2012	RP003	Repas "poisson"	4	40	160
15/07/2012	B001	Café	1	10	10
15/07/2012	GL004	Glace "Magnum"	2	15	30
16/07/2012	BG020	Baguette	1	1,2	1,2
				Total dû:	201,2

Code mnémonique	Désignation	Nature	Type de donnée	Taille	Remarque
nomR	Nom restaurant	Paramètre	Alphanumérique	30	
adresseR	Adresse restaurant	Paramètre	Alphanumérique	60	
TelR	Téléphone restaurant	Paramètre	Alphanumérique	10	
numCli	Numéro client	Elémentaire	Numérique		
nomCli	Nom du client	Elémentaire	Alphanumérique	30	
prenomCli	Prénom client	Elémentaire	Alphanumérique	30	
adresseCli	Adresse client	Elémentaire	Alphanumérique	60	
codePostalCli	Code postal client	Elémentaire	Alphanumérique		
villeCli	Ville client	Elémentaire	Alphanumérique	20	
telCli	Téléphone client	Elémentaire	Alphanumérique	10	
date	Date consommation	Elémentaire	Date		
codeArticle	Code article	Elémentaire	Numérique		
designation	Désignation	Elémentaire	Alphanumérique	50	
PU	Prix unitaire	Elémentaire	Numérique		
Qte	Quantité	Elémentaire	Numérique		
TotalLigne	Total par ligne	calculée	Numérique		Qte*PU
TotalFacture	Total à payer	Calculée	Numérique		Somme des TotalLigne

Le dictionnaire des données

Exercice d'application :

Extraire le dictionnaire des données figurant à la facture ci-contre

Entreprise AU BON BEURRE 7, rue de la vache joyeuse 34200 SÈTE		FACTURE		
N° client : 152 Nom client : Casino 7 montée St Clair 34200 SÈTE				
		N° facture : 1258 Date facture : 14/08/1999		
Référence	Désignation	Prix unitaire	Quantité	Montant brut
X25	Plaquette beurre 250 g	4,50	500	2 250,00
X32	Crème UHT 125 g	3	400	1 200,00
			Total HT	3 450,00
			Total TVA	189,75
			TTC	3 639,75

Graphe des dépendances fonctionnelles (GDF)

❑ Dépendance fonctionnelle

- ❑ Soit deux propriétés (ou données) P1 et P2. On dit que P1 et P2 sont reliées par une **dépendance fonctionnelle** (ou p2 dépend fonctionnellement de p1) si et seulement si une **occurrence** (ou valeur) de P1 permet de connaître une et une seule occurrence de P2.

- ❑ Cette dépendance est représentée comme ceci :

$$P1 \rightarrow P2$$

- ❑ On dit que P1 est la **source** de la DF et que P2 en est le **but**.

❑ Exemple :

- ❑ Si on connaît le numéro d'inscription d'un étudiant, on détermine son nom (un et un seul).
Le nom de l'étudiant dépend de son numéro d'inscription.

On peut donc écrire: **numInscEtud \rightarrow nomEtud**

Graphe des dépendances fonctionnelles (GDF)

❑ Dépendances fonctionnelles composées :

Une dépendance fonctionnelle qui comporte plusieurs attributs est dite composée.

❑ Exemple:

Cne, codeMatiere \rightarrow noteDS

Numéro coureur, Numéro de course \rightarrow Temps

Connaissant le n° de coureur et le n° de la course, nous connaissons de façon certaine le temps chronométré d'un coureur précis sur une course précise.

❑ Dépendances fonctionnelles élémentaire:

Une dépendance fonctionnelle $A \rightarrow B$ est élémentaire s'il n'existe pas une donnée C, sous-ensemble de A, décrivant une dépendance fonctionnelle avec B ($C \rightarrow B$).

❑ Exemple:

RéférenceProduit \rightarrow Désignation

NuméroCommande, RéférenceProduit \rightarrow Quantité

NuméroCommande, RéférenceProduit \rightarrow Désignation

La troisième dépendance fonctionnelle n'est élémentaire car il existe à l'intérieur d'elle: **RéférenceProduit \rightarrow Désignation** qui était déjà une dépendance fonctionnelle élémentaire.

Graphe des dépendances fonctionnelles (GDF)

□ Dépendances fonctionnelles élémentaire directe :

On dit que la dépendance fonctionnelle $A \rightarrow B$ est directe s'il n'existe aucune propriété C tel que l'on puisse avoir $A \rightarrow C$ et $C \rightarrow B$. En d'autres termes, cela signifie que la dépendance entre A et B ne peut être obtenue par transivité.

□ Exemple:

- **NumEleve \rightarrow codeFiliere**
- **codeFiliere \rightarrow nomFiliere**
- **NumEleve \rightarrow nomFiliere**

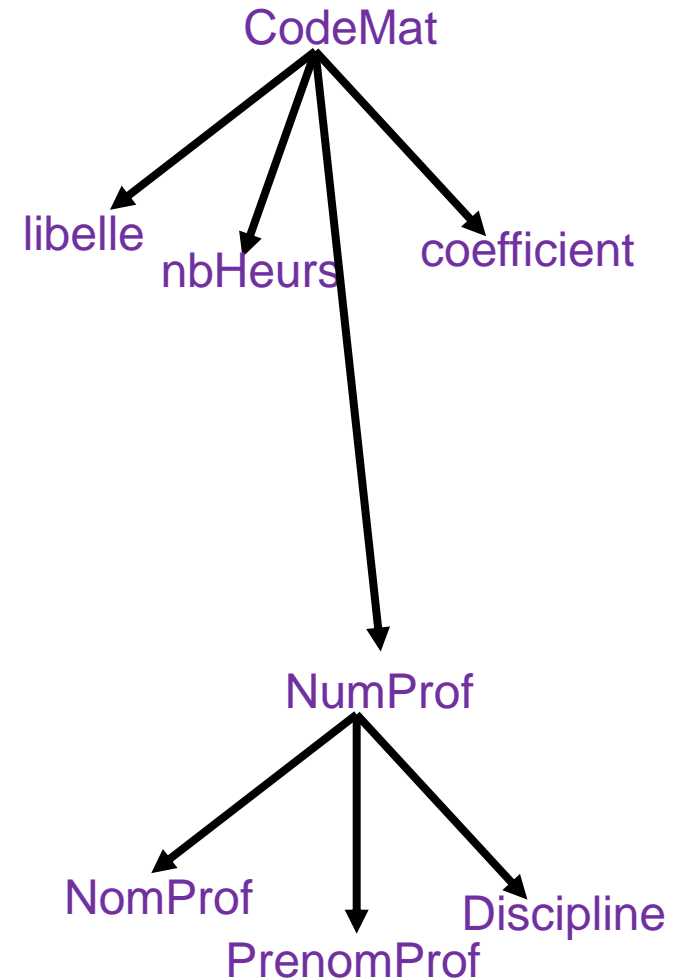
La troisième dépendance fonctionnelle n'est pas directe car nous pourrions écrire:

NumEleve \rightarrow codeFiliere \rightarrow nomFiliere

Graphe des dépendances fonctionnelles (GDF)

❑ **Exemple :** Soit les faits suivants :

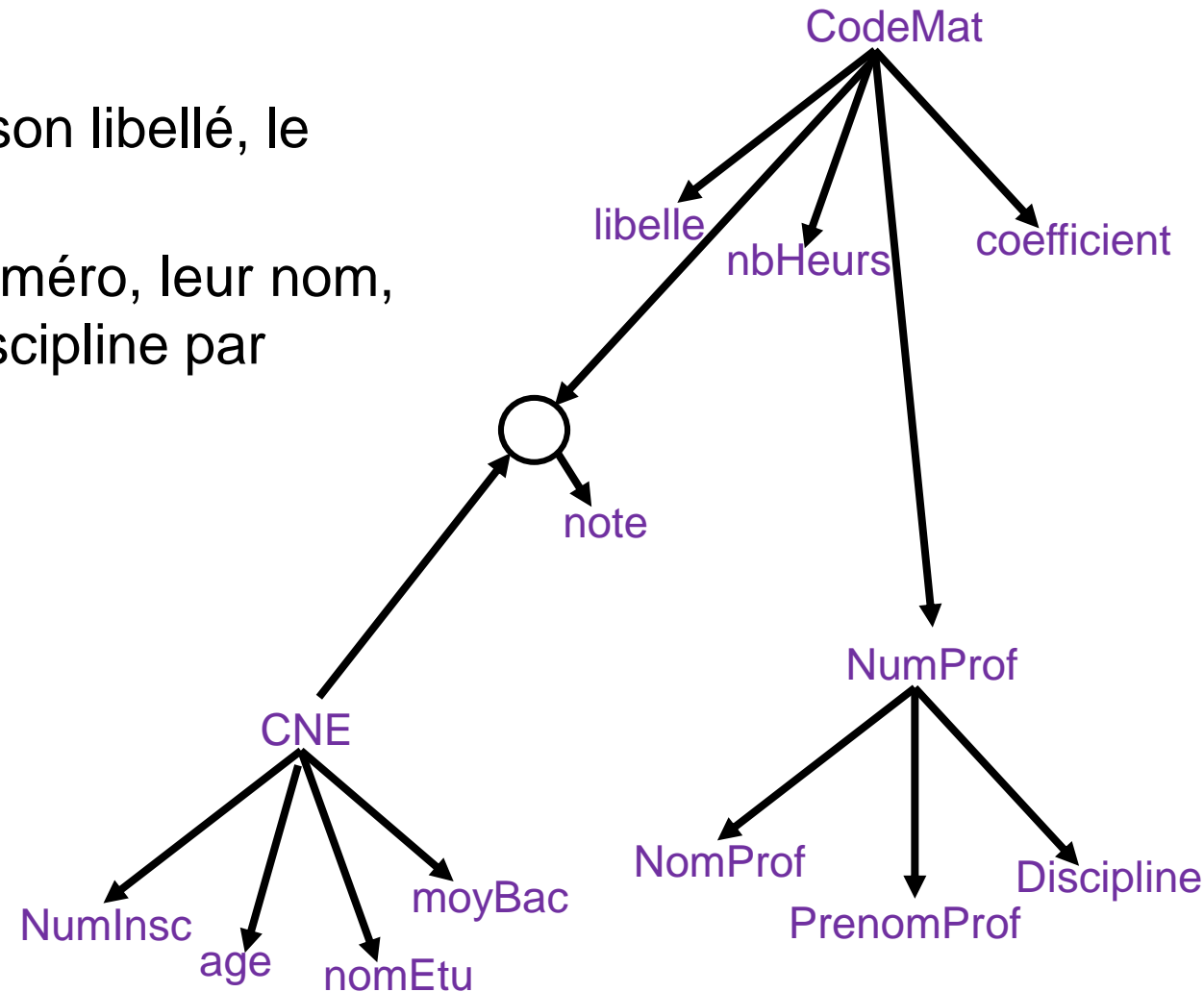
- ❑ Une matière est caractérisé par son code, son libellé, le nombre d'heurs, et son coefficient
- ❑ les professeurs sont caractérisés par un numéro, leur nom, leur prénom et leur discipline (une seule discipline par professeur)
- ❑ Une matière est enseigné par un et un seul professeur



Graphe des dépendances fonctionnelles (GDF)

❑ **Exemple** : Soit les faits suivants :

- ❑ Une matière est caractérisé par son code, son libellé, le nombre d'heurs, et son coefficient
- ❑ les professeurs sont caractérisés par un numéro, leur nom, leur prénom et leur discipline (une seule discipline par professeur)
- ❑ Une matière est enseigné par un et un seul professeur
- ❑ Un étudiant est caractérisé par son numéro d'inscription, le CNE, son nom, son age, ainsi que son moyen de bac
- ❑ Un étudiant a une note pour chaque matière



Le modèle conceptuel de données (MCD)

- ❑ Le modèle conceptuel de données permet d'écrire de façon formelle les données qui seront utilisées par le système d'information.
- ❑ Il s'agit donc d'une représentation des données, facilement compréhensible.
- ❑ Les éléments de base d'un MCD :
 - ❑ Les propriétés.
 - ❑ Les entités.
 - ❑ Les associations.

Le modèle conceptuel de données (MCD)

❑ Les concepts de base

- ❑ **Entités:** Une entité peut être défini comme une conceptualisation des objets concrets du monde réel perçu.
- ❑ Une entité possède un nom, des propriétés, et un identifiant
- ❑ Exemple:
 - ❑ Personne
 - ❑ Voiture
- ❑ **Propriétés:** une propriété est une donnée élémentaire d'une entité. C'est une caractéristique de l'entité
- ❑ Exemple:
 - ❑ Personne (cin, nom, prenom, age, ...)
 - ❑ Voiture (matricule, marque, modele, ...)

Personne

cin
nom
prenom
age
...

voiture

matricule
marque
modele
...

Le modèle conceptuel de données (MCD)

❑ Les concepts de base

- ❑ **Identifiant:** identifiant d'une entité est un ensemble minimum de propriétés tels qu'il n'existe pas 2 occurrences de l'entité qui ont les mêmes valeurs pour ces propriétés
- ❑ la valeur de l'identifiant ne peut pas appartenir à plus qu'une occurrence d'entité
- ❑ Toutes les propriétés dépendent de l'identifiant. Pour une valeur de l'identifiant, on ne peut faire correspondre qu'une seule valeur au plus de la propriété.
- ❑ **Exemple:**
 - ❑ **Personne** (cin, nom, prenom, age, ...)
 - ❑ **Voiture** (matricule, marque, modele, ...)

Personne

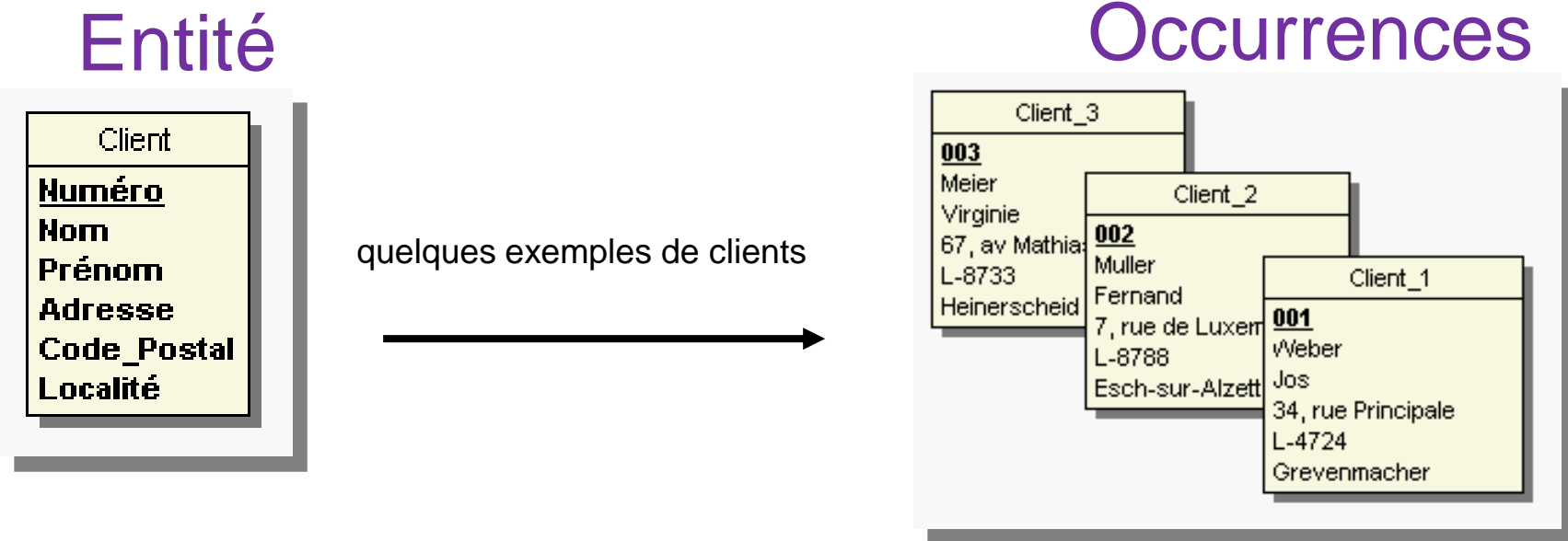
cin
nom
prenom
age
...

voiture

matricule
marque
modele
...

Le modèle conceptuel de données (MCD)

□ Exemple:



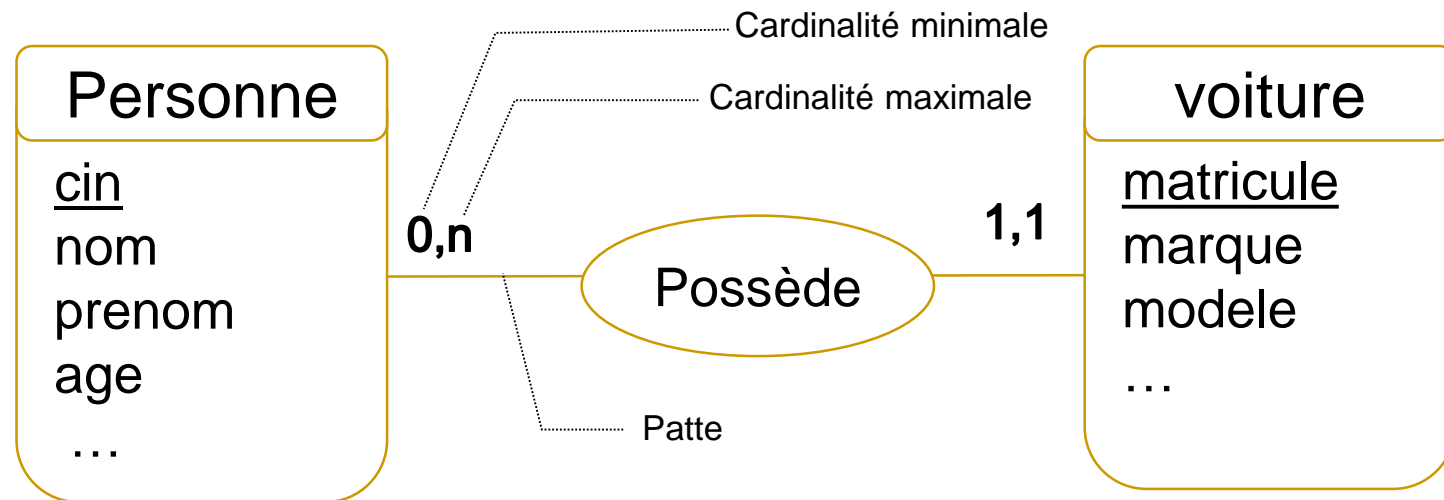
Chacun de ces clients représente une occurrence de l'entité Client.

Le modèle conceptuel de données (MCD)

- ❑ **Associations:** L'association est un lien sémantique entre entités
 - ❑ Une association a un nom et peut éventuellement être porteuse de propriétés.
- ❑ **Cardinalités :**
 - ❑ la description complète d'une relation nécessite la définition précise de la participation des entités.
 - ❑ La cardinalité est le nombre de participation d'une entité à une relation.

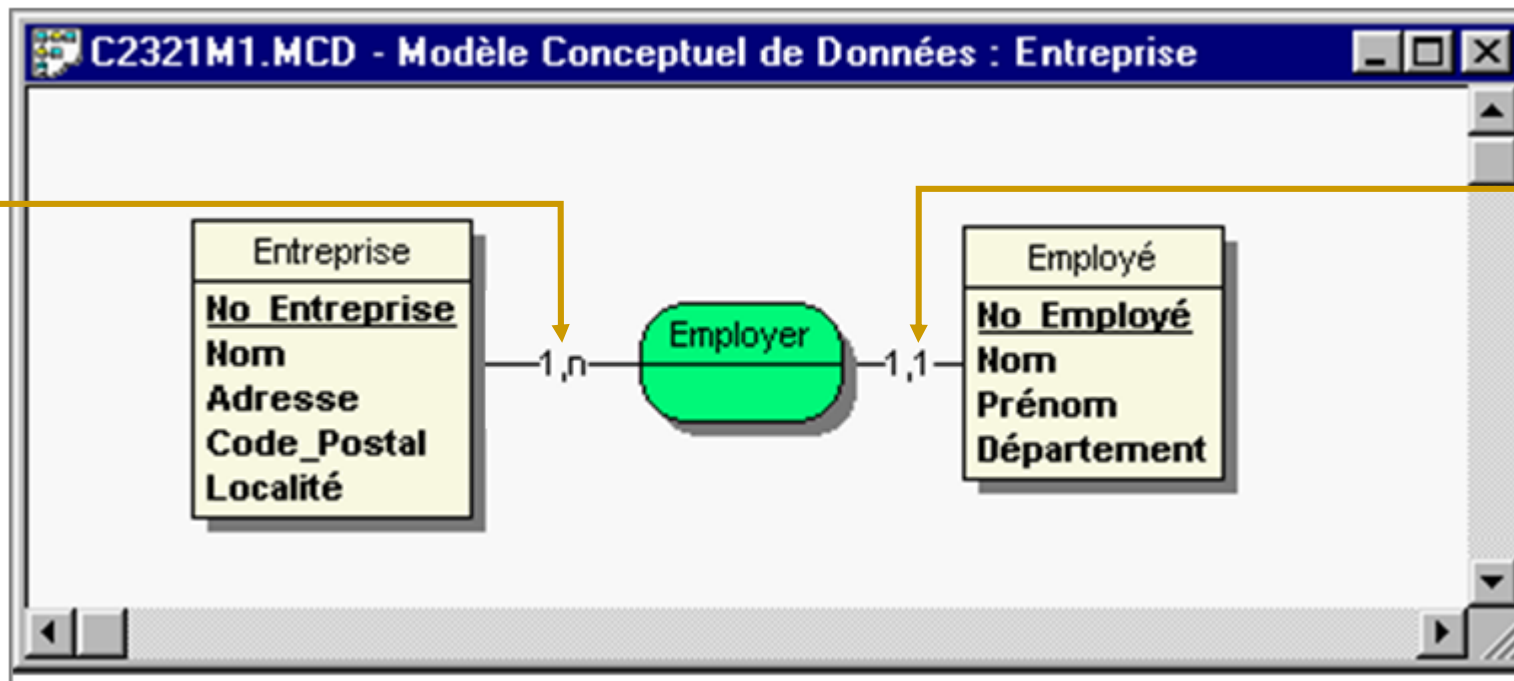
- ❑ **Exemple:**

- ❑ Une personne peut avoir aucune ou plusieurs voitures (0,n)
- ❑ Une voiture peut être possédée par une et une seule personne (1,1)



Le modèle conceptuel de données (MCD)

- ❑ **Exemple :** un MCD qui représente une entreprise avec ses employés.



Dans une entreprise, il y a un ou plusieurs employés

Un employé travaille dans une et une seule entreprise

Le modèle conceptuel de données (MCD)

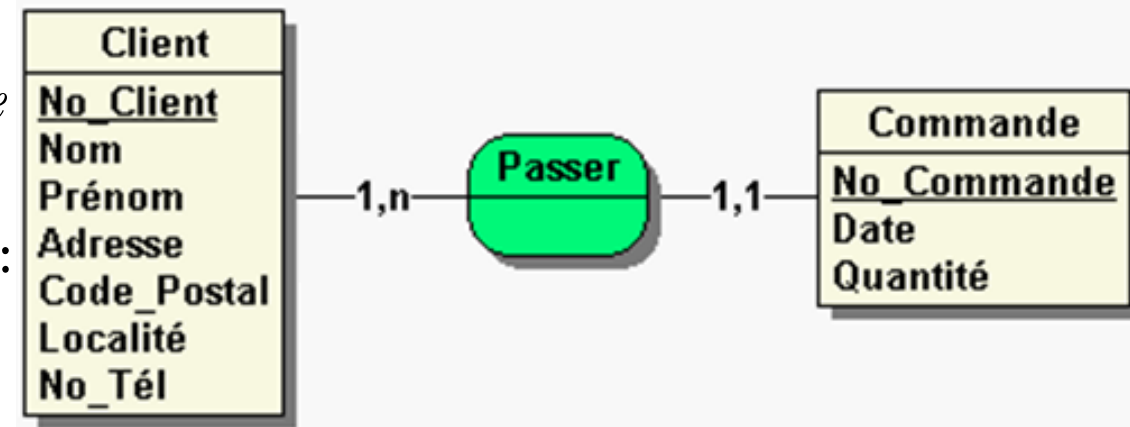
Exemple: Que signifie Les cardinalités suivantes?

Entre l'entité *Client* et la relation *Passer*, nous avons :

- ❑ Cardinalité **minimale = 1** , *ce qui veut dire que chaque client passe au moins une commande*
- ❑ Cardinalité **maximale = n** , *ce qui veut dire que chaque client peut passer plusieurs (n) commandes*

Entre l'entité *Commande* et la relation *Passer* , nous avons :

- ❑ Cardinalité **minimale = 1** , *donc chaque commande est passée par au moins un client*
- ❑ Cardinalité **maximale = 1** , *chaque commande est passée au maximum par un seul client*

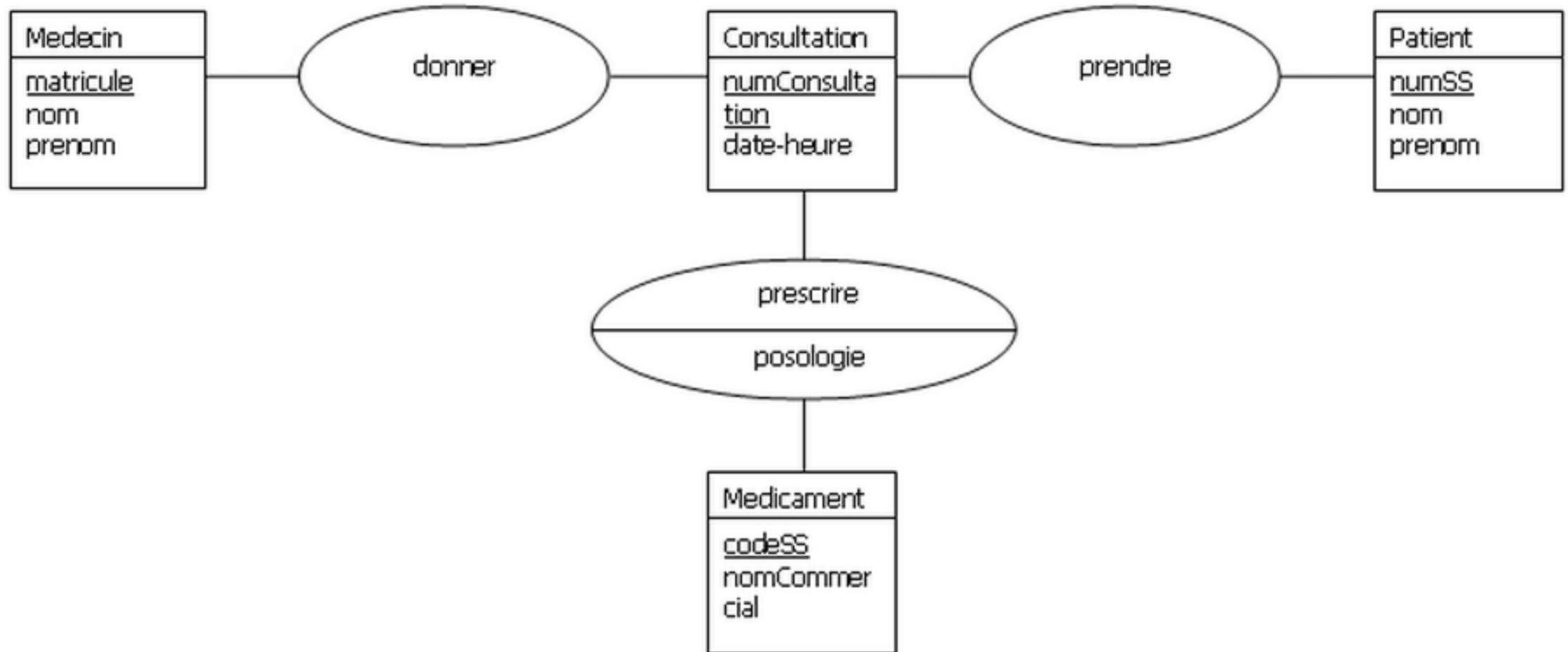


Le modèle conceptuel de données (MCD)

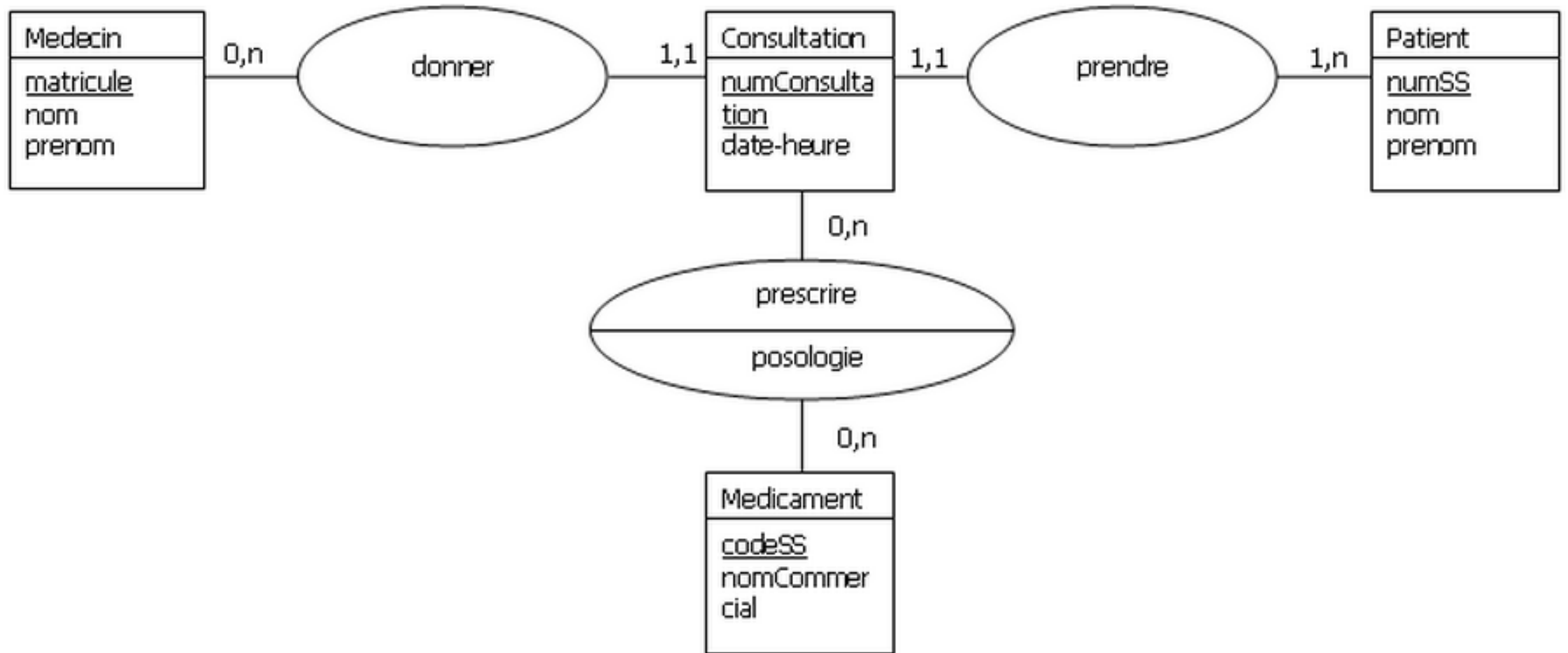
- ❑ Les cardinalités maximales et minimales traduisent les contraintes propres aux entités et associations. Dans un schéma conceptuel, elles sont représentées comme suit :
 - ❑ **0-1** aucune ou une seule
 - ❑ **1-1** une et une seule
 - ❑ **0-N** aucune ou plusieurs
 - ❑ **1-N** une ou plusieurs

Exercice :

Ajouter les cardinalités au MCD suivant:



Solution



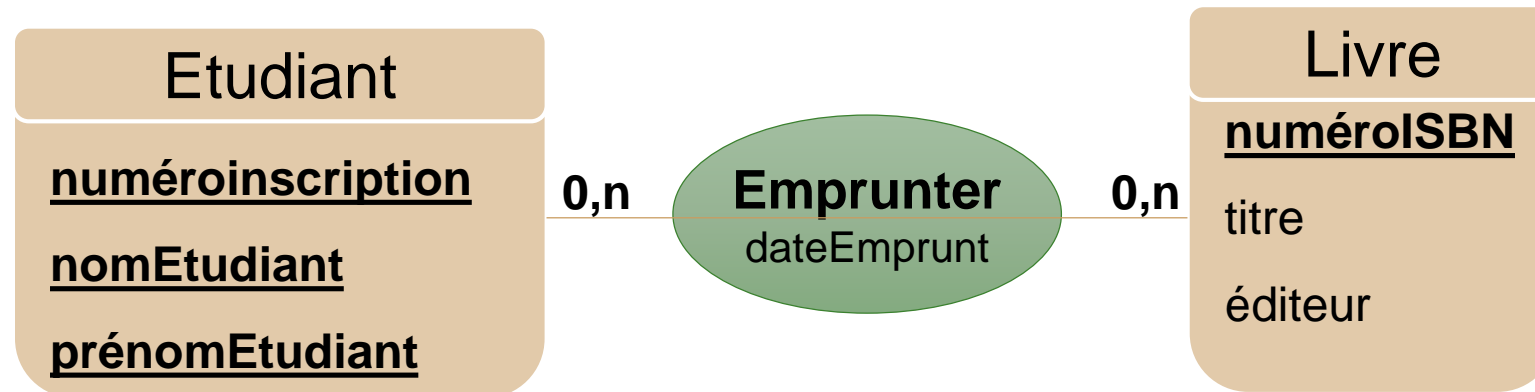
Le modèle conceptuel de données (MCD)

- Association porteuse de données.

Une association peut généralement être dotée de propriétés

- Exemple :

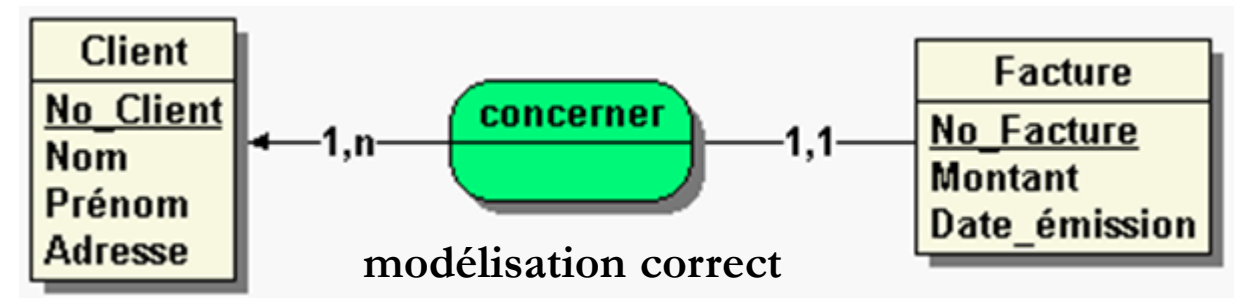
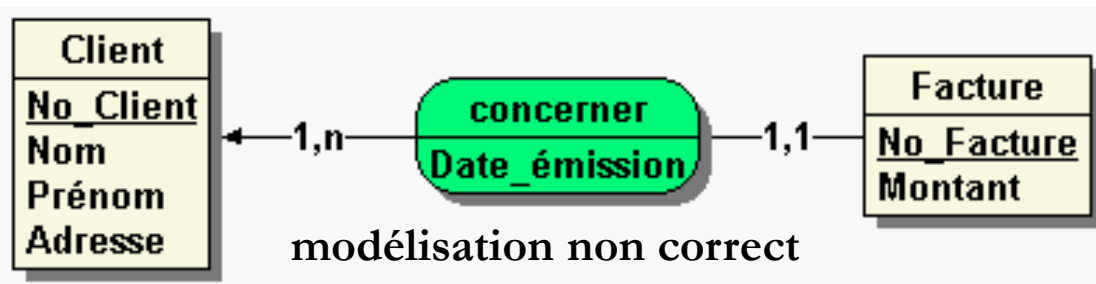
- Etudiant emprunte un ou plusieurs livres, et le livre peut être emprunté par plusieurs étudiants à des dates différentes
- Une commande contient des produits, et un produit peut-être figuré dans plusieurs commandes



Le modèle conceptuel de données (MCD)

Une association à cardinalité (1,1) n'est jamais porteuse de propriétés. Dans ce cas, les propriétés migrent dans l'entité portant cette cardinalité (1,1).

Exemple:



Chaque facture ne possède qu'une seule date d'émission, ce qui fait que la propriété Date_émission doit migrer dans l'entité Facture.

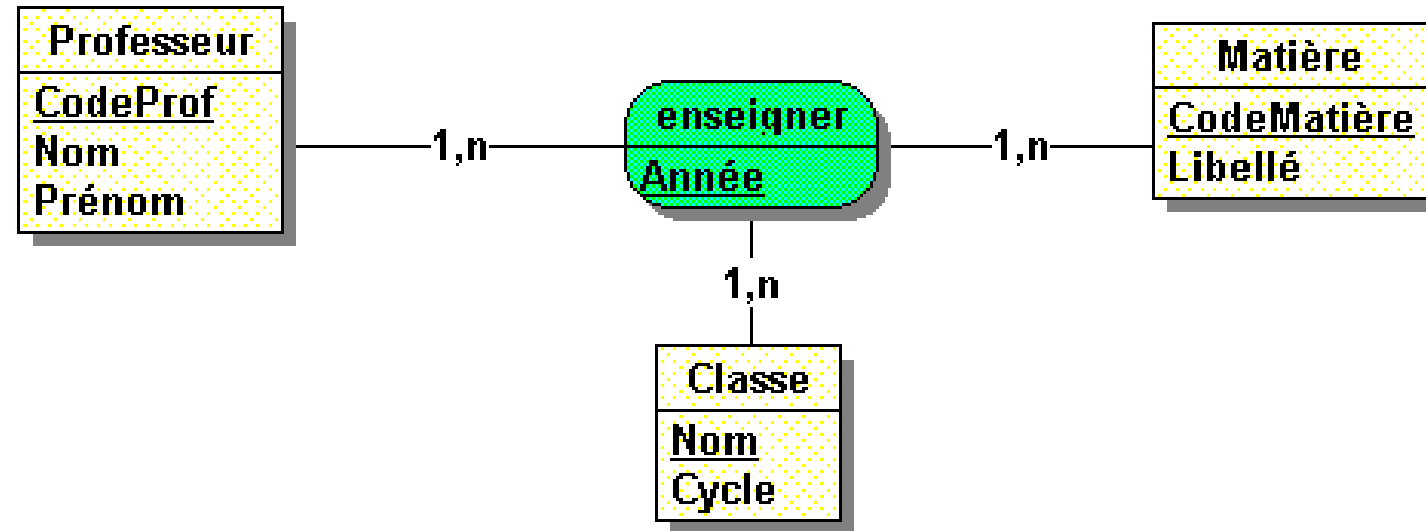
Le modèle conceptuel de données (MCD)

Association ternaire

Une association ternaire est une association à laquelle sont liées 3 entités

Exemple :

A partir des 3 entités Professeur (CodeProf, Nom, Prénom); Matière(CodeMatière, Libellé) et Classe(Nom, Cycle) il s'agit de créer un MCD qui renseigne sur le fait quelle matière est enseignée dans quelle classe par quel professeur pour une année scolaire donnée.



Le modèle conceptuel de données (MCD)

Association ternaire

Quand faut-il utiliser une relation ternaire ?

- ☐ Si une ou plusieurs des entités liées à une relation ternaire possèdent une cardinalité maximale de 1.
- ☐ Il faudrait mieux décomposer la relation ternaire, c.à.d. la représenter par 2 relations binaires.

☐ Exemple:

- ☐ La direction d'une chaîne d'hôtels désire gérer les séjours des clients dans les différents hôtels.
- ☐ Comme on peut effectivement dire "Un client effectue un séjour dans un hôtel" on est amené à proposer la modélisation suivante.

Le modèle conceptuel de données (MCD)

Association ternaire

❑ Solution proposée de l'exemple précédent:

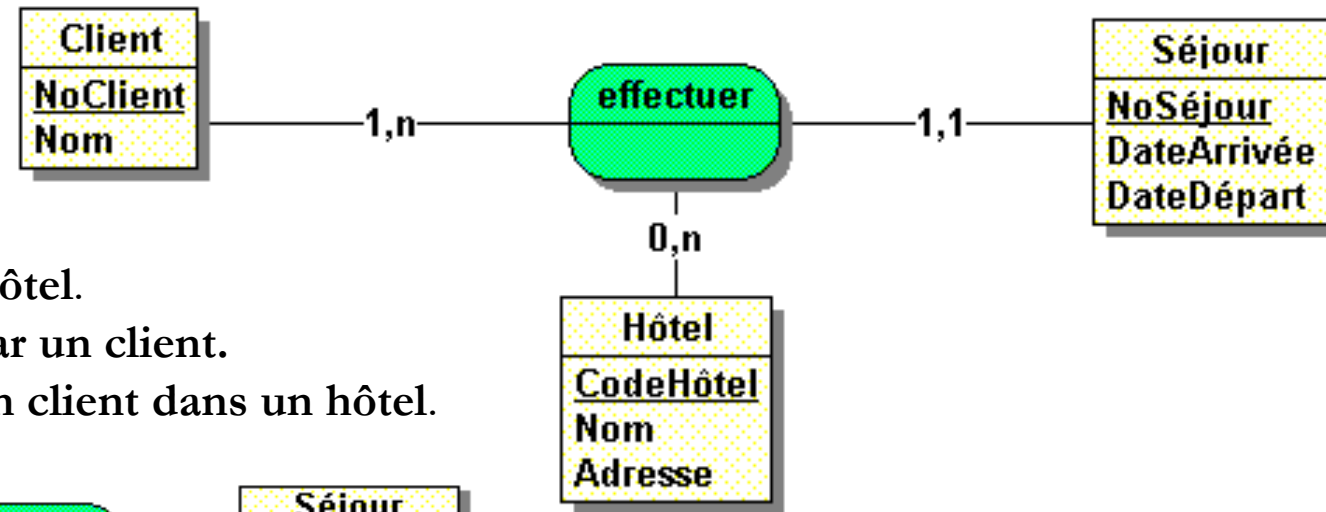
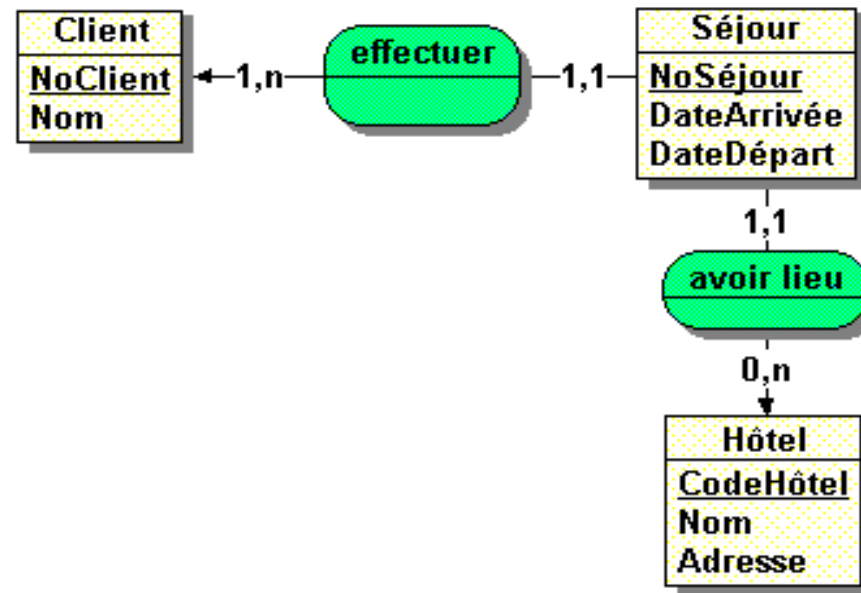
Il existe 3 façons pour lire/interpréter ce modèle:

Un client peut effectuer 1 à n fois **un séjour dans un hôtel**.

Dans un hôtel peut être effectué 0 à n fois **un séjour par un client**.

Un séjour peut être effectué une et une seule fois **par un client dans un hôtel**.

❑ L'Exemple précédant amélioré:

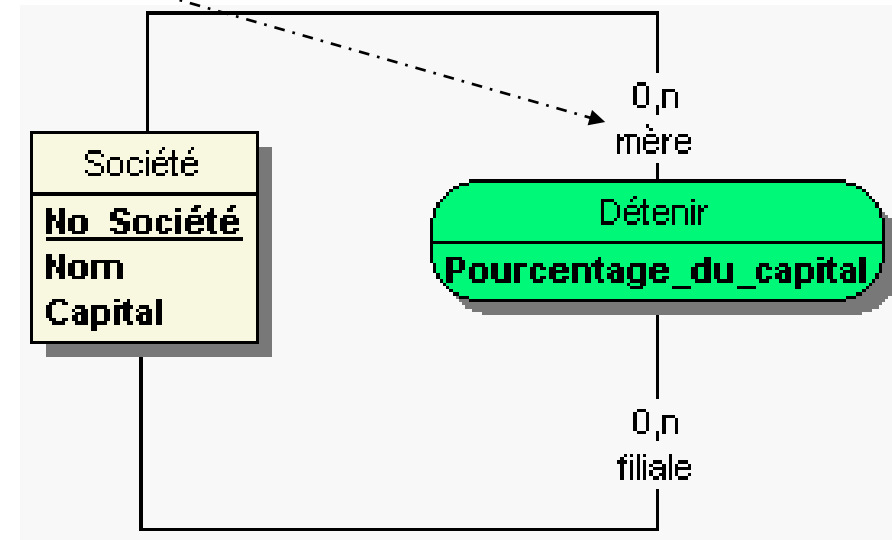
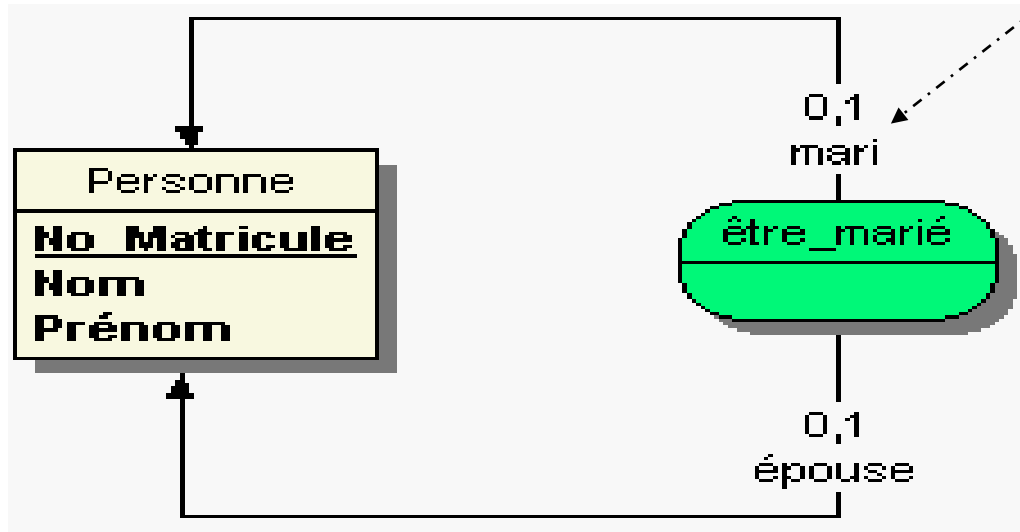


Le modèle conceptuel de données (MCD)

Association réflexive

- ❑ Une association réflexive est une association reliant des occurrences de la même entité. Ces associations sont quasiment toujours binaire (2 branches).
- ❑ Pour lire une association réflexive, il faut connaître le rôle attribué à chaque branche de l'association, c'est-à-dire le rôle de chaque occurrence dans le lien

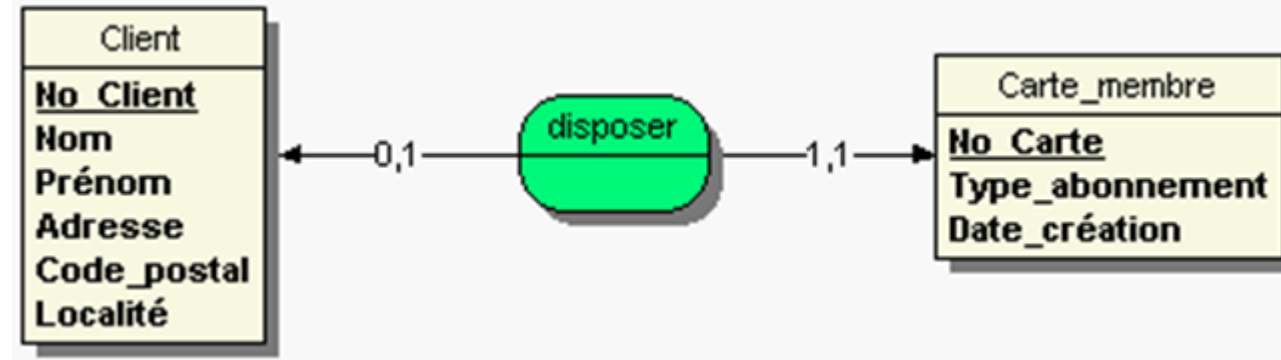
❑ Exemple:



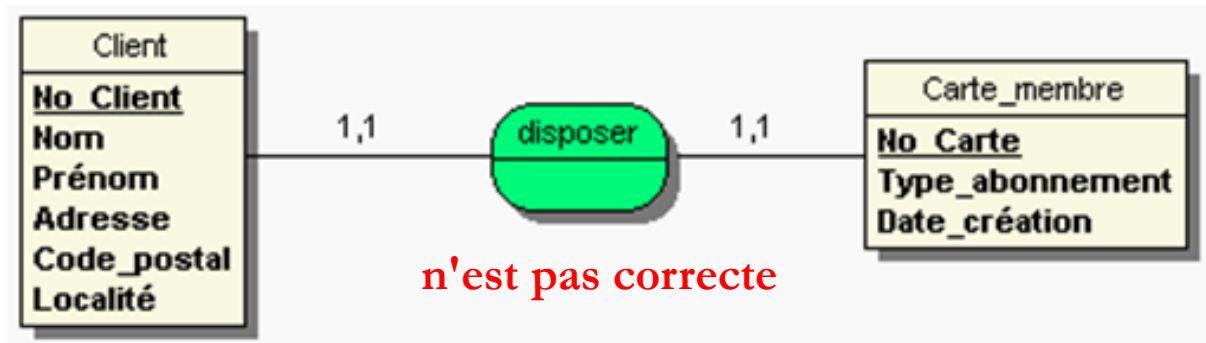
Le modèle conceptuel de données (MCD)

La dépendance d'une relation

On dit qu'une entité est **indépendante** par rapport à une relation lorsque sa cardinalité **minimale vaut 0**, et **dépendante** par rapport à une relation lorsque sa cardinalité **minimale vaut 1**.



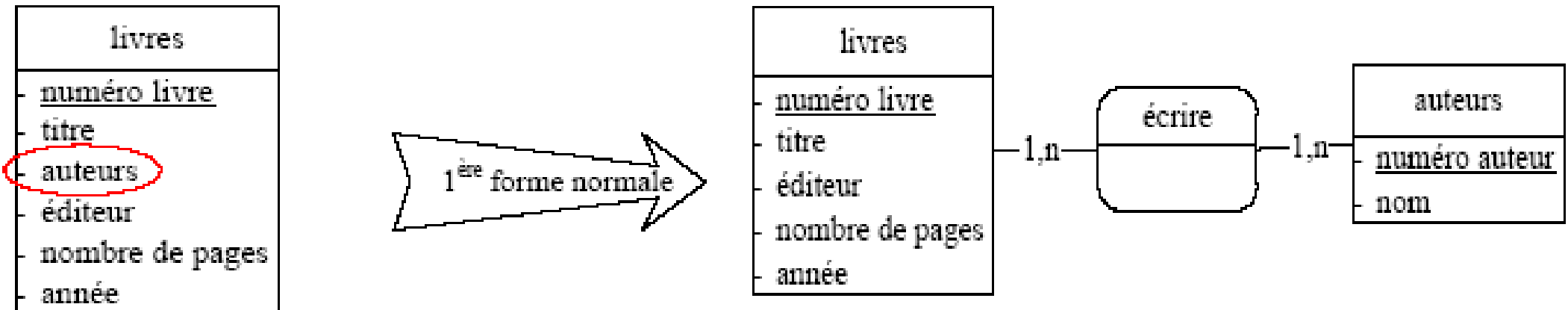
Une relation ne peut pas être liée uniquement à des entités dépendantes ayant en plus une cardinalité maximale de 1 !!!



Dans ce cas il faut réunir les propriétés des deux entités dans une seule.

Amélioration d'un MCD

- **Première forme normale (1FN - 1NF) : attribut élémentaire**
 - Une relation est en première forme normale si et seulement si :
 - Elle possède un identifiant
 - tout attribut contient une valeur atomique (monovalué). c.-à-d., dans un attribut, on ne peut mettre qu'une seule valeur.
- Exemple:
 - L'entité suivante n'est pas en 1NF : *Personne*(noPers, nom, *prénoms*)
Il faut faire : *Personne*(noPers, nom, *prénom1*, *prénom2*, *prénom3*)
 - L'entité suivante n'est pas en 1NF : *Livres*(numéro_livre, titre, *auteurs*, nombre_de_pages, année)
 - Scinder l'entité :

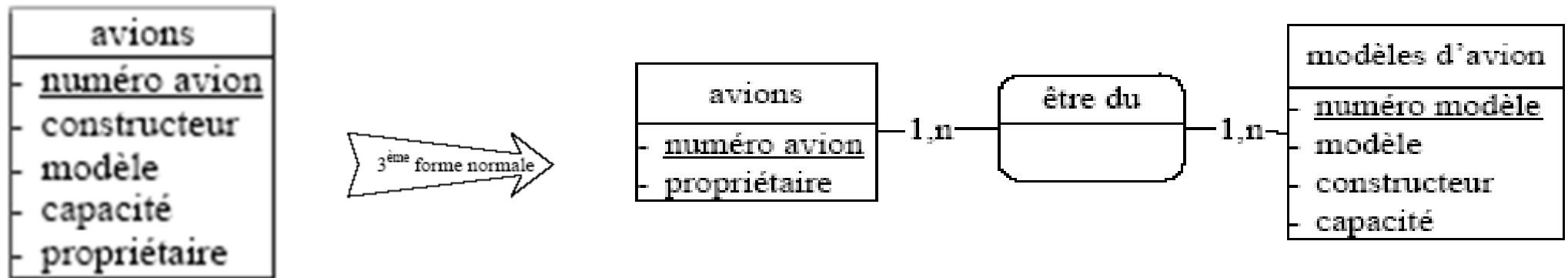


Amélioration d'un MCD

- **Deuxième forme normale (2FN - 2NF) dépendance fonctionnelle élémentaire**
 - Une relation est en 2FN lorsque :
 - Elle est en 1FN,
 - Les attributs non identifiant dépendent de toute l'identifiant et non d'une partie de l'identifiant.
 - Elimination des dépendances entre partie de l'identifiant et attributs non identifiant
 - Si ce n'est pas le cas, scinder la relation en autant de relation que de groupes d'attributs dépendant de chaque morceau de clé et de la clé entière.
 - Exemple:
 - L'entité suivante n'est pas en 2FN :
 - Commande(noFournisseur, refArticle, raisonSocialeFou, adresseFou, quantité, prix)
 - Scinder l'entité :
 - Fournisseur(noFournisseur, raisonSociale, adresse)
 - Commande(noFournisseur#, refArticle#, quantité, prix)

Amélioration d'un MCD

- **Troisième forme normale (3FN - 3NF) dépendance fonctionnelle élémentaire directe**
- Une relation est en 3FN lorsque :
 - Elle est en 2FN,
 - chaque attribut de la relation ne dépend que de l'identifiant et pas d'un autre attribut de la relation.
 - Elimination des transitivités, dépendances entre attributs non identifiant.
- Si ce n'est pas le cas scinder la relation en autant de relations que de dépendance entre attributs non identifiant.
- Exemple:
 - L'entité suivante n'est pas en 3FN : *avions*(numéro avion, constructeur, modèle, capacité, propriétaire)
 - la capacité et le constructeur d'un avion ne dépendent pas du numéro d'avion mais de son modèle.



Le Modèle logique de données (MLD)

■ Définition:

- ❑ Rappel: Le niveau logique répond à la question comment
- ❑ Un MLD permet de modéliser la structure selon laquelle les données seront stockées dans la future base de données
- ❑ Il est adapté à une famille de SGBD : ici les SGBD relationnels (MLD Relationnels ou MLD-R)
- ❑ Il utilise le formalisme graphique Merise
- ❑ Il permet d'implémenter la base de données dans un SGBD donné

Le Modèle logique de données (MLD-R)

- Lorsque les données ont la même structure (par ex. renseignements relatifs à des étudiants), on peut alors les organiser en tables dans lesquelles :
 - les colonnes décrivent les champs (**attributs**) en commun
 - les lignes contiennent les valeurs de ces champs pour chaque enregistrement donné

Etudiant

cne	Nom	Email
122409871	Alami	alami@gmail.com
122409872	Saadi	saadi@yahoo.fr
2409873	Mamouni	Mamouni@hotmail.com
...

Le Modèle logique de données (MLD-R)

Définitions :

- ❑ **Relation (ou table) :** une relation correspond à un tableau ou une matrice à deux dimensions, elle est caractérisée par un nom et est composée de :
 - ❑ **Un Schéma :** La structure de la table, c'est un ensemble d'**attributs** avec leurs domaines.
 - ❑ **Les enregistrements (n-uplets) :** ou encore l'extension de la relation, ils sont représentés par les enregistrements des tables, leur ordre n'a pas d'importance.

Exemple : la relation (table) étudiant

❑ **Schéma :**

etudiant {**Cne**, **Nom**, **Email**}

❑ **enregistrements :**

```
{  
  {cne: 122409871, Nom: "Alami", Email: "alami@gmail.com"}  
  {cne: 122409872, Nom: "Saadi", Email: "saadi@yahoo.fr"}  
  ...  
}
```

etudiant	
Cne	
Nom	
Email	

Le Modèle logique de données (MLD-R)

Définitions :

■ Clé primaire :

- ❑ La clé primaire d'une relation est un attribut dont la valeur est différente pour chaque enregistrement de la relation.
- ❑ En base de données la clé primaire garantit l'unicité des enregistrements.

❑ Exemple:

etudiant { Cne, Nom, Email }

etudiant
<u>Cne</u>
Nom
Email

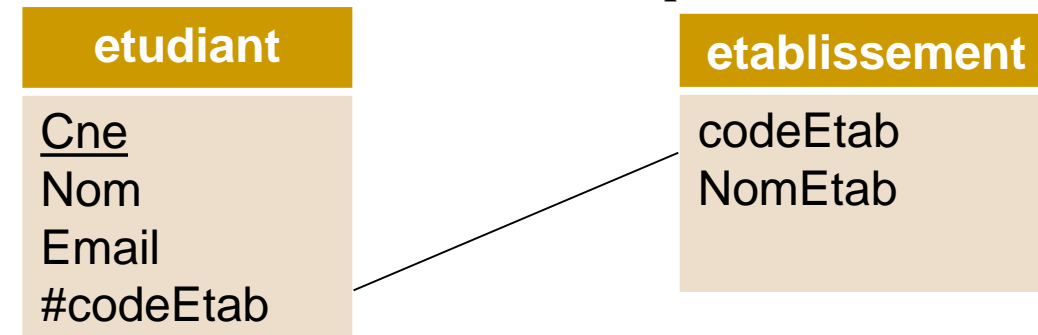
■ Clé étrangère

- ❑ Une clé étrangère dans une relation est un attribut qui se réfère à une donnée connexe.
- ❑ Dans la pratique une clé étrangère fait référence à une clé primaire d'une autre relation. Elle permet, en quelque sorte, de créer des liens entre les relations.

❑ Exemple:

etudiant { Cne, Nom, Email, #codeEtab }

etablissement(codeEtab, nomEtab)



Le Modèle logique de données (MLD-R)

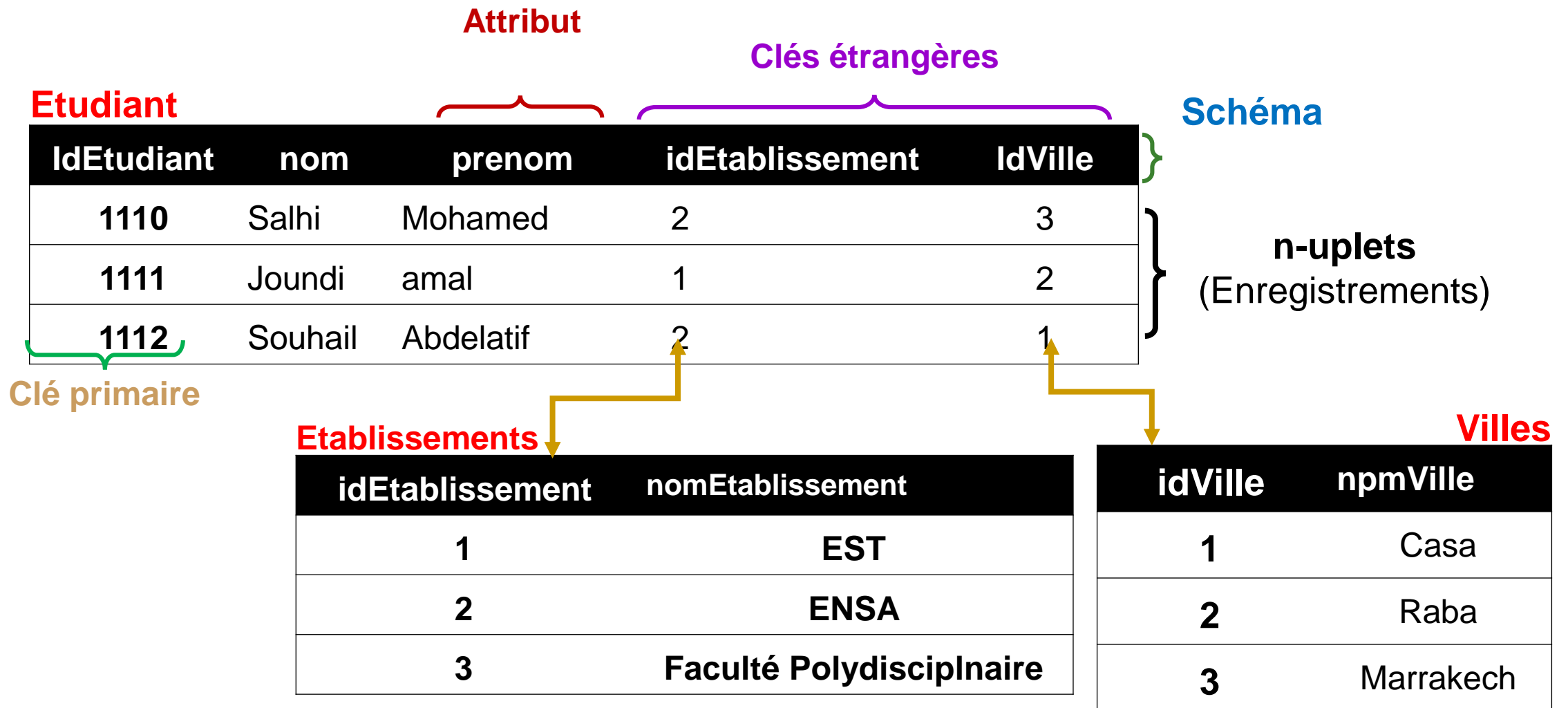
❑ Conventions :

- ❑ Les clés primaires sont soulignées
- ❑ Les clés étrangères sont précédées par un #
- ❑ Par exemple dans la description des colonnes d'une table :
 - clients(numclient, nom client, prénom, adresse)
 - commandes(numcommande, date commande, #numclient)

■ Remarque :

- ❑ une même table peut avoir plusieurs clés étrangères mais une seule clé primaire (éventuellement composée de plusieurs colonnes)
- ❑ une clé étrangère peut aussi être primaire (dans la même table)
- ❑ implicitement chaque colonne qui compose une clé primaire ne peut pas recevoir la valeur vide (NULL interdit)
- ❑ par contre, si une clé étrangère ne doit pas recevoir la valeur vide, alors il faut le préciser dans la description des
- ❑ colonnes

Exemple récapitulant



Traduction d'un MCD en 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 .

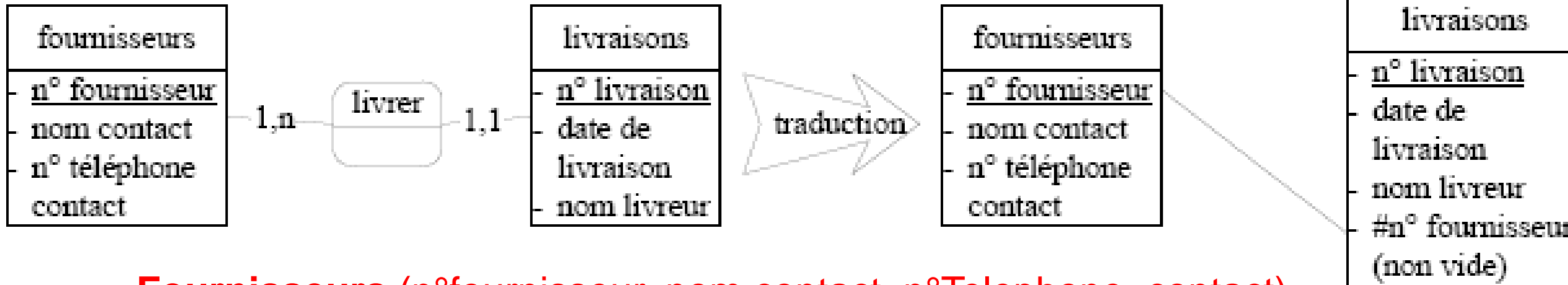
Traduction d'un MCD en MLD

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

❑ **Règle 1 :**

- ❑ L'entité devient une **table** dans laquelle:
- ❑ les **propriétés** deviennent **les attributs**.
- ❑ L'identifiant de l'entité constitue alors **la clé primaire** de la table.

❑ **Règle 2 :** une **association** binaire de type **1: n** disparaît, au profit d'une **clé étrangère** dans la table source (1,1).

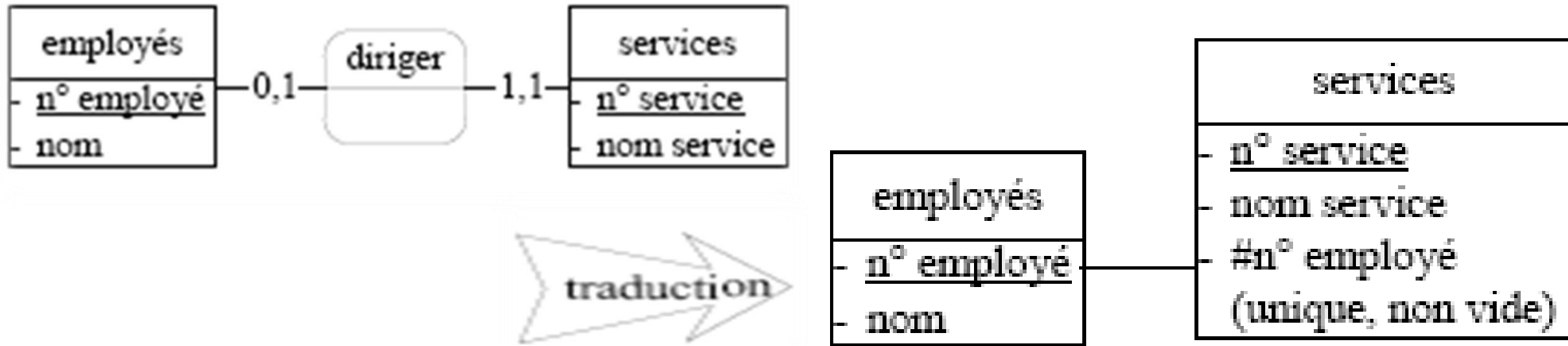


Fournisseurs (n°fournisseur, nom contact, n°Telephone, contact)

Livraison (n°livraisaon, dateLivraison, nomLivreur,#n°fournisseur)

Traduction d'un MCD en MLD

❑ **Règle 3** : une **association** de type **1:1** est traduite comme une association binaire de type 1:n
La cardinalité 1,1 est une contrainte plus forte que la cardinalité 0,1. Donc: L'identifiant de l'entité but du lien 1,1 devient **clé étrangère** dans la table issue de l'entité source.



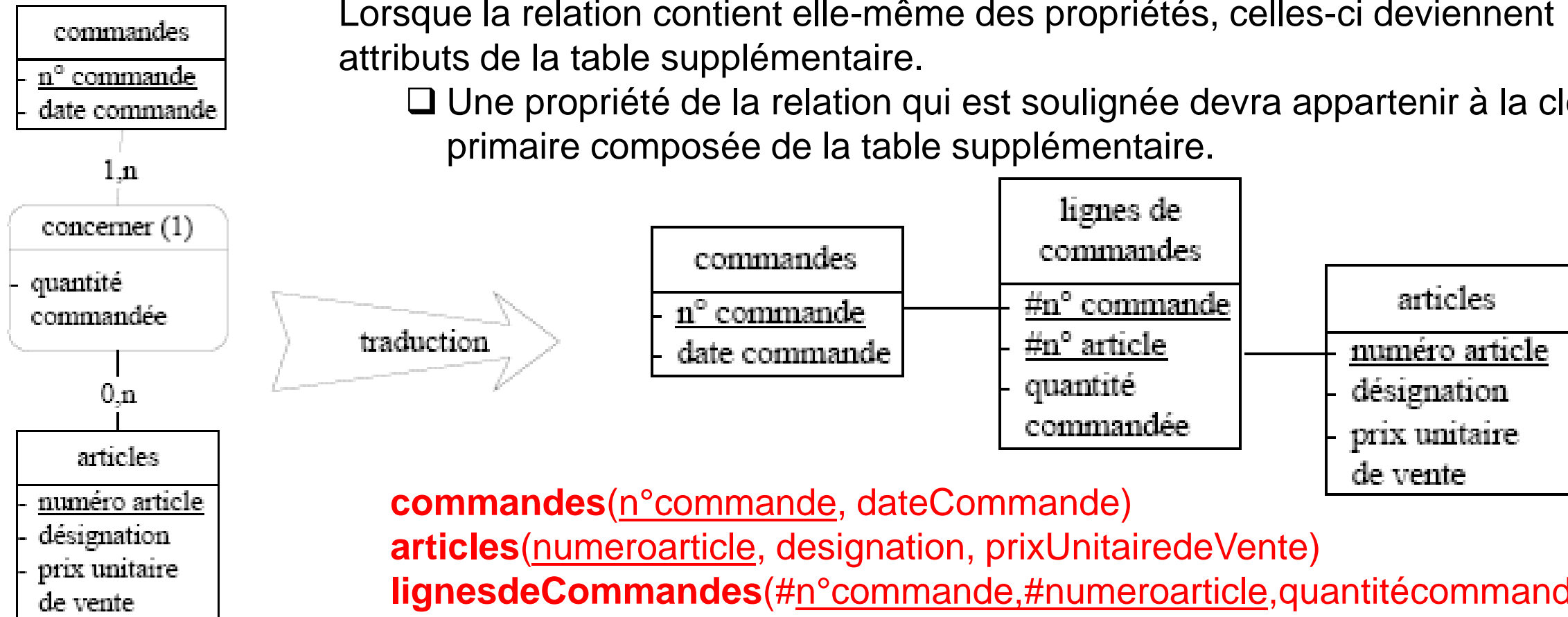
employés(n°employe, nom)

services(n°service, nom_service, #n°employe)

Traduction d'un MCD en MLD

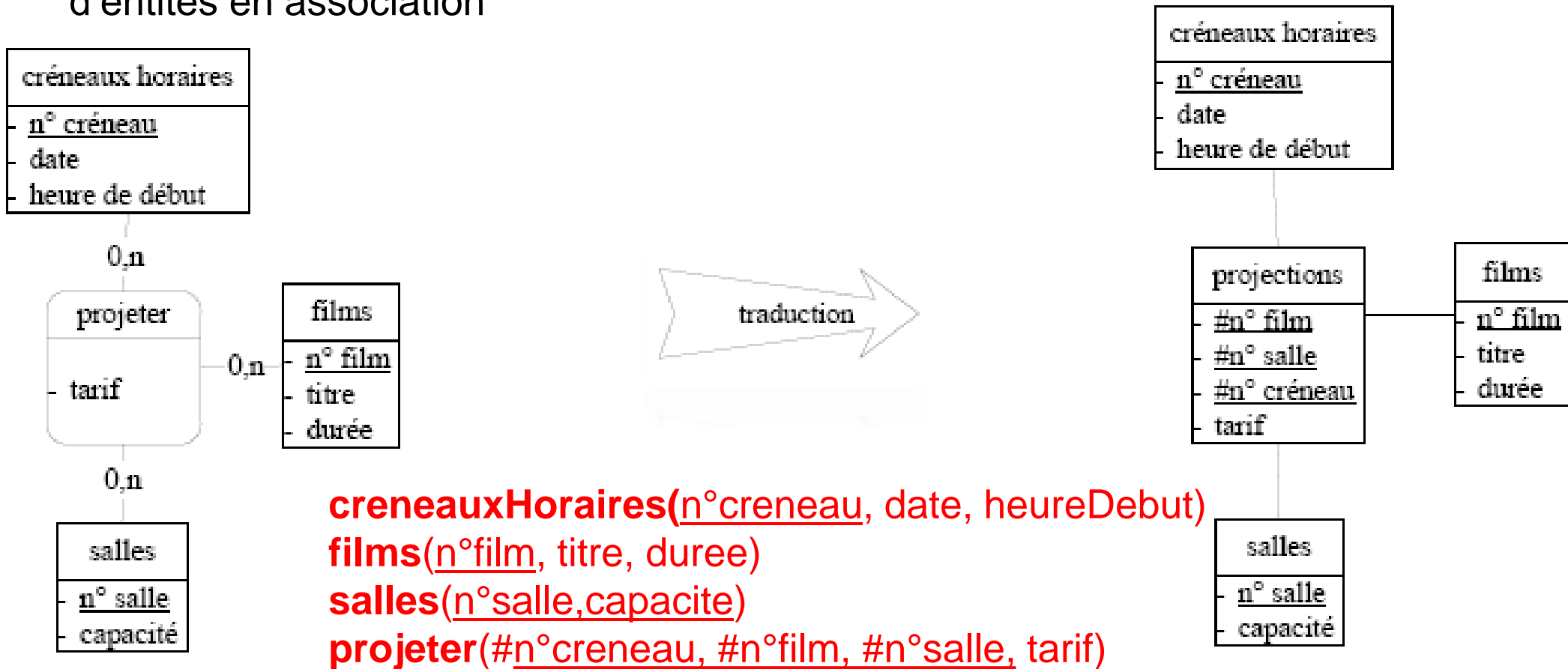
❑ **Règle 4** : une **association** binaire de type **n:m** devient une **table** supplémentaire ayant comme clé primaire une clé composée des clés primaires des 2 tables. Lorsque la relation contient elle-même des propriétés, celles-ci deviennent attributs de la table supplémentaire.

❑ Une propriété de la relation qui est soulignée devra appartenir à la clé primaire composée de la table supplémentaire.



Traduction d'un MCD en MLD

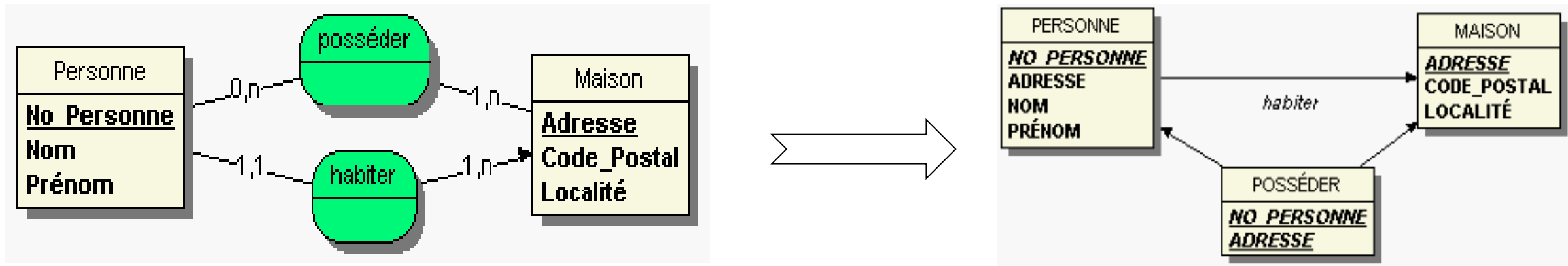
- ❑ **Règle 5** : une association non binaire (ternaire au plus) est traduite par une table supplémentaire dont la clé primaire est composée d'autant de clés étrangères que d'entités en association



Traduction d'un MCD en MLD

Transformation de plusieurs relations entre 2 entités

Les règles générales s'appliquent



personne(no_personne, #adresse, nom, prenom)

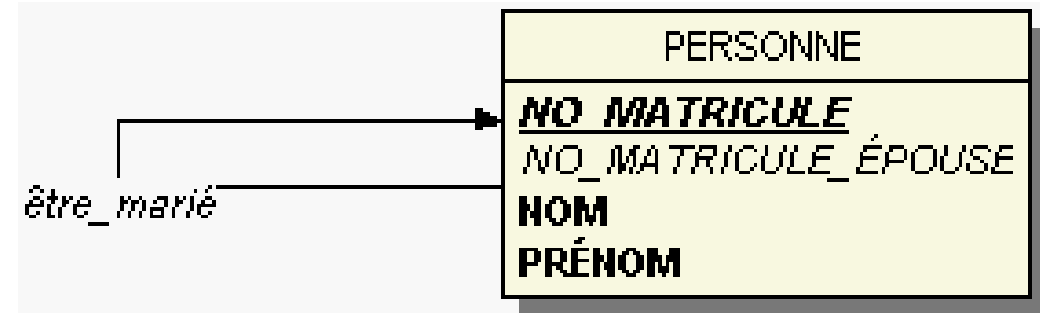
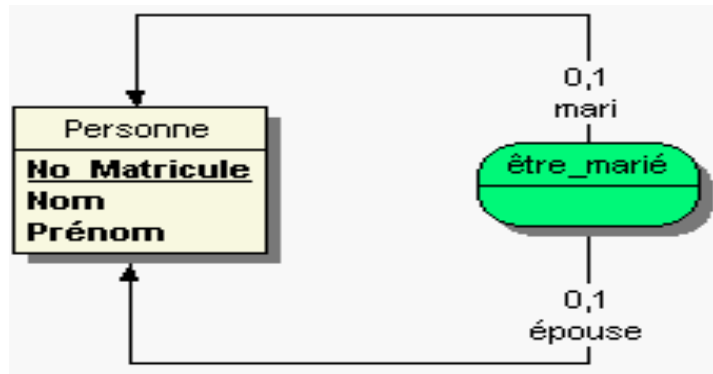
maison(adresse, code_postal, localite)

posseder(#no_personne, #adresse)

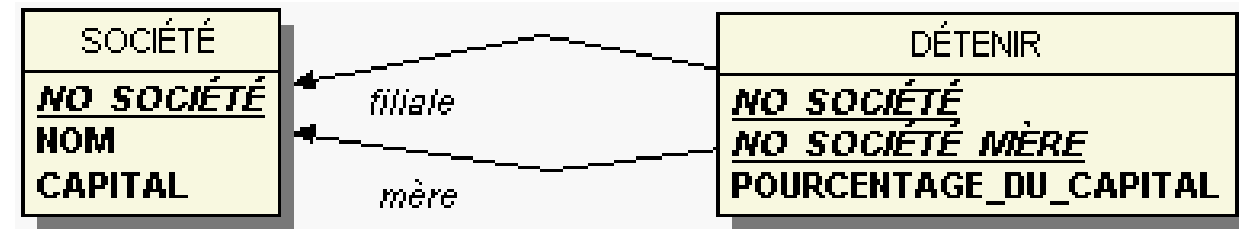
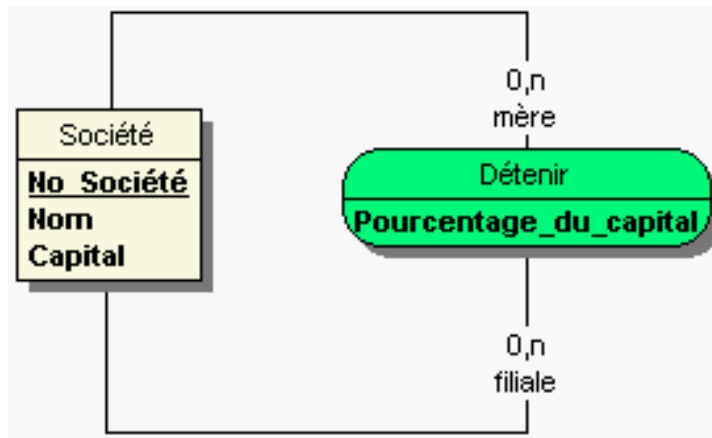
Traduction d'un MCD en MLD

Transformation des relations réflexives

Nous appliquons les règles générales avec la seule différence que la relation est 2 fois reliée à la même entité



Personne (no_matricule, nom, prenom, no_matricule_epouse)



Societe(no_societe, nom, capital)

detenir (#no_societe, #no_societe_mere, pourcentage_du_capital)