

3. La conception: Modèle relationnel

Le Modèle Conceptuel de Données (MCD)

- ❑ **Le niveau conceptuel** correspond à une **formalisation** du SI indépendante de toute contrainte d'organisation
- ❑ La formalisation des données, au niveau conceptuel, constitue **le MODELE CONCEPTUEL DES DONNEES (M.C.D)**

Objectifs:

- Rassembler les données (regrouper les données ayant des traits communs)
- Structurer les données qui seront utilisées par le S.I
- Décrire les données avec des outils (concepts, schémas)
- Modéliser
- Schématiser

Le Modèle Conceptuel de Données (MCD)

Le M.C.D décrit la **sémantique** c'est à dire le sens attaché à ces données et à leurs rapports et non à l'utilisation qui peut en être faite.

Le **M.C.D** est connu sous différentes dénominations, diagramme:

- **entité-association**,
- entité-relation (dénomination de l'ISO),
- objet-relation.
- individu-relation

Principes fondamentaux

- Le **MCD** a pour but de **décrire** le système d'information à l'aide **d'entités** et de **relations**.
- Il est à la base de tous les **SGBD dits relationnels** (Access, Oracle, DB2...) qui sont les plus utilisés actuellement dans les entreprises.
- Il permet donc une représentation du “ réel perçu ” sous la forme de :
ENTITES, **ASOOCIATIONS** et **PROPRIETES**.

Entité : définition

- Est la **représentation formelle** de la mémorisation d'une **information complexe et cohérente**.
- Sert à décrire plusieurs **occurrences** de la même information complexe.
 - Exemple :
 - une *Ford Kuga*, une *Renault Kadger* et une *Peugeot 3008* sont **trois occurrences** de la même **entité** qu'on peut appeler **voiture**.
- Les éléments de l'information (complexe) modélisée par l'entité sont les **propriétés**.
 - Exemple :
 - *Ford*, *Renaud* et *Peugeot* sont des « **marques** » ;
 - *Kuga*, *Kadger* et *3008* sont des « **modèles** ».
 - Ainsi, **Marque** et **Modèle** peuvent être des **propriétés** de l'entité **voiture**.
- Remarque : on l'appelle aussi **Entité-Type**.

Entité : propriété

Une propriété est **un élément** d'une entité, et **d'une seule** :

- décrit la mémorisation **d'une information élémentaire**,
- a un nom **unique**,
- permet de mémoriser **une valeur**,
- doit avoir **un sens** (donc une valeur) pour chacune des **occurrences** de la composante → ***Domaine de valeurs***.

Domaine= ensemble de valeurs atomiques que peut prendre un attribut.

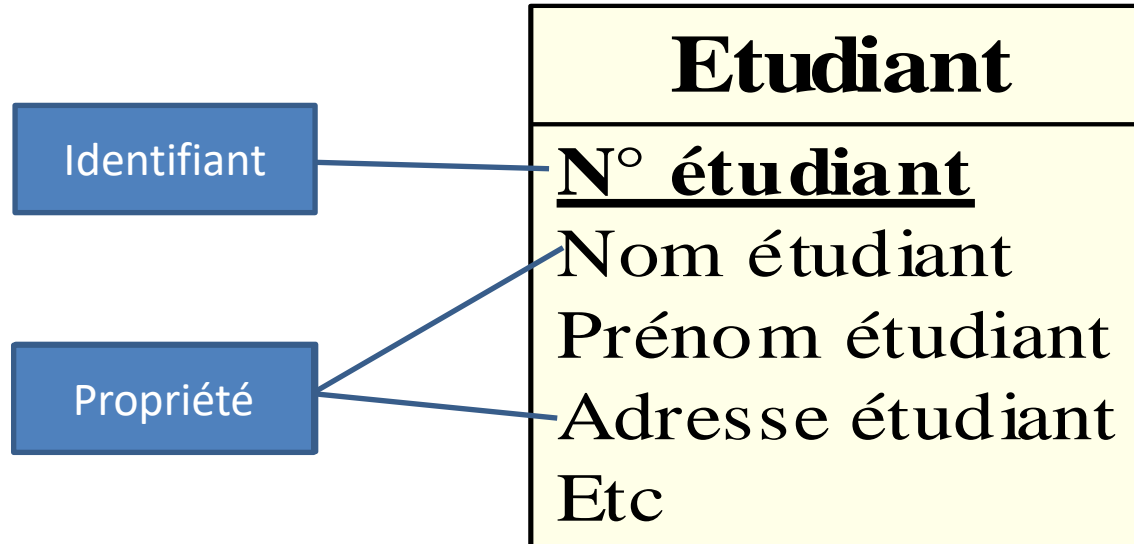
- Exemples de domaines:
 - nom : chaînes de caractères de longueur maximale 30
 - num : entiers compris entre 0 et 99999
 - couleur : {"bleu", "vert", "jaune"}
 - âge : entiers compris entre 16 et 65

Une propriété doit-elle être toujours renseignée ?

Entité : identifiant

- **L'identifiant** de l'entité est une **propriété qui ne peut pas prendre deux fois la même valeur** dans deux occurrences de l'entité.
- **C'est l'identifiant qui fait l'entité.**

Entité : représentation schématique



Entité : occurrences

- Pour une valeur de l'identifiant, on a une valeur de chacune des propriétés.
- Deux occurrences de l'entité ne peuvent avoir la même valeur d'identifiant.
- Les domaines de valeurs des propriétés ne sont pas disjoints.

Etudiant
<u>E0215632</u> Alami Soufiane SidiKacem

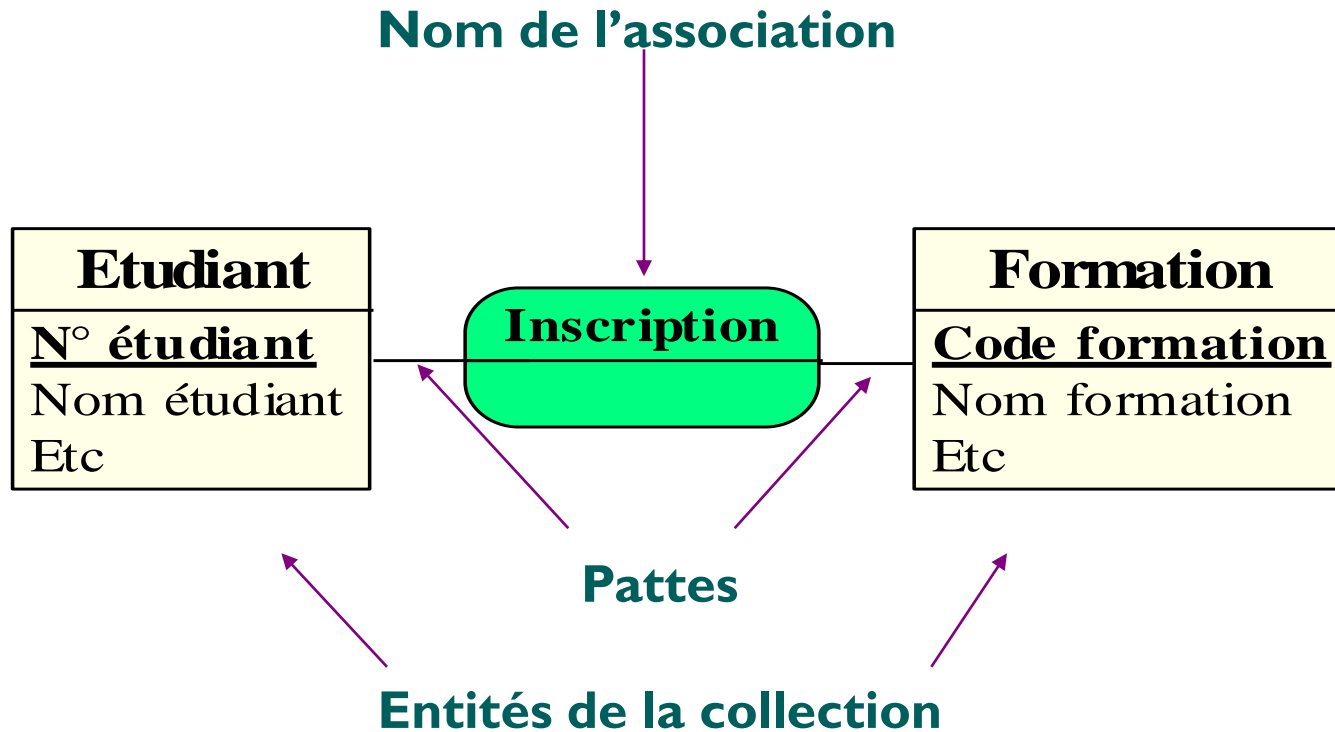
Etudiant
<u>M017895</u> Idrissi Majda Kénitra

Etudiant
<u>Z026793</u> Idrissi Anas Kénitra

Association : introduction

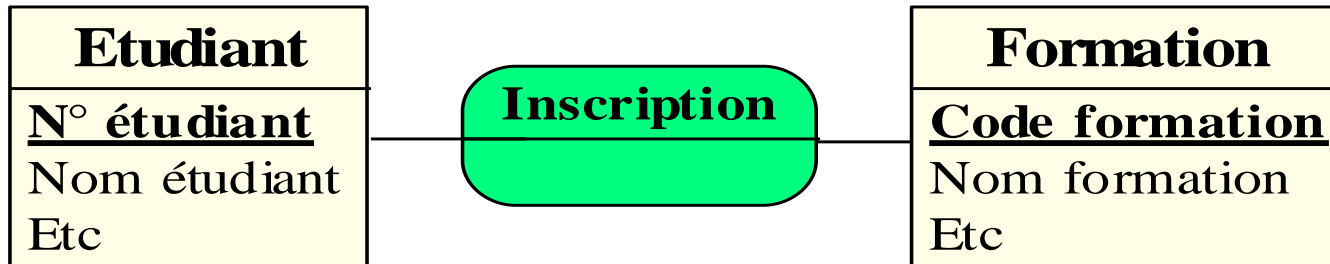
- C'est la représentation abstraite de la mémorisation d'un **lien entre des informations complexes** (représentées par des entités).
- On appelle **collection de l'association** l'ensemble des entités qu'elle relie.
- **Une occurrence de l'association** représente un lien sémantique qui concerne une occurrence de chacune des entités de la collection.

Association : représentation schématique



Association : identifiant

- Il est **implicite** !
- C'est **un n-uplet** composé des identifiants des entités-types concernées.
 - Exemple : l'identifiant de **Inscription** est le couple (N° étudiant, Code formation).



Association : les contraintes de cardinalité

- Une cardinalité est une **précision apportée sur une patte d'une association**.
- Elle indique combien de fois l'entité peut intervenir dans l'association.
- Pour un **MCD**, il est utile de savoir :
 - **la cardinalité minimale** ➔ Si une occurrence de l'entité est obligatoirement concernée par une (au moins) occurrence de l'association ;
 - **la cardinalité maximale** ➔ Si une occurrence de l'entité peut (éventuellement) être concernée par plusieurs occurrences de l'association.

Association : cardinalité minimale

Elle peut prendre deux valeurs :

- ❑ **0**, signifie que l'entité peut **ne pas** intervenir dans l'association.
- ❑ **1**, signifie au contraire qu'elle intervient **obligatoirement une fois**.

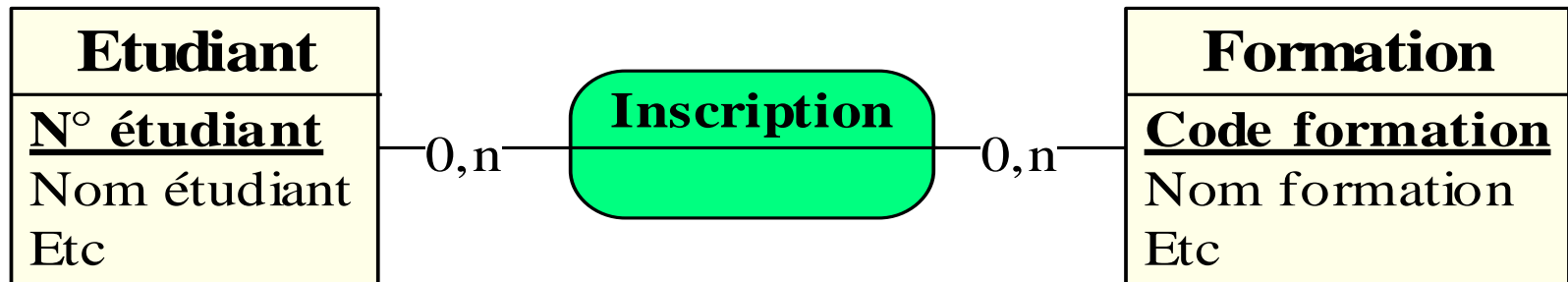
Association : cardinalité maximale

Elle peut prendre deux valeurs :

- ❑ **1**, signifie que l'entité ne peut intervenir **plus d'une seule fois** dans l'association-type ;
- ❑ **n**, signifie au contraire qu'elle peut intervenir **plusieurs fois** dans l'association.

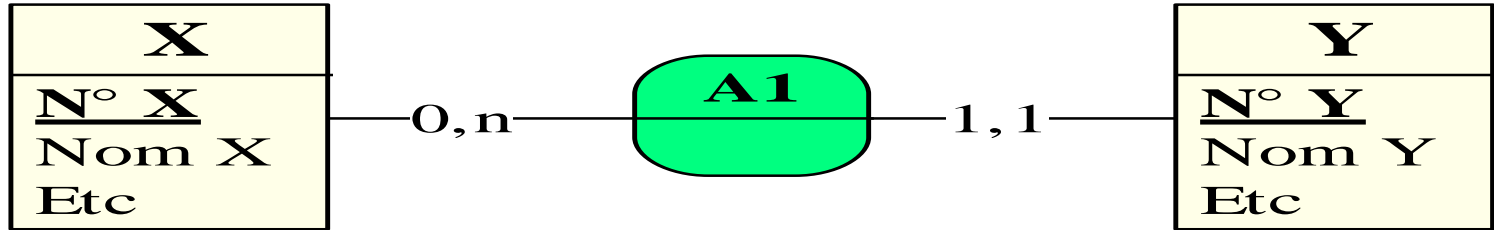
Association : c'est le 1 qui représente la contrainte

Une patte sans contrainte aura
pour cardinalités : (0, n)



Association : double contrainte sur une patte

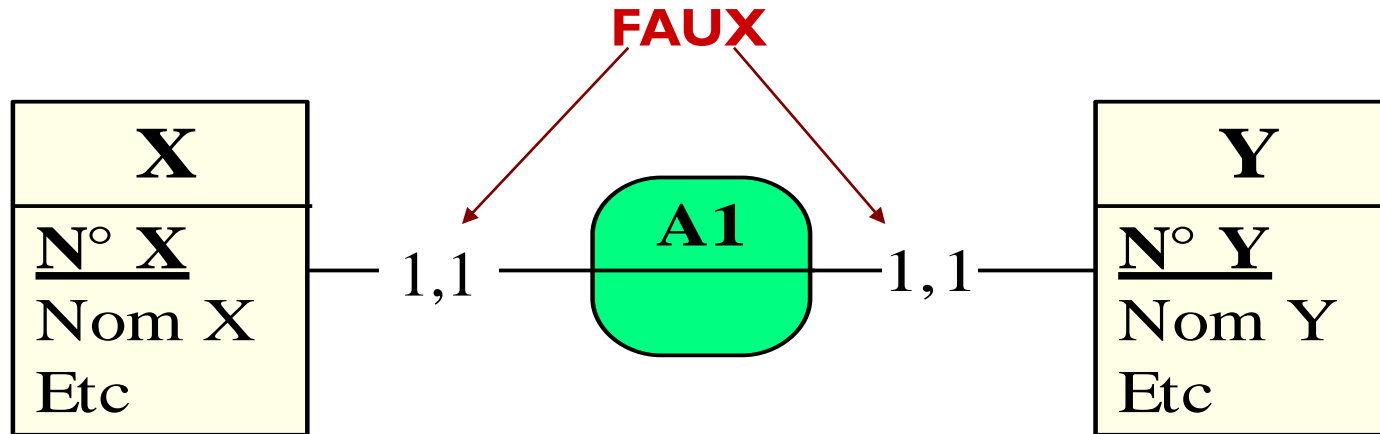
- La cardinalité minimale indique ce qui est **obligatoire**, dans le cas **le plus restrictif**, le plus souvent lors de sa création.



- Sur la patte **X**, le **0** signifie que **X** peut ne pas être reliée à **Y** lors de sa création.
- Le **1** en minimum de la patte **Y** signifie qu'en aucun cas on ne peut créer une occurrence de **Y** sans la relier en même temps à une occurrence de **X**... Cette dernière doit donc avoir été créée avant !

Association : combinaisons de cardinalités possibles

Interdit : (1, 1) / (1, 1)

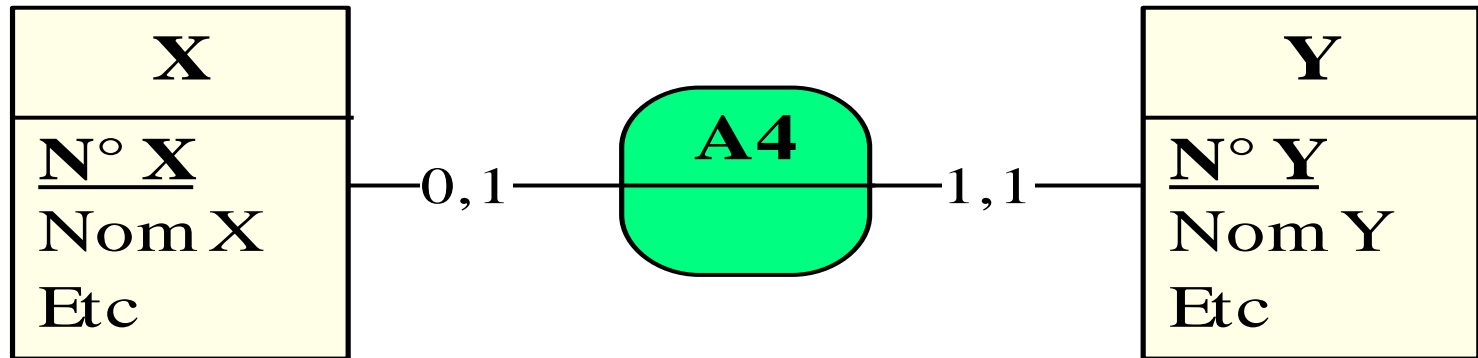


Pourquoi ?

Il faut réunir les propriétés des deux entités dans une seule.

Association : combinaisons de cardinalités possibles

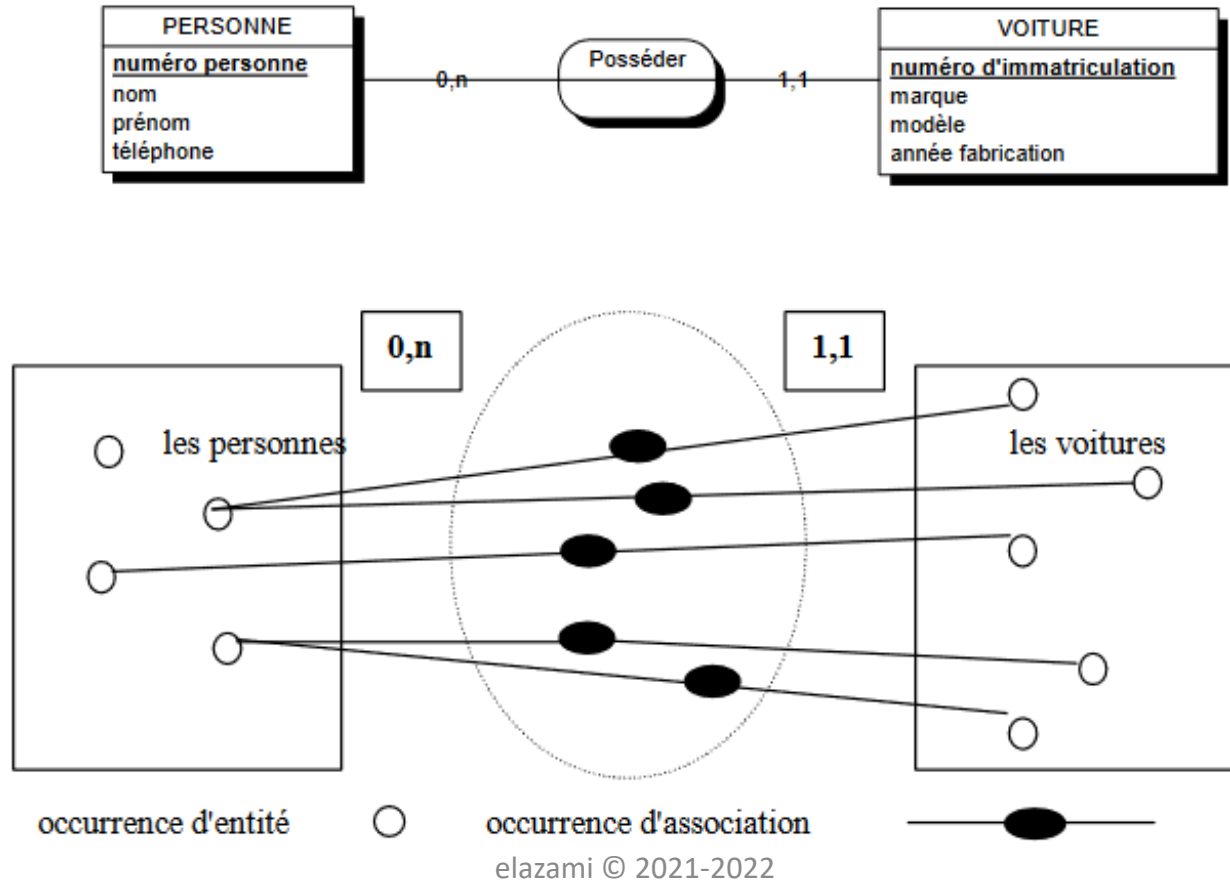
Rare et délicat : (0, 1) / (1, 1)



Significations possibles ?

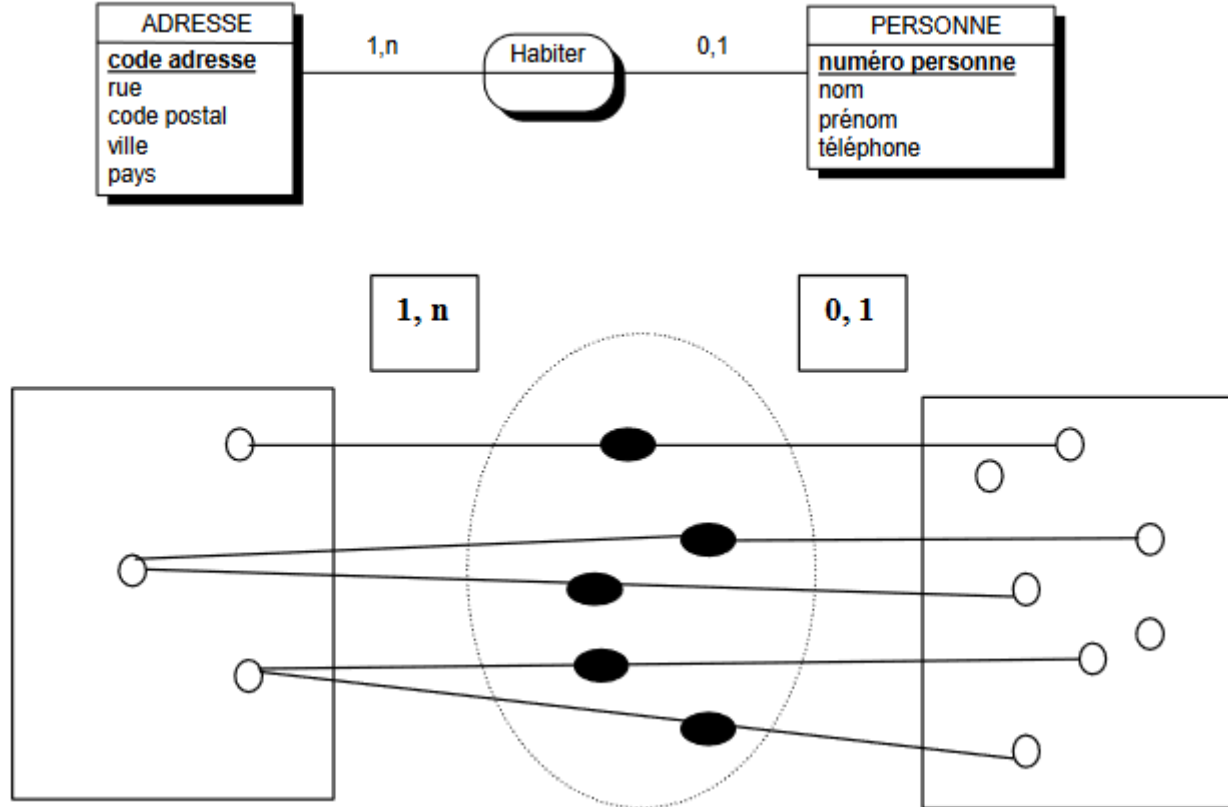
Association: Représentation des cardinalités

Une voiture est possédée par une seule personne. Une personne peut posséder de 0 à plusieurs voitures.



Association: Représentation des cardinalités

Une personne a une adresse ou est sans domicile. A une adresse, il y a une ou plusieurs personnes qui y habitent.



Premières règles...

Entités

Règle 1 : Existence d'un identifiant pour chaque entité

Règle 2 : Pour chaque occurrence d'une entité, chaque propriété ne peut prendre qu'une valeur

Règle 3 : Toutes les propriétés doivent être élémentaires (non décomposables)

Règle 4 : Toutes les propriétés autres que l'identifiant doivent dépendre pleinement et directement de l'identifiant

Association

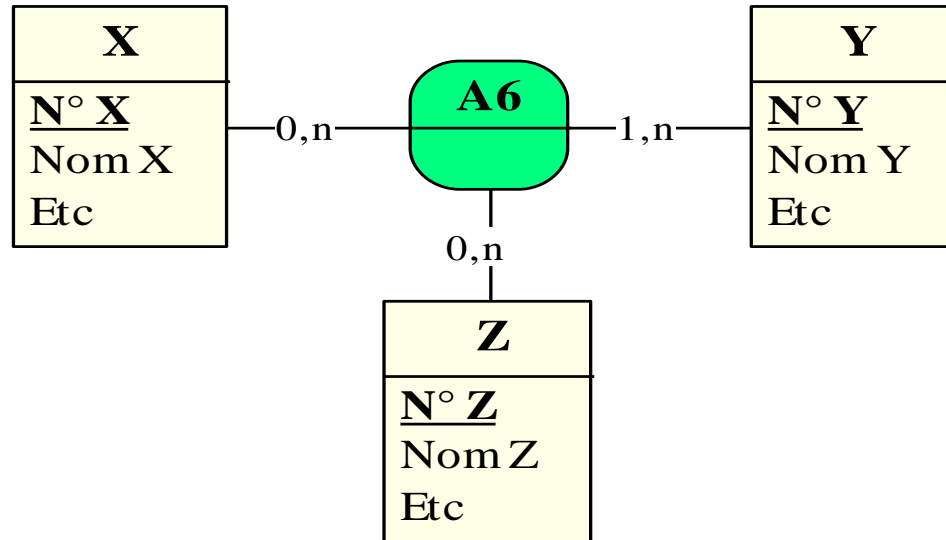
Règle 5 : A chaque occurrence d'une association correspond une et une seule occurrence de chaque entité qui participe à la relation

Règle 6 : Pour chaque occurrence de l'association, il ne peut exister qu'une et une seule valeur pour chaque propriété de l'association

Règle 7 : Toutes les propriétés d'une association doivent dépendre pleinement de l'identifiant de la relation.

Associations ternaire

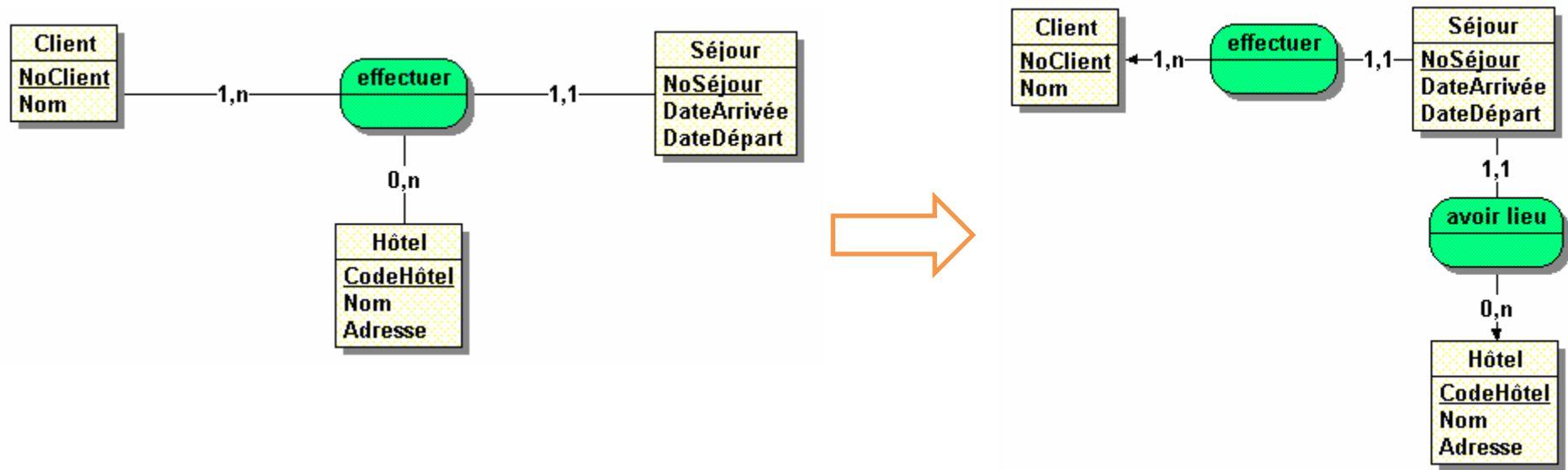
- L'association A6 décrit un lien sémantique entre les entités X, Y et Z.



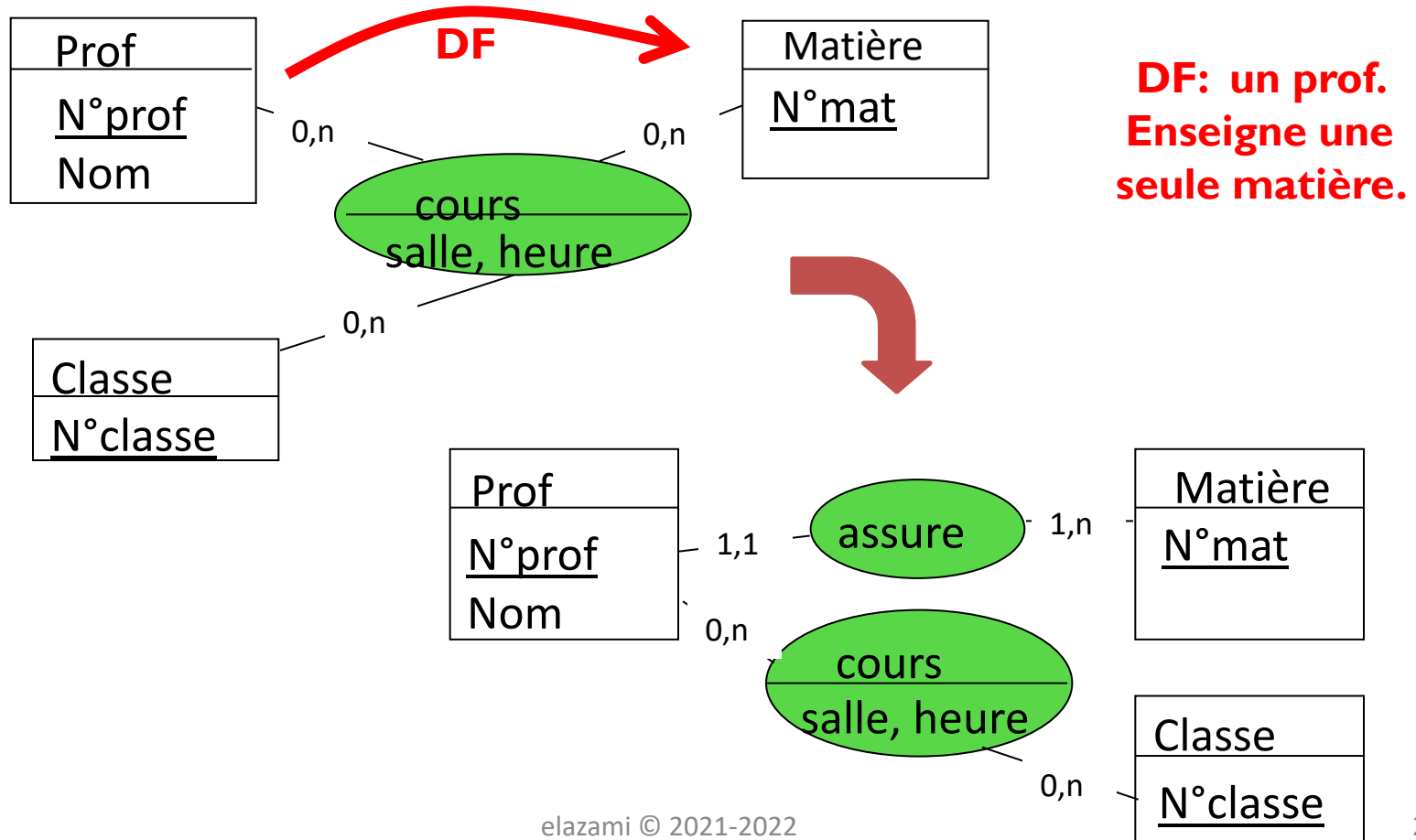
- Difficile à gérer en pratique !!! Il faut essayer d'en avoir le moins possible.

Associations ternaire

- Généralement, si une ou plusieurs des entités liées à une **relation ternaire** possèdent une **cardinalité maximale de 1**, la modélisation n'est pas optimisée
- Il faudrait mieux **décomposer** la relation ternaire, c.à.d. la représenter par 2 relations binaires.



Association ternaire : dépendance fonctionnelle



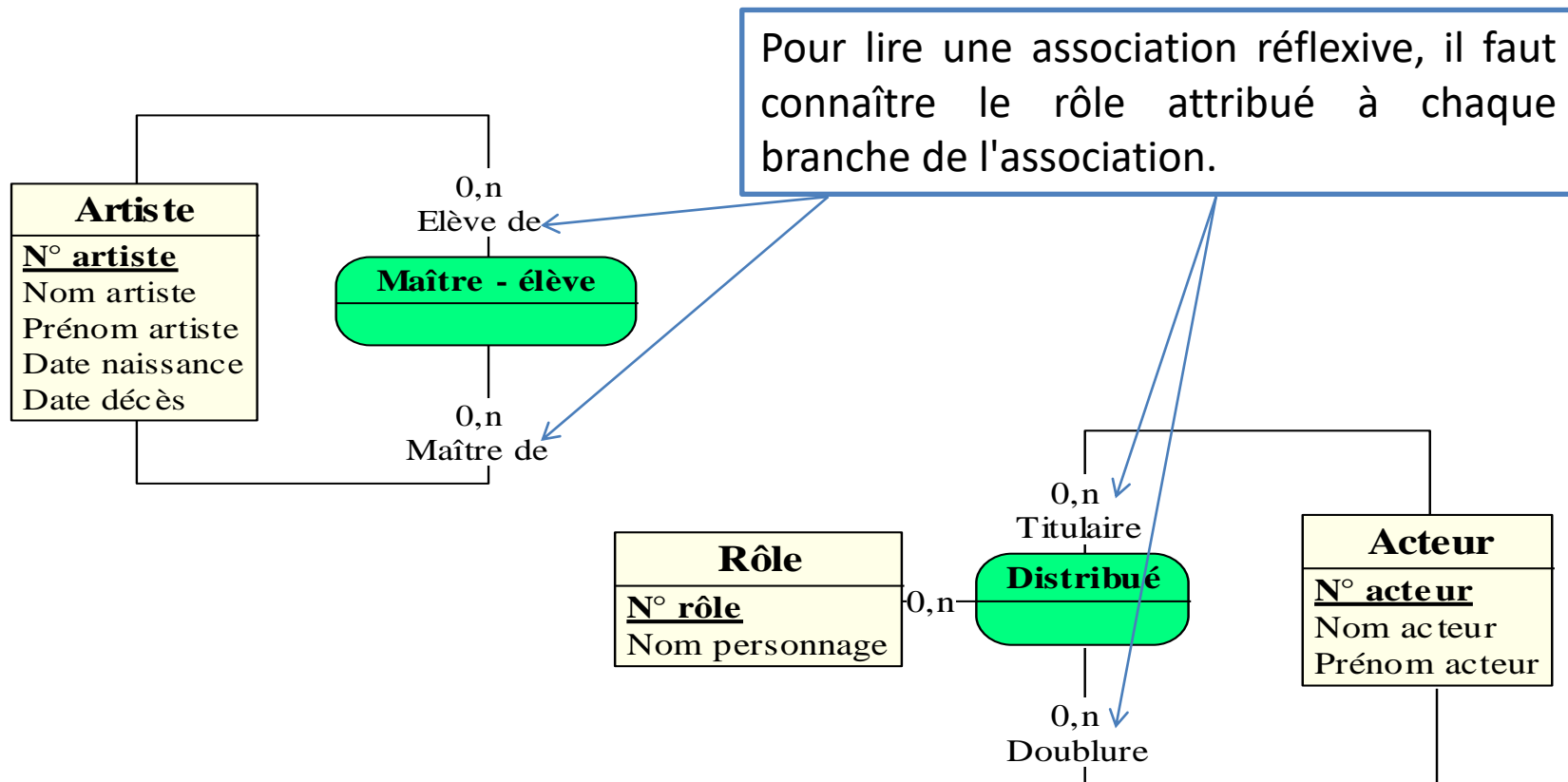
Association ternaire : règle absolue

Pour garder une association ternaire (et au-delà)...

**Toutes les cardinalités maximum
d'une
association ternaire
(et au-delà) doivent être égales à n et
pas de dépendance fonctionnelle**

Association réflexive:

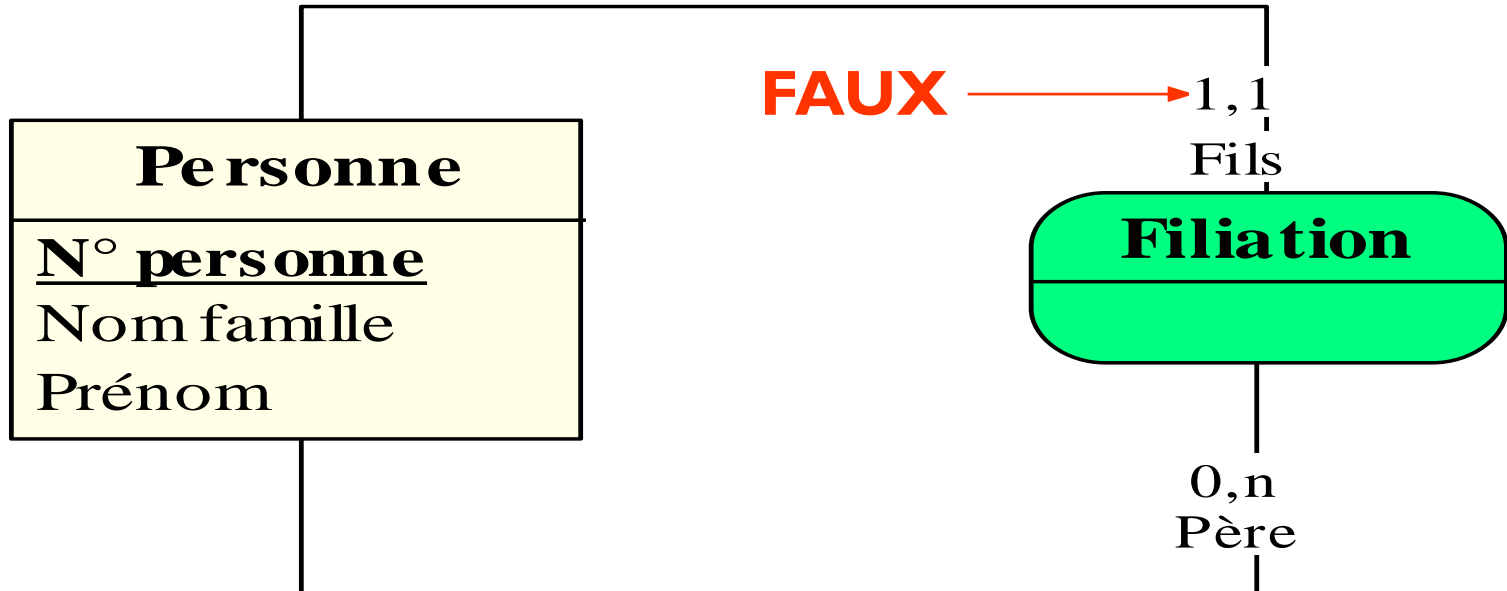
Association entre occurrences d'une même entité (1/2)



Association réflexive:

Association entre occurrences d'une même entité (2/2)

Toutes les cardinalités minimums doivent être égales à zéro !



Règles de gestion

- Elles expriment les **CONTRAINTES D'INTÉGRITÉ** du modèle et représentent les lois de l'univers réel modélisé dans le S.I.
- **Contraintes statiques:** portent sur:
 - une propriété (liste de valeurs possibles ...)
 - plusieurs propriétés d'une même relation ou entité
commande(no,date-cmde,date-livr) avec $date-cmde < dte-livr$
 - des propriété d'entités/relation différentes
 - les cardinalité
 - les dépendances fonctionnelles
- **Contraintes dynamiques :** règles d'évolution
ex: un salaire ne doit pas baisser

Règles de gestion

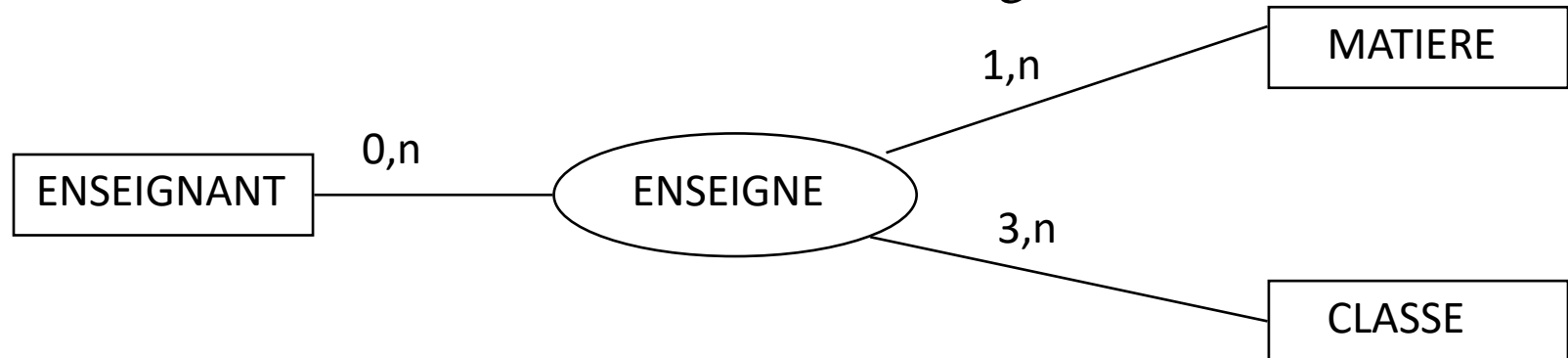
Exemple:

Dans le MCD d'une école les règles de gestion peuvent être les suivants :

RG1 : tout professeur enseigne en principe au moins une matière, mais certains d'entre eux peuvent être dispensés d'enseignement en raison de leurs travaux de recherche

RG2 : toute matière est enseignée dans au moins une classe

RG3 : toute classe a au moins trois enseignants



Dépendance fonctionnelle (DF)

Définition

- On dit que **b** est en dépendance fonctionnelle (DF) de **a** si à une valeur quelconque de la propriété **a**, on ne peut faire correspondre qu'une seule valeur au plus de la propriété **b**.
- On note: **a** → **b**

Exemple :

La dépendance fonctionnelle **CIN** → **NOM** signifie qu'à **un numéro** est associé **un nom seulement**.

*Remarquons qu'une dépendance fonctionnelle n'est généralement pas symétrique, c'est-à-dire que **CIN** → **NOM** n'interdit pas que deux personnes distinctes (correspondant à deux CIN différents) portent le même nom.*

Dépendance fonctionnelle (DF)

Dépendance fonctionnelle élémentaire:

on parle de **DFE** entre les propriétés **a** et **b** ($a \rightarrow b$) si ($a \text{ --- df--- } b$) et si aucune partie de **a** ne détermine **b**

Exemples :

- $\text{Code-client} + \text{Nom-Client} \text{ --- df --- } \text{Adresse-Client}$
n'est pas élémentaire, puisque la connaissance de **Code-Client** (partie de $\text{Code-Client} + \text{Nom-Client}$) suffit à déterminer l'adresse
- $\text{Code-client} \text{ --- df --- } \text{Adresse-Client}$
est élémentaire et on peut écrire: $\text{Code-Client} \rightarrow \text{Adresse-Client}$

Dépendance fonctionnelle (DF)

Dépendance fonctionnelle élémentaire directe:

on dit que la propriété **b** dépend fonctionnellement de **a** par une dépendance fonctionnelle élémentaire directe si cette dépendance est élémentaire $a \rightarrow b$ et s'il n'existe pas de propriété **c** telle que :

$a \text{ --- df --- } > c \text{ et } c \text{ --- df --- } > b$ (c-à-d on élimine toute **transitivité**)

Exemple :

- $N^{\circ}\text{Prof} \text{ ---- } > \text{Code-matière}$
- $\text{Code-matière} \text{ ---- } > \text{Nom-matière}$
- $N^{\circ}\text{Prof} \text{ --- } > \text{Nom-matière}$

Les 2 premières dépendances fonctionnelles sont directes, mais la troisième ne l'est pas en raison de la transitivité :

$N^{\circ}\text{Prof} \text{ --- } > \text{Code-matière} \text{ ----- } > \text{Nom-matière}.$

Dépendance fonctionnelle (DF)

Notion de clé

Une clé est une propriété (ou une concaténation de propriétés) d'une entité telle que **toutes les autres propriétés de l'entité dépendent d'elle fonctionnellement** et telle que ce ne soit plus vrai pour aucune de ses parties.

Dépendance fonctionnelle entre entités

- notée $A \rightarrow B$
- si toute occurrence de A est déterminée par **une et une seule** occurrence de B
- NB: les cardinalités 1-1 correspondent toujours à une DF

Propriétés de DF

Réflexivité

$$a \xrightarrow{\text{df}} a$$

Projection

$$a \xrightarrow{\text{df}} b+c \implies a \xrightarrow{\text{df}} b \text{ et } a \xrightarrow{\text{df}} c$$

Augmentation

$$a \xrightarrow{\text{df}} b \implies \forall c : a+c \xrightarrow{\text{df}} b$$

Additivité

$$a \xrightarrow{\text{df}} b \text{ et } a \xrightarrow{\text{df}} c \implies a \xrightarrow{\text{df}} b+c$$

Transitivité

$$a \xrightarrow{\text{df}} b \text{ et } b \xrightarrow{\text{df}} c \implies a \xrightarrow{\text{df}} c$$

Pseudo-transitivité

$$a \xrightarrow{\text{df}} b \text{ et } b+c \xrightarrow{\text{df}} d \implies a+c \xrightarrow{\text{df}} d$$

Propriétés de DF: Exemples

Réflexivité

$$\text{REF} \xrightarrow{\text{df}} \text{REF}$$

Projection

$$\text{REF} \xrightarrow{\text{df}} \text{DESIGN} + \text{PU} \implies \text{REF} \xrightarrow{\text{df}} \text{DESIGN}$$

Augmentation

$$\text{REF} \xrightarrow{\text{df}} \text{PU}$$

$$\text{REF} \xrightarrow{\text{df}} \text{PU} \implies \text{REF} + \text{DESIGN} \xrightarrow{\text{df}} \text{PU}$$

Additivité

$$\text{REF} \xrightarrow{\text{df}} \text{DESIGN}$$

$$\text{et } \text{REF} \xrightarrow{\text{df}} \text{PU}$$

$$\implies \text{REF} \xrightarrow{\text{df}} \text{DESIGN} + \text{PU}$$

Transitivité

$$\text{REF} \xrightarrow{\text{df}} \text{CODETVA}$$

$$\text{et } \text{CODETVA} \xrightarrow{\text{df}} \text{TXTVA}$$

$$\implies \text{REF} \xrightarrow{\text{df}} \text{TXTVA}$$

Pseudo-transitivité

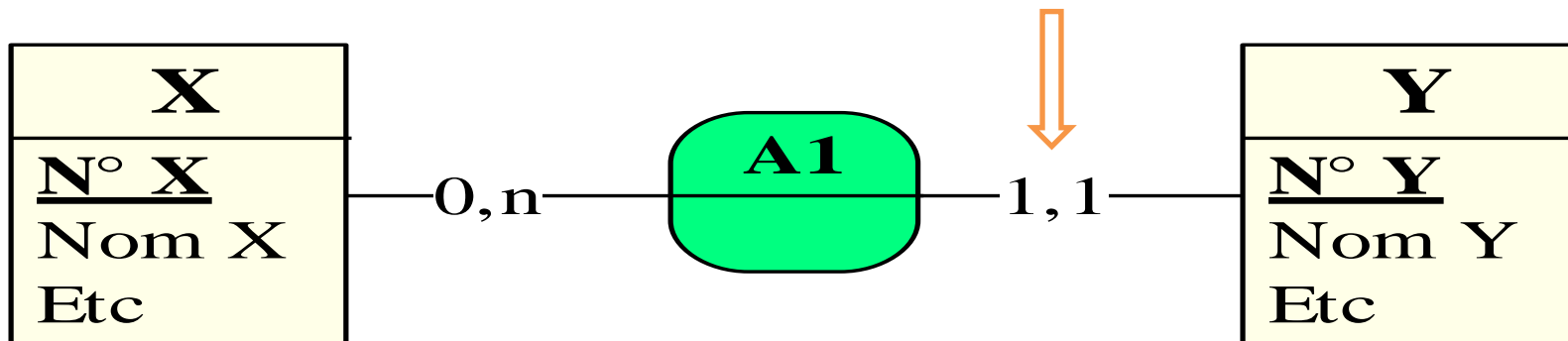
$$\text{REF} \xrightarrow{\text{df}} \text{CODETVA}$$

$$\text{CODETVA} + \text{PU} \xrightarrow{\text{df}} \text{TXTVA}$$

$$\implies \text{REF} + \text{PU} \xrightarrow{\text{df}} \text{TXTVA}$$

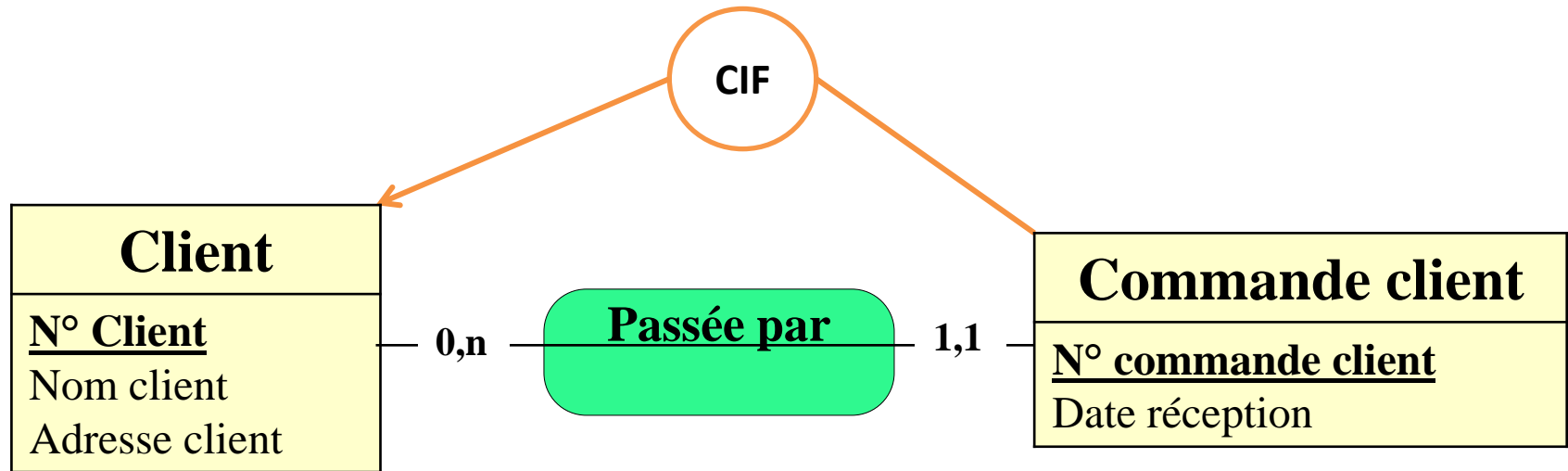
Association binaire fonctionnelle

Lorsqu'une association-type binaire a une patte dont la **cardinalité maximale** est **égale à 1**, alors elle est dite **fonctionnelle**.

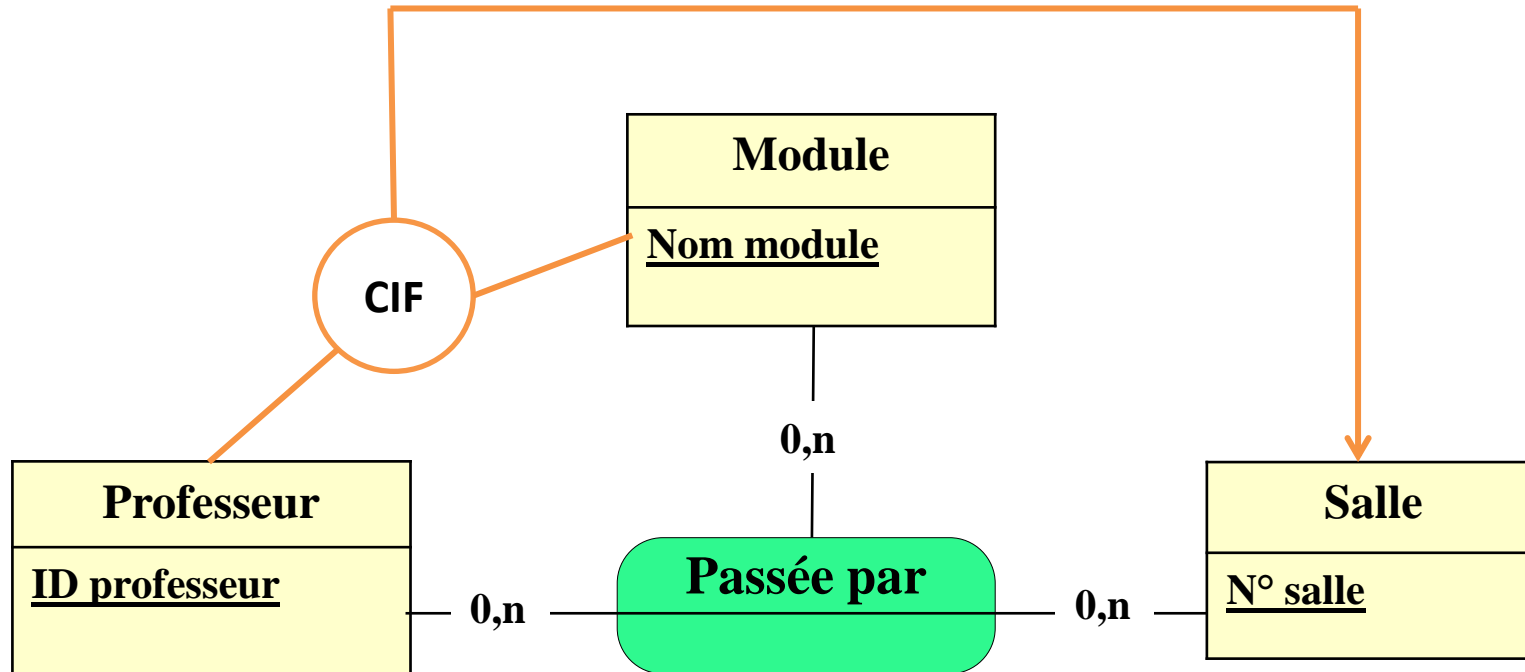


Contraintes d'Intégrité Fonctionnelle (CIF)

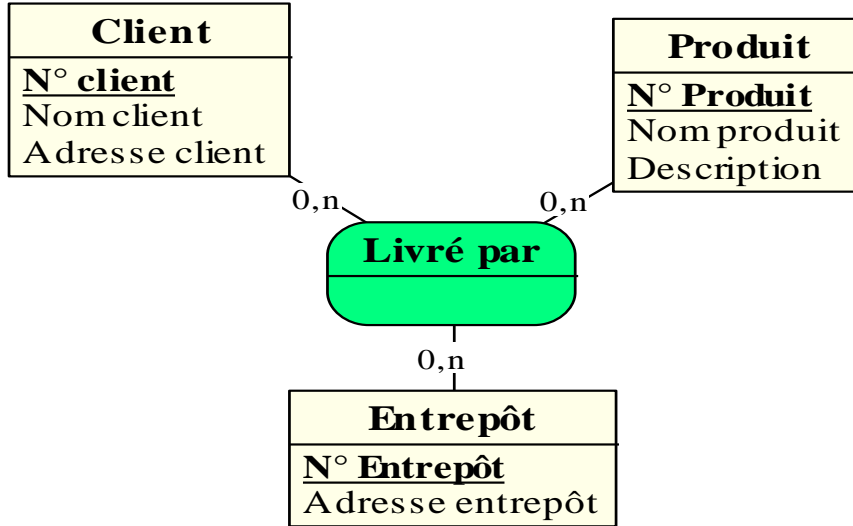
- Sur association-type **binaire** :



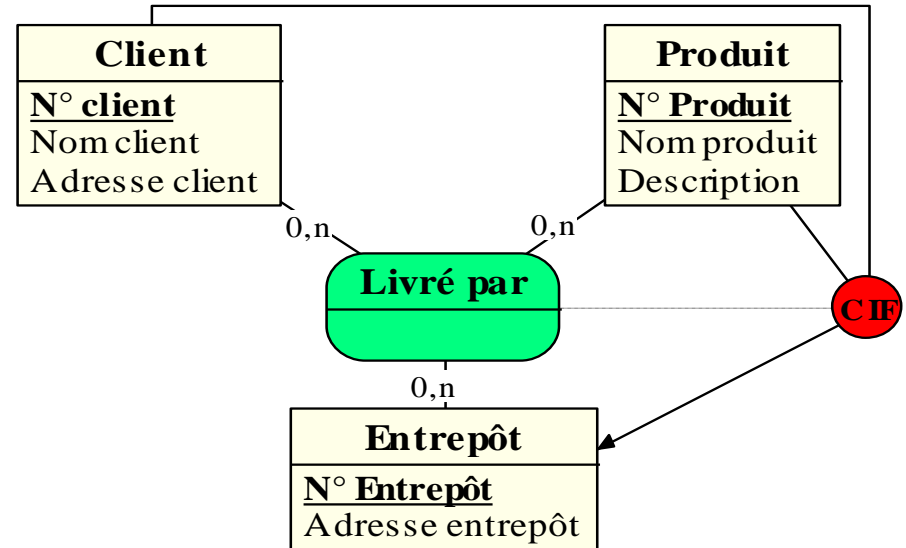
CIF : cas d'une association non binaire



CIF : cas d'une association non binaire (Contre exemple)



Comment garantir l'unicité de l'entrepôt pour un couple (Client,Produit) ?



Exercices:

Exercice 1: Formation

On veut gérer **les étudiants** et **les professeurs** d'un ensemble de **formations** dispensés par une université.

- Un **étudiant** est identifié par son **nom**, son **prénom**, et son **âge**.
- Un étudiant suit une formation.
- Chaque **formation** a un **nom** et une **durée** (nombre d'années).
- Elle est assurées par un ensemble d'enseignants.
- Chaque **enseignant** est connu par son **nom**, son **prénom** et la matière qu'il enseigne.
- On désire savoir le nombre d'heurs qu'un enseignant effectue pour chacune des formations dans lesquelles il enseigne.

Donner le modèle **Entité-Association**.

Exercices:

Exercice 2 : Bibliothèque

- Le système informatique d'une bibliothèque enregistre **le numéro national, le nom, le prénom et l'adresse (rue, numéro, code postal, ville)** de chacun de ses **clients**.
- Le système enregistre pour chaque **livre** disponible son numéro unique **ISBN**, son **nom, le nom de ses auteurs** et sa **date d'achat**.
- On suppose que la bibliothèque possède **au plus** un livre par numéro **ISBN**.
- Le système enregistre également tous **les emprunts** des clients.
- On identifie ces emprunts avec **un numéro unique**, on retient **la date d'emprunt** et **la date de retour** lorsque le livre rentre.

Donner un modèle **entité-association** de ce système et préciser **les contraintes d'intégrité**.

Exercices:

Exercice 3 : Employés

- Un employé est identifié par son **numéro de sécurité sociale (SSN)** et a **un nom** et **un prénom**.
- Chaque employé peut **être supervisé** par **au plus** un autre employé.

Donner un modèle **entité-association** de ce problème et préciser les contraintes d'intégrité.