

## **Chapitre 3**

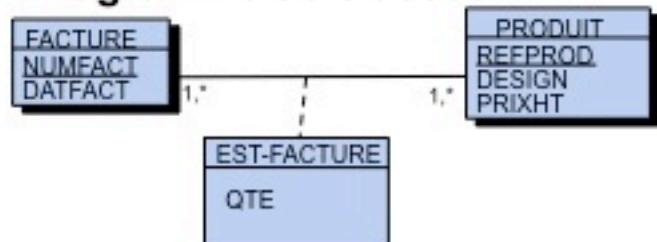
de l'UML vers une BD relationnelle

# Étapes de la conception et réalisation d'une BD

## 1. Dico. des données

Attribut	Signification	Domaine
REFPROD	Référence du produit	Chaîne(12)
DESIGN	Désignation du produit	Chaîne(30)
PRIXHT	Prix unitaire HT	réel
...	...	...

## 3. Diagramme de classes

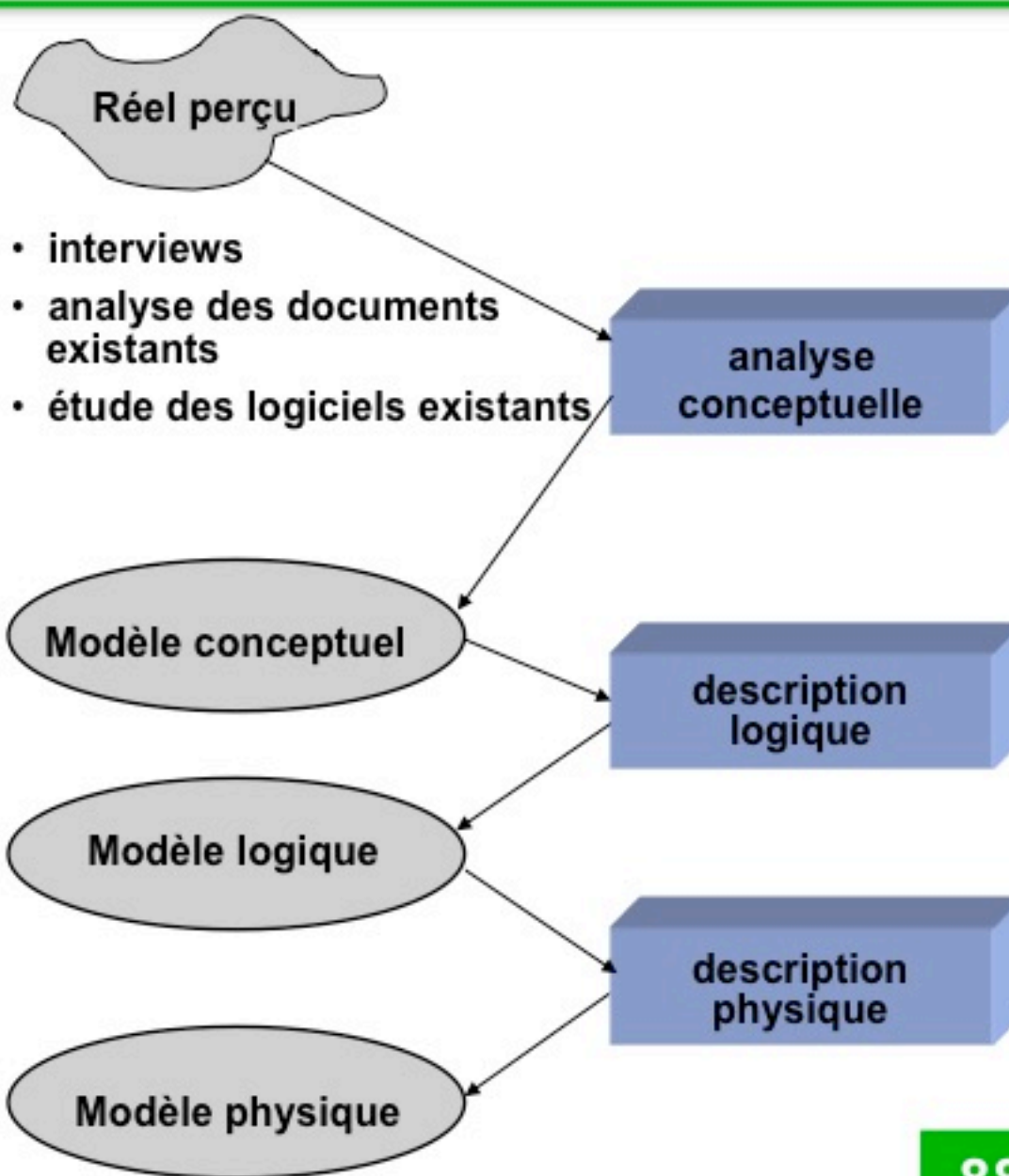


## 4. Schéma de la BD

FACTURE(NUMFACT,DATACT)  
 PRODUIT(REFPROD,DESIGN,PRIXHT)  
 EST-FACTURE(NUMFACT,REFPROD,QTE)

## 5. Mise en œuvre sur le SGBD

FACTURE : Table		
	Nom du champ	Type de données
🔍	NUMFACT	Numérique
📄	DATACT	Date/Heure

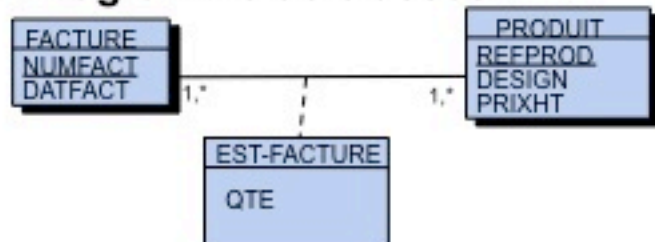


# Étapes de la conception et réalisation d'une BD

## 1. Dico. des données

Attribut	Signification	Domaine
REFPROD	Référence du produit	Chaîne(12)
DESIGN	Désignation du produit	Chaîne(30)
PRIXHT	Prix unitaire HT	réel
...	...	...

## 3. Diagramme de classes

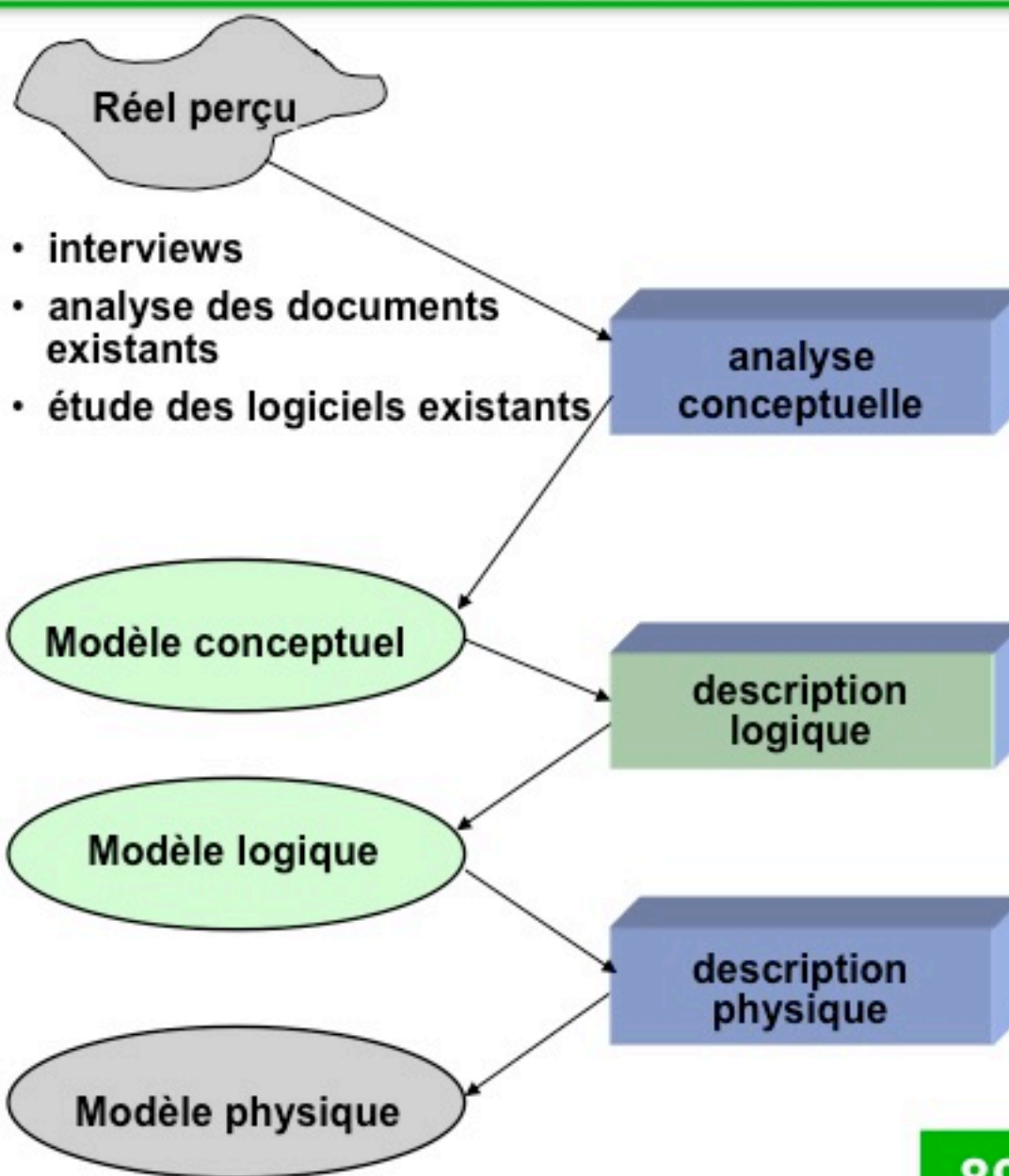


## 4. Schéma de la BD

FACTURE(NUMFACT,DATACT)  
PRODUIT(REFPROD,DESIGN,PRIXHT)  
EST-FACTURE(NUMFACT,REFPROD,QTE)

## 5. Mise en œuvre sur le SGBD

FACTURE : Table		
	Nom du champ	Type de données
🔍	NUMFACT	Numérique
🔍	DATACT	Date/Heure



# Modèle Logique de Données (MLD)

---

- ❖ MLD : Un modèle de données découlant d'un modèle conceptuel mais qui le raffine pour tenir compte des caractéristiques du type de SGBD utilisé pour la réalisation de la BD

*Employé*

no_employé	nom_employé	prénom_employé	salaire_annuel	no_département	← Clé étrangère
BLO001	Blouin	Paul	55 000 \$	10	
CAN020	Cantin	Sophie	58 000 \$	20	
TRA034	Trahan	Pierre	60 000 \$	20	

*Département*

no_département	nom_département	nb_employés
10	Comptabilité	13
20	Recherche et développement	8

- ❖ Objectifs de la phase
  - Règles et principes assurant la conversion d'un modèle conceptuel en un modèle logique de données de type relationnel
  - Détail de la formulation d'un modèle relationnel de données dans la notation UML



## Dérivation des relations à partir d'un diagramme de classes UML

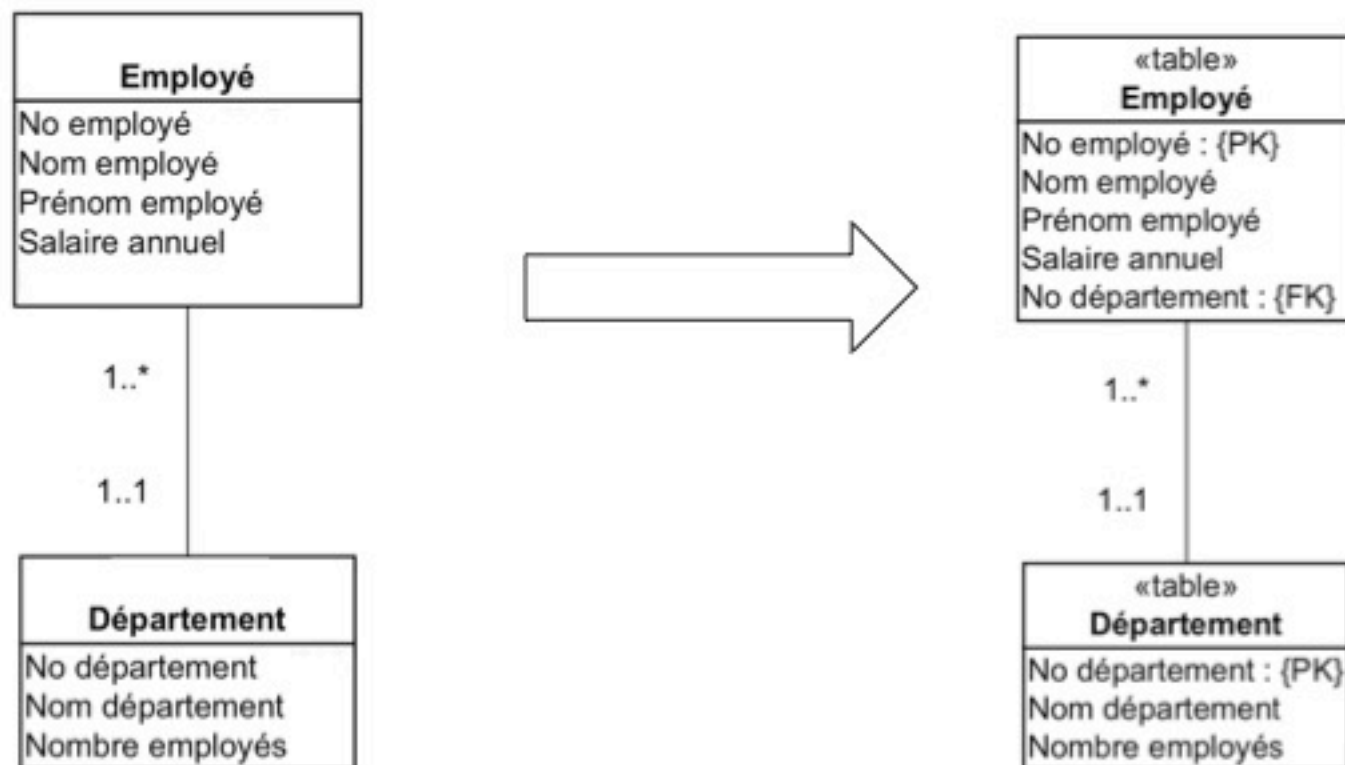
---

- ❖ Dans le MLD, une classe est stéréotypée « Table »
- ❖ Les associations entre les tables n'ont pas à être nommées. Elles indiquent simplement qu'il existe un lien entre les deux tables et que la clé primaire d'une table marquée {PK} (pour *Primary Key*) est reproduite dans l'autre table sous forme de clé étrangère. Une clé étrangère est marquée {FK} (pour *Foreign Key*)
- ❖ Les associations **sont toutes des associations binaires** portant des contraintes de multiplicité entre des tables



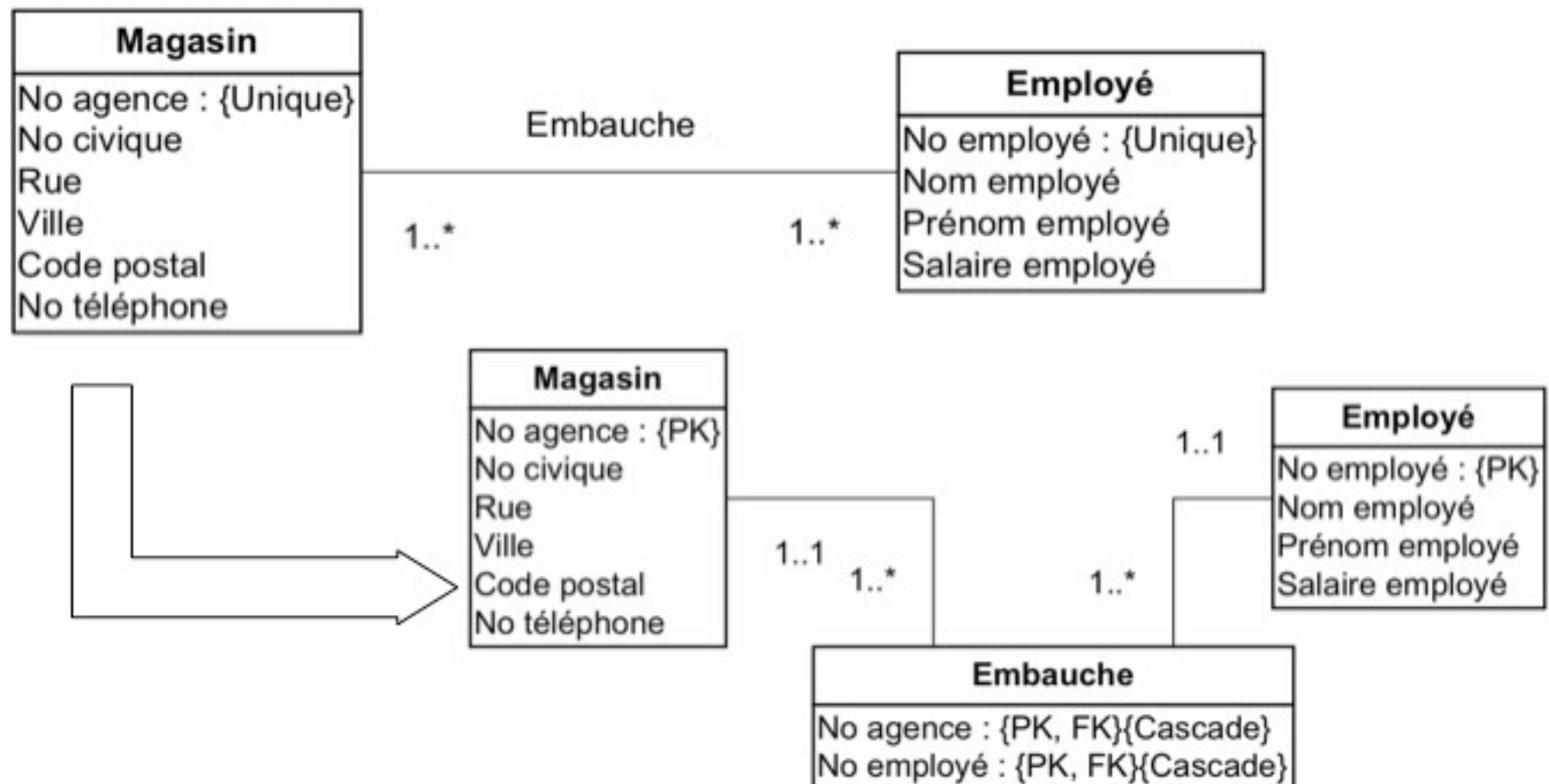
# Dérivation à partir d'une association binaire 1 à n

- ❖ **Règle 1** : la table dérivée de l'entité du côté de la multiplicité maximale (n) est considérée la table *filles*. En conséquence, une copie de la clé primaire {PK} (pour *Primary Key*) de la deuxième table, la table *mère*, est déposée dans la table *filles* à titre de clé étrangère {FK} (pour *Foreign Key*)



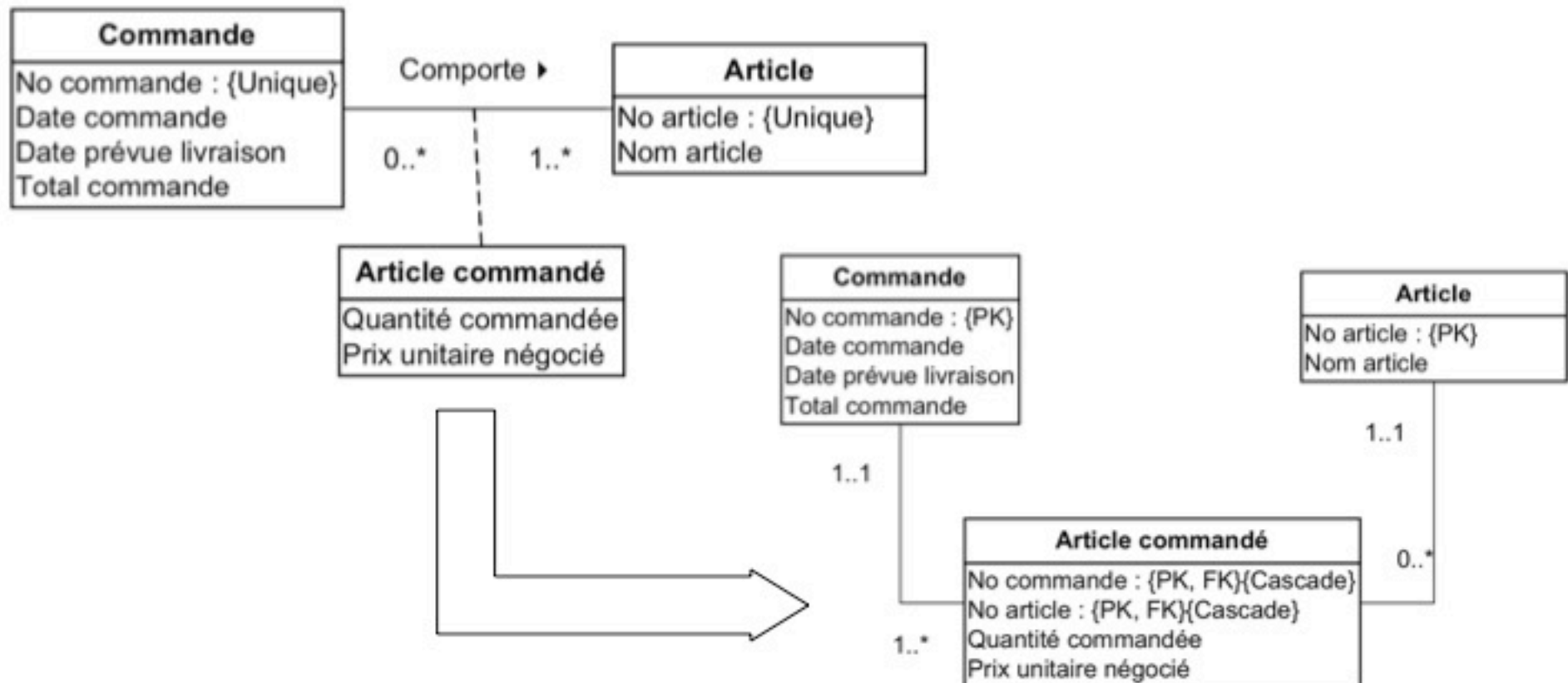
## Dérivation à partir d'une association binaire n à n

- ❖ **Règle 2 :** Une association *n à n* sans classe association donne lieu à une table fille comportant comme clé primaire la combinaison des clés primaires des tables mères. La mention {Cascade} comporte des restrictions supplémentaires : si une ligne de clé primaire d'une ligne est modifiée, les valeurs des clés étrangères des lignes associées dans la table fille sont modifiées



# Dérivation à partir d'une classe association

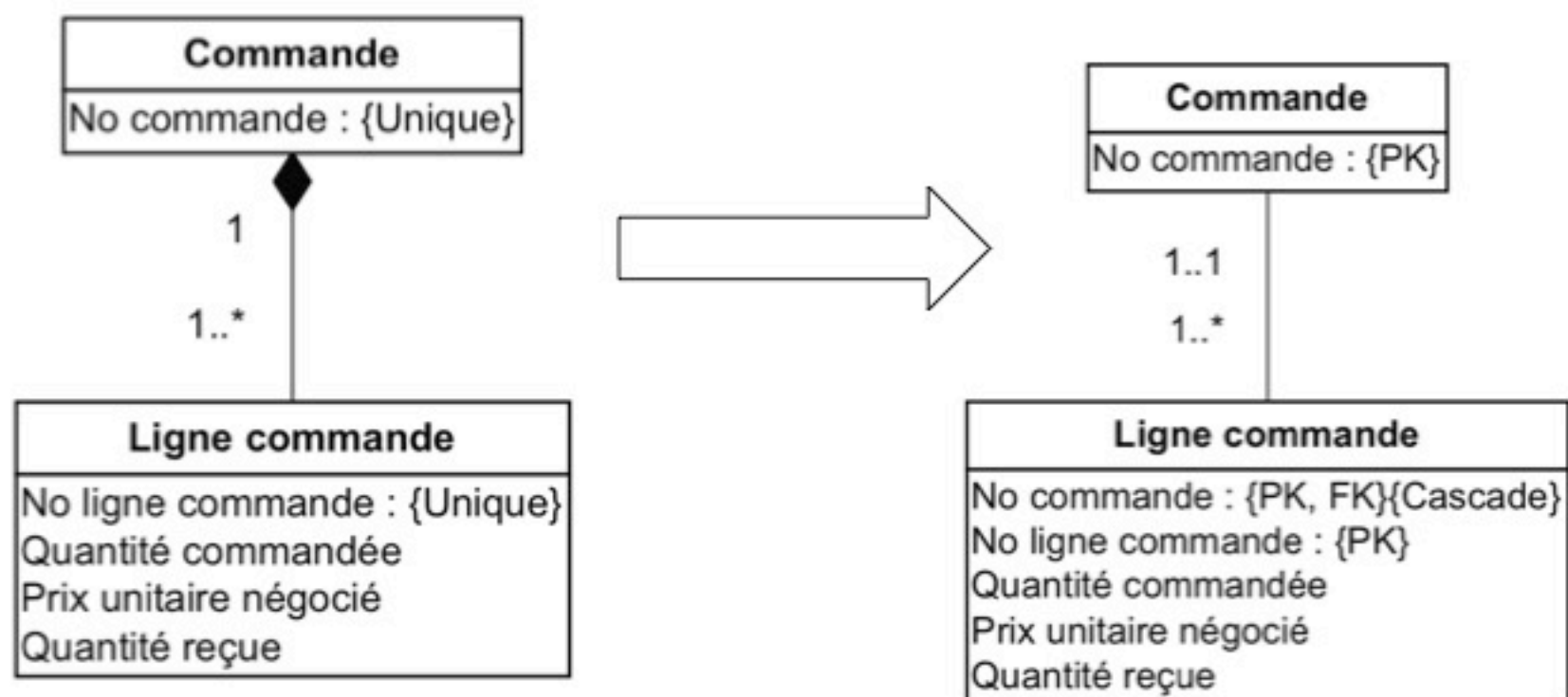
- ❖ **Règle 3** : La table correspondante aura comme clé primaire une clé obligatoirement composée des attributs de tous les identifiants des classes participant à l'association et chaque attribut de la clé composée portera la mention {PK, FK}





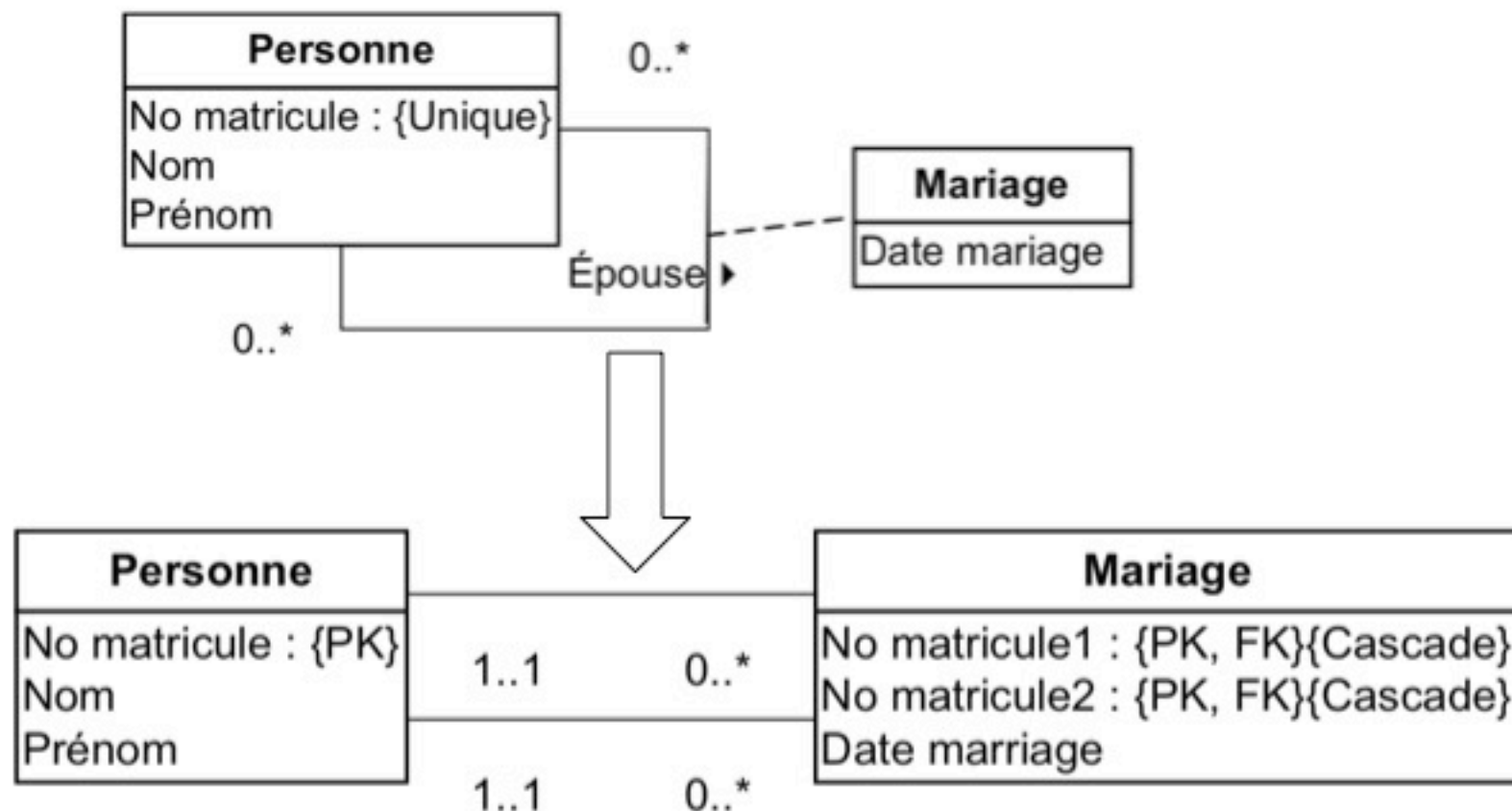
## Dérivation à partir d'une relation de composition

- ❖ **Règle 4** : Un composant lié à une classe composite a un identifiant partiellement défini par son composite. En effet, elle possède son propre identifiant qui garantit une identification unique des occurrences d'un même composite mais il ne peut assurer une identification unique de toutes les occurrences de l'entité en général. Pour ce faire, son identifiant doit être combiné à l'identifiant du composite



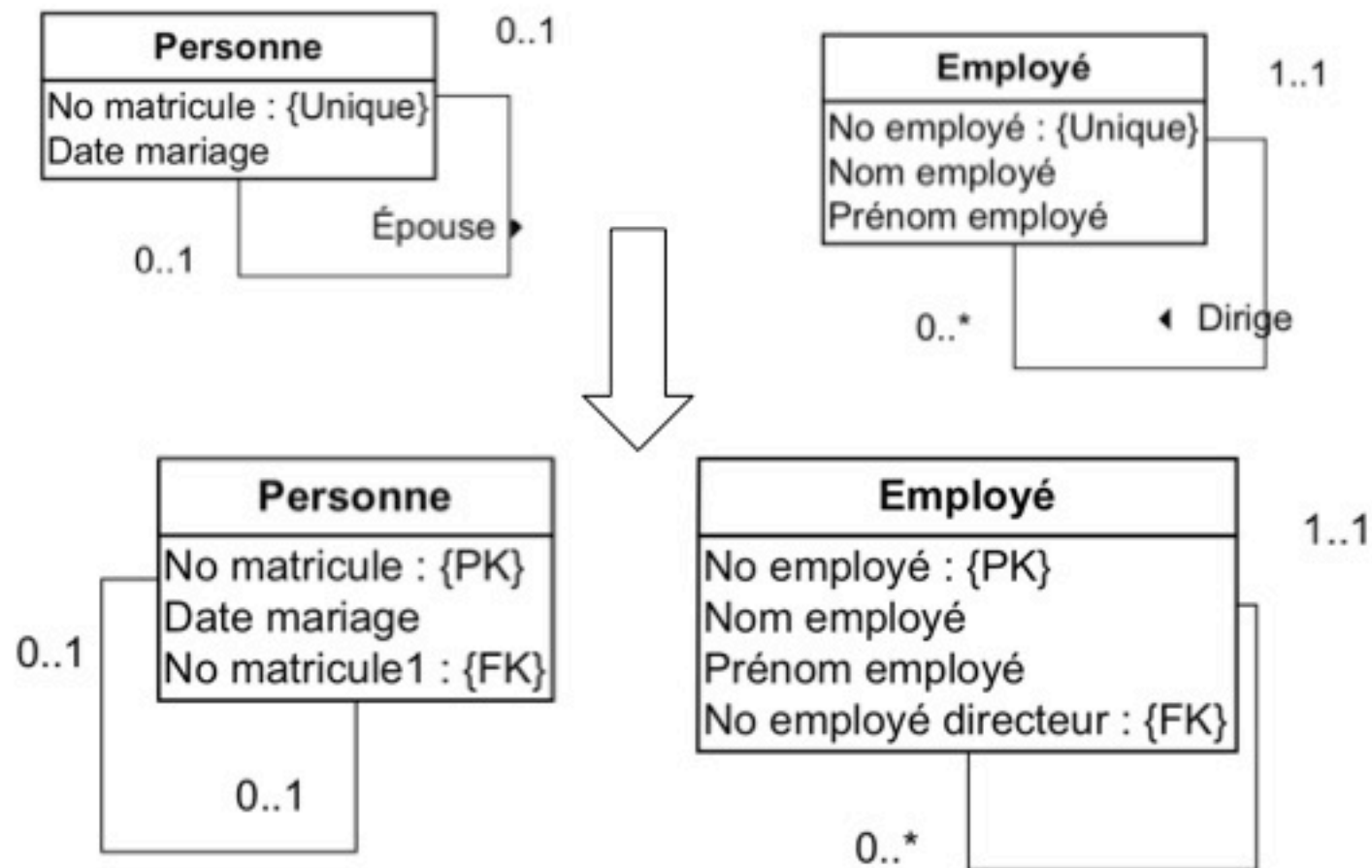
## Dérivation d'une association réflexive comportant une classe association

- ❖ **Règle 5 :** la classe association donne lieu à une table fille associée à la table mère dérivée de la seule entité en cause. La table dérivée de la classe association comporte les attributs de cette dernière avec comme clé primaire deux exemplaires de la clé primaire de la table mère, les exemplaires portent un nom différent pour éviter toute redondance



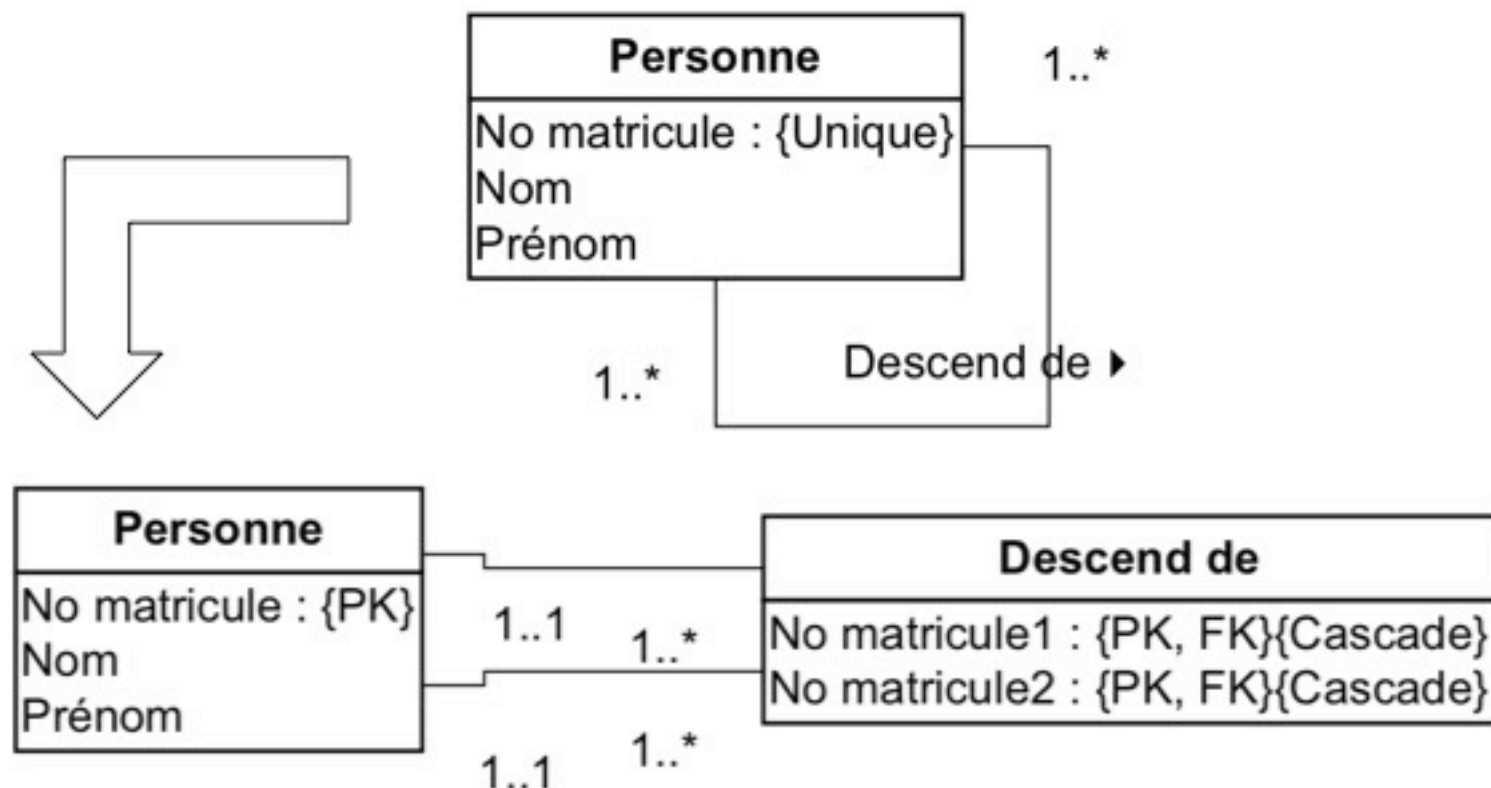
## Dérivation d'une association réflexive de type 1 à 1 ou 1 à n

- ❖ **Règle 6 :** Une seule table est dérivée et elle comporte une clé étrangère. Celle-ci est un double de sa clé primaire qui doit porter un nom différent. Les multiplicités de l'association au modèle conceptuel sont reprises intégralement dans le modèle relationnel dérivé



## Dérivation d'une association réflexive de type n à n

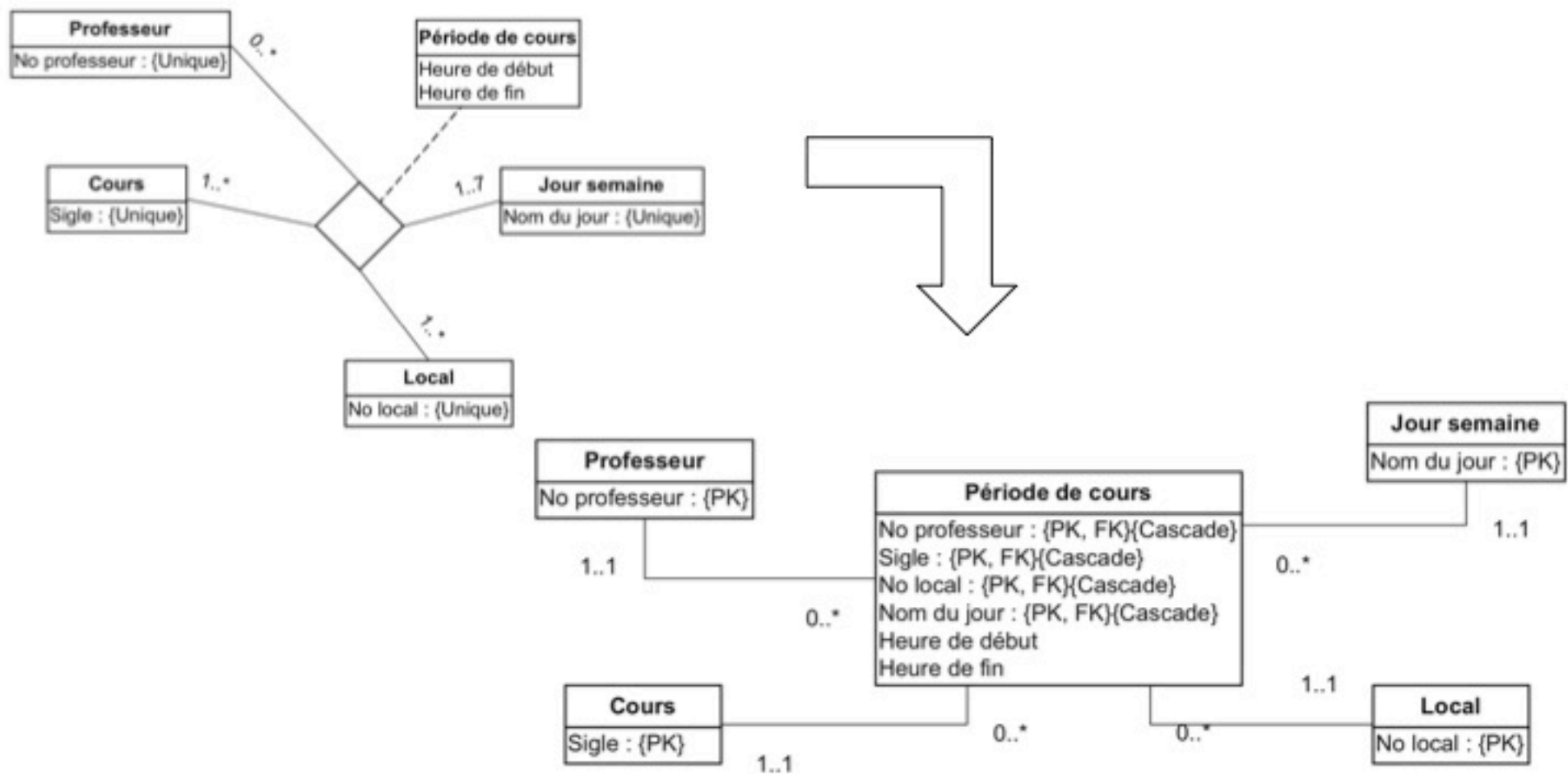
- ❖ **Règle 7** : Une table doit être créée pour assurer une liaison n à n sur la même table. Cette table n'a pas d'attribut propre. Sa clé primaire est une combinaison des deux exemplaires de la clé primaire de la table dérivée de la classe. Chaque exemplaire aura un nom différent et sera traité comme une clé étrangère avec contrainte d'intégrité référentielle en ajout, suppression et mise à jour (mention {Cascade})





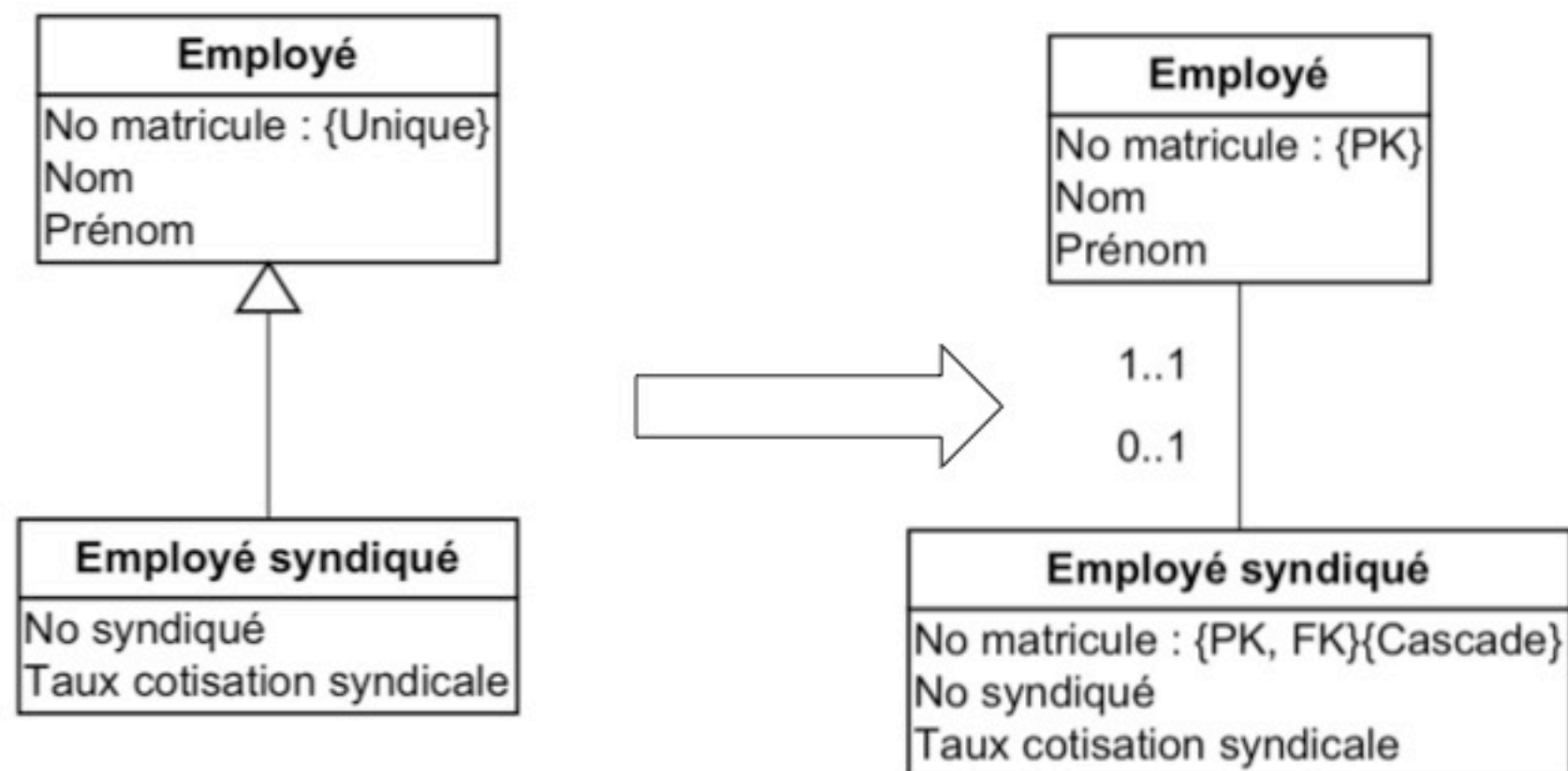
## Dérivation d'une association n-aire

- ❖ **Règle 8** : Chaque classe associée devient une table mère; la classe association devient une table fille; la clé primaire de la table fille combine les clés primaires des tables dérivées des tables mères associées et chaque élément de la clé est une clé étrangère



## Dérivation d'une association d'héritage

- ❖ **Règle 9** : il faut créer une table mère pour le supertype et une table fille pour chaque sous-type. L'identifiant du supertype devient la clé primaire de la table mère et aussi clé étrangère avec intégrité référentielle (ajout, suppression, mise à jour) pour la classe fille. Multiplicités 1..1 côté mère, 0..1 côté fille



# Modèle relationnel : Avantages et faiblesses

---

## ❖ Modèle relationnel

- le schéma définit la structure de la relation
- les n-uplets représentent les différents éléments
- les tables respectent des propriétés définies sur les dépendances fonctionnelles ( en général, troisième forme normale (3NF) )

## ❖ Avantages du modèle relationnel et des SGBD relationnels

- Organisation structurée des données
- Permanence des données
- Accessibilité par des utilisateurs concurrents
- Bien implanté dans le monde professionnel

## ❖ Faiblesses du modèle relationnel

- Absence de pointeurs visibles : pour lier des données qui se correspondent, on a besoin de faire des jointures (opérations coûteuses)
- Non support des domaines composés : on ne peut pas avoir par exemple un attribut qui correspond à une adresse avec le numéro de la rue, le nom de la rue, la ville,...impossible à cause de la première forme normale, qui impose l'atomicité des attributs