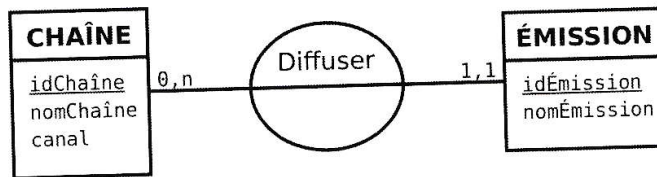


## 2

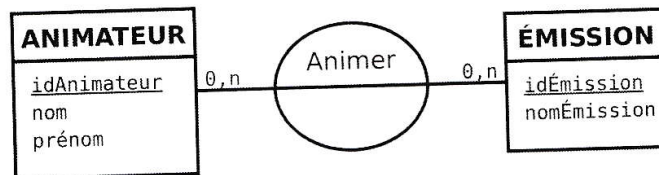
## Correction du TD 2.4 - Modèle entités-associations (1<sup>re</sup> partie)

### ★ Réponse à la question 1



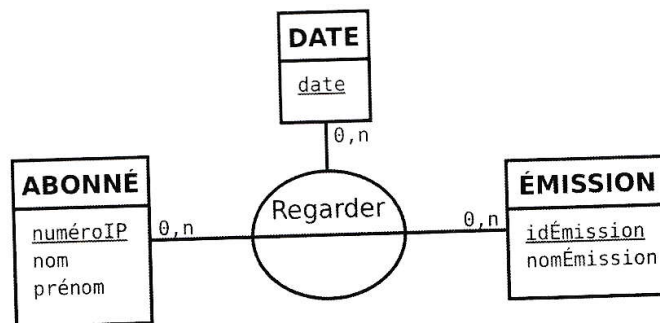
L'identifiant de *Diffuser* est *idÉmission*.

### ★ Réponse à la question 2



L'identifiant de *Animer* est le couple  $(idAnimateur, idÉmission)$ .

### ★ Réponse à la question 3



L'identifiant de *Animer* est le couple  $(numéroIP, idÉmission, date)$ .

**Vérifier la minimalité de la clé !**

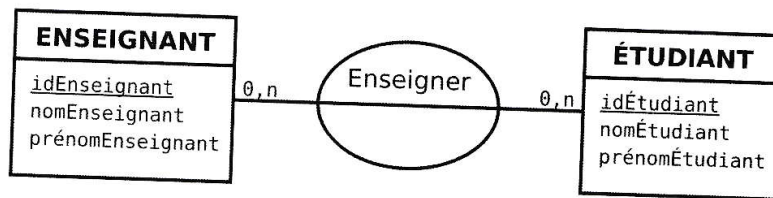
### ★ Réponse à la question 4

Les problèmes de cette modélisation sont nombreux :

- l'attribut *étudiants* est un attribut multiple (car il y a généralement plusieurs étudiants) ce qui n'est pas autorisé dans un modèle E-A (cf. règle 2.27) ;
- ce modèle enfreint la règle 2.13 qui précise qu'un type-entité et ses attributs doivent ne traiter que d'un seul sujet ;
- un étudiant qui suit plusieurs enseignements sera dupliqué pour chacun de ces enseignements.
- que se passe-t-il si deux enseignants possèdent le même nom et le même prénom ?
- que se passe-t-il si un même enseignant dispense deux enseignements ?

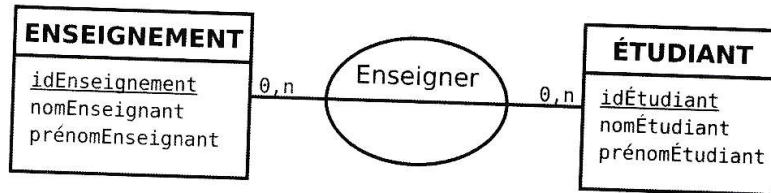
### ★ Réponse à la question 5

Certains étudiants pensent à :



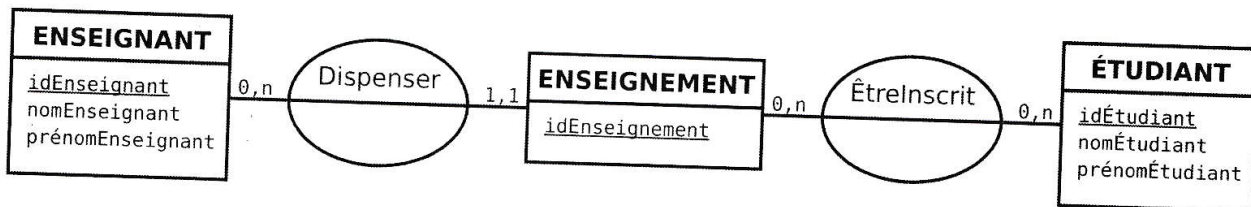
Le problème de ce modèle est que le type-entité enseignement n'existe plus et ne peut être déduit. En effet, il est impossible de répondre à la simple question : quelle est la liste des enseignements ? De plus, tous les étudiants associés à un même enseignant peuvent très bien se répartir sur deux enseignements différents.

Une autre réponse est :

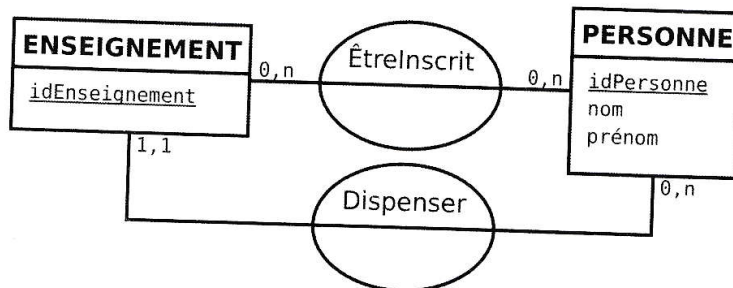


Ici, le type-entité enseignement existe bien. Par contre, comment savoir si deux enseignements sont dispensés par des homonymes ou par une même personne ?

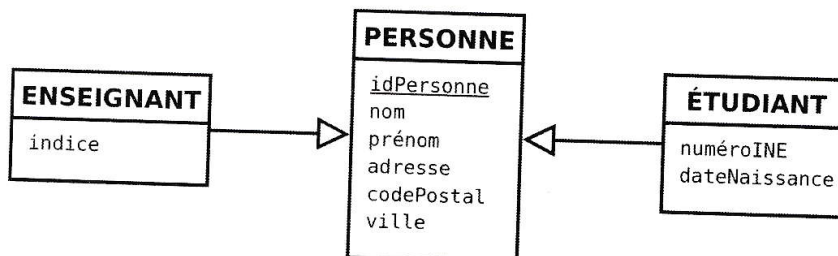
Il vaut donc mieux partir sur une solution du type :



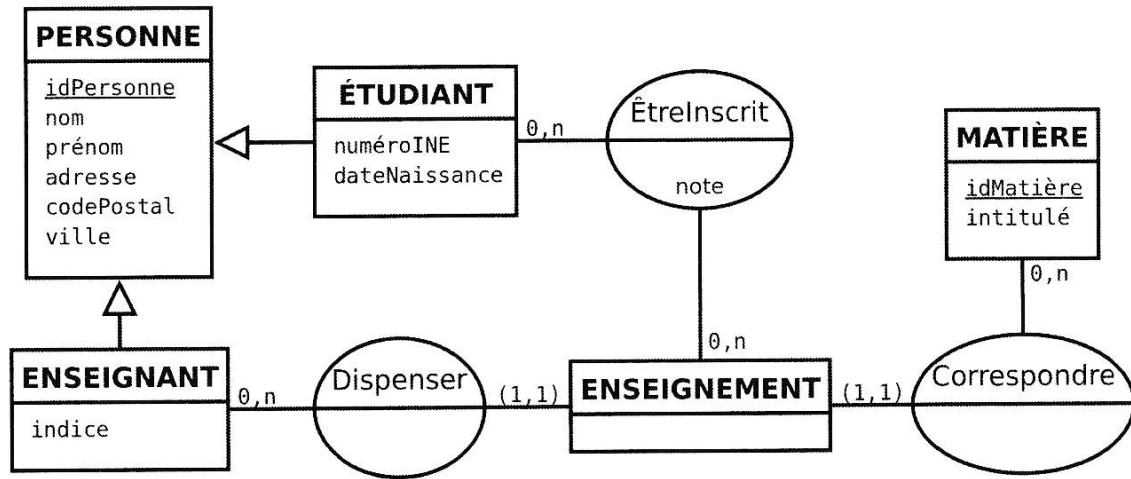
ou en utilisant des type-associations pluriels :



## ★ Réponse à la question 6



## ★ Réponse à la question 7



Nous verrons section 2.8.5 que le type-entité *ENSEIGNEMENT* devrait être avantageusement remplacé par un agrégat.

## ★ Réponse à la question 8

Identification des type-entités :

Extrait	Entité	Attribut	Remarque
... un <b>abonné</b> devra pouvoir ... retrouver le <u>nom</u> des <b>abonnés</b> détenant ...	ABONNÉ	nom prénom adresse	Les attributs <i>prénom</i> et <i>adresse</i> n'apparaissent pas explicitement dans le texte, mais sont évidents.
... retrouver un <b>livre</b> en connaissant son <u>titre</u> . ... la liste des <b>livres</b> ... emprunter au maximum dix <b>ouvrages</b> ...	LIVRE	titre	Les termes <i>ouvrage</i> et <i>livre</i> renvoient au même type-entité.
...connaître la liste des livres d'un <b>auteur</b> ...	PERSONNE	nom prénom	Si nous considérons qu'une personne possède un nom et un prénom, un auteur est une personne et un abonné est une personne qui possède en plus une adresse. D'où le choix de spécialiser <i>ABONNÉ</i> à partir de <i>PERSONNE</i> et de donc de ne pas appeler <i>PERSONNE AUTEUR</i> .
... est <i>acheté</i> en un ou plusieurs <b>exemplaires</b> .	EXEMPLAIRE	dateAchât	La date d'achat peut constituer une information utile qui se modélise comme un attribut de type-entité <i>LIVRE</i>

Identification des type-associations :

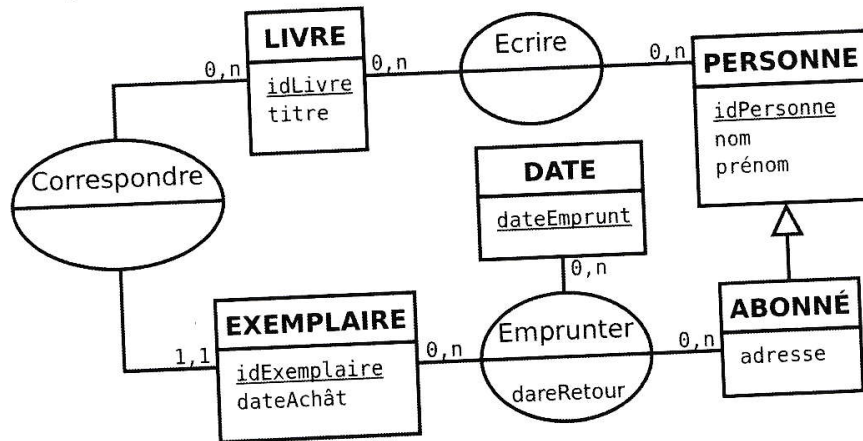
Extrait	Association	Attribut	Remarque
... connaître la liste des livres d'un auteur ... Un livre peut être écrit par plusieurs auteurs.	Ecrire		Il existe un type-association maillé ( <i>i.e.</i> $n$ vers $n$ ) entre <i>LIVRE</i> et <i>PERSONNE</i> ( <i>i.e.</i> auteur).
... Chaque livre est acheté en un ou plusieurs exemplaires ...	Correspondre		Un exemplaire correspond à un livre et un livre peut exister en plusieurs exemplaires : type-association fonctionnel ( <i>i.e.</i> 1 vers $n$ ) entre <i>EXEMPLAIRE</i> et <i>LIVRE</i> .
... Un abonné a le droit d'emprunter au maximum dix ouvrages ... Les prêts sont accordés pour une durée de quinze jours ... connaître [...] la liste des livres détenus [...] et [...] retrouver le nom des abonnés détenant un livre ...	Emprunter	dateEmprunt dateRetour	Un exemplaire peut être emprunté $n$ fois (successivement) et un abonné peut emprunté 10 (donc $n$ ) livres : il existe un type-association maillé entre <i>EXEMPLAIRE</i> et <i>ABONNÉ</i> . Pour mesurer la durée du prêt, <i>Emprunter</i> doit posséder un attribut <i>dateEmprunt</i> . Pour savoir si un exemplaire est en cours de prêt ou a été rendu, <i>Emprunter</i> doit également posséder un attribut <i>dateRetour</i> .

Les identifiants :

Élément	Identifiant	Remarque
PERSONNE	idPersonne	Le couple ( <i>nom</i> , <i>prénom</i> ) n'est pas un bon identifiant en raison des homonymes possibles. Nous préférons un identifiant arbitraire dédié.
ABONNÉ	idPersonne	Puisque qu'un type-association de spécialisation lie <i>ABONNÉ</i> à <i>PERSONNE</i>
LIVRE	idLivre	Le titre n'est pas un bon identifiant. Nous préférons un identifiant arbitraire dédié.
EXEMPLAIRE	idExemplaire	
Ecrire	(idLivre, idPersonne)	Puisqu'il s'agit d'un type-association maillé liant <i>LIVRE</i> à <i>PERSONNE</i> .
Correspondre	idExemplaire	Puisqu'il s'agit d'un type-association fonctionnel dont le participant situé du côté de la cardinalité maximale de 1 est <i>EXEMPLAIRE</i>
Emprunter	(idExemplaire, idPersonne)	Puisqu'il s'agit d'un type-association maillé liant <i>EXEMPLAIRE</i> à <i>ABONNÉ</i> . Cet identifiant pose un problème...

L'identifiant (*idExemplaire*, *idPersonne*) du type-association *Emprunter* pose un problème, car un même abonné doit pouvoir emprunter un même exemplaire à des dates différentes. Il faut donc faire entrer le temps dans l'identifiant de ce type-association comme le propose la modélisation de la figure 2.14. Cette solution comporte néanmoins des défauts qui seront développés section 2.8.3.

## ★ Réponse à la question 9



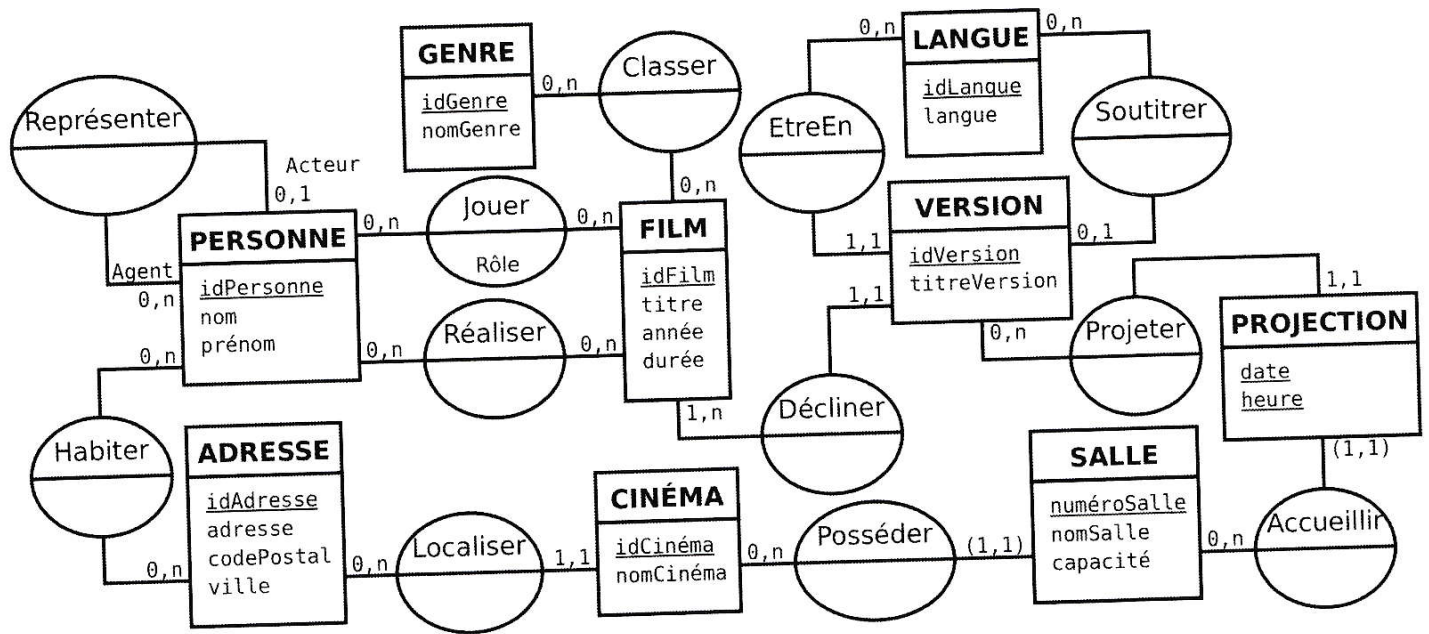
## 3 Correction du TD 2.7 - Modèle entités-associations (2<sup>e</sup> partie)

### ★ Réponse à la question 1

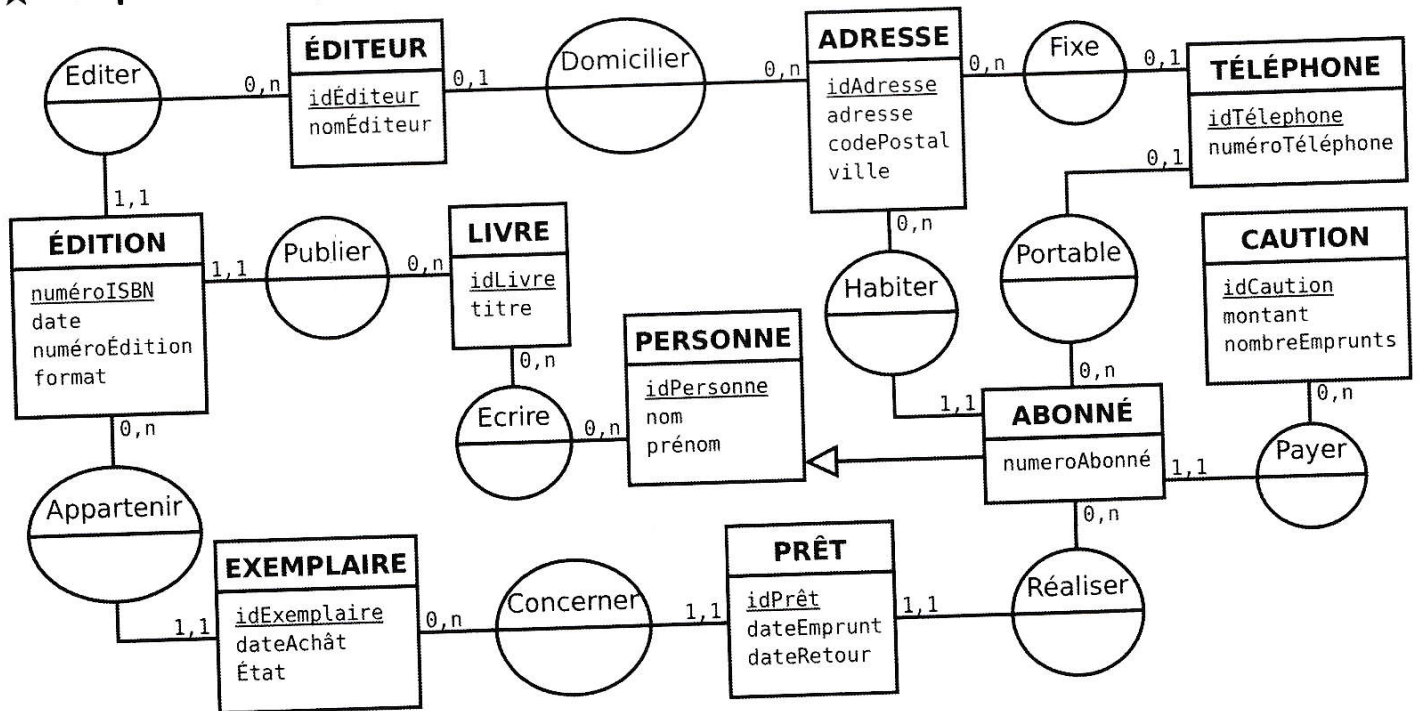
- Les clés du type (*nom, prénom*) sont à éviter, plus généralement choisir une clé artificielle dédiée (cf. section 2.5.1).
- Entorse à la règle 2.26 sur les noms : plusieurs attributs *nom* et *adresses*.
- Factoriser *ACTEUR* et *RÉALISATEUR* en *PERSONNE* (cf. règles 2.26 et 2.29).
- Entorse à la définition 2.34 de la première forme normale : les attributs *adresses* sont composés.
- *duréeDuFilm* devrait être dans *FILM* (cf. règle 2.13).
- *Date*, *Heure* et *Version* dans *Projeté* (cf. règles 2.13 et 2.14).
- Les cardinalités 2, *n* et 2, 20 sont interdites (cf. règle 2.19).
- Entorse à la règle 2.33 sur la redondance de type-association : le type-association *JouerDans-FilmDe* est à supprimer.
- Entorse à la règle 2.14 : l'attribut *capacité* du type-association *Posséder* ne dépend que du type-entité *SALLE*.
- Déplacer l'attribut *capacité* dans le type-entité *SALLE* crée une entorse à la 2FN (définition 2.35) : *capacité* ne dépend que de *nom*.
- Il y a une dépendance structurelle entre *CINÉMA* et *SALLE* qui devrait se traduire par un lien identifiant (cf. section 2.3.4).
- Le titre d'un film change souvent en fonction de sa version.
- Il faudra probablement créer les type-entités suivants :
  - *ADRESSE* ;
  - *GENRE* ;
  - *VERSION* ;
  - *LANGUE*.

### ★ Réponse à la question 2

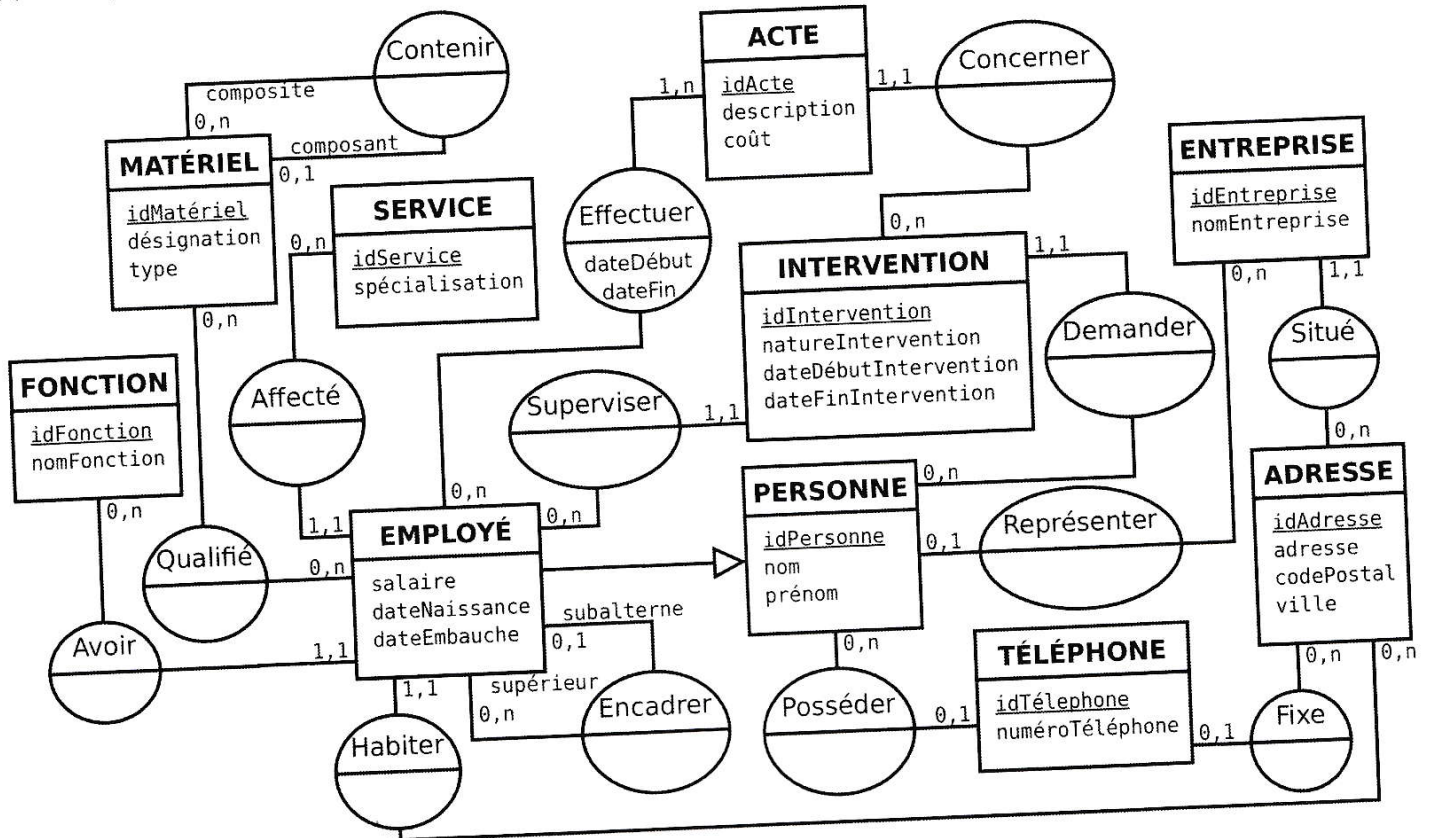
Modèle de la figure 2.53 corrigé (bien formé) :



### ★ Réponse à la question 3



## ★ Réponse à la question 4



## 4 Correction du TD 3.4 - Modèle relationnel

### ★ Réponse à la question 1

- ÉTUDIANT(idÉtudiant, nom, prénom)
- COURS(idCours, intitulé)
- INSCRIPTION(idÉtudiant, idCours, note) où idÉtudiant et idCours sont des clés étrangères qui font respectivement référence aux schémas de relation ÉTUDIANT et COURS

### ★ Réponse à la question 2

ÉTUDIANT		
idÉtudiant	nom	prénom
1	Roux	Sarah
2	Simon	Chloe
3	Durand	Lea

INSCRIPTION	
idÉtudiant	idCours
1	1
1	2
2	1
3	1
3	2

COURS	
idCours	intitulé
1	BD
2	UML
3	Java

### ★ Réponse à la question 3

La traduction en schéma relationnel doit contenir 2 schémas de relation.

### ★ Réponse à la question 4

- ENSEIGNANT(idEnseignant, nom, prénom)
- COURS(idCours, intitulé)
- DISPENSER(idEnseignant, idCours) où idEnseignant et idCours sont des clés étrangères qui font respectivement référence aux schémas de relation ENSEIGNANT et COURS

### ★ Réponse à la question 5

ENSEIGNANT		
idEnseignant	nom	prénom
1	Roux	Sarah
2	Simon	Chloe
3	Durand	Lea
4	Bernard	Lucas

DISPENSER	
idEnseignant	idCours
1	1
1	2
3	3

COURS	
idCours	intitulé
1	BD
2	UML
3	Java

### ★ Réponse à la question 6

Il y a bijection entre les lignes des relations DISPENSER et COURS. De plus, la véritable clé de DISPENSER est idCours et non pas le couple (idEnseignant, idCours). Les deux tables ont forcément le même nombre de lignes, la même clé et les mêmes valeurs dans la colonne idCours. Il faut donc supprimer DISPENSER et ajouter la colonne idEnseignant à la table COURS.

### ★ Réponse à la question 7

- ENSEIGNANT(idEnseignant, nom, prénom)
- COURS(idCours, intitulé, idEnseignant) où idEnseignant est une clé étrangère qui fait référence au schéma de relation ENSEIGNANT

### ★ Réponse à la question 8

- PERSONNE(idPersonne, nom, prénom)
- ADRESSE(idAdresse, adresse, code-postal, ville)
- FILM(idFilm, titre, année, durée)
- GENRE(idGenre, nomGenre)
- VERSION(idVersion, idFilm, idLangue, idLangueST, titreVersion) où idFilm, idLangue et idLangueST sont des clés étrangères qui font respectivement référence aux schémas de relation FILM, LANGUE et LANGUE
- LANGUE(idLangue, langue)
- CINÉMA(idCinéma, idAdresse, nomCinéma) où idAdresse est une clé étrangère qui fait référence au schéma de relation ADRESSE
- SALLE(numéroSalle, idCinéma, nomSalle, capacité) où idCinéma est une clé étrangère qui fait référence au schéma de relation CINÉMA
- JOUER(idActeur, idFilm, rôle) où idActeur et idFilm sont des clés étrangères qui font respectivement référence aux schémas de relation PERSONNE et FILM
- RÉALISER(idRéalisateur, idFilm) où idRéalisateur et idFilm sont des clés étrangères qui font respectivement référence aux schémas de relation PERSONNE et FILM
- REPRÉSENTER(idActeur, idAgent) où idActeur et idAgent sont des clés étrangères qui font respectivement référence aux schémas de relation PERSONNE
- HABITER(idPersonne, idAdresse) où idPersonne et idAdresse sont des clés étrangères qui font respectivement référence aux schémas de relation PERSONNE et ADRESSE

- CLASSIFICATION(idFilm, idGgenre) où idFilm et idGgenre sont des clés étrangères qui font respectivement référence aux schémas de relation FILM et GENRE
- PROJECTION(numéroSalle, idCinéma, date, heure, idVersion) où numéroSalle, idCinéma et idVersion sont des clés étrangères qui font respectivement référence aux schémas de relation SALLE, CINÉMA et VERSION

### ★ Réponse à la question 9

idPièce

### ★ Réponse à la question 10

Le schéma de relation PIÈCE est en 2FN puisque la clé primaire est réduite à un seul attribut, mais n'est pas en 3FN puisque l'attribut tva dépend de l'attribut catégorie qui n'est pas une clé candidate.

### ★ Réponse à la question 11

- PIÈCE(idPièce, prixUnit, libellé, catégorie) où catégorie est une clé étrangère qui fait référence au schéma de relation CATÉGORIE
- CATÉGORIE(catégorie, tva)

### ★ Réponse à la question 12

idMachine, idTechnicien

### ★ Réponse à la question 13

Le schéma de relation PRIME n'est pas en 2FN puisque certains attributs ne dépendent que d'une partie de la clé : atelier ne dépend que de idMachine et nomTechnicien ne dépend que de idTechnicien.

### ★ Réponse à la question 14

- MACHINE(idMachine, atelier)
- TECHNICIEN(idTechnicien, nomTechnicien)
- PRIME(idMachine, idTechnicien, montantPrime) où idMachine et idTechnicien sont des clés étrangères qui font respectivement référence aux schémas de relation MACHINE et TECHNICIEN

### ★ Réponse à la question 15

- (nomEnseignant, prénomEnseignant) → salaire
- nomCours → volumeHoraire

### ★ Réponse à la question 16

- ÉTUDIANT(idÉtudiant, nomÉtudiant, prénomÉtudiant)
- ENSEIGNANT(idEnseignant, nomEnseignant, prénomEnseignant, salaire)
- COURS(idCours, nomCours, volumeHoraire)
- ENSEIGNEMENT(idEnseignant, idÉtudiant, idCours) où idEnseignant, idÉtudiant et idCours sont des clés étrangères qui font respectivement référence aux schémas de relation ENSEIGNANT, ÉTUDIANT et COURS

### ★ Réponse à la question 17

ÉTUDIANT :

- $\text{idÉtudiant} \rightarrow \text{nomÉtudiant}$
- $\text{idÉtudiant} \rightarrow \text{prénomÉtudiant}$

ENSEIGNANT :

- $\text{idEnseignant} \rightarrow \text{nomEnseignant}$
- $\text{idEnseignant} \rightarrow \text{prénomEnseignant}$
- $\text{idEnseignant} \rightarrow \text{salaire}$

COURS :

- $\text{idCours} \rightarrow \text{nomCours}$
- $\text{idCours} \rightarrow \text{volumeHoraire}$

ENSEIGNEMENT :

- $(\text{idÉtudiant}, \text{idCours}) \rightarrow \text{idEnseignant}$
- $\text{idEnseignant} \rightarrow \text{idCours}$

### ★ Réponse à la question 18

La clé du schéma de relation ENSEIGNEMENT n'est plus correcte, car elle n'est plus minimale puisque  $(\text{idÉtudiant}, \text{idCours}) \rightarrow \text{idEnseignant}$ . Le couple  $\text{idÉtudiant}, \text{idCours}$  constitue une clé candidate qui permet au schéma de relation ENSEIGNEMENT de rester en 3FN. Le couple  $\text{idÉtudiant}, \text{idEnseignant}$  constitue également une clé candidate, mais ENSEIGNEMENT n'est même pas en 2FN par rapport à cette clé.

Le schéma de relation ENSEIGNEMENT devient donc :

ENSEIGNEMENT( $\text{idÉtudiant}, \text{idCours}$ ,  $\text{idEnseignant}$ )

### ★ Réponse à la question 19

Le schéma de relation ENSEIGNEMENT est en 3NF, mais pas en BCNF puisque l'attribut  $\text{idEnseignant}$  qui n'est pas une clé candidate détermine  $\text{idCours}$ .

### ★ Réponse à la question 20

Le fait de ne pas être en BCNF génère de la redondance. Dans notre cas, dans la relation ENSEIGNEMENT, le même cours doit apparaître systématiquement sur les toutes les lignes d'un enseignant donné.

### ★ Réponse à la question 21

- ÉTUDIANT( $\text{idÉtudiant}$ ,  $\text{nomÉtudiant}$ ,  $\text{prénomÉtudiant}$ )
- ENSEIGNANT( $\text{idEnseignant}$ ,  $\text{nomEnseignant}$ ,  $\text{prénomEnseignant}$ ,  $\text{salaire}$ ,  $\text{idCours}$ ) où  $\text{idCours}$  est une clé étrangère qui fait référence au schéma de relation COURS
- COURS( $\text{idCours}$ ,  $\text{nomCours}$ ,  $\text{volumeHoraire}$ )
- ENSEIGNEMENT( $\text{idEnseignant}, \text{idÉtudiant}$ ) où  $\text{idEnseignant}$  et  $\text{idÉtudiant}$  sont des clés étrangères qui font respectivement référence aux schémas de relation ENSEIGNANT et ÉTUDIANT

**Remarque :** Cette décomposition résout le problème de redondance, mais introduit un autre problème. La dépendance  $(\text{idÉtudiant}, \text{idCours}) \rightarrow \text{idEnseignant}$  est maintenant répartie sur deux schémas de relation. Par conséquent, les deux schémas de relation **ENSEIGNANT** et **ENSEIGNEMENT** ne peuvent plus être mis à jour indépendamment. Par exemple, un tuple  $(\text{Ens}, \text{Etu})$  ne peut être inséré dans **ENSEIGNEMENT** que si l'enseignant *Ens* n'enseigne pas un cours que l'étudiant *Etu* suit déjà avec un autre enseignant. La vérification de cette contrainte ne peut plus se faire au sein d'une simple relation, mais uniquement sur la jointure de deux relations.