Licence d'informatique

Bases de données 1

Conception de BD relationnelles

Florence Petit

4 février 2009



Conception de BD relationnelles

- Difficultés de conception d'une BD :
  - conception non intuitive
  - mais points communs dans les cas à traiter (souvent similaires)
- Objectifs :
  - éviter redondance de données
  - assurer cohérence des données
  - permettre des requêtes
- Utilisation d'une méthode de conception :
  - graphe des dépendances fonctionnelles (DF)
  - Schéma Entité/Association(E/A)
    ou Modèle Conceptuel de Données (MCD)

Exemple 1 : un exemple de problème de conception

- Comment stocker les données sur les étudiants ainsi que leur département de résidence sous forme de N° et de libellé ?
- Exemple de table mal construite, non normalisée

numE	nom	prenom	licence	dep	libDep
1	Martin	Véra	2	77	Seine-et-Marne
2	Martin	Annie	1	75	Paris
3	Dupont	Sylvie	2	77	Seine-et-Marne
4	Martin	Annie	1	93	Seine-Saint-Denis
5	Dupond	Laurent	2	92	Hauts-de-Seine
6	Lefèvre	Laurent	3		

clé primaire

– Redondance
– Volume de données (espace-disque) plus important
– Risque d'incohérence des données : anomalie à l'insertion/modification
– Risque de perte d'information (département) : anomalie à la suppression

Exemple 1 : Solution par répartition en 2 tables

table ETUDIANT

numE	nom	prenom	licence	numDep
1	Martin	Véra	2	77
2	Martin	Annie	1	75
3	Dupont	Sylvie	2	77
4	Martin	Annie	1	93
5	Dupond	Laurent	2	92
6	Lefèvre	Laurent	3	

table DEPARTEMENT

numDep	libDep	NbHab
01	Ain	…
…	...	…
75	Paris	…
…	…	…
77	Seine-et-Marne	…
92	Hauts-de-Seine	…
93	Seine-Saint-Denis	…

clé primaire

clé étrangère

clé primaire

Table dynamique
(avec clé étrangère)

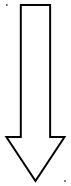
- gain de taille
- non redondance
- cohérence des données

Table statique
(sans clé étrangère)

Table de référence

Exemple 1 : Schéma relationnel

- ETUD (numE, nom, prenom, licence, numDep, ~~libDep, nbHab~~)



Comment obtenir un schéma relationnel correct (normalisé) ?

- ETUDIANT (numE, nom, prenom, licence, *numDep*)
- DEPARTEMENT (numDep, libDep, nbHab)

Légende
clé primaire : soulignée (convention usuelle)
clé étrangère : entourée (utilisée ici à des fins pédagogiques)

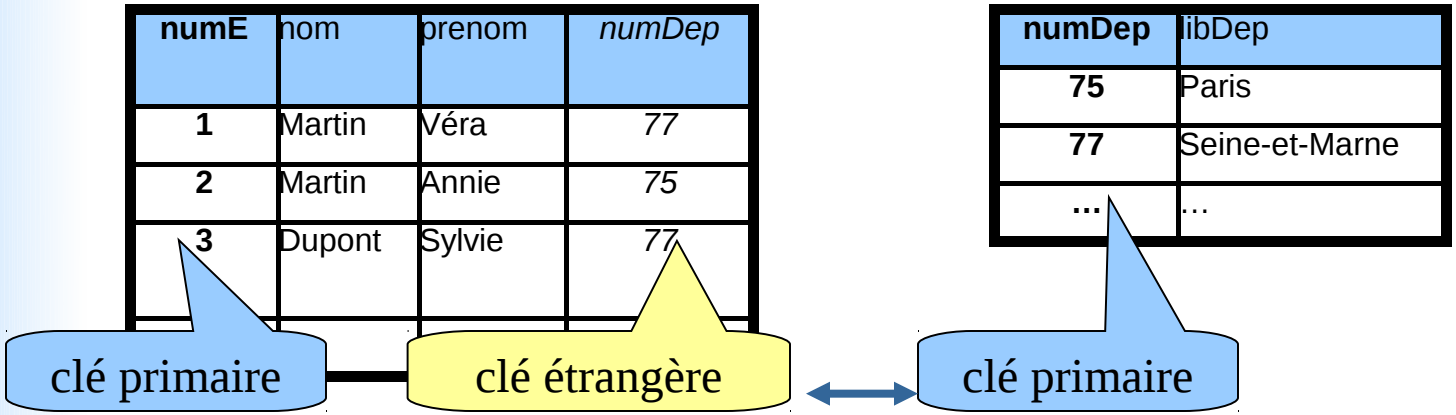
Clé primaire - Clé étrangère

Clé primaire (Primary Key = PK)

- attribut ou ensemble d'attributs d'une table, permettant d'identifier et de distinguer ses occurrences
- par convention, est soulignée dans le schéma relationnel

Clé étrangère (Foreign Key = FK)

- fait la liaison entre 2 tables d'une base de données
- clé étrangère de la table T1, clé primaire de la table T2.
- doit être d'un type de données compatible à la table de référence
- peut ou non être tout ou partie de clé primaire de sa table



Exemple 2 : Décomposition d'une table

numE	nomE	prenomE	libService	nomResp
00001	Dupont	René	Ressources humaines	Rouault
00002	Pairat	Luc	Recherche et développement	Davy
00003	Dupond	Jean	Ressources humaines	Rouault
…	…	…	Production	Proudhon

Transformation (normalisation) :

numE	nomE	prenomE	*codService*
00001	Dupont	René	*RH*
00002	Pairat	Luc	*R&D*
00003	Dupond	Jean	*RH*
…	…	…	*Prod*

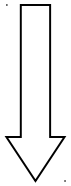
clé étrangère

codService	libService	nomResp
RH	Ressources humaines	Rouault
R&D	Recherche et développement	Davy
Prod	Production	Proudhon
…	…	…

clé primaire

Exemple 2 : Schéma relationnel

- SALARIE (numE, nomE, prenomE, libService, nomResp)

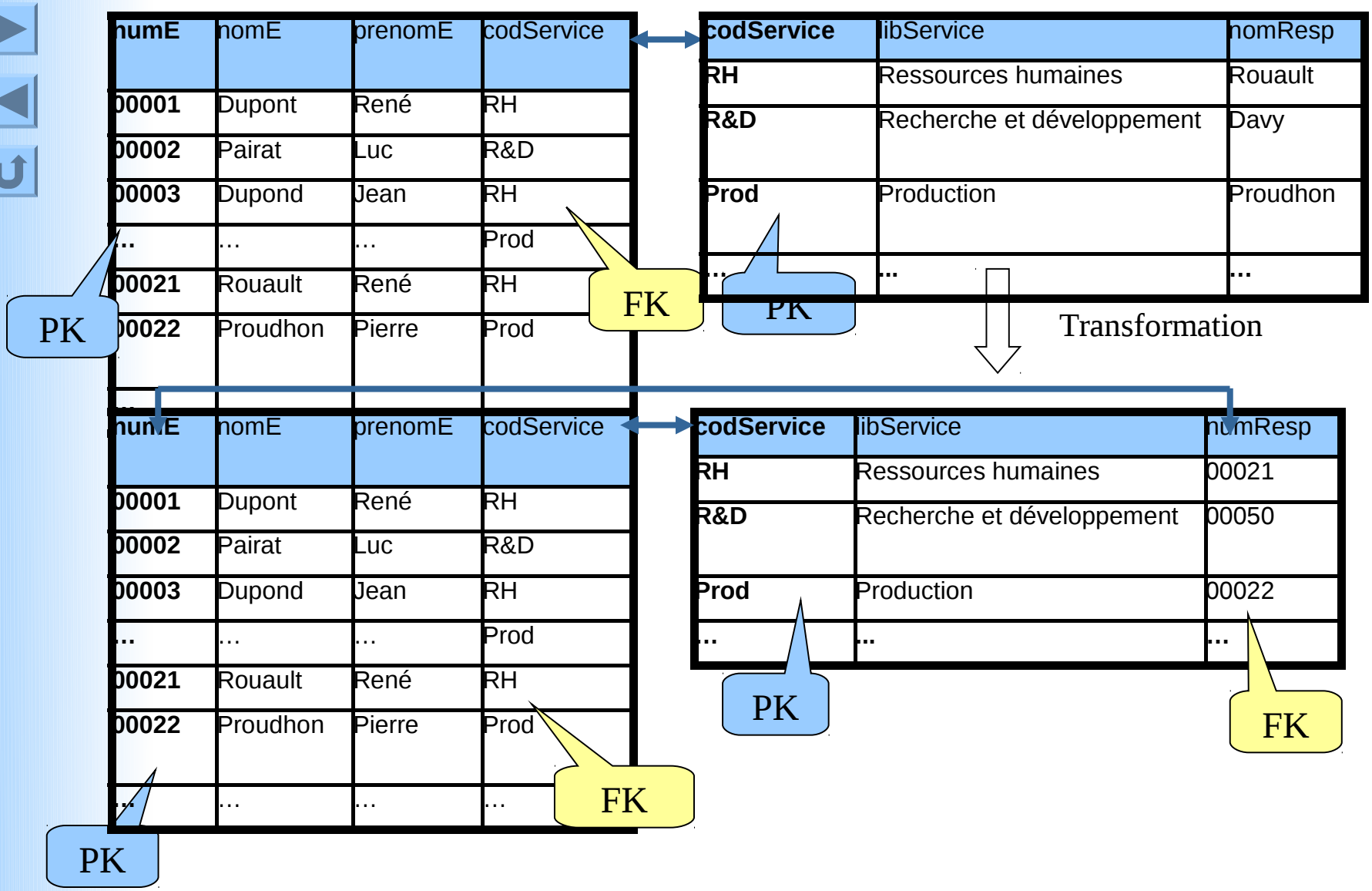


Transformation

de la table initiale (mal construite)

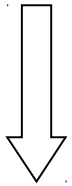
- SALARIE (**numE**, nomE, prenomE, *codService*)
- SERVICE (**codService**, libService, nomResp)

Exemple 2bis : Décomposition de tables



Exemple 2bis : Schéma relationnel

- SALARIE (**numE**, nomE, prenomE, *codService*)
- SERVICE (**codService**, libService, nomResp)



- SALARIE (**numE**, nomE, prenomE, *codService*)
- SERVICE (**codService**, libService, *numResp*)

Exemple 3 : trouver le schéma relationnel…

- Cas à modéliser :
  Un client (numéro de client, nom, adresse) passe des commandes (numéro de commande, date de commande) de produits (code, libellé). Chaque produit a un prix et peut être commandé en une certaine quantité.

- COMMANDE (num-cli, nom, adresse, num-com, date-com, code-prod, lib-prod, prix, quantite)

  ⇩ Comment obtenir un schéma relationnel correct (normalisé) ?

- CLIENT (num-cli, nom, adresse)
  COMMANDE (num-com, date-com, num-cli)
  PRODUIT (code-prod, lib-prod, prix)
  LIGNE-DE-COMMANDE (num-com, code-prod, quantité)

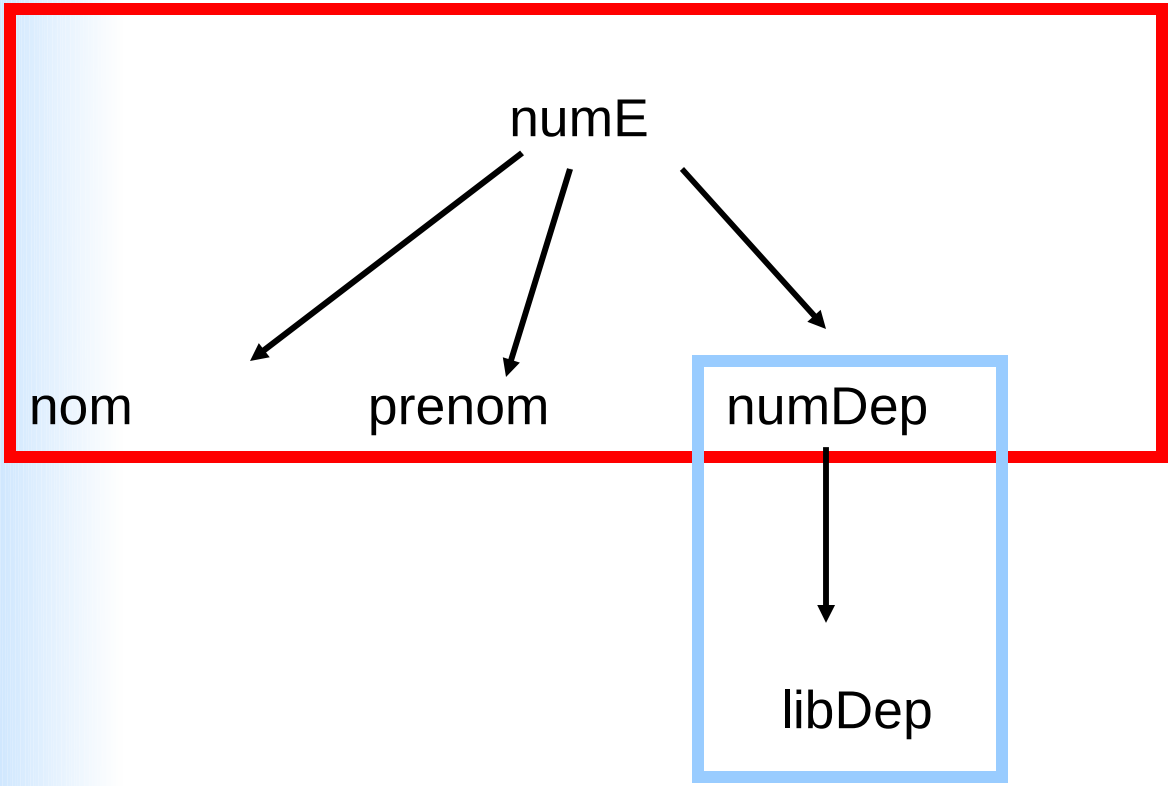
  quantite ?

Dépendances Fonctionnelles
Graphe des DF

Une méthode de conception

pour les bases de données relationnelles

Exemple 1 : Graphe des dépendances fonctionnelles



- ETUDIANT (<u>numE</u>, nom, prenom, numDep)
- DEPARTEMENT (<u>numDep</u>, libDep)

Dépendance fonctionnelle (DF)

- Soit 2 attributs A et B, il existe une DF de A vers B si et seulement si à chaque valeur de A correspond une valeur unique de B.
  A et B peuvent être des attributs composés (ensemble d'attributs)
- On peut exprimer une DF en disant :
  - Il existe une DF de A vers B
  - A détermine B
  - B dépend fonctionnellement de A

Notation d'une DF : A → B

A : source

B : destination, cible

Exemple de DF :

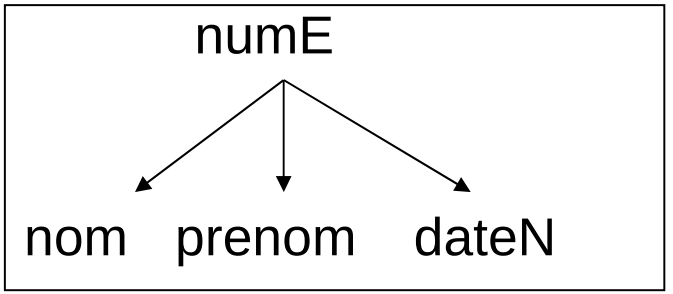
numE → nom

numE → prenom

numE → dateN

Graphe des DF

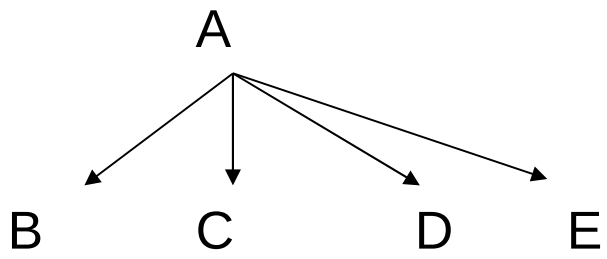
- nœud : attribut
- arc : DF



DF et clé primaire

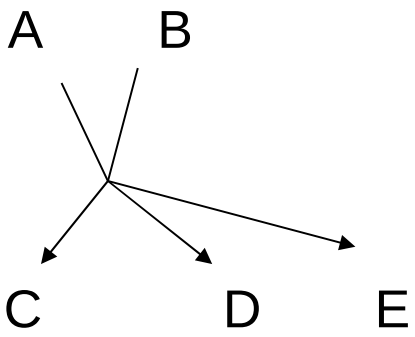
- TABLE (A, B, C, D, E)
- Si un attribut détermine tous les autres attributs de la table, il est clé primaire de la table.

DF
A → B
A → C
A → D
A → E



- TABLE (<u>A</u>, B, C, D, E)

DF
A,B → C
A,B → D
A,B → E



- TABLE (<u>A, B</u>, C, D, E)

Propriétés des DF

Non commutatif

$A \rightarrow B \not\Rightarrow B \rightarrow A$

n°E → nomE $\not\Rightarrow$ nomE → numE

Transitivité

$A \rightarrow B,\ B \rightarrow C \Rightarrow A \rightarrow C$

n°E → UFR, UFR → n°Resp ⇒ n°E → n°Resp

Union

$A \rightarrow B,\ A \rightarrow C \Leftrightarrow A \rightarrow B, C$

n°E → nomE, n°E → dateN ⇔ n°E → nomE, dateN

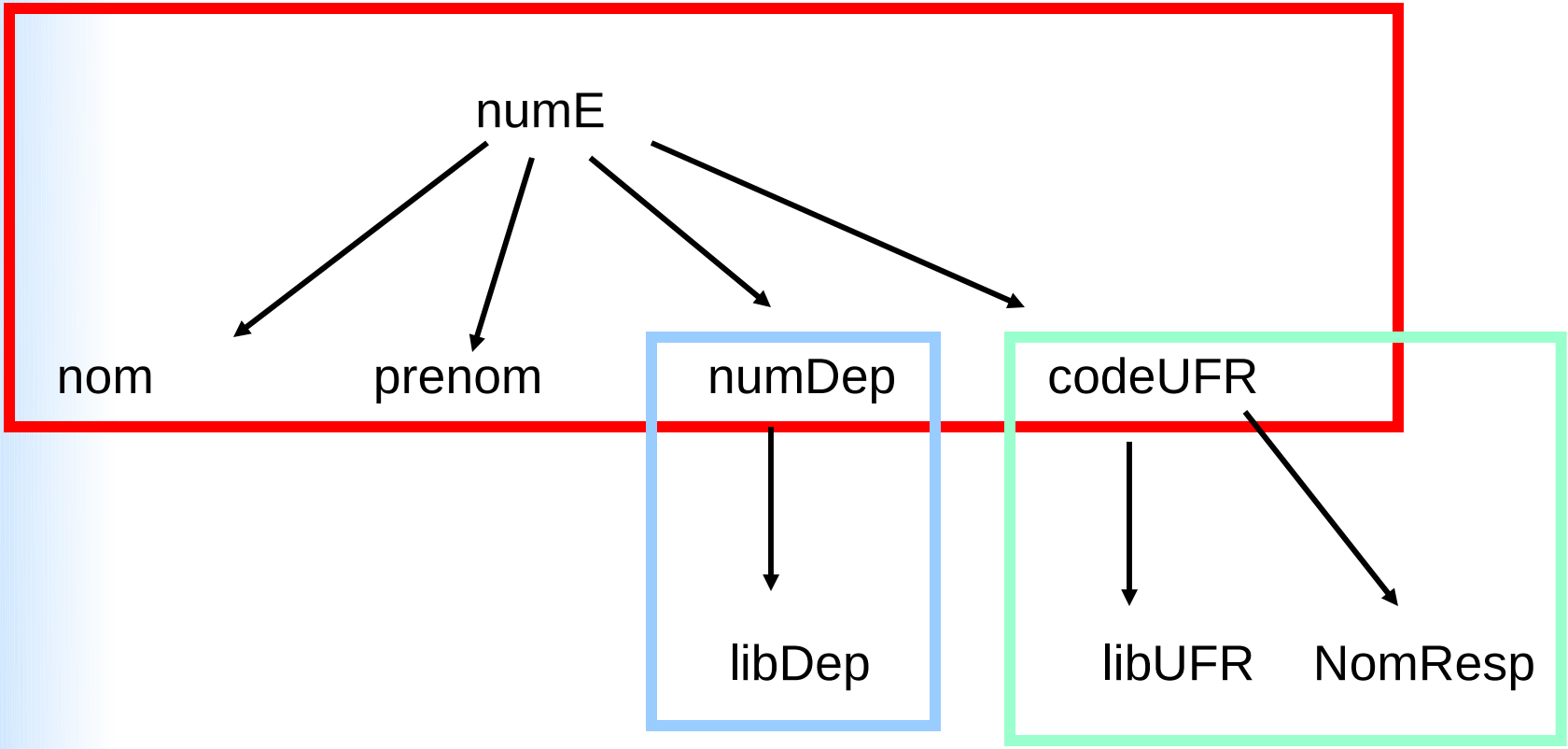
$A, B \rightarrow C \not\Leftrightarrow A \rightarrow C,\ B \rightarrow C$

n°E, matiere → note $\not\Leftrightarrow$ n°E → note, matiere → note

Exemple 1 : Dépendances fonctionnelles

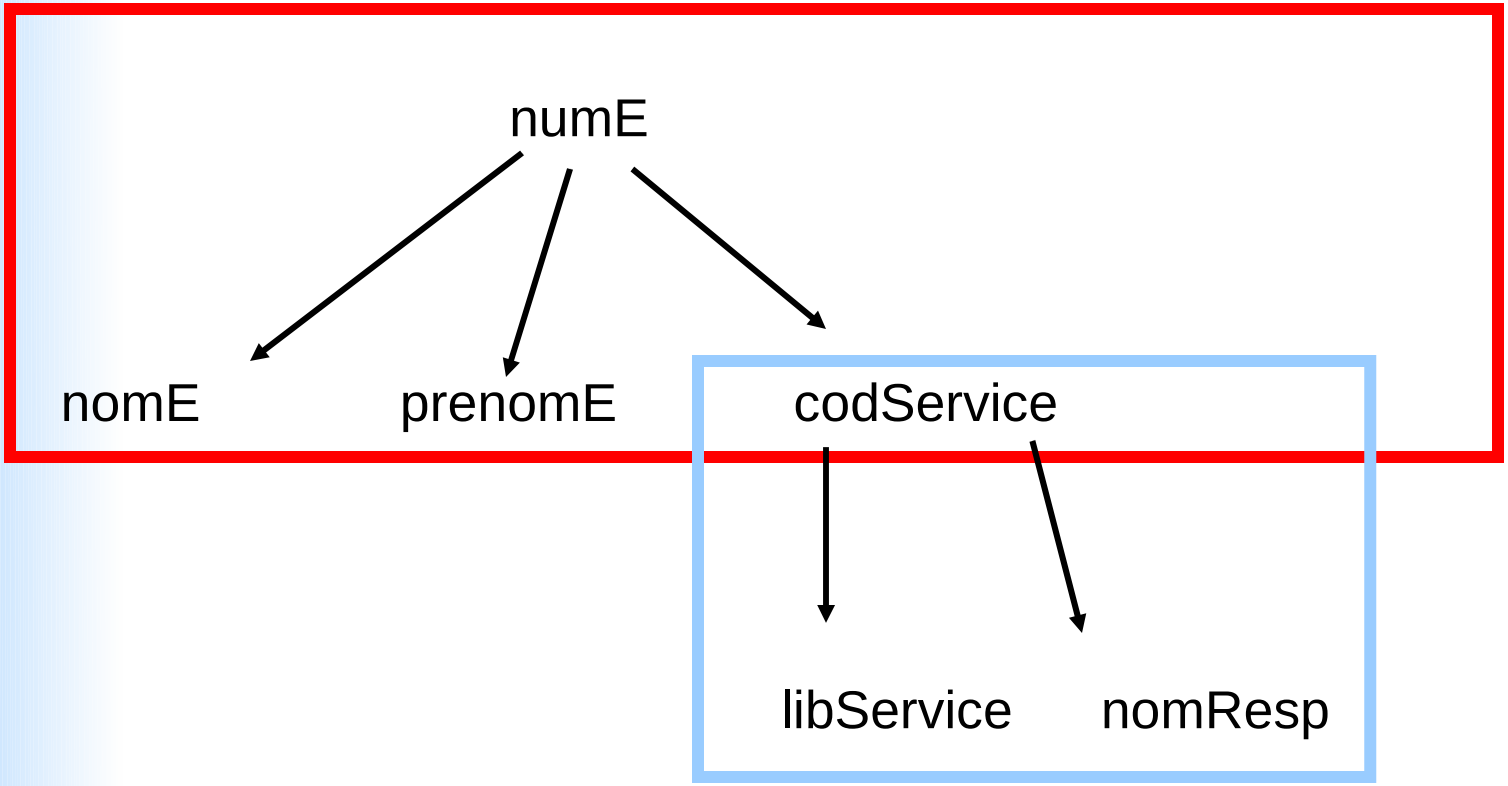
- Soit : numE, nom, prenom, numDep, libDep
- numE → nom
  numE → prenom
  numE → numDep
  numE → libDep
  numdep → libDep
  libdep → numDep

Exemple 1bis : Graphe des dépendances fonctionnelles



- ETUDIANT (numE, nom, prenom, numDep, codeUFR)
- DEPARTEMENT (numDep, libDep)
- UFR (codeUFR, libUFR, nomResp)

Exemple 2 : Graphe des dépendances fonctionnelles



- SALARIE (numE, nomE, prenomE, codService)
- SERVICE (codService, libService, nomResp)

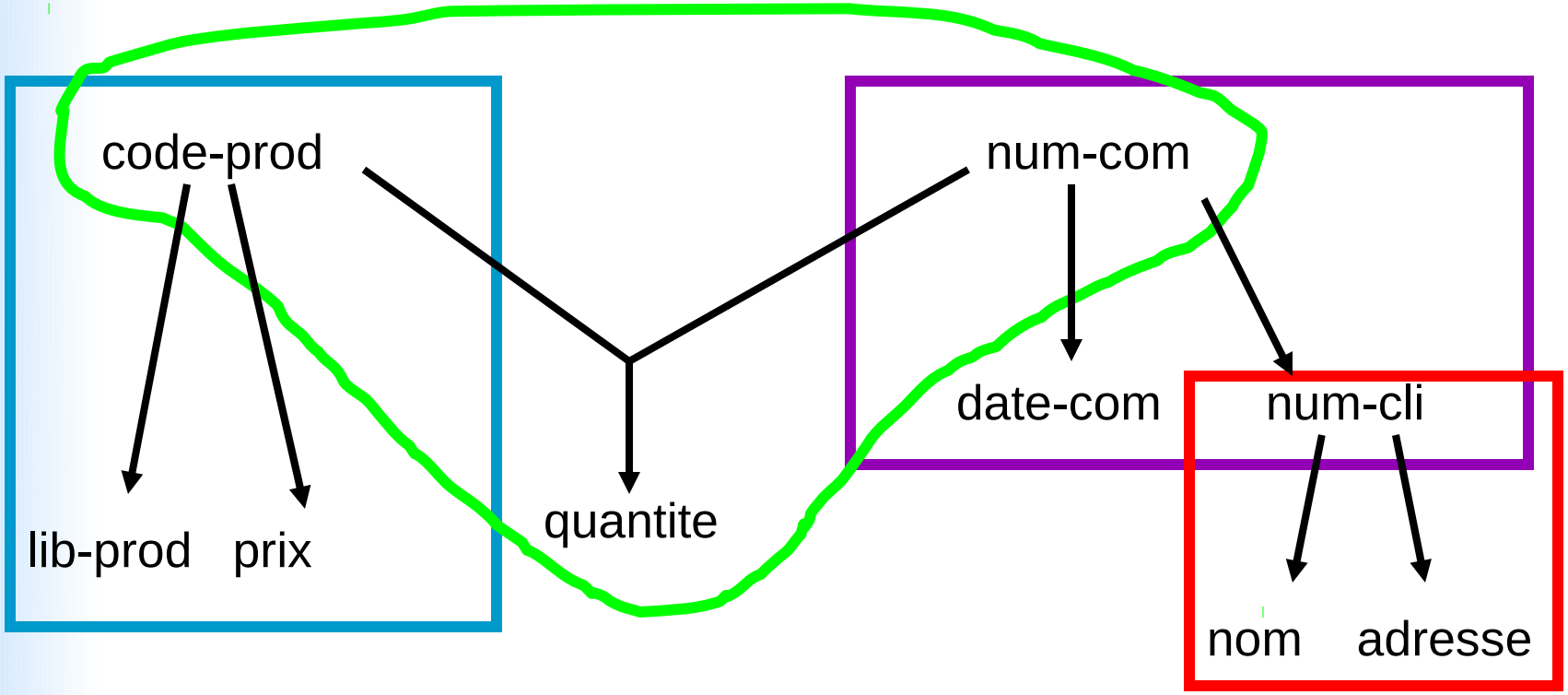
Exemple 3 : Dépendances fonctionnelles

Soit les attributs :

num-cli, nom, adresse, num-com, date-com, code-prod, prix, lib-prod, quantite

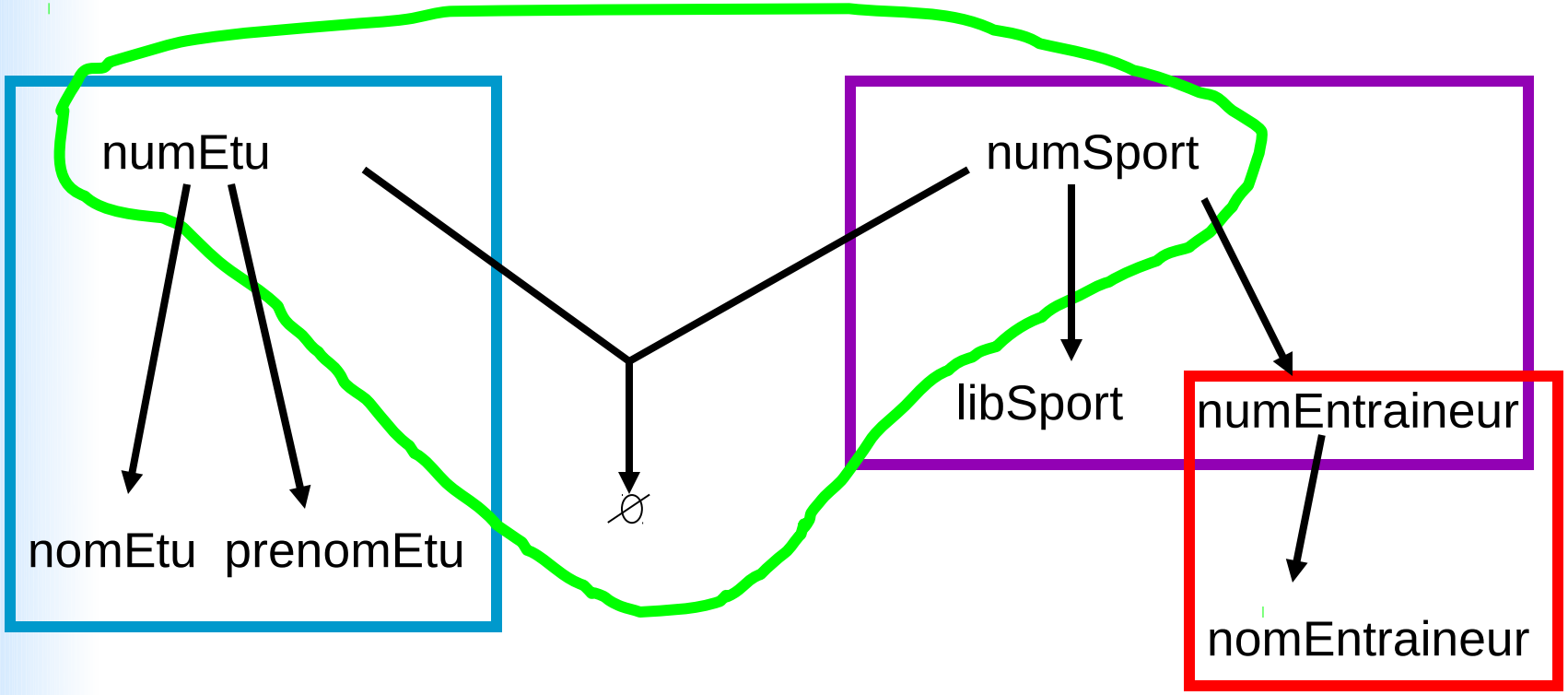
- num-cli --> nom, adresse
- num-com --> date-com, num-cli
- code-prod --> lib-prod, prix
- code-prod, num-com --> quantite

Exemple 3 : Graphe des dépendances fonctionnelles



- PRODUIT (<u>code-prod</u>, lib-prod, prix)
- CLIENT (<u>num-cli</u>, nom, adresse)
- COMMANDE (<u>num-com</u>, date-com, num-cli)
- LIGNE-DE-COMMANDE(<u>code-prod</u>, <u>num-com</u>, quantite)

Exemple 4 : Graphe des DF



- ETUDIANT (numEtu, nomEtu, prenomEtu)
- ENTRAINEUR (numEntraineur, nomEntraineur)
- SPORT (numSport, libSport, numEntraineur)
- PRATIQUER(Numetu, numSport)

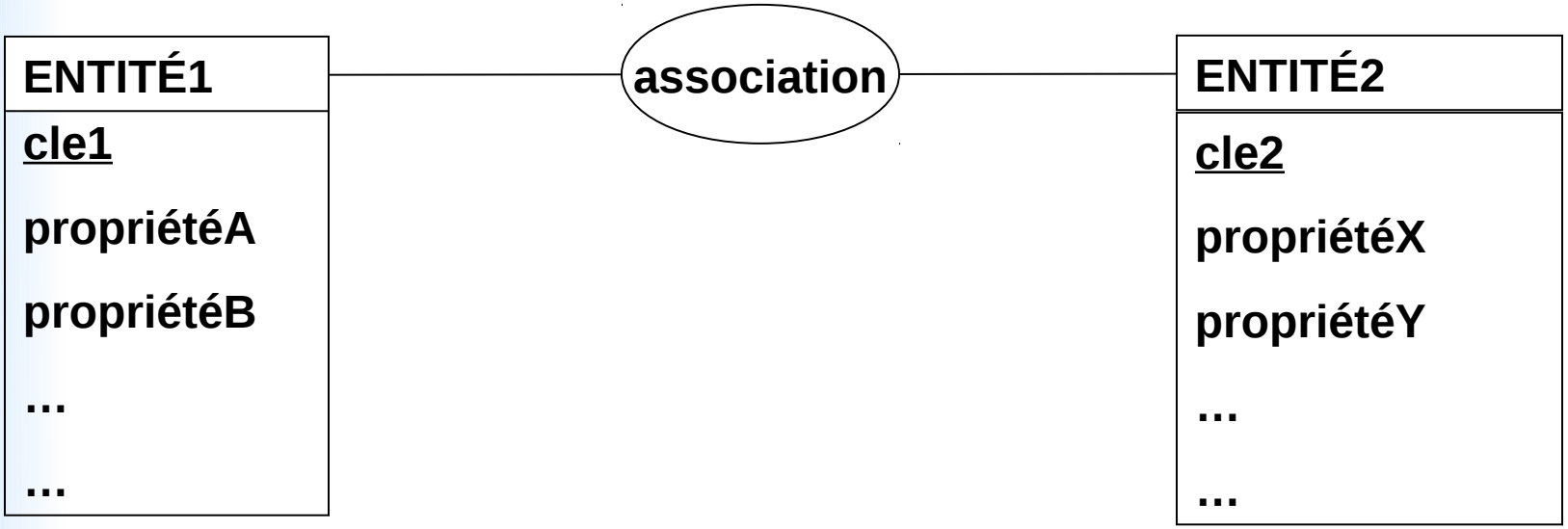
Cas particulier : pas d'attribut cible

Entité / Association

Méthode de conception
de bases de données

Schéma Entité/Association (E/A)

- ou Entite/Relation (E/R) ou modèle conceptuel de données (MCD)
- Inventé par Chen en 1976 (USA), utilisé dans la méthode Merise
- Les entités (ensemble d'objets) possèdent des propriétés
- Les entités sont reliées par des associations

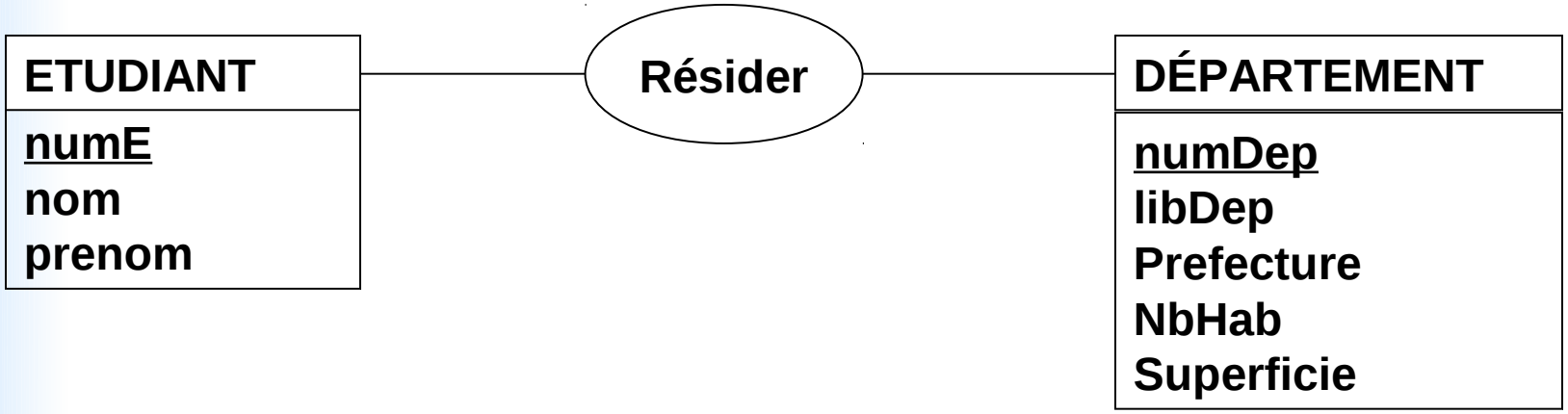


Exemples d'entités

- Objets munis de propriétés
- Objets concrets :
  - employés
  - étudiants
  - visiteurs
  - animaux
  - produits
  - meubles
  - immeubles
  - ordinateurs
  - livres
- Objets abstraits :
  - morceaux de musique
  - concerts
  - commandes
  - consultations médicales
  - UFR
  - entreprises
  - évènements

Schéma E/A : exemple 1

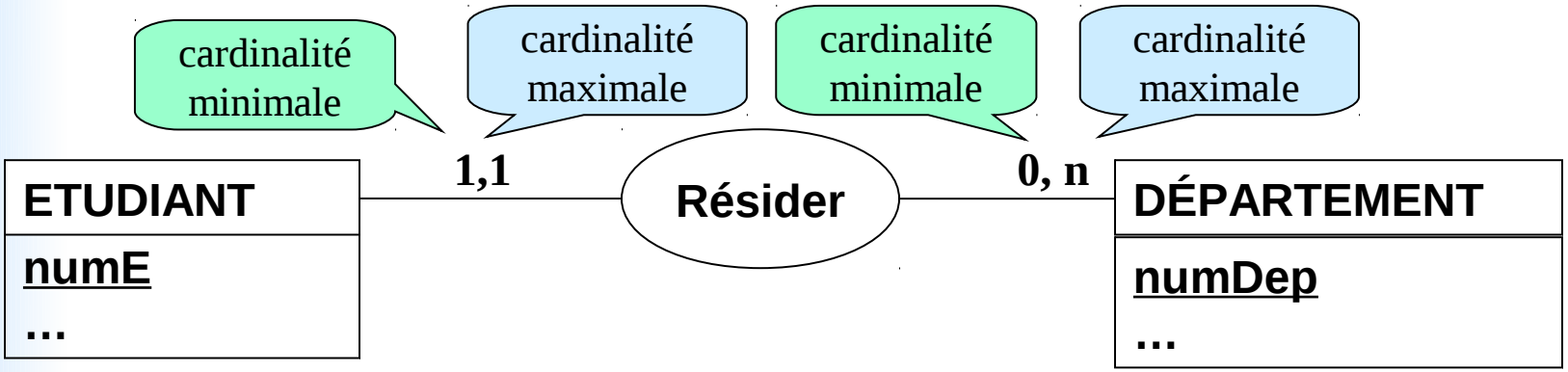
- Un étudiant réside (administrativement) dans un département



Attention : pas de clé étrangère sur un schéma E/A

Schéma Entité/Association : cardinalités

- Des cardinalités précisent la participation de l'entité à l'association
- La cardinalité minimale peut être de 0 ou de 1
- La cardinalité maximale peut être de 1 ou de n



Un étudiant réside dans un et un seul département

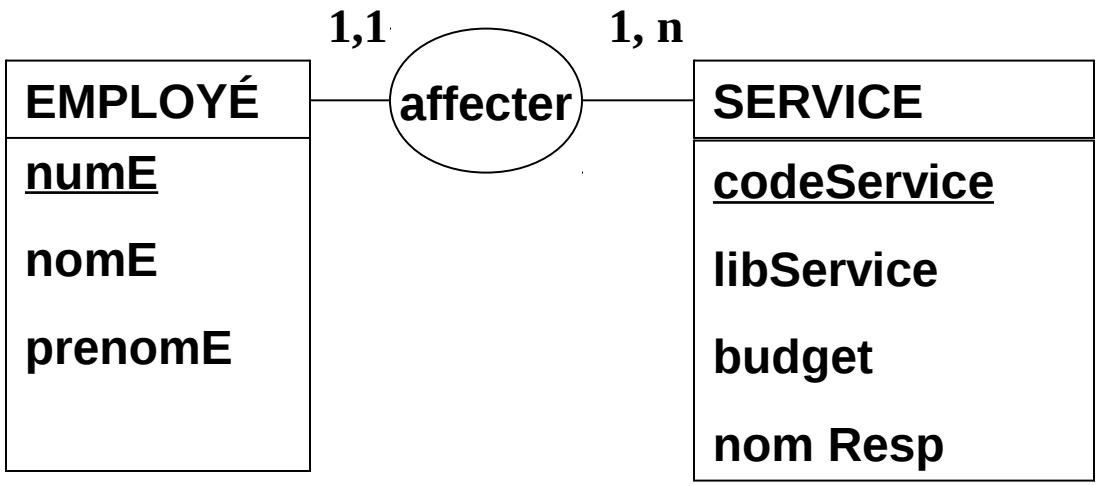
-> cardinalités 1,1

Un département a pour résident aucun ou plusieurs étudiants

-> cardinalités 0, n

Schéma E/A : exemple 2

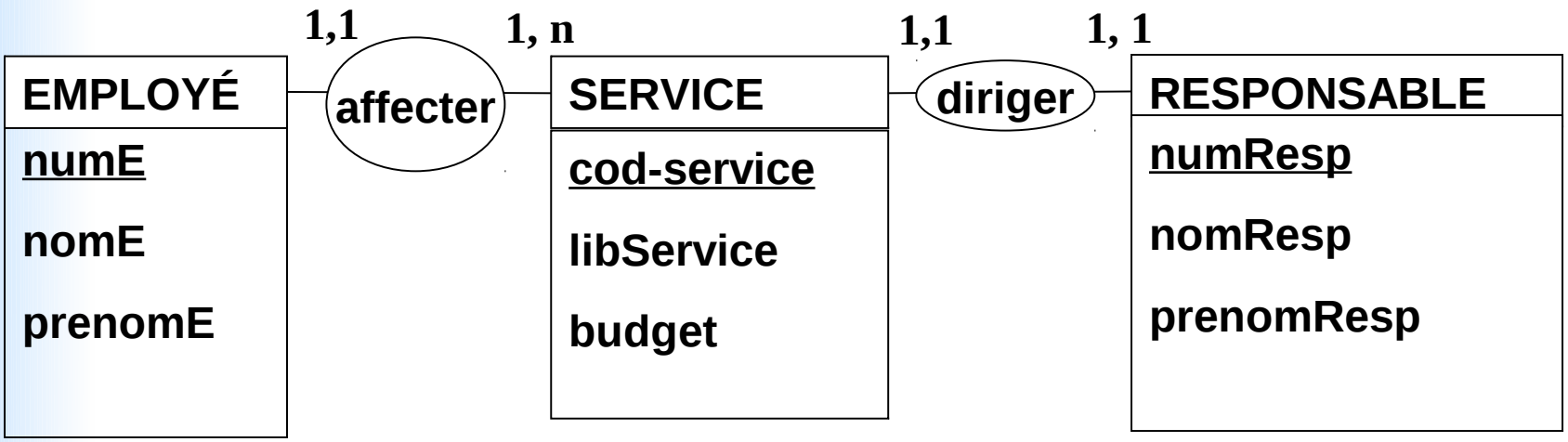
- Un employé est affecté dans un et un seul service.
- Le service a 1 ou plusieurs employés



Attention : pas de clé étrangère sur un schéma E/A

Schéma E/A : exemple 2bis

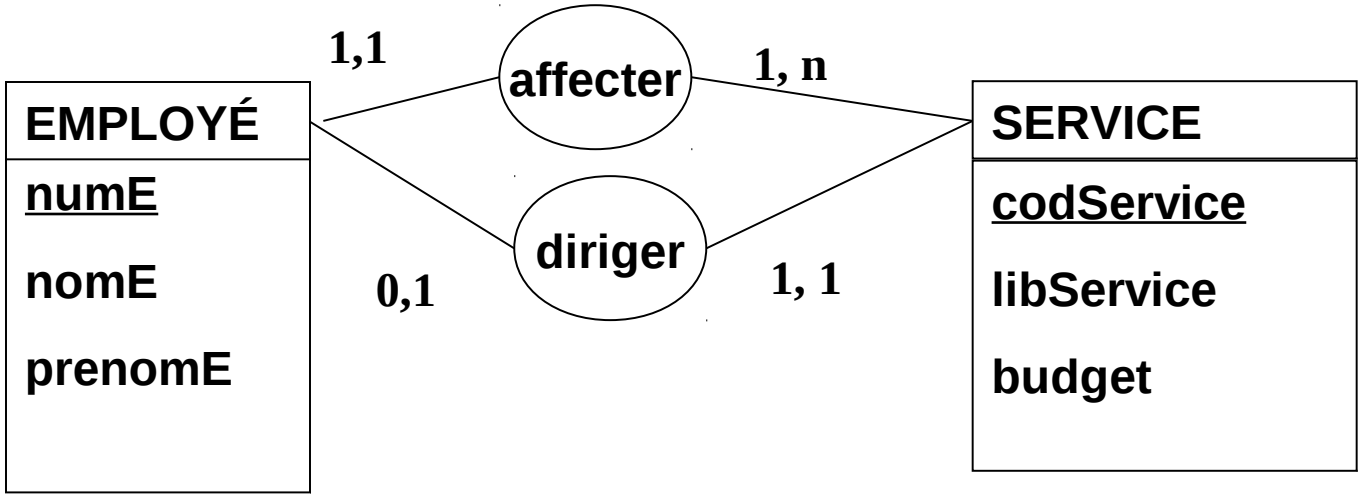
- Un employé est affecté dans un et un seul service.
- Le service comporte 1 ou plusieurs employés
- Le service est dirigé par un et un seul responsable.
- Le responsable dirige un et un seul service



Attention : pas de clé étrangère sur un schéma E/A

Schéma E/A : exemple 2bis (variante)

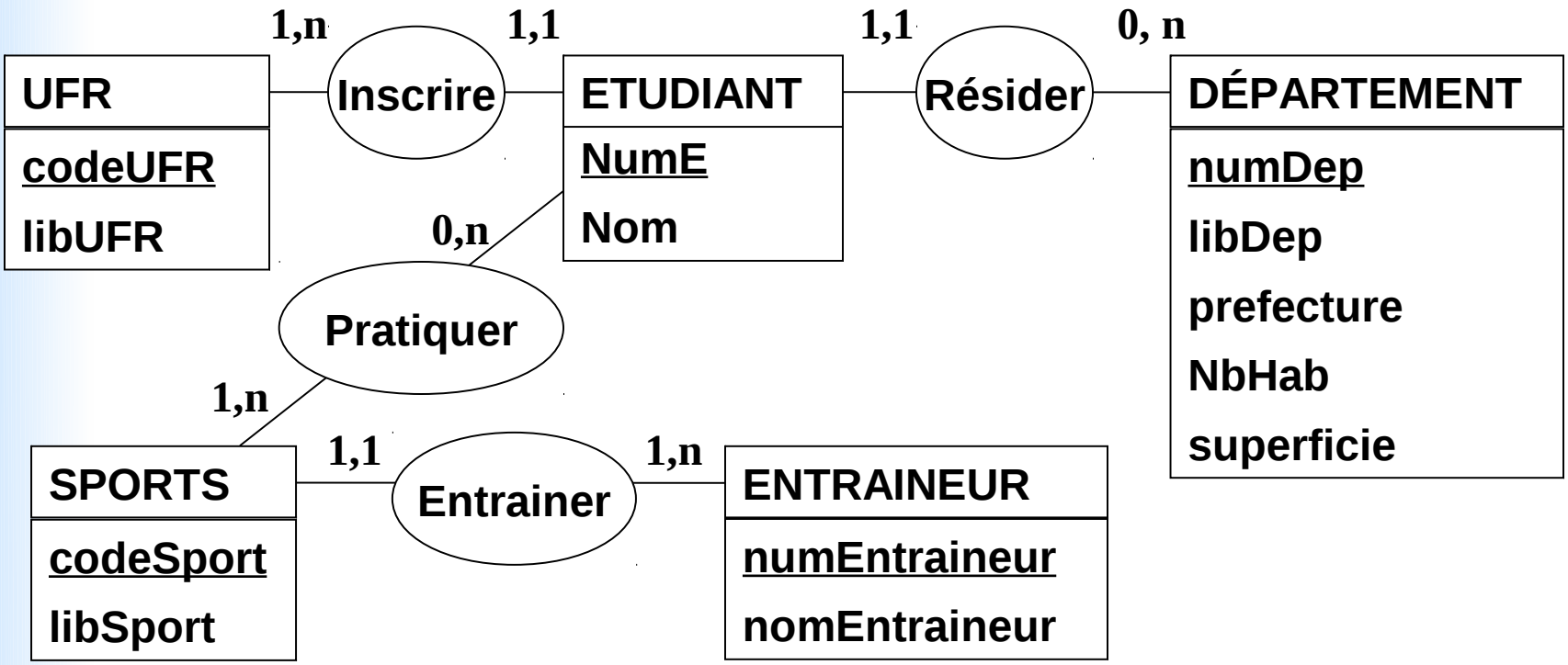
- Un employé est affecté dans un et un seul service.
- Le service comporte 1 ou plusieurs employés
- Le service est dirigé par un et un seul responsable (qui est aussi un employé).
- Le responsable dirige un et un seul service



2 entités peuvent être reliées par des associations ayant des significations différentes

Schéma E/A : exemple (1bis + 4)

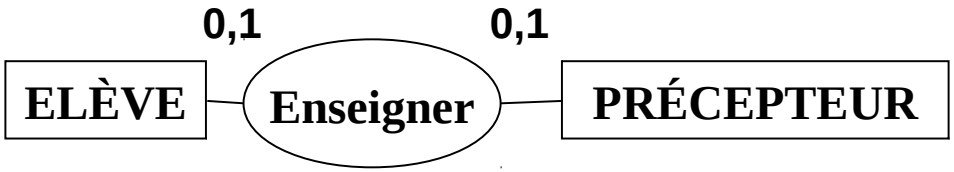
- Un étudiant réside (administrativement) dans un département
- Un étudiant est inscrit dans une UFR
- Un éudiant peut pratiquer un ou plusieurs sports



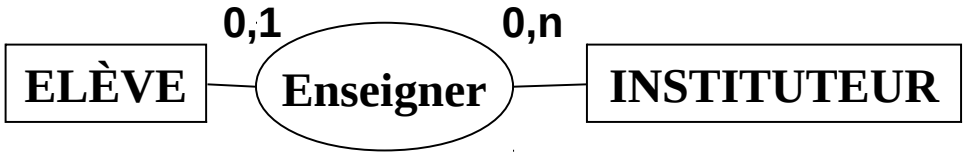
Attention : pas de clé étrangère sur un schéma E/A

Types d'associations (selon cardinalités maximales)

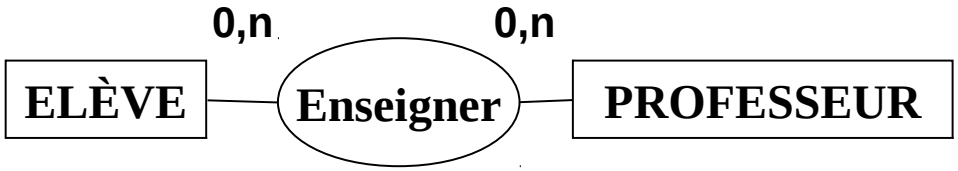
- Association 1 - 1 (un à un)



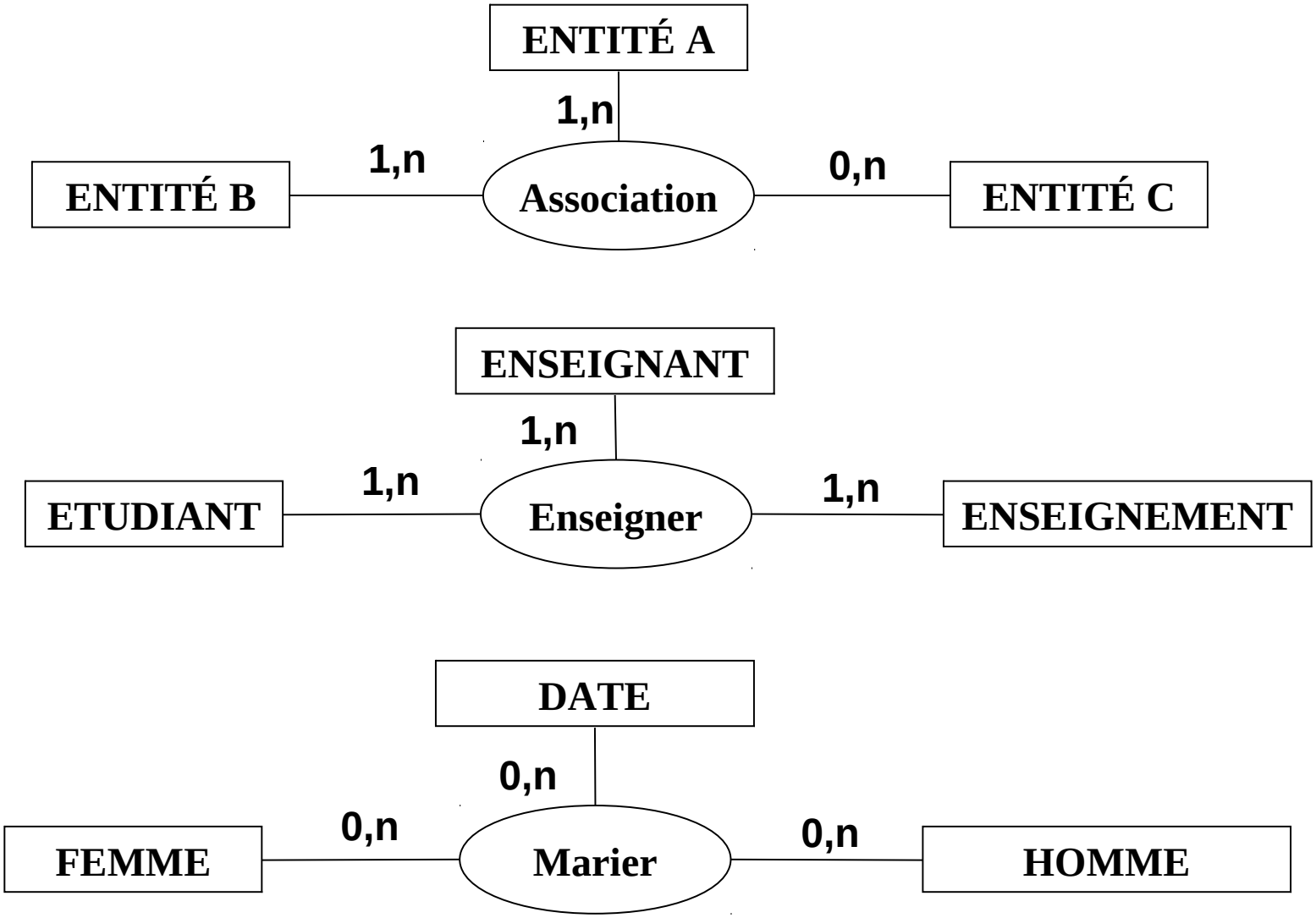
- Association 1 - N (un à plusieurs)



- Association N - M (plusieurs à plusieurs)



Associations ternaires (à "3 pattes")

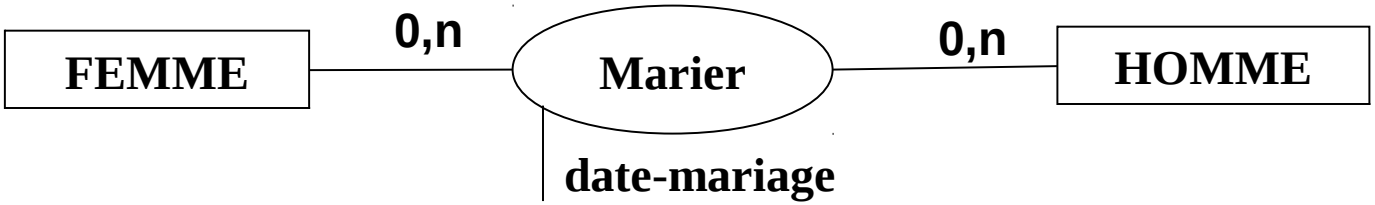


Associations ternaires et historique

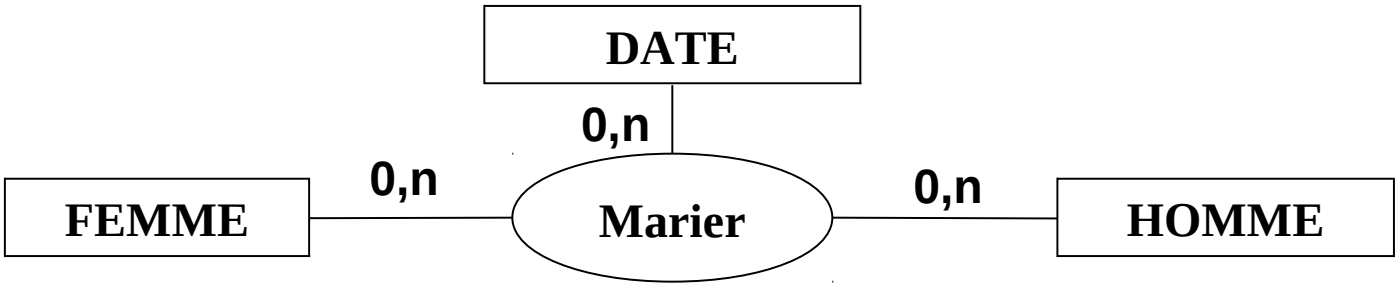
Pas d'historique



Historique : un couple ne se marie qu'une fois (date unique)



Historique : un couple peut se remarier n fois



Règles de passage E/A->relationnel

- Règle 1 : entité
  - Chaque entité donne une table
  - Sa clé primaire est la clé de la table
- Règle 2 : association de type 1-N ou 1-1
  - La clé primaire de l'entité côté N est ajoutée du côté 1 où elle devient clé étrangère
- Règle 3: association de type N-M
  - Création d'une nouvelle table dont la clé est l'ensemble des clés primaires des entités concernées
  - Tout attribut de l'association devient attribut de la nouvelle table

Représentation d'une relation de type 1-N

Graphe des DF

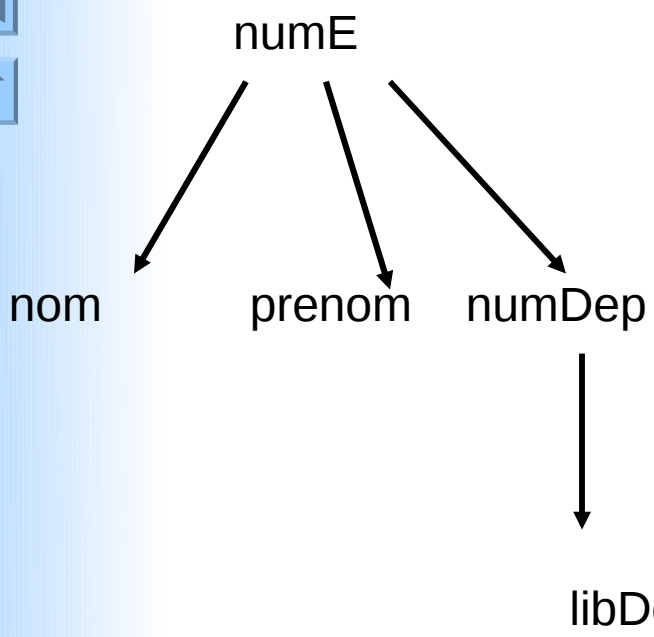


Schéma Entité/Association

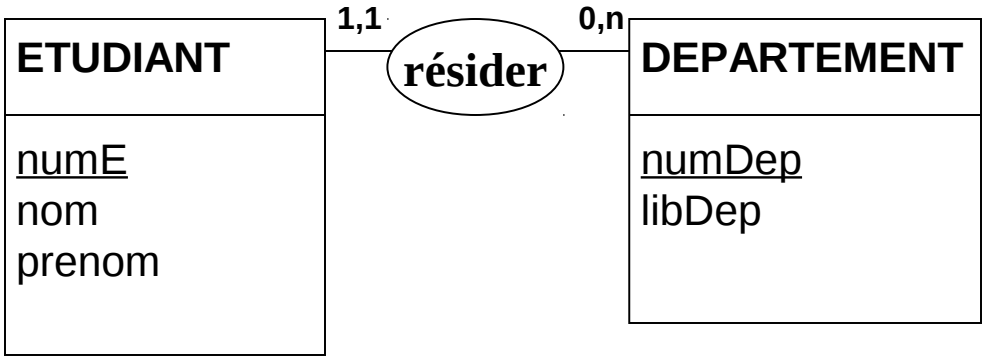


Schéma relationnel en intension

- ETUDIANT (numE, nom, prenom, numDep)
- DEPARTEMENT (numDep, libDep)

Représentation d'une relation de type N-M

Graphe des DF

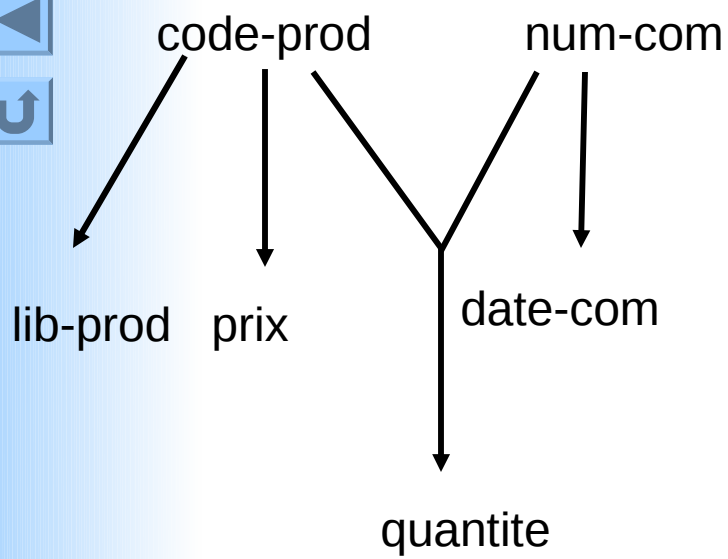


Schéma Entité/Association

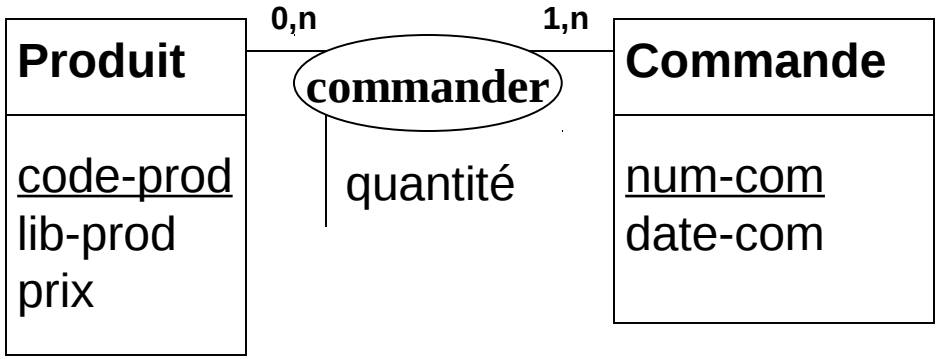


Schéma relationnel en intension

- PRODUIT (code-prod, lib-prod, prix)
- COMMANDE (num-com, date-com)
- LIGNE-DE-COMMANDE(code-prod, num-com, quantite)

Représentation d'une relation de type N-M

Graphe des DF

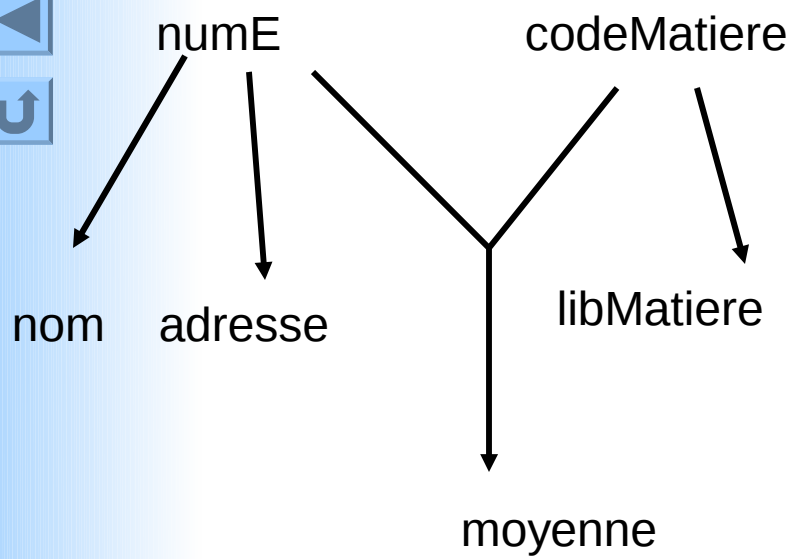


Schéma Entité/Association

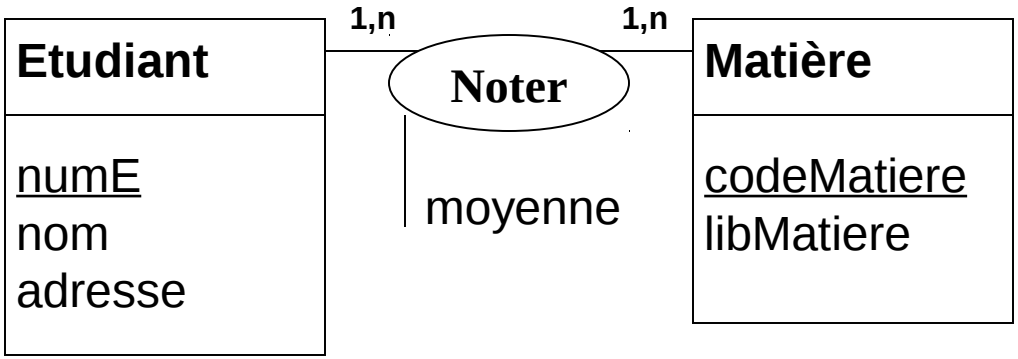


Schéma relationnel en intension

- ETUDIANT (numE, nom, adresse)
- MATIERE (codeMatiere, libMatiere)
- NOTER(numE, codeMatiere, moyenne)

Représentation d'une relation de type N-M (vide)

Graphe des DF

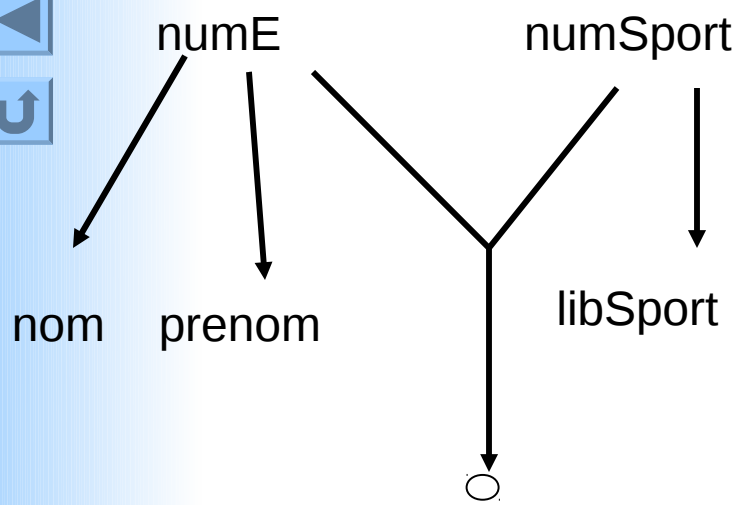


Schéma Entité/Association

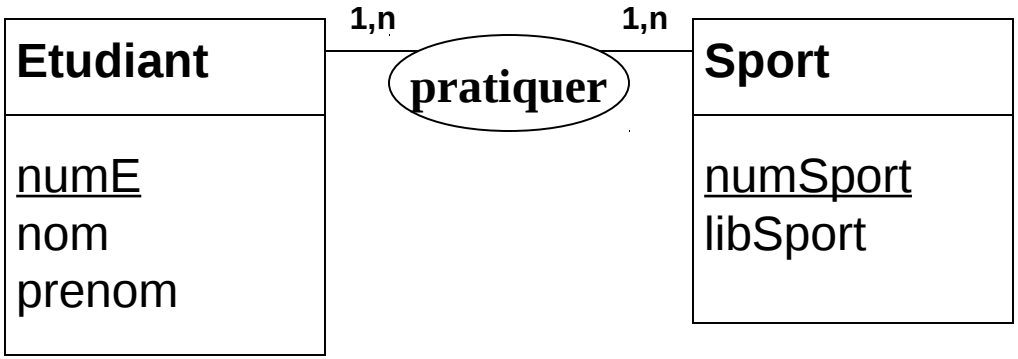


Schéma relationnel en intension

- ETUDIANT (numE, nom, prenom)
- SPORT (numSport, libSport)
- PRATIQUER (numE, numSport)

Représentation d'une relation de type 1-1

Graphe des DF

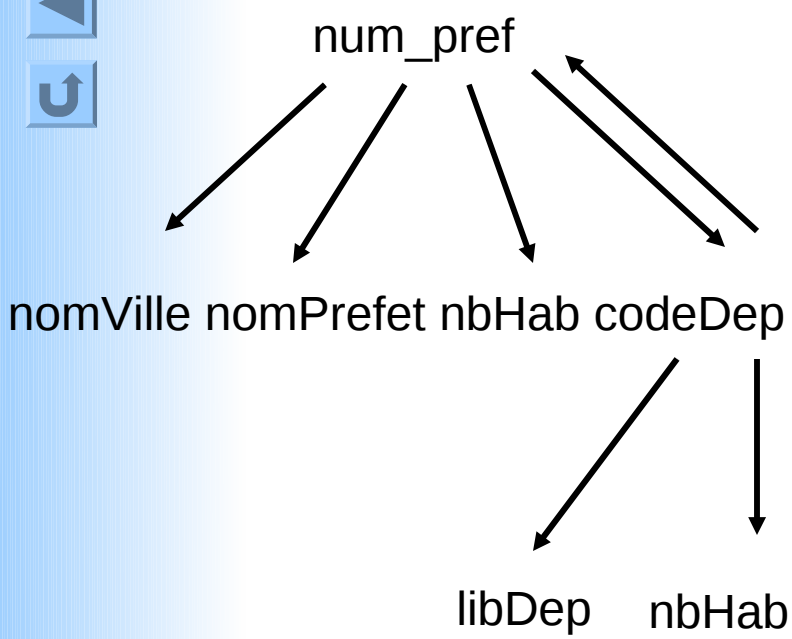


Schéma Entité/Association

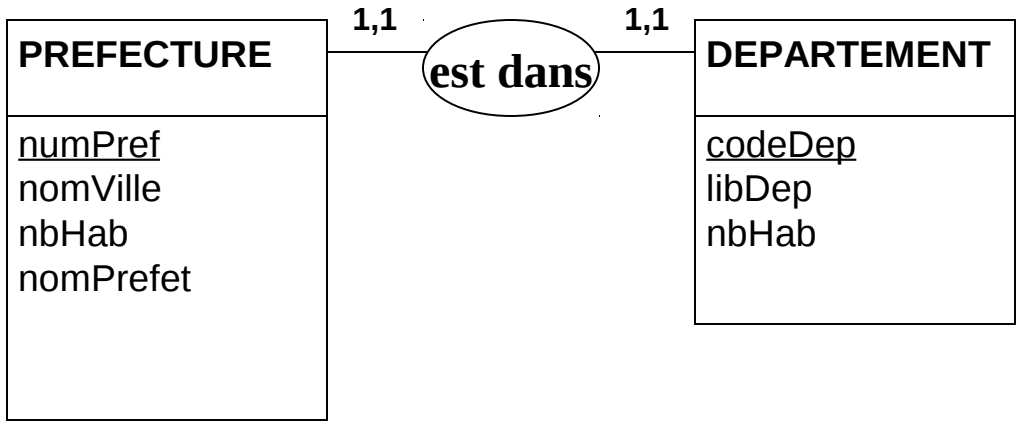


Schéma relationnel en intension

- PREFECTURE (numPref, nomVille, nbHab, nomPrefet, codeDep)
- DEPARTEMENT (codeDep, libDep)

Représentation d'une relation de type 1-1

Graphe des DF

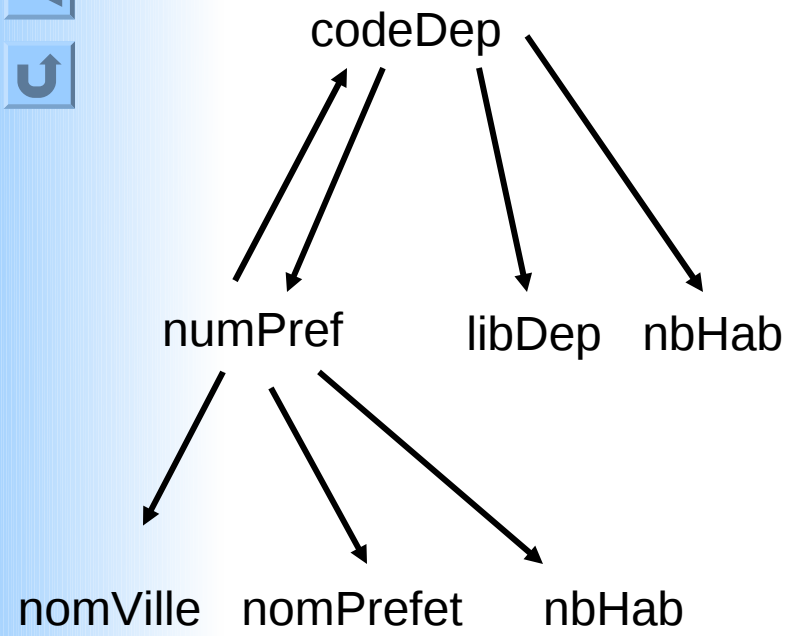


Schéma Entité/Association

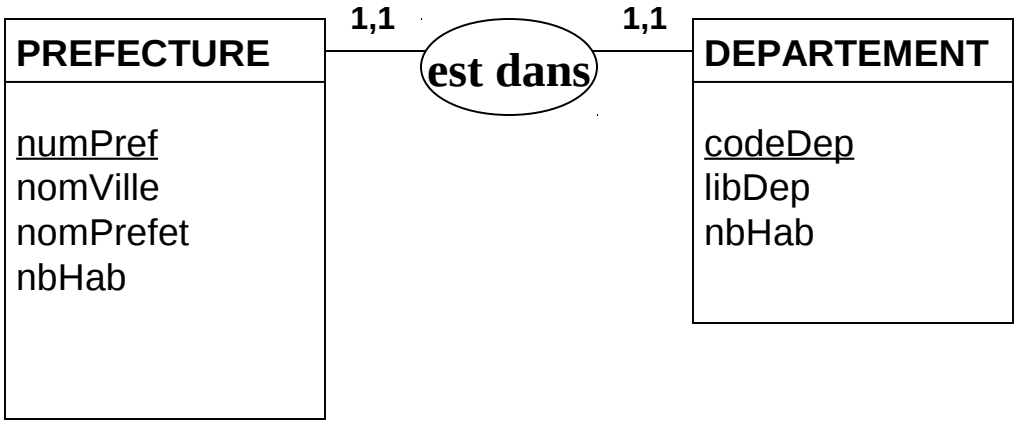


Schéma relationnel en intension

- PREFECTURE (numPref, nomVille, nbHab, nomPrefet)
- DEPARTEMENT (codeDep, libDep, nbHab, numPref)

Représentation d'une relation de type 1-1

Graphe des DF

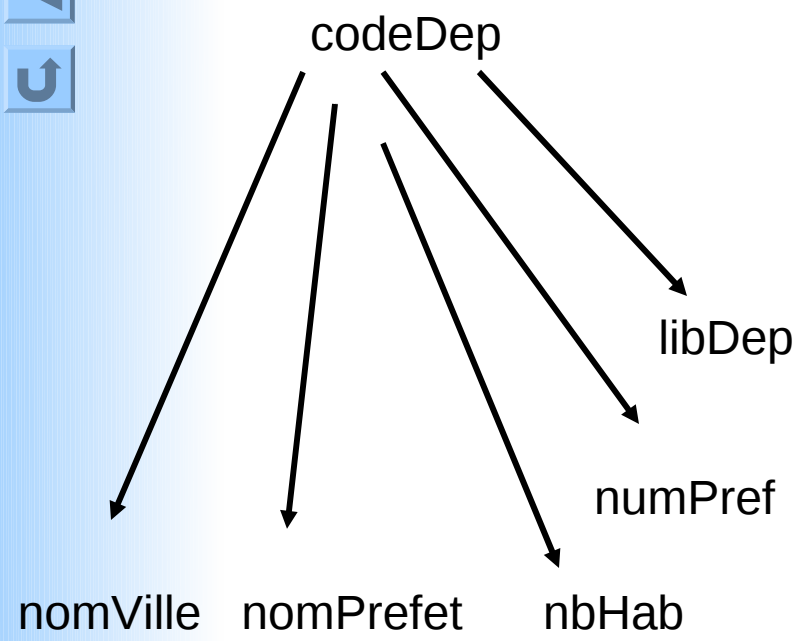


Schéma Entité/Association



Schéma relationnel en intension

- DEPARTEMENT (codeDep, libDep, nbHabDep, numPref, nomVille, nomPrefet, nbHabPref)

Représentation d'une relation de type 1-1

Graphe des DF

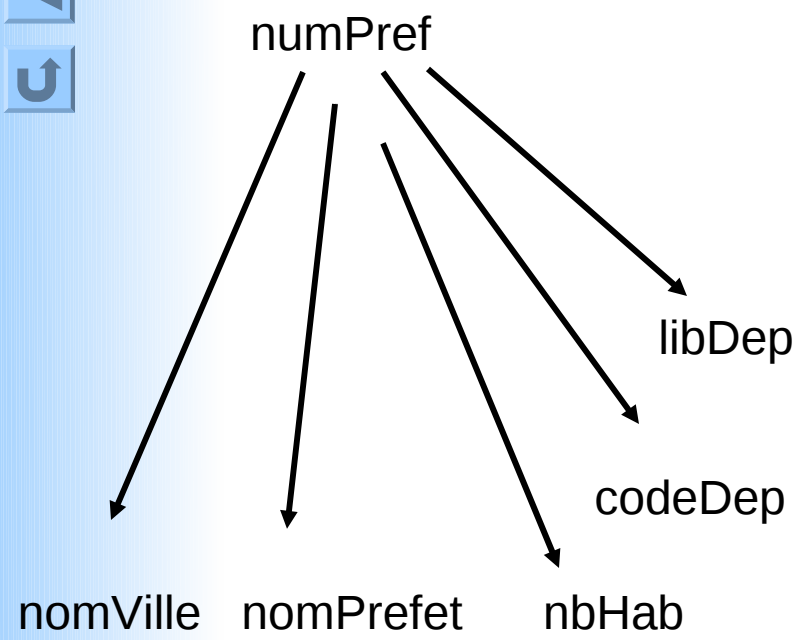


Schéma Entité/Association

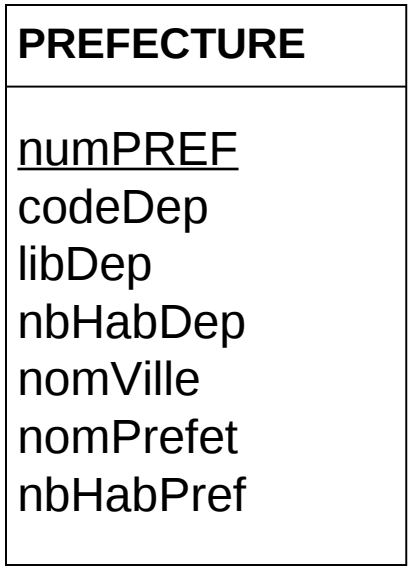


Schéma relationnel en intension

- PREFECTURE (numPref, codeDep, libDep, nbHabDep, nomVille, nomPrefet, nbHabPref)

Etapes de conception pour un schéma E/A

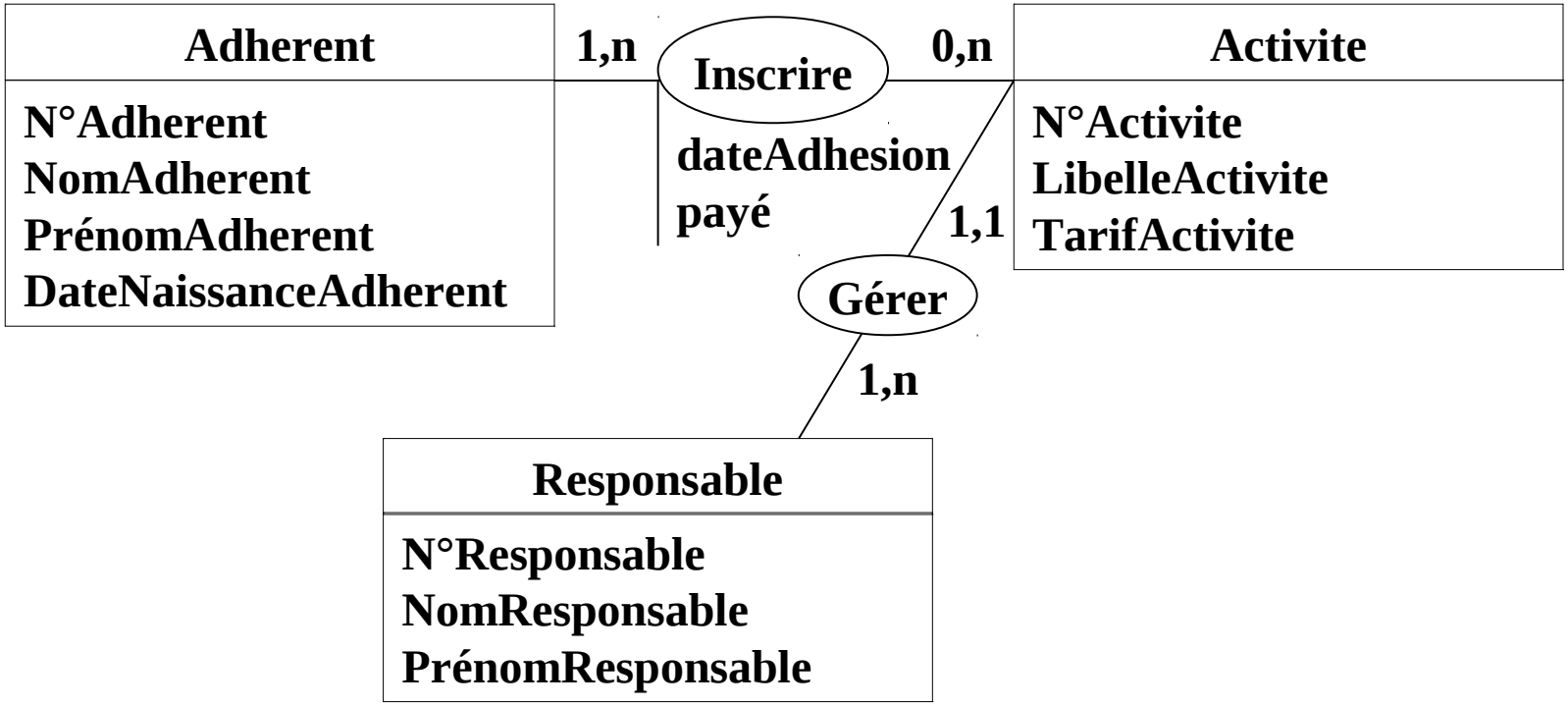
1. Analyser l'existant et lister les données (propriétés)
2. Epurer les données : synonymes, polysèmes
3. Constituer le dictionnaire des données
4. Dégager les entités naturelles
5. Rattacher les propriétés aux entités
6. Trouver les clés primaires des entités
7. Recenser les associations et leurs propriétés
8. Déterminer les cardinalités de chaque couple E-A
9. Appliquer les règles de passage pour en déduire le schéma relationnel

Forme normale 1

n°F	titre	acteur	cachet	CodeCat	LibCat
10	Camille Claudel	Adjani	200	COMD	comédie dramatique
10	Camille Claudel	Depardieu	250	COMD	comédie dramatique
20	Fenêtre sur cour	Kelly	120	COMD	comédie dramatique
30	Sueurs froides				
40	Shrek			DESA	dessin animé

Exercices de modélisation

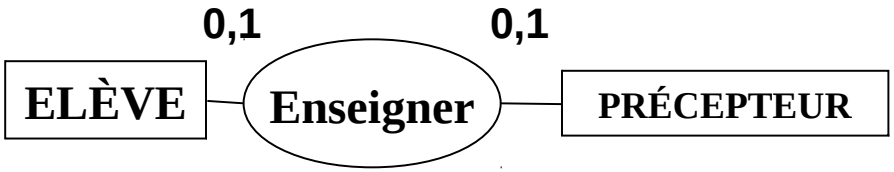
- Une association gère des adhérents (nom, prénom, date de naissance) qui s'inscrivent à un choix d'activités (Bridge, Tricot, Judo…). Un tarif de cotisation annuelle est fixé pour chaque activité. Chaque activité a un seul responsable (nom, prénom) et un nombre de participants maximum. Pour chaque activité, on indique la date d'adhésion du participant et s'il a payé sa cotisation.



Types d'associations (selon cardinalités maximales)

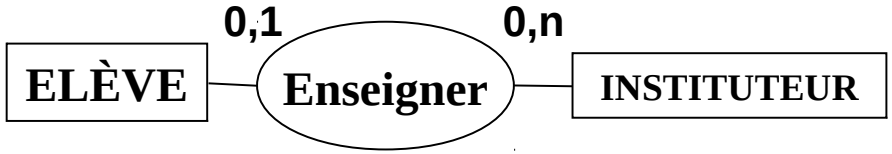
- Association 1 - 1 (un à un)





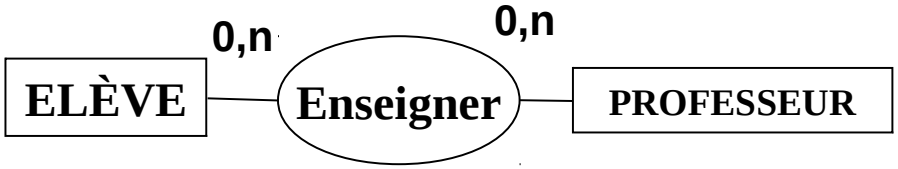
- Association 1 - N (un à plusieurs)





- Association N - M (plusieurs à plusieurs)





Etapes de conception pour un schéma E/A

1. Analyser l'existant et lister les données
2. Epurer les données : synonymes, polysèmes
3. Constituer une liste des données (dictionnaire des données)
4. Dégager les entités naturelles
5. Rattacher les propriétés aux entités
6. Trouver les clés primaires des entités
7. Recenser les associations et leurs propriétés
8. Déterminer les cardinalités de chaque couple E-A
9. Décomposer à l'aide des contraintes d'intégrité fonctionnelle
10. Vérifier la conformité du modèle avec les règles de gestion
11. Normaliser le modèle au moins en 3^{e} Forme Normale

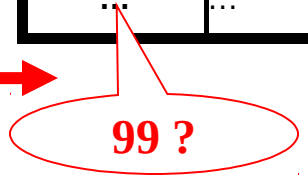
Intégrité référentielle

- Il y a intégrité référentielle quand :
  les valeurs de clé étrangère sont prises parmi les valeurs de clé primaire

num	nom	prenom	*codeDep*
1	Martin	Véra	*77*
2	Martin	Annie	*75*
3	Dupont	Sylvie	*77*
…	…	…	*99*

codeDep	libDep
75	Paris
77	Seine-et-Marne
…	…





Contraintes d'intégrité

- Intégrité de domaine / Intégrité de valeurs
  - type de données
  - valeur Null non autorisée (obligation de saisie)
  - valeur par défaut
  - spécifique (ex: note entre 0 et 20, date >=date du jour)
  - valeur d'attribut se référant à une autre colonne (Remise < Prix)
- Intégrité d'entité
  - clé primaire
  - unicité d'attribut(s) dans la table
- Intégrité référentielle
  - domaine de FK correspondant au domaine de PK
  - valeur de clé étrangère existant comme valeur de clé primaire

Comment normaliser une relation ?

n°F	titre	distribution	catégorie
10	Camille Claudel	Adjani;200, Depardieu; 250	COMD;comédie dramatique
20	Fenêtre sur cour	Kelly;120	COMD;comédie dramatique
30	Sueurs froides		
40	Shrek		DESA;dessin animé

Forme normale 2

n°F	titre	CodeCat	LibCat
10	Camille Claudel	COMD	comédie dramatique
20	Fenêtre sur cour	COMD	comédie dramatique
30	Sueurs froides		
40	Shrek	DESA	dessin animé

n°F	acteur	cachet
10	Adjani	200
10	Depardieu	250
20	Kelly	120

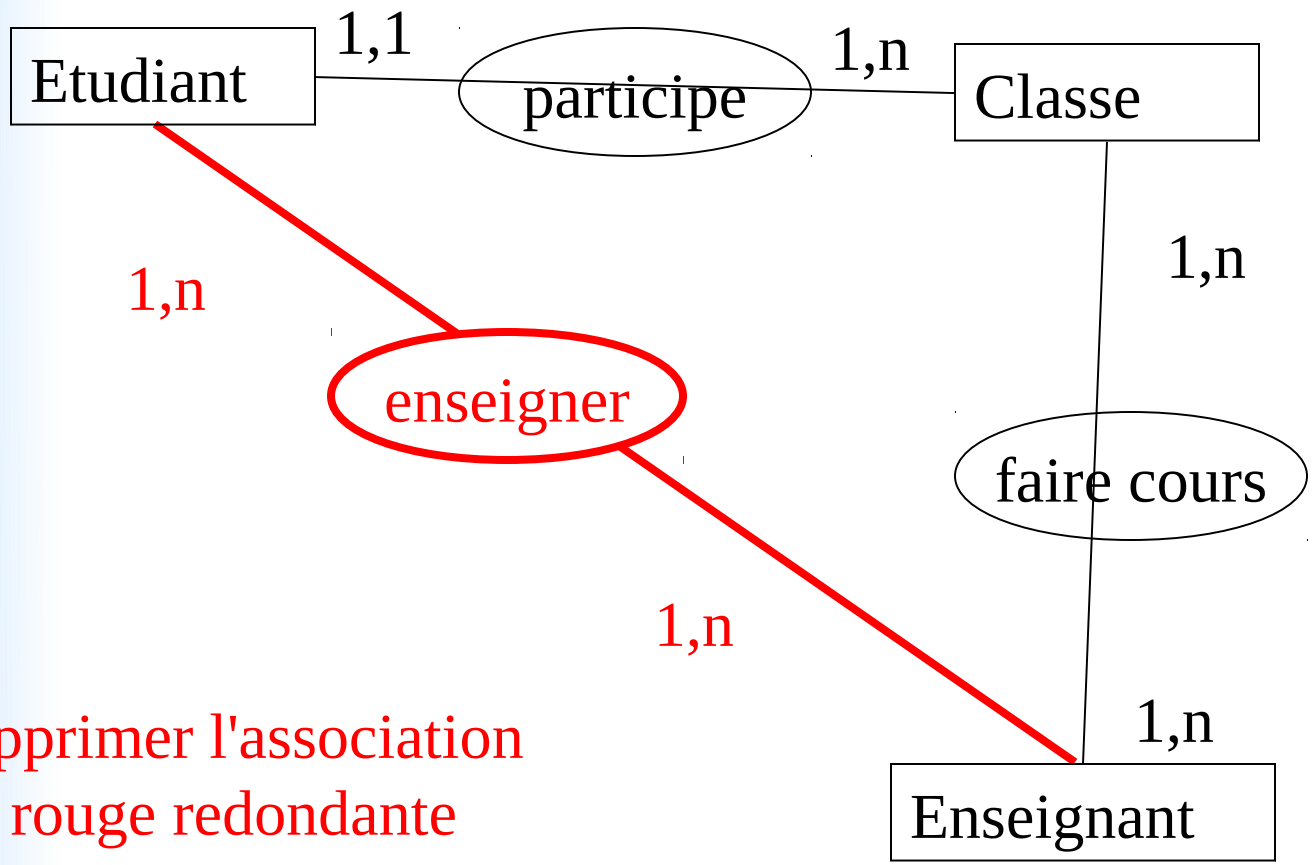
Forme normale 3

n°F	titre	CodeCat
10	Camille Claudel	COMD
20	Fenêtre sur cour	COMD
30	Sueurs froides	
40	Shrek	DESA

CodeCat	LibCat
COMD	comédie dramatique
DESA	dessin animé

n°F	acteur	cachet
10	Adjani	200
10	Depardieu	250
20	Kelly	120

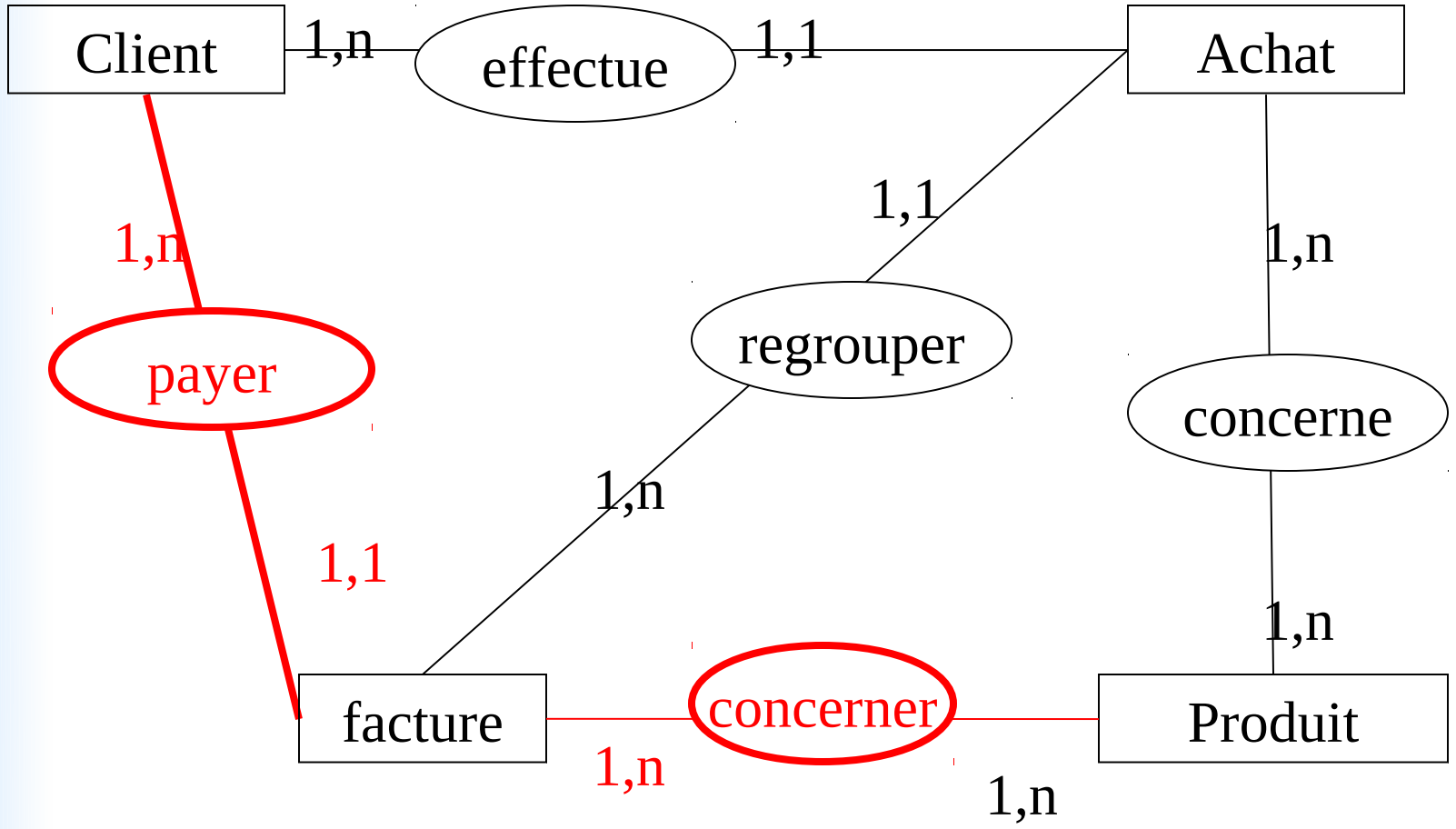
Association dérivée



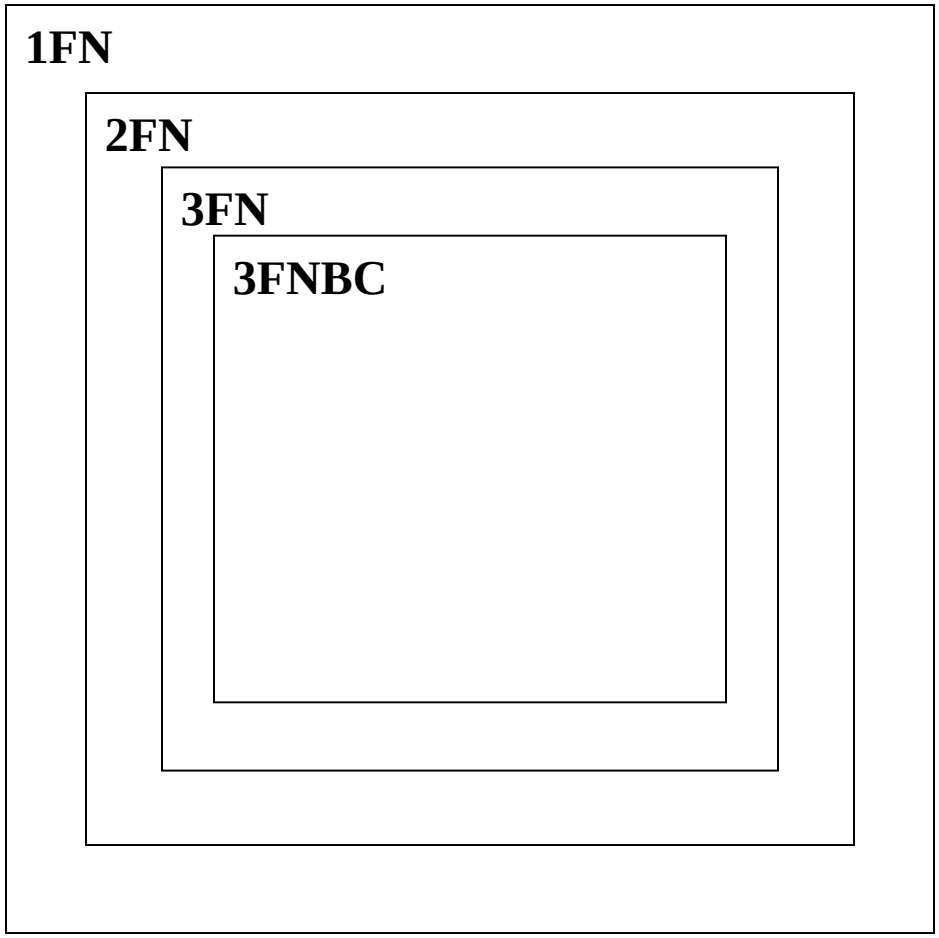
supprimer l'association
en rouge redondante

Association dérivée

supprimer
l'association
en rouge
redondante



Formes normales



Formes normales

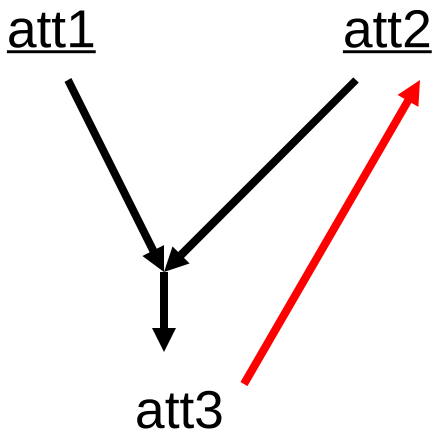
- 1FN : Tout attribut est monovalué
  contre-exemple : personne(Num, Nom, Prénom, PrénomEnfants)
- 2FN (clé composée) : Tout attribut non clé de peut dépendre d'une seule partie de clé, mais doit dépendre de l'ensemble de la clé.
  contre-exemple : Etudiant(NumEtud, Matière, Note, prénom)
- 3FN :
  Tout attribut non-clé doit dépendre directement de la clé et non pas transitivement.
  contre-exemple : Etudiant(NumEtud, codeUFR, libUFR, Directeur)
- 3FNBC (clé composée) : Un attribut d'une clé composée ne doit dépendre d'un attribut non clé ?

3FNBC

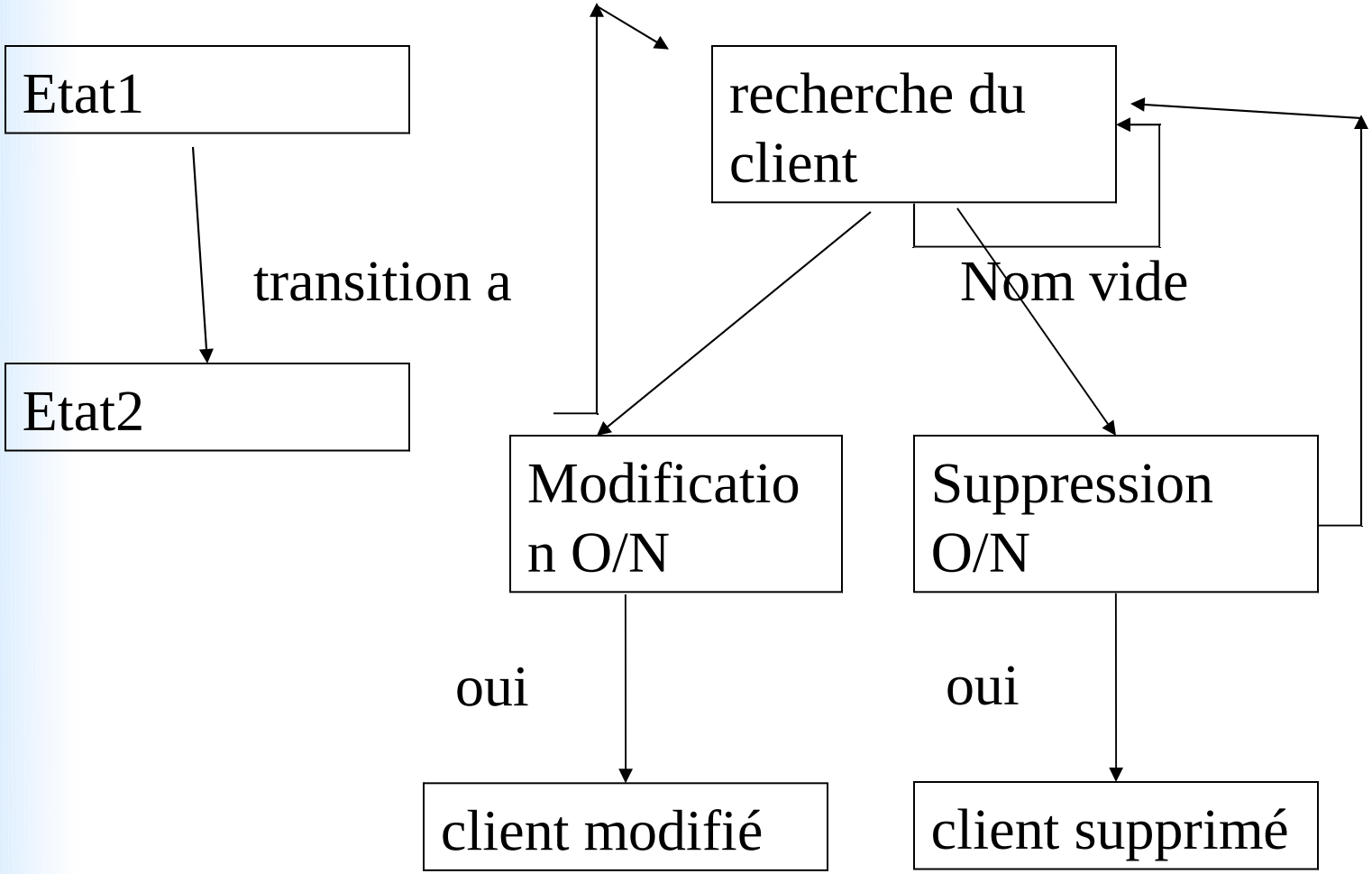
- table(Att1, Att2, Att3)



- table(Att1, Att3)
- table(Att3, Att2)



A verifier


Etat1
transition a
Etat2
recherche du client
Nom vide
Modificatio n O/N
Suppression O/N
oui
oui
client modifié
client supprimé

Aeroport

- AVION (NumA,)
- PILOTE (NumP, Nom, adresse)
- VILLE(CodeV, NomV)
- VOL(NumV, NumP, NumA, VilleD, VilleA, HeureD, HeureA)

Excursion

- Excursion(Code, Date, Lieu)
- Touriste(NumT,Nom, prenom, adresse)
- Inscription(NumInsc,NumT,Code)

Exemple 2bis : Normalisation de tables

num	nom	prenom	service	responsable
0001	Dupont	René	Personnel	Einstein
0002	Pairat	Luc	Recherche	Galileo
0003	Dupond	Jean	Personnel	Einstein

Emplo

num	nom	prenom	*cod-serv*
0001	Dupont	René	*pers*
0002	Pairat	Luc	*rech*
0003	Dupond	Jean	*prod*

cod-serv	lib-serv	responsable
pers	Personnel	Einstein
rech	Recherche	Galileo

Service

cod-serv	lib-serv	*num-responsable*
pers	Personnel	0005
rech	Recherche	0008

Exemple x : Résultat (solution a)

num	nom	prenom	formation	responsable	secretaire	tel_secretaire
0001	Dupont	René	Licence 1ère année économie	Einstein	Martin	12 34
0002	Dupond	Jean	Licence 1ère année économie	Einstein	Martin	12 34
0003	Deltour	Anne	Licence 2ème année économie	Galileo	Martin	12 34

ETUDIANT

num	nom	prenom	*form_cod*
0001	Dupont	René	*L1eco*
0002	Dupond	Jean	*L1eco*
0003	Deltour	Anne	*L2eco*

SECRETAIRE

secr_cod	secr_nom	secr_tel
MARTI02	Martin	01 22 22 22 22
…	…	…

RESPONSABLE

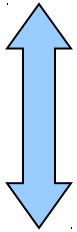
resp_cod	resp_nom
EINST01	Einstein
GALIL01	Galileo

FORMATION

form_cod	form_lib	*resp_code*	*secr_cod*
L1eco	Licence 1ère année économie	*EINST01*	*MARTI02*
L2eco	Licence 2ème année économie	*GALIL01*	*MARTI02*

Exemple x : Schéma relationnel (solution a)

- Etudiant (num, nom, prenom, formation, responsable, secretaire, tel_secretaire)



Transformation

de la table initiale (mal construite)

- Etudiant (**num**, nom, prenom, form_cod)
- Formation (**form_cod**, form_lib, secr_cod, resp_code)
- Secrétaire (**secr_cod**, secr_nom, secr_prenom, secr_tel)
- Responsable (**resp_cod**, resp_nom)

Exemple x : Résultat (solution b)

num	nom	prenom	formation	secretaire	tel_secretaire	responsable
0001	Dupont	René	Licence 1ère année économie	Martin	12 34	Einstein
0002	Dupond	Jean	Licence 1ère année économie	Martin	12 34	Einstein
0003	Deltour	Anne	Licence 2ème année économie	Martin	12 34	Galileo

ETUDIANT

num	nom	prenom	form_cod
0001	Dupont	René	L1eco
0002	Dupond	Jean	L1eco
0003	Deltour	Anne	L2eco

PERSONNEL

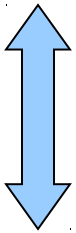
pers_cod	pers_nom	pers_tel	fonct
MARTI02	Martin	01 22 22 22 22	S
EINST01	Einstein		R
GALIL01	Galileo		R
…	…		

FORMATION

form_cod	form_lib	secr_cod	resp_code
L1eco	Licence 1ère année économie	MARTI02	EINST01
L2eco	Licence 2ème année économie	MARTI02	GALIL01

Exemple 3 : Schéma relationnel (solution b)

- Etudiant (num, nom, prenom, formation, secretaire, tel secretaire, responsable)



Transformation

de la table initiale (mal construite)

- Etudiant (num, nom, prenom, form_cod)
- Formation (form_cod, form_lib, secr_cod, resp_code)
- Personnel (pers_cod, pers_nom, pers_prenom, pers_tel)

Schéma E/A : exemple 3 (solution b)

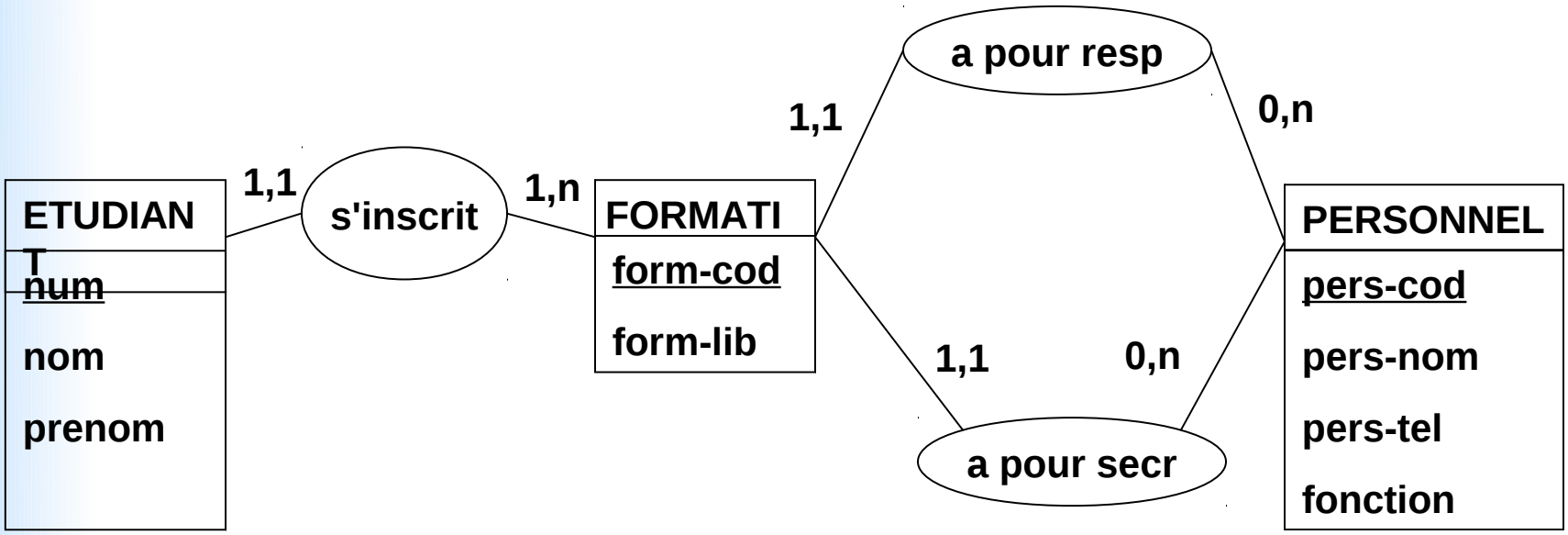


Schéma E/A : exemple 3 (solution a)

