

## DI

DEVELOPPEUR INFORMATIQUE

# Modèle Conceptuel de Données Normalisation

Dossier d'Informations et de Travaux Pratiques

<u>Module</u>	1
<u>Séquence</u>	4
<u>Capacité</u>	4

La normalisation consiste en un ensemble de règles introduites avec le modèle relationnel des données. Ces règles ont pour but de garantir la cohérence d'une base de données en particulier lors de sa mise à jour. De plus la normalisation permet de définir des entités et des associations qui sont parfaitement à l'image des objets du monde réel qu'elles modélisent.

Les règles de normalisation sont appelées **formes normales**.

Les formes normales ont été intégrées à la plupart des méthodes de conception utilisant des modèles de type entité-association.

Dans ce dossier nous développerons les 3 premières formes normales en précisant toutefois qu'il en existe 3 autres (forme de Boyce-Codd, 4<sup>ème</sup> et 5<sup>ème</sup> forme normale) et en rappelant l'utilisation des CIF pour la simplification d'un modèle de données

## 1. Première forme normale : 1FN

Une entité ou association est dite en 1FN si:

- elle possède un **identifiant**
- toutes ses propriétés sont **atomiques**.

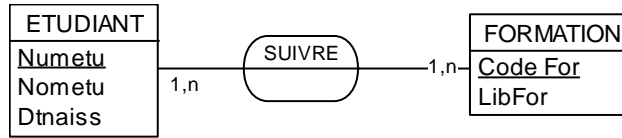
Une propriété est atomique si elle n'a, à un instant donné, qu'une seule valeur. Dans le cas contraire, elle correspond elle-même à une entité ou une association.

Une entité ETUDIANT a été modélisée par le schéma suivant:

ETUDIANT
<u>Numetu</u>
Nometu
Dtnaiss
LibFor 1
LibFor 2
...

La propriété LibFor représente le libellé des formations suivies par un étudiant. Cette propriété peut avoir plusieurs valeurs dans le mesure où un étudiant peut changer de filière. Cette solution présente l'inconvénient que dans certaines occurrences, des libellés de formation ne seront pas définis alors que dans d'autres, plusieurs seront renseignés (Quelle limite?). De plus la notion de formation possède une existence propre.

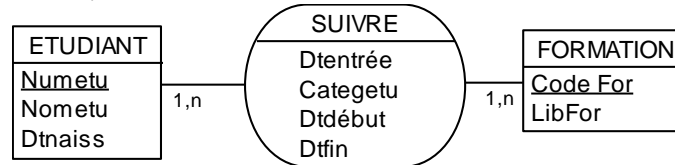
La solution ici est de créer une entité FORMATION.



## 2. Deuxième forme normale : 2FN

Une association est en 2FN si:

- elle est en 1FN
- toutes les propriétés n'appartenant pas à l'identifiant dépend pleinement de l'identifiant et ne peut se déduire d'un sous-ensemble de cet identifiant



Dtentrée: date d'entrée de l'étudiant dans l'organisme de formation

Categetu: catégorie de l'étudiant (Interne, externe,  $\frac{1}{2}$  pensionnaire)

Dtdébut: date d'entrée de l'étudiant dans une formation

Dtfin: date de sortie de l'étudiant dans une formation

L'association SUIVRE est identifiée par la concaténation des identifiants des entités de sa collection: Numetu et CodeFor. Les propriétés étant en DF avec l'identifiant, les dépendances modélisées sont:

Numetu, CodeFor → Dtentrée

Numetu, CodeFor → Categetu

Numetu, CodeFor → Dtdébut

Numetu, CodeFor → Dtfin

Or, d'après la signification (la sémantique) des données les DF sont les suivantes:

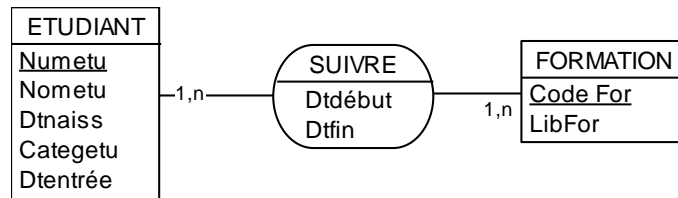
Numetu → Dtentrée

Numetu → Categetu

Numetu, CodeFor → Dtdébut

Numetu, CodeFor → Dtfin

La modélisation ci-dessous est en 2FN car elle est en 1FN et toutes les propriétés dépendent de l'intégralité de l'identifiant.



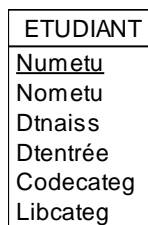
Le non respect de la 2FN entraîne évidemment des redondances donc des problèmes pour la mise à jour des données

### 3. Troisième forme normale : 3FN

Une entité ou une association est en 3FN si:

- elle est en 2FN
- toutes les DF entre ses propriétés sont directes

L'entité ETUDIANT suivante:



modélise les DF:

Numetu → Nometu  
 Numetu → Dtnaiss  
 Numetu → Dtentrée  
 Numetu → Codecateg  
 Numetu → Libcateg

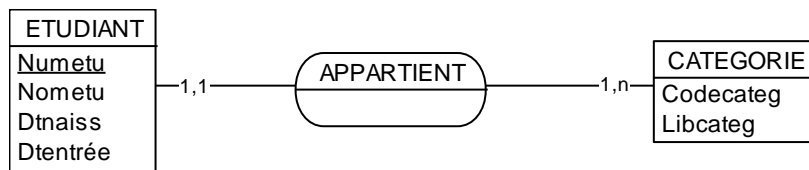
Or il existe la DF:

Codecateg → Libcateg

L'entité n'est pas en 3FN car il existe une dépendance non directe:

Numetu → Codecateg → Libcateg

Pour la rendre en 3FN, il faut créer une entité CATEGORIE:



Le non-respect de la 3FN entraîne également des redondances et des problèmes de mise à jour des données.

## Enoncé 1

Pour les Entités / Associations suivantes :

- Donnez leur forme normale
- Si besoin, apportez une solution pour respecter la troisième forme normale

COMMANDE\_CLIENT(N° client, n° produit, quantité commandée, nom produit)

LIGNE\_COMMANDE(N° commande, n° produit, quantité commandée)

CLIENT(N° client, nom client, Numéro représentant, nom représentant)

Dans les cas suivants :

- un client est en relation avec un seul représentant
- un client peut être en relation avec plusieurs représentants

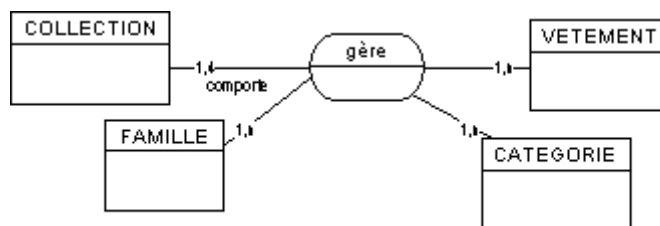
PRODUIT(N° produit, nom produit, n° atelier, nom chef d'atelier)

PRODUIT(N° produit, n° fournisseur, désignation produit, nom fournisseur)

## Enoncé 2

La société doit gérer de plus près ses stocks pour les collections (une par saison) de vêtements (blue-jean Levis, jupe droite pied de poule, ...) pour chaque famille (pantalon, robe, jupe, chemisier, ...) et pour chaque catégorie (homme, femme, enfant, ...)

Dans une première approche, ce domaine a été modélisé comme suit :

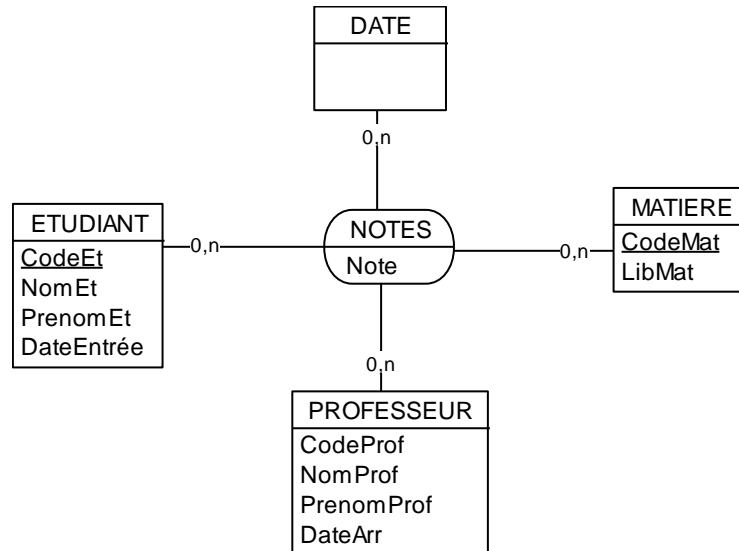


Une règle de gestion fait apparaître le fait que un vêtement appartient à une famille et une seule.

- Simplifier le modèle en tenant compte de cette règle.

### Enoncé 3

Le schéma ci-dessous représente un domaine "Notes". Pour une matière donnée, plusieurs notes sont attribuées à des dates différentes. D'autre part, un professeur n'enseigne qu'une matière

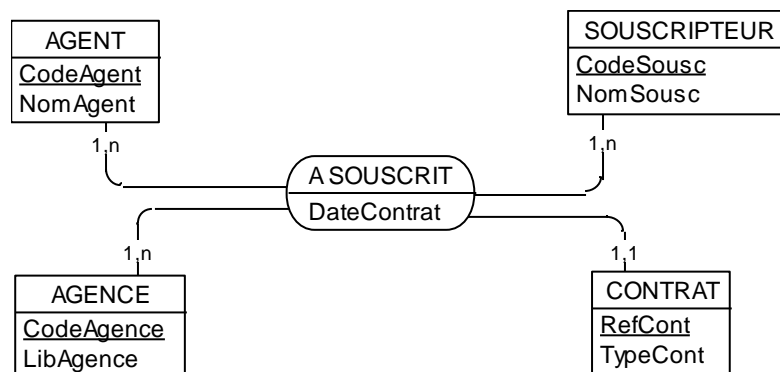


Apportez une critique au schéma proposé et la correction adaptée.

### Enoncé 4

Un modèle de données a été proposé pour le domaine "Souscription d'un contrat d'assurance".

Un souscripteur (client), en relation avec un seul agent, signe un contrat d'un certain type à une date donnée. Un agent est affecté à une seule agence. La modélisation proposée est la suivante:



Vous êtes chargé de valider ce schéma en respectant les règles de gestion formulées plus haut, vous repérez des anomalies génératrices de redondances donc de risques de lourdeur et d'incohérences; vous proposez un schéma simplifié et normalisé.