

# Modélisation

## Cours 02 – Modèle Relationnel normalisé

Bertrand LIAUDET

### SOMMAIRE

<b>Sommaire</b>	<b>1</b>
<b>modèle relationnel normalisé</b>	<b>2</b>
<b>Notion de dépendance fonctionnelle : DF</b>	<b>2</b>
Définition des DF	2
Théorème de l'équivalence entre DF et modèle relationnel	3
Propriétés de base des DF	3
Théorème de décomposition (de Casey et Delobel)	5
<b>Graphe des dépendances fonctionnelles : GDF</b>	<b>6</b>
Présentation	6
Exemples	6
Sémantique du graphe du GDF	10
<b>Les 3 premières formes normales de CODD et les 2 suivantes</b>	<b>11</b>
Principe de la normalisation	11
DF élémentaire et DF directe	11
1 <sup>ère</sup> forme normale : FN1 : la clé	12
Formes normales 2, 3, 4	14
2 <sup>ème</sup> forme normale : FN2 : toute la clé	15
3 <sup>ème</sup> forme normale : FN3: rien que la clé	16
4 <sup>ème</sup> forme normale : par réduction de la clé : facile !	17
5 <sup>ème</sup> forme normale, dite de BOYCE-CODD : BCNF : Complicé !	18
<b>Normalisations sémantiques :</b>	<b>20</b>
Présentation	20
6 <sup>ème</sup> normalisation : attributs calculés	20
7 <sup>ème</sup> normalisation : transitivité	21
8 <sup>ème</sup> normalisation : conflit de clé primaire – clé secondaire	22
9 <sup>ème</sup> normalisation : dépendance fonctionnelle entre espèce et genre	23
<b>Exercices</b>	<b>24</b>
Série 1 – Exercices théoriques	24
Série 2 – A partir de données	25
Série 3 – A partir d'un MR	27
Série 4 – A partir de données dans des tables excel - Normalisation	29

*Edition octobre 2018*

# MODELE RELATIONNEL NORMALISE

## Notion de dépendance fonctionnelle : DF

### Définition des DF

#### Présentation

Un attribut Y (ou premier groupe d'attributs) dépend fonctionnellement d'un attribut X (ou second groupe d'attributs), si étant donné une valeur de X, il lui correspond une valeur unique de Y (et ceci quel que soit l'instant considéré).

On dit que X détermine fonctionnellement Y et on note :

$$X \twoheadrightarrow Y$$

X est le déterminant, Y le déterminé.

#### Intuition

La flèche cible 1 et une seule cible.

#### Exemple

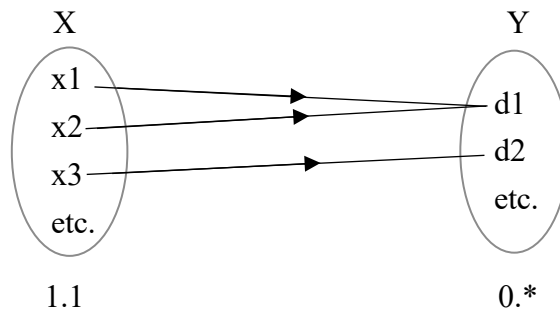
La clé de toute table détermine fonctionnellement tous les autres attributs de la table.

**Le numéro de sécurité sociale --> nom de la personne**

#### Schéma sagittal

On peut considérer X et Y comme des ensembles de valeurs

Si  $X \twoheadrightarrow Y$ , on a le schéma sagittal suivant :



Ce qu'on peut lire :

A une valeur de X correspond **une** valeur (1) de Y »

A une valeur de Y correspond **plusieurs** valeurs (\*) de X.

#### Exemple

Il y a numéros de sécurité sociale correspond un nom et un seul.

Au même nom (les parents et les enfants par exemple), correspondent plusieurs numéros de sécurité sociale.

## Théorème de l'équivalence entre DF et modèle relationnel

$$X \twoheadrightarrow Y \\ \Leftrightarrow R(\mathbf{X}, Y)$$

### Exemple 1

Si :  $NE \twoheadrightarrow \text{nomEmployé}$   
Alors : on a la table (**NE**, nomEmployé)

### Exemple 2

Si :  $NE \twoheadrightarrow \text{nomEmployé}, ND$   
Alors : on a la table (**NE**, nomEmployé, ND)

### Exemple 3

Si :  $NE \twoheadrightarrow \text{nomEmployé}, ND$   
et  $ND \twoheadrightarrow \text{nomDept}$   
Alors : on a les tables (**NE**, nomEmployé, #ND)

## Propriétés de base des DF

L'étude mathématique des DF fait apparaître plusieurs propriétés.

Ces propriétés sont assez évidentes intuitivement. Elles le sont plus encore avec un graphique.

Elles permettent de mieux comprendre la notion de dépendance fonctionnelle.

Elles interviennent en partie dans la normalisation.

Quelles que soient A, B, C, D des propriétés d'une relation :

### 1 : Réflexivité (évident)

$$A \twoheadrightarrow A$$

Exemple : numéro de sécu (NSS)  $\twoheadrightarrow$  NSS

### 2 : Augmentation (évident)

$$\text{si : } A \twoheadrightarrow B \\ \text{alors : } A, C \twoheadrightarrow B$$

Exemple : si : NSS  $\twoheadrightarrow$  date de naissance (DN),  
alors : NSS et adresse  $\twoheadrightarrow$  DN

### 3 : Transitivité (évident)

<b>si :</b> A --> B et B --> C <b>alors :</b> A --> C
---

Exemple : **si :** NSS --> DN **et** DN --> Signe astrologique (SA),  
**alors :** NSS --> SA1

### 4 : Pseudo-transitivité (pas évident et pas très utile !)

<b>si :</b> A --> B et B, C --> D <b>alors :</b> A, C --> D
---

Exemple : **si :** NSS --> DN  
**et** DN + heure et lieu de naissance (HLN)  
--> ascendant astrologique (AA)  
**alors :** NSS et HLN --> AA.

### 5 : Union (évident)

<b>si :</b> A --> B et A --> C <b>alors :</b> A --> B, C
--

Exemple : **si :** NSS --> DN  
**et** NSS --> département de naissance (DEPT)  
**alors :** NSS --> DN et DEPT

### 6 : Désunion (évident)

<b>si :</b> A --> B, C <b>alors :</b> A --> B et A --> C
--

Exemple : **si :** NSS --> DN et DEPT,  
**alors :** NSS --> DN  
**et** NSS --> DEPT

<sup>1</sup> Sur l'astrologie, pour un logicielien : <http://www.homme-moderne.org/societe/socio/teissier/analyse/philo.html>, et pour un cadre plus général : [http://bliaudet.free.fr/article.php3?id\\_article=82](http://bliaudet.free.fr/article.php3?id_article=82)

## Théorème de décomposition (de Casey et Delobel)

### Définition

Soit la relation :

**R (A1, A2, A3)**

S'il n'y a pas de clé primaire, c'est que le triplet des attributs forment la clé primaire.

Si A1 --> A2 alors on peut remplacer la relation R par :

**R1 (A1, A2)**

**R2 (#A1, A3)**

La jointure naturelle entre R1 et R2 produira la relation R de départ.

### Exemple

Un employé a un nom et est caractérisé par un numéro d'employé. Dans l'entreprise, il participe à plusieurs équipes, une équipe étant caractérisée par son nom qui est unique.

La relation :

**TableDeDépart(NE, nomEmp, nomEquip)**

peut traduire la situation.

Toutefois :

**NE -> nomEmp**

On peut donc créer les deux relations suivantes :

**Emp(NE, nomEmp)**

**Participer(#NE, nomEquip)**

La jointure suivante :

**Select e.ne, e.nomEmp, p.nomEquip**

**From emp e, participer p**

**Where e.ne = p.ne**

permet de reconstituer la TableDeDépart.

## Graphe des dépendances fonctionnelles : GDF

### Présentation

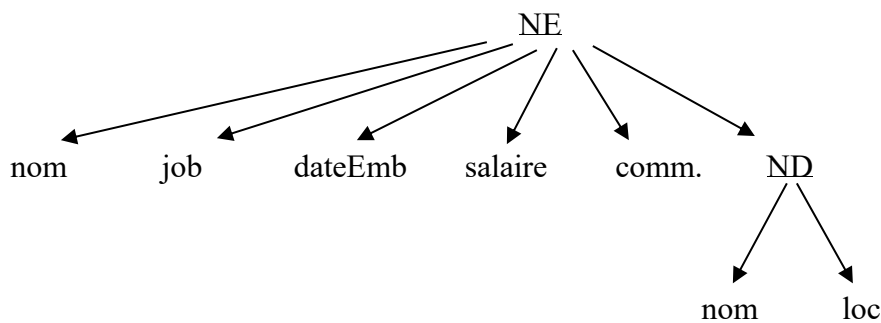
Le graphe des dépendances fonctionnelles montre toutes les dépendances fonctionnelles sous la forme d'un graphe orienté : chaque attribut est un nœud du graphe et chaque DF est une arrête du graphe.

### Exemples

#### Exemple 1

**EMPLOYES**(**NE**, nom, job, dateEmb, salaire, comm., #ND)

**DEPARTEMENTS** (**ND**, nom, ville)



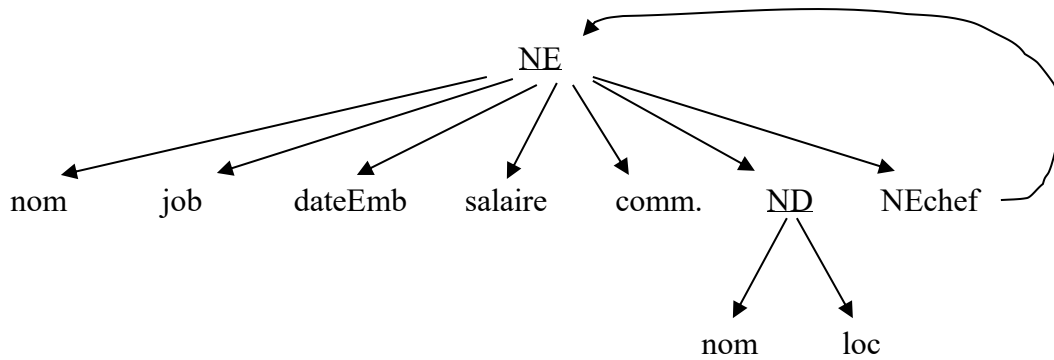
➤ *Le graphe se lit ainsi*

- NE est clé primaire de nom, job, dateEmb, salaire, comme, et ND
- ND est clé étrangère de NE
- ND est clé primaire de nom et loc.

### Exemple 2 : auto-jointure

**EMPLOYES**(NE, nom, job, dateEmb, salaire, comm., #ND, \*NEchef)

**DEPARTEMENTS** (ND, nom, ville)



➤ *Le graphe se lit ainsi*

- NE est clé primaire de nom, job, dateEmb, salaire, comm, ND et NEchef
- NEchef est clé étrangère réflexive de NE..
- ND est clé étrangère de NE.
- ND est clé primaire de nom et loc.

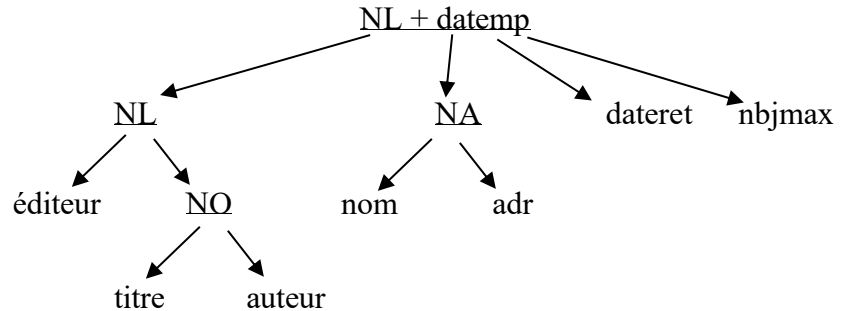
### Exemple 3 : clé primaire concaténée

ADHERENTS (NA, nom, adr)

OEUVRES (NO, titre, auteur)

LIVRES (NL, editeur, #NO)

EMPRUNTER(#NL, datemp, nbjmax, datret, #NA)



#### ➤ *Le graphe se lit ainsi*

- NL+datemp » est clé primaire concaténée de NA, dateret et nbjmax.
- NA est clé étrangère de NL+datemp.
- NA est clé primaire de nom et adr.
- NL est clé étrangère de NL+datemp.
- NL est clé primaire de éditeur et NO.
- NO est clé étrangère de NL.
- NO est clé primaire de titre et auteur.



#### Exemple 4 : jointure artificielle

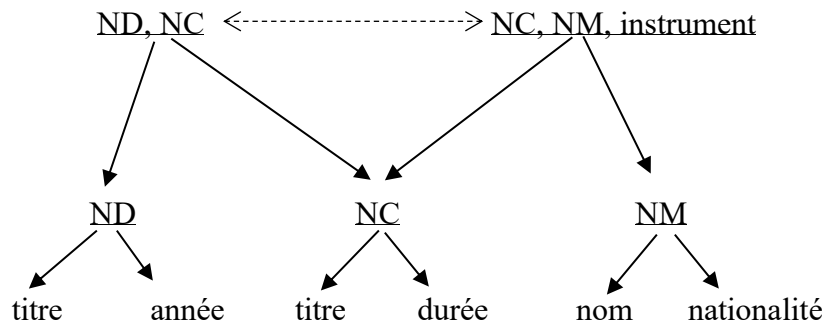
**DISQUES** (**ND**, titre, année)

**CHANSONS** (**NC**, titre, durée)

**MUSICIENS** (**NM**, nom, nationalité)

**JOUER** (**#NM, #NC, instrument**)

**REGROUPER** (**(#ND, #NC, piste)**)



➤ **Le graphe se lit ainsi :**

- **NC+ND** est clé primaire concaténée
- **ND** est clé étrangère de **NC+ND**
- **ND** est clé primaire de titre et année
- **NC** est clé étrangère de **NC+ND**
- **NC** est clé primaire de titre et durée
- **NC+NM+instrument** est clé primaire concaténée
- **NM** est clé primaire de **NC+NM+instrument**
- **NM** est clé primaire de nom et nationalité
- Il existe une jointure artificielle entre la table dont la clé primaire est (**ND, NC**) et celle dont la clé primaire est (**NC, NM, instrument**). Cette jointure passe par **NC**. Elle est représentée par une flèche à double sens en pointillé.

### Sémantique générale

1. Les attributs qui pointent vers d'autres attributs sont des clés primaires et sont soulignés.
2. Les attributs pointés par une clé primaire sont les attributs de la table correspondant à la clé primaire.
3. Les attributs pointés par une clé primaire CP1, et qui pointent eux aussi vers des attributs, sont à la fois clé étrangère dans la table de CP1 et clé primaire CP2 pour une nouvelle table.
4. Un attribut peut pointer sur lui-même. Dans ce cas, on précise sur la flèche le nom de l'attribut clé étrangère.
5. Quand un attribut d'une clé primaire concaténée est clé étrangère, on duplique cet attribut dans le graphe pour faciliter la lisibilité.
6. Les jointures artificielles entre deux attributs de deux tables sont représentées par une flèche à double sens en pointillé entre les attributs concernés.

### Passage d'un GDF à un schéma relationnel

Le déterminant est appelé « nœud parent ». Le déterminé : « nœud enfant ».

Tout nœud parent est clé primaire pour ses nœuds enfants.

Tout nœud enfant qui est aussi parent sera clé étrangère en tant qu'enfant.

### Passage d'un schéma relationnel à un GDF

A partir du schéma d'une base de données, on peut faire le graphe des dépendances fonctionnelles.

Trois règles permettent de construire le graphe :

1. Toutes les clés primaires sont des nœuds parents du graphe : un nœud peut donc être constitué de plusieurs attributs (cas des clés primaires concaténés).
2. Tous les attributs non clé-primaire sont des nœuds enfants du graphe.

**attribut (s) clé primaire** ———▶ **attributs non-clés primaires**

3. Les attributs n'apparaissent qu'une seule fois dans le graphe, sauf les clés étrangères participant à une clé primaire concaténée qui apparaissent dans la clé primaire concaténée et individuellement en tant que clé primaire

**#CP1, att** ———▶ **CP1**

## Les 3 premières formes normales de CODD et les 2 suivantes

### Principe de la normalisation

La normalisation consiste à éviter la duplication d'information pour éviter les incohérences.

À partir de l'analyse des dépendances fonctionnelles, plusieurs règles ont été trouvées qui permettent de garantir que la table a une forme normale.

### DF élémentaire et DF directe

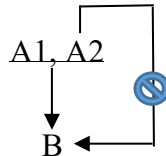
#### Notion de DF élémentaire

**La DF :  $A \twoheadrightarrow B$  est dite élémentaire si :**  
**aucune partie de A ne détermine B.**

Autrement dit, la DF est élémentaire si c'est « tout A » qui détermine B. Donc la DF n'est pas le résultat d'une augmentation (cf. propriété des DF).

On peut représenter A avec 2 attributs, A1 et A2 pour être plus clair. On obtient donc la règle :

**La DF :  $A1, A2 \twoheadrightarrow B$  est dite élémentaire si :**  
**aucune partie du déterminant (A1 ou A2) ne détermine B.**  
**Les DF  $A1 \twoheadrightarrow B$  et  $A2 \twoheadrightarrow B$  n'existent pas.**

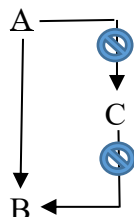


Pas de DF d'une partie de la clé vers un attribut non clé

#### Notion de DF directe

**La DF :  $A \twoheadrightarrow B$  est directe si :**  
**il n'existe pas C (distinct de A et B)**  
**tel que  $A \twoheadrightarrow C$  et  $C \twoheadrightarrow B$ .**

Autrement dit, la DF est directe s'il n'y a pas d'intermédiaire entre les deux attributs. Donc la DF n'est pas le résultat d'une transitivité (cf. propriété des DF).



Pas de DF de A vers B en passant par C

**1ère forme normale : FN1 : la clé**

**Définition**

**La FN1 concerne toutes les tables. Une table est en première forme normale si elle possède une clé primaire et si la valeur de tout attribut est « atomique », c'est-à-dire n'est pas constituée par une liste de valeurs.**

**Exemple 1**

Dans la table des Lions, on précise le nom du lion et les buffles qu'il a attaqué.

**LION**

NL	Nom du Lion	Liste des buffles attaqués
11	lion_1	buffle_1, buffle_3
12	lion_2	buffle_1, buffle_4
13	lion_3	buffle_1, buffle2, buffle_3

NL est la clé de la table.

Il y a **violation de la FN1**. En effet, il y a des listes de valeurs dans le 3ème attribut.

➤ *Normalisation : le modèle en FN1 devient :*

**LION**

NL	Nom du Lion
11	lion_1
12	lion_2
13	lion_3

**BUFFLE**

NB	Nom du Buffle
b1	buffle_1
b2	buffle_2
b3	buffle_3
b4	buffle_4

**ATTAQUER**

#NL	#NB
11	b1
11	b2
12	b1
12	b4
11	b1
13	b2
13	b3

Les listes de valeurs sont transformées en une table (BUFFLE) et en une table de liaison (EMPRUNTER) : les buffles sont attaqués par plusieurs lions.

## Exemple 2

Dans un département, on trouve une liste d'employés :

### DEPARTEMENT

ND	Nom du Département	Liste des employés du département
d1	dept_1	emp_1, emp_3
d2	dept_2	emp_2, emp_4
d3	dept_3	emp_5, emp_6, emp_7

NL est la clé de la table.

Il y a violation de la FN1. En effet, il y a des listes de valeurs dans le 3ème attribut.

➤ *Normalisation : le modèle en FN1 devient :*

### EMPLOYE

NE	Nom de l'emp.	#ND
e1	emp_1	d1
e2	emp_2	d2
e3	emp_3	d1
e4	emp_4	d2
e5	emp_5	d3
e6	emp_6	d3
e7	emp_7	d3

### DEPARTEMENT

ND	Nom du Dept
b1	dept_1
b2	dept_2
b3	dept_3

Les listes de valeurs sont transformés en une table avec une simple clé étrangère : l'employé travaille dans 1 département et 1 seul.

**Principe de la normalisation 2, 3, 4 et 5 :**

**Il n'y a pas d'autres dépendances fonctionnelles que celles de la clé primaire dans sa totalité vers les attributs non clé.**

Autrement dit, si on considère la relation (A1, A2, A3, A4), il ne doit pas exister :

- A1 --> A3 : FN2
- A3 --> A4 : FN3
- A1 --> A2 : FN4
- A3 --> A1 : FN5 dite BCNF

Les formes normales 2, 3 et 4, FN2, FN3 et FN4 sont les plus importantes.

La FN5 dite forme normale de Boyce-Cood est plus anecdotique.

La numérotation FN1, FN2 et FN3 est standardisée.

A partir de FN4, la numérotation n'est pas standardisée.

**2ème forme normale : FN2 : toute la clé**

**Définition**

La FN2 concerne les relations dont la clé est constituée de plusieurs attributs.

**Une relation est en FN2 si elle est en FN1 et si et si aucun attribut non clé ne dépend fonctionnellement d'une partie de cette clé (tout attribut dépend de toute la clé).**

Ce qu'on pourrait écrire schématiquement et intuitivement ainsi :

La relation (A1, A2, A3, A4) est **FN2** si il n'existe pas A1 --> A3

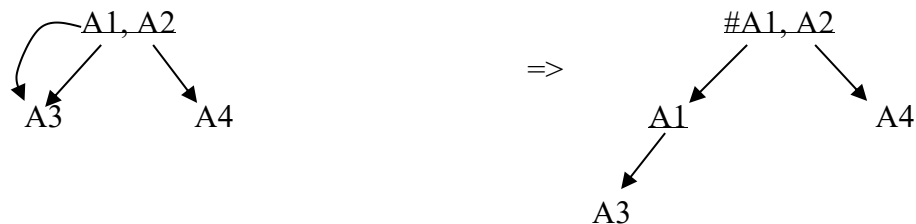
Dans le cas contraire, la correction consiste à créer deux tables :

(A1, A3)

(#A1, A2, A4)

Cette règle permet d'éviter la redondance des données, donc les risques d'incohérence.

**Représentation avec un GDF**



**Exemple**

La bibliothèque : les adhérents empruntent des livres.

EMPRUNTER (**NL**, **dateEmprunt**, dateRetour, dureeMaxAutorisée, titre, auteur, #NA)  
 ADHERENT (**NA**, nom, adresse, CP, ville, email)

Il y a violation de la FN2.

La solution est :

EMPRUNTER (**#NL**, **dateEmprunt**, dateRetour, dureeMaxAutorisée, #NA)  
 LIVRE (**NL**, titre, auteur)  
 ADHERENT (**NA**, nom, adresse, CP, ville, email)

**Avec des données**

On voit que la violation de FN2 conduit à des duplications d'information :

**EMPRUNTER**

NL	dateEmprunt	dateRetour	dureeMax	titre	auteur	NA
11	1 janvier	10 janvier	21	Titre 1	Auteur 1	a1
12	1 janvier	10 janvier	21	Titre 2	Auteur 2	a1
11	15 janvier	NULL	21	Titre 1	Auteur 1	a2

Duplication

**3ème forme normale : FN3: rien que la clé**

**Définition**

Une relation est en FN3 si elle est en FN2 et si aucun attribut non clé ne dépend fonctionnellement d'un attribut non-clé (tout attribut ne dépend que de la clé).

Ce qu'on pourrait écrire schématiquement et intuitivement ainsi :

La relation (A1, A2, A3, A4) est en FN3 s'il n'existe pas  $A3 \rightarrow A4$

Dans le cas contraire, la correction consiste à créer deux tables :

- (A3, A4)
- (A1, A2, #A3)

Cette règle évite la duplication d'information due à la transitivité.

Cette règle permet d'éviter la redondance des données, donc les risques d'incohérence.

**Représentation avec un GDF**



**Exemple**

Les employés dans le département : les adhérents empruntent des livres.

EMPLOYE (**NE**, nom, fonction, dateEmbauche, salaire, ND, nomDept, villeDept)

Il y a violation de la FN3.

La solution est :

EMPLOYE (**NE**, nom, fonction, dateEmbauche, salaire, #ND)

DEPARTEMENT (**ND**, nomDept, villeDept)

**Avec des données**

On voit que la violation de FN3 conduit à des duplications d'information :

**EMPLOYE**

NE	nom	fonction	dateEmbauche	salaire	ND	nomDept	villeDept
e1	nom 1	f 1	1 janvier	2000	d1	dept 1	ville 1
e2	nom 3	f 1	10 janvier	3000	d2	dept 2	ville 2
E3	nom 3	f 2	1 janvier	2500	d1	dept 1	ville 1

Duplication



## 4ème forme normale : par réduction de la clé : facile !

### Définition

**Une relation est en FN4 si elle est en FN3 et si aucune partie de la clé ne dépend fonctionnellement d'une autre partie de la clé**

Ce qu'on pourrait écrire schématiquement et intuitivement ainsi :

La relation (A1, A2, A3, A4) est en **FN4** s'il n'existe pas  $A1 \rightarrow A2$

Dans ce cas, la relation devient :

(A1, A2, A3, A4)

C'est ce qu'on peut appeler la **forme normale par réduction de la clé**.

### Exemple :

La table EMPRUNTER dans la BD Bibliothèque peut être vue ainsi :

EMPRUNTER (~~#NA~~, ~~#NL~~, **dateEmprunt**, dateRetour, duréeMaximumAutorisée)

Dans ce cas, on a :  $\#NL, \text{dateEmprunt} \rightarrow \#NA$

Une partie de la clé détermine une autre partie de la clé : il y a violation de la FN4.

Pour éviter cette violation, EMPRUNTER devient :

EMPRUNTER (~~#NL~~, **dateEmprunt**, dateRetour, duréeMaximumAutorisée, ~~#NA~~)

## 5ème forme normale, dite de BOYCE-CODD : BCNF : Complicé !

### Définition

La FN5 dite aussi forme normale de BOYCE-CODD (BCNF) concerne les relations dont la clé est constituée de plusieurs attributs.

**Une relation est en BCNF si elle est en FN4 et si aucun attribut non clé ne dépend fonctionnellement d'une partie de la clé.**

Ce qu'on pourrait écrire schématiquement et intuitivement ainsi :

La relation  $(A1, A2, A3, A4)$  est en BCNF s'il n'existe pas  $A3 \twoheadrightarrow A1$

Il n'y a pas de solution interne au MR pour ce problème.

### Remarque

C'est un cas assez rare.

### Solution technique

Quand on a  $R(A1, A2, A3)$  et  $A3 \twoheadrightarrow A1$

On crée une table  $(A3, A1)$

On vérifie ensuite avec un trigger à chaque insertion dans la table  $(A1, A2, A3)$  que le couple  $(A3, A1)$  existe dans la table  $(A3, A1)$ . Il n'y a pas de solution automatique par simple application des contraintes d'intégrité.

### Propriété intuitive

Quand on a  $R(A1, A2, A3)$  et  $A3 \twoheadrightarrow A1$ , on a  $R(A1, A2, A3)$ .

En effet, si on a un doublon sur  $A2, A3$ , alors il faut deux valeurs différentes pour les  $A1$  pour distinguer le triplet  $A1, A2, A3$ .

Or  $A3 \twoheadrightarrow A1$

Donc les deux valeurs de  $A1$  doivent être identiques.

Donc on ne peut pas avoir de doublon sur  $A2$  et  $A3$ .  $A2$  et  $A3$  sont donc clés primaires.

Donc  $R$  devient :  $(A1, A2, A3)$ .

Avec  $A3 \twoheadrightarrow A1$ , on retombe sur une simple violation FN2.

Cette propriété peut être intéressante à utiliser en fonction de la sémantique du modèle.

## Exemple

### ➤ *Situation de départ :*

Soit la relation : Etudiant, Professeur, Matière : (E, P, M)

On considère que :

- pour une matière et un étudiant donnés, il y a seulement un professeur : E,M ->P

On arrive au modèle suivant : (E, M, P)

### Exemple de données :

Elève	Matière	Prof
1	1	1
1	2	2
1	3	4
1	4	6
2	1	1
2	2	2
2	3	5
3	1	1
3	2	3

### ➤ *Ajout d'une contrainte*

- Un professeur enseigne une matière et une seule : P -> M

On se retrouve avec (E, M, P) et P -> M. On a donc une violation BCNF. Pour corriger cette violation, on crée la table (P, M) :

Prof	Matière
1	1
2	2
3	2
4	3
5	3
6	4

Si on veut ajouter pour l'élève 3 dans (E, M, P) la matière 4 enseigné par un nouveau prof, le 7, le système doit prévenir que le prof 7 n'existe pas. Il faut commencer par créer le prof 7 avec la matière 4.

Si on veut ajouter pour l'élève 3 dans (E, M, P) la matière 4 enseigné par le prof 5, le système doit prévenir que le prof 5 n'enseigne pas la matière 4 mais la 3.

## Normalisations sémantiques :

### Présentation

Les normalisations sémantiques sont des cas qui font intervenir la sémantique du modèle au delà de la seule dépendance fonctionnelle.

4 cas sont présentés :

- le problème des attributs calculés
- le problème de la transitivité
- le problème des conflits de clés primaires
- le problème des situation d'héritage

### 6ème normalisation : attributs calculés

Soit le schéma suivant :

(A, A1, A2)

(B, B1, B2)

Soit A2 attribut calculé tel que  $A2 = f(A1, B1)$

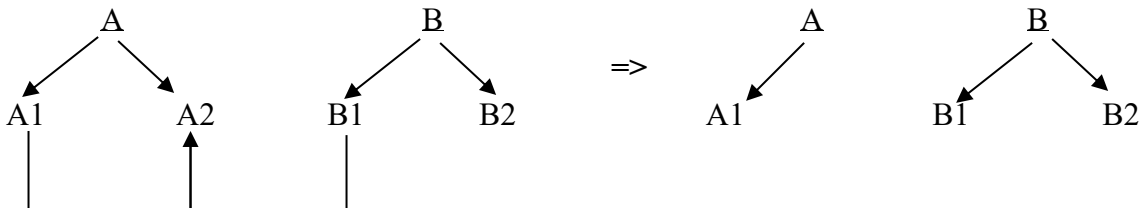
On a donc  $A1, B1 \rightarrow A2$

La normalisation consiste à supprimer les attributs calculés :

(A, A1)

(B, B1, B2)

### Représentation avec un GDF



### Analyse par les 3 premières formes normales

On a  $A1 \rightarrow A2$  : il y a donc violation de la 3FN

La correction est (A, #A1) et (A1, A2)

Mais on a alors  $A2 \rightarrow A1$  : on retombe sur un conflit de clé primaire. On peut donc se séparer d'un des deux attributs.

## 7<sup>ème</sup> normalisation : transitivité

Principe de la transitivité des DF :

si  $A \rightarrow B$

et  $B \rightarrow C$

alors  $A \rightarrow C$

Application : soit le schéma suivant :

(A, Ax, #B, #C)

(B, Bx, #C)

(C, Cx)

Ce schéma montre une transitivité des clés étrangères. La transitivité est la suivante :

$A \rightarrow B$

$+ B \rightarrow C$

$= A \rightarrow C$

La DF  $A \rightarrow C$  duplique donc les DF  $A \rightarrow B$  et  $B \rightarrow C$  et risque de conduire à une incohérence, si toutefois,  $A \rightarrow C$  a bien la même signification que  $A \rightarrow B$  et  $B \rightarrow C$ , **ce que seule l'analyse sémantique permet de dire.**

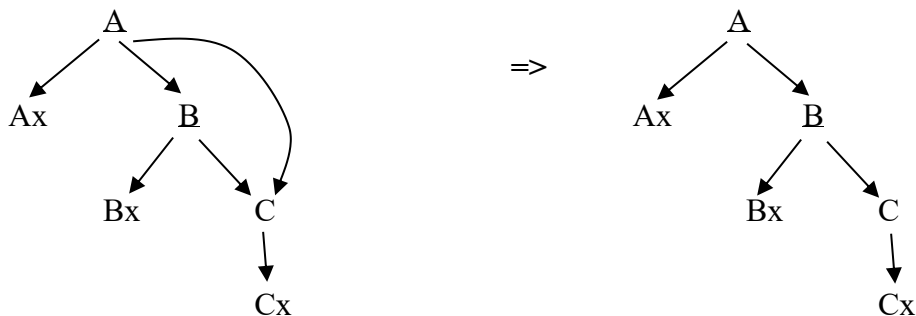
La normalisation consiste à supprimer les chemins directs :

(A, Ax, #B)

(B, Bx, #C)

(C, Cx)

### Représentation avec un GDF



### Analyse par les 3 premières formes normales

On a (A, B, C) avec  $B \rightarrow C$ , il y a donc violation de la 3FN.

La correction est donc : (A, #B) et (B, C)

**Principe**

Si

$A1 \twoheadrightarrow A2$  et  $A2 \twoheadrightarrow A1$  alors  $A1$  et  $A2$  sont clés candidates, c'est-à-dire deux attributs qui peuvent être clé primaire.

Alors, on a au choix :

Soit  $R1(A1, (A2))$  avec  $A2$  clé secondaire

Soit  $R1(A2, (A1))$  avec  $A1$  clé secondaire.

**Théorème des clés candidates**

Si :  $A \twoheadrightarrow C$  et  $B \twoheadrightarrow C$  et il n'existe pas  $D$  tel que  $A \twoheadrightarrow D$  ou exclusif  $B \twoheadrightarrow D$

Alors :  $A \twoheadrightarrow B$  et  $B \twoheadrightarrow A$

On a donc au choix :

Soit  $R1(A, (B), C)$

Soit  $R1(B, (A), C)$

**Approche sémantique**

Un des deux attributs peut s'avérer inutile.

On peut donc finir avec :

Soit  $R1(A, C)$

Soit  $R1(B, C)$

### Exemple

Les personnes (NP) ont un nom, un prénom, une adresse.

Les membres sont des personnes. Ils ont en plus une dateEntrée, une catégorie, une 2<sup>ème</sup> adresse.

Le modèle proposé est le suivant

Membres (NM, dateEntrée, catégorie, #NP)

Personnes (NP, nom, prénom, ad1)

On a #NP -> NM : il y a donc un conflit de clé primaire dans la table Membres.

### Solution

La solution suit le principe de la normalisation en cas de conflit de clé primaire.

On arrive à deux options

➤ *La clé primaire de l'espèce (NM) est significative*

Dans ce cas, on conserve les deux clés primaires.

On a donc deux solutions :

Membres (NM, (#NP), dateEntrée, catégorie)

Personnes (NP, nom, prénom, ad1)

#NP est clé secondaire

Ou :

Membres (#NP, (NM) dateEntrée, catégorie)

Personnes (NP, nom, prénom, ad1)

NM est clé secondaire

➤ *La clé primaire de l'espèce (NM) n'est pas significative*

Dans ce cas, on supprime NM :

Membres (#NP, dateEntrée, catégorie)

Personnes (NP, nom, prénom, ad1)

## Exercices

### Série 1 – Exercices théoriques

#### 1. Normalisation formelle : théorème de décomposition, formes normales

Soit les relations suivantes avec les DF associées :

- 1:  $R(A,B,C,D)$  avec  $(C \twoheadrightarrow D)$
- 2:  $R(A,B,C,D)$  avec  $(B \twoheadrightarrow C)$  ;  $(B \twoheadrightarrow D)$
- 3:  $R(A,B,C,D)$  avec  $(C \twoheadrightarrow D)$  ;  $(D \twoheadrightarrow A)$
- 4:  $R(A,B,C,D,E)$  avec  $(C \twoheadrightarrow D)$  ;  $(D \twoheadrightarrow B)$  ;  $(D \twoheadrightarrow E)$
- 5:  $R(A,B,C,D)$  avec  $(A,B \twoheadrightarrow C)$  ;  $(C \twoheadrightarrow D)$  ;  $(A,B \twoheadrightarrow D)$
- 6:  $R(A,B,C,D)$  avec  $(B \twoheadrightarrow C)$  ;  $(B \twoheadrightarrow D)$  ;  $(A,B \twoheadrightarrow C)$  ;

Pour chaque relation :

- Faire un schéma de BD en forme normale. Pour cela, utiliser le théorème d'équivalence entre DF et MR et le théorème de décomposition.
- Faire le graphe des dépendances fonctionnelles et le graphe des tables correspondant.
- Quelle est la clé primaire de la relation R de départ ?
- Identifier les violations de forme normale dans la relation de départ ?

#### 2. Normalisation formelle : BCNF

Soit les relations suivantes avec les DF associées :

- 1:  $R(A,B,C,D)$  avec  $(A,B \twoheadrightarrow C)$  ;  $(C \twoheadrightarrow D)$  ;  $(D \twoheadrightarrow A)$
- 2:  $R(A,B,C,D,E)$  avec  $(A, B \twoheadrightarrow C)$  ;  $(C \twoheadrightarrow D)$  ;  $(D \twoheadrightarrow B)$  ;  $(D \twoheadrightarrow E)$

Pour chaque relation :

- Faire un schéma de BD en forme normale. Pour cela, utiliser le théorème d'équivalence entre DF et MR et le théorème de décomposition.
- Faire le graphe des dépendances fonctionnelles et le graphe des tables correspondant.
- Quelle est la clé primaire de la relation R de départ ?
- Identifier les violations de forme normale dans la relation de départ ?
- **Quelle(s) contrainte(s) supplémentaire(s) faut-il gérer pour garantir l'intégrité des données ?**



## Série 2 – A partir de données

### 1. La table des ventes de voitures d'occasion

Soit la table suivante contient les informations sur la vente de voiture d'occasion.

NumV	NumIm	Marque	Type	Puis	Coul.	NumP	Nom	Prénom	Date	Prix
5	672 RH 75	RENAULT	R 12 TS	6	ROUGE	1	MARTIN	PAUL	10/02/98	10 000
1	800 AB 64	PEUGEOT	504	9	VERTE	1	MARTIN	PAUL	11/06/95	30 000
3	686 HK 75	CITROEN	2 CV	2	BLEUE	3	DUPOND	JEAN	20/04/97	5 000
2	720 CD 60	CITROEN	AMI 8	5	BLEUE	2	DUPOND	JEAN	20/08/96	15 000
4	672 RH 75	RENAULT	R 12 TS	6	ROUGE	4	DURANT	PIERRE	11/09/97	12 000

NumV est le numéro de la vente.

NumIm est le numéro d'immatriculation. Il ne peut pas être changé.

NumP est le numéro de la personne.

On connaît la date et le prix de vente.

Le type détermine la puissance et la marque.

Une voiture peut être vendue plusieurs fois (occasion)

- 1) Quelles sont les dépendances fonctionnelles ?
- 2) Faire le GDF.
- 3) Quelle est la clé primaire de la table de départ ?
- 4) Quelles sont les violations de formes normales ?
- 5) Faire le MR normalisé et valorisé.
- 6) Faire le graphe des tables.
- 7) Répartissez toutes les données de la table de départ dans les tables du MR valorisé.
- 8) Ecrire la requête SQL qui permet de reproduire la table de départ à l'identique.
- 9) Faire le MEA

### 3. Les avions

Soit le tableau de 5 lignes et 17 attributs suivant :

numVol	lieuDépart	lieuArrivée	heureDépart	heureArrivée	numVolEffectif	dateDépart	dateArrivée
1	Paris	Toulouse	7h30	8h30	250	01-juin-08	01-juin-08
1	Paris	Toulouse	7h30	8h30	250	01-juin-08	01-juin-08
1	Paris	Toulouse	7h30	8h30	262	02-juin-08	02-juin-08
1	Paris	Toulouse	7h30	8h30	279	03-juin-08	03-juin-08
2	Paris	Nice	16h00	17h15	255	01-juin-08	01-juin-08

Suite du tableau :

numAvion	typAvion	nbPlaces	dateRésa	nbRésa	totalPayé	numCli	nomCli	adrCli
1	A320	150	28-mai-08	2	80	1	Dupond	rue truc
1	A320	150	30-mai-08	1	45	2	Durand	rue chose
1	A320	150	30-mai-08	1	45	2	Durand	rue chose
2	B727	120	02-juin-08	3	100	3	Dugenou	rue truc
2	B727	120	30-mai-08	1	60	3	Dugenou	rue truc

- 1) Quelles sont les dépendances fonctionnelles ?
- 2) Faire le GDF.
- 3) Quelle est la clé primaire de la table de départ ?
- 4) Quelles sont les violations de formes normales ?
- 5) Faire le MR normalisé et valorisé.
- 6) Faire le graphe des tables.
- 7) Répartissez toutes les données de la table de départ dans les tables du MR valorisé.
- 8) Ecrire la requête SQL qui permet de reproduire la table de départ à l'identique.
- 9) Faire le MEA

### 4. L'éditeur

Soit le tableau de données suivant :

Editeur	Num_ouvrage	Langue
Ed1	1, 2, 3	français, italien, anglais
Ed2	4, 5	anglais, français
Ed3	3, 6	français, anglais, russe, espagnol

Ce tableau traduit la situation suivante :

Un éditeur édite des ouvrages. Chaque éditeur édite systématiquement tous ses ouvrages dans les mêmes langues. Un ouvrage peut être édité par plusieurs éditeurs.

- 1) Quelles sont les dépendances fonctionnelles ?
- 2) Faire le GDF.
- 3) Quelle est la clé primaire de la table de départ ?
- 4) Quelles sont les violations de formes normales ?
- 5) Faire le MR normalisé et valorisé.
- 6) Faire le graphe des tables.
- 7) Répartissez toutes les données de la table de départ dans les tables du MR valorisé.
- 8) Ecrire la requête SQL qui permet de reproduire la table de départ à l'identique.
- 9) Faire le MEA

**1. La table des commandes**

Commandes(NC, dateCommande, numClient, nomClient, adClient, réductionCom)

ArticlesCommandes(NC, numArticle, nomArticle, descriptionArticle, quantitéCommandée, prixUnitaireArticle, prixTotal, réduction)

Une commande est constituée de plusieurs articles. Un article commandé l'est dans une certaine quantité et pour un certain prix unitaire. Chaque article commandé peut bénéficier d'une réduction particulière. Chaque commande peut bénéficier d'une réduction particulière. Pour calculer le prix réel de l'article, on prend en compte une seule réduction : la plus favorable au client.

- 1) Quelles sont les dépendances fonctionnelles ?
- 2) Faire le GDF.
- 3) Quelle est la clé primaire de la (ou des) table(s) de départ ?
- 4) Quelles sont les violations de formes normales ?
- 5) Faire le MR normalisé et valorisé.
- 6) Faire le graphe des tables.
- 7) Faire le MEA

## 2. Les bouquets

Les tables suivantes permettent de gérer les commandes d'une chaîne de magasins de fleurs qui livrent des bouquets de fleurs à domicile :

Bouquet (numBouquet, nom, description, prix)

Commander (numBouquet, numClient, dateHeureCommande, dateHeureLivraisonPrévue, dateHeureLivraisonEffective, prix, numMagasin, nomMagasin, adresseMagasin, CPmagasin, villeMagasin)

Client (numClient, nom, prenom, adresse, telephone, admail, CP, ville)

Poster (#numClient, numCourrier, libellé, type, dateEnvoi)

CPRegion (CPcourt, région)

Le catalogue est stocké dans la table *Bouquets*. La table *Commander* gère les commandes et les livraisons. Pour chaque commande, on enregistre le prix de vente effectif, la date et l'heure de la commande, et la date et l'heure de livraison prévue. Quand la livraison est faite, on enregistre la date et l'heure de livraison.

La BD considère qu'un code postal n'identifie pas la ville. Par contre, le code postal court, sur 2 chiffres (75, 76, etc.) identifie la région.

La chaîne de magasin envoie des courriers publicitaires, par mail ou par la poste (c'est l'attribut type qui le précise) à certains de ses clients. Un même courrier publicitaire est envoyé en même temps à une série de clients. Le libellé des courriers et la date d'envoi sont conservés, ainsi que la liste des clients à qui on a posté le courrier.

- 1) Quelles sont les dépendances fonctionnelles ?
- 2) Faire le GDF.
- 3) Quelle est la clé primaire de la (ou des) table(s) de départ ?
- 4) Quelles sont les violations de formes normales ?
- 5) Faire le MR normalisé et valorisé.
- 6) Faire le graphe des tables.
- 7) Faire le MEA
- 8) Faire le graphe de la question pour la question suivante : quels sont les clients de la région bretonne qui ont commandé un bouquet dans le mois suivant le ou les mails envoyés le 15 juin 2018.

## **L'association**

### ➤ **Présentation**

Dans un fichier excel, on trouve plusieurs onglets

#### Onglet « Chèques et formulaires » :

Il contient les informations des chèques reçus et les informations des formulaires qui accompagnaient les chèques, quand il y en avait un. Contient aussi un « id » de personne, pour chaque chèque reçu. Cette table excel est l'équivalent d'une pile de chèques accompagnés ou pas de formulaires.

#### Onglet « Contact » :

Il contient une liste de personnes ont été sollicitées pour envoyer des dons mais qui n'en n'ont pas répondu. Ces personnes ont un « id ».

#### Onglet « Courrier » :

Il contient les informations des 3 courriers envoyés par l'association. Les formulaire sont des coupons papier au format A4/3 à remplir par le donateur. Le ou les documents associés au courrier c'est la brochure publicitaire qu'on envoie.

#### Onglets « listes » :

Il contiennent la liste des personnes à qui l'association a envoyé chacun des 3 courriers. Quand c'est un courrier par mail, on a l'adresse mail. Quand c'est un courrier postal, on a l'adresse postale.

#### Onglet « Retour de mail » :

Il contient les adresses mail revenu comme « undelivered mail »

#### Onglet « NPAI » :

Il contient les informations des personnes dont l'adresse postale n'est plus valable

### ➤ **Objectif de l'exercice**

Vous devez remplir des tableaux excel de type « lignes-colonnes » permettant de ranger l'information manipulées en évitant les duplications, autrement dit sous la forme de tables normalisées.

### ➤ **Etapas du travail**

Consolidez d'abord un tableau de personnes.

Ensuite, faites un tableau de dons.

Et finalisez avec toutes les autres informations !

### ➤ **Passage à une BD MySQL**

Vous pouvez importer les données des tables excel dans une BD MySQL en passant par un format CSV et en faisant une importation (avec phpMyAdmin de WAMP ou MAMP par exemple).

Afficher les donateurs avec leur nombre de dons et le montant total des dons qu'ils ont effectués.

Vous pouvez ensuite mettre à jour le type des champs et toutes les contraintes d'intégrité

Vous pouvez enfin exporter le code de création de la BD (dump de la BD).

### **Modélisation à réaliser sans jeu de données**

1. Faire le schéma de la BD correspondant au MR Brut
2. Faire le MEA correspondant
3. Faire le graphe des tables correspondant.
4. Pour chaque table, préciser le type de la clé primaire et la sémantique de la table (ontologie relationnelle).
5. Mettre des tuples dans vos tables.
6. Valoriser le modèle (obligatoire, unique, modifiable, valeur par défaut)
7. Quels attributs calculés peut-on envisager ?