

# Les formes normales et dépendances fonctionnelles

---

OU COMMENT NORMALISER OU... DÉNORMALISER

# Un peu d'humour...

---



# Normalisation : but

---

Imposer des règles de construction de la structure afin de

- Respecter la cohérence des données
- Eviter des redondances

Il existe **5 formes normales** : dans la pratique les 3 premières doivent être respectées

# Intérêt de la normalisation : un exemple

---

Soit l'entité suivante:

Fournisseur	
Nom	Varchar
Adresse	Varchar
Produit	Int
Quantité	Int

Un exemple de données pourrait être:

Nom	Adresse	Produit	Quantité
Jean Martin	Rue de la gare 21, 1000 Lausanne	Chaises	15
		Tables	20
François Dupont	Avenue des Toises 8, 1005 Lausanne	Lit	2
		Lampes	4
Jean Martin	Chemin de Chalamont 15, 1400 Yverdon	Tables	20

# Intérêt de la normalisation : problèmes

---

- *Quels sont les problèmes de cette entité?*
- **Pas de clé primaire** : Est-ce que les deux «Jean Martin» sont la même personne?
- **L'adresse n'est pas décomposée** : Comment trouver tous les fournisseurs présents à Lausanne?
- Dans la table correspondant à cette relation on aura plusieurs fois le même tuple «Nom / Adresse»
  - L'adresse sera dupliquée (redondance des données)
  - Si l'on veut modifier l'adresse, il faudra effectuer une recherche et modifier tous les tuples correspondants
  - Si l'on insère un nouveau produit pour un fournisseur déjà existant, il faudra vérifier que l'adresse soit identique
  - Si l'on veut supprimer un fournisseur il faudra supprimer tous les tuples correspondants

# Avantages de la normalisation

---

- Les avantages de la normalisation sont donc les suivants:
  - Diminution des risques d'incohérences
  - Eviter une mise à jour multiples des données
  - Diminution de la taille de la base de données
- Anomalies possibles
  - De redondance : même informations présente plusieurs fois
  - Lors de l'ajout : l'ajout de certaines informations n'est possible que si d'autres sont déjà présentes
  - Lors de la suppression : suppression de données à conserver lors de la suppression de certaines données
  - Lors de la mise à jour : la modification d'une information doit être répercutée autant de fois qu'elle apparait dans une table

# Détecter les anomalies

---

*Quelles sont les anomalies détectées dans cette relation?*

Produit	Quantité	Couleur	Fournisseur	Adresse
parapluie	110	rouge	Martin	Lausanne
chapeau	50	vert	Martin	Lausanne
sac à main	65	noir	Doe	Fribourg
parasol	15	jeune	Martin	Lausanne
ombrelle	5	rouge	Martin	Lausanne
ceinture	25	vert	Dupond	Genève
sac à main	65	noir	Legrand	Neuchâtel

**Redonance** : l'information du fournisseur et de son adresse apparaît plusieurs fois

**Ajout** : Il faut effectuer une commande pour ajouter un fournisseur

**Suppression** : si l'on supprime le produit «ceinture» on supprime toutes les infos du fournisseur «Dupond»

**Mise à jour** : si le fournisseur «Martin» change d'adresse il faudra modifier tous les tuples correspondants

# Dépendances fonctionnelles

---

- Permet de passer d'un ensemble de données non structurées (DD) à un modèle de données formés d'entités et d'associations
- On dit que **b** est en dépendance fonctionnelle (DF) de **a** si à une valeur quelconque de la propriété **a** on ne peut faire correspondre qu'une seule valeur de **b**

**On note  $a \rightarrow b$**

- On dit alors que **a** détermine **b** et **b** dépend fonctionnellement de **a**
- Si on connaît la valeur de **a** on peut déduire **une seule valeur de b** mais la réciproque n'est pas vraie (si on connaît **b** on ne peut pas déduire **a**)

**Numéro client  $\rightarrow$  Nom du client**

**Nom du client  $\nrightarrow$  Numéro du client**



# Dépendances fonctionnelles

---

Quelques propriétés des dépendances fonctionnelles:

- **Décomposition** : si  $A \rightarrow BC$  alors  $A \rightarrow B$  et  $A \rightarrow C$
- **Composition** : si  $A \rightarrow B$  et  $A \rightarrow C$  alors  $A \rightarrow BC$
- **Transitivité** : si  $A \rightarrow B$  et  $B \rightarrow C$  alors  $A \rightarrow C$

# Exemple DF

---

Même exemple avec les fournisseurs :

Produit	Quantité	Couleur	Fournisseur	Adresse
parapluie	110	rouge	Martin	Lausanne
chapeau	50	vert	Martin	Lausanne
sac à main	65	noir	Doe	Fribourg
parasol	15	jeune	Martin	Lausanne
ombrelle	5	rouge	Martin	Lausanne
ceinture	25	vert	Dupond	Genève
sac à main	65	noir	Legrand	Neuchâtel

- Si l'on connaît le fournisseur on connaît son adresse:
  - Donc Fournisseur → Adresse
- Si l'on connaît un produit on ne connaît pas forcément son fournisseur:
  - Donc pas de DF Produit ↗ Fournisseur

# Exemple DF

---

Pour l'exemple précédent on a la relation *Livraison* (*Produit*, *Quantité*, *NomFournisseur*, *Adresse*)

Quelles sont les DF de cette relation?

**Fournisseur → Adresse**

Dans ce cas on doit aussi rajouter un n° de fournisseur pour la non unicité des noms

**(Fournisseur, date) → Produit, quantité**

Ici on gère une commande qui est déterminée par un fournisseur (son n°) et une date de commande

# 1<sup>ère</sup> forme normale (1FN)

---



*Toute intersection de ligne (enregistrement) et de colonne(champ) ne doit contenir qu'une seule information.*

Les règles pour qu'une relation soit normalisée en 1FN sont:

- elle possède une clé identifiant de manière unique et stable
- aucun attribut n'est décomposable en plusieurs attributs significatifs → atomicité

L'atomicité d'une donnée peut être relative : adresse ou date → atomique ou pas?

*Pourquoi cette relation n'est-elle pas en 1FN?*

Nom Prenom	Date de naissance
John Doe	15.05.1968
Jean Dupond	22.04.1987
Georges Martin	04.06.2001

Id Personne	Nom	Prénom	Date de naissance
1	Doe	John	15.05.1968
2	Dupond	Jean	22.04.1987
3	Martin	Georges	04.06.2001

# 2<sup>ème</sup> forme normale (2FN)

---

Une relation est en 2FN si

- Elle est en 1FN
  - Tous les attributs n'appartenant pas à la clé sont en dépendance fonctionnelle élémentaire avec la clé
    - Si un attribut non-clé dépend d'une partie de la clé la relation n'est pas en 2FN
  - La 2FN n'est à vérifier que pour des relations ayant des clés composées
- 
- C'est la phase d'identification des clés
  - Evite certaines redondances



# 2<sup>ème</sup> forme normale (2FN)

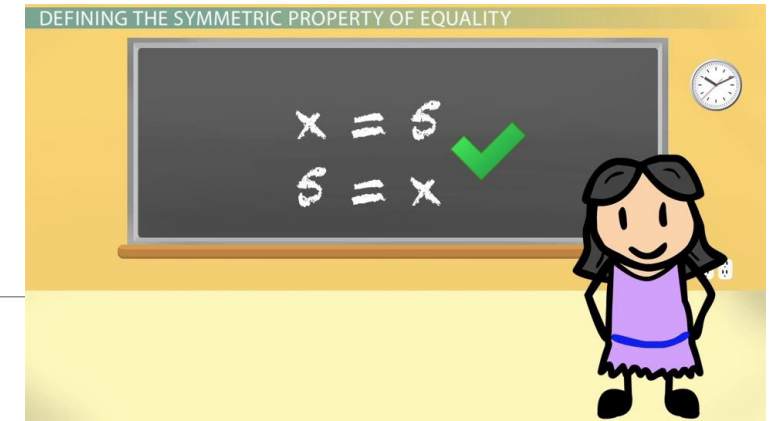
---

- Exemple avec les fournisseurs :
- Si l'on considère la relation (n° fournisseur, rue, NPA, ville, quantité, n° Produit, type produit) on est bien en 1FN (atomicité) mais pas en 2FN car des dépendances fonctionnelles ne dépendent que d'une partie de la clé :
- N° fournisseur → (rue, NPA, ville)
- N° produit → type produit
- N° fournisseur, n° produit → quantité
- Il faut donc décomposer en trois relations **Commande** (n°fournisseur, n°produit, date), **Produit** (n°produit, type produit) et **Fournisseur** (n° fournisseur, nom, prénom, rue, NPA, ville)
- **Chaque attribut qui ne fait pas partie de l'identifiant dépend d'un identifiant entier**

# 3<sup>ème</sup> forme normale (3FN)

Une relation est en 3FN si :

- Elle est en 2FN (ou 1 FN si clé simple)
- Tout attribut n'appartenant pas à la clé ne dépend pas d'un attribut non clé → tout attribut doit dépendre directement de la clé
- Liées à la transitivité des DF
- Un schéma en 3FN préserve les DF et n'entraîne pas de perte d'informations



# 3<sup>ème</sup> forme normale

---

Exemple:

On considère une relation CLIENT(n° client, Nom client , code categ, nom categ)

Pas en 3FN car :

- N° client  $\rightarrow$  nom categ n'est pas une DF
- La DF est client  $\rightarrow$  code categ  $\rightarrow$  nom categ



# En résumé

---

- Vérifier toujours que relation en 1FN puis 2FN puis 3FN dans cet ordre
- Si une des règles n'est pas respectée → modifier le schéma pour respecter ces 3 règles
- Il existe d'autre FN (par exemple la forme normale de Boyce Codd notée FNBC)
- Modèle normalisé = relations avec
  - une clé, qui permet de distinguer chaque occurrence
  - des **attributs élémentaires** (1FN)
  - en **dépendance de TOUTE la clé** (2FN)
  - et **RIEN QUE de la clé** (3FN)

# Liens

---

Quelques liens :

- [https://fr.wikipedia.org/wiki/Forme\\_normale\\_\(bases\\_de\\_donn%C3%A9es\\_relationnelles\)](https://fr.wikipedia.org/wiki/Forme_normale_(bases_de_donn%C3%A9es_relationnelles))
- <https://stph.scenari-community.org/bdd/nor1-lin/co/norUE00.html>