

# Conception des bases de données

## La normalisation

# Objectifs du cours

- Normalisation des relations
- Première forme normale
- Deuxième forme normale
- Troisième forme normale
- Boyce Codd forme normale

# Normalisation des relations

- Il existe différents modèles relationnels pour un même problème
- Parmi ces possibilités, quelles sont les plus adaptées afin d'éviter les problèmes de cohérence lors d'opérations de mise à jour ?
- Ces relations « bien faites » sont dites « normalisées »
- La normalisation consiste à éliminer les redondances des données
- Ils existes différentes formes de normalisation

# Besoin de normaliser

Détails de la facture									
Fact. N°.	Date	client	N° . Client	Adresse	N° inventaire	Produit	N° .Produit	No.	Prix
123	29/01/2018	John Public	11	35 Wood Lane, Springfield, ME 04487	1	Monitor	2-0023-D	10	\$200
123	29/01/2018	John Public	11	35 Wood Lane, Springfield, ME 04487	2	Tapis de souris	4-0023-D	12	50c
123	29/01/2018	John Public	11	35 Wood Lane, Springfield, ME 04487	3	Chaise	5-0023-D	1	\$120
124	30/01/2018	Jane Doe	12	72 Windy Road, Springfield, ME 04487	1	Ordinateur	1-0023-D	2	\$1,200
124	30/01/2018	Jane Doe	12	72 Windy Road, Springfield, ME 04487	2	Casque	3-0023-D	2	\$75

# Problème de mise à jour

- Anomalie de modification: Toute modification d'une adresse d'un client doit être répercutée sur toutes les autres lignes de la table facture  
→ Risques d'incohérence en cas de M.A.J

# Principe

- La théorie de la normalisation repose sur l'analyse des dépendances entre attributs.
- Elle propose des méthodes systématiques pour décomposer les relations incriminées.
- Il doit toujours être possible de reconstituer la relation originelle par jointure.
- Ces méthodes conduisent à des résultats "normaux".
- Attention à vouloir normaliser en toutes circonstances : dans certaines situations, on est amené à dénormaliser.

# Première forme normale

- Une relation est en première forme normale si et seulement si :
  - Toutes les colonnes ont une valeur atomique
  - Toutes les colonnes du tableau contiennent des valeurs identiques (appartiennent au même domaine)
  - Toutes les colonnes ont des valeurs constantes dans le temps

Client(numclient, nom, age, adresse, tel) →

Client(numclient, nom, date\_nais, adresse, code\_postal, ville, tel)

# Première forme normale :

## Le problème

Fact. N°.	Date	Client	N° Client	Adresse	N° inventaire	Produit	N°.Produit	No.	Prix
123	29/01/2018	John Public	11	35 Wood Lane, Springfield, ME 04487	1	Monitor	2-0023-D	10	\$200
123	29/01/2018	John Public	11	35 Wood Lane, Springfield, ME 04487	2	Tapis de souris	4-0023-D	12	50c
123	29/01/2018	John Public	11	35 Wood Lane, Springfield, ME 04487	3	Chaise	5-0023-D	1	\$120
124	30/01/2018	Jane Doe	12	72 Windy Road, Springfield, ME 04487	1	Ordinateur	1-0023-D	2	\$1,200
124	30/01/2018	Jane Doe	12	72 Windy Road, Springfield, ME 04487	2	Casque	3-0023-D	2	\$75



# Première forme normale :

## La solution

Fact. N°.	Date	Prénom	Nom	N°. Client	Adresse Postale	Ville	Etat	Code Postal	N° inventaire	Produit	N°.Produit	No.	Prix en €
123	01/29/2018	John	Public	11	35 Wood Lane	Springfield	ME	04487	1	Monitor	2-0023-D	10	200
123	01/29/2018	John	Public	11	35 Wood Lane	Springfield	ME	04487	2	Tapis de souris	4-0023-D	12	0.50
123	01/29/2018	John	Public	11	35 Wood Lane	Springfield	ME	04487	3	Chaise	5-0023-D	1	120
124	01/30/2018	Jane	Doe	12	72 Windy Road	Springfield	ME	04487	1	Ordinateur	1-0023-D	2	1200
124	01/30/2018	Jane	Doe	12	72 Windy Road	Springfield	ME	04487	2	Casque	3-0023-D	2	75

## Deuxième forme normale

- Une relation est en deuxième forme normale si et seulement si :
  - elle est en première forme normale
  - tous les attributs non-clés ne dépendent pas d'une partie de la clé primaire mais bien de la totalité de la clé primaire

Commande (Numcli, CodeArticle, Date, QtéCommandée, Désignation)

→

Commande (Numcli, CodeArticle, Date, QtéCommandée)

Article(CodeArticle, Désignation)

# Deuxième forme normale

## Définir la clé primaire

<u>Fact. N°.</u>	<u>N° inventaire</u>	Date	Prénom	Nom	N° Client	Adresse Postale	Ville	Etat	Code Postal	Produit	N° Produit	No.	Prix en €
123	1	29/01/2018	John	Public	11	35 Wood Lane	Springfield	ME	04487	Monitor	2-0023-D	10	200
123	2	29/01/2018	John	Public	11	35 Wood Lane	Springfield	ME	04487	Tapis de souris	4-0023-D	12	0.50
123	3	29/01/2018	John	Public	11	35 Wood Lane	Springfield	ME	04487	Chaise	5-0023-D	1	120
124	1	30/01/2018	Jane	Doe	12	72 Windy Road	Springfield	ME	04487	Ordinateur	1-0023-D	2	1200
124	2	30/01/2018	Jane	Doe	12	72 Windy Road	Springfield	ME	04487	Casque	3-0023-D	2	75

# Deuxième forme normale

facture								
<u>Fact. N°.</u>	Date	Prénom	Nom	N° Client	Adresse Postale	ville	Etat	Code postal
123	29/01/2018	John	Public	11	35 Wood Lane	Springfield	ME	04487
124	30/01/2018	Jane	Doe	12	72 Windy Road	Springfield	ME	04487

Poste de facture					
<u>Fact. N°.</u>	<u>N°. inventaire</u>	Produit	N° Produit	No.	Prix en €
123	1	Monitor	2-0023-D	10	200
123	2	Tapis de souris	4-0023-D	12	0.50
123	3	Chaise	5-0023-D	1	120
124	1	Ordinateur	1-0023-D	2	1200
124	2	Casque	3-0023-D	2	75

# Troisième forme normale

- Une relation est en troisième forme normale si et seulement si :
  - elle est en deuxième forme normale
  - tous les attributs n'appartenant pas à la clé ne dépendent pas d'un attribut non-clé (la dépendance fonctionnelle est indirecte)

Commande(NuméroCommande, #CodeClient, Nom client, #RefArticle)

→

Commande(NuméroCommande, #CodeClient, #RefArticle)

Clients(CodeClient, Nom client)

# Troisième forme normale

## Le problème

Facture								
<u>Fact. N°</u>	Date	Prénom	Nom	N° Client	Adresse Postale	Ville	Etat	Code postal
123	29/01/2018	John	Public	11	35 Wood Lane	Springfield	ME	04487
124	29/01/2018	Jane	Doe	12	72 Windy Road	Springfield	ME	04487

Poste de Facture					
<u>Fact. N°</u>	<u>N° inventaire</u>	Produit	N° Produit	No.	Prix en €
123	1	Monitor	2-0023-D	10	200
123	2	Mousepad	4-0023-D	12	0.50
123	3	Chair	5-0023-D	1	120
124	1	Laptop	1-0023-D	2	1200
124	2	Headset	3-0023-D	2	75

# Troisième forme normale

Facture		
<u>Fact.</u> N°	Date	N° Client
123	01/29/2018	11
124	01/30/2018	12

Produits		
<u>N° Produit</u>	Produit	Prix en €
1-0023-D	Ordinateur	1200
2-0023-D	Monitor	200
3-0023-D	Casque	75
4-0023-D	Tapis de souris	0.50
5-0023-D	Chaise	120

Poste de facture			
<u>Fact.</u> N°	<u>N°</u> Inventaire	N° Produit	No.
123	1	2-0023-D	10
123	2	4-0023-D	12
123	3	5-0023-D	1
124	1	1-0023-D	2
124	2	3-0023-D	2

Client						
<u>N°</u> Client	Prénom	Nom	Adresse Postale	Ville	Etat	Code Postal
11	John	Public	35 Wood Lane	Springfield	ME	04487
12	Jane	Doe	72 Windy Road	Springfield	ME	04487

# Boyce Codd forme normale

- La Boyce Codd forme normale est un resserrement de la 3<sup>ème</sup> forme normale :
  - elle est en troisième forme normale
  - Il n'existe aucun attribut faisant partie de la clé qui dépend d'un attribut ne faisant pas partie de la clé



# Boyce Codd forme normale

## Le problème

Clé primaire : Vendeur + N° Produit

Nombre de produits par Vendeur			
N° V.	Vendeur	N° Produit	Nombre
V-012	Example Suppliers, Inc.	1-0023-D	900
V-012	Example Suppliers, Inc.	2-0023-D	250
V-012	Example Suppliers, Inc.	3-0023-D	395
V-077	Jones & Sons Equipment	4-0023-D	1275
V-077	Jones & Sons Equipment	5-0023-D	12000

# Troisième forme normale

## La solution

Vendeurs	
<u>N° V</u>	Vendeur
V-012	Example Suppliers, Inc.
V-077	Jones & Sons Equipment

Nombre		
<u>N° V</u>	<u>N° Produit</u>	Nombre
V-012	1-0023-D	900
V-012	2-0023-D	250
V-012	3-0023-D	395
V-077	4-0023-D	1275
V-077	5-0023-D	12000

## 5) principes pour une BDDR performante:

- 1) **AUCUNE REDONDANCE** : sinon cela augmente le volume de la base et multiplie les mises à jour par autant de données redondées, donc augmente la durée du traitement, donc dégrade les performances.
- 2) **PAS D'ANOMALIE TRANSACTIONNELLE** : la modification d'une valeur sémantique doit se traduire par la modification d'une seule donnée dans une seule table. Sinon il faudra mettre à jour de nombreuses lignes et c'est pénalisant : durée de la requête plus longue.
- 3) **LES DONNÉES DOIVENT ÊTRE ATOMIQUES** : une colonne de table ne doit contenir qu'une seule information. En revanche, une vue peut contenir des données concaténées, agrégées, calculées... Une donnée non atomique se traduit par des requêtes alambiquées (SUBSTRING par exemple), OU, multiples critères dans la clause WHERE et souvent par l'incapacité à utiliser les index. En plus cela empêche une évolution aisée de la structure de la base.

**4) LES COLONNES D'UNE TABLE DOIVENT CORRESPONDRE À DES ATTRIBUTS PROPRE À L'ENTITÉ QUE LA TABLE MODÉLISE :** un numéro de téléphone, un mail ou une adresse ne sont pas des attributs propre à une personne, mais relatif à une table de téléphone, de mail ou d'adresse associé à la personne. En créant des tables artificiellement obèses (nombreuses colonnes), les requêtes sont plus complexes, l'accès aux données se fait la plupart du temps par balayage de la table (aucune utilisation d'index), la durée de verrouillage est très notablement allongée, car c'est toujours la même table qui est impactée par les mises à jour.

**5) LES CLEFS PRIMAIRE DES TABLES DOIVENT ÊTRE CONCISES, ASSÉMANTIQUE ET INVARIABLE** : la meilleure clef primaire est un auto-incrément de type INT ou BIGINT. Car courte (4 ou 8 octets, c'est-à-dire la longueur du mot en mémoire en 32 ou 64 bits), assémantique (aucune signification de la données dont la valeur est arbitraire) et invariables (la clef ne sera jamais mise à jour). Avec une telle clef, les FOREIGN KEY sont correctement indexées et la base de données sera performante peu importe le nombre de jointures, car ce n'est pas la jointure qui pose des problèmes de performance, mais l'encombrement de la RAM avec des tables obèses, et l'impossibilité d'exploiter les index avec des tables ne respectant pas ces principes de modélisation...