



# Formes normales pour les bases de données relationnelles

L. Nerima  
Université de Genève

avril 2020



# Bibliographie

- ❖ Elmasri, Ramez & Navathe, Shamkant B.  
Fundamentals of Database Systems, 7<sup>th</sup> ed. Pearson, 2017. (Chapitre 15, Functional Dependencies and Normalization for Relational Databases )
  - *Disponible à la bibliothèque de Linguistique*
  - *PDF de la 6<sup>th</sup> ed. disponible sur Chamilo, Documents/Documentation*
- ❖ Steven Roman  
Access Database, Design & Programming. O'Reilly 1997 (Chapitre 4, Database Design Principles)
- ❖ Ullman, Jeffrey & Widom, Jennifer  
A First Course in Database Systems, 3rd ed. Pearson, 2014. (Chapitre 3, p. 63, Design Theory for Relational Databases)

# Plan

- ❖ Rappel conception statique
- ❖ Rappel anomalies de mise à jour
- ❖ Intérêt pour nous
- ❖ Discussion sur un exemple
- ❖ Dépendances fonctionnelles
- ❖ Formes normales
- ❖ 1ère forme normale 1FN
- ❖ 2ème forme normale 2FN
- ❖ 3ème forme normale 3FN
- ❖ Forme normale de Boyce-Codd FNBC
- ❖ Normalisation
- ❖ Décomposition

# Rappel conception statique

- ❖ Conception de la structure statique avec le modèles des classes UML
  - ❖ produire le schéma qui répond aux besoins des cas d'utilisation
  - ❖ produire le schéma le plus significatif possible
  - ❖ éviter la redondance des informations
- ❖ Règles de modélisation des classes
  - ❖ règles de mise sous forme canonique
  - ❖ 1) éliminer les synonymes
  - ❖ 2) éliminer les associations redondantes
  - ❖ 3) mise en évidence des attributs dérivables
  - ❖ 4) éclatement de classes
  - ❖ 5) transformer une classe en une structure de généralisation
  - ❖ ... éventuellement d'autres règles

# Rappel anomalies de mise à jour

- ❖ ➔ 00\_BDintro, Anomalies de mise à jour (p. 20)
- ❖ se produisent lorsqu'il y a des données redondantes
- ❖ Anomalies d'insertion
  - ❖ lors de l'insertion d'un tuple, il faut vérifier que les valeurs d'attributs insérées soient cohérentes pour les attributs où il y a de la redondance
- ❖ Anomalies de suppression
  - ❖ lors de la suppression d'un tuple on peut aussi supprimer d'autres données involontairement
- ❖ Anomalies de modification
  - ❖ lors de la modification d'une valeur d'un attribut où il y a de la redondance il faut s'assurer que toutes les valeurs redondantes soient modifiées de la même manière

# Intérêt pour nous

- ❖ Les outils que nous verrons dans ce chapitre sont
- ❖ ... inutiles si les relations ont été obtenue à partir d'un schéma de classes "bien fait" (vérifié au sens des règles de mise sous forme canonique) :-)
  - ➔ cours Modélisation des SI et des services
- ❖ ... mais très utiles
  - ❖ pour aider dans un choix de modélisation (vérification des conséquences du choix)
  - ❖ pour analyser / valider la qualité du schéma d'une base de données relationnelle existante (rétro conception)

# Discussion sur un exemple

## ❖ Reprenons l'exemple des notices d'articles de journaux

### Editeur

<i>nom éditeur</i>	<i>adresse éditeur</i>
NAC	Genève
Ringier	Lausanne
Edipresse	Lausanne
Tamedia	Zurich

### Auteur

<i>nom auteur</i>	<i>adresse auteur</i>
G. Aublanc	Genève
P. Nusslé	Genève
A. Dayer	Lausanne
G. Moeckli	Carouge

### Journaux

<i>nom journal</i>	<i>rédacteur en chef</i>	<i>nom éditeur</i>
Le Courrier	L. Drompt	NAC
L'Illustré	M. Jeanneret	Ringier
La Tribune de Genève	F. Juliard	Tamedia
Le temps stratégique	C. Monnier	Edipresse

### Article

<i>titre article</i>	<i>nom aut.</i>	<i>nom journal</i>	<i>n°journal</i>
L'ordinateur va-t'il imposer sa manière de penser	G. Moeckli	Le temps stratégique	10
L'éthique scientifique au cœur des plaidoiries	C. Aublanc	Le Courrier	234
Alinghi brise l'invicibilité d'Oracle	P. Nusslé	La Tribune de Genève	237-41
Rat des champs contre rats du loft	J.D. Urbain	L'Illustré	40
Lettre à une grande dame	A. Dayer	L'Illustré	40

# Discussion sur un exemple (suite)

## ❖ Autre solution

### Editeur

<i>nom éditeur</i>	<i>adresse éditeur</i>
NAC	Genève
Ringier	Lausanne
Edipresse	Lausanne
Tamedia	Zurich

### Auteur

<i>nom auteur</i>	<i>adresse auteur</i>
G. Aublanc	Genève
P. Nusslé	Genève
A. Dayer	Lausanne
G. Moeckli	Carouge

### Article\_journaux

<i>titre article</i>	<i>nom auteur</i>	<i>nom journal</i>	<i>n° journal</i>	<i>rédacteur</i>	<i>nom éditeur</i>
L'ordinateur va-t'il imposer...	Moeckli	Le temps...	10	C. Monnier	Edipresse
L'éthique scientifique...	Aublanc	Le Courrier	234	L. Drompt	NAC
Alinghi brise l'invicibilité...	P. Nusslé	La Tribune	237-41	F. Juliard	Tamedia
Rat des champs contre rats...	D. Urbain	L'Illustré	40	M. Jeanneret	Ringier
Lettre à une grande dame	A. Dayer	L'Illustré	40	M. Jeanneret	Ringier

## ❖ Commentaires ...

## ❖ Par conséquent, laquelle de ces deux solutions est préférable ?



# Schémas de relation: les bonnes propriétés

On aimerait avoir des schémas de relation

1. qui sont significatifs, c-à-d dont le sens des attributs est facile à comprendre (ou à expliquer)
  2. qui ne font pas apparaître des anomalies de mise à jour lors des insertions, suppressions et modifications
  3. qui ne contiennent pas, dans la mesure du possible, des attributs susceptibles de prendre la valeur nulle
  4. dont la jointure se fait avec une condition d'égalité (équijointure) sur les attributs clé ou clés étrangères
- ❖ Commentaire à propos des valeurs nulles -> plusieurs variantes possibles:  
"ne s'applique pas", "valeur inconnue", "valeur connue mais absente"

# Dépendances fonctionnelles

- ❖ C'est le concept le plus important dans la conception de schémas relationnels
- ❖ Se définit de manière formelle
- ❖ Définition

"Une dépendance fonctionnelle est une contrainte entre deux ensembles d'attributs de la base de données. Soit  $A_1, A_2, \dots, A_n$  les  $n$  attributs du schéma de la bd. Supposons que la bd ait été décrite par une seule relation **universelle**  $R = \{ A_1, A_2, \dots, A_n \}$ . Une **dépendance fonctionnelle**, notée  $X \rightarrow Y$ , entre deux ensembles d'attributs  $X$  et  $Y$  de  $R$  spécifie une contrainte sur les tuples possibles qui peuvent former une instance  $r$  de  $R$ . La contrainte dit que pour tout tuple  $t_1$  et  $t_2$  dans  $r$  t.q.  $t_1[X] = t_2[X]$  nous devons aussi avoir  $t_1[Y] = t_2[Y]$ "

Notation:  $t[X]$  dénote les valeurs des attributs de  $X$  du tuple  $t$

# Dépendances fonctionnelles (commentaires)

- ❖ Une dépendance fonctionnelle  $X \rightarrow Y$  indique que les valeurs de la composante Y d'un tuple t dépendent - ou sont déterminées par - les valeurs de la composante X
- ❖ ou encore (exprimé de manière duale) que les valeurs de la composante X d'un tuple déterminent de manière **unique** (ou **fonctionnelle**) les valeurs de la composante Y.
- ❖ Abréviation: DF
  
- ❖ Pourquoi le terme « fonctionnelle » ?  
Rappel: « en mathématiques, une **fonction** relie deux grandeurs (à priori numériques) de telles façon que la connaissance de la première permet de déterminer la deuxième » *Wikipédia*

# Dépendances fonctionnelles et clé

- ❖ Dans le chapitre sur le modèle relationnel, nous avons défini la clé de la manière suivante :  
« *La clé d'une relation est un ensemble d'attributs minimum dont la connaissance permet d'identifier un tuple unique de la relation considérée.* »
- ❖ Nous pouvons aussi définir la clé d'une relation avec la notion de dépendance fonctionnelle :  
« *Un ensemble d'un ou de plusieurs attributs  $\{ A_1, A_2, \dots A_n \}$  est la clé d'une relation si :*
  1. *Cet ensemble d'attributs détermine fonctionnellement tous les autres attributs de la relation;*
  2. *il n'existe pas de sous-ensemble strict de  $\{ A_1, A_2, \dots A_n \}$  qui détermine fonctionnellement tous les autres attributs de la relation (minimalité de la clé).* »

# Dépendances fonctionnelles (exercice)

## ❖ Exercice:

- ❖ 1) trouver toutes les DF entre attributs des notices d'articles de journaux. Les attributs sont:  
*nom éditeur, adresse éditeur, nom auteur, adresse auteur, nom journal, rédacteur en chef, titre article, n° journal*
- ❖ 2) chercher les DF d'un exemple de votre choix  
P.ex. le domaine d'application de votre projet de bases de données

# Dépendances fonctionnelles (solution exercice)

- ❖ 1) trouver toutes les DF entre attributs des notices de journaux
  - ❖ *nom éditeur*  $\rightarrow$  *adresse éditeur*
  - ❖ *nom auteur*  $\rightarrow$  *adresse auteur*
  - ❖ *nom journal*  $\rightarrow$  *rédacteur en chef* \*
  - ❖ *nom journal*  $\rightarrow$  *nom éditeur*
  - ❖ *titre article*  $\rightarrow$  *nom auteur*
  
- ❖ \* Dans une optique historique on aurait:  
 $\{ \textit{nom journal}, \textit{n}^{\circ} \textit{journal} \} \rightarrow \textit{rédacteur en chef}$

# Formes normales

- ❖ Formes (ou contraintes) sur les schémas de relation
- ❖ Possèdent de « bonnes » propriétés (élimination des anomalies de mise à jour)
- ❖ Nous allons étudier 4 formes normales
  - ❖ 1FN
  - ❖ 2FN
  - ❖ 3FN
  - ❖ FNBC
- ❖ Chaque forme est plus forte que la précédente
  - ❖ p.e. un schéma de relation en 3FN est aussi en 2FN et en 1FN
- ❖ 2FN, 3FN et FNBC sont définies à l'aides des DF

# Formes normales et normalisation

- ❖ On appelle **normalisation** le processus de transformation d'un schéma pour l'amener dans une forme normale de degré supérieur
- ❖ Il est généralement souhaitable d'avoir un haut degré de normalisation (p.e. FNBC)
  - ❖ mais c'est parfois très (trop) compliqué !
- ❖ Problème: perte d'informations après normalisation



# 1FN

- ❖ La 1FN est très simple
- ❖ Définition: « Un schéma de relation est en 1FN si les domaines des attributs contiennent des valeurs atomiques (indivisibles) »
- ❖ Valeur atomique: « bases de données », « 17 », « frs 12.50 »
- ❖ Valeur non atomique (ou structurée)
  - ❖ Multivaluée: « Henri, André, Julien », « info1, BD, outils formels, ntic »
  - ❖ Composée: «5, av. Soret, 1203 Genève», «TALN, Wehrli, A 112», «1 mai 06»
- ❖ Quels sont les attributs qui contiennent des valeurs non-atomiques dans la relation ci-dessous ?

<u>ISBN</u>	Titre	Auteurs	Editeur	Année
0-8053-0145-3	Fundamentals of database systems	Elmasri R. & Navathe S.	Benjamin	1989
1-56592-297-2	Access Database	Roman S.	O'Reilly	1997

# 1FN (suite)

- ❖ Problèmes avec les valeurs non atomiques
  - ❖ Interrogation, p.e. « tous les livres écrits par Navathe S. »
  - ❖ Relation « imbriquée », p.e. t[cours]= «TALN, Wehrli, A 112»
- ❖ Atomique <-> non atomique
  - ❖ Dépend du contexte de l'application
  - ❖ Sémantique des attributs
  - ❖ Exemple: attribut *Adresse* (ex. val. « 5, av. Soret, 1203 Genève »)
    - ❖ La question: le n° de la rue, le nom de la rue, le n° postal et la ville doivent-ils être séparé dans des attributs séparés ?
    - ❖ 1) oui, dans le contexte ...
    - ❖ 2) non, dans le contexte ...

# 1FN exemple de normalisation

## ❖ Exemple de normalisation

<u>ISBN</u>	Titre	Auteurs	Editeur	Année
0-8053-0145-3	Fundamentals of database systems	Elmasri R. & Navathe S.	Benjamin	1989
1-56592-297-2	Access Database	Roman S.	O'Reilly	1997



# 1FN exemple de normalisation

## ❖ Exemple de normalisation

<u>ISBN</u>	<i>Titre</i>	<i>Auteurs</i>	<i>Editeur</i>	<i>Année</i>
0-8053-0145-3	Fundamentals of database systems	Elmasri R. & Navathe S.	Benjamin	1989
1-56592-297-2	Access Database	Roman S.	O'Reilly	1997



### Livre

<u>ISBN</u>	<i>Titre</i>	<i>Editeur</i>	<i>Année</i>
0-8053-0145-3	Fundamentals of database systems	Benjamin	1989
1-56592-297-2	Access Database	O'Reilly	1997

### AuteurDeLivre

<u>ISBN</u>	<u>Auteur</u>
0-8053-0145-3	Elmasri R.
0-8053-0145-3	Navathe S.
1-56592-297-2	Roman S.

## 2FN

- ❖ La 2FN est basée sur le concept de DF totale (ou complète)
- ❖ Une DF  $X \rightarrow Y$  est une DF totale si la suppression de n'importe quel attribut A de X invalide la DF (en dépendance partielle sinon)
- ❖ Définition: « Un schéma de relation en 1FN est en 2FN si chaque attribut A ne-faisant-pas-partie-de-la-clé de R est en dépendance fonctionnelle totale de la clé de R »
- ❖ Autre définition: « ... si aucun attribut ne-faisant-pas-partie-de-la-clé de R est en dépendance fonctionnelle partielle de la clé de R »
- ❖ Exemple: la relation ci-dessous n'est pas en 2FN car il y a une DF  $Auteur \rightarrow Adresse\ auteur$ . Conséquence: redondance de la valeur en rouge

<u>Titre</u>	<u>Auteur</u>	Adresse auteur	Journal	N°
L'ordinateur va-t'il imposer...	G. Moeckli	Carouge	Le temps stratégique	10
Frankenstein est genevois !	G. Moeckli	Carouge	Le journal de Genève	1997

# 3FN

- ❖ La 3FN est basée sur le concept de DF transitive
- ❖ Une DF  $X \rightarrow Y$  est une DF transitive s'il existe un ensemble d'attributs  $Z$  qui n'est sous-ensemble d'aucune clé de  $R$  et que  $X \rightarrow Z$  et  $Z \rightarrow Y$
- ❖ Définition: « Un schéma de relation en 2FN est en 3FN si aucun attribut ne-faisant-pas-partie-de-la-clé de  $R$  n'est transitivement dépendant de la clé »
- ❖ En d'autres termes, « ... s'il n'existe pas de DF entre les attributs ne-faisant-pas-partie-de-la-clé »
- ❖ Exemple: Supposons que les éditeurs détermine le prix des livres en fonctions du nombre de pages (c-à-d qu'il y a une DF *Editeur*, *Nb pages*  $\rightarrow$  *Prix*), alors la relation ci-dessous n'est pas en 3FN

<u>ISBN</u>	Titre	Editeur	Nb pages	Prix
1-73542-035-1	Practical RDF	O'Reilly	250	CHF 52.-
1-56592-297-2	Access Database	O'Reilly	250	CHF 52.-

# FNBC - Forme normale de Boyce-Codd

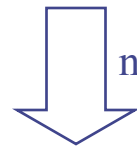
- ❖ Définition: « Un schéma de relation en 3FN est en FNBC si aucun attribut de R n'est transitivement dépendant de la clé »
- ❖ Commentaire: cela revient à relâcher dans la définition de la 3FN la condition que l'attribut transitivement dépendant ne doit pas faire partie de la clé
- ❖ Exemple: dans la relation ci-dessous, on peut retrouver la ville à partir du code postal, c-à-d qu'il y a une DF *Code postal* → *Ville*. La relation (qui est en 3FN) n'est donc pas en FNBC

<u>Ville</u>	<u>Rue</u>	<i>Code postal</i>
Genève	Soret	1203
Genève	Châtelain	1203

# FNBC exemple de normalisation

- ❖ Nous savons qu'il y a 2 DF
  - ❖  $\{Ville, Rue\} \rightarrow code\ postal$
  - ❖  $code\ postal \rightarrow Ville$

<u>Ville</u>	<u>Rue</u>	<i>Code postal</i>
Genève	Soret	1203
Genève	Châtelain	1203



normalisation FNBC

<u>Code postal</u>	<i>Ville</i>
1203	Genève

<u>Code postal</u>	<u>Rue</u>
1203	Soret
1203	Châtelain

- ❖ La normalisation en FNBC a éliminé la redondance !



# Normalisation

- ❖ Le processus de transformation d'un schéma pour l'amener dans une forme normale de degré supérieur s'appelle **normalisation**
- ❖ Il est généralement souhaitable d'avoir un haut degré de normalisation (p.e. FNBC)
  - ❖ mais c'est parfois très (trop) compliqué !
- ❖ Nous avons vu deux exemples de normalisation
  - ❖ → 1FN, *livre(ISBN, titre, auteurs, éditeur, année)* p. 325
  - ❖ → FNBC, *adresse(ville, rue, code postal)* p. 329
- ❖ Dans les deux cas, la normalisation a consisté à scinder la relation en 2
  - ❖ On parle de **décomposition** de relation

# Décomposition

- ❖ La décomposition d'un schéma de relation en schémas plus petits (et si possible normalisés) est souhaitable du point de vue de l'élimination de la redondance
- ❖ ... mais peut entraîner deux problèmes:
  - ❖ Perte d'information
  - ❖ Perte de dépendances fonctionnelles

# Perte d'information - exemple

- ❖ Soit le schéma de relation (*idAuteur*, *nomAuteur*, *idEditeur*) et la df *idAuteur*  $\rightarrow$  *nomAuteur*
- ❖ Décomposition en deux schémas de relation
  - ❖ (*idAuteur*, *nomAuteur*)
  - ❖ (*nomAuteur*, *nomEditeur*)
- ❖ Considérons la relation *r*

<i>idAuteur</i>	<i>nomAuteur</i>	<i>idEditeur</i>
a1	Jacques Guyot	e1
a2	Jacques Guyot	e2

- ❖ et les deux relations *q* et *s* issues de la décomposition

<i>idAuteur</i>	<i>nomAuteur</i>
a1	Jacques Guyot
a2	Jacques Guyot

<i>nomAuteur</i>	<i>idEditeur</i>
Jacques Guyot	e1
Jacques Guyot	e2

- ❖ Est-ce qu'il y a eu perte d'information ?

# Perte d'information (suite)

- ❖ Après la jointure de q et s, on obtient la relation

<i>idAuteur</i>	<i>nomAuteur</i>	<i>idEditeur</i>
a1	Jacques Guyot	e1
a1	Jacques Guyot	e2
a2	Jacques Guyot	e1
a2	Jacques Guyot	e2

- ❖ La relation obtenue est plus grande que la relation d'origine
- ❖ ... mais en fait on a perdu de l'information car on ne plus retrouver le fait que chacun des deux auteurs a publié chez un éditeur différent !

# Perte de dépendances fonctionnelles

- ❖ Soit le schéma de relation (*ISBN*, *nbPages*, *prix*)
- ❖ et les df
  - ❖  $ISBN \rightarrow nbPages$
  - ❖  $nbPages \rightarrow prix$
- ❖ Considérons la relation *t*

<i>ISBN</i>	<i>nbPages</i>	<i>prix</i>
0-8053-0145-3	250	CHF 52.-
1-56592-297-2	250	CHF 52.-

- ❖ et les deux relations *u* et *v* issues de la décomposition

<i>ISBN</i>	<i>nbPages</i>
0-8053-0145-3	250
1-56592-297-2	250

<i>ISBN</i>	<i>prix</i>
0-8053-0145-3	CHF 52.-
1-56592-297-2	CHF 52.-

- ❖ La décomposition a eu pour conséquence la perte de la df  $nbPages \rightarrow prix$ , les deux attributs *nbPages* et *prix* n'étant plus dans la même table

# Perte de dépendances fonctionnelles (suite)

- ❖ La relation  $v$  n'implémente plus la df  $nbPages \rightarrow prix$  et il est donc possible, par exemple, de changer le prix du deuxième livre

<i>ISBN</i>	<i>prix</i>
0-8053-0145-3	CHF 52.-
1-56592-297-2	CHF 32.-

- ❖ En joignant  $u$  avec  $v'$  on obtient

<i>ISBN</i>	<i>nbPages</i>	<i>prix</i>
0-8053-0145-3	250	CHF 52.-
1-56592-297-2	250	CHF 32.-

- ❖ qui viole la df  $nbPages \rightarrow prix$  !

# Exemple de décomposition sans perte de df

- ❖ Soit la décomposition de t

<i>ISBN</i>	<i>nbPages</i>	<i>prix</i>
0-8053-0145-3	250	CHF 52.-
1-56592-297-2	250	CHF 52.-

- ❖ en

<i>ISBN</i>	<i>nbPages</i>
0-8053-0145-3	250
1-56592-297-2	250

<i>nbPages</i>	<i>prix</i>
250	CHF 52.-

- ❖ Ici la décomposition préserve les df et on peut mettre à jour sans risque chacune des deux relations indépendamment (la jointure des deux relations donne exactement la relation d'origine)

# Décomposition - conclusions

- ❖ Si la décomposition n'occasionne pas de perte d'information, elle est appelée décomposition sans perte d'information ou décomposition joignable (*lossless decomposition*)
- ❖ Si la décomposition n'occasionne pas de perte de dépendance fonctionnelle, elle est appelée décomposition préservant les df (*dependency-preserving decomposition*)
- ❖ Il est possible de montrer que toute relation peut-être décomposée en une collection de relations plus petites en FNBC...
- ❖ ... mais il n'est pas possible de garantir que cette décomposition préserve les df
- ❖ Par contre il est toujours possible de décomposer un schéma de relation en 3FN sans perte d'information et préservant les df
- ❖ Il existe
  - ❖ Des algorithmes de décomposition sans perte d'information (mais qui conduisent parfois à des schémas peu intuitifs)
  - ❖ Des tests de préservation des df
- ❖ Steven Roman dit: « une bonne conception de base de données relationnelle requiert huit parts d'intuition et deux parts de théorie »