

Contexte

Le but de ce tutoriel est de pratiquer la normalisation des bases de données, de la forme non normalisée (UNF) jusqu'à la 3NF. Ce tutoriel détaille chaque étape du processus de normalisation, vue lors de la semaine 3, en fournissant des instructions et explications pour vous familiariser avec le processus. L'objectif est d'apprendre à identifier et éliminer les dépendances fonctionnelles et les dépendances transitives afin d'obtenir des relations bien structurées.

1NF et 2NF

a) Pourquoi la table n'est pas en 1NF

La table n'est pas en 1NF car elle contient des cellules avec plusieurs valeurs. Par exemple, dans la colonne NO_EMPLOYE, certaines cellules contiennent plusieurs numéros d'employés, ce qui va à l'encontre du principe de la 1NF qui stipule qu'une cellule doit contenir une seule valeur atomique.

b) Les 6 attributs nécessaires pour ce cas

Les attributs nécessaires à cette table sont :

1. NO_DEPARTEMENT
2. NOM_DEP
3. NO_EMPLOYE
4. POSTE_EMP
5. NOM_EMP
6. TAUX_HORAIRE

Ces attributs couvrent toutes les informations nécessaires pour représenter les départements, les employés et leurs postes dans cette table.

c) Normalisation en 1NF

i) Mise à plat de la table

Pour normaliser la table en 1NF, nous devons répéter les informations manquantes et s'assurer qu'il n'y a qu'une seule valeur par cellule. La table devient :

NO_DEPARTEMENT	NOM_DEP	NO_EMPLOYE	POSTE_EMP	NOM_EMP	TAUX_HORAIRE
Q22	Quincaillerie	12564	Assistant	Louis-Thomas Vachon	12
Q22	Quincaillerie	12566	Commis	Julie Tremblay	10
E11	Électroménagers	13255	Commis	Jules Girard	12
E11	Électroménagers	12566	Commis	Julie Tremblay	11
E11	Électroménagers	32155	Gérante	Suzie Dupain	14

ii) Les 5 dépendances fonctionnelles de la table

Voici les 5 dépendances fonctionnelles de cette table :

- **df1** : NO_DEPARTEMENT → NOM_DEP
 - Le numéro de département (NO_DEPARTEMENT) détermine le nom du département (NOM_DEP).
- **df2** : NO_EMPLOYE → NOM_EMP, POSTE_EMP, TAUX_HORAIRE
 - Le numéro de l'employé (NO_EMPLOYE) détermine le nom de l'employé (NOM_EMP), son poste (POSTE_EMP), et son taux horaire (TAUX_HORAIRE).
- **df3** : (NO_DEPARTEMENT, NO_EMPLOYE) → NOM_DEP, POSTE_EMP, NOM_EMP, TAUX_HORAIRE
 - La combinaison du numéro de département et du numéro d'employé détermine les autres attributs de la table.
- **df4** : NO_EMPLOYE, NOM_EMP → TAUX_HORAIRE
 - La combinaison du numéro et du nom de l'employé détermine le taux horaire de l'employé.
- **df5** : NOM_DEP → NO_DEPARTEMENT
 - Le nom du département (NOM_DEP) détermine le numéro du département (NO_DEPARTEMENT).

iii) Clé primaire

La clé primaire de la table est la combinaison (NO_DEPARTEMENT, NO_EMPLOYE), car cette combinaison permet d'identifier de manière unique chaque ligne de la table.

d) Pourquoi la table obtenue n'est pas en 2NF

Dans cette table, les attributs POSTE_EMP, NOM_EMP, et TAUX_HORAIRE sont dépendants uniquement de NO_EMPLOYE et non de la combinaison (NO_DEPARTEMENT, NO_EMPLOYE). Cela signifie qu'il y a une **dépendance partielle**,

ce qui viole la règle de la 2NF. Nous devons donc encore normaliser cette table pour supprimer ces dépendances partielles.

e) Il faut normaliser cette table en 2NF en y identifiant les dépendances partielles par rapport à la clé primaire

i) Identification des dépendances partielles

Parmi les cinq dépendances fonctionnelles trouvées précédemment, les deux dépendances partielles par rapport à la clé primaire (NO_DEPARTEMENT, NO_EMPLOYE) sont :

- **df1** : NO_DEPARTEMENT → NOM_DEP
 - Le nom du département dépend uniquement du numéro de département (NO_DEPARTEMENT), ce qui constitue une dépendance partielle.
- **df2** : NO_EMPLOYE → NOM_EMP, TAUX_HORAIRE
 - Le nom de l'employé et son taux horaire dépendent uniquement du numéro d'employé (NO_EMPLOYE), ce qui constitue également une dépendance partielle.

ii) Création des nouvelles tables

Pour normaliser la table en 2NF, nous devons extraire ces dépendances partielles et créer de nouvelles tables. Cela implique de déplacer les attributs dépendants (NOM_DEP pour NO_DEPARTEMENT, et NOM_EMP, TAUX_HORAIRE pour NO_EMPLOYE) dans des tables distinctes, tout en conservant les déterminants (NO_DEPARTEMENT et NO_EMPLOYE) dans ces nouvelles relations.

Nous obtenons alors trois tables avec les clés primaires respectives suivantes :

- (NO_DEPARTEMENT, NO_EMPLOYE) pour la relation d'affectation
- NO_DEPARTEMENT pour la relation des départements
- NO_EMPLOYE pour la relation des employés

Voici les tables avec leur contenu :

Table 1 : DEPARTEMENT

NO_DEPARTEMENT	NOM_DEP
Q22	Quincaillerie
E11	Électroménagers

Dans cette table, la clé primaire est NO_DEPARTEMENT, et NOM_DEP est un attribut dépendant de cette clé.

Table 2 : EMPLOYE

NO_EMPLOYE	NOM_EMP	TAUX_HORAIRE
12564	Louis-Thomas Vachon	12
12566	Julie Tremblay	10
13255	Jules Girard	12
12566	Julie Tremblay	11
32155	Suzie Dupain	14

Ici, la clé primaire est NO_EMPLOYE, et NOM_EMP ainsi que TAUX_HORAIRE sont des attributs dépendant de cette clé.

Table 3 : AFFECCATION

NO_DEPARTEMENT	NO_EMPLOYE	POSTE_EMP
Q22	12564	Assistant
Q22	12566	Commis
E11	13255	Commis
E11	12566	Commis
E11	32155	Gérante

Dans cette table, la clé primaire est la combinaison (NO_DEPARTEMENT, NO_EMPLOYE), et POSTE_EMP est un attribut dépendant de cette clé.

iii) Clés primaires, alternatives et étrangères

Clés primaires :

- Table 1 (DEPARTEMENT) : NO_DEPARTEMENT
- Table 2 (EMPLOYE) : NO_EMPLOYE
- Table3 (AFFECTATION): (NO_DEPARTEMENT, NO_EMPLOYE)

Clé alternative :

- Table 1 (DEPARTEMENT) : NOM_DEP (qui pourrait aussi être utilisé pour identifier un département)

Clés étrangères :

- Dans la table AFFECTATION, NO_DEPARTEMENT et NO_EMPLOYE sont des **clés étrangères** reliant respectivement la table DEPARTEMENT et la table EMPLOYE.

Normalisation en 3NF

a) Identification des dépendances fonctionnelles

Les dépendances fonctionnelles de la table sont les suivantes :

- **df1** : NO_LIVRE → TITRE_LIV, NB_STOCK, ISBN, CODE_MAISON_EDITION

- Un numéro de livre (NO_LIVRE) détermine son titre, la quantité en stock, son ISBN et le code de la maison d'édition.
- **df2** : CODE_MAISON_EDITION → NOM_MAI
 - Un code de maison d'édition (CODE_MAISON_EDITION) détermine le nom de la maison d'édition (NOM_MAI).
- **df3** : (NO_LIVRE, CODE_MAISON_EDITION) → TITRE_LIV, NB_STOCK, ISBN, NOM_MAI
 - La combinaison du numéro de livre et du code de la maison d'édition détermine le titre du livre, la quantité en stock, l'ISBN et le nom de la maison.
- **df4** : ISBN → TITRE_LIV, CODE_MAISON_EDITION
 - L'ISBN détermine le titre du livre et le code de la maison d'édition.

b) Pourquoi la table n'est pas en 3NF

La table n'est pas en 3NF en raison de **dépendances transitives**. En particulier, la dépendance CODE_MAISON_EDITION → NOM_MAI est transitive par rapport à la clé primaire NO_LIVRE. En d'autres termes, NO_LIVRE détermine CODE_MAISON_EDITION, qui à son tour détermine NOM_MAI, ce qui constitue une violation de la 3NF.

c) Normalisation en 3NF

Pour normaliser cette table en 3NF, nous devons éliminer les dépendances transitives.

i) Dépendance transitive

La dépendance transitive identifiée est :

- **df2** : CODE_MAISON_EDITION → NOM_MAI
 - Cette dépendance est transitive car NOM_MAI dépend indirectement de NO_LIVRE à travers CODE_MAISON_EDITION.

ii) Extraction de la dépendance transitive dans une nouvelle table

Nous créons une nouvelle table MAISON_EDITION qui contiendra les informations sur les maisons d'édition, en déplaçant NOM_MAI dans cette nouvelle table. La table originale est alors décomposée en deux tables : LIVRE et MAISON_EDITION.

Table MAISON_EDITION:

CODE_MAISON_EDITION	NOM_MAI
WE1	Webster
EY2	Eyrolles
IN4	Les incroyables
WE1	Webster

Table Livre:

NO_LIVRE	TITRE_LIV	ISBN	NB_STOCK
001	Les bases de données en folie	1503-1555-3618	51
002	L'informatique pour les nuls	2206-1555-6756	125
003	La bible du bit	1212-1371-3000	10
004	Un clavier pour tous	1326-6555-3131	1

d) Clés primaires, alternatives et étrangères en 3NF

i) Clés primaires

- Table LIVRE : NO_LIVRE
- Table MAISON_EDITION : CODE_MAISON_EDITION

ii) Clés alternatives

- Table LIVRE : ISBN (comme clé alternative car l'ISBN identifie aussi de manière unique un livre)
- Table MAISON_EDITION : NOM_MAI (le nom de la maison d'édition pourrait être une clé alternative)

iii) Clé étrangère

- Table LIVRE : CODE_MAISON_EDITION (référence à la table MAISON_EDITION)

Révision

- a)** La normalisation vise à créer un ensemble de relations qui répondent aux besoins de données d'une entreprise en minimisant le nombre d'attributs et la redondance, tout en regroupant logiquement les attributs liés. Cela permet de simplifier l'accès aux données, de faciliter leur maintenance, et d'optimiser l'utilisation de l'espace de stockage. Cette technique peut être appliquée à différents moments de la conception d'une base de données, soit comme méthode indépendante pour une conception ascendante, soit comme outil de validation après une modélisation ER, l'objectif restant de concevoir des relations bien structurées adaptées aux besoins de l'entreprise.
- b)** 14.2 Discuss the alternative ways that normalization can be used to support database design

La normalisation peut soutenir la conception de bases de données de deux façons : soit en tant qu'approche ascendante (bottom-up), où elle sert à structurer progressivement les données brutes en relations bien organisées, soit comme méthode de validation (top-down), permettant de vérifier que les relations créées respectent les principes de normalisation après une modélisation conceptuelle. Dans les deux cas, l'objectif est de minimiser la redondance et d'assurer une structure cohérente qui répond aux besoins de l'entreprise.

c) 15.2 What conditions would make a 3NF relation violate Boyce-Codd Normal Form (BCNF)?

Une relation en troisième forme normale (3NF) peut violer la forme normale de Boyce-Codd (BCNF) dans certaines situations. En 3NF, une dépendance fonctionnelle $A \rightarrow B$ est autorisée si A fait partie de la clé primaire et que B n'est pas une clé candidate. Cependant, BCNF exige que tout déterminant d'une dépendance soit une clé candidate. Une violation de BCNF peut donc se produire lorsque :

1. La relation comporte plusieurs clés candidates composites.
2. Les clés candidates se chevauchent, c'est-à-dire qu'elles partagent un ou plusieurs attributs.

Ces cas peuvent entraîner des redondances et des anomalies de mise à jour qui ne sont pas complètement éliminées en 3NF.