

*Département Informatique*

*BUT 1*

**Ressource R1.05 :  
Introduction aux bases de données et SQL**

November 14, 2022

**Cours et exercices**



# Contents

<b>1</b>	<b>Le modèle relationnel</b>	<b>3</b>
1	Introduction . . . . .	3
2	Fondements . . . . .	4
<b>2</b>	<b>TD/TP</b>	<b>9</b>
1	degré, cardinalité, clés (primaire, candidate), dépendance fonctionnelle et forme normale . . . . .	9
1.1	<i>Exercice</i> : . . . . .	9
1.2	<i>Exercice</i> : . . . . .	9
1.3	<i>Exercice</i> : . . . . .	10
1.4	<i>Exercice</i> : . . . . .	10
1.5	<i>Exercice</i> : . . . . .	10
1.6	<i>Exercice</i> : . . . . .	10
1.7	<i>Exercice</i> : . . . . .	11
1.8	<i>Exercice</i> : . . . . .	11
2	Clés étrangères . . . . .	11
2.1	<i>Exercice</i> : . . . . .	11
2.2	<i>Exercice</i> : . . . . .	12

# 1 Le modèle relationnel

## 1 Introduction

Ce chapitre est consacré au modèle relationnel, le modèle logique, supporté par les SGBD relationnels. La modélisation logique (le processus) permettant d'obtenir le schéma logique, qui sera implémenté par les SGBD, est présentée dans le chapitre suivant.

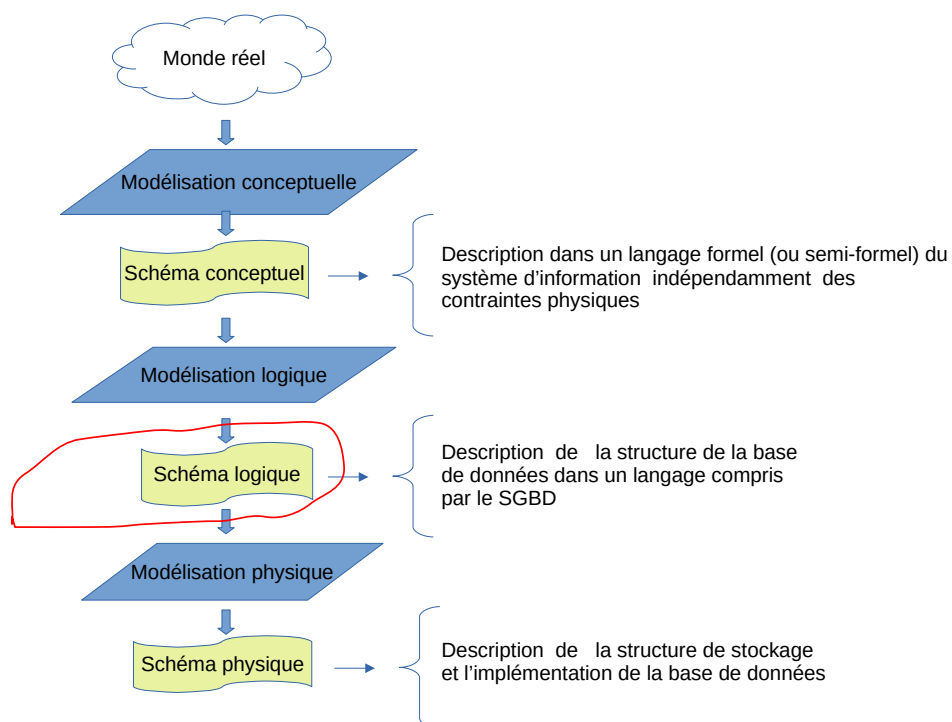


Figure 1.1: Modélisation conceptuelle : les étapes

1. le modèle relationnel a été proposé dans les années 70 par Codd (chercheur chez IBM),
- 

2. le modèle relationnel exploite la notion de relation mathématique, qui se fonde sur la théorie des ensembles et la logique des prédicats de premier ordre,

## 2 Fondements

1. Une relation dans le modèle relationnel est conçue comme une table de valeurs,
  2. Une base de données est conçue comme un ensemble de relations,
  3. Les en-têtes de colonnes d'une relation sont appelés attributs,
  4. domaine: Un domaine  $D_i$  est un ensemble de valeurs **atomiques** (indivisible dans le cas du modèle relationnel);
  5. Un attribut  $A_i$  est une variable qui prend ses valeurs dans un ensemble  $D_i$  appelé domaine, noté :  $\text{dom}(A_i)=D_i$ ;
  6. le degré d'une relation est le nombre d'attributs qu'elle contient,
  7. Chaque ligne dans une relation est appelée tuple (ou n-uplet),
  8. n-uplet (ou tuple) : un n-uplet  $t=(v_1,v_2,...,v_n)$  est une liste ordonnée de valeurs où chaque valeur  $v_i$  est un élément de  $\text{dom}(A_i)$  ou une valeur spéciale nulle (inconnue ou inexistante),
- 
1. Relation : Soient  $D_1, D_2,..., D_n$ , une suite ordonnée de  $n$  domaines non forcément distincts, une relation  $R$  sur ces  $n$  domaines est une partie du produit cartésien :  $D_1 \times D_2 \times D_n$ ,
  2. le cardinal d'une relation est le nombre de n-uplet qu'elle contient,
  3. Dans une relation il n'y a aucun tuple en double,
  4. l'ordre des tuples dans une relation n'a pas de signification,
  5. l'extension d'une relation représente son état à un instant donné (les n-uplets),
  6. Dans une base de données, une relation a un nom distinct de toutes les autres relations.

A	B	C
1	2	3
5	1	1

Table 1.1: R1

**Exemple d'une Relation** On considère la relation R1, noté R1(A,B,C) :

Attributs : A,B et C;

Dom(A) = {1,5}, dom(B) = {1,2}, dom(C) = {3,1}

Extension de R1 :

n-uplets : (1,2,3) et (5,1,1)

degré(R)= 3; cardinal(R)=2

**Schéma de relation** Le schéma de relation noté

$$\overline{R} = R < U, F >$$

est une définition en intention.

1. il contient le nom de la relation R,
2. la liste U de ses attributs et le domaine de chaque attribut,
3. l'ensemble F des contraintes d'intégrité associées à la relation.

**Exemple** Le schéma de relation R1 précédente est noté comme suit :

$$\overline{R1} = R1 < U1, F1 >$$

U1 = {A,B,C}

Dom(A) = {1,5}, dom(B) = {1,2}, dom(C) = {3,1}

F1 = {contraintes sur les attributs de R1}

### Contraintes d'intégrités

- contraintes de domaine: il est possible de décrire le domaine d'un attribut par un sous ensemble de valeurs ou un type énuméré dans lequel toutes les valeurs possibles sont explicitement listées

Exemple : l'attribut A prend ses valeurs 1 et 4 :

Dom(A) = {1,4}

- contraintes de clé: Dans une relation R, il existe (au moins) un sous-ensemble d'attributs dont les valeurs permettent de distinguer les tuples de R les uns des autres.

Si cet ensemble d'attributs est minimal alors il constitue une **clé** de la relation.

Lorsqu'il existe plusieurs clés, l'une d'elles est choisie comme clé **primaire** et les autres clés sont des clés **candidates**.

**Exemple** On considère la relation R1 suivante:

A	B	C
1	2	3
2	3	1
3	4	1
5	1	3

Table 1.2: R1

Deux clés dans R1 : A et B

Par exemple, on choisit A comme clé primaire et B clé candidate

- contrainte de clé étrangère : Soient deux relations R1 et R2. On dit que l'ensemble K1 est une clé étrangère de R1 (qui référence à la clé K2 de R2) si :
  1. les attributs de K1 ont les mêmes domaines que les attributs de clé K2 de R2
  2. la valeur en K1 d'un tuple t1 de R1 est la valeur en K2 d'un tuple t2 de R2
- Autres contraintes : Ces contraintes sont respectées en appliquant un langage spécifique : le langage des déclencheurs (ou triggers).

A	B	C
1	1	3
2	3	1
3	3	1

Table 1.3: R1;  $\text{dom}(A) = \{1,2,3\}$ ;  $\text{dom}(B)=\{1,3\}$ ;  $\text{dom}(C)=\{1,3\}$   
A est la clé primaire de R1

D	E	F
1	4	3
2	3	3
3	3	1

Table 1.4: R2

D est la clé primaire de R2; F est une clé étrangère dans R2.

**Dépendances fonctionnelles** Soit la relation  $R(A,B,C)$ , l'attribut B est dit fonctionnellement dépendant de l'attribut A si étant donnée 2 n-uplets :  $\langle a1, b1, c1 \rangle$  et  $\langle a2, b2, c2 \rangle$   $a1 = a2 \Rightarrow b1 = b2$ . On la note  $A \rightarrow B$

A	B	C
1	2	3
1	3	1
3	3	1
5	2	3
5	2	4

Table 1.5: R

On a une dépendance fonctionnelle dans R :  $C \rightarrow B$

**Propriétés des dépendances fonctionnelles** Soit une relation R (A,B,C,D)

1. réflexivité :  $A \rightarrow A$
2. augmentation : si  $A \rightarrow B$  alors  $A, C \rightarrow B$
3. transitivité : si  $A \rightarrow B$  et  $B \rightarrow C$  alors  $A \rightarrow C$
4. pseudo-transitivité : si  $A \rightarrow B$  et  $B, C \rightarrow D$  alors  $A, C \rightarrow D$
5. union : si  $A \rightarrow B$  et  $A \rightarrow C$  alors  $A \rightarrow B, C$
6. décomposition : si  $A \rightarrow B, C$  alors  $A \rightarrow B$  et  $A \rightarrow C$ .

### Typologie des dépendances fonctionnelles

- dépendance trivial : une dépendance  $X \rightarrow Y$  est triviale si  $Y \subset X$ ,
- dépendance simple : une dépendance  $X \rightarrow Y$  est dite simple si sa partie droite ne comporte qu'un seul attribut,
- dépendance élémentaire (complète): une dépendance  $X \rightarrow Y$  est élémentaire si pour tout  $X' \subset X$ , la dépendance fonctionnelle  $X' \rightarrow Y$  n'est pas vérifiée. En d'autre terme Y ne dépend pas fonctionnellement d'une partie de X,
- dépendance directe : une dépendance  $X \rightarrow Y$  est directe si Y ne dépend pas de X par transitivité (cad il existe Z tel que  $X \rightarrow Z$  et  $Z \rightarrow Y$ ).



A	B	C
1	2	3
1	4	1
3	3	1
2	2	3
4	2	3
4	4	1

Table 1.6: R3

**Exemple : dépendance fonctionnelle élémentaire** La clé primaire de R3 : AB; On déduit :  $AB \rightarrow C$ ; On a aussi la dépendance  $B \rightarrow C$ ; La dépendance  $AB \rightarrow C$  n'est pas élémentaire.

A	B	C
1	2	3
3	4	1
5	3	1
4	2	3
2	2	3

Table 1.7: R4

**Exemple : dépendance fonctionnelle transitive** La clé primaire de R4 : A; On déduit :  $A \rightarrow B$  et  $A \rightarrow C$ ; On a aussi la dépendance  $B \rightarrow C$ ; La dépendance  $A \rightarrow C$  est une dépendance transitive (elle n'est pas directe).

**Formes Normales** Les trois premières formes normales:

- 1NF: une relation est dite en 1NF si tout attribut a un domaine qui contient uniquement des valeurs atomique,
- 2NF : une relation est dite en 2NF si et seulement si elle est en 1NF et que tout attribut n'appartenant pas à la clé de la relation R ne dépend pas d'une partie de la clé de R,
- 3NF: une relation est dite en 3NF si elle est en 2NF et si tout attribut n'appartenant pas à la clé de R ne dépend pas transitivement de la clé.

**Exemple : formes normales**

- La relation R3 est en 1NF (elle n'est pas en 2NF à cause de la dépendance  $AB \rightarrow C$  qui n'est pas complète et donc elle n'est pas en 3NF)
- La relation R4 est en 1NF et en 2NF (elle n'est pas en 3NF à cause de la dépendance transitive)

## 2 TD/TP

### 1 degré, cardinalité, clés (primaire, candidate), dépendance fonctionnelle et forme normale

#### 1.1 Exercice :

On considère la relation suivante R constituée de quatre attributs A, B, C et D :

A	B	C	D
a1	b2	c1	d3
a1	b2	c1	d2
a1	b2	c2	d3
a3	b1	c2	d2
a4	b4	c3	d2

- 1 ► Quelle est la cardinalité de cette relation ?
- 2 ► Quel est son degré ?
- 3 ► Quelles sont les clés de cette relation ?
- 4 ► Existe-t-il une dépendance fonctionnelle  $A \rightarrow B$ ,  $A \rightarrow C$  ?

#### 1.2 Exercice :

On considère la relation suivante R constituée de quatre attributs A, B, C et D :

A	B	C	D
a1	b2	c1	d1
a1	b2	c1	d2
a1	b2	c2	d3
a2	b4	c2	d2
a3	b3	c4	d2

- 1 ► Quelles sont les clés candidates de cette relation ?
- 2 ► Existe-t-il une dépendance fonctionnelle  $A \rightarrow B$ ,  $A \rightarrow C$  ?

### 1.3 Exercice :

On considère la relation  $R(A,B,C)$  munie de l'ensemble  $F$  de dépendances fonctionnelles suivant:  $F = \{A \rightarrow B, B \rightarrow C\}$

- 1 ► Quelle est la clé de cette relation ?
- 2 ► Déterminez la forme normale de  $R$
- 3 ► La relation suivante est-elle conforme au schéma de relation  $R$  ?

A	B	C
1	1	1
2	1	2
3	2	1
4	3	3

### 1.4 Exercice :

Soit la relation  $R(A,B,C,D,E)$  munie de l'ensemble de dépendances fonctionnelles suivant:

$F = \{A \rightarrow B, A \rightarrow D, A \rightarrow E, E \rightarrow C\}$

- 1 ► Trouvez la clé de  $R$
- 2 ► Quelle est la forme normale de  $R$  ?

### 1.5 Exercice :

Soit la relation  $R(A,B,C,D,E)$  munie de l'ensemble de dépendances fonctionnelles suivant:  $F = \{A \rightarrow B, C \rightarrow E, AE \rightarrow D\}$

- 1 ► Trouvez la clé de  $R$
- 2 ► Quelle est la forme normale de  $R$  ?

\_\_\_\_\_

### 1.6 Exercice :

On considère la relation  $R$  suivante composée de cinq attributs  $A, B, C, D$  et  $E$  :

A	B	C	D	E
1	1	1	1	1
1	1	2	1	1
1	1	5	1	1
1	2	5	4	1
2	1	2	1	1
2	1	5	1	1
2	1	5	3	1
2	2	5	4	1
3	3	3	5	2

1 ► Quelles sont les dépendances fonctionnelles incorporées dans cette relation ?

2 ► Quelles sont les clés candidates de la relation ?

3 ► Quelle la forme normale de la relation ?

## 1.7 Exercice :

1 ► Quel est l'intérêt des formes normales ? Donnez des exemples justifiant vos réponses

## 1.8 Exercice :

Soit un schéma de bases de données contenant les relation suivantes :

DRAGONS (Dragon, Sexe, Longueur, NombreEcailles, CracheduFeu, ComportementAmoureux)  
avec FDRAGONS = { Dragon → Sexe, Longueur, NombreEcailles, CracheduFeu, ComportementAmoureux }

AIME = (DragonAimant, DragonAimé, Force),

avec FAIME = { DragonAimant → DragonAimé, Force }

NOURRITURES (Produit, Calories)

REPAS (Dragon, Produit, Quantité)

Les contraintes ci-dessous sont-elles vérifiées par ce schéma de bases de données?

- 1 ► Un dragon peut avoir plusieurs comportements amoureux;
- 2 ► Un dragon peut aimer plusieurs autres dragons;
- 3 ► Un dragon peut aimer un autre dragon de plusieurs "forces";
- 4 ► Un dragon peut consommer plusieurs produits;

## 2 Clés étrangères

### 2.1 Exercice :

On considère le schéma d'une base de données consignant les fourniture de produits de différents fournisseurs. Ce schéma est composé de plusieurs relations :

Fournisseur (nofr, nom, ville)

Fourniture(nofr, nopr, quantite)

Produit(nopr, nom, prix)

1 ► Indiquez quelles clés étrangères s'appliquent à ce schéma en justifiant vos hypothèses

## 2.2 Exercice :

On considère le schéma d'une base de données décrivant les ventes d'un concessionnaire automobile:

Voiture(noserie,modèle,marque,prix)

Vendeur(idvendeur,nom,tél)

Vente(idvendeur,noserie,date,montant)

- 1 ► Indiquez quelles clés étrangères s'appliquent à ce schéma en justifiant vos hypothèses
- 2 ► Renseignez les relations à l'aide de quelques tuples
- 3 ► Donnez un exemple d'insertion de tuples dans les relations Vente et Vendeur qui viole les contraintes d'intégrité ainsi qu'un exemple qui le respecte.
- 4 ► Est-il possible de supprimer un tuple dans la table Vente ?
- 5 ► Quelles sont les vérifications à faire si on veut supprimer un tuple dans la table Vendeur ?