Cours de base de donnéesé

chap 4
Dépendances fonctionnelles et normalisation

Par: Kamal BAL

Université AMOB de Bouira
Faculté des sciences et des sciences appliquées
Département d'informatique

https://sites.google.com/a/esi.dz/kamalbal

La normalisation d'un schéma relationnel

- **L'objectif** : construire un schéma de base de données **cohérent**.
- Un **mauvais schéma logique** peut conduire à un certain nombre **d'anomalies** pendant la phase d'exploitation de la base de donnée.
- Pour qu'un modèle relationnel soit **normalisé**, il faut qu'il respecte certaines contraintes appelées **les formes normales.**
- Les formes normales s'appuient sur la notion des **dépendances fonctionnelles** entre attributs

La normalisation d'un schéma relationnel

La construction d'un modèle E/A mais également du modèle relationnel correspondant, repose presque entièrement sur le concept de **dépendance fonctionnelle.**

C'est ce concept qui permet de passer d'un **ensemble de propriétés non structuré** à un modèle conceptuel des données formé d'entités et d'associations et au modèle relationnel correspondant

Les dépendances fonctionnelles (DF)

- Soit une relation R: **R(A1,A2, … ,An)**
- **R** est définit sur un les attributs **A= {A1,A2, … ,An}**
- Soient: **X** et **Y** des sous-ensembles de **A**.
- **Y dépend fonctionnellement de X ou X détermine Y** **si, et seulement si :**
  - Des **valeurs identiques** de **X** impliquent des **valeurs identiques de Y.**
  - A une valeur **x** de **X** correspond une et une seule valeur **y** de **Y**
- **Notation :** $X \longrightarrow Y$

Les dépendances fonctionnelles (DF)

- Soit **R(A, B, C)** une relation.
- L'attribut **B** est dit **fonctionnellement dépendant** de l'attribut **A** si:
  - Etant donné : $<a_1, b_1, c_1>$ et $<a_2, b_2, c_2>$ $\in$ **R**
  - Si $a_1 = a_2 \Rightarrow b_1 = b_2$
  - Ou encore, **A détermine B** si étant donné une valeur de **A**, il lui correspond une seule valeur unique de **B**.

ON NOTE A → B

A: SOURCE de la DF ET B: Cible (BUT) de la DF

Les dépendances fonctionnelles (DF)

Exemple :

Produit(reference, designation, prix, qte_stock)

Contient les DF suivantes :

reference → designation

reference → prix, qte_stock

Evaluation (Matrivule, nom, prenom, niveau, module, note_dans_mdoule)

Contient les DF suivantes :

Matricule → nom, prenom, niveau

Matricule, module→note_dans_module

DF Elémentaire

DF élémentaire :

- **X → A** est une **DF élémentaire** si **A** est un attribut **unique** non inclus dans **X** et il n'existe pas de **X' inclus dans X** tel que **X' → A**

- **Ex.**
  - `Ref_produit → designation_produit ;`
  - `N_commande, ref_produit → quantité_commandée`
  - `N_commande, ref _produit → designation_produit`
    - (**Non élémentaire**) Car : ref _produit → designation_produit

DF Elémentaire

DF directe :

- **X → A** est une **DF directe** si elle ne peut être déduite par **transitivité** à partir d'autres DFs
- ie: **Il n'existe pas** un attribut **B** tel que :**X → B → A**

EX.

```
N_client → Nom_Client ;   est directe
N_commande→ N_Client ;    est directe
```

`N_commande → Nom_client;` n'est pas **Directe** car

N_commande→ N_Client → Nom_client

Les dépendances fonctionnelles

DF directe :

- **X → A** est une **DF directe** si elle ne peut être déduite par **transitivité** à partir d'autres DFs
- ie: **Il n'existe pas** un attribut **B** tel que :**X → B → A**

- **N_conducteur →Nom_Conducteur; N_permis → N_Cconducteur;**

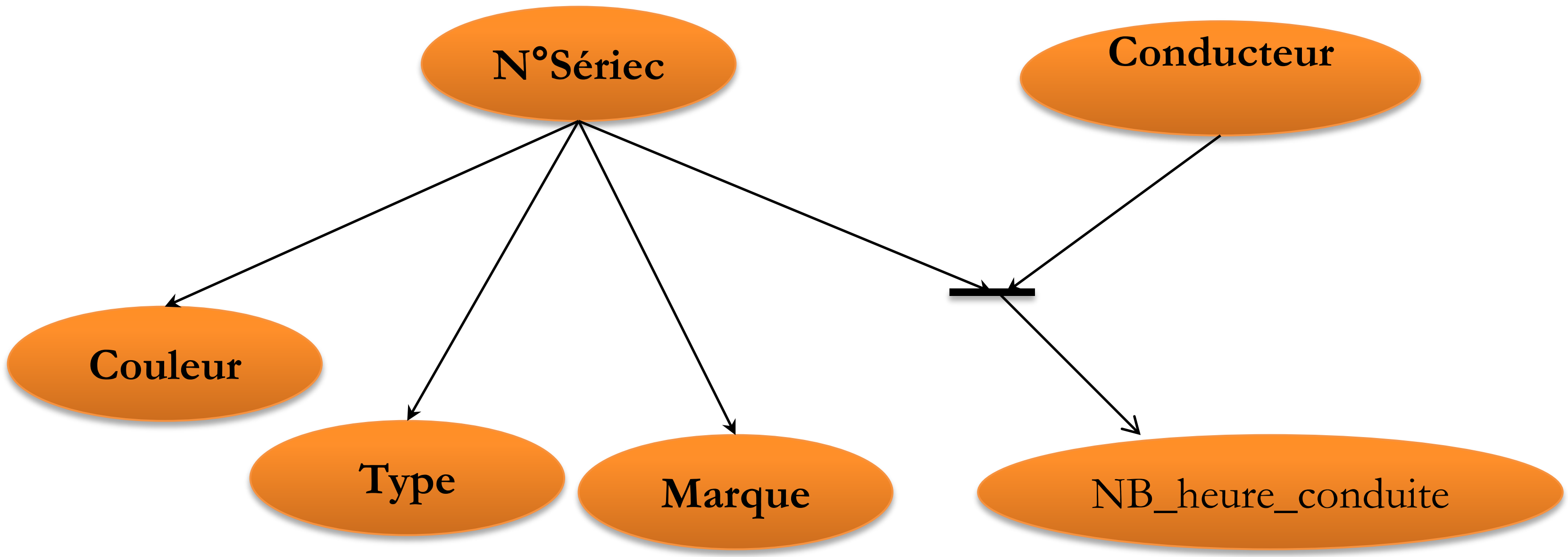
- **N_permis → Nom_Conducteur ; n'est pas une DF Directe car :**

  N_permis→ N_Cconducteur → Nom_Conducteur

Graphe des DFs

Graphe de dépendances fonctionnelles

- N°Série → Couleur
- N°Série → Type, Marque
- N°Série, Conducteur → NB_heure_conduite



Les dépendances fonctionnelles

- **Exemple** : Soit la relation suivante R de schéma R (A, B, C, D, E).

A	B	C	D	E
a1	b1	c1	d1	e1
a1	b2	c2	d2	e1
a2	b1	c3	d3	e1
a2	b1	c4	d3	e1
a3	b2	c5	d1	e1

Les dépendances fonctionnelles satisfaites par R sont ????

Les dépendances fonctionnelles

- **Exemple** : Soit la relation suivante R de schéma R (A, B, C, D, E).

A	B	C	D	E
a1	b1	c1	d1	e1
a1	b2	c2	d2	e1
a2	b1	c3	d3	e1
a2	b1	c4	d3	e1
a3	b2	c5	d1	e1

Les dépendances fonctionnelles satisfaites par R sont les suivantes ??? :

A → E ; B → E ; C → B BD → A

C → A ; D → E ; C → D AB → D ;

AD → B ; C → E

Les dépendances fonctionnelles

- **Exemple :** Soit une relation R exprimant l'emploi du temps d'une école construite sur les attributs suivants :
  - **P** (professeur),
  - **H** (heure du cours),
  - **S** (Salle),
  - **C** (classe) et
  - **M** (matière).
- La signification d'un n-uplet de cette relation est : Le professeur P enseigne la matière M à l'heure H dans la salle S à la classe C. Donnez la liste des dépendances fonctionnelles.
- Solution :

  P→M; S,H→M; S,H→C; C,H→S,M; …….

Propriétés des DFs : axiomes d'Armstrong

Axiomes d'Armstrong :

- Système de règles d'inférences définit par Armstrong en 1974 :
- Déduire d'autres DFs à partir des trois propriétés suivantes :

Transitivité :

- Si **X → Y**, et **Y → Z**, alors **X → Z**

Augmentation :

- Si **X →Y**, alors **XZ →YZ**
  - pour tout groupe **Z** d'attributs appartenant au schéma de relation

Réflexivité :

- si **X contient Y**, alors **X → Y** (ex. A → A ; AB → A)
  - **Y í X alors X→Y (et donc X→X)**

Propriétés des DFs : axiomes d'Armstrong

A partir de ces trois axiomes de base, on peut déduire d'autres règles :

- **Union :**
  - si $X \rightarrow Y$ et $X \rightarrow Z$, alors $X \rightarrow YZ$,
- **Pseudo-transitivité :**
  - si $X \rightarrow Y$ et $WY \rightarrow Z$, alors $WX \rightarrow Z$,
- **Décomposition :**
  - si $X \rightarrow Y$ et $Z \subseteq Y$, alors $X \rightarrow Z$.

Fermeture transitive

Fermeture transitive d'un ensemble de DFs :

- Soit **D**, un ensemble de DFs **élémentaires**,
- la **fermeture transitive** **(D^+)** de D est l'ensemble des DFs de D **enrichi** de toutes les DFs élémentaires obtenues par **transitivité**

- **Exemple :**

D={NV→Type ; Type →Marque ; Type →Puiss ; NV →coul}

on a : **NV→Type** et **Type → Marque** => **NV → Marque**
NV→Type et **Type → Puiss** => **NV → Puiss**

Donc : D+ = D UNION { NV → Marque ; NV → Puiss }

- Deux ensembles de DFs élémentaires sont **équivalents** s'ils ont la **même** fermeture transitive.

Fermeture d’un ensemble d’attributs

- **La fermeture d'un ensemble d'attributs X:**
  - **La fermeture transitive d’un ensemble d’attributs X** sous un ensemble **F** de DFs , notée **$(X)^+$** représente l'ensemble des **attributs qui peuvent être déduits** de X à partir de l’ensemble F des DFs.
  - **Ainsi, Y** sera inclus dans $(X)^+$ ssi $X \rightarrow Y$.

Fermeture d'un ensemble d'attributs

- **Calcul de la fermeture d'un ensemble d'attributs** :

a) initialiser **(X)+ à X**,

b) Chercher une df ***f*** de F tel que :

La partie gauche de ***f*** inclus dans (X)+

c) Ajouter les attributs de la partie gauche de ***f*** a **(X)+**

d) Répéter les étapes **b)** et **c)** jusqu'à ce que **(X)+** n'évolue plus.

Fermeture d'un ensemble d'attributs

- **Exemple:** :
  - **F = {A→D; AB→E ; BI→E; CD→I; E→C}.**
  - Calculer la fermeture, sous F, de AE.
- **Solution** :
  - **au départ, (AE)+ = AE,**
  - **A →D permet d'ajouter D : (AE)+ = AED,**
  - **E → C permet d'ajouter C : (AE)+ = AEDC,**
  - **CD → I permet d'ajouter I : (AE)+ = AEDCI.**

Fermeture d'un ensemble d'attributs

- **Calcul de la fermeture d'un ensemble d'attributs** :

a) initialiser **(X)+ à X**,

b) Chercher une df ***f*** de F tel que :

La partie gauche de ***f*** inclus dans (X)+

c) Ajouter les attributs de la partie gauche de ***f*** a **(X)+**

d) Répéter les étapes **b)** et **c)** jusqu'à ce que **(X)+** n'évolue plus.

- **Exemple:** :
  - **F={A→D; AB→E; BI→E; CD→I; E→C }.**
  - Calculer la fermeture, sous F, de BE
- **Solution :**
  - **au départ, (BE)+ = BE,**
  - **E → C permet d'ajouter C : (BE)+ = BEC.**

Couverture minimale d'un ensemble de DFs

- Soit un ensemble de dépendances fonctionnelles élémentaires **F** pour un ensemble d'attributs **A,**
- **CM(F)** est une **couverture minimale** de F si
  - Toute **DF *f* de F** n'est pas **redondante** (***F-f*** n'est pas équivalant à **F**)
  - Toute DF élémentaire de A est dans la fermeture transitive F+

C'est le sous ensemble minimale de DFs permettant de générer toutes les autres DFs.

CM(F)+= F+

Il n'existe pas F' ⊆ CM(F) tel que : F'+ = F+

Tout ensemble de DFs élémentaires a une couverture minimale – Cette couverture peut ne pas être unique.

Les dépendances fonctionnelles

Couverture minimale

- **Exemple**
  - F = {
    - NV → Type ;
    - Type → Marque ;
    - NV → Couleur ;
    - NV → Marque ;
    - NV → Puiss ;
    - Type, Marque) → Rabais ;
    - NV → Rabais

    }
  - CM(F) = {
    - NV → Type ;
    - Type → Marque ;
    - NV → Couleur ;
    - NV → Marque ;
    - NV → Puiss ;
    - Type, → Rabais ;

    }

Les dépendances fonctionnelles

DF redondante

- Soit **F** un ensemble de DFs
- Une DF f: $(X \rightarrow Y)$ est redondante dans **F** SSI :
- $\mathbf{Y} \in \mathbf{X}^+$ sous F-{f}

- **Exemple :**
  - F={A→B, B→C, A→C} ;
  - A→C est redondante car $C \subseteq A^+$ sous F-(A→C)
    - A^+{A→B, B→C}={A, B, C}

Couverture minimale : Algorithme

- Soit **F:** un ensemble de DFs élémentaires sur un ensemble d'attribut **A**

```
Début
CM(F) = F   // l'ensemble des DFs
Eclater les parties droites des DF
   Remplacer X→A1,A2,…, Ak Par: X→A1;X→A2;  ; X→Ak;
Pour chaque DF f (X→A)
      Calculer (X)+ Sous (F-{f})// couverture de X Sous
      (F-{f})
      Si A ∈  (X)+ alors CM(F) = CM(F) - f
Fin pour
Retourner CM(F);
Fin
```

Les DFs et la notion de clé

- **DF et notion de clé**
- Soit R (A1 , A2 , … , An) une relation, X un ensemble d'attributs inclus dans {A1 , A2 , … , An } est une clé de R si
  - **X → A1 A2 … An**
  - **Il n'existe pas Y inclus dans X, tel que Y → A1 A2 … An**

- **X est une clé alors :**
  - **$(X)^{+}$= {A1 , A2 , … , An }**
  - $\neg\exists$ X' $\subseteq$ X / (X')+= **{A1 , A2 , … , An }**

Les DFs et la notion de clé

Comment calculer une clé d'une relation ?

Calcule une clé **K** d'un schéma relationnel **R** sur un ensemble d'attributs **U** avec un ensemble de DFs **F :**

- **Debut**
  - K := U; {l'ensemble de tout les attributs}
  - Tant que ∃ un attribut A ∈ K tel que K-{A} →U faire
    - K := K -{A};
  - Fin tant que
- **Fin**

Il existe d'autres algorithmes capables de calculer l'ensemble des clés possibles pour un schéma *R.*