Bases de Données: Normalisation

Sergio Peignier

Quelques questions

- Pour une même application il est souvent possible de proposer plusieurs schémas.
- Un mauvais schéma conceptuel $\Rightarrow$ anomalies :
    - Pendant la phase d'exploitation de la base (redondances d'information ...).
    - Lors des opérations de mise à jour (insertions, suppressions, modifications).
- Comment peut-on distinguer un bon schéma d'un mauvais schéma ?
- Quel schéma doit-on choisir ?

Un exemple

- Considérons le schéma de relation KEBAB(Nom_kebab, Adresse_kebab, Produit_vendu, Prix).
- Cette relation pourra éventuellement contenir plusieurs produits pour un même kebab.
- Quels problèmes identifiez vous ?

Un exemple

- L'adresse du Kebab sera répété pour chaque produit vendu par le Kebab en question (redondance).
- Si Kebab change d'adresse il faudra rechercher et modifier tous les tuples associées à ce Kebab.
- Si un Kebab (déjà existant) sort un nouveau produit il faudra vérifier que l'adresse connue et l'adresse de la nouvelle instances sont les mêmes.
- Si un Kebab fait faillite et disparait il faudra retrouver et supprimer tous les tuples correspondant à ce Kebab (pour différents produits) dans la table.

Objectif de la normalisation

- Construire un schéma de base de données cohérent et possédant certaines propriétés vérifiées par la satisfaction de "formes normales" (évitant ainsi l'apparition des anomalies).

Définition : Dépendances Fonctionnelles (DF)

- Un ensemble d'attributs B est dit **fonctionnellement dépendant** d'un ensemble d'attributs A si on sait que lorsque deux tuples quelconques coincidant sur leurs attributs A, alors ils coincident sur leurs attributs B.
- Pour x et y deux tuples quelconques, si $x_A = y_A \Rightarrow x_B = y_B$ alors B est fonctionnellement dépendant de A.
- On dit aussi que A détermine B (on écrit $A \rightarrow B$).
- Les DF sont des contraintes qui décrivent les relations entre les attributs d'une relation.
- Les formes normales s'appuient sur les DF entre attributs d'un schéma de base de données.
- Une notation habituelle en BD : Soient deux ensembles d'attributs X et Y On note XY l'union de X et Y (au lieu de X∪Y).

Exemple

- Écrire les dépendances fonctionnelles de la relations suivante : ETUDIANT (No_SS, Nom, Prenom, Adresse, Age).
- Que remarquez vous ?

Dépendances Fonctionnelles

- Une clé détermine tous les attributs du schéma de relation. Il s'agit d'une propriété de la clé d'un schéma de relation.
- Les dépendances fonctionnelles ne se définissent pas par rapport à **une** table en particulier (ou à la table à **un instant donné**) mais par rapport aux **caractéristiques intrinséques des attributs** !
- Imaginez la relation suivante :
  Kebab :

Nom_kebab	Adresse_kebab	Produit_vendu	Prix
Ambiance	bvd blabla	miche kebab	5
Mosaique	bvd blabla	maxi kebab	7.5

- Est-ce que le produit détermine le nom, l'adresse et le prix ?

Dépendances Fonctionnelles

Soit la relation suivante r ayant le schéma $R(A, B, C, D, E)$. Si on imagine que toutes les dépendances fonctionnelles sont visibles sur le tableau (ce n'est pas toujours le cas attention ! ! !), quelles sont les dépendances fonctionnelles satisfaites par R ?

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

Réponse

- $A \rightarrow E$
- $B \rightarrow E$
- $C \rightarrow ABDE$
- $D \rightarrow E$
- $AB \rightarrow D$
- $AD \rightarrow B$
- $BD \rightarrow A$

Axiomes d'Armstrong

A partir d'un ensemble F de DF entre les attributs A d'une relation R on peut en déduire d'autres grâce aux propriétés suivantes :

- Réflexivité : si $Y \subseteq X$, alors $X \to Y$.
- Augmentation : si $X \to Y$ alors $XZ \to YZ$ pour tout ensemble d'attributs Z appartenant à R
- Transitivité : si $X \to Y$, et $Y \to Z$, alors $X \to Z$.

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

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

Retrouver ces 3 régles à partir des axiomes.

Axiomes d'Armstrong (Union)

- $X \rightarrow Y$ par augmentation : $XZ \rightarrow YZ$
- $X \rightarrow Z$ par augmentation : $XX \rightarrow XZ$
- Or $XX = X$ donc $X \rightarrow XZ$
- Par transitivité : $X \rightarrow YZ$

Axiomes d'Armstrong (Pseudo-transitivité)

- $X \rightarrow Y$ par augmentation : $WX \rightarrow WY$
- Or $WY \rightarrow Z$ par transitivité : $WX \rightarrow Z$

Axiomes d'Armstrong (Décomposition)

- $Z \subseteq Y$ par réfléxivité : $Y \rightarrow Z$
- Or $X \rightarrow Y$ par transitivité : $X \rightarrow Z$

Retour sur les DF

- DF triviale $\mapsto$ Obtenue par réfléxivité (donner un exemple).
- DF simple $\mapsto$ Un seul élément à droite (donner un exemple).
- DF directe $\mapsto$ Qui ne peut pas être obtenue par transitivité :
  $A \rightarrow B$ est directe si $\nexists C$ tel que $A \rightarrow C$ et $C \rightarrow B$
- DF élémentaire (DFE) $\mapsto$ Simple et dont la partie gauche n'est pas décomposable :
  $A \rightarrow B$ est DFE $\nexists C \subset A$ tel que $C \rightarrow B$ et si $|B| = 1$.
- DF complète ou pleine (dans le cas contraire on parle de DF partielle) $\mapsto$ Dont la partie gauche n'est pas décomposable
  $X \rightarrow Y$ est complète si Y n'est pas fonctionnellement dépendant d'un sous-ensemble de X.

Fermeture

- Soit R un schéma de relation.
- Soit X un ensemble d'attributs.
- X^+ fermeture de $X \mapsto$
  Ensemble des attributs de R qui peuvent être déduits de X à partir d'une famille de DF en appliquant les axiomes d'Armstrong.
- Y sera inclus dans X^+ ssi $X \rightarrow Y$

Fermeture

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

- Initialisation $X^+ = X$
- Trouver une dépendance fonctionnelle possédant en partie gauche des attributs inclus dans X^+
- Ajouter dans X^+ les attributs placés en partie droite de la dépendance fonctionnelle.
- Répéter les deux étapes précédentes jusqu'à ce que X^+ n'évolue plus.

Exercice 1

Soit l'ensemble de DF suivant :

$$F = \{A \rightarrow D; AB \rightarrow E; BI \rightarrow E; CD \rightarrow I; E \rightarrow C\}$$

Calculer la fermeture sous F, de AE et de BE.

Exercice 2

Soit l'ensemble de DF suivant :

$$F = \{AB \rightarrow C; B \rightarrow D; CD \rightarrow E; CE \rightarrow GH; G \rightarrow A\}.$$

Montrer que $AB \rightarrow E$,

Fermeture Transitive (F^+) et couverture minimale

- Soit F un ensemble de DFE.
- Quels opérations peut-on appliquer à F pour trouver toutes les DFE qui en découlent ?
- F^+ fermeture transitive de $F \mapsto$
  - Plus grand ensemble (non trivial) de DFE obtenu à partir des DFE de F.
  - $F \cup \{$ DFE obtenues par transitivité ou pseudo transitivité $\}$
  - 2 ensembles de DF sont équivalents s'ils ont la même F^+
- $MIN(F)$ Couverture minimale de F
  - Ensemble minimal $MIN(F)$ de F obtenu en supprimant les dépendances fonctionnelles redondantes, c'est à dire celles qui peuvent être déduites à partir de $MIN(F)$.
  - $(MIN(F))^+ = F^+$ et $\nexists F' \subset MIN(F)$ tel que $F'^+ = F^+$
  - Théorème : Tout ensemble de DFE admet une couverture minimale, en général non unique.

Algorithme de calcul de la couverture minimale

- Écrire les DF sous la forme X $\rightarrow$ A. X étant un ensemble d'attributs et A un attribut élémentaire. (Écrire toutes les DF de ce type que vous pouvez $\rightarrow$ fermeture transitive)
- Les DF du type $X \rightarrow A_1 A_2 \ldots A_n$ sont remplacé par n DF du type $X \rightarrow A_i$ (parfois utile mais ce n'est pas nécéssaire)
- Supprimer des DF grâce aux opérations liés à la notion de fermeture (trouver l'ensemble de DF le plus simple qui permet de retrouver la fermeture transitive).

Retour à la normalisation

- Normalisation $\mapsto$ Méthodologie de conception descendante pour produire un "bon schéma" par décomposition d'un schéma d'origine
- Le schéma produit doit :
  - Éviter des anomalies lors de la mise à jour de la BD (forme normale)
  - Preserver la sémantique du schéma d'origine sans perdre de l'information ni les DF
  - Cependant, la décomposition entraine parfois une dégradation des performances (besoin de faire des jointures pour faire des recherches).
- Il existent plusieurs formes normales. Nous allons voir les 3 premières formes normales (NF1, NF2 et NF3) proposées par E.F. Codd en 1972.

NF1

- Une relation est NF1 si elle ne contient que des "valeurs atomiques" non multi-valuées.
- Exemple : "01-01-2000" pour l'attribut Jour.
- Contre-exemple : "01-01-2000, 06-06-2006" pour l'attribut Jour.
- Non-respect de 1NF $\Rightarrow$ recherche parmi les données plus lente (il faut analyser le contenu des attributs)
- Pensez à un exemple ou cette retriction peut être contraignante.

NF2

- NF2 s'appuie sur la notion de dépendance fonctionnelle "complète" (ou "pleine").
- Une relation est NF2 si elle est NF1, et si tout attribut n'appartenant pas à la clé (primaire) dépend complètement de cette clé, c'est-à-dire si toutes les dépendances fonctionnelles s'appliquant sur R sont complètes.
- En gros une relation est NF2 si elle est NF1 et les attributs non-clé ne doivent pas deépendre que d'une partie de la clé mais de sa totalité.
- Remarque : si toutes les DF ont un seul attribut en partie gauche, ou si les clés sont atomiques, alors la relation est NF2.
- Non-respect de 2NF $\Rightarrow$ Redondance encombrant inutilement la mémoire.

NF2

Soit $Employe(idEmp, idService, nom, salaire, nomService)$ avec $[idEmp, idService]$ étant la clé de la table et on sait que $idEmp \rightarrow idService, nom, salaire, nomService$ Est-ce que cette relation est NF2 ?

NF3

- Une relation est NF3 si elle est NF2 et si les attributs non clés ne dépendent pas d'un attribut non clé.
- Une relation est NF3 si tous les attributs non clé sont en DFE directe avec la clé. Il ne doit pas y avoir de dépendance fonctionnelle entre des attributs non clé.
- Il peut rester des dépendances fonctionnelles entre attributs de la clé et entre attributs non clé vers des attributs de la clé
- Une relation est NF3 si, pour toute DF $X \rightarrow A$ s'appliquant sur R avec A non inclus dans X, soit X est clé de R, soit A fait partie d'une clé de R.
- Pour rendre la relation NF3, il faut donc éliminer les dépendances fonctionnelles transitives en plaçant certains attributs dans une autre relation.
- Non-respect de 3NF $\Rightarrow$ Redondance encombrant inutilement la mémoire.

NF3

Soit $R(Nom_F, Adresse_F, Produit, Prix)$ et $DF = \{Nom_F \rightarrow Adresse_F; Nom_F, Produit \rightarrow Prix\}$ La clé de R étant $[Nom_F, Produit]$. La relation est-elle NF3 ?

Théorème de Heat

Soit la relation $R\{A, B, C\}$ dans laquelle A, B et C sont des sous-ensembles d'attributs de R. Si R satisfait la DF $A \rightarrow B$, alors R est égale à la jointure de ses projections sur $\{A, B\}$ et $\{A, C\}$.

Décomposition

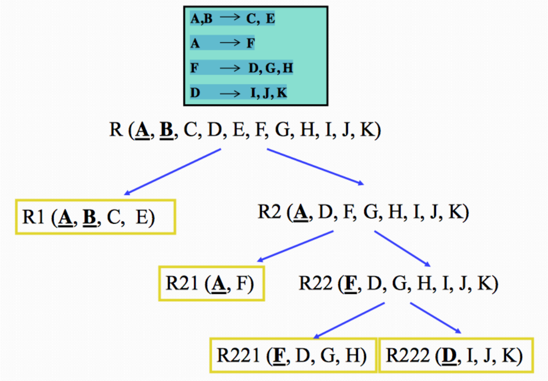
- Soit F un ensemble de DF définies sur l'ensemble des attributs de la relation
- Déterminer la couverture minimale MIN(F) de F
- Construire la relation universelle R, relation composée de tous les attributs
- Déterminer la clé primaire de R à partir de MIN(F)
- Pour chaque DF, tant que la DF ne contient pas tous les attributs de la relation, décomposer la relation en deux nouvelles relations en utilisant le théorème de Heat :
  Si la relation $R(A, B, C)$ avec la DF $B \rightarrow C$ n'est pas en NF3, elle sera décomposée en $R(A, B)$ set $R2(B, C)$
- Appliquer ce processus de décomposition sur les relations jusqu'à l'obtention de relations en NF3. La décomposition peut être représenté sous forme d'arbre, les feuilles de l'arbre constituent les relations de la base.

Exercice : Decomposer la relation pour qu'elle soit en NF3

Relation universelle R : $R(A, B, C, D, E, F, G, H, I, J, K)$

Couverture minimale : $F = \{AB \rightarrow CE; A \rightarrow F; F \rightarrow DGH; D \rightarrow IJK\}$

Correction


A,B → C, E
A → F
F → D, G, H
D → I, J, K
R (A, B, C, D, E, F, G, H, I, J, K)
R1 (A, B, C, E)
R2 (A, D, F, G, H, I, J, K)
R21 (A, F)
R22 (F, D, G, H, I, J, K)
R221 (F, D, G, H)
R222 (D, I, J, K)