5.5 Décompositions d'un schéma : propriétés SPI et SPD

A nouveau l'exemple de mauvais schéma, donnant lieu à des redondances et des anomalies, déjà consideré :

Relation FOURNISSEUR

Nom	Adr	$Produit$	$Prix$
n1	a1	i1	p1
n1	a1	i2	p2
n2	a2	i2	p3
n2	a2	i3	p4

$$F = Nom \rightarrow Adr,\ Nom\ Produit \rightarrow Prix$$

Il faudrais le remplacer par un autre schéma. Comment ?

Une solution possible : décomposer le schéma de relation en 2 ou plusieurs sous-schémas. Par ex. décomposer :

$$\text{FOURNISSEUR}(Nom, Adr, Produit, Prix)$$

en :

$$\text{NA}(Nom, Adr) \text{ et } \text{NIP}(Nom, Produit, Prix)$$

Contenu de la table NA : $\pi_{Nom,Adr}(\text{FOURNISSEUR})$.
Contenu de la table NIP : $\pi_{Nom,Produit,Prix}(\text{FOURNISSEUR})$.

Ceci est un exemple de la façon générale de décomposer une table de schéma S en n tables de schémas $S_1, \cdots, S_n$ où $(S_1 \cup \cdots \cup S_n) = S$:
on projette le contenu de la table decomposée sur les sous-schémas.

- **Toute décomposition possible est OK** ?
- Un des critères qu'une bonne décomposition doit respecter : il faut pouvoir reconstruire, par jointure, la relation de départ.
- Dans notre exemple :
  $\pi_{Nom,Adr}(FOURNISSEUR) \bowtie \pi_{Nom,Produit,Prix}(FOURNISSEUR) = FOURNISSEUR$? OUI.
- Toujours vrai, peu importe la décomposition ? NON. Contre-ex : Table R :

A	B	C
a1	b1	c1
a2	b1	c2

Décomposition en R1(A,B) et R2(B,C) (contenus=projections de R).
$\langle a1, b1, c2\rangle \in (\pi_{A,B}(R) \bowtie \pi_{B,C}(R))$ mais $\langle a1, b1, c2\rangle \notin$ R !
Cette décomposition fait perdre l'information NEGATIVE :
"$\langle a1, b1, c2\rangle$ **n'est pas** une ligne de R".

- Soit F un ensemble de DF. Une décomposition d'un schéma S d'une table R en $S_1, \cdots, S_n$ (où $(S_1 \cup \cdots \cup S_n) = S$) est dite sans perte d'information (SPI) par rapport à F ssi, qque soit le contenu de la table de nom R :

$$(\pi_{S_1}(R) \bowtie \cdots \bowtie \pi_{S_n}(R)) = R$$

- *N.B.* :

$$R \subseteq (\pi_{S_1}(R) \bowtie \cdots \bowtie \pi_{S_n}(R))$$

est toujours vrai, la $\subseteq$ réciproque non (contre-exemple précédant).

Rôle de F ??

$$\Longrightarrow$$

Pour le schéma de table de l'example précédant, on n'aurait pas pu avoir perte d'information négative si $B \rightarrow C$ ou $B \rightarrow A$:
on obtient $\langle a1, b1, c2 \rangle \in (\pi_{A,B}(R) \bowtie \pi_{B,C}(R))$ mais $\langle a1, b1, c2 \rangle \notin R$ car R ne satisfait pas $B \rightarrow C$ et ne satisfait pas $B \rightarrow A$

Soit $S = \{A, B, C\}$ un schéma de table et soit F un ensemble de DF sur S.
Si F implique $B \rightarrow C$ ou $B \rightarrow A$, alors la déc. de S en $\{A, B\}$ et $\{B, C\}$ est forcement SPI par rapport à F.

Remarque :
$B \rightarrow A$= $(\{A, B\} \cap \{B, C\} \rightarrow (\{A, B\} \setminus \{B, C\}$
$B \rightarrow C$= $(\{A, B\} \cap \{B, C\} \rightarrow (\{B, C\} \setminus \{A, B\}$

Théorème : Une déc. de S en red 2 sous-schémas S_1 et S_2 est SPI par rapport à un ensemble de dépendances fonctionnelles F ssi F implique

$$(S_1 \cap S_2) \rightarrow (S_1 \setminus S_2)$$

ou

$$(S_1 \cap S_2) \rightarrow (S_2 \setminus S_1)$$

NB : ce théorème donne un algo pour tester si une décomposition en 2 est SPI par rapport à F.

Et si on veut décomposer en n tables où $n > 2$? Algo qui teste si une déc. est SPI : chase. **A voir en TD**.

Critère 1 qu'une décomposition doit respecter : être SPI par rapport aux DF.

- SPI : seul critère à souhaiter pour une déc. d ?
- **Exemple**. V : ville, R : rue, C : Code Postal. Schéma : INFOS(R,C,V).
  $F = \{VR \to C,\ C \to V\}$.
  Déc. d en INFO1(RC) et INFO2(CV) :

	INFOS	
R	C	V
r1	c1	v1
r1	c2	v2
r3	c3	v3

INFO1	
R	C
r1	c1
r1	c2
r3	c3

INFO2	
C	V
c1	v1
c2	v2
c3	v1

d est SPI car F implique ((RC) $\cap$ (CV)) $\to$ (CV $\setminus$ RC).

Mais que dévient $VR \to C$? Pas applicable.

Insertion de $\langle r1, c3\rangle$ dans INFO1 : OK par rapport aux dép de INFO1, mais : $\langle r1, c3, v1\rangle$ **et** $\langle r1, c1, v1\rangle \in$ INFO1 $\bowtie$ INFO2. Violation de $VR \to C$!

La décomposition l'exemple précédent ne fait pas perdre d'information mais elle fait perdre des dépendances fonctionnelles.

Pour savoir si une insertion préserve la contrainte $VR \to C$ il faudrait réfaire la jointure INFO1 $\bowtie$ INFO2 !

On aimerait travailler sans besoin de faire des $\bowtie$ chaque fois qu'on insère.

Formalisons : $\Rightarrow$

Déf. La *fermeture* d'un ensemble de DF F, noté F^+, est l'ensemble de *toutes* les dép. impliquées par F.

Déf. La *projection* d'un ensemble de dép. F su'un ensemble d'attributs X est l'ensemble d'éléments de F^+ (pas seulement de F !) comportant seulement des attributs de X. Notation : F_X.

Déf. Une décomposition d de S en $S_1 \cdots S_n$ est sans perte de dépendances (SPD) par rapport à un ensemble de dép. F ssi

$$(F_{S_1} \cup \cdots \cup F_{S_n})^+ = F^+$$

NB : $\subseteq$ est banal. C'est l'inclusion reciproque qui compte.
(Le pb. crucial est : risque-t-on de perdre des dépendances impliquées par F, en projettant sur les sous-schémas ?)

Exemple de calcul de F_{S_i}. Soit $S = \{A, B, C\}$, $S_1 = \{A, C\}$

$F = \{A \to B, B \to A, B \to C, C \to B\}$.

$F^+ = \{A \to A,\ A \to B,\ A \to C,$
$B \to A,\ B \to B,\ B \to C,$
$C \to A,\ C \to B,\ C \to C, ...\}$

F_{S_1} contient $A \to C$ et $C \to A$ (même si $C \to A \notin F$!) et toutes les dépendances impliquées par ces deux la.

Attention : Oublier que $A \to C$ et $C \to A$ sont dans F_{S_1} est un exemple typique d'erreur !

Exemple de déc. Soit $S = \{A, B, C\}$, $S_1 = \{A, B\}$, $S_2 = \{B, C\}$, $F = \{A \rightarrow B, B \rightarrow C, C \rightarrow A\}$. Soit d une décomposition de S en S_1 et S_2.

SPD ? Clair que l'on retrouve $A \rightarrow B$ et $B \rightarrow C$. Mais $C \rightarrow A$??

$B \rightarrow A \in F_{S_1}$

$C \rightarrow B \in F_{S_2}$

$\{C \rightarrow B,\ B \rightarrow A\}\ \subset\ (F_{S_1} \cup F_{S_2})$

$C \rightarrow A \in (F_{S_1} \cup F_{S_2})^+$

Donc $F \subseteq (F_{S_1} \cup F_{S_2})^+$.

Donc $F^+ \subseteq (F_{S_1} \cup F_{S_2})^+$.

OUI, SPD !

- Critère 2 qu'une décomposition doit respecter : être SPD par rapport aux dépéndances fonctionnelles.

- Une déc. d peut être SPI sans être SPD : ex. vu.

- Une déc. d peut être SPD sans être SPI. Etudier la déc. de :

A	B	C	D
a	b	c1	d1
a1	b1	c	d

$$F = \{A \to B, C \to D\}$$

  en AB et CD.

- Question. Existe-t-il un algo qui <u>engendre</u> une décomposition qui est forcement SPI **et** SPD par rapport aux DF ? **OUI**.

Préliminaires à la déf. de l'algo qui engendre une décomposition SPI **et** SPD par rapport à un ensemble de dépendances fonctionnelles F

Soit $X \to Y$ une dép. fonct. et soit F un ensemble de dép. fonct..

DEFINITIONS

- $X \to Y$ est réduite à droite si Y est un singleton.
- $X \to Y$ est réduite à gauche (par rapport à F) s'il n'existe pas de X' strictement inclus dans X t.q. F implique $X' \to Y$.
- F est redondant s'il existe une dép. fonct. $Z \to W \in F$ t.q. $F \setminus \{Z \to W\}$ implique $Z \to W$.
- F est réduit si
  1. F n'est pas redondant
  2. $\forall\ f \in F$, f est réduite à gauche **et** à droite.

Exemple.

Soit F=

$$\{AB \to C, B \to A, A \to D, D \to C\}$$

1. Les dép. de F sont réduites à droite, mais ne sont pas réduites à gauche. On a : $AB \to C \in F$, mais F implique $B \to C$.
   (Calculer $\{B\}^+_F$!) L'ensemble $\{F\}$ est strictement inclus dans $\{A, B\}$.

2. F est redondant, car $F \setminus \{AB \to C\}$ implique $AB \to C$.
   (Calculer $\{AB\}^+_{F\setminus\{AB\to C\}}$!)

- **Définition**

  Soit F un ensemble de dép. fonct. Une couverture de F est un ensemble G de dép. fonct. tq. F est équivalent à G (c.à.d. : $F^+ = G^+$). Une couverture minimale de F est un ensemble G de dép. fonct. tq. G est réduit et F est équivalent à G.

- **Exemple très simple**

$$F = \{A \rightarrow BC, AC \rightarrow D, A \rightarrow D\}$$

  F n'est pas réduit : $A \rightarrow BC$ n'est pas réduite à droite, $AC \rightarrow D$ n'est pas réduite à gauche, et F est redondant, car $F \setminus \{A \rightarrow D\}$ implique $A \rightarrow D$.

  Soit :

$$G = \{A \rightarrow B, A \rightarrow C, A \rightarrow D\}$$

  G est une couverture minimale de F (développement au tableau).

- Calcul mécanique d'une couverture minimale pour un ensemble de dép. fonct. quelconque ?

Algorithme de Calcul d'une Couverture Minimale

Entrée : Un ensemble F de dép. fonct.

Sortie : Une couverture minimale G de F

1. *Initialisation* : $G{:=}F$;
2. **Faire**, dans l'ordre :
   (a) Remplaçer toute $X \rightarrow A_1, \cdots, A_n$ de F tq $n > 1$ par les dép. $X \rightarrow A_1, \ \cdots \ , X \rightarrow A_n$.
   G := l'ensemble de dép. ainsi obtenu;
   (b) Remplacer toute $X \rightarrow Y \in G$ par $X' \rightarrow Y$ où X' est un sous-ensemble minimal de X tq G implique $X' \rightarrow Y$.
   G := l'ensemble de dép. ainsi obtenu;
   (c) Tant que il existe $X \rightarrow A \in G$ tq $G \setminus \{X \rightarrow A\}$ implique $X \rightarrow A$,
   faire $G := G \setminus \{X \rightarrow A\}$;
3. **Retourner** G

Dans l'algorithme de calcul de la couverture minimale de F :

- Dans l'étape 2b, comment trouver un sous-ensemble minimal X' de X tq G implique $X' \to Y$?
  On peut calculer ${E^+}_G$ pour tout sous-ensemble strict E de X, en partant des sous-ensembles les + petits.

- Dans l'étape 2c, comment tester s'il existe $X \to A \in G$ tq $G \setminus \{X \to A\}$ implique $X \to A$?
  On peut calculer ${X^+}_{G\setminus\{X\to A\}}$ pour toute $X \to A \in G$.

Attention ! Il faut faire les étapes 2a et 2b avant l'étape 2c !!

Exemple

On fait tourner l'algo de calcul d'une couverture minimale sur :
$F_1 = \{ABCD \to E, E \to D, A \to B, AC \to D\}$

Mauvaise façon de faire

1. On fait l'étape 2c tout de suite : rien est redondant !
2. Etape 2a : rien à faire.
3. Etape 2b : on remplace $ABCD \to E$ par $AC \to E$.

Fini ! On a obtenu $F_2 = \{AC \to E, E \to D, A \to B, AC \to D\}$ qui est équivalent à F_1, mais qui contient la dépendance redondante $AC \to D$! Une couverture minimale aurait été

$$F_3 = \{AC \to E, E \to D, A \to B, \}$$

On fait à nouveau tourner l'algo de calcul d'une couverture minimale sur : $F_1 = \{ABCD \to E, E \to D, A \to B, AC \to D\}$, mais avec le bon ordre des étapes.

1. On fait l'étape 2a : rien à faire
2. Etape 2a : on obtient $F_1' = \{AC \to E, E \to D, A \to B, AC \to D\}$
3. Etape 2c : on élimine $AC \to D$.
4. Fini, avec le bon ensemble $F3$ comme résultat !

On est prêts à donner l'algo pour le calcul d'une déc. qui soit SPI et SPD $\rightsquigarrow$ page suivante.

Algorithme qui génére une dec. SPI et SPD

Entrée : Un schéma de relation S, un ensemble F de dép. fonct. sur S.

Sortie : Une décomposition d de S qui est SPI et SPD par rapport à F.

1. Choisir un ensemble de DF $F' = \{X_1 \rightarrow A_1, \cdots, X_n \rightarrow A_n\}$ tq F' est une couverture minimale de F;
2. Pour chaque $f_i = X \rightarrow A_j \in F'$, créer un schéma $S_i = XA_j$ et poser $d=$ l'ensemble de ces S_i.
3. Si aucune des clés de S (calculées grâce à F') n'est contenue dans un élément de d, modifier d en y ajoutant une clé Y comme élément.
4. S'il existe 2 éléments S_i et S_j de d tels que $S_i \subset S_j$ supprimer S_i, afin d'obtenir la valeur courante de d.
5. Si on a des éléments de d de la forme $XA_1, \cdots, XA_k$ obtenus car $X \rightarrow A_1, \cdots, X \rightarrow A_k$ sont des éléments de F', les remplaçer par l'unique (sous)schéma $XA_1 ... A_k$ afin d'obtenir la valeur finale de d.

N.B Si plusieurs choix de couvertures minimales et de clés, possibilité de plusieurs décompositions.

Exemple d'application.

Soit $S = ABCDE$. Soit

$$F_1 = \{ABCD \to E, E \to D, A \to B, AC \to D\}$$

Application de l'algo de décomposition SPI et SPD : au tableau.