

TD 4 SGBD, L3 2009-2010

1 Décompositions Sans Perte d'Information (SPI)

1.1 Le problème

La table *Planning_Examens* a le schéma S suivant :

Etudiant Examen Heure Date

qui est équipé de l'ensemble de dépendances fonctionnelles suivant :

$$\{Etudiant\ Examen \rightarrow Heure\ Date, \quad Etudiant\ Heure\ Date \rightarrow Examen\}$$

On décompose S en $S_1 = Etudiant\ Examen$ et $S_2 = Etudiant\ Heure\ Date$. Donnez un exemple de valeurs pour la table initiale *Planning_Examens* qui montre que cette décomposition est problématique.

1.2 Algorithme *Chase* : Complément de cours

En cours on a vu un algorithme simple permettant de tester si la décomposition d'un schéma de relation S en deux sous-schémas S_1 et S_2 est SPI (*sans perte d'information*) par rapport à un ensemble de dépendances F . L'algorithme suivant, dit *chase*, est plus général : il permet de tester si la décomposition d'un schéma de relation S en n sous-schémas $S_1, \dots, S_n$ est SPI par rapport à un ensemble de dépendances F , quand où n est n'importe quel entier positif ≥ 2 . Pour le décrire, nous utilisons les conventions suivantes :

- Nous supposons que les attributs de S sont $A_1, \dots, A_k$, où $k \geq 2$;
- Nous associons à chaque attribut A_i une et une seule *constante* a_i
- Les expressions de la forme x^i_j sont des *variables*

Entrées : Le schéma S , les schémas de sa décomposition : $S_1, \dots, S_n$, l'ensemble de dépendances fonctionnelles F . Sans perte de généralité, on peut supposer que toute dépendance de F est réduite à droite.

Sortie : "OUI", si la décomposition est SPI par rapport à F , "NON" sinon.

1. *Initialisation.*

- a) Construire une table dont les k colonnes sont $A_1, \dots, A_k$ et ayant n lignes. Chaque ligne j est étiquetée par S_j :

	A_1	$\dots$	A_i	$\dots$	A_n
S_1					
$\vdots$					
S_j					
$\vdots$					
S_n					

- b) Dans chaque case de coordonnées i, j (i est la colonne, j est la ligne), placer la constante a_i si l'attribut $A_i \in S_j$, placer la variable x^i_j sinon.

2. **Tant que** ou bien on obtient au moins une ligne ne contenant aucune variable, ou bien la table reste stable, **faire** :
 pour chaque dépendance $X \rightarrow A_i$ de F ,
 pour chaque couple de lignes,
 si ces deux lignes ont les mêmes valeurs pour X mais des valeurs différents pour A_i , alors modifier ces lignes de façon qu'elles aient la même valeur pour A_i aussi.
 Pour effectuer cette modification, si un ligne contient une variable pour la colonne A_i et l'autre la constante a_i , remplacer la variable par la constante, tandis que si les deux lignes contiennent deux variables pour la colonne A_i , remplacer une des 2 variables (peu importe laquelle) par l'autre.
3. Si table finale contient au moins une ligne ne contenant aucune variable, la sortie est "OUI", sinon c'est "NON".

Soit $S = ABCDE$ et soit F l'ensemble de dépendances :

$$\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$$

Appliquer l'algorithme *chase* pour déterminer si la décomposition AD, AB, BE, CDE, AE est SPI par rapport à F .

1.3 Des exercices sur SPI

1. Soit $S = ABCD$, $F = \{A \rightarrow B, B \rightarrow C\}$ et la décomposition AB, BC, CD . Déterminer si cette décomposition est SPI ou pas par rapport à F . Si ce ne l'est pas, fournir un contre-exemple.
2. Exactement les mêmes questions qu'avant, mais en prenant, au lieu de F , l'ensemble de dépendances $F' = \{A \rightarrow B, B \rightarrow C, C \rightarrow D\}$.
3. Soit $S = ABCD$, et soit F l'ensemble de dépendances $\{AB \rightarrow C, BC \rightarrow D\}$. Soit la décomposition ABC, BCD . Mêmes questions qu'avant, mais, attention, ici, vous avez le choix entre **deux** algorithmes possible pour tester SPI. Pourquoi ?

2 Décompositions SPD

2.1 Le problème

Soit R une table ayant comme schéma : *Num_Vol Date Porte Heure Destination* et comme ensemble de dépendances :

$$F = \{Num_Vol \text{ Date} \rightarrow Porte, Num_Vol \rightarrow Destination \text{ Heure} \\ Date \text{ Porte} \text{ Heure} \rightarrow Num_Vol\}$$

on décompose R en $R_1 = Num_Vol \text{ Destination} \text{ Heure}$ et $R_2 = Num_Vol \text{ Porte} \text{ Date}$. Indiquer les ensembles F_{R_1} et F_{R_2} des dépendances applicables aux deux nouveau schémas.

Donner une suite d'insertions dans la base ainsi décomposée qui respecte F_{R_1} et F_{R_2} mais telle que, si on fait la jointure des relations obtenues, on obtient un n -uplet qui viole au moins une des contraintes établies en origine par F (laquelle ?).

2.2 Questions

En appliquant les définitions du cours, tester si les deux décompositions d_1 et d_2 suivantes sont SPD (*sans perte de dépendances*) ou pas :

1. Le schéma à décomposer est $ABCD$, $F = \{A \rightarrow C, D \rightarrow C, BD \rightarrow A\}$ et d_1 est la décomposition en 3 schémas : AB, ACD et BCD .
2. Le schéma à décomposer est $ABCD$, $F = \{A \rightarrow B, B \rightarrow CD, A \rightarrow D\}$ et d_1 est la décomposition en 2 schémas : AB et BCD .

Corrigé

Question de la Section 1.1

Ce n'est pas SPI, comme illustré, par exemple, par la table suivante :

Marie	SGBD	09.00	7/01/10
Marie	Algo	08.00	15/01/10

La jointure naturelle entre $\pi_{Et\ Ex}(Planning_Examens)$ et $\pi_{Et\ H\ D}(Planning_Examens)$ donne la ligne Marie SGBD 08.00 15/01/10 qui n'existait pas dans la table.

Question de la Section 1.2

SPI, puisque on a une ligne de constantes pour le sous-schéma BE.

Questions de la Section 1.3

1. Pas SPI. La table finale que l'on obtient est un contre-exemple : si l'on la voit comme une relation r satisfaisant F , on constate que la jointure naturelle entre $\pi_{AB}(r)$, $\pi_{AB}(r)$ et $\pi_{AB}(r)$ est différente de r . C'est toujours vrai que la table finale obtenue par *chase* donne un contre-exemple à SPI en cas de réponse négative, à condition de faire l'hypothèse que les domaines des attributs sont *assez grands*. Par exemple, ici, il faut supposer que $|Dom(A)|_g \geq 3$, $|Dom(B)|_g \geq 2$, $|Dom(D)|_g \geq 3$.
2. Cette fois ci on obtient une ligne de constantes pour AB , donc SPI.
3. Avec *chase*, on obtient SPI. On peut aussi utiliser l'algorithme qui teste si une décomposition en 2 est SPI donné en cours.

Questions de la Section 2.1

Puisque :

$F = \{Num_Vol \rightarrow Date, Num_Vol \rightarrow Destination, Heure$

$Date \rightarrow Porte, Date \rightarrow Num_Vol\}$,

$R_1 = Num_Vol \rightarrow Destination, Heure,$

$R_2 = Num_Vol \rightarrow Porte, Date\}$

on a $F_{R_1} = \{Num_Vol \rightarrow Destination, Heure\}$ et $F_{R_2} = \{Num_Vol \rightarrow Date, Porte\}$. La dépendance $Date \rightarrow Porte, Date \rightarrow Num_Vol$ est perdue. Supposons que R_1 et R_2 soient vides. Si on insère dans R_1 : AF15, Rome, 08.00 et AZ112, Madrid, 08.00

alors on respecte bien F_{R_1} . Si on insère dans R_2 :

AF15, 07/01/09, B1 et AZ112, 07/01/09, B1 on respecte bien F_{R_2} (remarquons que l'on ne dit pas que l'heure est la même, pour ces deux lignes de R_2). Mais, par jointure on obtient :

AF15, Rome, 08.00, 07/01/09, B1 et AZ112, Madrid, 08.00, 07/01/09, B1.

Ces deux quadruplets ont les mêmes valeurs pour Heure, Date, Porte mais valeurs différentes pour Num_Vol. Ainsi, $Date \rightarrow Porte, Date \rightarrow Num_Vol$ est violée.

Questions de la Section 2.2

1. Pas SPD, car on perd $BD \rightarrow A$
2. SPD, car $A \rightarrow D$ est récupérée par implication.