

# FONDEMENTS DES BASES DE DONNÉES

## Inférence de dépendances fonctionnelles : Fermeture d'un ensemble d'attributs

Équipe pédagogique BD



## Problème d'inférence de DF

Le système d'Armstrong permet de systématiser la recherche de toutes les DF impliquées par d'autres DF.

## Problème d'inférence des DF

Soit  $F$  un ensemble de DF, et  $f$  une DF, a-t-on  $F \models f$  ?

- ▶ La résolution du problème d'inférence est “facile” pour les DF,
- ▶ linéaire en la taille de  $F$  et de  $f$ .
- ▶ Le concept fondamental utilisé dans cet algorithme est celui de **fermeture**.

## Fermeture d'un ensemble de DFs

Soit  $F$  un ensemble de DF, on note  $F^+$  la **fermeture de  $F$** , l'ensemble de toutes les Dfs logiquement impliquées par  $F$  :

$$F^+ = \{f \mid F \models f\}$$

# Fermeture d'un ensemble d'attributs

## Fermeture d'un ensemble d'attributs

$X$  est un ensemble d'attribut et  $F$  un ensemble de DF, on note  $X^+$  la fermeture de  $X$  par rapport à  $F$  l'ensemble de tous les attributs qu'on peut "déduire" de  $X$  par des dépendances fonctionnelles :

$$X^+ = \{A \mid F \models X \rightarrow A\}$$

D'après la définition de la fermeture de  $F$ , on a de façon équivalente :

$$X^+ = \{A \mid X \rightarrow A \in F^+\}$$

Il faut aussi tenir compte de *toutes* les DFs qui sont dérivables à partir de  $F$ .

## Définition alternative de la notion de clé

- ▶ une clé de  $R$  est un ensemble  $X \subseteq R$  tels que  $R : X \rightarrow R$
- ▶  $X$  est clé d'un schéma  $R$  ssi  $X^+ = R$

## Lemme

Soient  $F$  un ensemble de DF et  $X \rightarrow Y$  une DF :

$$F \models X \rightarrow Y \text{ ssi } Y \subseteq X^+$$

- ▶ Ainsi, pour tester si on a  $F \models X \rightarrow Y$ , on calcule  $X^+$  et on vérifie si  $Y \subseteq X^+$
- ▶ On va utiliser un algorithme pour calculer simplement  $X^+$
- ▶ On obtient ainsi un algorithme pour décide de l'implication logique des DFs<sup>1</sup>.

---

1. L'algorithme permet de dire si  $F \models X \rightarrow Y$  ou  $F \not\models X \rightarrow Y$

## Algorithme : fermeture d'un ensemble d'attributs

**Data:**  $F$  un ensemble de DF,  $X$  un ensemble d'attributs.

**Result:**  $X^+$ , la fermeture de  $X$  par  $F$ .

$unused := F$

$closure := X$

**repeat**

$closure' := closure$

**if**  $W \rightarrow Z \in unused$  and  $W \subseteq closure$  **then**

$unused := unused - \{W \rightarrow Z\}$

$closure := closure \cup Z$

**end**

**until**  $closure' = closure$ ;

**retourner**  $closure$

- ▶ L'algorithme permet de vérifier si un ensemble de DF implique logiquement une dépendance d'après le lemme vu avant.
- ▶ Pour tester l'implication d'un *ensemble de dépendances*, il suffit de tester l'implication de *chaque* dépendance.

Combien de fois (au plus) teste-t-on  $W \subseteq \text{closure}$   
en fonction de  $|F| = n$ ?

## Un algorithme linéaire

Amélioration pour obtenir un temps linéaire en la taille de  $|F|$  *faire en sorte de ne se servir d'une DF qu'une seule fois* quand c'est nécessaire :

- ▶ Pour chaque  $X \rightarrow Y \in F$  non utilisée, il faut stocker le *nombre* d'attributs de  $X$  non encore dans *closure*.
- ▶ Pour le faire efficacement, il faut maintenir à jour une liste pour chaque attribut  $A$  des DFs de  $F$  non utilisées pour lesquelles  $A$  apparaît en partie gauche.

# Algorithme : fermeture linéaire

```
for  $W \rightarrow Z \in F$  do  
   $count[W \rightarrow Z] := |W|$   
  for  $A \in W$  do  
     $list[A] := list[A] \cup W \rightarrow Z$   
  end  
end  
 $closure := X, update := X$   
while  $update \neq \emptyset$  do  
  Choose  $A \in update$   
   $update := update \setminus \{A\}$   
  for  $W \rightarrow Z \in list[A]$  do  
     $count[W \rightarrow Z] := count[W \rightarrow Z] - 1$   
    if  $count[W \rightarrow Z] = 0$  then  
       $update := update \cup (Z \setminus closure)$   
       $closure := closure \cup Z$   
    end  
  end  
end  
return  $closure$ 
```

## Exemple : $AE^+$

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

## Exemple : $AE^+$

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

### Initialisation

$List[A] = \{A \rightarrow D; AB \rightarrow E\}$	$count[A \rightarrow D] = 1$
$List[B] = \{AB \rightarrow E; BI \rightarrow E\}$	$count[AB \rightarrow E] = 2$
$List[C] = \{CD \rightarrow I\}$	$count[BI \rightarrow E] = 2$
$List[D] = \{CD \rightarrow I\}$	$count[CD \rightarrow I] = 2$
$List[E] = \{E \rightarrow C\}$	$count[E \rightarrow C] = 1$
$List[I] = \{BI \rightarrow E\}$	

$update = AE$

$closure = AE$

## Exemple : $AE^+$

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

Choose A

$update \setminus A;$

A partir de  $List[A] = \{A \rightarrow D; AB \rightarrow E\} :$

$count[A \rightarrow D] = count[A \rightarrow D] - 1 = 0$

$\rightsquigarrow update \cup D \setminus closure; closure \cup D;$

$count[AB \rightarrow E] = count[AB \rightarrow E] - 1 = 1;$

$$List[A] = \{A \rightarrow D; AB \rightarrow E\} \quad \begin{array}{l} count[A \rightarrow D] = 0 \\ count[AB \rightarrow E] = 1 \end{array}$$

$update = DE$

$closure = ADE$

## Exemple : $AE^+$

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

### Choose E

$update \setminus E;$

A partir de  $List[E] = \{E \rightarrow C\} :$

$$count[E \rightarrow C] = count[E \rightarrow C] - 1 = 0$$

$\rightsquigarrow update \cup C \setminus closure; closure \cup C;$

$$List[E] = \{E \rightarrow C\} \quad count[E \rightarrow C] = 0$$

$update = CD$

$closure = ACDE$

## Exemple : $AE^+$

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

Choose C

*update* \ C ;

A partir de  $List[C] = \{CD \rightarrow I\}$  :

$$count[CD \rightarrow I] = count[CD \rightarrow I] - 1 = 1$$

$$List[C] = \{CD \rightarrow I\}$$

$$count[CD \rightarrow I] = 1$$

*update* = D

*closure* = ACDE

## Exemple : $AE^+$

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

### Choose D

$update \setminus D;$

A partir de  $List[D] = \{CD \rightarrow I\} :$

$$count[CD \rightarrow I] = count[CD \rightarrow I] - 1 = 0$$

$\rightsquigarrow update \cup I \setminus closure; closure \cup I;$

$$List[D] = \{CD \rightarrow I\} \quad count[CD \rightarrow I] = 0$$

$update = I$

$closure = ACDEI$

## Exemple : $AE^+$

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

Choose I

$update \setminus I;$

A partir de  $List[I] = \{BI \rightarrow E\}$  :

$$count[BI \rightarrow E] = count[BI \rightarrow E] - 1 = 1$$

$$List[I] = \{BI \rightarrow E\}$$

$$count[BI \rightarrow E] = 1$$

$update = \emptyset$  **Condition d'arrêt**

$closure = ACDEI$

## Exercice

Soit l'ensemble  $F$  de DF suivant sur le schéma  $R = ABCDEFG$  :

- ▶  $A \rightarrow B$
- ▶  $A \rightarrow C$
- ▶  $A \rightarrow D$
- ▶  $CD \rightarrow E$
- ▶  $BE \rightarrow F$
- ▶  $ABE \rightarrow G$
- ▶  $EG \rightarrow ABD$
- ▶  $FG \rightarrow AE$
  
- ▶ Démontrer que  $F \models A \rightarrow F$  (avec fermeture).
- ▶ Démontrer que  $F \models A \rightarrow G$  (avec fermeture).
- ▶ Démontrer que  $BEF$  n'est pas une clé de  $R$
- ▶ Démontrer que  $CDE \rightarrow A$  n'est pas impliquée par  $F$

*Fin de la séance.*