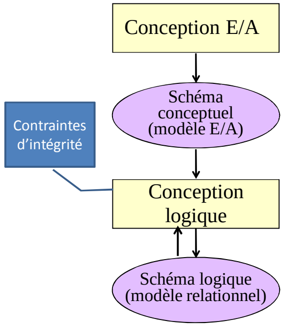
UFR 919 Ingénierie – module 3I009
cours 8 - Conception de bases de données

- Conception logique
- Dépendances fonctionnelles
- Décomposition de schémas et normalisation
- Formes normales
- Algorithme de décomposition

Étapes de conception



Conception "graphique" et conversion E/A-relationnel

Conception logique pour la validation formelle et l'amélioration du schéma relationnel

Objectifs de la conception logique

- Éviter des **incohérences** dans les données :

  Une personne n'a qu'une date de naissance, le prix d'un produit est unique, …. Coûteux à vérifier
- Éviter la **redondance** d'information :

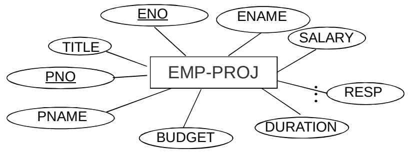
  La même information est stockée plusieurs fois

  Anomalies: insertion, suppression, modification
- Éviter les **valeurs nulles** :

  Difficiles à interpréter : inconnu, connu mais non disponible, inapplicable, Rend les jointures difficiles à spécifier
- Éviter les **jointures inutiles** :

  Améliorer les performances : la jointure est coûteuse

Pourquoi pas une seule relation ? (le cas extrême)



EMP-PROJ

ENO	ENAME	TITLE	SALARY	PNO	PNAME	BUDGET	DURATION	RESP
E1	J. Doe	Elect. Eng.	40000	P1	Instrumentation	150000	12	Manager
E2	M. Smith	Analyst	34000	P1	Instrumentation	150000	24	Analyst
E2	M. Smith	Analyst	34000	P2	Database Develop.	135000	6	Analyst
E3	A. Lee	Mech. Eng.	27000	P3	CAD/CAM	250000	10	Consultant
E3	A. Lee	Mech. Eng.	27000	P4	Maintenance	310000	48	Engineer
E4	J. Miller	Programmer	24000	P2	Database Develop.	135000	18	Programmer
E5	B. Casey	Syst. Anal.	34000	P2	Database Develop.	135000	24	Manager
E6	L. Chu	Elect. Eng.	40000	P4	Maintenance	310000	48	Manager
E7	R. Davis	Mech. Eng.	27000	P3	CAD/CAM	250000	36	Engineer
E8	J. Jones	Syst. Anal.	34000	P3	CAD/CAM	250000	40	Manager

Problèmes

Problème principal : Redondance d'information

les attributs TITLE, SALARY, BUDGET, ENAME… sont répétés pour chaque projet dans lequel travaille l'employé.

***Trop d'attributs pour une seule relat*ion**

Conséquence : Anomalies

Anomalie d'insertion : il est difficile/impossible d'insérer un nouveau projet tant qu'il n'a pas d'employé affecté (valeurs nulles)

Anomalie de suppression : si le *dernier employé* d'un projet est supprimé, le projet est automatiquement supprimé aussi. Pour éviter cela, il faut prévoir un traitement spécifique à l'effacement du dernier employé.

Anomalie de modification : si un attribut *d'un projet*, par ex. son budget, est modifié, *tous les n-uplets des employés du projet* doivent être modifiés (pas de on update cascade oracle => trigger)

Solution : ***couper la relation en plusieurs. Comment ? Ce cours et le suivant***

Plan

Comment « décrire » plus précisément et formellement les données stockées dans une base de données ?

⇨Dépendances fonctionnelles

Comment passer d'un schéma relationnel vers un autre schéma « équivalent » par rapport à cette description ?

⇨ Décomposition de schémas

Comment caractériser un « bon schéma » ?

⇨ Formes normales

Comment passer d'un schéma quelconque (universel) vers un « bon » schéma équivalent ?

⇨ Algorithme de décomposition

Conception de bases de données

- Conception logique
- Dépendances fonctionnelles
- Décomposition de schémas et normalisation
- Formes normales
- Algorithme de décomposition

Dépendances fonctionnelles

Préliminaire important

Une dépendance fonctionnelle est une contrainte d'intégrité. Par conséquent :

- **Elle est observée dans le monde réel**
- **Elle doit être maintenue par le système (donc de manière efficace)**
- **Elle fait partie du schéma de la base**
- **Elle ne peut pas être déduite des valeurs d'une instance particulière (ce n'est pas parce qu'une DF est satisfaite par une instance qu'elle existe dans le monde réel)**

Dépendances fonctionnelles

R(Etudiant, Prof, Cours, Salle, Heure)

On sait (on l'a observé sur le monde réel)

qu'un étudiant ne peut pas être dans deux salles en même temps

ou

si on connaît le nom de l'étudiant et l'heure on peut identifier la salle (si elle existe)

ou

si t est un n-uplet dans R, il n'y a pas un autre n-uplet t' où t.Etudiant=t'.Etudiant et t.Heure=t'.Heure et t.Salle <> t'.Salle

On note alors

Etudiant,Heure → Salle

Dépendances fonctionnelles

Les *dépendances fonctionnelles* sont un moyen pour exprimer des ***contraintes sémantiques*** sur les données :

R(Etudiant, Prof, Cours, Salle, Heure)

Etudiant, Heure → Salle : un étudiant ne peut pas être dans deux salles différentes à la même heure

Salle,Heure → Cours : il ne peut pas avoir deux cours différents en même temps dans la même salle

Cours → Prof, Salle : il y a exactement un prof et une salle pour chaque cours

Inférence : On peut montrer à partir des dépendances ci-dessus

- qu'un étudiant ne peut pas suivre deux cours en même temps : **Etudiant,Heure → Cours**
- que (Salle,Heure, Etudiant) est une clé de R.

Notations

Attributs : A, B, C, …
Ensembles d'attributs : X, Y, Z
U = univers des attributs; XY correspond à $X \cup Y$

Schémas de relations R, R' :
avec leurs attributs: R(X), R'(XY), …
Relations : r, r'
Relations avec leurs attributs: r(X), r'(XY), …

N-uplets : t, t'
Valeurs d'attributs de n-uplets: t.A, t.X

Dépendances fonctionnelles (DF): f, g, h
Ensembles de DF: F, G, H

Dépendances fonctionnelles

•Une dépendance fonctionnelle f est une expression

$$f: X \rightarrow Y$$

où X et Y sont des ensembles d'attributs (on dit que f est définie sur XY).

•Une *relation r(Z)* satisfait *la dépendance* fonctionnelle f: $X \rightarrow Y$ si

1. $XY \subseteq Z$ (elle contient tous les attributs dans XY) et
2. tous les n-uplets t et t' dans r(Z) qui ont les mêmes valeurs pour les attributs dans X, partagent également les mêmes valeurs pour les attributs Y :

$$\forall t,t' \in r: t.X = t'.X \Rightarrow t.Y = t'.Y.$$

Inférence de DFs

Soit F un *ensemble de dépendancesfonctionnelles* défini sur un ensemble d'attributs U :

- On dit que F implique (logiquement) la DF $X \to Y$ (avec $XY \subseteq U$) si *toutes les relations r(U)* qui satisfont toutes les dépendances fonctionnelles dans F, satisfont également $X \to Y$.
- On note alors :

$$F \models X \to Y \quad \text{(parfois } F \Rightarrow X \to Y\text{)}$$

Exemple

- Est-ce que $F \models A \to C$ si $F = \{A \to B, B \to C\}$?
- Problème d'inférence : Il faut montrer que toutes les relations r(A,B,C) qui satisfont F, satisfont également $A \to C$.
- Preuve logique :

On sait que r satisfait F ssi

$\forall t,t' \in r$: $t.A = t'.A \Rightarrow t.B = t'.B$ et

$t.B = t'.B \Rightarrow t.C = t'.C$

On peut conclure que (modus ponens)

$\forall t,t' \in r$: $t.A = t'.A \Rightarrow t.C = t'.C$

Surclés et clés

- Soit R(Z) un schéma de relation, F un ensemble de dépendances fonctionnelles sur Z, et X un sous-ensemble des attributs de R, $X \subseteq Z$:
  - X est une surclé de R avec F si $F \models X \to Z$.
  - une surclé X est appelé un clé de R, s'*il n'existe pas* de sous-ensemble stricte $Y \subset X$ *de* X qui est aussi une surclé.

Une clé de R est un ensemble minimal (-au sens de l'inclusion) qui détermine tous les autres

Remarques :

- Une *clé primaire* est une surclé (généralement une clé) qui peut être utilisée pour organiser *physiquement* les données (*organizing index* dans *create table*).
- On peut trouver toutes les surclés et clés d'une relation à partir d'un ensemble de DF par *inférence.*
- *Chaque clé est une surclé mais pas inversement.*
- *L'ensemble Z est une surclé triviale de R(Z).*
- *Si un attribut A est une surclé, il est forcément une clé.*
- *Quand est-ce que Z est une clé de R(Z) ?*

Problème d'inférence

- **Question :** Est-ce que A est une clé de R(ABC) avec $F = \{A \to B, B \to C\}$?
- Il faut montrer que $F \models A \to ABC$ **(inférence)**.
- Preuve logique :
  - On sait que r satisfait F ssi

    $\forall t,t' \in r$: $t.A = t'.A \Rightarrow t.B = t'.B$ *et*

    $t.B = t'.B \Rightarrow t.C = t'.C$
  - On peut conclure

    $\forall t,t' \in r$: $t.A = t'.A \Rightarrow t.A = t'.A \wedge t.B = t'.B \wedge t.C = t'.C$

Exemple

- **Question :**
  - Est-ce que A est une clé de R(AC) avec F = {A → B, B → C} ?
  - Ou : Est-ce que **F |= A → C ?**
- Attention : il *ne suffit pas* de regarder les dépendances dans F !
- Explication :
- Comme R ne contient pas l'attribut B, on pourrait conclure que A n'est pas une clé.
  - Mais : on peut montrer que **F |= A → C (inférence)** et ainsi que l'attribut A est une clé de R(AC) avec F = {A → B, B → C}

Il nous faut un moyen pratique de calculer les inférences :
•Fermeture d'un ensemble d'attribut
•Axiomes d'Armstrong

Inférence par le calcul de fermeture transitive d'un ensemble de DF

- La fermeture transitive (ou clôture) d'un ensemble de dépendances fonctionnelles F est l'ensemble de toutes les dépendances fonctionnelles *qu'on peut déduire de* F:

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

- Une relation r(Z) satisfait un ensemble de dépendances fonctionnelles F si elle satisfait toutes les dépendances fonctionnelles X → Y où
- XY ⊆ Z et
- X → Y est *dans la fermeture transitive de F*.

Fermeture transitive d'un ensemble de dépendances fonctionnelles

- F = {A → B, B → C}
- F^+ = {A→B, B→C, A→C, *A→A, A→AB, A→AC, A→ABC, AB→A, AB→AB, AB→AC, AB→ABC, ABC→A, ABC→AB, ABC→AC, AB→ABC, … C→C, BC→C*}

- La fermeture contient beaucoup de *solutions triviales* et peut être très longue à calculer. Mais pas nécessaire car ce dont on a besoin de savoir c'est si une DF f ∈ F^+

Inférence par le calcul de la fermeture d'un ensemble d'attributs

La *fermeture d'un ensemble d'attributs* X contient tous les attributs qui dépendent des attributs dans X:

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

Exemple avec F = {A → B, B → C}, R(AC)
$[A]^+_F$ = { A, B, C } : on peut conclure que
- A → C est dans la fermeture de F et
- A est un surclé (et clé) de R

On peut montrer facilement que

$$F \models X \rightarrow A \Leftrightarrow A \in [X]^+_F$$

Fermeture d'un ensemble d'attributs

function *ComputeX*$^+$(*X*, *F*)
begin
$X^+ := X$
while exists $(Y \rightarrow Z) \in F$ **tel que** $Y \subseteq X^+$ et $Z \not\subseteq X^+$
$X^+ := X^+ \cup Z$
$F := F - \{Y \rightarrow Z\}$
return(X^+)
end

Exemple de fermeture d'attributs

Soit *F* l'ensemble de DF sur R(A,B,C,D,E,G,H)
- $A \rightarrow B$
- $C \rightarrow DE$ ***Est-ce que CG est une clé de R ?***
- $EG \rightarrow H$

ComputeX$^+$({C, G}, *F*)
- Initialisation : $X^+ = X = \{C, G\}$
- Itération 1($C \rightarrow DE$): $X^+ = \{C, G, D, E\}$
- Itération 2 ($EG \rightarrow H$): $X^+ = \{C, G, D, E, H\}$
- Iteration 3 : on n'a plus de DF à appliquer

Conclusion ?

Calcul de clés : Exemple

R(Etudiant, Prof, Cours, Salle, Heure)
- Etudiant Heure $\rightarrow$ Salle
- Salle Heure $\rightarrow$ Cours
- Cours $\rightarrow$ Prof Salle
- Etudiant Salle $\rightarrow$ Heure

Les attributs de R qui ne sont jamais en partie droite d'une DF font partie de toutes les clés (ne peuvent être « obtenus » à partir des autres)

Comment trouver toutes les clés par rapport aux DF ?
- Quels attributs n'apparaissent jamais à droite d'une DF ? : Etudiant
- Est-ce que Etudiant est une clé ? : Non : Etudiant+ = Etudiant
- Quels attributs on essaie en premier à ajouter ? : Heure ou Salle
- Pourquoi ?
- Est-ce que (Etudiant,Heure) est une clé ?
  Oui : (Etudiant, Heure)+ = Etudiant Heure Salle Cours Prof
- Est-ce que (Etudiant ,Salle) est une clé ?
  Oui : (Etudiant ,Salle)+ = Etudiant Salle Heure Cours Prof
- On peut s'arrêter là ? Non : On peut aussi essayer Etudiant Cours ? ...

Équivalence et couverture minimale (1/2)

- Deux ensembles de DF différents peuvent exprimer les même contraintes:

$F = \{AB \rightarrow D, D \rightarrow C, C \rightarrow D, AB \rightarrow C\}$
$G = \{AB \rightarrow C, D \rightarrow C, C \rightarrow D\}$

- F et G sont *équivalents* (expriment les mêmes contraintes).
On note : $F \equiv G$
- G est plus « compacte »

Équivalence et couverture minimale (2/2)

F est équivalent à G ($F \equiv G$) si $F^+ = G^+$

pas de panique, on n'a pas besoin de calculer F^+ ni G^+ car on peut montrer que

$F \equiv G$ ssi $\forall f \in F$, $G \models f$ et $\forall g \in G$, $F \models g$

F est un *ensemble minimal* de DF ssi

1. Toute DF de F est sous la forme $X \rightarrow A$ (un seul attribut à droite, forme canonique)
2. F *ne contient pas* de DF $X \rightarrow A$ qui
   - *Est redondante* : $F^+ = (F - \{ X \rightarrow A \})^+$ *ou*
   - *Contient des attributs en trop à gauche :* $X' \subset X$ et $F \models X' \rightarrow A \notin F^+$

F est une *couverture minimale* de G si

$F \equiv G$ et F est un *ensemble minimal* de DF.

Exemple de couverture minimale

$F=\{AB \rightarrow CD, ACE \rightarrow B, D \rightarrow C, C \rightarrow D, CD \rightarrow BE\}$

Forme canonique (un seul attribut sur le côté droit) :

$F_1=\{AB \rightarrow C, AB \rightarrow D, ACE \rightarrow B, D \rightarrow C, C \rightarrow D, CD \rightarrow B, CD \rightarrow E\}$

Test de redondance : on peut enlever

- $ACE \rightarrow B$
  - $C \rightarrow D, CD \rightarrow B \models C \rightarrow B$: on peut enlever $ACE \rightarrow B$
- et $AB \rightarrow C$ ou $AB \rightarrow D$:
  - $AB \rightarrow D, D \rightarrow C \models AB \rightarrow C$: on peut enlever $AB \rightarrow C$ ou
  - $AB \rightarrow C, C \rightarrow D \models AB \rightarrow D$: on peut enlever $AB \rightarrow D$

Exemple de couverture minimale

$F=\{AB \rightarrow CD, ACE \rightarrow B, D \rightarrow C, C \rightarrow D, CD \rightarrow BE\}$

Forme canonique (un seul attribut sur le côté droit) :

- $F_1=\{AB \rightarrow C, AB \rightarrow D, ACE \rightarrow B, D \rightarrow C, C \rightarrow D, CD \rightarrow B, CD \rightarrow E\}$

Test de redondance : on peut enlever $ACE \rightarrow B$ et ($AB \rightarrow C$ ou $AB \rightarrow D$) (deux solutions):

1. $F_2=\{AB \rightarrow C, D \rightarrow C, C \rightarrow D, CD \rightarrow B, CD \rightarrow E\}$

   Test de redondance des attributs à gauche :

- $[C]^+_{F2} = \{C, D, B, E\}$ et $[D]^+_{F2} = \{C, D, B, E\}$
- On peut enlever C ou D dans $CD \rightarrow B$ et $CD \rightarrow E$ (4 solutions)

1. $F_3=\{AB \rightarrow D, D \rightarrow C, C \rightarrow D, CD \rightarrow B, CD \rightarrow E\}$

   Test de redondance des attributs à gauche (idem; 4 solutions)

8 couvertures minimales

1. $F_{2.1}=\{AB \rightarrow C, D \rightarrow C, C \rightarrow D, C \rightarrow B, C \rightarrow E\}$
2. $F_{2.2}=\{AB \rightarrow C, D \rightarrow C, C \rightarrow D, C \rightarrow B, D \rightarrow E\}$
3. $F_{2.3}=\{AB \rightarrow C, D \rightarrow C, C \rightarrow D, D \rightarrow B, C \rightarrow E\}$
4. $F_{2.4}=\{AB \rightarrow C, D \rightarrow C, C \rightarrow D, D \rightarrow B, D \rightarrow E\}$
5. $F_{3.1}=\{AB \rightarrow D, D \rightarrow C, C \rightarrow D, C \rightarrow B, C \rightarrow E\}$
6. $F_{3.2}=\{AB \rightarrow D, D \rightarrow C, C \rightarrow D, C \rightarrow B, D \rightarrow E\}$
7. $F_{3.3}=\{AB \rightarrow D, D \rightarrow C, C \rightarrow D, D \rightarrow B, C \rightarrow E\}$
8. $F_{3.4}=\{AB \rightarrow D, D \rightarrow C, C \rightarrow D, D \rightarrow B, D \rightarrow E\}$

Inférence avec les axiomes d'Armstrong

Il est possible de montrer qu'un ensemble de dépendances fonctionnelles F implique une DF $X \rightarrow A$, grâce aux axiomes d'Armstrong, qui sont

- sains : si $F \vdash X \rightarrow A$ alors $F \models X \rightarrow A$ et
- complets: si $F \models X \rightarrow A$ alors $F \vdash X \rightarrow A$

Remarques :

- $F \vdash X \rightarrow A$ signifie que la DF $X \rightarrow A$ peut être déduite à partir de F et les axiomes d'Armstrong.
- F^+ = ensemble de DF qu'on peut déduire de F en appliquant (récursivement) les axiomes d'Armstrong

Axiomes d'Armstrong

Soient *X*, *Y* et *Z* des ensembles d'attributs du schéma de relation *R*.

Axiomes d'Armstrong :

- Augmentation: $\{X \rightarrow Y\} \Rightarrow \{XZ \rightarrow YZ\}$
- Transitivité: $\{X \rightarrow Y, Y \rightarrow Z\} \Rightarrow \{X \rightarrow Z\}$
- Réflexivité: $W \subseteq X \Rightarrow \{X \rightarrow W\}$

Règles additionnelles :

- Union: $\{X \rightarrow Y, X \rightarrow Z\} \Rightarrow (X \rightarrow YZ)$
- Décomposition: $\{X \rightarrow YZ\} \Rightarrow \{X \rightarrow Y, X \rightarrow Z\}$
- Pseudotransitivité: $\{X \rightarrow Y, WY \rightarrow Z\} \Rightarrow \{XW \rightarrow Z\}$

Axiomes d'Armstrong : exemple

- $F = \{A \rightarrow C, B \rightarrow D\}$
- On peut montrer que AB est une surclé de R=ABCD :

1. $AB \rightarrow ABC$ ($A \rightarrow C$ + augmentation)
2. $ABC \rightarrow ABCD$ ($B \rightarrow D$ + augmentation)
3. $AB \rightarrow ABCD$ (1.+2. + transitivité)