

# Plan

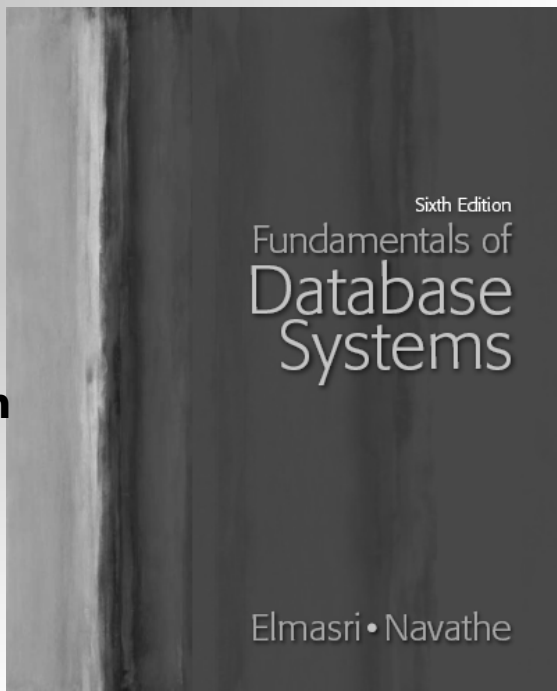
- Principes informels pour la conception des schémas relationnels
- Dépendances fonctionnelles
- Def des formes normales basées sur clés primaires
- Définitions générales de 2<sup>ème</sup> et 3<sup>ème</sup> forme normale
- Forme normale Boyce-Codd
- Règles d'inférence pour les DFs

L3 Informatique, Lille 1

BDD Cours 10

## Dépendances fonctionnelles et normalisation des BDRs

Traduction et adaptation:  
C.Kuttler



Addison-Wesley  
is an imprint of  
**PEARSON**

Copyright © 2011 Pearson Education, Inc. Publishing as Pearson Addison-Wesley

## Introduction

- La *justesse* de la conception d'un schéma relationnel peut être appréciée au:
  - Niveau logique ou conceptuel
  - Niveau de l'implémentation (ou stockage physique)
- Deux démarches pour la conception de bases:
  - Méthodologie de conception ascendante ou descendante

Addison-Wesley  
is an imprint of  
**PEARSON**

Copyright © 2011 Ramez Elmasri and Shamkant Navathe

## Principes informels pour la conception des schémas

- Mesures de la qualité de la conception:
  - Clarté de la sémantique des attributs
  - Réduire la redondance d'information dans les tuples
  - Réduire les valeurs NULL dans les tuples
  - Ne pas permettre la génération d'attributs parasites

Addison-Wesley  
is an imprint of  
**PEARSON**

Copyright © 2011 Ramez Elmasri and Shamkant Navathe

# Sémantique des attributs des relations

- Sémantique d'une relation
  - Sa signification résulte de l'interprétation des valeurs de ses attributs
- Une sémantique de relation plus facile à expliquer
  - Indique une meilleure conception du schéma

Addison-Wesley  
is an imprint of

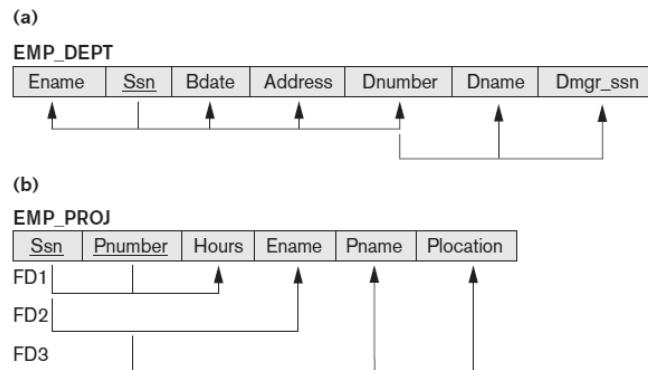
PEARSON

Copyright © 2011 Ramez Elmasri and Shamkant Navathe

## Exemple de non respect du Principe 1

Figure 15.3

Two relation schemas suffering from update anomalies. (a) EMP\_DEPT and (b) EMP\_PROJ.



Addison-Wesley  
is an imprint of

PEARSON

Copyright © 2011 Ramez Elmasri and Shamkant Navathe

## Principe 1

- Concevez un schéma de face à ce qu'il soit facile d'en expliquer la signification
- Ne combinez pas des attributs provenant d'entités et de liaisons de différents types en une même relation

Addison-Wesley  
is an imprint of

PEARSON

Copyright © 2011 Ramez Elmasri and Shamkant Navathe

## Informations redondantes et anomalies de mise à jour

- Regroupement d'attributs dans des schémas
  - Effet important sur l'espace de stockage
- De stocker des jointures naturelles de relations de bases peut mener à des **anomalies de mises à jour**
- Types d'anomalies de mise à jours:
  - Insertion
  - Suppression
  - Modification

Addison-Wesley  
is an imprint of

PEARSON

Copyright © 2011 Ramez Elmasri and Shamkant Navathe

## Principe 2

- Concevez les schémas des relations de telle sorte qu'il ne puisse pas survenir d'anomalies d'insertion, de suppression et de modifications des données
- S'il y a des anomalies:
  - Indiquez-les clairement
  - Assurez-vous que les programmes qui mettent à jour la base opèreront correctement

## Valeurs NULL dans les tupes

- Il arrive qu'on regroupe de nombreux attributs dans une relation “bien garnie”
  - Ceci peut entraîner un grand nombre de NULLs
- Problèmes résultant des NULLs
  - Gaspillage d'espace de stockage
  - Difficultés de compréhension

## Principe 3

- Evitez de placer dans une relation de base des attributs dont les valeurs sont susceptibles d'être souvent NULL
- Si les NULLs sont inévitables:
  - Faites en sorte qu'ils n'apparaissent que pour des cas exceptionnels et qu'ils ne concernent pas une majorité de tuples dans la relation

## Principe 4

- Veillez, au cours de la définition des schémas relationnels, à ce qu'ils puissent être réunis à l'aide de conditions d'égalité spécifiées sur des attributs jouant le rôle de clés primaires ou étrangères
  - Garanti l'absence des tuples parasites
- Evitez de faire des jointures sur des attributs qui ne sont pas uniques (clés primaires ou étrangères)

# Génération de tuples parasites

- Figure 15.5(a)
  - schemas des relations EMP\_LOCS and EMP\_PROJ1
- NATURAL JOIN
  - Le résultat produit beaucoup plus de tuples qu'il y en avait dans EMP\_PROJ
  - On appelle ces tuples supplémentaires **parasites**
  - Représentent des données parasites ou erronées qui ne sont pas valides

# Récap des principes de conception

- Les anomalies entraînent un surcroît de travail
- Gaspillage d'espace de stockage lié aux valeurs nulles
- Difficulté de réaliser des opérations d'aggrégation et de jointures
- Production de données non valides et parasites par des jointures

# Dépendances fonctionnelles

Avec les clés, les DF permettent de définir des formes normales, pour

- spécifier des mesures formelles pour la qualité d'une conception;
- détecter des problèmes vu dans le précédent, de manière précise.

# Définition: dépendance fonctionnelle

- Contrainte entre deux ensembles d'attributs de la base  $(R(A_1, A_2, \dots, A_n))$ .
- Propriété de la sémantique ou signification des attributs
  - Def: Une *dépendance fonctionnelle*, notée  $X \rightarrow Y$ , entre deux ensembles d'attributs X et Y qui sont des sous-ensembles de R, spécifie une contrainte sur les tuples possibles pouvant former un état de R. Selon cette contrainte, si deux tuples  $t_1$  et  $t_2$  sont tels que  $t_1[X] = t_2[X]$ , alors ceux-ci doivent être tels que  $t_1[Y] = t_2[Y]$ .
- **Etats relationnels légaux** satisfont les contraintes de dépendance fonctionnelles

## Définition: dépendance fonctionnelle (cont'd.)

- Donnée une relation peuplée:
  - On ne peut pas inférer automatiquement quelles DFs sont valides ou non
  - Sauf si la signification et les relations entre les attributs sont connus
  - Il suffit de trouver un seul contre exemple pour réfuter une DF

## Formes normales basées sur les clés primaires

- Processus de normalisation
- Approches pour les projets de conception relationnelle
  - La réalisation préalable d'un schéma conceptuel puis sa traduction en un ensemble de relations
  - La conception tenant en compte de la connaissance externe dérivée d'une implémentation de fichiers ou de formulaires ou de rapports existants

## Normalisation des relations

- Fait subir à un schéma relationnel une série de tests
  - Certifier qu'il satisfait à une forme normale déterminée
  - Opère du haut vers le bas
- **Test de formes normale**

**Définition:** la forme normale d'une relation est liée à la condition de forme normale qu'elle satisfait et indique le degré auquel celle-ci a été normalisée

## Normalisation des relations (cont'd.)

- Propriétés désirées du schéma relationnel:
  - **Jointure sans perte (non-additivité)**
    - Fortement recommandée
    - Garantie de ne pas générer de tuples parasites
  - **Préservation des dépendances**
    - Désirable, mais parfois sacrifiée pour d'autres aspects

# Usage pratique des formes normales

- Dans la pratique, la conception
  - est entreprise en manière à ce que les conceptions auxquelles on aboutit soient de haute qualité et satisfassent les propriétés précédemment évoquées.
  - mène la procédure de normalisation seulement jusqu'à la 3NF, BCNF, ou la 4NF
- Pas besoin de normaliser jusqu'à la forme normale la plus élevée

Def : Dénormalisation : stockage de la jointure de relations normalisées à un degré plus élevé, sous forme d'une relation de base.

## Définition: clés et attributs primaires

**Def : Une superclé** d'un schéma relationnel  $R=\{A_1, A_2, \dots, A_n\}$  est un sous-ensemble d'attributs  $S$  de  $R$  caractérisé par la propriété suivante : Dans chacun état relationnel  $r$  de  $R$ , deux tuples  $t_1$  et  $t_2$  ne seront pas tels que  $t_1[S] \neq t_2[S]$ .

Une **clé** est une superclé dotée de la propriété supplémentaire suivante : la suppression d'un de ses attributs annule son statut de superclé.

**Terminologie:** Si un schéma relationnel compte plusieurs clés, chacune est appelée clé candidate. On désigne une comme **clé primaire**. Les autres sont dites **clés secondaires**

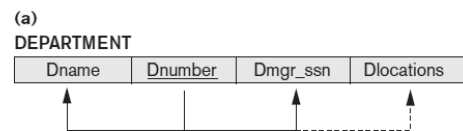
**Def :** L'attribut d'un schéma relationnel est  $R$  est appelé **attribut primaire** s'il est membre d'une des clés

## Première forme normale

- Fait partie de la définition formelle d'une relation du modèle relation de base (à un niveau)
- Seules valeurs permises pour les attributs sont **atomiques** (ou **indivisibles**)
- Techniques pour passer en 1NF
  - Supprimer un attribut et le placer dans une relation séparée
  - Étendre la clé
  - Utiliser plusieurs attributs atomiques

## Première forme normale (cont'd.)

- Interdit les **relations imbriquées**
  - Où un tuple peut avoir une relation à l'intérieur de lui-même
- Passage en 1NF:
  - Placer des attributs multivalués, dans une nouvelle relation
  - Propager la clé primaire dans cette nouvelle relation
  - **Désimbriquer** la relation, en faire un ensemble de relations 1NF



(b)

DEPARTMENT

Dname	Dnumber	Dmgr_ssn	Dlocations
Research	5	333445555	{Bellaire, Sugarland, Houston}
Administration	4	987654321	{Stafford}
Headquarters	1	888665555	{Houston}

(c)

DEPARTMENT

Dname	Dnumber	Dmgr_ssn	Dlocation
Research	5	333445555	Bellaire
Research	5	333445555	Sugarland
Research	5	333445555	Houston
Administration	4	987654321	Stafford
Headquarters	1	888665555	Houston

**Figure 15.9**  
Normalization into 1NF. (a) A relation schema that is not in 1NF. (b) Sample state of relation DEPARTMENT. (c) 1NF version of the same relation with redundancy.

## Deuxième forme normale

$X \rightarrow Y$  est **dépendance fonctionnelle complète**, si la suppression d'un des attributs  $A$  de  $X$  annule la DF. Opposée à **dépendance fct partielle**.

**Def.** Un schéma relationnel est en 2NF si chaque attribut non primaire de  $R$  est complètement dépendant fonctionnellement de la clé primaire de  $R$ .

Décomposition en plusieurs relations 2NF: Attributs non primaires ne sont associés qu'à la clé primaire dont ils sont fonctionnellement complètement dépendants

## Troisième forme normale

- $X \rightarrow Y$  est transitive s'il y a un ensemble d'attributs  $Z$  qui n'est ni une clé candidate ni un sous-ensemble d'une des clés de  $R$ , et que  $X \rightarrow Z$  et  $Z \rightarrow Y$  sont toutes les deux vraies.
- Def:** un schéma relationnel est en 3NF s'il satisfait 2NF et qu'aucun attribut non primaire n'est transitivement dépendant de la clé primaire
- DF problématique**
  - Partie gauche fait partie de la clé primaire
  - Partie gauche est un attribut ne jouant pas le rôle de clé

## Résumé: définitions 2NF et 3NF

Forme normale	Test	Remède (normalisation)
Première (1FN)	La relation ne doit pas comporter d'attributs non atomiques ou de relations imbriquées.	Former de nouvelles relations pour chaque attribut non atomique ou relation imbriquée.
Deuxième (2FN)	Pour les relations dans lesquelles la clé primaire contient plusieurs attributs, il ne doit pas y avoir d'attribut non-clé fonctionnellement dépendant d'une partie de la clé primaire.	Décomposer et construire une nouvelle relation pour chaque clé partielle et son (ses) attribut(s) dépendant(s). Il faut s'assurer de préserver une relation avec la clé primaire d'origine et tous les attributs qui dépendent fonctionnellement d'elle.
Troisième (3FN)	La relation ne doit pas avoir d'attribut non-clé fonctionnellement déterminé par un autre attribut non-clé (ou par un ensemble de plusieurs attributs non-clés). Autrement dit, il ne doit pas y avoir de dépendance transitive d'un attribut non-clé vis-à-vis de la clé primaire.	Décomposer et construire une relation qui inclut le ou les attributs non-clés qui déterminent fonctionnellement le ou les autres attributs non-clés.

## Définitions générales 2NF et 3NF

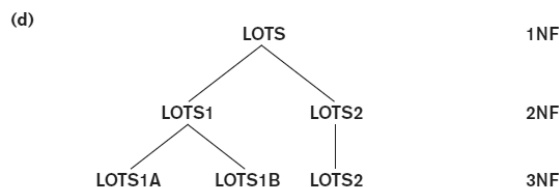
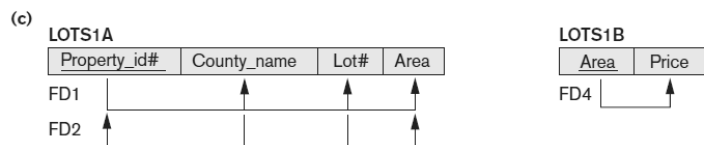
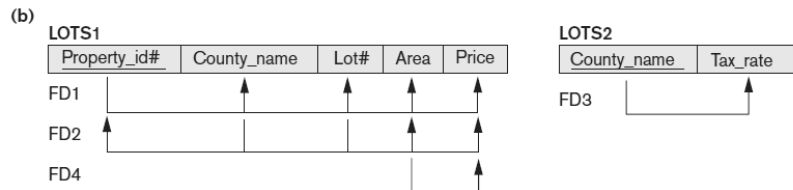
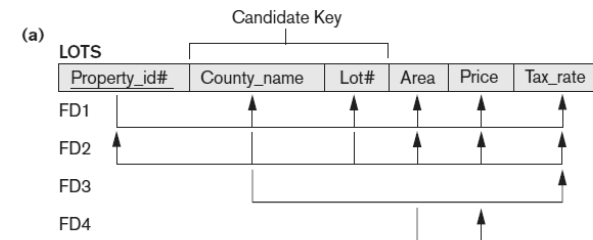
- Les définitions précédentes interdisent les dépendances partielles et transitives sur la clé primaire.
- Les dépendances fonctionnelles partielles, complètes et transitives seront désormais envisagés en tenant compte de toutes les clés candidates d'une relation
- Attribut primaire: tout attribut qui fait partie d'une des clés candidates

## Définition générale de seconde forme normale

**Def:** Un schéma relationnel R est en seconde forme normale (2FN) si chacun des attributs non primaires A de R n'est pas partiellement dépendant d'une des clés de R.

**Figure 15.12**

Normalization into 2NF and 3NF. (a) The LOTS relation with its functional dependencies FD1 through FD4. (b) Decomposing into the 2NF relations LOTS1 and LOTS2. (c) Decomposing LOTS1 into the 3NF relations LOTS1A and LOTS1B. (d) Summary of the progressive normalization of LOTS.



## Définition générale de troisième forme normale

**Definition.** Un schéma relationnel R est en troisième forme normale (3NF) si, lorsqu'une dépendance fonctionnelle non triviale  $X \rightarrow A$  est vraie dans R, soit

- X est une superclé de R, ou alors,
- A est un attribut primaire de R

**Definition alternative:** un schéma relationnel R est en 3NF si chacun des attributs non primaires de R satisfait les deux conditions suivantes:

- Il est fonctionnellement dépendant de chacun des clés de R
- Il est non transitivement dépendant de chacun des clés de R

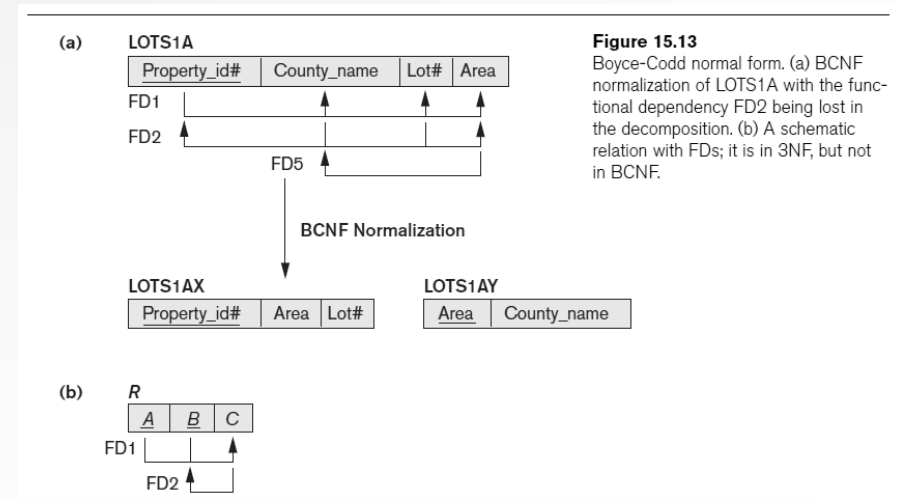


# Forme normale de Boyce-Codd

- Toute relation en FNBC est aussi en 3NF
  - Mais une relation en 3NF n'est pas nécessairement en FNBC

**Def: Un schéma relationnel est en FNBC si lorsqu'une dépendance fonctionnelle non triviale  $X \rightarrow A$  est vraie dans R, alors X est une superclé de R.**

- Différence entre FNBC et 3NF:
  - Condition 2, qui permet à A d'être primaire, est absente pour FNBC
- La plupart des schémas relationnels qui sont en 3NF, sont aussi en FNBC.



## Résumé

- Principes informels pour une conception de qualité
- Dépendance fonctionnelle
  - Outil de base pour l'analyse des schémas relationnels
- Normalisation:
  - 1NF, 2NF, 3NF, FNBC, 4NF, 5NF

## Règles d'inférence pour DFs (1)

Etant donné un ensemble de DFs F, nous pouvons en déduire des DFs supplémentaires, qui sont vérifiées par tous les états légaux de la base.

Les règles d'inférence d'Armstrong sont:

- IR1. (**Réflexivité**) si Y est sous ensemble de X, alors  $X \rightarrow Y$
- IR2. (**Extensibilité**) Si  $X \rightarrow Y$ , alors  $XZ \rightarrow YZ$ 
  - (Notation: XZ signifie  $X \cup Z$ )
- IR3. (**Transitivité**) si  $X \rightarrow Y$  et  $Y \rightarrow Z$ , alors  $X \rightarrow Z$

IR1, IR2, IR3 constituent un ensemble de règles d'inférence correct et complet

- Ses règles sont valides et toutes les autres règles déduites d'elles également

## Règles d'inférence pour DFs (2)

Quelques règles supplémentaires sont utiles:

- **Décomposition:** Si  $X \rightarrow YZ$ , alors  $X \rightarrow Y$  et  $X \rightarrow Z$
- **Union:** Si  $X \rightarrow Y$  et  $X \rightarrow Z$ , alors  $X \rightarrow YZ$
- **Pseudo-transitivité:** Si  $X \rightarrow Y$  et  $WY \rightarrow Z$ , alors  $WX \rightarrow Z$

Les trois dernières règles peuvent être déduites de IR1, IR2 et IR3 (propriété de compétence)

## Règles d'inférence pour DFs (3)

**Clôture** d'un ensemble  $F$  de DFs est l'ensemble  $F^+$  de toutes les DFs pouvant en être déduites

**Clôture** d'un ensemble d'attributs  $X$  par rapport à  $F$  est l'ensemble  $X^+$  de tous les attributs qui dépendent fonctionnellement de  $X$

$X^+$  peut être calculé en appliquant IR1, IR2, IR3 avec les FDs de  $F$ , de manière itérée

## Equivalence d'ensembles de DFs

Deux ensembles de DFs  $F$  et  $G$  sont **équivalents** si:

- Chaque DF dans  $F$  peut être déduite de  $G$ , et
- Chaque DF dans  $G$  peut être déduite de  $F$
- Donc,  $F$  et  $G$  sont équivalents si  $F^+ = G^+$

Définition (**Couvrir**):

- $F$  **couvre**  $G$  si chaque DF dans  $G$  peut être déduite de  $F$ 
  - (i.e., si  $G^+$  est sous-ensemble de  $F^+$ )

$F$  et  $G$  sont équivalents s'ils se couvrent mutuellement

Il y a un algorithme pour tester l'équivalence d'ensemble de DFs