

# SGBDR

## Systèmes de Gestion de Bases de Données Relationnelles

*Alexis NEDELEC*

LISYC EA 3883 UBO-ENIB-ENSIETA  
Centre Européen de Réalité Virtuelle  
Ecole Nationale d'Ingénieurs de Brest

*enib ©2007*



# Introduction

## Modélisation d'un SI

Modéliser les données d'un système logiciel, c'est construire une **structure de données contraintes**, représentative du réel observé.

## Structuration des informations

Apprendre à structurer et à exprimer des contraintes.

- ce qui formalise le point de vue de l'organisation.
  - on ne retient pas tout ce qu'on peut observer
- on recherche des similitudes, on élabore des types.
  - On ne recense pas tout ce qui est passé, présent où à venir

# Abstraction

## Concrètement c'est quoi l'abstraction ?

- classifier, instancier : "Jean est un homme"
- composer, décomposer : "Un homme a une tête, des bras ..."
- généraliser, spécialiser : "Un homme est un mammifère"
- grouper, séparer : "Jumbo est un éléphant, l'éléphant est une espèce en voie de disparition"
- associer : "Jean, un homme, dresse Jumbo, un éléphant"

# Abstraction

## Similarité de conception : "Jean est un homme"

- Philosophie : Concept (Homme), Réalisation (Jean)
- Biologie : Espèce (Homme), Spécimen (Jean)
- Mathématiques : Ensemble (Homme), Élément (Jean)
- Programmation Objet : Classe (Homme), Instance (Jean)
- Modèle Relationnel : Type (Homme), Occurrence (Jean)
- ...

# Systèmes d'Information

## But des Systèmes d'Information

- amplification du pouvoir de mémorisation
- dans le but de faciliter les prises de décision

## Trois axes de Modélisation de SI

- QUE manipule système (les informations, données)
- COMMENT le système manipule (opérations, fonctions)
- QUAND le système manipule (contrôle dynamique)

## Catégories d'applications des SI

- Gestion d'Informations : stockage d'informations
- Calcul Scientifique : traitement d'informations
- Systèmes Temps Réel : contrôle d'informations

# Systèmes d'Information

## Analogie Programmation

- QUOI : les variables représentent les informations
- COMMENT : les fonctions représentent les opérations
- QUAND : l'algorithme représente le contrôle.

## Analogie UML

- QUOI : diagramme de classes, composants...
- COMMENT : diagramme de cas d'utilisation
- QUAND : diagramme d'états, séquences, collaboration...

# Appellation Bases de Données

## Les trois axes

- Description de Données : représentation d'informations
- Manipulation de Données : modification d'informations
- Contrôle de Données : transactions sur une Base de Données

## Appellation Base de Données

- ensemble de données non-indépendantes
- interrogeables par le contenu
- selon n'importe quel critère

## Exemple : Employés dans une Entreprise

- que représente un **employé** dans une **entreprise** ?
- rechercher les **employés** de **salaire** supérieur à **2 000 Euros**
- demander les **caractéristiques** (structure) de l'**employé**

# Appellation SGBD

## Systèmes de Gestion de Bases de Données

“Ensemble de logiciels systèmes pour manipuler une information dans une grande masse d’informations partagées par de nombreux utilisateurs”

## Problématique des SGBD

“Retrouver une aiguille dans une botte de foin ... avant les autres”

## SGBD ou système de fichiers

- SGBD : savoir quelles informations trouver
- Fichiers : savoir comment aller chercher les informations



# Appellation SGBD

## Fonctionnalités de base : gestion des données

- sauvegarde : persistance des données sur disque
- interrogation : récupération des données dans une application
- recherche : critères permettant de retrouver des données
- mise en forme : présentation des données à l'utilisateur

## Fonctionnalités évolués d'un SGBD : traitement sur les données

- intégrité des données : cohérence et redondance
- partage de données : mises à jour concurrentes
- efficacité d'accès aux données : goulot d'étranglement
- protection des données : autorisations, récupération sur panne

# Evolutions des SGBD

## Les années 60/70

- CODASYL : **C**onference **O**n **D**Ata **S**Ystems **L**anguages.
- 1ère génération de SGBD (IBM, Honeywell, TOTAL ...)
- Modèle relationnel, mode client/serveur
- 2ème génération de SGBD (ORACLE, INGRES, SYBASE...)

## Les années 80/90

- Modèle Objet : enrichir le modèle des données manipulables
- 3ème génération de SGBD (ObjectStore, O2, Versant ...)

## Les années 2000

- SGBD et Internet : Data Warehouse, Data Mining, Web sémantique...

# Modélisation sémantique

## Entité-Association : Entité-Relation (ER)

- **Hiérarchique** : association Père-fils
- **Réseau** : tout type d'association
- **Relationnel** : opérations ensemblistes

## E.F. CODD (1970) : Modèle Relationnel

- indépendance description/stockage de données
- représentation simple des données
- gestion de cohérence, redondance de données
- manipulation de données par des langages non-procéduraux

# Modèle Entité-Association

## Modèle Relationnel

- théorie des ensembles, algèbre relationnelle
- définition de relation (entre ensembles, domaines, tables)
- définition de règles d'intégrité sur les relations
- étude des Dépendances Fonctionnelles (DF) entre données
- problèmes de "normalisation" pour structurer une BD
- définition des opérations de l'algèbre relationnelle

## SGBD Relationnel

- standard de description (LDD), manipulation (LMD) de BD
- langage associé (SQL) : **S**tructured **Q**uery **L**anguage

# Ensemble

## Définition intuitive d'un ensemble

- collection d'objets appelés **éléments**
- **sémantiquement défini** mais non-structurée
- **logique booléenne d'appartenance** à un ensemble ( $E$ )
  - l'élément ( $x$ ) est ( $x \in E$ )/n'est pas ( $x \notin E$ ) dans l'ensemble
- **unicité** : un élément n'est pas 2 fois dans le même ensemble

## Définition des valeurs d'un domaine (ensemble)

- **extension (compréhension)** : énumération des valeurs  
*couleurBieres* = {*blonde, rousse, brune*}
- **intention** : propriétés du domaine  
*BiereBlondes* = { $x \mid blonde(x)$ }

# Ensemble et Logique

## Ensemble ( $E$ ) et Equivalence logique ( $p \Leftrightarrow q$ )

- $(p : x \in E)$  et  $(q : prop(x))$  ont même valeur de vérité
- $E = \{x \mid prop(x)\}$
- $prop(x)$  : condition **nécessaire et suffisante** pour que  $x \in E$
- $p \Leftrightarrow q : \forall x ((x \in E) \Leftrightarrow prop(x))$

## Sous-ensemble ( $A \subseteq E$ ) et implication ( $p \Rightarrow q$ )

- $(q : x \in E)$  ne peut être faux si  $(p : x \in A)$  est vraie
- $x \in E$  : condition **nécessaire** pour que  $x \in A$
- $x \in A$  : condition **suffisante** pour que  $x \in E$
- $x \in E$  : **nécessaire** mais **non-suffisant** pour que  $x \in A$
- $p \Rightarrow q : \forall x ((x \in A) \Rightarrow (x \in E))$

# Ensembles et Produit Cartésien

## Produit cartésien d'ensembles

$$A \times B = \{(x, y) \mid (x \in A) \wedge (y \in B)\}$$

## Exemple de produit cartésien

$$Cru \times Couleur = \{Sancerre, Chablis\} \times \{blanc, rouge, rosé\}$$

<b>Cru</b>	<b>Couleur</b>
Sancerre	rouge
Sancerre	rosé
Sancerre	blanc
Chablis	rouge
Chablis	rosé
Chablis	blanc

# Ensembles et Relation

## Relation $R$

Tout sous-ensemble de produit cartésien  $(A \times B)$  :

$$R \subseteq (A \times B) = \{(a, b) \mid (a, b) \in R\}$$

Notation :  $a R b$

## Exemple de relation

$$A = \{2, 4, 6\}, B = \{1, 2, 3, 4, 5, 6\}$$

La relation  $(a R b)$  suivante :

$$R = \{(2, 3), (2, 4), (2, 5), (2, 6), (4, 5), (4, 6)\}$$

est le sous-ensemble **inférieur** à du produit cartésien  $A \times B$



# Relation : Domaine, Image et Fonction

## Domaine, Image

Domaine du sous-ensemble  $R \subseteq (A \times B)$  :

$$\text{dom}(R) = \{x \in A \mid \exists y \in B : xRy\}$$

Image du sous-ensemble  $R \subseteq (A \times B)$  :

$$\text{image}(R) = \{y \in B \mid \exists x \in A : xRy\}$$

## Relation fonctionnelle : “au plus” un élément de $B$

Une **relation**  $f \subseteq (A \times B)$  est **fonctionnelle** si :

$$\forall x \in A, \forall y, z \in B : (xRy) \wedge (xRz) \Rightarrow y = z$$

## Application : “un et un seul” élément de $B$

Une relation fonctionnelle est une **fonction** :

$$f : A \rightarrow B \text{ si } \text{dom}(f) = A$$

# Relation : Injection, Surjection, Bijection

## Application injective : Injection

Deux éléments du domaine de la relation ont 2 éléments distincts dans l'image de la relation ( $x \neq y \Rightarrow f(x) \neq f(y)$ )

$$\forall x, y \in \text{dom}(f), f(x) = f(y) \Rightarrow x = y$$

## Application surjective : Surjection

L'image de la relation est l'ensemble d'arrivée :  $\text{image}(f) = B$

## Bijection = Injection + surjection

Pour tout élément de l'ensemble d'arrivée, il existe UN SEUL élément correspondant du domaine de la relation.

# Modèle Relationnel

## Définition de relation

“Sous-ensemble du produit cartésien d’une liste de domaines caractérisé par un nom”

## Relation ou Table

- chaque colonne (domaine) est identifié par un **nom**
- ce nom est un **attribut**
- un attribut varie dans son **domaine** respectif
- l’ensemble des attributs est appelé **schéma** de la relation
- le nombre d’attributs s’appelle **arité** de la relation
- un élément (ligne) d’une relation est un **n-uplet**

# Modèle de relation

## Instance de relation

“Ensemble des n-uplets qui constituent, **à un instant donné**, le contenu de la relation”

## Exemple : relation VIN(Cru, Millésime, Degré)

Cru	Millésime	Degré
Juliéna	1986	12
Chablis	1983	11.5
Volnay	1979	12.5

# Contraintes d'intégrité

## Règles d'intégrité

- sur une relation : unicité de clé, contrainte référentielle
- sur une colonne : entité, domaine, valeur nulle, par défaut ...

## Relation : Unicité de clé

“**Ensemble minimal** d'attributs dont la connaissance des valeurs permet d'identifier de façon unique un n-uplet de la relation considérée”

## Relation : Contrainte référentielle

“Contrainte d'intégrité, portant sur une relation R1, consistant à imposer que la valeur d'un groupe d'attributs de R1 apparaisse comme valeur de clé dans une autre relation R2”

# Contraintes d'intégrité

## Colonne : Valeur nulle

“Valeur conventionnelle introduite dans une relation pour représenter une information inconnue ou inapplicable”

## Colonne : contrainte d' Entité

“Toute relation possède une clé primaire et tout attribut participant à cette clé est non-nul”

## Colonne : contrainte de Domaine

“imposer à une colonne de relation de comporter des valeurs vérifiant une assertion logique”

# Dépendances Fonctionnelles

## Modélisation des contraintes

- Dépendances Fonctionnelles (DF) : structuration de la base
- Dépendances Multi-Valuées (DMV) : décomposition de relation en deux tables
- Dépendances de Jointures (DJ) : décomposition de relation en plusieurs tables

## Dépendances Fonctionnelles

- trouver les dépendances entre les informations à modéliser
- déterminer les “clés” d’un schéma de relation

# Dépendances Fonctionnelles

## Définition

Soit le schéma relationnel  $R(X, Y, Z)$ , il existe une **DF** :  $X \rightarrow Y$  entre les groupes d'attributs  $X$  et  $Y$  **ssi**  $\forall(x, y, y', z, z')$  :

$$\left. \begin{array}{l} (x, y, z) \in R(X, Y, Z) \\ (x, y', z') \in R(X, Y, Z) \end{array} \right\} \Rightarrow y = y'$$

## Interprétation

- $Y$  dépend fonctionnellement de  $X$
- les valeurs de  $X$  déterminent les valeurs de  $Y$

**Clé** d'un schéma de relation  $R(X, Y)$  :

- Il existe une DF :  $X \rightarrow Y$



# Dépendances Fonctionnelles

## Définition : Projection ( $\Pi$ )

Soient :

- $n_1, n_2$  : n-uplets de la relation  $R$
- $r$  : instance de la relation  $R$

Une DF  $X \rightarrow Y$  devra vérifier :

- $\Pi_X(n_1) = \Pi_X(n_2) \Rightarrow \Pi_Y(n_1) = \Pi_Y(n_2)$

## Définition : Projection, restriction ( $\Pi, \sigma$ )

Soit :

- $r$  : instance de la relation  $R$

Une DF  $X \rightarrow Y$  devra vérifier :

- $\forall x \in X, \Pi_Y(\sigma_{X=x}(r))$ , UN SEUL n-uplet en résultat

# Dépendances Fonctionnelles

## Définition : Schéma de relation

On appelle **Schéma Relationnel** d'une relation  $R$ , noté  $R(U, F)$

- l'ensemble  $U$  des attributs de la relation  $R$
- l'ensemble  $F$  des dépendances fonctionnelles applicables à  $R$

## Définition : Fermeture

On appelle **Fermeture**, notée  $F'$ , d'un ensemble de DF :

- l'ensemble  $F$  augmenté de toutes les **implications logiques**
- déduit par les **axiomes d'Armstrong**

# Dépendances Fonctionnelles

## Définition : Règles d'inférences, axiomes d'Armstrong

① **Réflexivité :**

$X \rightarrow X$  ou bien : **si**  $Y \subset X \subset U$  **alors**  $X \rightarrow Y$

② **Transitivité :**

**si**  $(X \rightarrow Y, Y \rightarrow Z)$  **alors**  $X \rightarrow Z$

③ **Pseudo-Transitivité :**

**si**  $(X \rightarrow Y, YZ \rightarrow W)$  **alors**  $XZ \rightarrow W$

④ **Augmentation :**

**si**  $(X \rightarrow Y, Z \subset U)$  **alors**  $XZ \rightarrow Y$

⑤ **Additivité (Union) :**

**si**  $(X \rightarrow Y, X \rightarrow Z)$  **alors**  $X \rightarrow YZ$

⑥ **Décomposition (Projectivité) :**

**si**  $(X \rightarrow Y, Z \subset Y)$  **alors**  $X \rightarrow Z$

# Dépendances Fonctionnelles Elémentaire

## Définition : DFE

Une DFE  $X \rightarrow A$ , est une DF où :

- $A$  est un attribut unique tel que  $A \notin X$
- $\nexists X' \subset X, X' \rightarrow A$

## Remarques sur les DFE

- la cible ( $A$ ) est un attribut unique
- la source ( $X$ ) ne comporte pas d'attributs superflus
- transitivité : seule règle d'inférence qui s'applique aux DFE

## Définition : Fermeture Transitive

On appelle Fermeture Transitive notée  $F^+$

- L'ensemble des DFE enrichi des DFE déduites par transitivité

# Couverture minimale

## Définition : Couverture minimale ( $\bar{F}$ )

Le sous-ensemble minimal de la Fermeture Transitive

- ① Un seul attribut à droite de la DF
- ② Aucune DF ne peut être supprimée
- ③ Aucun attribut à gauche de la DF ne peut-être enlevé

Autrement dit :

- ① Les parties droites des DF sont réduites à un élément

$$A \rightarrow BC \Rightarrow A \rightarrow B, A \rightarrow C$$

- ② Il n'y a pas de DF redondantes

$$\forall X \in \bar{F}(X \rightarrow A), \bar{F} - \{X \rightarrow A\} \neq \bar{F}$$

- ③ les parties gauches sont “dégrossies”

$$\forall X \in \bar{F}(X \rightarrow A)$$

$$\text{Si } \exists Z \subset X \text{ Alors } (\bar{F} - \{X \rightarrow A\}) \cup \{Z \rightarrow A\} \neq \bar{F}$$

# Couverture minimale

## Exemple de cahier des charges d'entreprise

“Une société de ventes par correspondance veut créer une B.D. sur des clients, des produits et des commandes”

## Pour chaque client

- un numéro de client, unique à chaque client : **C**
- une ou plusieurs adresses d'expédition : **A**
- l'état de son compte (somme due ou à avoir) : **E**

# Couverture minimale

## Pour chaque Produit

- un numéro de produit, unique à chaque produit : **P**
- le genre du produit (alimentaire,électro-ménager,...) : **G**
- la marque du produit : **M**

## Pour chaque Commande

- une référence de commande, unique à chaque commande : **R**
- le numéro du client (**C**)
- l'adresse d'expédition (**A**)
- la date de commande : **D**
- un numéro de produit (**P**)
- une quantité commandée : **Q**

# Couverture minimale

## Contraintes d'intégrité

Sur l'ensemble des attributs  $U = (C, A, E, P, G, M, R, D, Q)$

- un client n'a qu'un état de compte
- il n'y a pas deux clients différents à une même adresse
- un produit n'a qu'un seul genre et une seule marque
- une commande ne concerne qu'un seul client  
n'est livrée qu'à une seule adresse  
et n'a qu'une seule date de livraison
- pour la même commande, le même produit, le même client  
il n'y a qu'une seule quantité commandée

## Problème

A partir du cahier des charges trouver la couverture minimale



# Normalisation de Base de Données

## Problème

- éviter la redondance et l'incohérence des données
- décomposition de relations, sans perte d'informations

## Etudes de normalisation

- 3 premières Formes Normales : étude des DF
- 4ème forme normale : Dépendances Multi-Valuées (DMV)
- 5ème forme normale : Dépendances de Jointures (DJ)

## Structuration de Base de Données

- garantir les trois premières formes normales

## Première Forme Normale (1NF)

## Définition 1NF

“Aucun attribut n’est lui-même une relation”

## Contre-exemple sur un schéma de clé (kba, kbi)

BAR DU MONDE (0NF)							
kba	bar	pays	cont.	kbi	biere	coul	stock
1	Bar	France	UE	1	Spat.	blo.	50
				2	Gui.	bru.	300
				3	Kilk.	rou.	100
				4	Pelf.	blo.	500
2	Pub	USA	Amér.	2	Guin.	bru.	50
3	Caffee	All	UE	1	Spat.	blo.	750
				3	Kilk.	rou.	50

## Première Forme Normale (1NF)

## Schéma relationnel en 1NF

BAR DU MONDE (1NF)							
kba	kbi	bar	pays	cont.	biere	coul	stock
1	1	Bar	France	Euro.	Spat.	blo.	50
1	2	Bar	France	Euro.	Guin.	bru.	300
1	3	Bar	France	Euro.	Kilk.	rou.	100
1	4	Bar	France	Euro	Pelf.	blo.	500
2	2	Pub	USA	Amér.	Guin.	bru.	50
3	1	Caffee	All.	Euro.	Spat.	blo.	750
3	3	Caffee	All.	Euro.	Kilk.	rou.	50

# Deuxième Forme Normale (2NF)

## Définition 2NF

“La relation est **1NF** et tous les attributs **non-primitifs** sont **pleinement dépendant d’une clé**”

- Attribut non-primitif : attribut n’appartenant à aucune clé
- soit  $X \rightarrow A$  une DF,  $A$  pleinement dépendant de  $X$  si :  
 $\exists X' \subset X, X' \rightarrow A \in F^+$

## Interprétation 2NF : Toute la clé

- si  $R(\underline{A}, \underline{B}, C, D)$  et  $B \rightarrow C$
- alors  $R_1(\underline{A}, \underline{B}, D)$  et  $R_2(\underline{B}, C)$

# Deuxième Forme Normale (2NF)

## Schéma relationnel de clé (kba, kbi)

- il existe une DF : kba  $\rightarrow$  (bar, pays, continent)
- **décomposition sans perte** de la relation en deux relations

## Théorème de décomposition sans perte

“Une relation  $R$  est décomposable sans perte en deux relations  $R_1, R_2$  **ssi** l’intersection des deux relations est une DF sur la différence des deux.”

Autrement dit :

- soit  $(R_1 \cap R_2) \rightarrow (R_1 - R_2)$
- soit  $(R_1 \cap R_2) \rightarrow (R_2 - R_1)$

## Deuxième Forme Normale (2NF)

## Exemple de décomposition sans perte

Soit le schéma relationnel :  $R[\{X, Y, Z\}, \{f\}]$  où  $f : X \rightarrow Y$

Décomposable sans perte en

$$R_1 = R[X, Y], R_2 = R[X, Z]$$

- $(X, Y) \cap (X, Z) = X, (X, Y) - (X, Z) = Y$
- $(X, Y) \cap (X, Z) \rightarrow (X, Y) - (X, Z)$

Non-décomposable sans perte en

$$R_1 = R[X, Y], R_2 = R[Y, Z]$$

- $(X, Y) \cap (Y, Z) = Y$
- $(X, Y) - (Y, Z) = X, (Y, Z) - (X, Y) = Z$
- $(X, Y) \cap (Y, Z) \not\rightarrow (X, Y) - (Y, Z)$
- $(X, Y) \cap (Y, Z) \not\rightarrow (Y, Z) - (X, Y)$

## Deuxième Forme Normale (2NF)

## Schéma relationnel en 2NF

BAR DU MONDE (1NF')								
kba	bar	pays	cont.	kba	kbi	bi.	coul	sto.
1	Bar	France	Euro.	1	1	Spa.	blo.	50
2	Pub	USA	Amér.	1	2	Gui.	bru.	300
3	Caffee	All.	Euro.	1	3	Kil.	rou.	100
				1	4	Pel.	blo.	500
			.	2	2	Gui.	bru.	50
				3	1	Spa.	blo.	750
				3	3	Kil.	rou.	50

## Deuxième Forme Normale (2NF)

## Schéma relationnel de clé (kba, kbi)

- il existe encore une DF :  $kbi \rightarrow (\text{biere}, \text{couleur})$

## BAR DU MONDE (2NF)

SERVICES		
kba	kbi	stock
1	1	50
1	2	300
1	3	100
1	4	500
2	2	50
3	1	750
3	3	50



## Deuxième Forme Normale (2NF)

## BAR DU MONDE (2NF)

<b>BARS</b>			
<b>kba</b>	<b>bar</b>	<b>pays</b>	<b>continent</b>
1	Bar du Coin	France	Euro.
2	Corner's Pub	USA	Amérique
3	Caffee der Ecke	Allemagne	Euro.

<b>BIERES</b>		
<b>kbi</b>	<b>biere</b>	<b>couleur</b>
1	Spaten	blonde
2	Guinness	brune
3	Kilkenny	rousse
4	Pelforth	blonde

## Troisième Forme Normale (3NF)

## Définition 3NF

“La relation est **2NF** et tous les attributs **non-primitifs** sont **directement dépendant** d'une clé”

Soit une DF  $X \rightarrow A$ ,  $A$  directement dépendant de  $X$

- $A$  n'est pas **transitivement dépendant** de  $X$

Dépendance transitive

- $\exists Y$  tel que  $A \notin Y$  et  $X \rightarrow Y, Y \rightarrow A$
- où  $Y \not\subset X$  et  $Y \not\rightarrow X$

Interprétation 3NF : Toute la clé (2NF) et rien que la clé (3NF)

- si  $R(\underline{A}, \underline{B}, C, D, E)$  et  $C \rightarrow D$
- alors  $R_1(\underline{A}, \underline{B}, C, E)$  et  $R_2(\underline{C}, D)$

# Troisième Forme Normale (3NF)

## Schéma relationnel de clé (kba, kbi)

- il existe une dépendance transitive : **bar**  $\rightarrow$  **continent**
- du fait de la dépendance fonctionnelle : **pays**  $\rightarrow$  **continent**

## BAR DU MONDE (3NF)

- BARS(kbar, bar, pays)
- MONDE(pays, continent)
- BIERES(kbiere, biere, couleur)
- SERVICES(kbar, kbiere, stock)

# Les trois premières Formes Normales

## Bonne normalisation

### Première Forme Normale (1NF)

- aucun attribut n'est lui-même une relation

### Deuxième Forme Normale (2NF)

- 1NF et pas de sous-clé
- éliminer les dépendances : partie de clé et attributs non-clés

### Troisième Forme Normale (3NF)

- 2NF et rien que la clé
- éliminer les dépendances : entre attributs non-clés

## Boyce-Codd Normal Form (BCNF)

- 3NF et les seules DFE existantes sont des clés
- éliminer les dépendances : entre parties de clés

# Les trois premières Formes Normales

## En résumé

### Deuxième Forme Normale (2NF)

- Si  $R(\underline{A}, \underline{B}, C, D, E)$  et  $B \rightarrow C$
- Alors  $R_1(\underline{A}, \underline{B}, D, E)$  et  $R_2(\underline{B}, C)$

### Troisième Forme Normale (3NF)

- Si  $R(\underline{A}, \underline{B}, C, D, E)$  et  $C \rightarrow D$
- Alors  $R_1(\underline{A}, \underline{B}, C, E)$  et  $R_2(\underline{C}, D)$

### Boyce-Codd Normal Form (BCNF)

- Si  $R(\underline{A}, \underline{B}, C, D, E)$  et  $C \rightarrow B$
- Et  $(\underline{A}, C)$  clé candidate
- Alors  $R_1(\underline{A}, C, D, E)$  et  $R_2(\underline{C}, B)$

# Les 5 Formes Normales

## Les deux autres Formes Normales

### Quatrième Forme Normale (4NF)

- relation est en BCNF
- Décomposition sans perte en 2 tables

### Cinquième Forme Normale (5NF) :

- généralisation de la 4NF : décomposition en plusieurs tables

# Quatrième Forme Normale

## 4NF

- dépendances sur plusieurs valeurs redondantes dans une table

Exemple : ‘Des **Buveur** vont dans des **Bar** boire des **Biere**’

Il n'existe pas de Dépendances Fonctionnelles

- $\text{Buveur} \not\rightarrow \text{Bar}$ ,  $\text{Bar} \not\rightarrow \text{Biere}$ , ...

Avec ce schéma relationnel (Buveur,Bar,Biere) :

- un Buveur peut boire une Biere dans un Bar
- un Buveur peut boire **la même** Biere dans **un autre** Bar
- un Buveur peut boire **une autre** Biere dans **le même** Bar

# Quatrième Forme Normale

## Décomposition sans perte

seulement pour certaines instances de relation :

- BUVEUR(Buveur,Bar)
- BAR(Buveur,Biere)

Buveur	Bar	Biere
Jojo	bar du coin	Kanterbrau
Jojo	bar du coin	Heineken
Jojo	bar du stade	Kanterbrau
Jojo	bar du stade	Heineken
Nono	bar des amis	Guinness
Nono	bar des amis	Pelforth



# Quatrième Forme Normale

## Décomposition sans perte

Buveur	Bar	Buveur	Biere
Jojo	bar du coin	Jojo	Kanterbrau
Jojo	bar du stade	Jojo	Heineken
Nono	bar des amis	Nono	Guinness
		Nono	Pelforth

exemple de relation non- décomposable

Buveur	Bar	Biere
Jojo	bar du coin	Kanterbrau
Jojo	bar du stade	Kanterbrau
Jojo	bar du stade	Heineken
Nono	bar des amis	Guinness
Nono	bar des amis	Pelforth

# Quatrième Forme Normale

## Définition : Dépendance Multi-Valuée (DMV)

Il existe une DMV sur une relation  $R(X,Y,Z)$ , notée  $X \twoheadrightarrow Y, Z$

ssi :  $\forall(x, y, y', z, z')$ ,

$$\left. \begin{array}{l} (x, y, z) \in R(X, Y, Z) \\ (x, y', z') \in R(X, Y, Z) \end{array} \right\} \Rightarrow \begin{array}{l} (x, y, z') \in R(X, Y, Z) \\ (x, y', z) \in R(X, Y, Z) \end{array}$$

## Exemple : ‘Des **Buveur** vont dans des **Bar** boire des **Biere**’

On a une DMV dans la table (Buveur,Bar,Biere)

- Buveur  $\twoheadrightarrow$  Bar (et donc Buveur  $\twoheadrightarrow$  Biere)

## Schéma de relation en 4NF

- il n'existe qu'une DMV par table
- les DMV sont des super-clés (contiennent une clé)

## Cinquième Forme Normale

## 5NF

- étude des décompositions en plusieurs tables

## Exemple de monde à modéliser

“ un **Buveur** va dans un ensemble de **Bar** et aime un ensemble de **Bière** mais ces bars ne servent pas forcément toutes ces bières”

Buveur	Bar	Biere
Jojo	bar du coin	Kanterbrau
Jojo	bar du coin	Heineken
Jojo	bar des amis	Heineken
Nono	bar du coin	Heineken

Pas de DF, DMV (le Bar des Amis ne sert pas de Kanterbrau)

# Cinquième Forme Normale

## Exemple de monde à modéliser

“Tout **Buveur** appréciant une **Bière** et ayant commandé dans un **Bar** servant cette **Bière** a commandé cette marque de **Bière** dans ce **Bar**”

que l'on peut formaliser de la manière suivante :

**Si**  $(bu, bi) \in R_1$  et  $(bu, ba) \in R_2$  et  $(ba, bi) \in R_3$   
**alors**  $(bu, bi, ba) \in R$

Décomposition de la relation précédente en trois relations :

- FREQUENTATION(Buveur,Bar)
- PREFERENCE(Buveur,Biere)
- SERVICE(Bar,Biere)

## Cinquième Forme Normale

## Dépendance de Jointure (DJ)

Il existe une DJ entre X et Y, notée  $X \leftrightarrow Y$ , sur  $R(X,Y,Z)$   
**ssi**  $\forall(x, y1, y2, z1, z2) :$

$$si \left\{ \begin{array}{l} (x, y1, z1) \in R(X, Y, Z) \\ (x, y2, z2) \in R(X, Y, Z) \end{array} \right. \text{ et } \left\{ \begin{array}{l} (y1, z2) \in R[Y, Z] \\ (y2, z1) \in R[Y, Z] \end{array} \right.$$

$$alors \left\{ \begin{array}{l} (x, y1, z2) \in R(X, Y, Z) \\ (x, y2, z1) \in R(X, Y, Z) \end{array} \right.$$

## Schéma de relation en 5NF

- Toute dépendance de jointure est due aux clés candidates
- $R$  : jointure de ces projections  $R = R_1 \bowtie R_2 \bowtie R_3 \dots \bowtie R_n$

## DMV et DJ

Définition : Dépendance Multi-Valuée (DMV) :  $X \twoheadrightarrow Y/Z$

ssi :  $\forall x \in X,$

$$R[X = x/Y, Z] = R[X = x/Y] \times R[X = x/Z]$$

Définition : Dépendance Multi-Valuée (DMV) :  $X \twoheadrightarrow Y/Z$

ssi :  $\forall (x, y, y', z, z'),$

$$\left. \begin{array}{l} (x, y, z) \in R(X, Y, Z) \\ (x, y', z') \in R(X, Y, Z) \end{array} \right\} \Rightarrow \begin{array}{l} (x, y, z') \in R(X, Y, Z) \\ \wedge \\ (x, y', z) \in R(X, Y, Z) \end{array}$$

## DMV et DJ

Définition : Dépendance (mutuelle) de Jointure (DJ) :  $X \leftrightarrow Y/Z$

ssi :  $\forall (x, y, y', z, z')$ ,

$$\left. \begin{array}{l} (x, y, z) \in R(X, Y, Z) \\ (x, y', z') \in R(X, Y, Z) \end{array} \right\} \Rightarrow \begin{array}{l} (x, y, z') \notin R(X, Y, Z) \\ \vee \\ (x, y', z) \notin R(X, Y, Z) \end{array}$$

# Opérateurs relationnels

## Cinq opérations de base (unaire et binaire)

- Projection ( $\Pi$ ), Restriction ( $\sigma$ )
- Produit cartésien ( $\times$ ), Union ( $\cup$ ), Différence ( $-$ )

## Opérations binaires dérivées

- Intersection ( $\cap$ ), Jointure ( $\bowtie$ ), Division (quotient) ( $\div$ )
- Jointure externe, Semi-Jointure
- ...

## Opérations unaires dérivées

- Complément, Eclatement
- Fermeture Transitive
- ...



# Opérateurs relationnels

Projection : Relation  $\times$  {attributs}  $\rightarrow$  Relation

$$\Pi_Y(R(X, Y)) = \{ \langle y \rangle \mid \exists x \langle x, y \rangle \in R(X, Y) \}$$

Restriction : Relation  $\times$  expression logique  $\rightarrow$  Relation

$$\sigma_E(R(X)) = \{ \langle x \rangle \mid \langle x \rangle \in R(X) \wedge E(x) \}$$

Produit cartésien : Relation  $\times$  Relation  $\rightarrow$  Relation

$$T(Z) = R(X) \times S(Y) = \{ \langle z \rangle \mid \forall \langle x \rangle, x \in R, \forall \langle y \rangle, y \in S \\ \exists z \in T \wedge \Pi_X(z) = x \wedge \Pi_Y(z) = y \}$$

$\cup, \cap, -$  : Relation  $\times$  Relation  $\rightarrow$  Relation

$$R(X) \cup S(X) = \{ \langle x \rangle \mid \langle x \rangle \in R(X) \vee \langle x \rangle \in S(X) \}$$

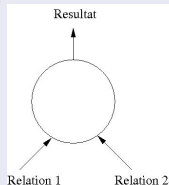
$$R(X) \cap S(X) = \{ \langle x \rangle \mid \langle x \rangle \in R(X) \wedge \langle x \rangle \in S(X) \}$$

$$R(X) - S(X) = \{ \langle x \rangle \mid \langle x \rangle \in R(X) \wedge \langle x \rangle \notin S(X) \}$$

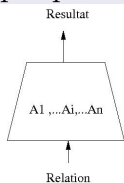
# Opérateurs relationnels

## Symboles de l'algèbre relationnelle

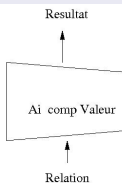
### Représentation Graphique :



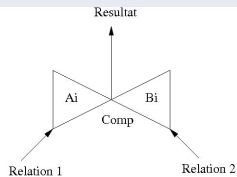
ensemblistes



Projection



Restriction



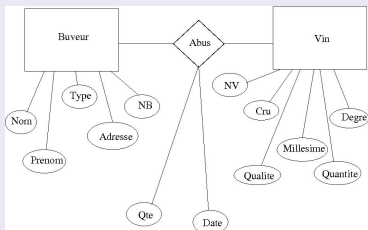
Jointure

# Langage Algèbre

## Représentation Graphique

Interrogation par les opérateurs de l'algèbre relationnelle

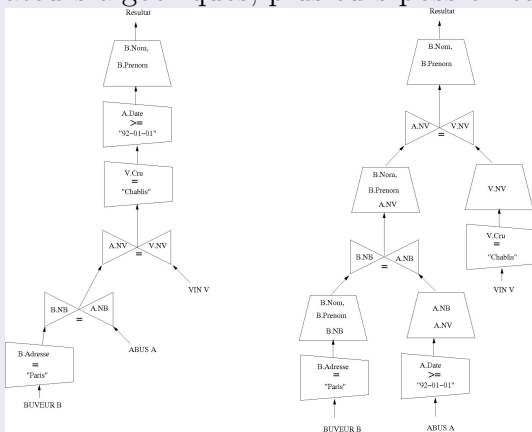
**Nom** et **Prénom** des BUVEUR habitant "PARIS"  
ayant bu (ABUS) du VIN "chablis" depuis le 01-01-92



# Langage Algèbre

## Représentation Graphique

Arbres d'opérateurs algébriques, plusieurs possibilités :



⇒ Optimisation de requêtes

# Bibliographie

## Livres

- **N. Boudjlida** : “base de Données et systèmes d’information”  
Editions Dunod (1999)
- **J.L. Hainaut** : “Bases de Données et modèles de calcul”  
Edition Dunod (2000)
- **G. Gardarin** : “Bases de Données : objet et relationnel”  
Edition Eyrolles (1999)
- **C.J. Date** : “Introduction aux Bases de Données”  
Editions Vuibert (2000)
- **R.K. Stephens, R.R. Plew** : “Conception de Bases de Données”  
Éditions Campus Press (2001)

# Bibliographie

## Adresses “au Net”

- [www.postgresql.org](http://www.postgresql.org) : le site officiel
- [www.lamsade.dauphine.fr/~manouvri](http://www.lamsade.dauphine.fr/~manouvri) : Maude Manouvrier ([www.librecours.org](http://www.librecours.org))
- [sqlpro.developpez.com/biblio/SQL\\_bibl.html](http://sqlpro.developpez.com/biblio/SQL_bibl.html) : des références SQL
- [georges.gardarin.free.fr](http://georges.gardarin.free.fr) : le site de Georges Gardarin
- [www.bd.enst.fr/tablegenerale.html](http://www.bd.enst.fr/tablegenerale.html) : les SGBD à l'ENSTB
- [www.developpez.com](http://www.developpez.com) : entre autre du SQL et des SGBD ...
- [www.reseaucerta.org](http://www.reseaucerta.org) : Michel Auguste, modélisation de données