

Base de données sql

L3 Informatique Semestre 5

Cours donné par Rédigé par Antoine de ROQUEMAUREL

Table des matières

1	Con	acept de base de données et fonction d'un SGBD	3
	1.1	Concepts	3
	1.2	Objectifs d'une base de données	3
	1.3	Avantages	3
	1.4	Le SGBD	4
	1.5	Principale fonction d'un SGBD	4
	1.6	Types d'utilisateurs d'un SGBD	5
2	L'al	gèbre relationnelle	6
	2.1	Les opérations ensemblistes	6
	2.2	Les opérations spécifiques	8
	2.3	Les opérations dérivées	9
	2.4	Expression de l'algèbre relationnelle	10
_			
3	Dép	pendances fonctionnelles	12
3	Dé p		12 12
3	-	Problèmes posés par une mauvaise perception du réel	
3	3.1	Problèmes posés par une mauvaise perception du réel	12 12
3	3.1	Problèmes posés par une mauvaise perception du réel	12 12 13
3	3.1 3.2 3.3	Problèmes posés par une mauvaise perception du réel	12 12 13
3	3.1 3.2 3.3 3.4	Problèmes posés par une mauvaise perception du réel	12 12 13 15
	3.1 3.2 3.3 3.4 3.5 3.6	Problèmes posés par une mauvaise perception du réel	12 12 13 15 16
	3.1 3.2 3.3 3.4 3.5 3.6	Problèmes posés par une mauvaise perception du réel Approche par décomposition Notion de dépendances fonctionnelles Formes Normales (NF) Algorithme de décomposition 3NF Propriétés d'une décomposition en 3NF	12 12 13 15 16

Concept de base de données et fonction d'un SGBD

1.1 Concepts

Definition 1.1 Un base de données est un ensemble structurée de données enregistrées sur des supports accessibles par l'ordinateur pour satisfaire simultanément plusieurs utilisateurs.

Les utilisateurs n'ont pas la même vue de la donnée, ce qui induit l'administrateur.



On construit une base de données en vue de traiter plusieurs application utilisant plusieurs programmes spécifiques.

1.2 Objectifs d'une base de données

Centralisation de l'information La centralisation de l'information permet de réduire les couts matériels, réduire les couts humains ¹ et augmente la lisibilité.

Assurer l'indépendance données / programmes Lors de la modification de la base de données il faut essayer de conserver les programmes d'application.

Indépendance physique Modifier la représentation physique des données sans changer les programmes.

Indépendance logique Modification du schéma conceptuel sans changer le programme.

Permettre les liaisons entre les données

Intégrité des données Il faut qu'a tout moment les données soient cohérentes

Droits d'accès Offrir les moyens de gérer les conflits. Partage + Concurrence \Rightarrow conflits

1.3 Avantages

- La redondance est réduite.
- Les données peuvent être partagées
- Construction d'application sur des données existantes
- Maintient de l'intégrité
- Amélioration de la sécurité et de la disponibilité des données : Garantir une exploitation continue face aux pannes

^{1.} Saisie qu'une fois de l'information

- Résolution des conflits
- Les données sont relativement indépendantes des programmes

1.4 Le SGBD

Definition 1.2 Le Système de Gestion de Base de Données est un ensemble de programmes permettant à un utilisateur d'interagir avec une base de données

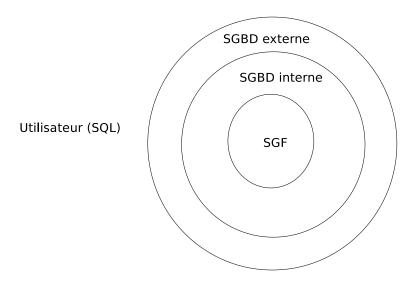


Figure 1.1 – Système de Gestion de Base de Données

R

Le SGBD est un écran entre les usagers et les mémoires secondaires.

Cet ensemble de programmes permet :

- La description des données
- L'accès aux données
- La mise à jour des données (insertions, modifications, destructions)
- La réalisation d'associations entre les données
- Le maintient de l'intégrité
- La sécurité d'exploitation

1.5 Principale fonction d'un SGBD

1.5.1 Fonction de description – Description des données (LDD)

On doit pouvoir définir les entités se rapportant à un monde réel bien précis, préciser les attributs et les liaisons entre les entités. Le LDD est un outil à la disposition de l'administrateur.

1.5.2 Fonction de Manipulation – Manipulation des données (LMD)

La structure de la base de données étant décrite :

- Stocker / Changer les données
- Accéder aux enregistrements pour les mettre à jour
- Interroger la base de données

1.5.3 Autres fonctions

Sécurité, intégrité, ...

1.6 Types d'utilisateurs d'un SGBD

Différents rôles que doivent jouer une personne ou un groupe de personnes pour concevoir, créer, mettre en œuvre et exploiter une base de données :

Administrateur

- Description formelle de la base de données
- Création des schémas externes pour les applications
- Définition des droits d'accès
- Spécifier les organisations physiques et méthodes d'accès utilisées dans l'optique de garantir les meilleures performances
- Définir les procédures de sécurité

Administrateur d'application Il est chargé de décrire la portion de la base de données concernée par une application.

Utilisateur

L'algèbre relationnelle

2.1 Les opérations ensemblistes

Ce sont les opérations qui sont directement issues de la théorie des ensembles.

2.1.1 Union

Definition 2.1 Opération portant sur deux relations de même schéma R_1 et R_2 consistant à construire une relation R_3 de même schéma et ayant pour tuples ceux appartenant à R_1 , R_2 ou aux deux relations

Notation 2.1. $R_3 = Union(R_1, R_2)$

\mathbf{Cru}	Millésime	Région	Couleur
Chablis	2008	Bourgogne	Blanc
Tavel	2010	Rhône	Rosé

Table
$$2.1 - Vins_1$$

\mathbf{Cru}	Millésime	Région	Couleur
Lirac	2009	Rhône	Rouge
Tavel	Tavel 2010		Rosé

Table
$$2.2 - Vins_2$$

$$Vins_3 = Union(Vins_1, Vins_2)$$

\mathbf{Cru}	Millésime	Millésime Région				
Chablis	2008	Bourgogne	Blanc			
Tavel	2010	Rhône	Rosé			
Lirac 2009		Rhône	Rouge			
There 9.2 IV:						

Table $2.3 - Vins_3$

2.1.2 Différence

Definition 2.2 Opération portant sur deux relations de même schéma R_1 et R_2 consistant à construire une relation R_3 de même schéma ayant pour tuples ceux appartenant à R_1 et n'appartenant pas à R_2 .

Notation 2.2. $R = difference(R_1, R_2)$

\mathbf{Cru}	Millésime	Région	Couleur
Chablis	2008	Bourgogne	Blanc
Tavel	2010	Rhône	Rosé

Table $2.4 - Vins_1$

Cru	Millésime	Région	Couleur
Lirac	2009	Rhône	Rouge
Tavel	2010	Rhône	Rosé

Table $2.5 - Vins_2$

 $Vins_3 = difference(Vins_1, Vins_2)$

\mathbf{Cru}	Millésime	Région	Couleur			
Chablis	2008	Bourgogne	Blanc			
T_{1} T_{2} T_{3} T_{4} T_{4}						

Table $2.6 - Vins_3$

2.1.3 Produit

Definition 2.3 Opération portant sur deux relations R_1 et R_2 consistant à construire une relation R_3 ayant pour schéma la concaténation de ceux de R_1 et R_2 et pour tuples les combinaisons des tuples de R_1 et R_2 .

Notation 2.3. $R_3 = pc(R_1, R_2)$

Cru	Millésime	Région	Couleur
Chablis	2008	Bourgogne	Blanc
Tavel	2010	Rhône	Rosé
	Table 2	$2.7-Vins_3$	

Cru	Millésime	Région	Couleur
Lirac	2009	Rhône	Rouge
Tavel	2010	Rhône	Rosé
Table $2.8 - Vins_2$			
1	$Vins_6 = produ$	$it(Vins_1, V$	$ins_2)$
~	3 543 4 4	54.	
 Cru	Millésime	Région	Couleur
Chablis	2008	Bourgogn	e Blanc
Tavel	2010	Rhône	Rosé
Lirac	2009	Rhône	Rouge
Tavel	2010	Rhône	Rosé
Tarif $20 - Vinsa$			

2.2 Les opérations spécifiques

2.2.1 Projection

Definition 2.4 Opération sur une relation R_1 consistant à construire une relation R_2 en enlevant de R_1 tous les attributs non mentionnés en opérande.

Notation 2.4. $R_2 = \prod_{attr_1, attr_2, \dots, attr_n} (R_1)$

$$vins_7 = \Pi_{cru,region}(vins_6)$$
 $\begin{array}{c|c} \mathbf{Cru} & \mathbf{R\acute{e}gion} \\ \hline \mathbf{Chablis} & \mathbf{Bourgogne} \\ \hline \mathbf{Tavel} & \mathbf{R\acute{h}\^{o}ne} \\ \mathbf{Lirac} & \mathbf{R\acute{h}\^{o}ne} \\ \hline \mathbf{Tavel} & \mathbf{R\acute{h}\^{o}ne} \\ \hline \mathbf{Tavel} & \mathbf{R\acute{h}\^{o}ne} \\ \hline \end{array}$
 $\mathbf{Table} \ 2.10 - Vins_7$

2.2.2 Restriction

Definition 2.5 Opération sur une relation R_1 produisant une relation R_2 de même schéma mais comportant uniquement les tuples vérifiant la condition booléenne précisée en argument.

Notation 2.5. $R_2 = \sigma_{condition}(R_1)$

$Vins_8 = \sigma_{region='Rhone'}(vins_6)$				
	Cru	Millésime	Région	Couleur
	Tavel	2010	Rhône	Rosé
	Lirac	2009	Rhône	Rouge
TABLE $2.11 - Vins_8$				

2.3 Les opérations dérivées

2.3.1 Jointure

Definition 2.6 Opération consistant à rapprocher les tuples de deux relations R_1 et R_2 afin de former une relation R_3 dont les attributs sont l'union de attributs de R_1 et R_2 et un tuple de R_2 vérifiant la condition précisée en argument.

Notation 2.6. $R_3 = join(R_1, R_2, cond)$

$$Vins_{10} = join(Vins_7, Vins_9, vins_7, cru = vins_9.cru)$$

$$\begin{array}{c|cccc} \mathbf{Cru} & \mathbf{Mill\acute{e}sime} & \mathbf{R\acute{e}gion} & \mathbf{Couleur} \\ \hline \mathbf{Chablis} & 2008 & \mathbf{Bourgogne} & \mathbf{Blanc} \\ \hline \mathbf{Tavel} & 2010 & \mathbf{Rh\^{o}ne} & \mathbf{Ros\'{e}} \\ \end{array}$$

L'opération de jointure est dérivée de l'opération de multiplication suivie d'une restriction : $join(R_1, R_2, cond)$ $\sigma_{cond}(pc(R_1, R_2))$

2.3.2 Intersection

Definition 2.7 Opération portant sur deux relations R_1 et R_2 de même schéma consistant à construire une relation R_3 de même schéma ayant pour tuples ceux appartenant à R_1 et appartenant à R_2 .

Notation 2.7. $R_3 = intersect(R_1, R_2)$

L'opération d'intersection est dérivée de deux différences : $intersect(R_1, R_2) = difference(R_1, difference(R_1, R_2)).$

2.3.3 Division

Definition 2.8

$$Q = D/d = \{ \langle a_1, a_2, \cdots, a_p \rangle / \forall \langle a_p 1, a_{p_2}, \cdots, a_n \rangle \in d$$

$$\langle a_1, a_2, \cdots, a_p, a_{p+1}, a_{p+2}, \cdots, a_n \rangle \in D \}$$

Notation 2.8.

NumC	nom
1	bdd
2	Archi
2	Graphe

Table 2.13 - cours

numC	numE
1	1
2	1
3	1
1	2
3	2

Table 2.14 - suit

$$T = suit/\Pi_{numC}(cours)$$

Table 2.15 - T

« Quels sont les numéros d'étudiants qui suivent tous les cours? »

2.4 Expression de l'algèbre relationnelle

A partir de l'algèbre relationnelle il est possible de composer un langage algébrique.

2.4.1 Opérateur algébriques

Comment obtenir les couleurs de vins de cru Morgon ou Volnay?

Deux solutions sont possibles:

2.4.1.1 Opérateur algébriques

$$T_1 = \sigma_{cru='morgon'}(vins)$$

$$T_2 = \sigma_{cru='volnay'}(vins)$$

$$T_3 = union(T_1, T_2)$$

$$resultat = \Pi_{couleur}(T3)$$

2.4.1.2 Langage algébrique

Une autre solution, plus performante permettant de ne pas utiliser de variables temporaires en utilisant le langage algébrique :

$$\Pi_{couleur}(union(\sigma_{cru='morgon'}(vins), \sigma_{cru='volvay'}(vins)))$$

2.4.1.3 Arbre algébrique

celui-ci peut aussi être représentée sous forme d'un arbre relationnel. Les nœud correspondent au représentation graphiques des opérations et les arcs aux flots de données entre les opérations.

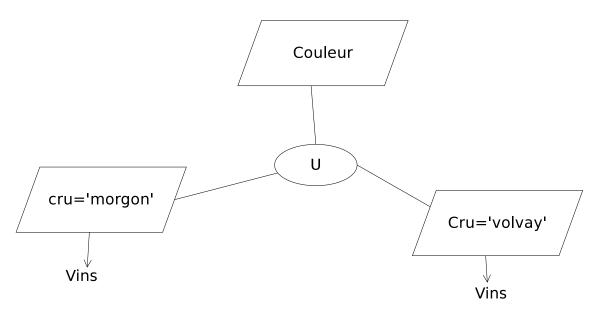


FIGURE 2.1 – Arbre relationnel

Dépendances fonctionnelles

3.1 Problèmes posés par une mauvaise perception du réel

\mathbf{numIm}	Marque	Type	Puissce	Couleur	numSecu	nom	prenom	\mathbf{date}	prix
61A3631	Renaut	clio	4	Rouge	1	Dupont	Pierre	10.2.13	10500
80XA631	Renaut	clio	4	bleue	1	Dupont	Pierre	11.6.13	11600
31AA31	Citroëne	C5	9	bleue	2	Martin	Jacques	22.07.13	22000
12X531	Citroëne	C3	4	verte	2	Martin	Jacques	13.09.13	9800

Table 3.1 – Relation propriétaire

La relation propriétaires souffre de plusieurs anomalies :

- 1. **Données redondantes** Risque d'incohérence, par exemple si Jacques Martin veut changer de nom, il faudra changer tous les tuples
- 2. Il est nécessaire d'autoriser les valeurs NULL afin de pouvoir conserver des voitures sans propriétaires ou des personnes ne possédant pas de voitures.
- 3. Risque de perte de données la suppression d'un tuple peut entrainer la perte d'un type de voiture ou d'un propriétaire.

3.2 Approche par décomposition

L'approche par décomposition, pour concevoir des schémas relationnels tend à partir d'une relation universelle à découper cette relation en sous relation qui ne souffriraient pas des anomalies précédentes.

Definition 3.1 La décomposition est le remplacement d'une relation $R(A_1, A_2, \dots, A_n)$ par une cllection de relations $R_1, R_2, \dots R_m$ obtenus par des projections de R et tels que la relation résultat des jointures $R_1 \bowtie R_2 \bowtie \dots \bowtie R_m$ ait le même schéma que R.

numIm		Marque	Type	Puissce	Couleur						
-	872RH75		Clio	6	Verte	_					
31AA31		Renaut Renaut	Clio	6	Rouge						
numI 872RH 31AA	75 Clio	Table 3.2 Couleur Verte Rouge	- <u>t</u>	ype Ma	e r que Pui ault	ssance 6					
Table 3.3 – Décomposition 1 $T_1 = R_1 \bowtie R_2$											
numIm 872RH7 31AA31	5 Člio	_	arque	_	uissance 6 6	Couleur Verte Rouge					
$T_2 = \bowtie v_1 \bowtie v_2 \bowtie v_3$											
$\phantom{aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa$		Marque	Type	Puissce	Couleur	_					
	872RH75	Renaut	Clio	6	Verte						
	31AA31	Renaut	Clio	6	Rouge						
		Table 3.4	– Décor	mposition 2	2						

Perte d'information sur la couleur.

La décomposition 1 permet de retrouver l'information par jointure alors que la décomposition 2 ne permet pas de retrouver la couleur d'une voiture.

Definition 3.2 La décomposition sans perte d'informations, ou spi est la décomposition d'une relation R en R_1, R_2, \dots, R_n par toute extension de R, on ait $R =_1 \bowtie R_2 \bowtie \dots \bowtie R_4$.

3.3 Notion de dépendances fonctionnelles

La notion de dépendances fonctionnelles (df) permet de caractériser les relations qui peuvent être décomposée sans perte d'information.

Definition 3.3 Dépendance fonctionnelle : soit $R(A_1, R_2, \dots, R_n)$ un schéma de relation et X et Y des sous-enembles de $\{A_1, A_2, \dots, A_n\}$. On dit que $X \to Y$ (x détermine Y) si pour tout extensions r de R, pour tout tuple t_1 et t_2 de r on a $\Pi_X(t_1) = \Pi_X(t_2) \Rightarrow \Pi_Y(t_1) = \Pi_Y(t_2)$

(R)

 $X \to \! Y$ si étant donné une valeur de X, il lui correspond une valeur unique de Y

 $\begin{array}{ll} numIm \to Type & Type \to Puisance \\ numIn \to Couleur & Type \to marque \end{array}$

 $Type,Marque \rightarrow puissance$

Par contre on a pas Puissance \rightarrow type.

Il est possible de visualiser cet ensemble de df par un graphe appelé graphe de dépendance fonctionnelle.

3.3.1 Théorèmes

3.3.1.1 Axiomes d'Amstrong

Réfléxivité $Y\subseteq X\Rightarrow X\to Y$ Tout ensemble d'attribut détermine les même où une partie de lui même

Augmentation $X \to Y \Rightarrow X, Z \to Y, Z$ si $X \to Y$ les deux ensembles d'attributs peuvent être enrichis par une troisième

Transitivité: $X \to Y$ et $Y \to Z \Rightarrow X \to Z$ Cette règle provient du fait que le composé de deux fonctions dont l'image de l'une et le domaine de l'autre est une fonction.

3.3.1.2 Règles déduites

Plusieurs règles se déduisent de ces axiomes :

Union $X \to Y$ et $X \to Z \Rightarrow X \to y, Z$

Pseudo transitivité $X \rightarrow Y$ et $W, Y \rightarrow Z \Rightarrow W, X \rightarrow Z$

Décomposition $X \to Y$ et $Z \subseteq \Rightarrow X \to Z$

Definition 3.4 Une dépendance fonctionnelle élémentaire est une dépendance de a forme $X \to A$, ou A est un attribut unique $\notin X$ et où $\not\exists X \subset X$ tel que $X \to A$.

type \rightarrow puissance n'est pas élémentaire



La seule règle d'inférance qui s'applique au dépendances fonctionnelles et la transitivté.

Definition 3.5 Une clé est un sous ensemble X des attributs d'une relation $R(A_1, A_2, \dots, A_n)$ tel que $X \to A_1, A_2, \dots, A_n$ et $\not\exists Y \subset Y$ tel que $Y \to A_1, A_2, \dots, A_n$.

Definition 3.6 La fermeture transitive est un ensemble de dépendances fonctionnelles considéré enrichi de toutes les dépendances fonctionnelles déduites par transitivité.

Notation : La fermeture transitive de F est noté F^+

Definition 3.7 2 ensembles F_1 et F_2 sont équivalent ssi $F_1^+ = F_2^+$

Definition 3.8 La convention minimale est un ensemble F de dfe associé à un ensemble d'attributs vérfiant les propriétés suivantes :

- 1. Aucune df n'est redondante $\forall f \in F, F.f$ n'est pas équivalent à F.
- 2. Toute dfe est dans F^+

Proposition Si une relation R(X, Y, Z) possède une dépendance $X \to Y$ alors R est décomposable sans perte d'information en $R_1(X, Y)$ et $R_2(X, Z)$.

3.4 Formes Normales (NF)

Definition 3.9 Une relation est en première forme normale si tout attriut contient une valeur atomique, éviter le domaine composé de plusieurs valeurs

Definition 3.10 Une relation R est en 2eme forme normale si :

- Elle est en 1NF
- Tout attribut n'appartenant pas à une clé ne dépend pas d'une partie de la clé

Fournisseur(nom, article, adresse, prix)

 $d1 : nom, article \rightarrow prix$ $d2 : nom \rightarrow adresse$

Cette relation n'est pas en 2NF à cause de d2

Definition 3.11 Une relation est en 3e forme normal si et seulement si :

- Elle est en 2NF
- Tout attribut n'appartenant pas à une clé ne dépend pas d'un attribut n'appartenant pas à un clé.

voitures(numIm, type, marque)

 $d1 : numIm \rightarrow type$ $d2 : type \rightarrow marque$

Voiture n'est pas en 3NF à cause de d2.

Definition 3.12 Une relation est en 3NF BKC, Bayo-Codd Kent, si et seulement si les seules différences sont celles dans lesquelles une clé détermine un attribut.

3.5 Algorithme de décomposition 3NF

Pour toute relation, il existe au moins une décomposition en 3NF préservant les dépendances fonctionnelles sans perte d'information. Le but d'un algorithme de décomposition en 3NF est de convertir un schéma de relation qui n'est pas 3NF en un ensemble de schémas 3NF.

Le principe de l'algorithme consiste à appliquer les règle e décomposition suivantes tel que les schémas ne sont pas 3NF :

Schéma non 2NF R(k1, k2, x, y); La relation doit être décomposée en $R1(\underline{k2}, X)$ et $R2(\underline{k1,k2}, Y)$

Schéma non 3NF R(k,x,y,z), la relation doit être décomposée en $R1(\underline{X},Z)$ et $R2(\underline{k},x,y)$

- Schéma non 2NF

3.6 Propriétés d'une décomposition en 3NF

Les dépendances fonctionnelles des règles indépendants du temps qui doivent vérifier les attributs, il est nécessaire qu'une décomposition préserve les règles

Definition 3.13 Une décomposition $\{R_1, R_2, \dots, R_n\}$ d'une relation R tel que la fermeture transitive de la df de R est le même que l'union des dépendances fonctionnelles de $\{R_1, R_2, \dots, R_n\}$.

Dans toute relation, il existe au moins une décomposition en 3NF préservant les dépendances fonctionnelles sans perte d'information. Le but d'un algorithme de décomposition en 3NF est de convertir un schéma de relation qui n'est pas 3NF.

Exercices

A.1 Algèbres relationnel

A.1.1 Opérateurs algébriques

```
vins(numv, cru, mill, region, degré)
buveurs(nums, nom, prenom, ville)
abus(nomv, nomb, date, quantite, place)
```

Avec les opérateurs algébriques et uniquement les opérateurs de base.

A.1.1.1 Donner le degré des vins de cru Morgon et Millésime 2001

$$T_1 = \sigma_{cru='morgon' \ and \ mill=2001}(vins)$$

$$T_2 = \pi_{degr\acute{e}}(T_1)$$

A.1.1.2 Numéro des buveurs de Chenas

$$T_1 = \sigma_{cru='chenas'}$$

 $T_2 = pc(T_1, abus)$
 $T_3 = \sigma_{T_1.numV=abus.numV}(T_2)$
 $Res = \Pi_{numB}(T_3)$

A.1.1.3 Nom et prénom des buveurs de chénon et de Tariquet



Autorisation d'utiliser le join

```
T_1 = \sigma_{cru='chenon'\ ou\ cru='tariquet'}(vins)

T_2 = join(T_1, abus, T_2.numV = abus.numV)

T_3 = join(T_2, buveurs, T_2.numB = buveurs.numB)

Res = \Pi_{nom.prenom}(T_3)
```

A.1.1.4 Noms des buveurs ayant bu uniquement du Tavel

```
T_{1} = \sigma_{cru='Tavel'}(vins)
T_{2} = \sigma_{cru\neq tavel}(vins)
T_{3} = join(abus, T_{1}, t_{1}.numv = abus.numv)
T_{4} = join(abus, T_{2}, T_{2}.numv = abus.numv)
T_{5} = \Pi_{numB}(T_{3})
T_{6} = \Pi_{numB}(T_{4})
T_{7} = difference(T_{5}, T_{6})
T_{8} = join(T_{7}, buveurs, T_{7}.numB = buveurs.num7)
Res = \Pi_{nom}(T_{8})
```

A.1.1.5 Écrire A.1.1.4 en langage algébrique

```
\Pi_{nom}(join(difference(\Pi_{numB}(join(abusa, \sigma_{cru='Tavel'}(vins), abus.numv = vins.numv)),
\Pi_{numB}(join(abus, \sigma_{cru\neq'tavel'(vins)}, abus.numv = vins.numv))),
buveurs, a.numB = buveurs.numB))
```

A.1.2 Arbres algébriques

```
Trafic(nTrain, nLige, gare)
Trains(nTrains, nRegion)
lignes(nLignes, rang, gare)
wagons(nWagon, type, poidsVide, capacie, etat)
```

- A.1.2.1 Types de wagons du train 4001
- A.1.2.2 Numéro et types des wagons des ligne 111 et 112
- A.1.2.3 Numéro des wagons communcs aux lignes 111 e 112
- A.1.2.4 Numéro des trains ayant au moins 2 wagons vides
- A.1.2.5 Liste des gares de dessertes pour le train 4001
- A.1.2.6 Numéro des lignes qui partent de la gare de Toulouse

A.2 Dépendances fonctionnelles

A.2.1

```
\begin{array}{lll} R1 &=& < u1, F1 > \\ u1 &=& \{nuPiece, pri, tva, libell\'e, cat\'egorie\} \\ F1 &=& \{d1: nuPiece \rightarrow prix, d2: nuPiece \rightarrow libelle, \\ && d3: nF \rightarrow catalogue, d4: catalogue \rightarrow tva, d5: nP \rightarrow tva\} \end{array}
```

- A.2.1.1 Forme normale de R_1 ?
- A.2.1.2 Proposer un schéma en 3NF