

Secteur Tertiaire Informatique Filière étude - développement

Activité « Développer la persistance des données »

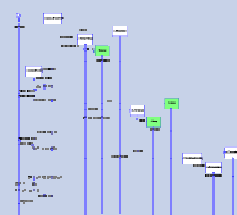
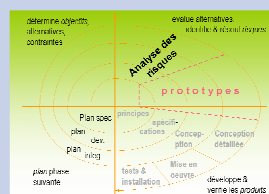
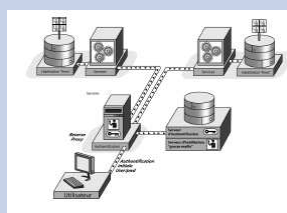
Algèbre relationnelle : Présentation

Accueil

Apprentissage

Période en
entreprise

Evaluation



Code barre

SOMMAIRE

I	Structure des données d'un SGBD relationnel	4
I.1	La notion de domaine	4
I.2	La notion de relation (ou table)	4
I.3	La notion d'attribut	5
II	DEFINITION DE L'ALGEBRE RELATIONNEL.....	6
III	BASE DE TRAVAIL.....	7
IV	OPERATEURS UNAIRES RELATIONNEL	9
IV.1	Sélection.....	9
IV.2	Projection.....	9
V	OPERATEURS ENSEMBLISTES.....	10
V.1	Intersection	10
V.2	Union	11
V.3	Différence	12
VI	OPERATEURS BINAIRES RELATIONNELS	14
VI.1	Le produit cartésien	14
VI.2	La jointure	15
VI.3	La division.....	17

I STRUCTURE DES DONNEES D'UN SGBD RELATIONNEL

Rappelons comment peut être défini simplement le modèle relationnel. Une base de données relationnelle se présente comme **un ensemble de relations** que l'on appelle couramment des **tables**. Toute relation a un **schéma** qui décrit sa structure. Le schéma est composé d'un ensemble **d'attributs** (on utilise souvent le terme de **colonnes**) dont un sous-ensemble constitue la **clé**. La clé permet d'identifier les autres attributs de la relation. Les valeurs des attributs sont atomiques, c'est-à-dire non décomposables, et appartiennent au même domaine.

Cependant, pour être capable de manipuler correctement les données d'une base relationnelle, des définitions plus précises et rigoureuses sont nécessaires. En fait, c'est à partir de la **théorie des ensembles**, base formelle sur laquelle repose le modèle relationnel, que se construisent les différents concepts.

I.1 LA NOTION DE DOMAINE

Le premier concept à définir est celui qui permet de préciser les **ensembles de départ**, ou **domaines**.

Un domaine représente **l'ensemble des valeurs admissibles** pour un composant d'une relation. Il est caractérisé par un **nom**.

Comme tout ensemble, un domaine peut être défini en **extension** (c'est-à-dire en donnant la liste exhaustive de toutes les valeurs qui le composent), ou en **intension** (en définissant une propriété caractéristique du domaine, d'ailleurs on parle aussi de **définition**).

Par exemple, il est identique de parler de l'ensemble {1,3,5,7,9} ou de l'ensemble {entiers impairs positifs et inférieurs à 10}.

Ce concept de domaine est primordial pour assurer un contrôle lors de la manipulation de la base : toute valeur n'appartenant pas au même domaine de définition doit se voir rejetée.

I.2 LA NOTION DE RELATION (OU TABLE)

Pour appréhender facilement cette notion, nous allons prendre l'exemple suivant :

si l'on considère le domaine **COULEUR = { rouge, vert, jaune }**
et le domaine **FORME = { cube, sphère, cône }**,
le produit cartésien des deux domaines est composé des neuf tuples (ou enregistrements) qui sont :

{ cube rouge, cube vert, cube jaune, sphère rouge, sphère verte,...}.

Au sens mathématique de Codd, **la relation (ou table) peut être définie comme un sous-ensemble, caractérisé par un nom, du produit cartésien d'une liste de domaines**.

On utilise généralement une représentation commode sous forme d'une table à deux dimensions, où **chaque ligne correspond à un tuple** (enregistrement) alors que **chaque colonne correspond à un domaine** du produit cartésien utilisé.

Dans la liste de domaines, un même domaine peut bien sûr apparaître plusieurs fois.

Par exemple, nous pouvons composer la relation suivante :

cube rouge,
cube jaune,
sphère rouge,
sphère vert,
sphère jaune,
cône rouge,
qui matérialise le fait que tous les objets ne sont pas construits dans toutes les couleurs.

I.3 LA NOTION D'ATTRIBUT

Le concept d'attribut a été introduit pour expliquer le **rôle joué par un domaine** dans une relation : il **permet de distinguer les différentes colonnes, tout en permettant plusieurs colonnes du même domaine**. Un attribut est une colonne d'une relation caractérisée par un nom.

Par exemple, une relation VOL peut être constituée d'un n° de vol, d'une ville de départ, d'une ville d'arrivée. Vous remarquerez que ces deux derniers attributs appartiennent au même domaine "**villes desservies par un aéroport**".

Remarque :

Les tuples correspondent à un ensemble de valeurs prises par les attributs d'une table pour représenter une entité, un objet ou un lien entre les objets du monde réel. Pour un objet donné, et à un instant donné, nous pouvons ne pas disposer de la totalité des informations permettant la description de cet objet ou de ce lien. Autrement dit, les valeurs de quelques attributs peuvent avoir des valeurs "inconnues" : elles sont dites NULLES (cela s'écrit **NULL**).

II DEFINITION DE L'ALGÈBRE RELATIONNEL

L'algèbre relationnelle est un concept mathématique de relation de la théorie des ensembles.

L'algèbre relationnelle a été inventée en 1970 par Edgar Frank Codd, le directeur de recherche du centre IBM de San José. Elle est constituée d'un ensemble d'opérations formelles sur les relations. Les opérations relationnelles permettent de créer une nouvelle relation (table) à partir d'opérations élémentaires sur d'autres tables (par exemple l'union, l'intersection, ou encore la différence).

Les principes de l'algèbre relationnelle sont beaucoup utilisés de nos jours par les SGDB pour la gestion des bases de données relationnelles.

Ce chapitre présente les opérateurs utilisés en algèbre relationnelle. Pour plus de précisions, vous pouvez consulter le cours « 3- Algèbre relationnelle - Compléments.doc ».

III BASE DE TRAVAIL

Tous les exemples et exercices dans ce cours concerneront la base de données suivante :

Vous pouvez accéder à la DESCRIPTION (structure de la table) en tapant la commande suivante : **DESC** avion ou **DESC** vol ou **DESC** pilote

Cette commande vous donne la liste des ATTRIBUTS et leur TYPE sous forme tabulaire.

AVION(AV#, AVMARQ, AVNOM, CAP, LOC)

PILOTE(PIL#, PILNOM, ADR)

VOL(VOL#, PIL#, AV#, VD, VA, HD, HA)

TABLE AVION (appelée aussi Relation AVION) :

AV# : numéro d'avion

AVMARQ : marque de l'avion

AVNOM : type de l'avion

CAP : capacité en nb de passagers

LOC : ville où est basé l'avion

N°	Marque	Type	Capacité	Localisation
AV#	AVMARQ	AVNOM	CAP	LOC
100	AIRBUS	A320	300	Nice
101	BOIENG	B707	250	Paris
102	AIRBUS	A320	300	Toulouse
103	CARAVELLE	Caravelle	200	Toulouse
104	BOEING	B747	400	Paris
105	AIRBUS	A320	300	Grenoble
106	ATR	ATR42	50	Paris
107	BOEING	B727	300	Lyon
108	BOEING	B727	300	Nantes
109	AIRBUS	A340	350	Bastia

TABLE PILOTE

PIL# : numéro du pilote

PILNOM : nom du pilote

ADR : adresse du pilote

N°	Nom	Adresse
PIL#	PILNOM	ADR
1	SERGE	Nice
2	JEAN	Paris
3	CLAUDE	Grenoble
4	ROBERT	Nantes
5	MICHEL	Paris
6	LUCIEN	Toulouse
7	BERTRAND	Lyon
8	HERVE	Bastia
9	LUC	Paris

TABLE VOL

VOL# : numéro du vol

AV# : numéro de l'avion

PIL# : numéro du pilote

VD : ville départ

VA : ville d'arrivée

HD : heure de départ

HA : heure d'arrivée

N°	Avion	Pilote	Ville Départ	Ville Arrivée	Heure Départ	Heure Arrivée
VOL#	AV#	PIL#	VD	VA	HD	HA
IT100	100	1	NICE	PARIS	7	9
IT101	100	2	PARIS	TOULOUSE	11	12
IT102	101	1	PARIS	NICE	12	14
IT103	105	3	GRENOBLE	TOULOUSE	9	11
IT104	105	3	TOULOUSE	GRENOBLE	17	19
IT105	107	7	LYON	PARIS	6	7
IT106	109	8	BASTIA	PARIS	10	13
IT107	106	9	PARIS	BRIVE	7	8
IT108	106	9	BRIVE	PARIS	19	20
IT109	107	7	PARIS	LYON	18	19
IT110	102	2	TOULOUSE	PARIS	15	16
IT111	101	4	NICE	NANTES	17	19

IV OPERATEURS UNAIRES RELATIONNEL

Les opérateurs **unaires** sont les opérateurs qui ne font intervenir qu'**une seule table**.

IV.1 SELECTION

Elle opère un **découpage horizontal** de la table.

Certains enregistrements et tous les attributs.

Syntaxe: R2 = Sélection de R1 (A1="condition de restriction")

Exemples :

Liste des avions avec toutes les informations

R1 = sélection AVION (*)

Liste des vols pour Nice

R1 = sélection VOL (va=nice)

Liste des avions qui ont plus de 200 places

R1 = sélection AVION (cap >200)

IV.2 PROJECTION

Elle opère un **découpage vertical** de la table.

Certains attributs et tous les enregistrements.

Syntaxe: R2 = Projection de R1 (A1="Attribut à projeter ")

Exemples :

Liste de tous les numéros d'avions

R1 = projection AVION (av#)

Liste des noms des pilotes

R1 = projection PILOTE (pilnom)

Liste des marques d'avions

R1 = projection AVION (avmarq)

V OPERATEURS ENSEMBLISTES

V.1 INTERSECTION

Est utilisé pour relier 2 tables uni compatibles, c'est à dire ayant le même nombre d'attributs et des attributs respectifs définis sur le même domaine.

Pour l'intersection il faut une **structure identique**.

R1 : TABLE AVION AIRBUS

AV#	AVMARQ	Ville Départ	Ville Arrivée
100	AIRBUS	NICE	PARIS
100	AIRBUS	PARIS	TOULOUSE
105	AIRBUS	GRENOBLE	TOULOUSE
105	AIRBUS	TOULOUSE	GRENOBLE
109	AIRBUS	BASTIA	PARIS
102	AIRBUS	TOULOUSE	PARIS

R2 : TABLE AVION PARIS

AV#	AVMARQ	Ville Départ	Ville Arrivée
100	AIRBUS	NICE	PARIS
107	BOEING	LYON	PARIS
109	AIRBUS	BASTIA	PARIS
106	ATR	BRIVE	PARIS
102	AIRBUS	TOULOUSE	PARIS

R3: Intersection de R1 et R2

Syntaxe: R3 = Intersection (R1, R2)

AV#	AVMARQ	Ville Départ	Ville Arrivée
100	AIRBUS	NICE	PARIS
109	AIRBUS	BASTIA	PARIS
102	AIRBUS	TOULOUSE	PARIS

Exemples :

Liste des avions AIRBUS allant à Paris

R1 = sélection AVION (avmarq=AIRBUS)	// tous les avions Airbus
R2 = projection R1 (av#, avmarq)	// projection des attributs
R3 = sélection VOL (va='Paris')	// tous les vols vers Paris
R4 = jointure R3 (av#=av#) AVION	// vols des avions pour Paris
R5 = projection R4 (av#, avmarq)	// projection des attributs
R6 = Intersection (R2, R5)	// intersection des 2 relations

V.2 UNION

Est utilisé pour relier 2 tables uni compatibles, c'est à dire ayant **le même nombre d'attributs** et des attributs respectifs définis sur le même domaine : il faut une **structure identique**.

R1 : TABLE AVION PARIS NICE

VOL	Avion	Pilote	Ville Départ	Ville Arrivée	Heure Départ	Heure Arrivée
IT102	101	1	Paris	Nice	12	14
IT118	256	1024	Paris	Nice	9	10
IT112	812	2048	Paris	Nice	12	14
IT123	999	1246	Paris	Nice	19	20

R2 : TABLE AVION BRIVE MARSEILLE

VOL	Avion	Pilote	Ville Départ	Ville Arrivée	Heure Départ	Heure Arrivée
IT99	999	2	Brive	Marseille	10	13

R3: Union de R1 et R2

Syntaxe : R3 =Union (R1, R2)

VOL	Avion	Pilote	Ville Départ	Ville Arrivée	Heure départ	Heure Arrivée
IT99	999	2	Brive	Marseille	10	13
IT102	101	1	Paris	Nice	12	14
IT118	256	1024	Paris	Nice	9	10
IT112	812	2048	Paris	Nice	12	14
IT123	999	1246	Paris	Nice	19	20

Exemples:

Liste des vols Paris-Nice et Toulouse-Paris

R1 = sélection VOL (vd=PARIS et va=NICE)

R2 = sélection VOL (vd=TOULOUSE et va=PARIS)

R3 = Union (R1, R2)

Liste des avions Airbus et Boeing

R1 = sélection AVION (avmarq=AIRBUS)

R2 = sélection AVION (avmarq=BOEING)

R3 = Union (R1, R2)

Liste des Airbus ou des avions de plus de 200 places

R1 = sélection AVION (avmarq=AIRBUS)

R2 = sélection AVION (cap > 200)

R3 = Union (R1, R2)

V.3 DIFFERENCE

Est utilisé pour relier 2 tables uni compatibles, c'est à dire ayant le même nombre d'attributs et des attributs respectifs définis sur le même domaine.

R1 : TABLE AVION AIRBUS

AV#	AVMARQ	Ville Départ	Ville Arrivée
100	AIRBUS	NICE	PARIS
100	AIRBUS	PARIS	TOULOUSE
105	AIRBUS	GRENOBLE	TOULOUSE
105	AIRBUS	TOULOUSE	GRENOBLE
109	AIRBUS	BASTIA	PARIS
102	AIRBUS	TOULOUSE	PARIS

R2 : TABLE AVION PARIS

AV#	AVMARQ	Ville Départ	Ville Arrivée
100	AIRBUS	NICE	PARIS
107	BOEING	LYON	PARIS
109	AIRBUS	BASTIA	PARIS
106	ATR	BRIVE	PARIS
102	AIRBUS	TOULOUSE	PARIS

R3 : Différence entre R1 et R2

Syntaxe: R3 = Différence (R1, R2)

AV#	AVMARQ	Ville Départ	Ville Arrivée
100	AIRBUS	PARIS	TOULOUSE
105	AIRBUS	GRENOBLE	TOULOUSE
105	AIRBUS	TOULOUSE	GRENOBLE

R4: Différence entre R2 et R1

Syntaxe: R4 = Différence (R2, R1)

AV#	AVMARQ	Ville Départ	Ville Arrivée
107	BOEING	LYON	PARIS
106	ATR	BRIVE	PARIS

Exemples :

Liste des Airbus qui ne vont pas à Paris

R1 = sélection AVION (avmarq='AIRBUS') // tous les Airbus
R2 = projection R1 (av#)
R3 = sélection VOL (va='PARIS') // tous les avions qui vont à Paris
R4 = projection R3 (av#)
R5 = différence (R1, R4) // tous les airbus qui ne vont pas à Paris

VI OPERATEURS BINAIRES RELATIONNELS

Les opérateurs binaires sont ceux qui font intervenir deux tables.

VI.1 LE PRODUIT CARTESIEN

Il consiste à rassembler deux tables en une seule.

R1

A
A1
A2
A3

R2

B
B1
B2

R3: Produit cartésien de R1 et R2

Syntaxe: $R3 = \text{Produit cartésien}(R1, R2)$

A	B
A1	B1
A2	B1
A3	B1
A1	B2
A2	B2
A3	B2

VI.2 LA JOINTURE

La **jointure** consiste à rechercher entre **deux tables ayant un attribut commun** (même type et même domaine de définition) tous les tuples (enregistrements) pour lesquels ces attributs ont la **même valeur**.

La jointure entre les tables VOL et PILOTE consiste à rechercher toutes les informations du pilote de chaque avion. Concrètement elle réalise la concaténation des tuples (ou n-uplets) de Avion avec Pilote chaque fois que AVION.PIL# = VOL.PIL#.Vol

Vol	Av	Pil	VD	VA	HD	HA
IT101	100	2	PARIS	TOULOUSE	11	12
IT100	100	1	NICE	PARIS	7	9
IT102	101	1	PARIS	NICE	12	14
IT103	105	3	GRENOBLE	TOULOUSE	9	11
IT104	105	3	TOULOUSE	GRENOBLE	17	19
IT105	107	7	LYON	PARIS	6	7
IT106	109	8	BASTIA	PARIS	10	13

Pilote

Pil	Nom
1	SERGE
2	JEAN
3	CLAUDE
7	BERTRAND
8	HERVE

R: Jointure de Vol et Pilote :

Syntaxe: R = jointure Vol (Pil = Pil) Pilote

Vol	Av	Pil	VD	VA	HD	HA	Nom
IT101	100	2	PARIS	TOULOUSE	11	12	JEAN
IT100	100	1	NICE	PARIS	7	9	SERGE
IT102	101	1	PARIS	NICE	12	14	SERGE
IT103	105	3	GRENOBLE	TOULOUSE	9	11	CLAUDE
IT104	105	3	TOULOUSE	GRENOBLE	17	19	CLAUDE
IT105	107	7	LYON	PARIS	6	7	BERTRAND
IT106	109	8	BASTIA	PARIS	10	13	HERVE

Dans notre exemple, la jointure est possible par l'attribut PIL présent dans VOL et PILOTE.

Exemples :

Liste des avions AIRBUS allant à Paris :

R1 = sélection VOL (va=Paris)

R2 = sélection AVION (avmarq=AIRBUS)

R3 = jointure R1(av#=av#) R2

Nom des pilotes en service :

On sait qu'un pilote est en service s'il effectue des vols.

R1 = sélection PILOTE (*) // on sélectionne tous les pilotes

R2 = jointure R1 (pil#=pil#) VOL // on effectue une jointure avec vol

R3 = projection R2(pilnom) // on affiche le nom des pilotes

Nom des pilotes assurant un vol au départ de Paris :

R1 = sélection VOL (vd=Paris)

R2 = jointure R1 (pil#=pil#) PILOTE

R3 = projection R2 (pilnom)

Nom des pilotes conduisant un Airbus :

R1 = sélection AVION (avmarq=AIRBUS)

R2 = jointure R1 (av# = av#) VOL

R3 = jointure R2 (pil# = pil#) PILOTE

R4 = projection R3 (pilnom)

VI.3 LA DIVISION

R1

B	A
B1	A1
B1	A2
B2	A2
B3	A1
B4	A1
B4	A2

R2

A
A1
A2

R3: Division de R1 et R2

Syntaxe: R3 = division (R1, R2)

B
B1
B4

La division consiste à trouver la liste des valeurs d'un attribut de **R1** *tel qu'il existe un tuple dans R1 pour chaque valeur de R2* pour l'attribut correspondant.

Exemples :

Liste des pilotes qui conduisent tous les avions

Liste des avions qui vont dans toutes les villes

Etablissement référent
Centre Paul et Liliane GUINOT

Equipe de conception
Véronique Brugnion
Philippe Bouget

Remerciements :

Reproduction interdite

Article L 122-4 du code de la propriété intellectuelle.
« toute représentation ou reproduction intégrale ou partielle faite sans le
consentement de l'auteur ou de ses ayants droits ou ayants cause est
illicite. Il en est de même pour la traduction, l'adaptation ou la reproduction
par un art ou un procédé quelconques. »

Date de mise à jour 05/05/2008
afpa © Date de dépôt légal mai 08



afpa / Direction de l'Ingénierie 13 place du Générale de Gaulle / 93108 Montreuil
Cedex
association nationale pour la formation professionnelle des
adultes
Ministère des Affaires sociales du Travail et de la
Solidarité