

# ESIR1 BD

## Bases de données

### calcul relationnel

Olivier Ridoux

# Plan

- Notions de base
  - schéma, relation, table, requête
- Opérations relationnelles

# Notions de base

# Calcul relationnel

- Un calcul dont les valeurs sont des **relations** (ou **tables**)
- Un calcul formalisé par **l'algèbre relationnelle (AR)**
- Un calcul implémenté dans le **langage de manipulation de données** (LMD) de SQL
  - présenter **AR** et **SQL** en //

# Une table

## Types of broadband connections people use at home

% of those with broadband at home

	DSL	Cable	Fixed wireless or satellite	Fiber	T-1	Other
2009	33%	41%	17%	5%	1%	2%
2008	46	39	11	3	*	1
2007	49	39	8	1	*	1

Source: Pew Internet & American Life Project April 2009 Survey.



<http://www.pewinternet.org/2009/06/17/connections-costs-and-choices/>

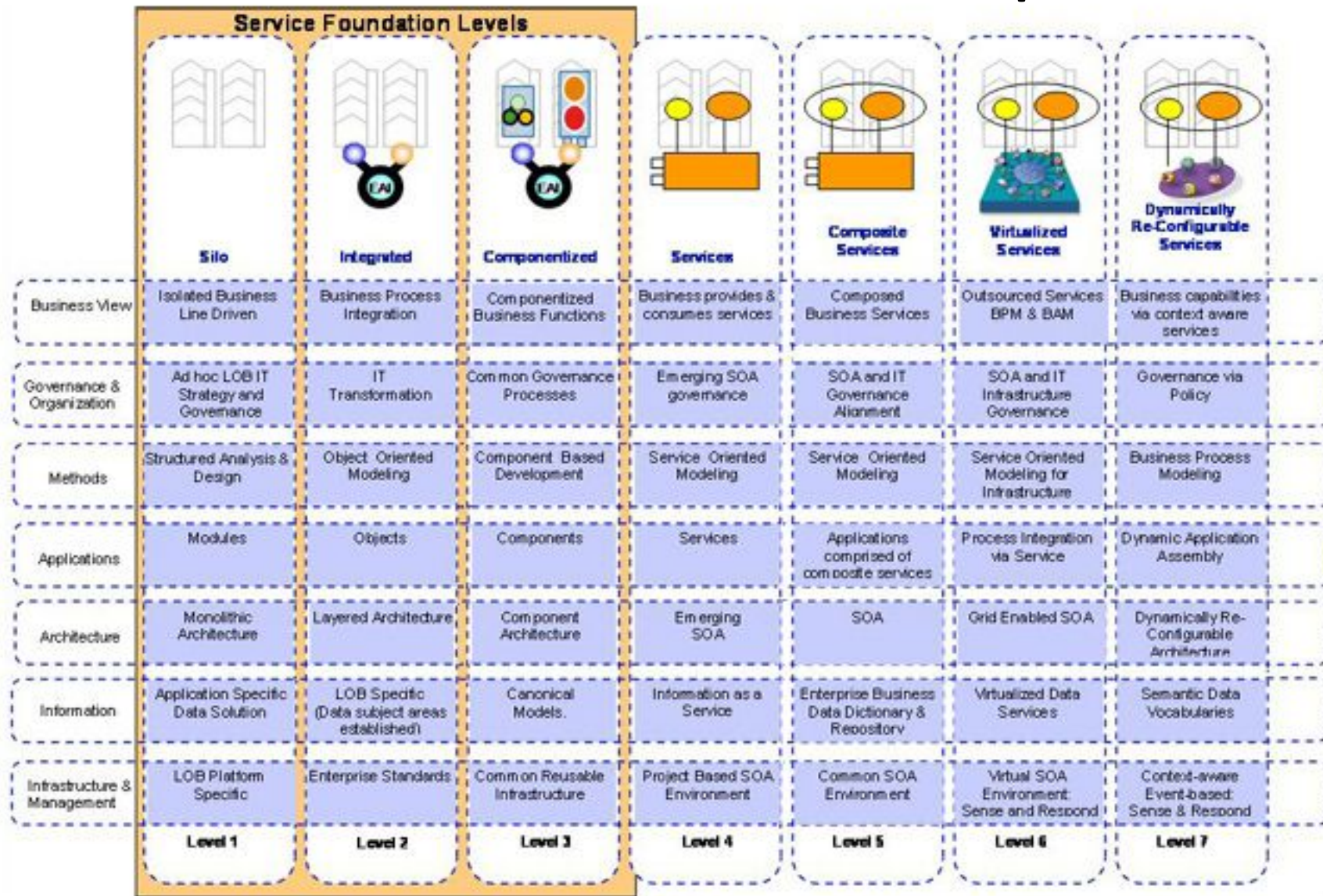
# Une autre table

	UWB	Bluetooth	Wi-Fi	Wi-Fi	Wi-Fi	WiMAX	WiMAX	Edge	CDMA2000 / 1x EV-DO	WCDMA / UMTS
Standard	802.15.3a	802.15.1	802.11a	802.11b	802.11g	802.16d	802.16e	2.5G	3G	3G
Usage	WPAN	WPAN	WLAN	WLAN	WLAN	WMAN Fixed	WMAN Portable	WWAN	WWAN	WWAN
Throughput	110- 480Mbps	Up to 720Kbps	Up to 54Mbps	Up to 11Mbps	Up to 54Mbps	Up to 75Mbps (20MHz BWW)	Up to 30Mbps (10MHz BWW)	Up to 384Kbps	Up to 2.4 Mbps (typical 300- 600Kbps)	Up to 2Mbps (Up to 10Mbps with HSDPA technology)
Range	Up to 30 feet	Up to 30 feet	Up to 300 feet	Up to 300 feet	Up to 300 feet	Typical 4-6 miles	Typical 1-3 miles	Typical 1-5 miles	Typical 1-5 miles	Typical 1-5 miles
Frequency	7.5GHz	2.4GHz	5GHz	2.4GHz	2.4GHz	Sub 11GHz	2-6GHz	1900MHz	400, 800, 900, 1700, 1800, 1900, 2100MHz	1800, 1900, 2100MHz

<http://www.embedded.com/design/communications-design/4017788/Beyond-3G-The-Changing-Face-of-Cellular>

calcul relationnel

# Une table hors champs



<https://www.opengroup.org/soa/source-book/osimmv2/model.htm>



# Notion de schéma

- **Nom-tab(Nom-attr<sub>1</sub> : Dom-attr<sub>1</sub>,  
Nom-attr<sub>2</sub> : Dom-attr<sub>2</sub>,  
...  
Nom-attr<sub>N</sub> : Dom-attr<sub>N</sub>)**
- Schéma  $\approx$  signature  $\approx$  type
- **CREATE TABLE Nom-tab  
(Nom-attr<sub>1</sub> Dom-attr<sub>1</sub> [opt<sub>1</sub>],  
Nom-attr<sub>2</sub> Dom-attr<sub>2</sub> [opt<sub>2</sub>],  
...  
Nom-attr<sub>N</sub> Dom-attr<sub>N</sub> [opt<sub>N</sub>]);**



# Quelques requêtes (1)

```
CREATE TABLE Adherent (  
  NumAdherent INT PRIMARY KEY,  
  Nom CHAR(20),  
  Prenom CHAR(20),  
  Telephone CHAR(20),  
  DateAdhesion DATETIME ,  
  Sexe CHAR(1) )
```

- Déclaration d'un **schéma**

## Quelques requêtes (2)

```
CREATE TABLE Activite(  
    NumActivite INT PRIMARY KEY,  
    Intitule CHAR(20),  
    NumResponsable INT )
```

```
CREATE TABLE AdherentActivite(  
    NumAdherent INT,  
    NumActivite INT,  
    Expertise INT )
```

# Représentation graphique du schéma

Adherent

IdAdherent	Nom	Prenom	...

AdherentActivite

NumAdhérent	NumActivité	Expertise

Activite

NumActivite	NumResponsable	Intitulé

calcul relationnel

# Notion de **relation** et de **table** (1)

- **Relation** = ensemble de n-uplets conformes à un schéma  
 $\{(x_1, \dots, x_N), \dots, (z_1, \dots, z_N)\}$
- **Pas d'ordre, pas de multiplicité**
- Relation  $\approx$  propriété
  - $(x_1, \dots, x_N)$  a la propriété désirée pour être dans cet ensemble

# Notion de **relation** et de **table** (2)

- **Table** = liste de n-uplets conformes à un schéma

**INSERT INTO Nom-tab**

**[(Nom-attr<sub>1</sub> , ... , Nom-attr<sub>N</sub> )]**

**VALUES (x<sub>1</sub>, x<sub>2</sub>, ... , x<sub>N</sub>);**

- Ordre (fortuit ou désiré) et multiplicité (doublons)

# Quelques requêtes (3)

INSERT INTO **Adherent**

VALUES (

17, 'Durand', 'Jean',

'0299123456', '27/01/2015', 'H')

# Quelques requêtes (4)

INSERT INTO **Adherent**

(NumAdherent, Nom, Prenom)

VALUES (123, 'Jones', 'Inigo')

UPDATE **Adherent**

SET Telephone = '0223231234'

WHERE NumAdherent = 123

- Insertion **partielle** et **progressive** des informations



# Notion de requête

- Relations stockées = relations explicites  
(**extensionnelles**)
- Requêtes = relations calculées  
= relations implicites (**intensionnelles**)
- Rôle de l'interpréteur de requêtes  
**implicite → explicite**
- Requêtes utilisent relations explicites / implicites

# Opérations relationnelles

calcul relationnel

17

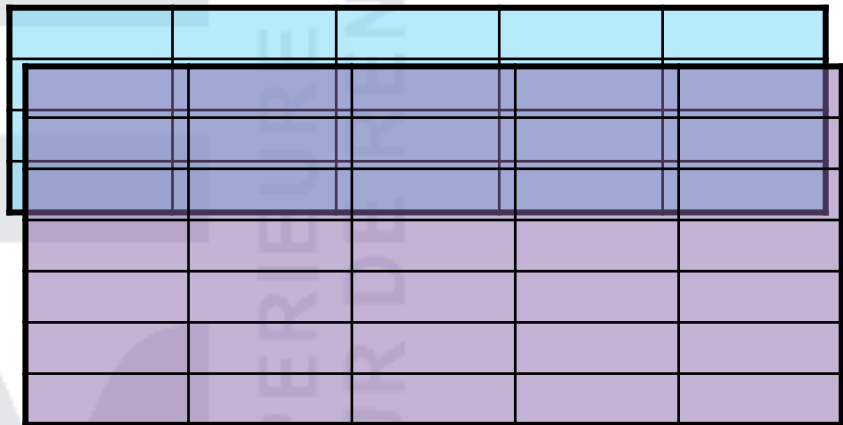
# Opérations relationnelles (1)

- Opérations **ensemblistes**
  - tables calculées
  - tables stockées
- Approximation en **SQL**
  - ordre et multiplicité

# Opérations relationnelles (2)

- Opérations binaires même schéma
  - INTERSECT, UNION, MINUS (EXCEPT)
- Opérations unaires
  - SELECT
- Opérations binaires schémas complémentaires

# INTERSECT (1)







} intersection

- BLEU et VIOLET même schéma !
- $\text{Relation}_1 \cap \text{Relation}_2$
- $\text{Propriété}_1 \wedge \text{Propriété}_2$
- $\text{Table}_1 \text{ INTERSECT } \text{Table}_2$

# INTERSECT (2)

a	1
c	3
e	5
g	7

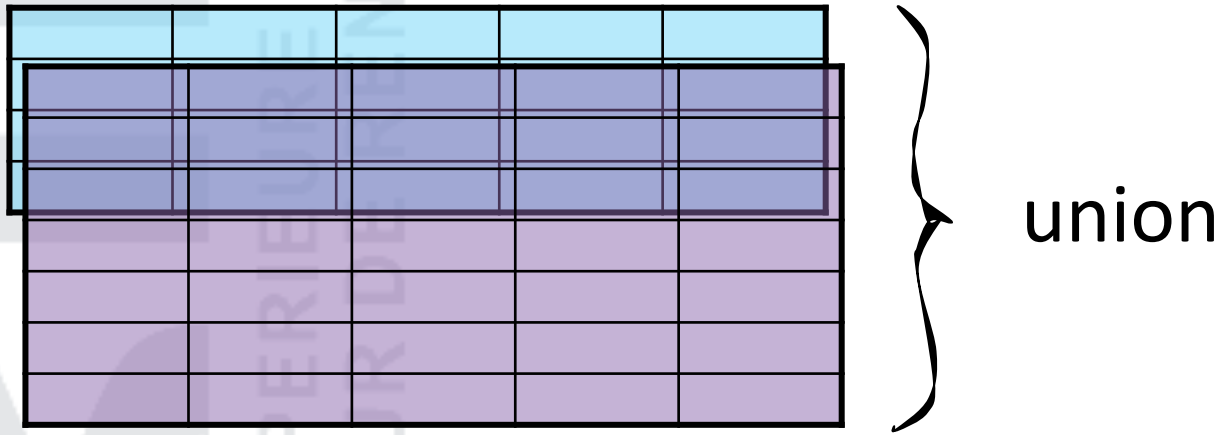
$\cap$

a	1
b	2
d	4
g	7
k	11
p	16
v	22

=

a	1
g	7

# UNION (1)



- **BLEU** et **VIOLET** même schéma !
- **$\text{Relation}_1 \cup \text{Relation}_2$**
- **$\text{Propriété}_1 \vee \text{Propriété}_2$**
- **$\text{Table}_1 \text{ UNION } \text{Table}_2$**



# UNION (2)

a	1
c	3
e	5
g	7

U

a	1
b	2
d	4
g	7
k	11
p	16
v	22

=

a	1
c	3
e	5
g	7
b	2
d	4
k	11
p	16
v	22


ou

a	1
c	3
e	5
g	7
a	1
b	2
d	4
g	7
k	11
p	16
v	22

...

Apparition possible de doublons !

# MINUS - EXCEPT (1)




} soustraction

- BLEU et VIOLET même schéma !
- $\text{Relation}_1 \setminus \text{Relation}_2$
- $\text{Propriété}_1 \wedge \neg \text{Propriété}_2$
- $\neg(\text{Propriété}_1 \Rightarrow \text{Propriété}_2)$
- $\text{Table}_1 \text{ MINUS } \text{Table}_2$

# MINUS - EXCEPT (2)

a	1
c	3
e	5
g	7

\

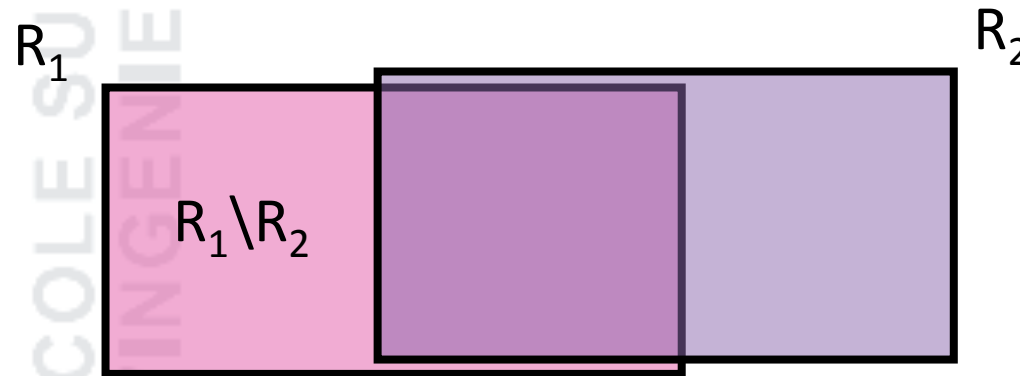
a	1
b	2
d	4
g	7
k	11
p	16
v	22

=

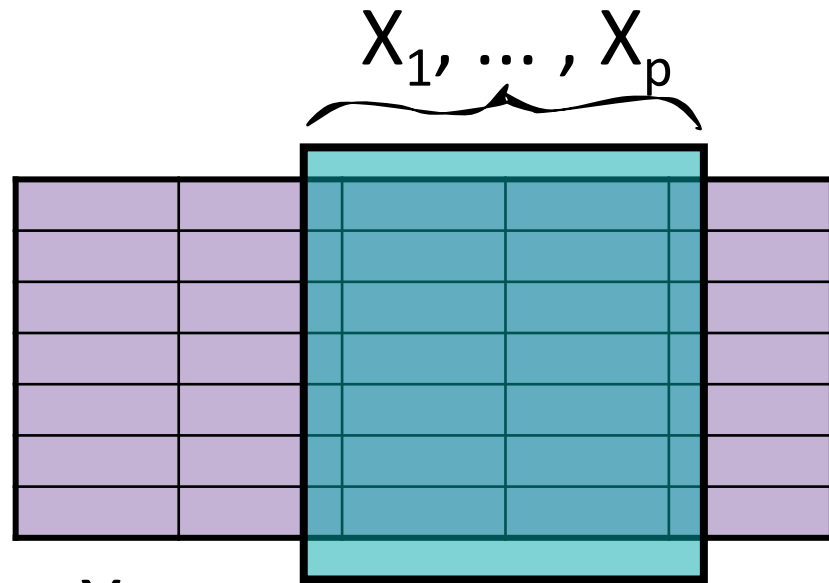
c	3
e	5

# Remarque - $\cap$ vs. MINUS

- $\text{Relation}_1 \cap \text{Relation}_2$   
=  $\text{Relation}_1 \setminus (\text{Relation}_1 \setminus \text{Relation}_2)$   
=  $\text{Relation}_2 \setminus (\text{Relation}_2 \setminus \text{Relation}_1)$



# SELECT – projection (1)



- Projection sur  $X_1, \dots, X_p$
- **$\text{proj}_{X_1 \dots X_p}(\text{Relation})$  ou  $\Pi_{X_1 \dots X_p}(\text{Relation})$**
- $\exists_{Y_1 \dots Y_m} (\text{Propriété}(Y_1, \dots, Y_m, X_1, \dots, X_p))$
- **$\text{SELECT } X_1, \dots, X_p \text{ FROM Relation}$**

# SELECT – projection (2)

$\Pi_{UW}(\text{$

T	U	V	W	X
a	12	1.5	"ac"	q
b	23	2.6	"zv"	s
c	34	3.7	"eb"	d
d	45	4.8	"rn"	f
e	56	5.9	"tb"	g
f	67	6.0	"yv"	h

$\text{) =$

U	W
12	"ac"
23	"zv"
34	"eb"
45	"rn"
56	"tb"
67	"yv"

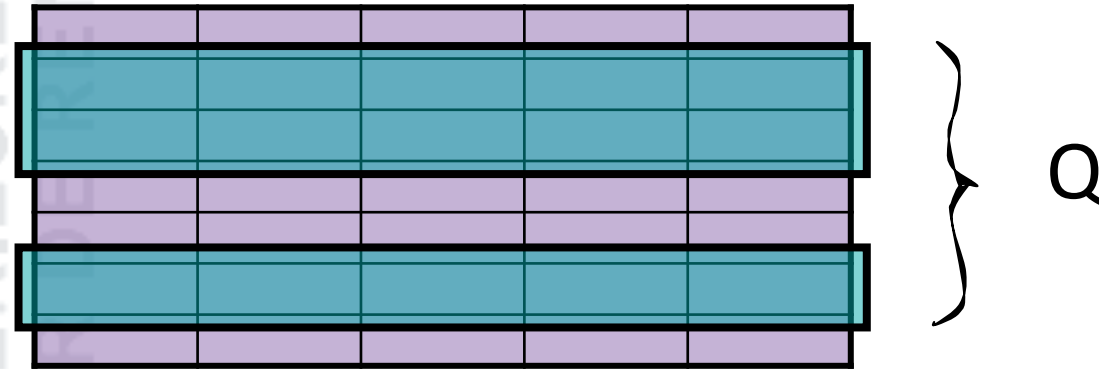
# Quelques requêtes (5)

$\pi_{\text{Nom, Prenom, Telephone}}(\text{Adherent})$

SELECT **Nom, Prenom, Telephone**  
FROM Adherent



# SELECT – restriction (1)



- Restriction / un prédicat  $Q$
- **$\text{select}_Q(\text{Relation})$  ou  $\sigma_Q(\text{Relation})$**
- Propriété  $\wedge Q$
- **$\text{SELECT} * \text{FROM Relation WHERE } Q$**

# Prédicats de restriction (1)

- Propriétés intensionnelles
- Arithmétique et logique  
<, <=, =, >=, >, <>, AND, OR, NOT

⋮  
**WHERE attr < 12 OR attr >= 25**

# Quelques requêtes (6)

$\sigma_{\text{Sexe}='H'}(\text{Adherent})$

```
SELECT *  
FROM Adherent  
WHERE Sexe = 'H'
```

# Quelques requêtes (7)

$\sigma_{\text{DataAdhesion} < 01/01/2000}(\text{Adherent})$

```
SELECT *  
FROM Adherent  
WHERE DateAdhesion < '01/01/2000'
```

- Une date est représentée par une chaîne, mais n'est pas une chaîne  
**'31/12/1999' < '01/01/2000'**

## Prédicats de restriction (2)

- Appartenance
  - BETWEEN, IN, LIKE

...

**WHERE attr NOT BETWEEN 12 AND 24**

...

**WHERE attr IN ('ESIR1', 'ESIR2', 'ESIR3')**

...

**WHERE attr LIKE 'E%'**

# Prédicats de restriction (3)

- Spécial BD relationnelle

**IS NULL, IS NOT NULL**

# Quelques requêtes (8)

$\sigma_{\text{DataAdhesion IS NULL}}(\text{Adherent})$

```
SELECT *  
FROM Adherent  
WHERE DateAdhesion IS NULL
```

- Prise en compte des champs pas encore informés



# SELECT – restriction (2)

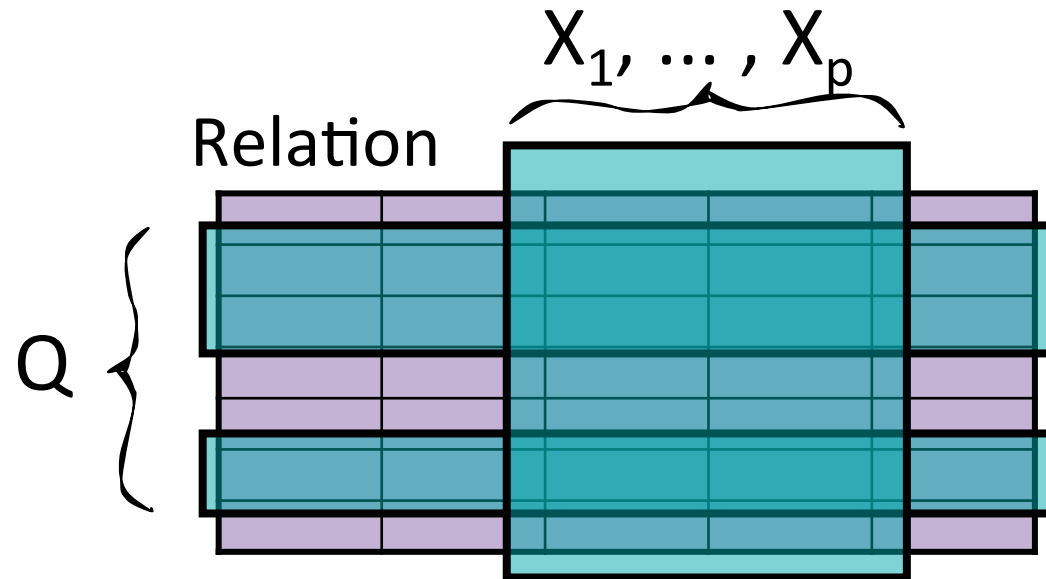
$\sigma_{U>40}()$

T	U	V	W	X
a	12	1.5	"ac"	q
b	23	2.6	"zv"	s
c	34	3.7	"eb"	d
d	45	4.8	"rn"	f
e	56	5.9	"tb"	g
f	67	6.0	"yv"	h

) =

T	U	V	W	X
d	45	4.8	"rn"	f
e	56	5.9	"tb"	g
f	67	6.0	"yv"	h

# SELECT (1)



- **SELECT** projection  $X_1, \dots, X_p$   
**FROM** Relation  
**WHERE** restriction  $Q$
- $\exists_{Y_1 \dots Y_m} (\text{Propriété}(Y_1, \dots, Y_m, X_1, \dots, X_p) \wedge Q(Y_1, \dots, Y_m, X_1, \dots, X_p))$

## SELECT (2)

$\Pi_{UW}(\sigma_{U>40}(\text{R}))$

T	U	V	W	X
a	12	1.5	"ac"	q
b	23	2.6	"zv"	s
c	34	3.7	"eb"	d
d	45	4.8	"rn"	f
e	56	5.9	"tb"	g
f	67	6.0	"yv"	h

$)) =$

U	W
45	"rn"
56	"tb"
67	"yv"

- En général,  $\Pi_X(\sigma_Q(R)) \neq \sigma_Q(\Pi_X(R))$

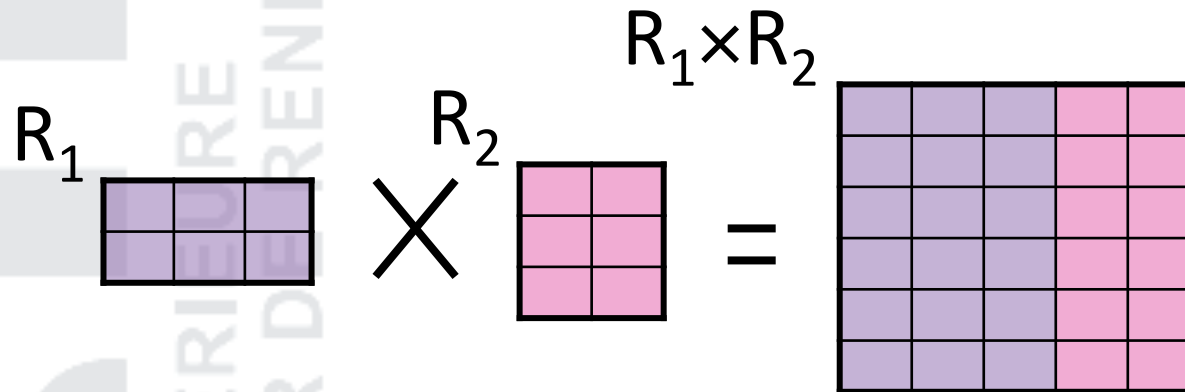
# Quelques requêtes (9)

$\pi_{\text{Nom}}(\sigma_{\text{Sexe}='H'}(\text{Adherent}))$

SELECT DISTINCT **Nom**  
FROM **Adherent**  
WHERE **Sexe** = 'H'

- Requête type

# Produit cartésien (1)



- **$\text{Prod}(R_1, R_2)$  ou  $R_1 \times R_2$**
- Complétion des noms d'attributs si nécessaire

$a \rightarrow R_i.a$

- **$\text{SELECT} * \text{FROM } R_1, R_2$**

# Produit cartésien (2)

X	Y	Z
1	a	"c"
2	z	"v"

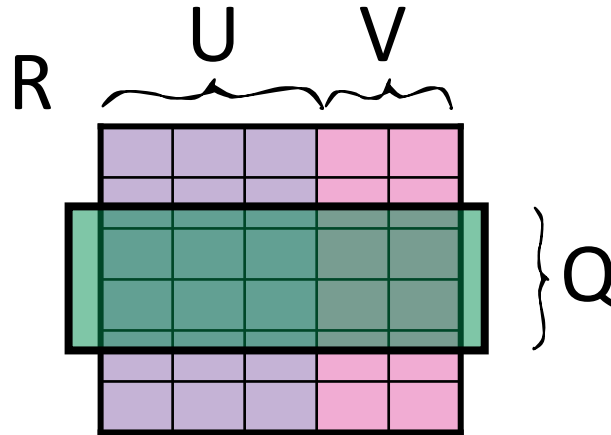


U	V
11	123
22	234
33	345



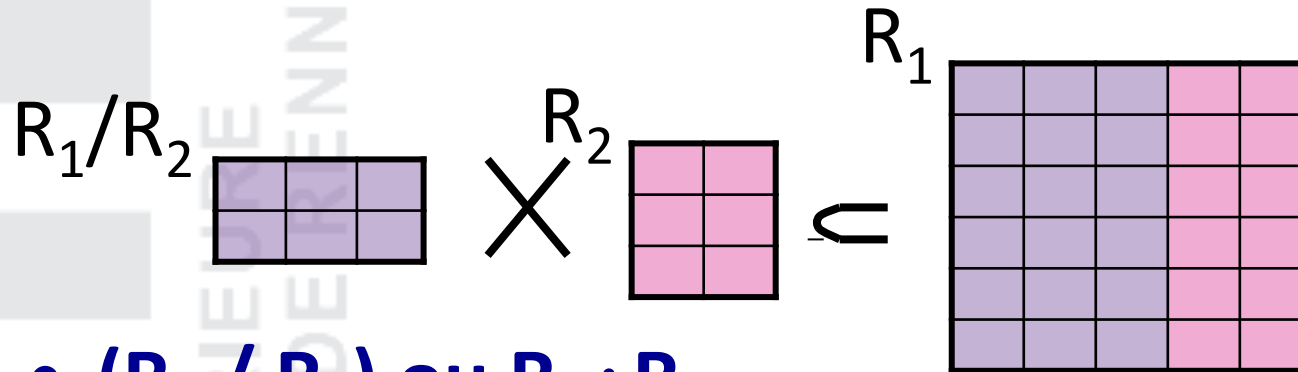
X	Y	Z	U	V
1	a	"c"	11	123
1	a	"c"	22	234
1	a	"c"	33	345
2	z	"v"	11	123
2	z	"v"	22	234
2	z	"v"	33	345

# Remarque - produit cartésien



- En général,  
 $\Pi_U(\sigma_Q(R)) \times \Pi_V(\sigma_Q(R))$   
n'est pas contenu dans R
- Le produit cartésien ne peut pas reconstituer une relation scindée par projection

# Quotient (1)



- $(R_1 / R_2)$  ou  $R_1 \div R_2$
- Tous les  $P$  tels que  $P \times R_2$  est dans  $R_1$   
les  $P$  que tous les  $R_2$  ont
- schéma de  $R_2$  devrait être inclus dans schéma de  $R_1$   
...sinon  $(R_1 / R_2)$  est sûrement vide



# Quotient (2)

1	a
2	b
2	c
3	b
2	a
1	b

$$/ \begin{array}{|c|} \hline a \\ \hline b \\ \hline \end{array} = \begin{array}{|c|} \hline 1 \\ \hline 2 \\ \hline \end{array}$$

## • Vérification

1
2

 $\times$ 

a
b

1	a
2	b
2	a
1	b

reste =

2	c
3	b

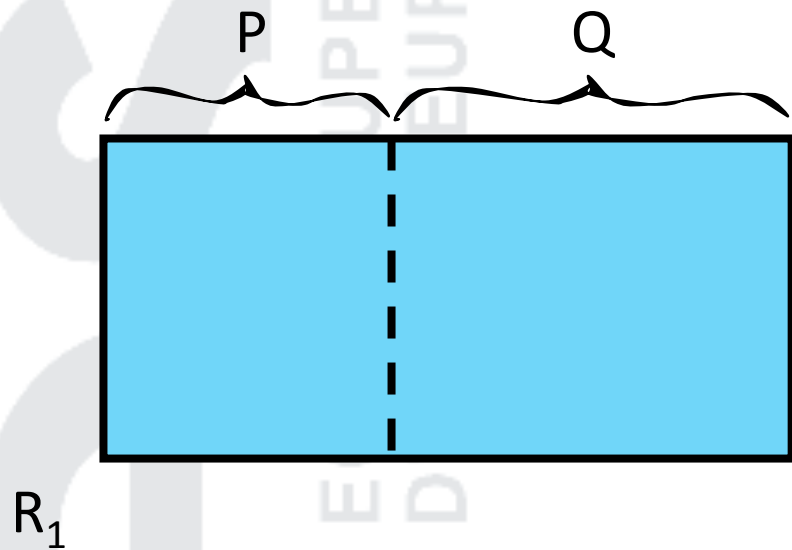
calcul relationnel

# Remarque – quotient (1)

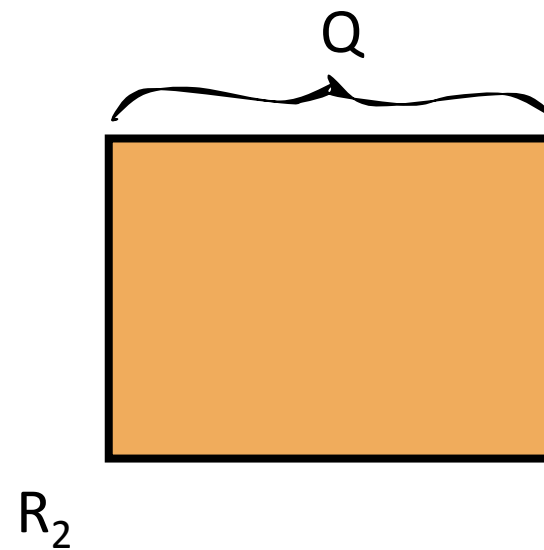
- $(R_1 / R_2) = \Pi_p(R_1) \setminus \Pi_p( (\Pi_p(R_1) \times R_2) \setminus R_1 )$

## Remarque – quotient (2)

- $(R_1 / R_2) = \Pi_P(R_1) \setminus \Pi_P((\Pi_P(R_1) \times R_2) \setminus R_1)$

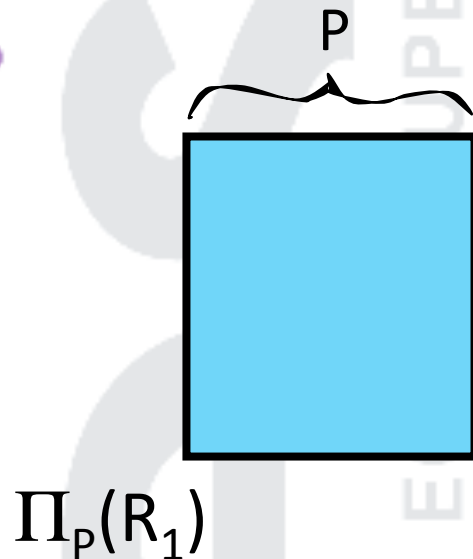


calcul relationnel

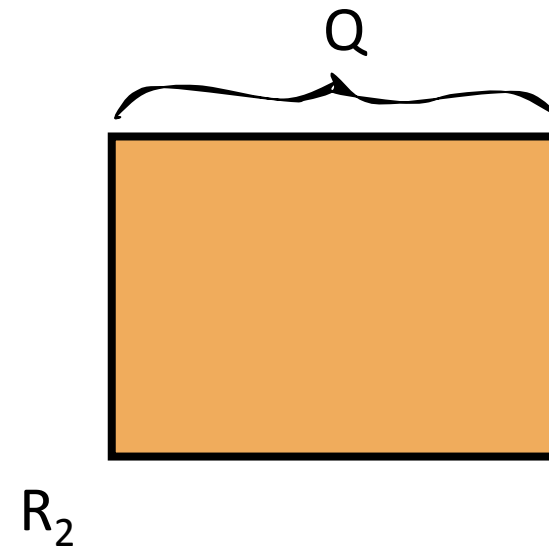


## Remarque – quotient (3)

- $(R_1 / R_2) = \Pi_P(R_1) \setminus \Pi_P( (\Pi_P(R_1) \times R_2) \setminus R_1 )$

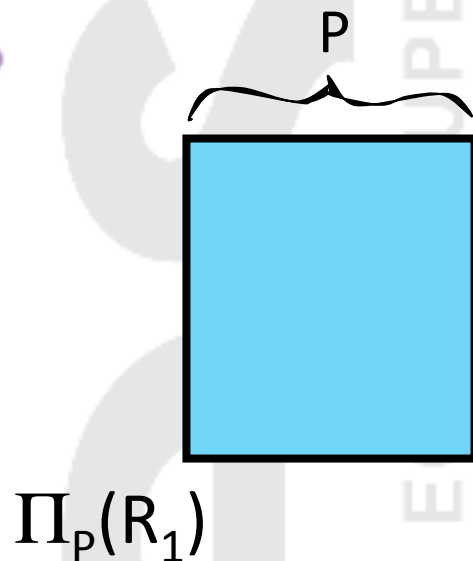


calcul relationnel



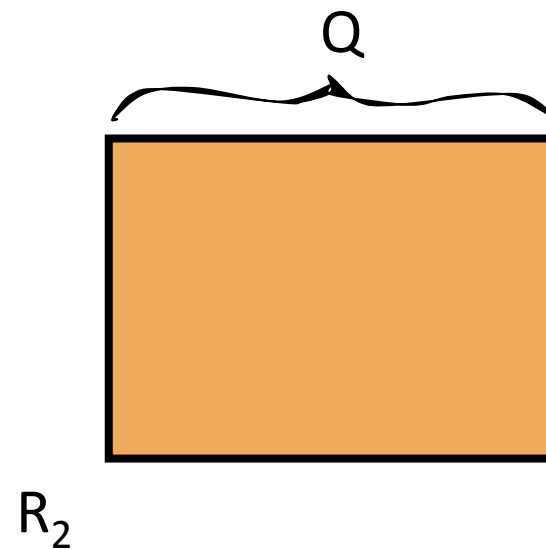
## Remarque – quotient (4)

- $(R_1 / R_2) = \Pi_P(R_1) \setminus \Pi_P( (\Pi_P(R_1) \times R_2) \setminus R_1 )$



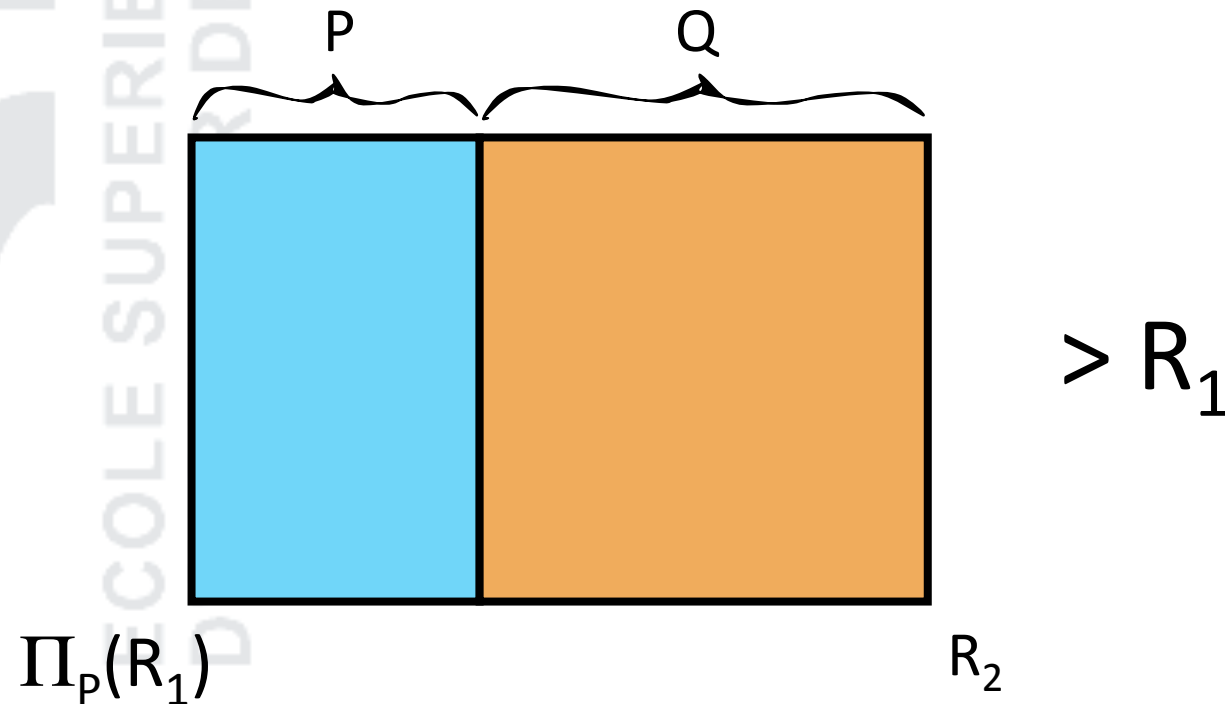
calcul relationnel

$\times$



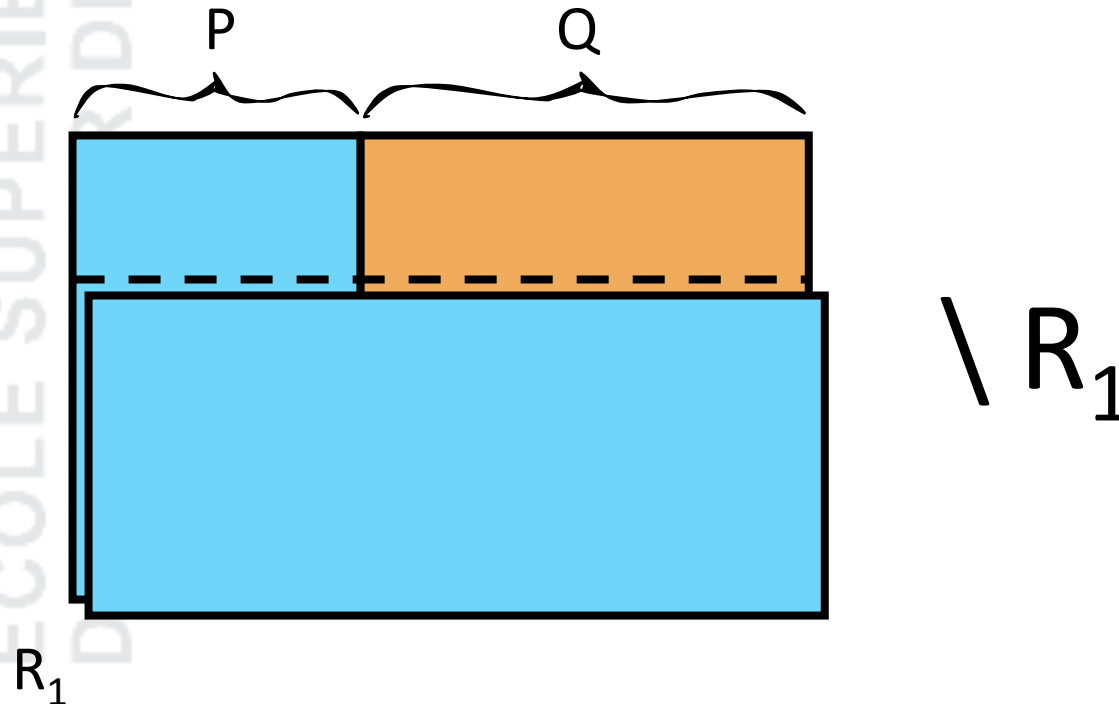
## Remarque – quotient (5)

- $(R_1 / R_2) = \Pi_p(R_1) \setminus \Pi_p((\Pi_p(R_1) \times R_2) \setminus R_1)$



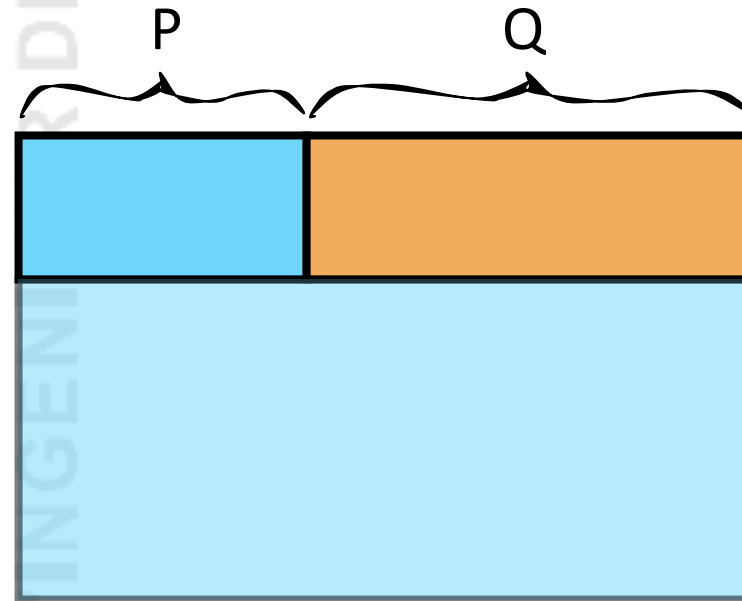
## Remarque – quotient (6)

- $(R_1 / R_2) = \Pi_P(R_1) \setminus \Pi_P((\Pi_P(R_1) \times R_2) \setminus R_1)$



## Remarque – quotient (7)

- $(R_1 / R_2) = \Pi_P(R_1) \setminus \Pi_P((\Pi_P(R_1) \times R_2) \setminus R_1)$

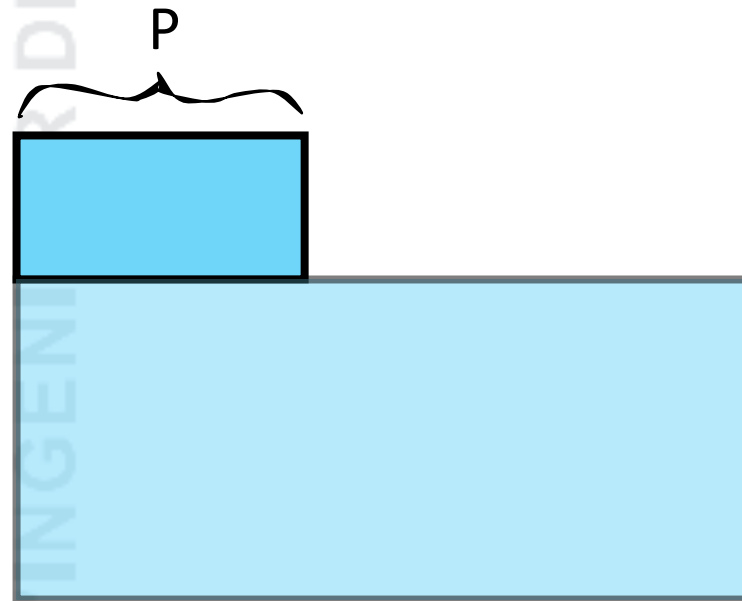


$\notin R_1$



## Remarque – quotient (8)

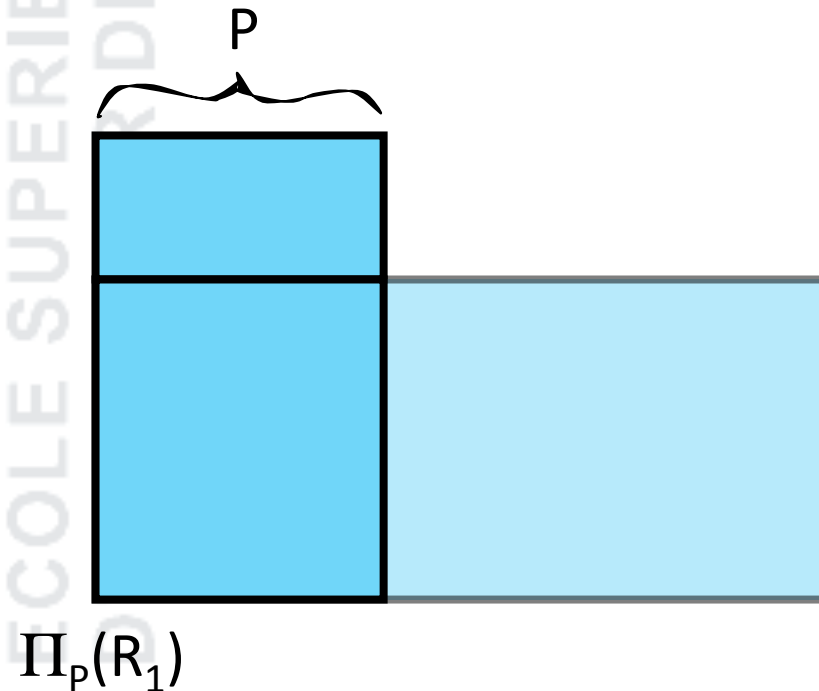
- $(R_1 / R_2) = \Pi_p(R_1) \setminus \Pi_p((\Pi_p(R_1) \times R_2) \setminus R_1)$



$R_1$

## Remarque – quotient (9)

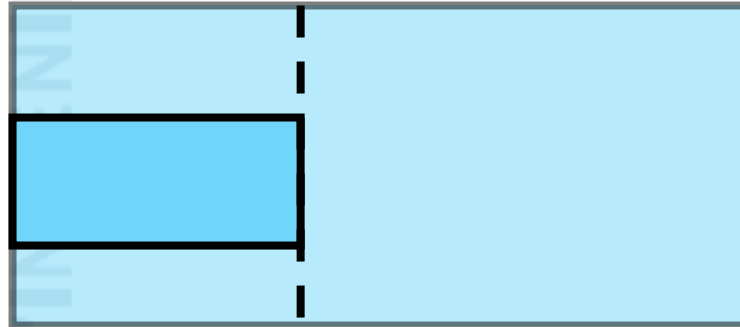
- $(R_1 / R_2) = \Pi_p(R_1) \setminus \Pi_p((\Pi_p(R_1) \times R_2) \setminus R_1)$



## Remarque – quotient (10)

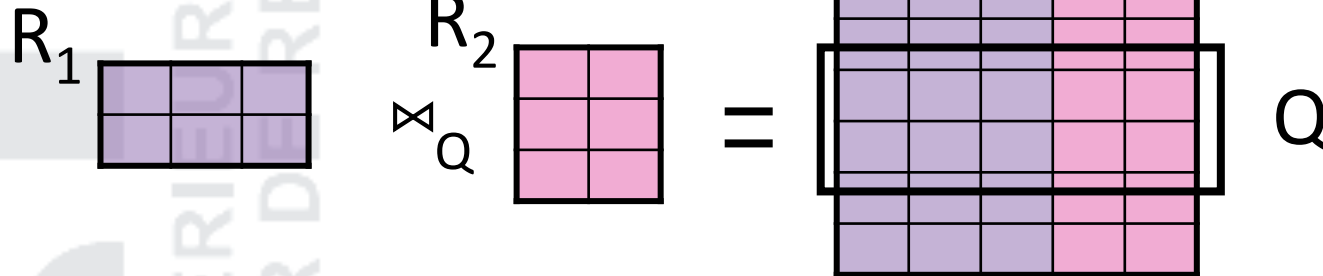
- $(R_1 / R_2) = \Pi_p(R_1) \setminus \Pi_p((\Pi_p(R_1) \times R_2) \setminus R_1)$

$(R_1/R_2)$



# $\Theta$ -Jointure (1)

$$R_1 \bowtie_Q R_2$$



- $\text{join}_Q(R_1, R_2)$  ou  $R_1 \bowtie_Q R_2$
- **SELECT ... FROM  $R_1, R_2$  WHERE  $Q$**

## $\Theta$ -Jointure (2)

X	Y	Z
1	a	3
2	z	7



$X+Z=V$

U	V
11	4
22	22
33	9

$\equiv$

X	Y	Z	U	V
1	a	3	11	4
2	z	7	33	9

# Variantes $\Theta$ -Jointure (1)

- $\text{join}_Q(R_1, R_2) = \sigma_Q(R_1 \times R_2)$
- Équijointure :  $Q = \text{EQUAL}$
- Autojointure :  $R_1 = R_2$ 
  - application ?

# Quelques requêtes (10)

```
SELECT DISTINCT adh1.Nom  
FROM Adherent adh1, Adherent adh2  
WHERE adh1.Nom = adh2.Nom  
AND adh1.NumAdherent  
    <> adh2.NumAdherent
```

- Jointure **implicite**

# Quelques requêtes (11)

SELECT Nom, Prenom, Intitule  
FROM **Adherent, AdherentActivite, Activite**  
WHERE **Adherent.NumAdherent =**  
          **AdherentActivite.NumAdherent**  
AND **AdherentActivite.NumActivite =**  
      **Activite.NumActivite**  
AND Expertise > 5

- Jointure implicite entre 3 relations



## Quelques requêtes (12)

SELECT Nom, Prenom, Intitule  
FROM Adherent **adh**, AdherentActivite **adac**,  
Activite **act**  
WHERE **adh**.NumAdherent = **adac**.NumAdherent  
AND **adac**.NumActivite = **act**.NumActivite  
AND **act**.Expertise > 5

- Même chose avec des **alias** pour alléger la notation.

## Variantes $\Theta$ -Jointure (2)

- Jointure naturelle :  $\bowtie$  (sans Q)

= équijointure + projection

$$= \Pi_{\text{UnionSchéma}}(\sigma_{=}(R_1 \times R_2))$$

**SELECT ...**

**FROM  $R_1$  INNER JOIN  $R_2$  ON  $R_1.attr = R_2.attr$**

## Quelques requêtes (13)

SELECT Nom, Prenom, Intitule

FROM ( Adherent adh

**INNER JOIN** AdherentActivite adac

**ON adh.NumAdherent = adac.NumAdherent )**

**INNER JOIN** Activite act

**ON adac.NumActivite = act.NumActivite**

WHERE act.Expertise > 5

- Même chose avec des jointures explicites

# Quelques requêtes (14)

SELECT **Nom, Prenom**

FROM Adherent adh

INNER JOIN AdherentActivite adac

ON adh.NumAdherent =  
adac.NumAdherent

WHERE act.Expertise > 5

- Ne faire la jointure qu'avec les tables nécessaires

# Quelques requêtes (15)

```
SELECT Nom, Prenom  
FROM ( Adherent adh  
      INNER JOIN AdherentActivite adac ...)  
      INNER JOIN Activite act ...  
WHERE act.Intitule <> 'Violon'
```

- Tous les noms-prénoms d'adhérents qui ont une activité autre que le violon

**ils peuvent quand même  
avoir l'activité violon !**

# Quelques requêtes (16)

**SELECT Nom, Prenom FROM Adherent  
EXCEPT**

SELECT Nom, Prenom  
FROM ( Adherent adh  
INNER JOIN AdherentActivite adac ...)  
INNER JOIN Activite act ...

WHERE **act.Intitule = 'Violon'**

- **Tous les noms-prénoms d'adhérents qui n'ont pas l'activité violon !**

# Remarques

- Différence entre  
« Aimez-vous Brahms ? »  
et  
« N'aimez-vous que Brahms ? »
- Les choses se compliquent si l'activité NULL est acceptée...  
...différence entre  
« Ne pas avoir d'activité enregistrée »  
et  
« Avoir une activité enregistrée mais non renseignée »

# Jointure externe (1)

- Pour conserver les informations d'une table
- Insérer des **(ligne, NULL, ...)**



# Jointure externe (2)

X	Y	Z
1	a	3
2	z	7
3	g	5



$X+Z=V$

U	V
11	4
22	22
33	9

$=$

X	Y	Z	U	V
1	a	3	11	4
2	z	7	33	9
3	g	5	NULL	NULL
NULL	NULL	NULL	22	22

# Doublons et ordre

- **SELECT DISTINCT projection...**  
**FROM produit...**  
**WHERE restriction...**
- **SELECT projection...**  
**FROM ...**  
**WHERE ...**  
**ORDER BY projection... ASC ou DESC**

# Quelques requêtes (17)

$\sigma_{\text{DataAdhesion IS NULL}}$  (Adherent)

```
SELECT *  
FROM Adherent  
WHERE DateAdhesion IS NULL  
ORDER BY Nom, Prenom
```

# Quelques requêtes (18)

SELECT Nom, Prenom, Intitule  
FROM ( Adherent adh  
INNER JOIN AdherentActivite adac ...)  
INNER JOIN Activite act ...  
**ORDER BY Intitule, Nom, Prenom**

# Agrégation (1)

- Statistique
  - AVG, COUNT, MAX, MIN, SUM

```
SELECT COUNT(*), SUM(qte), MAX(qte),  
MIN(qte), AVG(qte)  
FROM commandes ;
```

## Agrégation (2)

- COUNT(\*) compte toutes les lignes
- COUNT(ALL attr) compte les lignes où l'attribut n'est pas NULL
- COUNT(DISTINCT attr) compte les valeurs différentes

# Agrégation (3)

**SELECT COUNT(\*), SUM(qte), MAX(qte),  
MIN(qte), AVG(qte)  
FROM commandes ;**

ref	qte
36568	2
24086	5
90636	2
13234	10
63028	1
56664	5



count	sum	max	min	avg
6	25	10	1	4,16

# Quelques requêtes (19)

card( $\sigma_{\text{DataAdhesion IS NULL}}(\text{Adherent})$ )

```
SELECT COUNT(*)  
FROM Adherent  
WHERE DateAdhesion IS NULL
```



# Quelques requêtes (20)

```
SELECT Nom, Prenom, AVG(Expertise)  
FROM Adherent adh  
      INNER JOIN AdherentActivite adac  
      ON adh.NumAdherent = adac.NumAdherent  
WHERE adh.NumAdherent = act.NumResponsable
```

- L'expertise moyenne des adhérents qui sont responsables d'activité

# Regroupement (1)

**SELECT num-fournisseur, SUM(qte)  
FROM commandes  
GROUP BY num-fournisseur ;**

ref	qte	numf
36568	2	13
24086	5	25
90636	2	25
13234	10	7
63028	1	13
56664	5	25



numf	sum
13	3
25	12
7	10

# Regroupement (2)

**SELECT num-fournisseur, COUNT(\*)**  
**FROM commandes**  
**GROUP BY num-fournisseur ;**

ref	qte	numf
36568	2	13
24086	5	25
90636	2	25
13234	10	7
63028	1	13
56664	5	25



numf	count
13	2
25	3
7	1

# Regroupement (3)

**SELECT num-fournisseur, COUNT(\*)**  
**FROM commandes**  
**GROUP BY num-fournisseur**  
**HAVING COUNT(\*) < 3;**

ref	qte	numf
36568	2	13
24086	5	25
90636	2	25
13234	10	7
63028	1	13
56664	5	25



numf	count
13	2
7	1

# Quelques requêtes (21)

```
SELECT adh1.Nom  
FROM Adherent adh1, Adherent adh2  
WHERE adh1.Nom = adh2.Nom  
GROUP BY adh1.Nom  
HAVING COUNT(adh2.Nom) > 1
```

# Quelques requêtes (22)

```
SELECT Nom, Prenom, COUNT(NumActivite)  
FROM Adherent adh  
      INNER JOIN AdherentActivite adac ...  
GROUP BY Nom, Prenom
```

# Quelques requêtes (23)

```
SELECT Nom, Prenom, COUNT(NumActivite)
FROM Adherent adh
      INNER JOIN AdherentActivite adac ...
GROUP BY Nom, Prenom
HAVING COUNT(NumActivite) > 5
```

## Quelques requêtes (24)

```
SELECT Nom, Prenom, AVG(Expertise)
FROM Adherent adh
      INNER JOIN AdherentActivite adac
      ON adh.NumAdherent = adac.NumAdherent
WHERE adh.NumAdherent = act.NumResponsable
GROUP BY Nom, Prenom
```

- L'expertise moyenne pour chaque adhérent-responsable dans les activités dont il est responsable



# Remarque

## agrégats et groupements (1)

```
SELECT colonnesSELECT,  
        colonnesAGRÉGATS  
FROM tablesFROM  
WHERE ...  
GROUP BY colonnesGROUP  
HAVING ... ;
```

# Remarque

## agrégats et groupements (2)

On doit avoir

$$\text{colonnes}_{\text{SELECT}} \subseteq \text{colonnes}_{\text{GROUP}} \subseteq \text{tables}_{\text{FROM}}$$

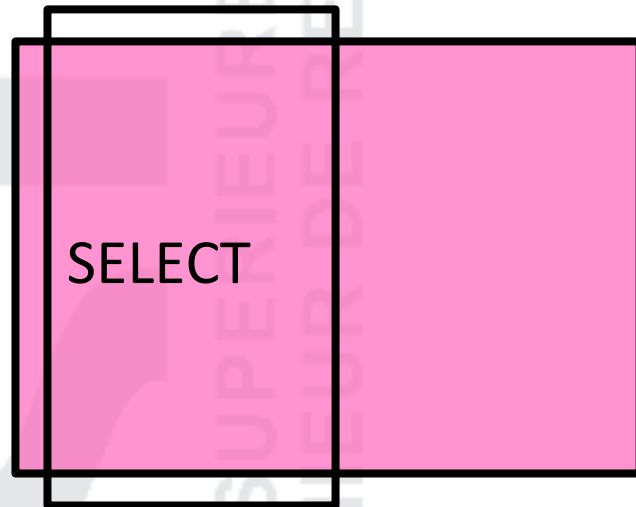
et souvent

$$\text{colonnes}_{\text{AGRÉGATS}} \subseteq \text{tables}_{\text{FROM}} \setminus \text{colonnes}_{\text{GROUP}}$$

GROUP BY est une sorte de projection qui n'oublie pas complètement les colonnes non-projetées

# Remarque

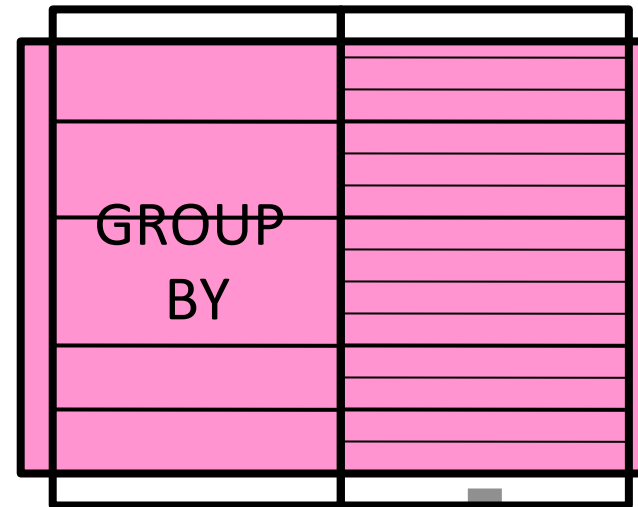
## agrégats et groupements (3)



projection



calcul relationnel



projection

agrégation  
par groupe



# Conclusion (1)

- Unité de calcul = relation ou table
- Opérations macroscopiques sur les relations ou tables
- Pas d'itération explicite

## Conclusion (2)

- SQL spécialisation de AR
  - $\Pi(\sigma(.x.))$
- SQL approximation de AR
  - ensemble  $\neq$  table
- SQL extension de AR
  - agrégats et groupements