

Persistence des données I

DON2

Denis Boigelot, Geneviève Cuvelier, Selim Rexhep, Yannick Voglaire



Haute École Bruxelles-Brabant
École Supérieure d'Informatique

Année académique 2020 / 2021

Plan du cours

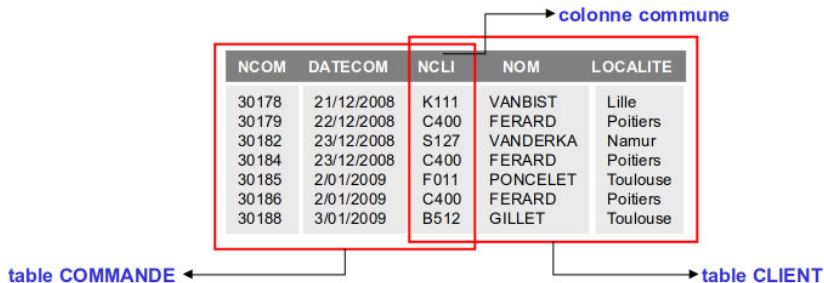
- 0 – Présentation
- 1 – Introduction
- 2 – Dépendance fonctionnelle
- 3 – Schéma conceptuel
- 4 – Projection et sélection
- 5 – Jointure
- 6 – Agrégat
- 7 – Sous-requête
- 8 – Fichiers

5 – Jointure

- 1 Introduction
- 2 Algèbre relationnelle
- 3 SQL - Produit cartésien
- 4 SQL - Jointure
- 5 SQL - Jointure - Sémantique
- 6 SQL - Jointure - Exercices
- 7 SQL - Jointure - Clé étrangère réflexive
- 8 SQL - Jointure - Clé étrangère nulle

Introduction

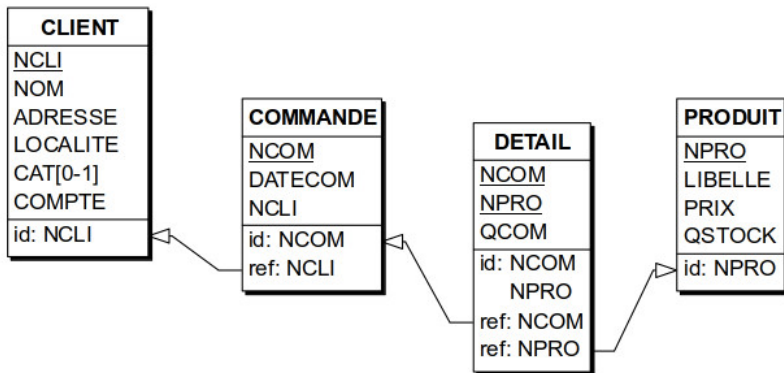
La **jointure** permet de produire une table constituée de données extraites de plusieurs tables :



```

SELECT ncom, datecom, client.ncli, nom, localite
FROM commande
JOIN client ON commande.ncli=client.ncli ;
  
```

Rappel - Identifiants et clés étrangères



Algèbre relationnelle - Introduction

A		
a1	a2	a3
10	aaaa	15
20	bbbb	15
30	cccc	22
40	dddd	33

B		
b1	b2	b3
33	xxxx	10
15	bbbb	40
41	tttt	40
49	dddd	75

Produit cartésien

Le *produit cartésien* de A et B est noté $A \times B$

Jointure

La *jointure*, notée

$$(A) \bowtie_{\langle \text{condition} \rangle} (B),$$

est l'ensemble des lignes du produit cartésien qui vérifient la condition.

Algèbre relationnelle - Produit cartésien

A		
a1	a2	a3
10	aaaa	15
20	bbbb	15
30	cccc	22
40	dddd	33

B		
b1	b2	b3
33	xxxx	10
15	bbbb	40
41	tttt	40
49	dddd	75

$$A \times B =$$

a1	a2	a3	b1	b2	b3
10	aaaa	15	33	xxxx	10
10	aaaa	15	15	bbbb	40
10	aaaa	15	41	tttt	40
10	aaaa	15	49	dddd	75
20	bbbb	15	33	xxxx	10
20	bbbb	15	15	bbbb	40
20	bbbb	15	41	tttt	40
20	bbbb	15	49	dddd	75
30	cccc	22	33	xxxx	10
30	cccc	22	15	bbbb	40
30	cccc	22	41	tttt	40
30	cccc	22	49	dddd	75
40	dddd	33	33	xxxx	10
40	dddd	33	15	bbbb	40
40	dddd	33	41	tttt	40
40	dddd	33	49	dddd	75

Algèbre relationnelle - Jointure

A		
a1	a2	a3
10	aaaa	15
20	bbbb	15
30	cccc	22
40	dddd	33

B		
b1	b2	b3
33	xxxx	10
15	bbbb	40
41	tttt	40
49	dddd	75

$$(A) \bowtie_{a1=b3} (B) =$$

a1	a2	a3	b1	b2	b3
10	aaaa	15	33	xxxx	10
10	aaaa	15	15	bbbb	40
10	aaaa	15	41	tttt	40
10	aaaa	15	49	dddd	75
20	bbbb	15	33	xxxx	10
20	bbbb	15	15	bbbb	40
20	bbbb	15	41	tttt	40
20	bbbb	15	49	dddd	75
30	cccc	22	33	xxxx	10
30	cccc	22	15	bbbb	40
30	cccc	22	41	tttt	40
30	cccc	22	49	dddd	75
40	dddd	33	33	xxxx	10
40	dddd	33	15	bbbb	40
40	dddd	33	41	tttt	40
40	dddd	33	49	dddd	75

Algèbre relationnelle - Jointure

A		
a1	a2	a3
10	aaaa	15
20	bbbb	15
30	cccc	22
40	dddd	33

B		
b1	b2	b3
33	xxxx	10
15	bbbb	40
41	tttt	40
49	dddd	75

$$(A) \bowtie_{a1=b3} (B) =$$

a1	a2	a3	b1	b2	b3
10	aaaa	15	33	xxxx	10
40	dddd	33	15	bbbb	40
40	dddd	33	41	tttt	40

Algèbre relationnelle - Jointure

A		
a1	a2	a3
10	aaaa	15
20	bbbb	15
30	cccc	22
40	dddd	33

B		
b1	b2	b3
33	xxxx	10
15	bbbb	40
41	tttt	40
49	dddd	75

$$(A) \bowtie_{a1 \neq b3} (B) =$$

a1	a2	a3	b1	b2	b3
10	aaaa	15	33	xxxx	10
10	aaaa	15	15	bbbb	40
10	aaaa	15	41	tttt	40
10	aaaa	15	49	dddd	75
20	bbbb	15	33	xxxx	10
20	bbbb	15	15	bbbb	40
20	bbbb	15	41	tttt	40
20	bbbb	15	49	dddd	75
30	cccc	22	33	xxxx	10
30	cccc	22	15	bbbb	40
30	cccc	22	41	tttt	40
30	cccc	22	49	dddd	75
40	dddd	33	33	xxxx	10
40	dddd	33	15	bbbb	40
40	dddd	33	41	tttt	40
40	dddd	33	49	dddd	75

Algèbre relationnelle - Jointure

A		
a1	a2	a3
10	aaaa	15
20	bbbb	15
30	cccc	22
40	dddd	33

B		
b1	b2	b3
33	xxxx	10
15	bbbb	40
41	tttt	40
49	dddd	75

$$(A) \bowtie_{a1 \neq b3} (B) =$$

a1	a2	a3	b1	b2	b3
10	aaaa	15	15	bbbb	40
10	aaaa	15	41	tttt	40
10	aaaa	15	49	dddd	75
20	bbbb	15	33	xxxx	10
20	bbbb	15	15	bbbb	40
20	bbbb	15	41	tttt	40
20	bbbb	15	49	dddd	75
30	cccc	22	33	xxxx	10
30	cccc	22	15	bbbb	40
30	cccc	22	41	tttt	40
30	cccc	22	49	dddd	75
40	dddd	33	33	xxxx	10
40	dddd	33	49	dddd	75

Algèbre relationnelle - Exercices

A		
a1	a2	a3
10	aaaa	15
20	bbbb	15
30	cccc	22
40	dddd	33

B		
b1	b2	b3
33	xxxx	10
15	bbbb	40
41	tttt	40
49	dddd	75

Donner la liste des tuples de A pour lesquels

- 1 la valeur de a1 apparaît dans b3
- 2 la valeur de a1 n'apparaît pas dans b3

Algèbre relationnelle - Exercices

A		
a1	a2	a3
10	aaaa	15
20	bbbb	15
30	cccc	22
40	dddd	33

B		
b1	b2	b3
33	xxxx	10
15	bbbb	40
41	tttt	40
49	dddd	75

Calculer

$$1 \quad (A) \bowtie_{a1=b3} (\sigma_{\{b3=75\}}(B))$$

$$2 \quad \pi_{\{a1,b1,b2,b3\}}(\sigma_{\{b1=15\}}((A) \bowtie_{a1=b3} (B)))$$

$$3 \quad \pi_{\{a1,a2,b1,b2\}}((A) \bowtie_{a2=b2} (B))$$

$$4 \quad (A) \bowtie_{A.a1=C.b3} ((A) \bowtie_{a3=b1} (B))$$

SQL - Produit cartésien

```
SELECT ncom, client.ncli, datecom, nom, adresse  
FROM commande, client ;
```

Pas de condition de jointure

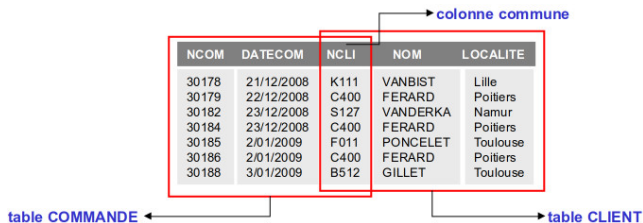
Produit cartésien : chaque ligne de COMMANDE est couplée avec chaque ligne de CLIENT

Requête valide mais d'utilité réduite dans ce cas.

Utile quand on veut combiner toutes les lignes d'une table avec toutes les lignes d'une autre table.

SQL - Jointure

La **jointure** permet de produire une table constituée de données extraites de plusieurs tables :



```
SELECT ncom, datecom, client.ncli, nom, localite
FROM commande
JOIN client ON commande.ncli = client.ncli ;
```

Il faut préfixer les noms de colonne s'il y a ambiguïté

SQL - Jointures

SELECT liste d'expressions
FROM table1
JOIN table2 **ON** condition de jointure ;

Condition de jointure :

Clé étrangère = identifiant primaire de la table référencée. (**FK=ID**)

CLIENT

NCLI	NOM	ADRESSE	LOCALITE	(C1)	COMPTE
B112	COFFIN	72, r. de la Gare	Namur	B2	-3000
B112	HANSENNE	23, r. Dumont	Poitiers	C1	1250
B332	MONTI	112, r. Neuve	Gendève	B2	0
B512	GILLET	14, r. de l'Est	Toulouse	B1	-8700
C003	AVIRON	8, r. de la Cure	Toulouse	B1	-1700
C113	MERCIER	25, r. Lomaille	Namur	C1	2300
C400	FERRARD	66, r. du Temps	Poitiers	B2	250
C083	MERCIER	201, bd du Nord	Toulouse		-2250
F010	TOUSSAINT	5, r. Godefrid	Poitiers	C1	0
F011	PONCELET	17, Clos des Etables	Toulouse	B2	0
F400	JACOB	76, ch. du Moulin	Bruxelles	C2	0
K111	VANBIST	180, r. Flémant	Lille	B1	720
K729	NEUMAN	40, r. Bransart	Toulouse		0
L422	FRANCK	80, r. de Wagon	Namur	C1	0
S107	VANDERVA	3, av. des Roses	Namur	C1	-4380
S712	GUILAUME	14a, ch. des Roses	Paris	B1	0

COMMANDE

NUM	NCLI	DATECOM
30176	C111	21/12/2008
30177	C400	22/12/2008
30182	S117	23/12/2008
30184	C400	23/12/2008
30185	F011	26/12/2008
30186	C400	26/12/2008
30189	B512	30/12/2008

identifiant

clé étrangère

dans le schéma
 dans les données

SQL - Jointures

```
SELECT client.ncli, nom, datecom, npro  
FROM client  
      JOIN commande ON client.ncli = commande.ncli  
      JOIN detail ON commande.ncom = detail.ncom ;
```

Jointure sur 3 tables

```
SELECT ncom, client.ncli, nom, datecom, adresse  
FROM commande  
      JOIN client ON client.ncli = commande.ncli  
WHERE cat = ' C1' AND datecom < ' 23 - 12 - 2009' ;
```

Condition de jointure
+
condition de sélection

SQL - Jointures - Lignes célibataires

```
SELECT localite, client.ncli, nom, ncom, datecom  
FROM client  
JOIN commande ON commande.ncli = client.ncli ;
```

Nous obtenons les clients ayant passé au moins une commande.

Cette requête ignore les lignes de CLIENT qui n'ont pas de lignes correspondantes dans COMMANDE

Ces lignes de CLIENT n'ayant pas de correspondance dans COMMANDE sont dites **célibataires**.

SQL - Jointures - Valeurs dérivées

```
SELECT ncom, d.npro, qcom*prix  
FROM detail d  
JOIN produit p ON d.npro = p.npro ;
```

d et p sont des alias de table

SQL - Jointures - Sémantique

- ◇ Une ligne de CLIENT représente un client.
- ◇ Une ligne de COMMANDE représente une commande.

Que représente chaque ligne de la jointure
 $(client\ c) \bowtie_{m.ncli=c.ncli} (commande\ m)$

```
SELECT c.ncli, nom, localite
FROM client c
JOIN commande m ON m.ncli = c.ncli ;
```

- ◇ un client ?
- ◇ un client qui a passé une commande ?
- ◇ une commande ?

SQL - Jointures - Sémantique

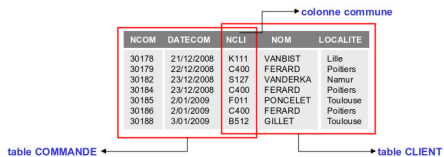
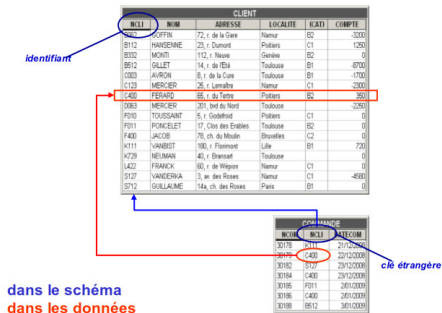
$(client\ c) \bowtie_{m.ncli=c.ncli} (commande\ m)$

```
SELECT c.ncli, nom, localite
FROM client c
JOIN commande m ON m.ncli = c.ncli ;
```

Autre formulation : il y a autant de lignes dans le résultat qu'il y a

- ◇ de clients ?
- ◇ de clients qui ont passé des commandes ?
- ◇ de commandes ?

SQL - Jointures - Sémantique



Le résultat d'une jointure FK/ID représente les entités de la table de la clé étrangère

SQL - Jointures - Sémantique

```
SELECT ncli, c.ncom, datecom, npro  
  FROM detail d  
    JOIN commande c ON d.ncom = c.ncom ;
```



```
SELECT localite, d.ncom, d.npro, libelle  
  FROM client c  
    JOIN commande com ON c.ncli = com.ncli  
    JOIN detail d ON com.ncom = d.ncom  
    JOIN produit p ON p.npro = d.npro ;
```

SQL - Jointures - Exercices

CLIENT					
NCLI	NOM	ADRESSE	LOCALITE	(CAT)	COMPTE
B032	GOFFIN	72, r. de la Gare	Namur	B2	-3200
B112	HANSENNE	23, r. Dumont	Poitiers	C1	1250
B332	MONTI	112, r. Neuve	Genève	B2	0
B512	GILLET	14, r. de l'Eté	Toulouse	B1	-8700
C003	AVIRON	8, r. de la Cure	Toulouse	B1	-1700
C123	MERCIER	25, r. Lemaitre	Namur	C1	-2300
C400	FERARD	65, r. du Tertre	Poitiers	B2	350
D063	MERCIER	201, bd Nord	Toulouse		-2250
F010	TOUSSAINT	5, r. Godefroid	Poitiers	C1	0
F011	PONCELET	17, Clos des Erables	Toulouse	B2	0
F400	JACOB	78, ch. du Moulin	Bruxelles	C2	0
K111	VANBIST	180, r. Florimont	Lille	B1	720
K729	NEUMAN	40, r. Bransart	Toulouse		0
L422	FRANCK	60, r. de Wépion	Namur	C1	0
S127	VANDERKA	3, av. des Roses	Namur	C1	-4580
S712	GUILAUME	14a, ch. des Roses	Paris	B1	0

PRODUIT			
NPRO	LIBELLE	PRIX	QSTOCK
CS262	CHEV. SAPIN 200x6x2	75	45
CS364	CHEV. SAPIN 200x6x4	120	2690
CS464	CHEV. SAPIN 400x6x4	220	450
PA45	POINTE ACIER 45 (2K)	105	580
PA60	POINTE ACIER 60 (1K)	95	134
PH222	PL. HETRE 200x20x2	230	782
PS222	PL. SAPIN 200x20x2	195	1220

COMMANDE		
NCOM	NCLI	DATECOM
30178	K111	21/12/2008
30179	C400	22/12/2008
30182	S127	23/12/2008
30184	C400	23/12/2008
30195	F011	2/01/2009
30186	C400	2/01/2009
30188	B512	3/01/2009

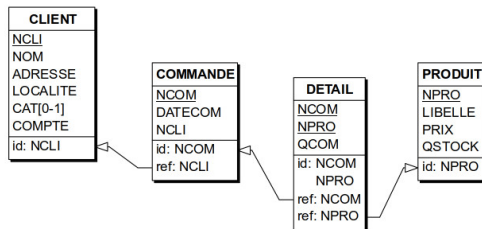
DETAIL		
NCOM	NPRO	QCOM
30178	CS464	25
30179	CS262	60
30179	PA60	20
30182	PA60	30
30184	CS464	120
30184	PA45	20
30185	CS464	260
30185	PA60	15
30185	PS222	600
30186	PA45	3
30188	CS464	180
30188	PA45	22
30188	PA60	70
30188	PH222	92

Donnez la sémantique de la requête suivante :

```

SELECT detail.ncom, detail.npro, qcom, qstock
FROM detail
JOIN produit ON detail.npro = produit.npro
WHERE qcom > qstock ;
  
```


SQL - Jointures - Exercices

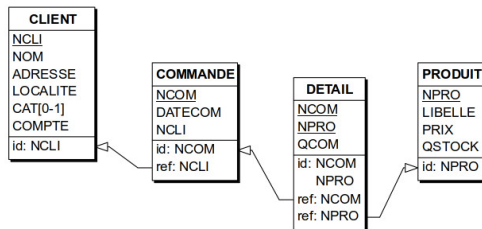


Donnez la sémantique de la requête suivante :

```

SELECT DISTINCT p.npro, libelle, prix
FROM produit p
JOIN detail d ON p.npro = d.npro
JOIN commande com ON com.ncom = d.ncom
JOIN client c ON c.ncli = com.ncli
WHERE localite IN ('Bruxelles', 'Liège', 'Namur') ;
  
```

SQL - Jointures - Exercices



Donnez la sémantique de la requête suivante :

```

SELECT DISTINCT client.ncli
FROM commande
JOIN client ON client.ncli = commande.ncli
WHERE compte < 0 ;
  
```

SQL - Jointures - Exercices

CLIENT					
NCLI	NOM	ADRESSE	LOCALITE	RCAT	COMPTE
B062	GOFFIN	72, r. de la Gare	Namur	B2	-3200
B112	HANSENNE	23, r. Dumont	Poitiers	C1	1250
B332	MONTI	112, r. Neuve	Genève	B2	0
B612	GILLET	14, r. de l'Est	Toulouse	B1	4700
C003	AVIRON	8, r. de la Cure	Toulouse	B1	-1700
C123	MERCIER	25, r. Lamaille	Namur	C1	-2300
C400	FERRARO	65, r. du Tetre	Poitiers	B2	350
D063	MERCIER	201, bd du Nord	Toulouse		-2250
F010	TOUSSAINT	5, r. Godefroid	Poitiers	C1	0
F011	PONCELET	17, Clos des Etables	Toulouse	B2	0
F400	JACOB	78, ch. du Moulin	Bruxelles	C2	0
K111	VANBIST	180, r. Florimont	Lille	B1	720
K29	NEUMAN	40, r. Bransart	Toulouse		0
L422	FRANCK	60, r. de Wispion	Namur	C1	0
S127	VANDERVA	3, av. des Roses	Namur	C1	-4580
S712	GUILLAUME	14a, ch. des Roses	Paris	B1	0

PRODUIT				
NPRO	LIBELLE	PRIX	OSTOCK	
CS262	CHEV SAPIN 200x6x2	75	45	
CS264	CHEV SAPIN 200x6x4	120	2690	
CS464	CHEV SAPIN 400x6x4	220	450	
PA45	POINTE ACIER 45 (2x)	105	580	
PA60	POINTE ACIER 60 (1x)	95	134	
PH222	PL. HETRE 200x20x2	230	760	
PS222	PL. SAPIN 200x20x2	185	1220	

COMMANDE			
NCOM	NCLI	DATECOM	
30178	K111	21/12/2000	
30179	C400	22/12/2000	
30182	S127	23/12/2000	
30184	C400	23/12/2000	
30195	F011	2/01/2009	
30196	C400	2/01/2009	
30198	B512	3/01/2009	

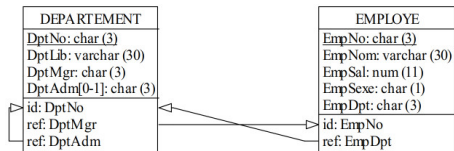
DETAIL			
NCOM	NPRO	QCOM	
30178	CS464	25	
30179	CS262	60	
30179	PA60	20	
30182	PA60	30	
30184	CS464	120	
30184	PA45	20	
30195	CS464	260	
30195	PA60	15	
30195	PS222	600	
30196	PA45	3	
30198	CS464	180	
30198	PA45	22	
30198	PA60	70	
30198	PH222	92	

Si la table CLIENT comporte 16 lignes et la table COMMANDE 7, combien de lignes seront retournées par la requête suivante ?

```
SELECT *
FROM commande
JOIN client ON client.ncli != commande.ncli ;
```

Quel est sa sémantique ?

SQL - Jointures - Clé étrangère réflexive



DptAdm est une clé étrangère vers DptNo.

C'est une **clé étrangère réflexive** (self-reference)

DptAdm est facultatif. Il peut être **NULL**.

Dessinez l'arborescence des départements suivants :

DptNo	DptLib	DptMgr	DptAdm
A00	DEVELOPPEMENT	320	D21
B01	PRODUCTION	020	A00
C01	MAINTENANCE	030	A00
D11	SUPPORT	060	E11
D21	DIRECTION	070	null
E01	MARKETING	050	E11
E11	VENTES	340	D21
E21	FORMATION	100	E11

SQL - Jointures - Clé étrangère réflexive

DptNo	DptLib	DptMgr	DptAdm
A00	DEVELOPPEMENT	320	D21
B01	PRODUCTION	020	A00
C01	MAINTENANCE	030	A00
D11	SUPPORT	060	E11
D21	DIRECTION	070	null
E01	MARKETING	050	E11
E11	VENTES	340	D21
E21	FORMATION	100	E11

Que retournera la requête suivante sur la table ci-dessus ?

```
SELECT sup.dptNo, sup.dptLib, dpt.dptLib
FROM departement sup
JOIN departement dpt ON dpt.dptAdm = sup.dptNo ;
```

SQL - Jointures - Clé étrangère nulle

DptNo	DptLib	DptMgr	DptAdm
A00	DEVELOPPEMENT	320	D21
B01	PRODUCTION	020	A00
C01	MAINTENANCE	030	A00
D11	SUPPORT	060	E11
D21	DIRECTION	070	null
E01	MARKETING	050	E11
E11	VENTES	340	D21
E21	FORMATION	100	E11

La requête suivante sur la table ci-dessus retournera combien de lignes ?

```
SELECT sup.dptNo, sup.dptLib, dpt.dptLib
FROM departement sup
JOIN departement dpt ON dpt.dptAdm != sup.dptNo ;
```

Quel est sa sémantique ?