

Introduction à la modélisation

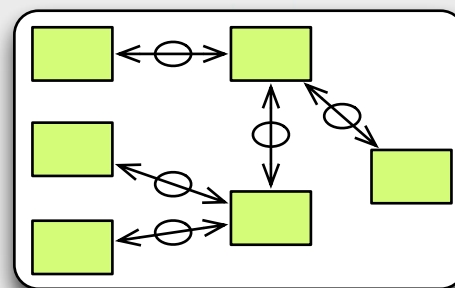
Modèle logique des données

E. Desmontils (Dpt Informatique)

emmanuel.desmontils@univ-nantes.fr

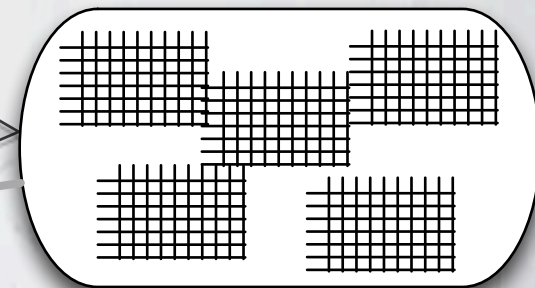
Université de Nantes

version 3.0



Modèle conceptuel

Élaboration du schéma
logique



Modèle relationnel





Modèle simple

Vous avez dit "simple" ?

- Dans certains cas "simples" : passage du MCD au MLD sous forme d'un simple tableau (type tableur).
 - Calc, Numbers, Excel...
 - solution classique adoptée par "M. tout le monde"
 - solution souvent satisfaisante
- Simple ?
 - Peu de données
 - Une seule entité ou des entités avec des associations "1..1" partout.
 - Des entités de type "liste" :
 - participent à des associations binaires "1..n"-"O/1..1"
 - ne possèdent qu'une seule propriété

Exemple



exUt

Arial 10 G / S

C14 $f(x)$ Σ =

	A	B	C	D	E
1	Total mois	1 135,23 €			
2					
3	No	Date	Somme	Type	Intitulé
4	1	04/09/08	1 367,56 €	Salaire	Mois de septembre
5	2	04/09/08	-34,56 €	Culturel	Livres FNAC
6	3	06/09/08	-5,50 €	Nourriture	Boulangerie
7	4	06/09/08	-176,87 €	Nourriture	Supermarché
8	5	08/09/08	-15,40 €	Culturel	CD Supermarché
9					
10					



Modèle relationnel



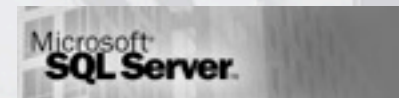


Modèle Relationnel



InterBase®
Base de données embarquée
multi-plate-forme

- Données en tables (ou relations) selon des règles précises
- Requêtes exprimées à l'aide d'opérateurs sur ces tables
- Avantages
 - Pour un ensemble d'informations à trouver, on décrit juste les opérateurs sur les relations qui permettent de les atteindre (expressions relationnelles)
 - Approche beaucoup plus détachée de l'implémentation physique
 - Simple, facile à comprendre, logique et fidèle à un cadre mathématique (algèbre relationnelle, théorie des ensembles)
 - Très utilisé (75% des bases de données)



Domaine et t-uple

- **Domaine** : ensemble fini ou infini de valeurs distinctes que peut prendre une donnée élémentaire (information)
 - Définition en intention ou en extension
 - Entiers, Booléens, {standard, admin}...
 - Exemple : $D_1=\{a,b,c\}$, $D_2=\{1,2\}$

Domaine et t-uple

- **n-uplet, t-uple, occurrence** :
élément $\langle v_1, v_2 \dots v_n \rangle$ du produit cartésien d'un ensemble de domaines $D_1, D_2 \dots D_n$ noté $D_1 \times D_2 \times \dots D_n$ tel que $\forall i, v_i \in D_i$.
 - Exemple : si $D_1 = \{a, b, c\}$ et $D_2 = \{1, 2\}$ alors
 $D_1 \times D_2 = \{\langle a, 1 \rangle, \langle a, 2 \rangle, \langle b, 1 \rangle, \langle b, 2 \rangle, \langle c, 1 \rangle, \langle c, 2 \rangle\}$
 - Remarque : le produit cartésien peut être généralisé sur des ensembles quelconques et, en particulier, des ensembles de t-uples.
- Chaque v_i est appelé **attribut**, c'est une information.

Exemples

- Des domaines
 - Nom = {Dupont, Martin, Robert, Denou, Bellamy...}
 - Prénom = {Franck, Isabelle, Adrien, Michelle, Jean, Gustave...}
 - Note = [0..20]
- Des t-uples :
 - Nom×Prénom = {<Dupont,Franck>, <Dupont,Isabelle>..., <Martin,Franck>, <Martin,Isabelle>...}
 - Nom×Note = {<Dupont,0>, <Dupont,1>...}

Tables - Relations

- Une **table** ou **relation** : sous-ensemble du produit cartésien d'un ensemble de domaines. Elle est identifiée par un nom.
- Représentation : table dont les lignes sont les **occurrences** ou **t-uples**. Les noms de colonnes sont les **attributs**. Chaque attribut est unique et identifié par son nom.
 - L'ordre des lignes et des colonnes est sans importance.
 - Tous les t-uples doivent être distincts : **notion de clé**
 - Chaque attribut doit être élémentaire et non calculable.

Tables - Relations

- Le **degré** d'une relation R : nombre d'attributs de R .
 - Notée : $\delta(R)$ = Nombre de colonnes de la table
- La **cardinalité** d'une relation R : nombre de t-uples.
 - Notée : $\text{Card}(R)$ ou $|R|$
 - Nombre de lignes de la table

Exemple

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

$$\delta(\text{Etudiants}) = 5$$

$$\text{Card}(\text{Etudiants}) = |\text{Etudiants}| = 6$$

Exemple

Etudiants	nom	mdp	noetu	login	prénom
	Dupont	ebnhertbez	46283B	dupont-i	Isabelle
	Dupont	khjgktyd	99322C	dupont-is	Isabelle
	Martin	ryrjerz	86719E	martin-a	Adrien
	Dupont	fhgtrzv	28936E	dupont-f	Franck
	Denou	zaazraz	99321C	denou-m	Michelle
	Robert	mpoutyt	99628C	robert-a	Adrien

$$\delta(\text{Etudiants}) = 5$$

$$\text{Card}(\text{Etudiants}) = |\text{Etudiants}| = 6$$

Exemple

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgkyd

$$\delta(\text{Etudiants}) = 5 ; \text{Card}(\text{Etudiant}) = 6$$

Matières	codemat	titre	responsable	diplôme
	MIAS2I5	I5	E238	Deug MIAS
	MIAS2I6	I6	E426	Deug MIAS
	IUP2MA	Automates	E238	Licence IUP-MIAGE
	LIL6	Systèmes	E236	Licence Informatique
	IUP2IS	Systèmes	E526	Licence IUP-MIAGE
	MIAS2I3	Math-Info	E426	Deug MIAS
	LIL5	Algo	E426	Licence Informatique

$$\delta(\text{Matières}) = 4 ; \text{Card}(\text{Matières}) = 7$$

Notes	noe	codemat	noteex	notecc
	99628C	MIAS2I5	12	15,5
	46283B	MIAS2I6	8	11
	46283B	MIAS2I5	9,5	2
	86719E	IUP2MA	12	5,5
	99321C	LIL6	18	16,5
	28936E	MIAS2I5	13,5	13,5
	86719E	IUP2IS	8,5	10
	99628C	MIAS2I6	3	6
	99321C	LIL5	15	14,5
	99322C	MIAS2I5	12	15,5
	28936E	MIAS2I6	12	null

$$\delta(\text{Notes}) = 4 ; \text{Card}(\text{Notes}) = 11$$

Clé d'une relation (v1)

- Relation = ensemble au sens mathématique
 - nécessité de distinguer, référencer, désigner
- L'ensemble d'attributs permettant d'identifier de manière unique un t-uple
 - Définition « intuitive »
 - Permet de différencier chacun des t-uples d'une table
 - De préférence, l'ensemble est le plus petit possible
- Attention : il peut y avoir plusieurs clés possibles pour une relation !

A rapprocher de l'identifiant d'une entité !

Exemple

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgirzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgkyd

Autre clé possible...

Identifier, différencier

Matières	codemat	titre	responsable	diplôme
	MIAS2I5	I5	E238	Deug MIAS
	MIAS2I6	I6	E426	Deug MIAS
	IUP2MA	Automates	E238	Licence IUP-MIAGE
	LIL6	Systèmes	E236	Licence Informatique
	IUP2IS	Systèmes	E526	Licence IUP-MIAGE
	MIAS2I3	Math-Info	E426	Deug MIAS
	LIL5	Algo	E426	Licence Informatique

Notes	noe	codemat	noteex	notecc
	99628C	MIAS2I5	12	15,5
	46283B	MIAS2I6	8	11
	46283B	MIAS2I5	9,5	2
	86719E	IUP2MA	12	5,5
	99321C	LIL6	18	16,5
	28936E	MIAS2I5	13,5	13,5
	86719E	IUP2IS	8,5	10
	99628C	MIAS2I6	3	6
	99321C	LIL5	15	14,5
	99322C	MIAS2I5	12	15,5
	28936E	MIAS2I6	12	null

Référencer

Quelques définitions

- Soient
 - A_R , l'ensemble des attributs de la relation R et O_R l'ensemble des occurrences de R
 - $\mathcal{P}(A_R)$, l'ensemble des parties de A_R .
- Soit $o \in O_R$, $v \in \mathcal{P}(A_R)$, $v \neq \emptyset$,
 $\Pi_v(o)$: projection de o sur les attributs v

Étudiants (noetu, nom, prénom, mdp, login)
 $\Pi_{\text{nom,prénom}}(\langle '92344C', 'Arreau', 'Jean', 'dfjevff', 'arreau-j' \rangle) = \langle 'Arreau', 'Jean' \rangle$

Quelques définitions

- **Clé candidate** : ens. d'attributs permettant de désigner, sans ambiguïté, une occurrence dans l'ensemble
 - $\text{Candidate}(A_R) = \{c : c \in \mathcal{P}(A_R)$

$$\wedge \forall (o, o') \in O_R \times O_R, \Pi_c(o) = \Pi_c(o') \Rightarrow o = o'\}$$

Étudiants (noetu, nom, prénom, mdp, login)
Candidate(Étudiant) = {<noetu>, <noetu, nom>, <noetu, nom, prénom>, <login, noetu>, <login, nom, prénom>...}

Clé d'une relation (v2)

- **Clé minimale** : clé candidate comporte un minimum d'attributs
 - $\text{Minimale}(A_R) = \{c : c \in \text{Candidate}(A_R) \wedge \nexists c' \in \text{Candidate}(A_R), |c'| < |c|\}$

Étudiants (noetu, nom, prénom, mdp, login)
Minimale(Étudiant) = {<noetu>, <login>}

- **Clé d'une relation**
 = clé minimale sélectionnée (mise en avant)

Remarque : un attribut appartenant à une clé est parfois appelé «attribut clé»

Étudiants (noetu, nom, prénom, mdp, login)
Clé de Étudiants : noetu !



Étudiants (noetu, nom, prénom, mdp, login)

Quelques définitions

- Clé étrangère** : ens. d'attributs servant de lien entre deux relations, clé minimale d'une autre relation.

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

Clés étrangères



Notes	noe	codemat	noteex	notecc
	99628C	MIAS2I5	12	15,5
	46283B	MIAS2I6	8	11
	46283B	MIAS2I5	9,5	2
	86719E	IUP2MA	12	5,5
	99321C	LIL6	18	16,5
	28936E	MIAS2I5	13,5	13,5
	86719E	IUP2IS	8,5	10
	99628C	MIAS2I6	3	6
	99321C	LIL5	15	14,5
	99322C	MIAS2I5	12	15,5
	28936E	MIAS2I6	12	null

Matières	codemat	titre	responsable	diplôme
	MIAS2I5	I5	E238	Deug MIAS
	MIAS2I6	I6	E426	Deug MIAS
	IUP2MA	Automates	E238	Licence IUP-MIAGE
	LIL6	Systèmes	E236	Licence Informatique
	IUP2IS	Systèmes	E526	Licence IUP-MIAGE
	MIAS2I3	Math-Info	E426	Deug MIAS
	LIL5	Algo	E426	Licence Informatique

Relation

- Représentation d'une relation (schéma restreint)
 - Le nom de la table
 - L'ensemble des noms des attributs
 - Les contraintes...

Notes	Field Name	Field Type	Description
	noe	Text [VARCHAR]	Numéro de l'étudiant
	codemat	Text [VARCHAR]	Code de la matière
	noteex	Real [REAL]	Note à l'examen
	notecc	Real [REAL]	Note au contrôle continu

Etudiants	noetu	nom	prénom	login	mdp
	CdC - Clé	CdC	CdC	CdC	CdC

Schéma

- Schéma d'une base de données : ensemble des schémas de toutes les relations.

Étudiants (noetu : CdC, nom : CdC, prénom : CdC, mdp : CdC, login : CdC)
Matières (codemat : CdC, titre : CdC, responsable : CdC, diplôme : CdC)
Notes (#noe : CdC, #codemat : CdC, noteex : [0.0,20.0], notecc : [0.0,20.0])
...

↑

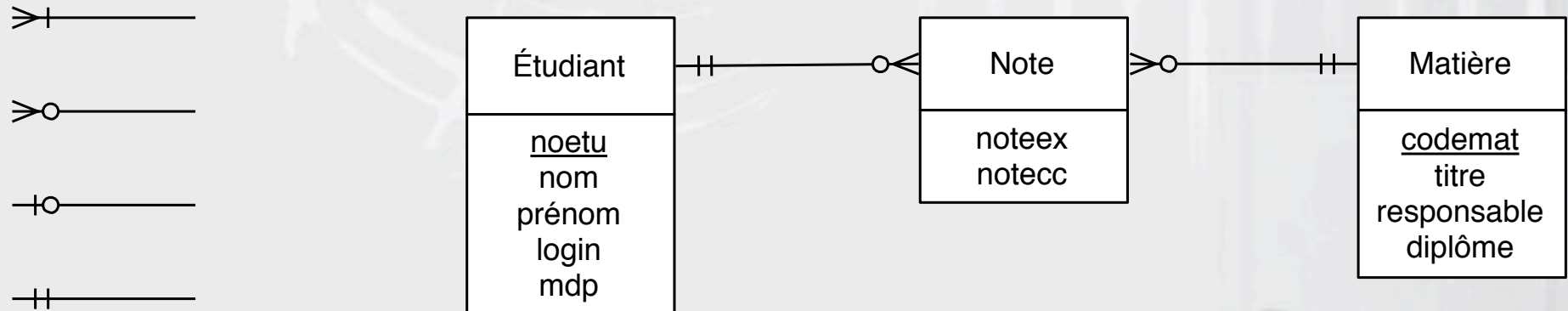
Notation simplifiée... mais souvent suffisante !

Base de donnée relationnelle

- **Base de donnée relationnelle** : base de données dont le schéma est un ensemble de schémas de tables relationnelles et dont les occurrences sont des t-uples de ces tables.
- **Expression relationnelle** : expression de syntaxe précise composée de noms de relations, d'attributs, de constantes et d'opérations. Elle décrit une relation.
- **Langage relationnel** : décrit des expressions relationnelles.

Schéma ER

- Représentation graphique du schéma de la base en mettant en évidence les liens entre les tables
 - jeux de clés primaire et de clés étrangères

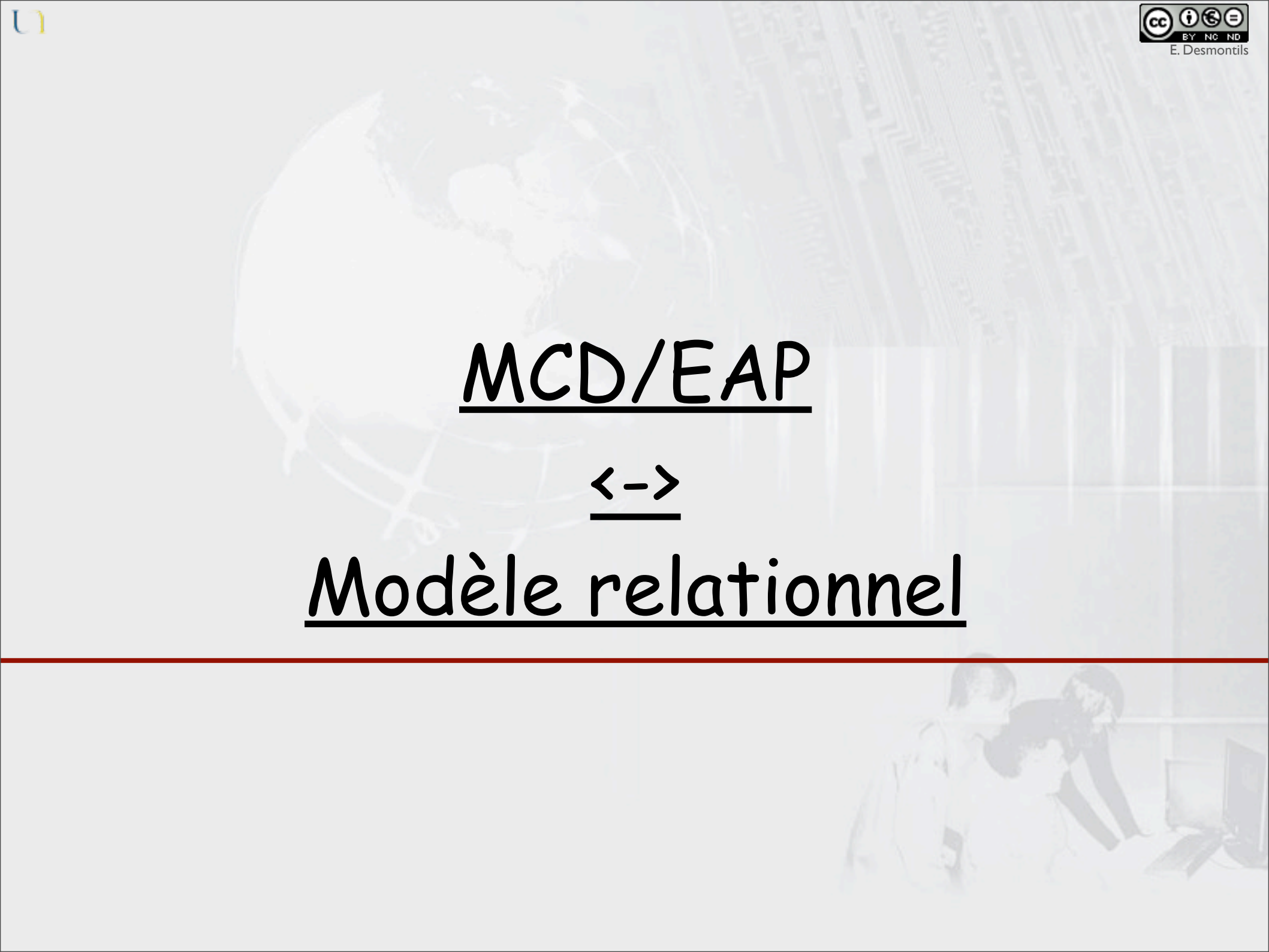


Il existe beaucoup d'autres représentations graphiques

Exemple de requête

Tous les étudiants ayant leur moyenne à l'examen de I5

```
Select noetu, nom, prenom  
From Etudiants, Notes, Matieres  
Where Etudiants.noetu=Notes.noet And  
Matiere.codemat=Notes.codemat And noteex>=10  
And titre='I5';
```

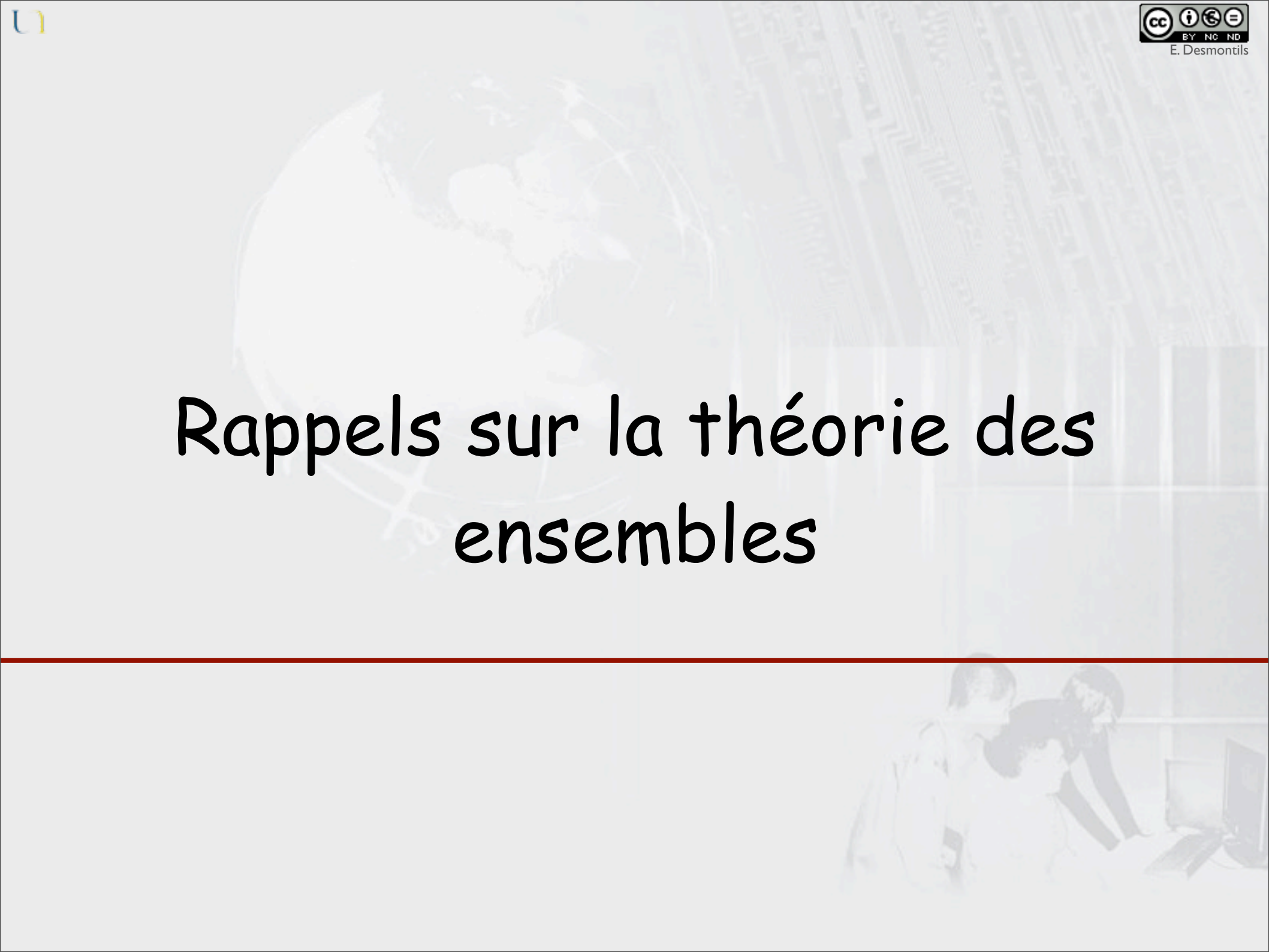


MCD/EAP

<->

Modèle relationnel





Rappels sur la théorie des ensembles



Définitions fondamentales

- Ensemble : collection non ordonnée d'objets différents.

Soit X un ensemble et p un objet :

- $p \in X$: signifie que l'objet p est un élément de l'ensemble X , ie p appartient à X
- $p \notin X$: signifie que l'objet p n'est pas un élément de l'ensemble X , ie p n'appartient pas à X
- $\{1,4\} = \{4,1\}$
 - Les éléments peuvent être présentés dans un ordre arbitraire.
- Un ensemble ne peut pas contenir deux éléments égaux
 - ~~$\{1,1,1\}$~~
Les éléments 1 et 1 ne sont pas distincts. Ils représentent le même élément.

Définir et représenter un ensemble

- Définir un ensemble

- Par extension, énumération exhaustive des éléments

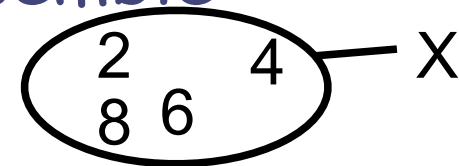
- $X = \{2, 4, 6, 8\}$

- Par caractérisation (compréhension, intension) :

- $X = \{p \mid \text{pair}(p) \text{ et } 0 < p < 10\}$

- Représentation graphique d'un ensemble

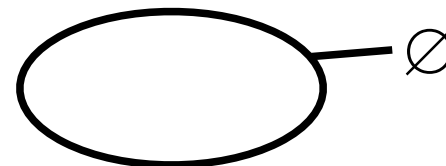
- Diagram de Venn



- Ensemble vide

- Ensemble ne contenant pas d'éléments

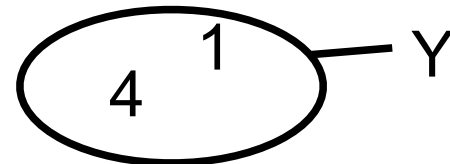
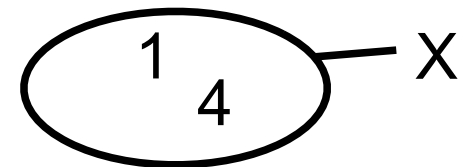
- $\{\} = \emptyset$



- Attention, $\{0\} \neq \{\}$

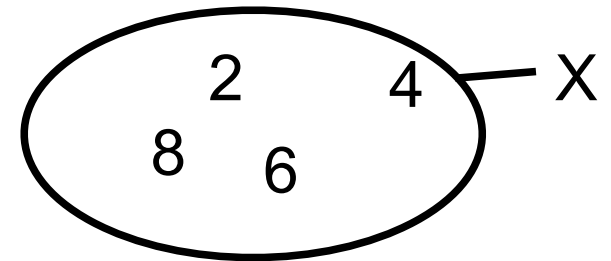
Égalité

- Égalité de deux ensembles :
 - deux ensembles X et Y sont égaux si et seulement si
 - chaque élément de X est aussi un élément de Y
 - ET que chaque élément de Y est aussi un élément de X
 - $X=Y$
 - $\forall x \in X$ alors $x \in Y$
et $\forall y \in Y$ alors $y \in X$



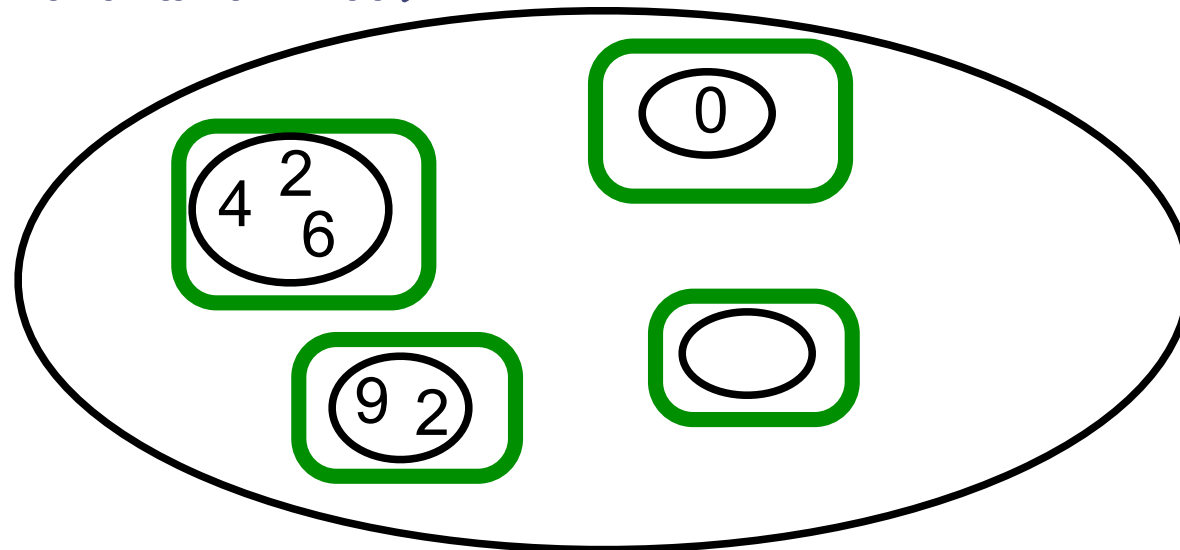
Cardinalité

- Cardinalité d'un ensemble :
 - le nombre d'éléments de l'ensemble



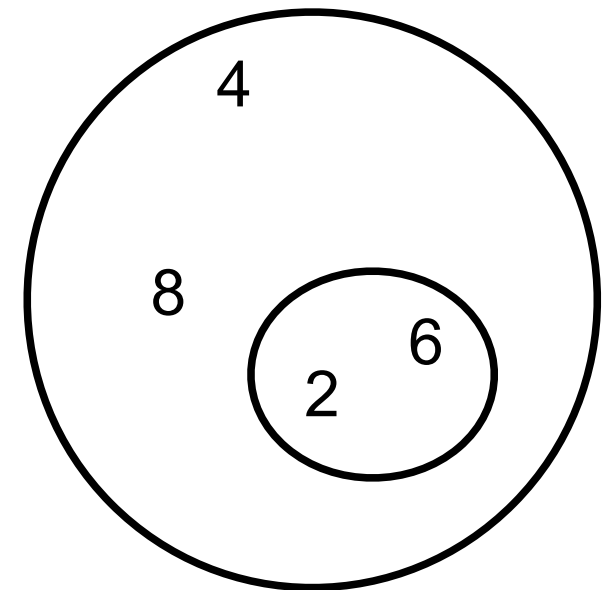
- $\text{Cardinalité}(X) = \text{Card}(X) = |X| = 4$

- $\text{Card}(\{\{4,2,6\},\{0\},\{\},\{9,2\}\}) = 4$

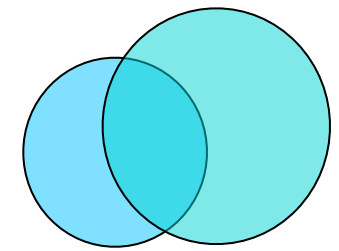


Sous-ensemble et inclusion

- Un ensemble Y est inclus dans un ensemble X si chaque élément de Y est aussi un élément de X .
- $Y \subseteq X$
- $Y \subseteq X$ et $Y \neq X \Leftrightarrow Y \subset X$
 - Y est appelé aussi sous-ensemble propre de X
- $Y \subseteq X$ et $X \subseteq Y \Rightarrow X=Y$
- X est parfois appelé ensemble universel
- $\{2,6\} \subset \{4,6,8,2\}$
- $\forall X, \emptyset \subseteq X$

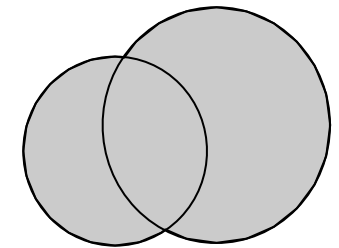


Opérations ensemblistes (1)



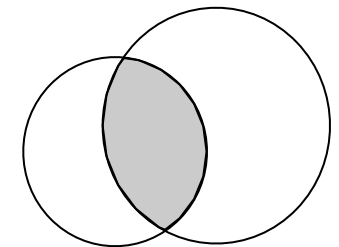
- **Union**

- $Z = X \cup Y$ ssi $Z = \{z \mid z \in X \vee z \in Y\}$
- $\{2,4,6,8\} \cup \{1,3,5,7,9\} = \{1,2,3,4,5,6,7,8,9\}$
- $X \cup \emptyset = X \cup \emptyset = X$

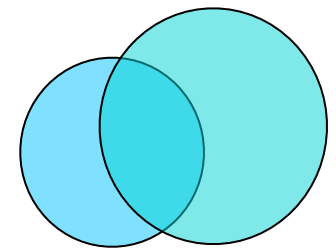


- **Intersection**

- $Z = X \cap Y$ ssi $Z = \{z \mid z \in X \wedge z \in Y\}$
- $\{2,4,6,8\} \cap \{1,2,3,4,5\} = \{2,4\}$
- $X \cap \emptyset = X \cap \emptyset = \emptyset$
- X et Y sont disjoints ssi $X \cap Y = \emptyset$
 - $\{1,3,5,7,9\}$ et $\{2,4,6,8\}$ sont disjoints



Opérations ensemblistes (2)



- **Ensemble complémentaire** : le complémentaire d'un sous-ensemble A d'un ensemble universel X est l'ensemble de tous les éléments qui ne sont pas dans A .

- $C_X^A = A^c = X - A = \{x \mid x \in X \wedge x \notin A\}$ avec $A \subseteq X$

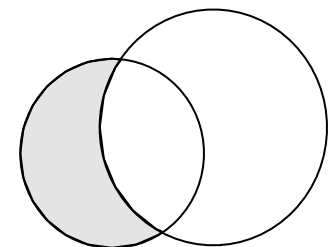
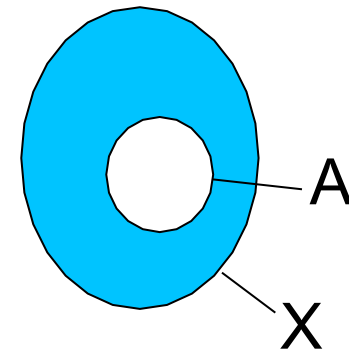
- $X = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}; A = \{2, 4, 6, 8\}$
 $A^c = \{1, 3, 5, 7, 9\}$

- $X^c = \emptyset; \emptyset^c = X; A^{cc} = A; A^c \cap A = \emptyset; A^c \cup A = X$

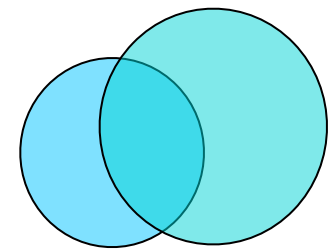
- **Différence** : $X \setminus Y$ sont les éléments de X qui n'appartiennent pas à Y

- $X \setminus Y = X - Y = \{x \mid x \in X \wedge x \notin Y\}$

- $\{1, 2, 3, 4, 5, 6, 7, 8, 9\} \setminus \{1, 2, 3, 5, 7, 9\} = \{4, 6, 8\}$



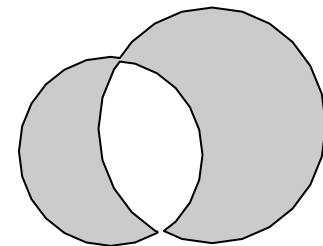
Opérations ensemblistes (3) □



- **Différence symétrique** : $X \Delta Y$ comprend les éléments dans X ou dans Y mais pas ceux qui sont dans X et dans Y

- $X \Delta Y = (X \setminus Y) \cup (Y \setminus X) = (X \cup Y) \setminus (X \cap Y)$

- $\{1,2,3,4,5,6\} \Delta \{4,5,6,7,8,9\} = \{1,2,3,7,8,9\}$



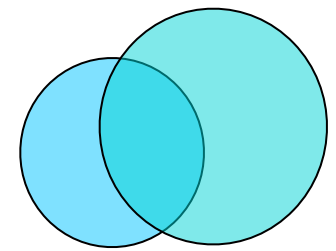
- **Produit cartésien (produit)** : $X \times Y$ est l'ensemble des couples (paires ordonnées) dont le premier élément appartient à X et le second à Y .

- $X \times Y = \{ z=(x,y) \mid x \in X \wedge y \in Y \}$

- $X=\{1,2,6\}$, $Y=\{1,6\}$,
 $X \times Y = \{(1,1), (1,6), (2,1), (2,6), (6,1), (6,6)\}$

- **Attention** $(1,6) \neq (6,1)$

Opérations ensemblistes (4) □



- Ensemble des parties :

$P(X)$ est l'ensemble de tous les sous-ensembles de X

- $\forall Y \subseteq X, Y \in P(X)$

- $X = \{1, 3, 2\}$

$$P(X) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 3\}, \{2, 3\}, \{1, 2\}, \{1, 2, 3\}\}$$

Algèbre des ensembles (1) □

- Lois de commutativité

- $X \cap Y = Y \cap X$
- $X \cup Y = Y \cup X$

- Lois d'associativité

- $X \cap Y \cap Z = (X \cap Y) \cap Z = X \cap (Y \cap Z)$
- $X \cup Y \cup Z = (X \cup Y) \cup Z = X \cup (Y \cup Z)$

- Lois d'identité

- $A \cup \emptyset = A \quad A \cap U = A$
- $A \cup U = U \quad A \cap \emptyset = \emptyset$

- Lois de distributivités

- $X \cap (Y \cup Z) = (X \cap Y) \cup (X \cap Z)$
- $X \cup (Y \cap Z) = (X \cup Y) \cap (X \cup Z)$
- $|X| + |Y| = |X \cup Y| + |X \cap Y|$

Algèbre des ensembles (2) □

- Lois d'idempotence
(équipotence) □

- $X \cap X = X$
- $X \cup X = X$

- Lois d'absorption

- $X \cup (X \cap Y) = X$
- $X \cap (X \cup Y) = X$

- Lois de De Morgan

- Soient $Z \subseteq X$ et $Y \subseteq X$
- $(Z \cap Y)^c = Z^c \cup Y^c$
- $(Z \cup Y)^c = Z^c \cap Y^c$

- Lois de complémentarité

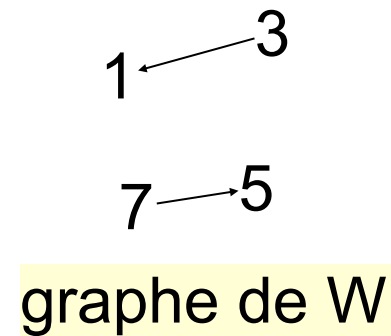
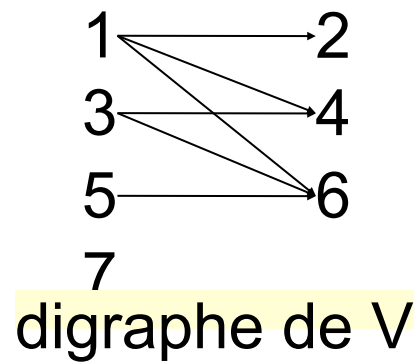
- $A \cup A^c = U$ $A \cap A^c = \emptyset$
- $U^c = \emptyset$ $\emptyset^c = U$
- $A^{cc} = A$

Notion de relation binaire (1) □

- Relation binaire entre deux ensembles A et B
 - sous ensemble R du produit cartésien $A \times B$.
 - si $A=B$, on dit simplement que R décrit une relation sur A .
- Exemples
 - Soient $A=\{1,3,5,7\}$ et $B=\{2,4,6\}$
 - $U=\{(x,y):x+y=9, x \in A, y \in B\}$ alors $U=\{(3,6),(5,4),(7,2)\}$
 - $V=\{(x,y):x < y, x \in A, y \in B\}$ alors $V=\{(1,2),(1,4),(1,6),(3,4),(3,6),(5,6)\}$
 - $W=\{(x,y):x=y+2, x \in A, y \in A\}$ dans A alors $W=\{(3,1),(7,5)\}$

Notion de relation binaire (1) □

- Représentation graphique
 - graphe orienté
 - Digraphe si c'est sur 2 ensembles
 - sous forme de matrice booléenne
 - $M[i,j] = \text{Vrai si } (i,j) \in R, \text{ Faux sinon}$



Notion de relation binaire : exemple

- Soit une relation R sur $A=\{1,2,3,4\}$



- Par extension : $R=\{(2,1), (3,2), (4,3)\}$

- Par matrice

	1	2	3	4
1	F	F	F	F
2	V	F	F	F
3	F	V	F	F
4	F	F	V	F

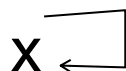
- Par intension (caractérisation) : $R=\{(x,y):x-y=1\}$

Notion de relation binaire (2) □

- Propriétés des relations sur un ensemble unique A

- Réflexive

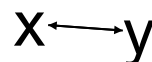
- $\forall x, xRx$



- $(x,x) \in R$

- Symétrique

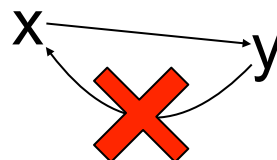
- $xRy \Rightarrow yRx$



- Si $(x,y) \in R$ alors $(y,x) \in R$

- Antisymétrique

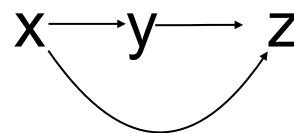
- xRy et $yRx \Rightarrow x=y$



- si $(x,y) \in R$ et $x \neq y$ alors $(y,x) \notin R$

- Transitive

- xRy et $yRz \Rightarrow xRz$



- $(x,y) \in R$ et $(y,z) \in R$ alors $(x,z) \in R$

Notion de relation binaire : exemples

- La relation d'inclusion d'ensembles
 - Réflexivité : $\forall X, X \subseteq X$
 - Anti-symétrie : $Y \subseteq X \text{ et } X \subseteq Y \Rightarrow X = Y$
 - Transitivité : $Y \subseteq X \subseteq Z \Rightarrow Y \subseteq Z$
- La relation « est diviseur de » sur les nombres naturels
 - Réflexive : pour tout x , x est diviseur de x
 - Non symétrique : 2 divise 6 mais 6 ne divise pas 2
 - Transitive : x divise y ($y=nx$) et y divise z ($z=my$) : x divise z ($z=mnx$)
 - Anti-symétrique : x divise y et y divise x alors $x=y$
- Propriétés de la relation « différent » sur les entiers
 - non réflexive ($x \neq x$ est faux), symétrique (si $x \neq y$ alors $y \neq x$), non transitive ($x=2$, $y=3$, $z=2$) et non anti-symétrique ($x \neq y$ et $y \neq x$ ne permet pas de conclure que $x=y$)

Notion de relation binaire (3) □

- Fermeture R^* d'une relation R sur un ensemble unique A par rapport à une propriété P
 - A ne possède pas nécessairement P
 - R^* possède P
 - $R \subseteq R^*$
 - R^* est sous-ensemble de toute autre relation qui inclut R et qui possède P
- Exemple
 - Soit $A=\{1,2,3\}$ et $R=\{(1,1),(1,2),(1,3),(3,1),(2,3)\}$
 - R est non réflexive, non symétrique et non transitive
 - Reflexivité : $R^*=\{(1,1),(1,2),(1,3),(3,1),(2,3), (2,2), (3,3)\}$
 - Symétrie : $R^*=\{(1,1),(1,2),(1,3),(3,1),(2,3), (2,1), (3,2)\}$
 - Transitivité : $R^*=\{(1,1),(1,2),(1,3),(3,1),(2,3), (3,2), (2,1), (3,3), (2,2)\}$

Notion de relation binaire (4) □

- Relation d'équivalence sur un ensemble A
 - Relation réflexive, symétrique et transitive
 - Abstraction de la notion d'égalité
 - exemple : xRy ssi $xy > 0$ sur les entiers (zéro exclu) □
 - les entiers de même signe
- Classe d'équivalence de x dans A par rapport à R (relation d'équivalence) □
 - ensemble $E_x = \{z \in A : zRx\}$

Notion de relation binaire (5) □

- Partition d'un ensemble A
 - collection de sous-ensembles non vides, disjoints dont la réunion donne A
 - $A = A_1 \cup A_2 \dots \cup A_n$ avec $A_i \cap A_j = \emptyset$ pour $i \neq j$
 - les A_i sont les blocs de la partitions
- Si R est une relation d'équivalence sur A , les différences classes d'équivalence forment une partition de A

Notion de relation binaire (6) □

- Ordre partiel sur un ensemble A
 - Relation réflexive, antisymétrique et transitive
 - ensembles partiellement ordonnés (posets en anglais) □
 - Soit R un ordre partiel sur A
 - si xRy et $x \neq y$ alors x est prédécesseur de y
 - S'il n'existe pas de z entre x et y alors x est prédécesseur immédiat de y
 - $x < y$
 - Diagramme de Hasse
 - graphe dont les sommets sont les éléments de A
 - si $x < y$ alors x est placé sous y et les deux sommets sont joints par un segment

Notion de relation binaire : exemples

- Exemples d'ordres partiels
 - $\leq$ sur $\mathbb{R}$
 - $\subseteq$ sur les sous-ensembles
 - « est un diviseur de » sur l'ensemble des nombres naturels
 - « est un diviseur de » sur $A=\{1,2,3,6,12,18\}$

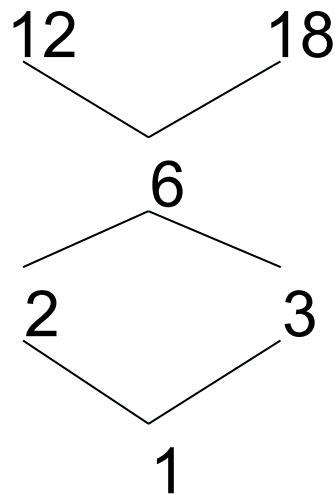


Diagramme de Hasse

Notion de relation binaire (7)□

- Ordre total
 - ordre partiel dans lequel chaque paire d'élément est en relation
- Exemples
 - $\leq$ sur $\mathbb{R}$
 - l'ordre lexicographique habituel des mots dans un dictionnaire
 - « est un diviseur de » sur $A=\{1,2,6,18\}$



Algèbre relationnel



Algèbre rationnelle et SQL

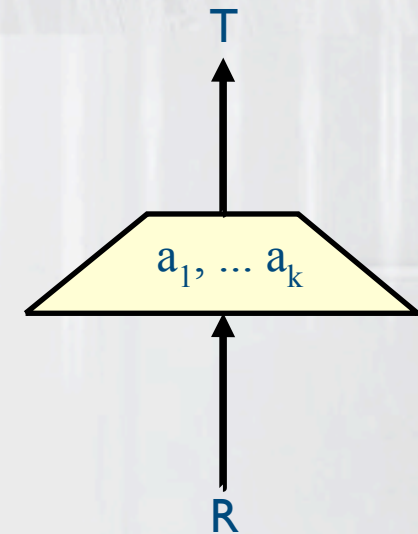
- **Algèbre relationnelle** : Codd 1970
 - Formaliser les opérations sur les ensembles.
 - Opérations de base : ensemble minimal d'opérations telles que toute autre opération peut être déduite de celles-ci.
 - Unaires (2) ou binaires (3)□
 - Opérations dérivées : opérations définies à partir des 5 opérateurs de base.
- **SQL** : langage relationnel extrêmement répandu
 - Très grande majorité des SGBD-R utilisent ce langage
 - Select ... From ... Where ... Group By ... Having ... Order By ... ;
 - Create Table ... (...);
 - ...

Opérations de bases unaires

- Opérations de base de l'algèbre relationnelle opérant sur une seule table.
- Objectif : éliminer des attributs ou des t-uples dans une relation
- **Projection** : réduire le nombre d'attributs pour ne garder que ceux considérés comme "utiles".
- **Sélection** : réduire le nombre de t-uples pour ne garder que ceux considérés comme significatifs.

La projection

Arguments	$R(a_1, a_2 \dots a_n)$
Notation	$T = \pi_{x_1, x_2 \dots x_k}(R)$ $= \text{Project}(R, x_1, x_2 \dots x_k)$ avec $\{x_1, x_2 \dots x_k\} \subseteq \{a_1, a_2 \dots a_n\}$
Schéma	$T(x_1, x_2 \dots x_k)$
Valeur	t-uples de R dont on ne garde que les attributs $x_1, x_2 \dots x_k$. Les doublons sont supprimés.
SQL	$\text{Select distinct } x_1, x_2 \dots x_k$ $\text{From } R$
Remarque	$\delta(\pi(R)) \leq \delta(R)$ et $ \pi(R) \leq R $

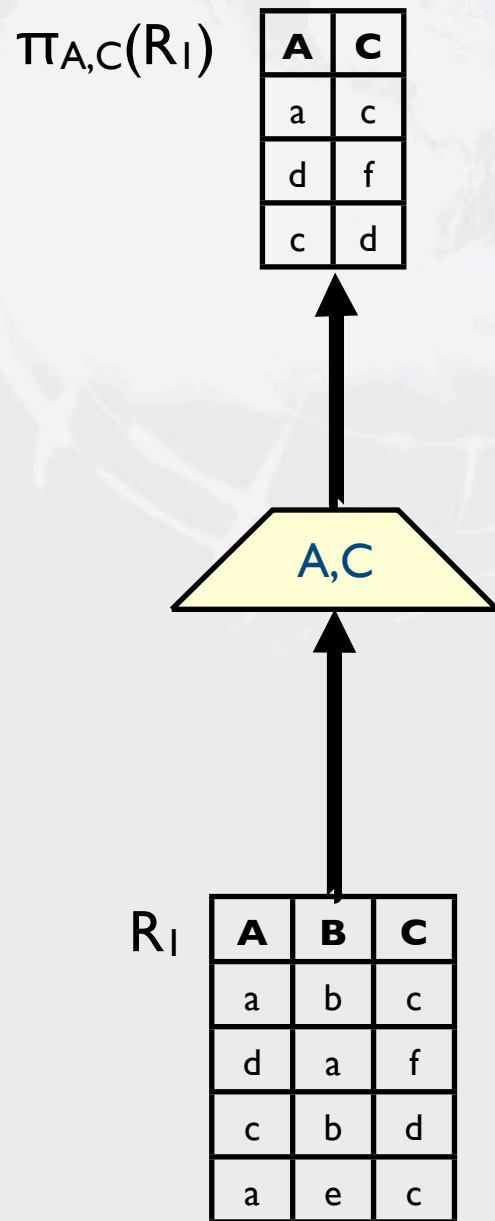


Exemple de projection

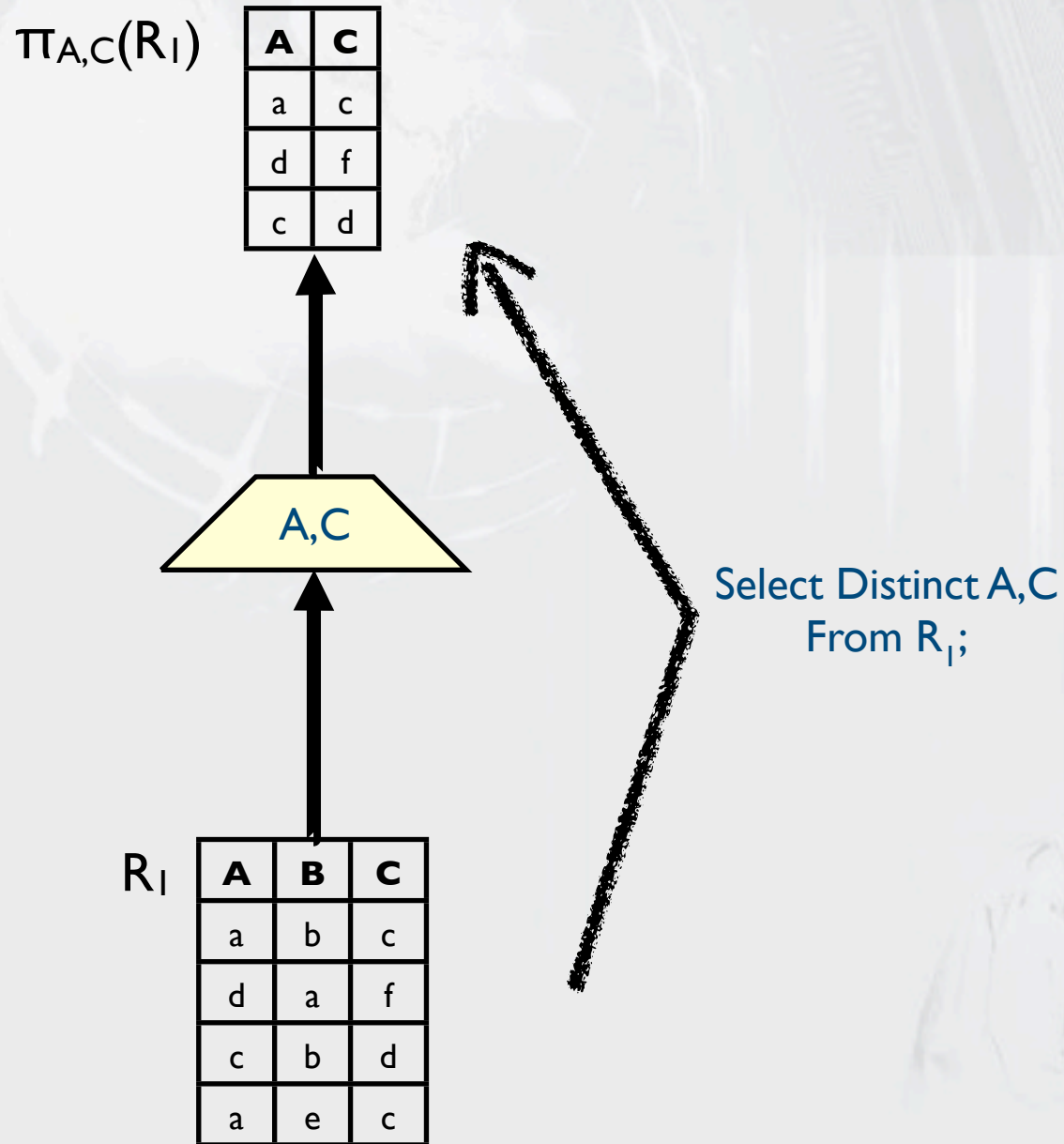
R_1

A	B	C
a	b	c
d	a	f
c	b	d
a	e	c

Exemple de projection



Exemple de projection



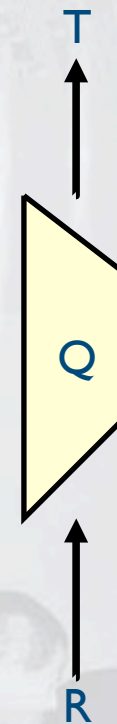
Clé d'une relation (Déf. 2) □

- Tous les t-uples de la relation se distinguent deux à deux par leur projection selon la clé.
- D'où : $|R| = |\pi_{\text{Clé}}(R)|$

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

La sélection

Arguments	$R(a_1, a_2 \dots a_n)$
Notation	$T = \sigma_Q(R)$ $= \text{Cond}(R, Q)$ Q : formule logique sur les attributs
Schéma	$T(a_1, a_2 \dots a_n)$
Valeur	t-uples de R satisfaisant l'expression Q .
SQL	Select * From R Where Q
Remarque	$\delta(\sigma(R)) = \delta(R)$ et $\sigma(R) \leq R$

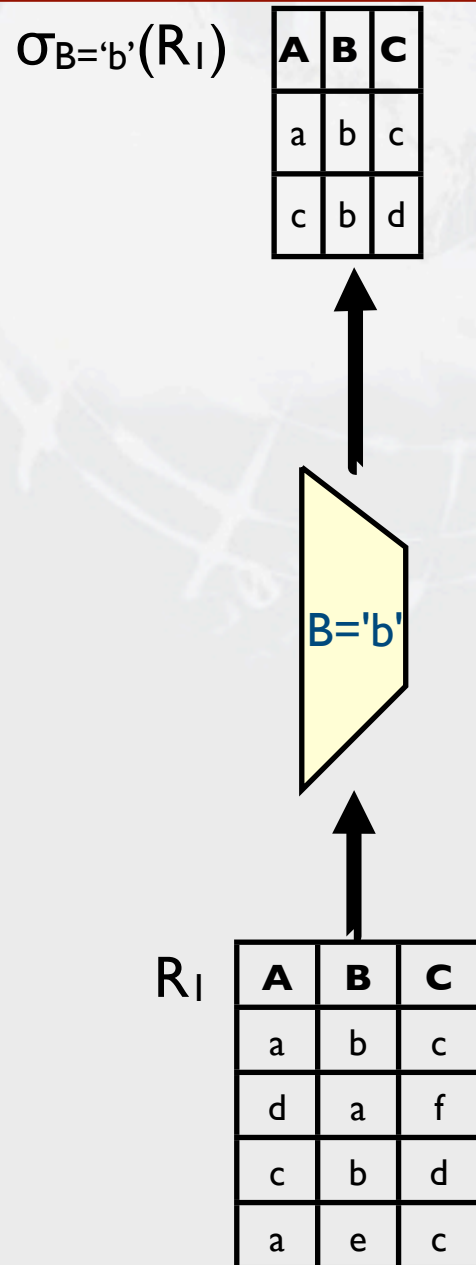


Exemple de sélection

R_I

A	B	C
a	b	c
d	a	f
c	b	d
a	e	c

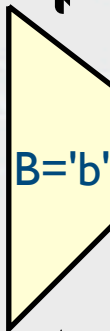
Exemple de sélection



Exemple de sélection

$\sigma_{B='b'}(R_1)$

A	B	C
a	b	c
c	b	d



R_1

A	B	C
a	b	c
d	a	f
c	b	d
a	e	c

Select *
From R_1
Where B='b';

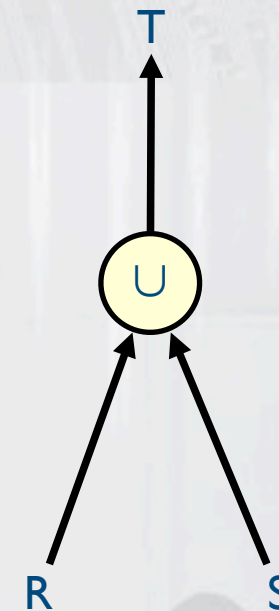
A	B	C
a	b	c
d	a	f
c	b	d
a	e	c

Opérations de base ensemblistes

- Combinaison de deux relations pour en former une troisième.
- **Union** : regroupement de données similaires.
- **Différence** : réduction d'un ensemble de données.
- **Produit cartésien** : peu utilisé directement (combinaison de données).

L'union

Arguments	$R(a_1, a_2 \dots a_n)$ $S(a_1, a_2 \dots a_n)$
Notation	$T = R \cup S = \text{Union}(R, S)$ $= \text{Or}(R, S) = \text{Append}(R, S)$
Schéma	$T(a_1, a_2 \dots a_n)$
Valeur	Union ensembliste sur $D_1 \times D_2 \dots \times D_n$ tel que $T = \{t : t \in R \vee t \in S\}$
SQL	Select * From R UNION Select * From S
Remarque	$\delta(R \cup S) = \delta(R) = \delta(S)$ et $ R \cup S \leq R + S $



Exemple d'union

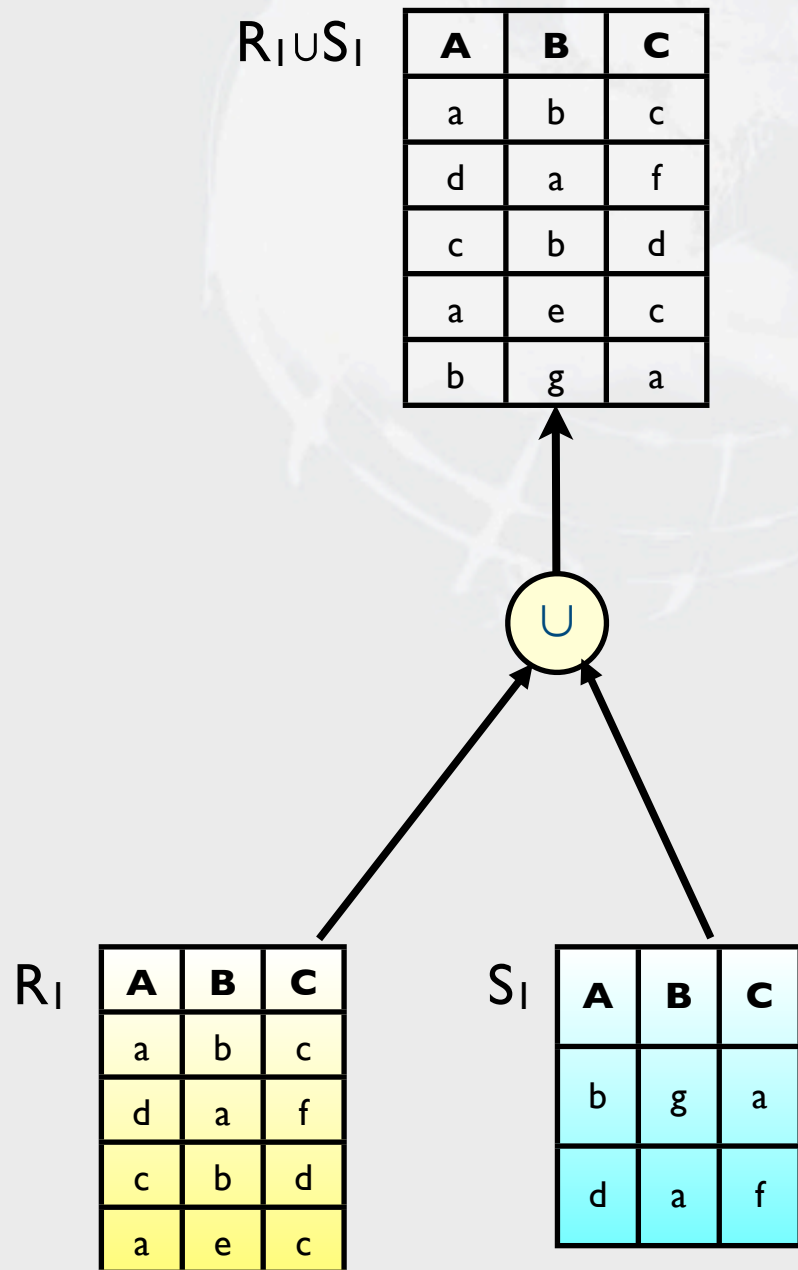
R_i

A	B	C
a	b	c
d	a	f
c	b	d
a	e	c

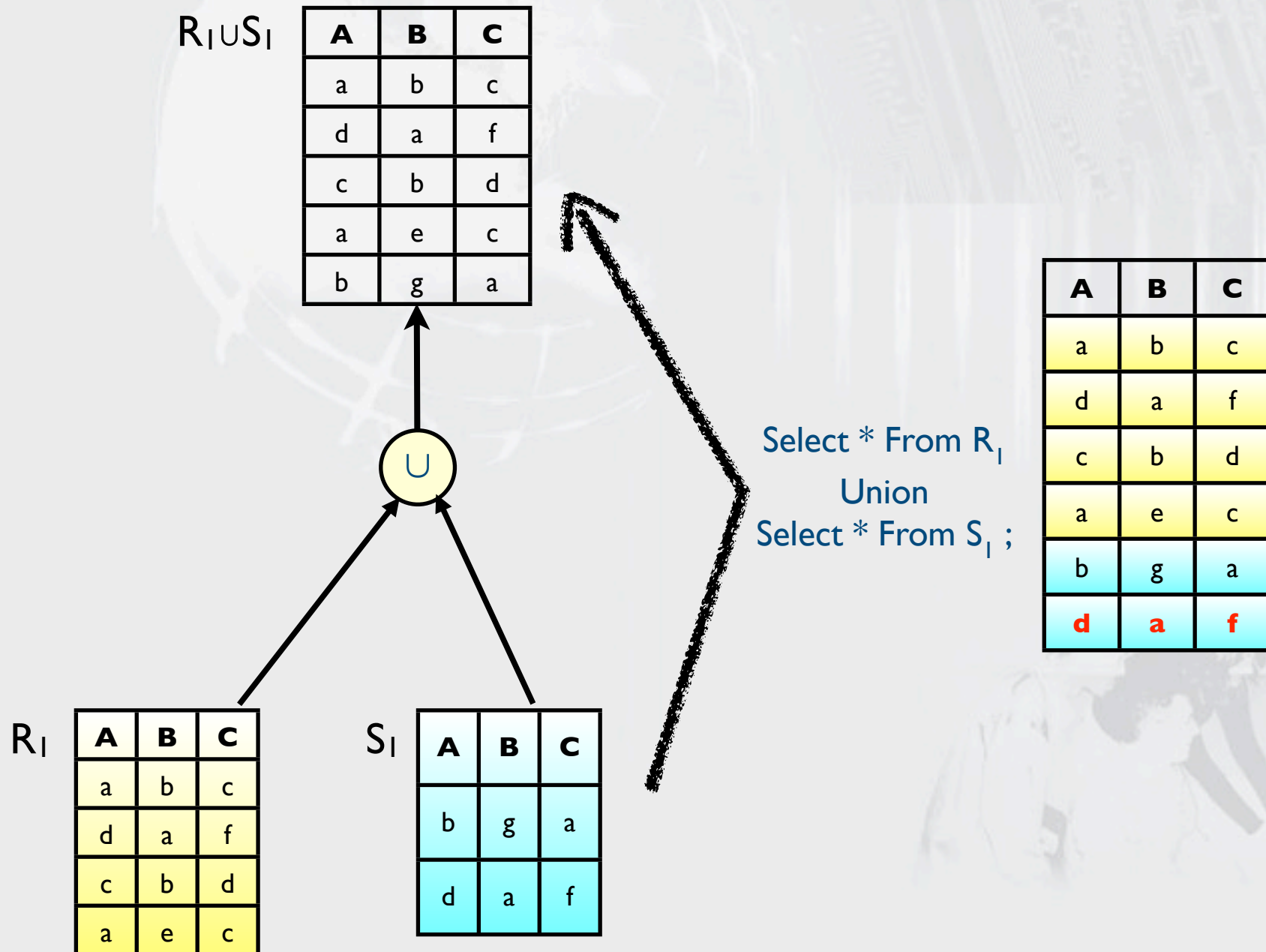
S_i

A	B	C
b	g	a
d	a	f

Exemple d'union

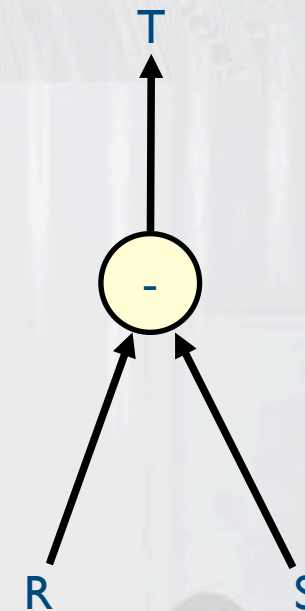


Exemple d'union



La différence

Arguments	$R(a_1, a_2 \dots a_n)$ $S(a_1, a_2 \dots a_n)$
Notation	$T = R - S = \text{Diff}(R, S)$ $= \text{Remove}(R, S) = \text{Minus}(R, S)$
Schéma	$T(a_1, a_2 \dots a_n)$
Valeur	Différence ensembliste sur $D_1 \times D_2 \dots \times D_n$ tel que $T = \{t : t \in R \wedge t \notin S\}$
SQL	Select * From R EXCEPT Select * From S
Remarque	$\delta(R-S) = \delta(R) = \delta(S)$ et $ R-S \leq R $



Exemple de différence

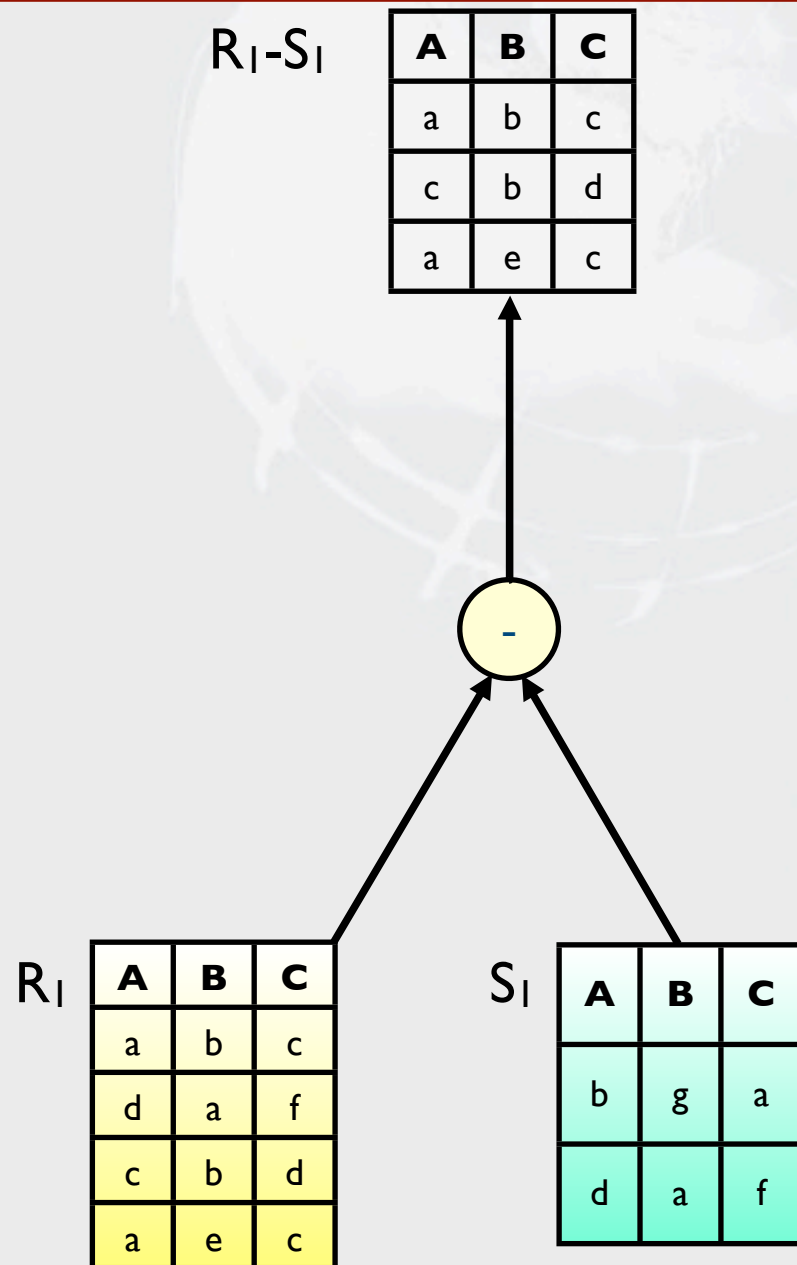
R_I

A	B	C
a	b	c
d	a	f
c	b	d
a	e	c

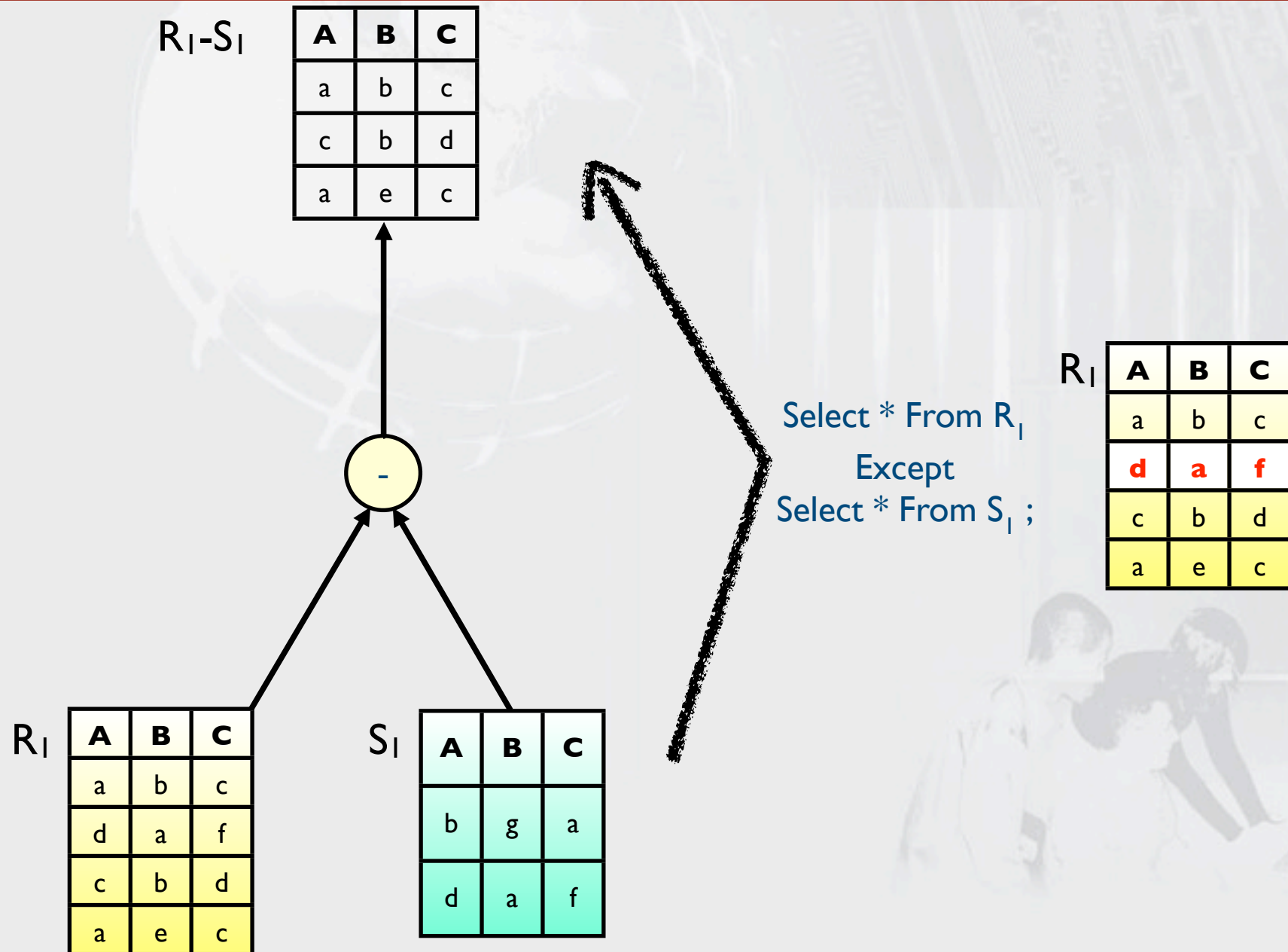
S_I

A	B	C
b	g	a
d	a	f

Exemple de différence

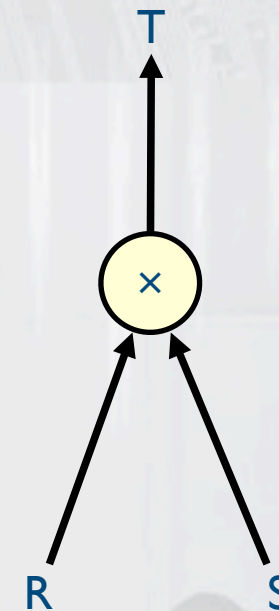


Exemple de différence



Le produit cartésien

Arguments	$R(a_1, a_2 \dots a_n)$ $S(b_1, b_2 \dots b_m)$
Notation	$T = R \times S = \text{Prod}(R, S)$ $= \text{Product}(R, S) = \text{Times}(R, S)$
Schéma	$T(a_1, a_2 \dots a_n, b_1, b_2 \dots b_m)$
Valeur	Ensemble de tous les t-uples ayant $n+m$ attributs dont les n premiers forment un t-uple de R et les m derniers un t-uple de S
SQL	$\text{Select } * \text{ From } R, S$ <i>ou</i> $\text{Select } * \text{ From } R \text{ Cross Join } S$
Remarque	$\delta(R \times S) = \delta(R) + \delta(S)$ et $ R \times S = R * S $



Exemple de produit cartésien

R_1

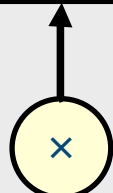
A	B	C
a	b	c
d	a	f
c	b	d
a	e	c

S_2

C	D
e	d
e	f

Exemple de produit cartésien

R₁.A	R₁.B	R₁.C	S₂.C	S₂.D
a	b	c	e	d
a	b	c	e	f
d	a	f	e	d
d	a	f	e	f
c	b	d	e	d
c	b	d	e	f
a	e	c	e	d
a	e	c	e	f



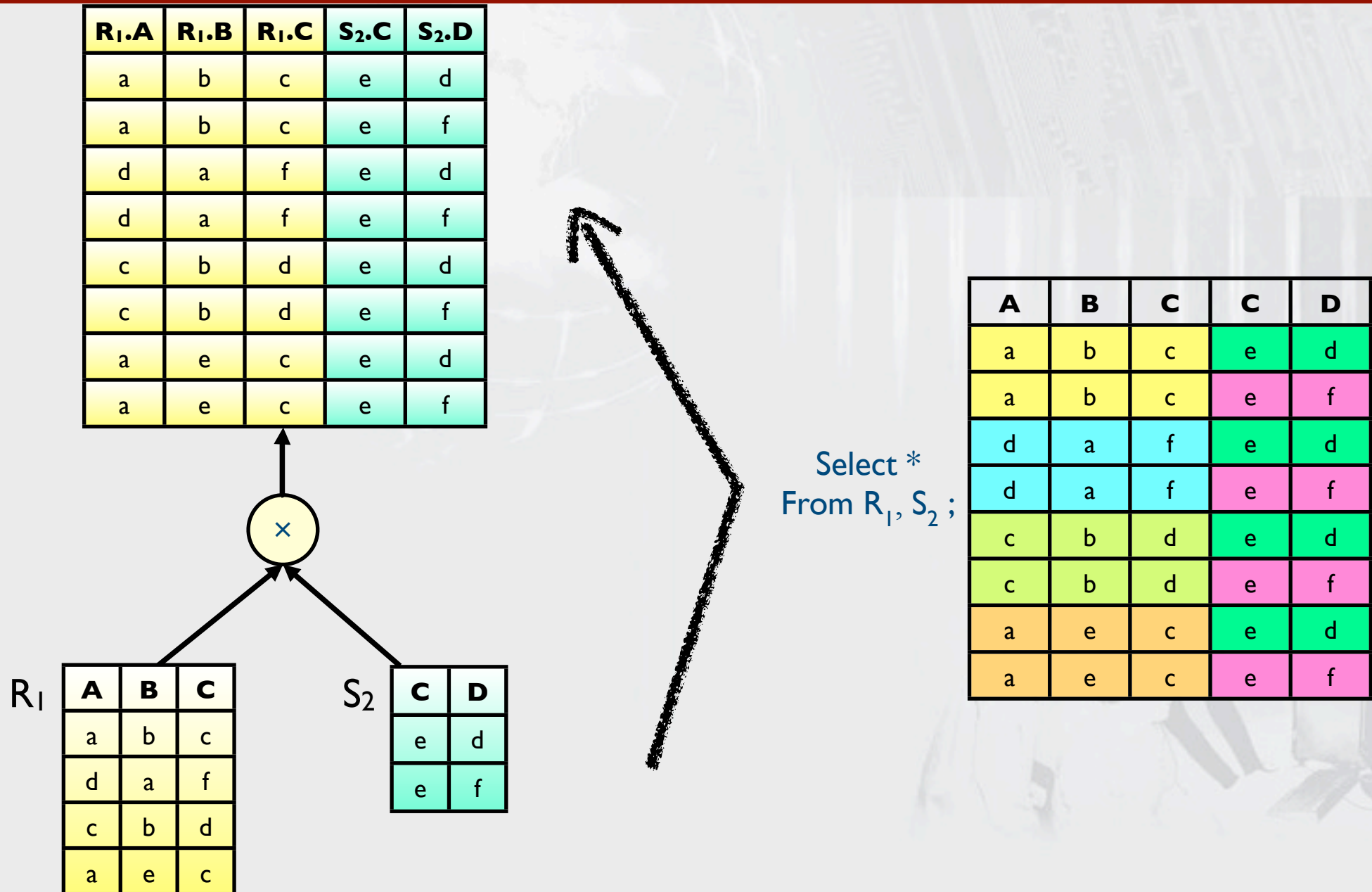
R₁

A	B	C
a	b	c
d	a	f
c	b	d
a	e	c

S₂

C	D
e	d
e	f

Exemple de produit cartésien

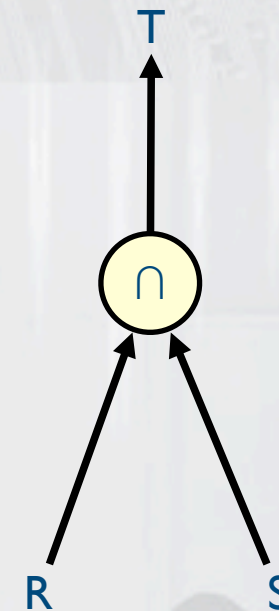


Opérations dérivées

- Intersection
- Complément
- θ -jointure : opération majeure dans les SGBDR
 - Elle permet d'enrichir des t-uples et de mettre en relation des attributs.
 - Jointure naturelle
 - Semi-jointure
 - Jointure externe

L'intersection

Arguments	$R(a_1, a_2 \dots a_n)$ $S(a_1, a_2 \dots a_n)$
Notation	$T = R \cap S = \text{Inter}(R, S)$ $= \text{And}(R, S)$
Schéma	$T(a_1, a_2 \dots a_n)$
Valeur	Intersection ensembliste sur $D_1 \times D_2 \dots \times D_n$ tel que $T = \{t : t \in R \wedge t \in S\}$ $= R - (R - S)$ $= S - (S - R)$
SQL	Select * From R INTERSECT Select * From S
Remarque	$\delta(R \cap S) = \delta(R) = \delta(S)$ et $ R \cap S \leq \min(R , S)$



Exemple d'intersection

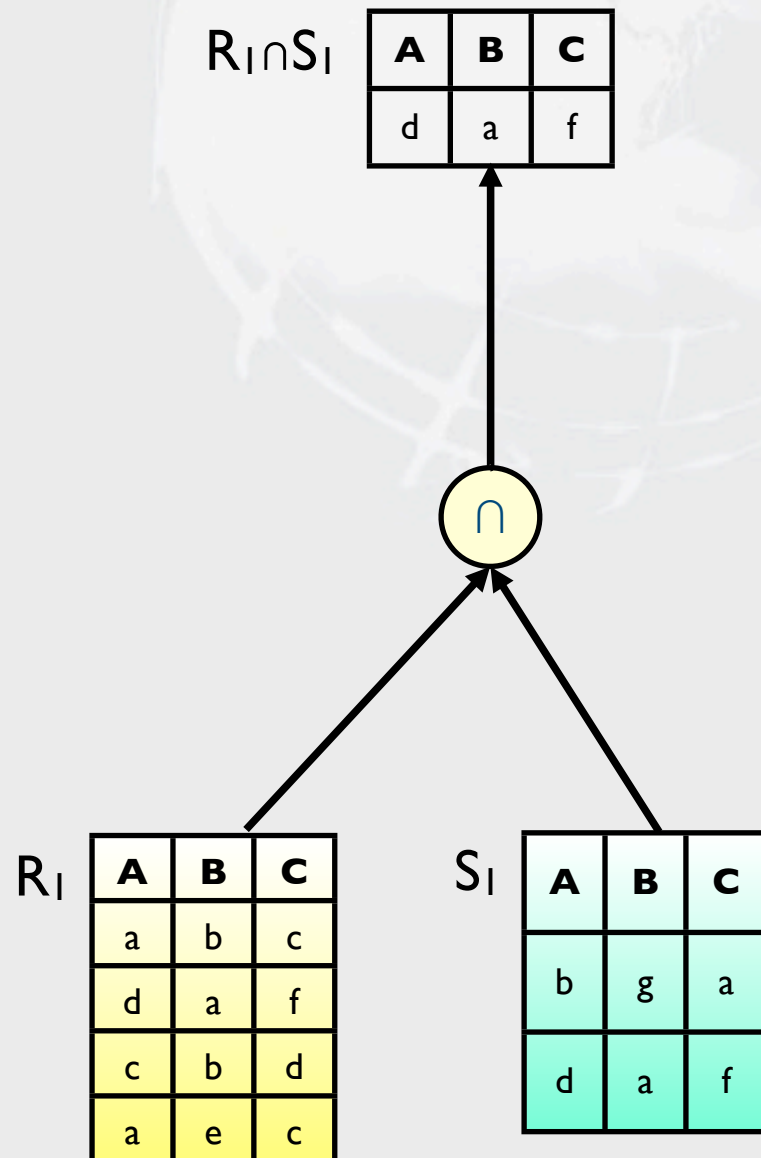
R_I

A	B	C
a	b	c
d	a	f
c	b	d
a	e	c

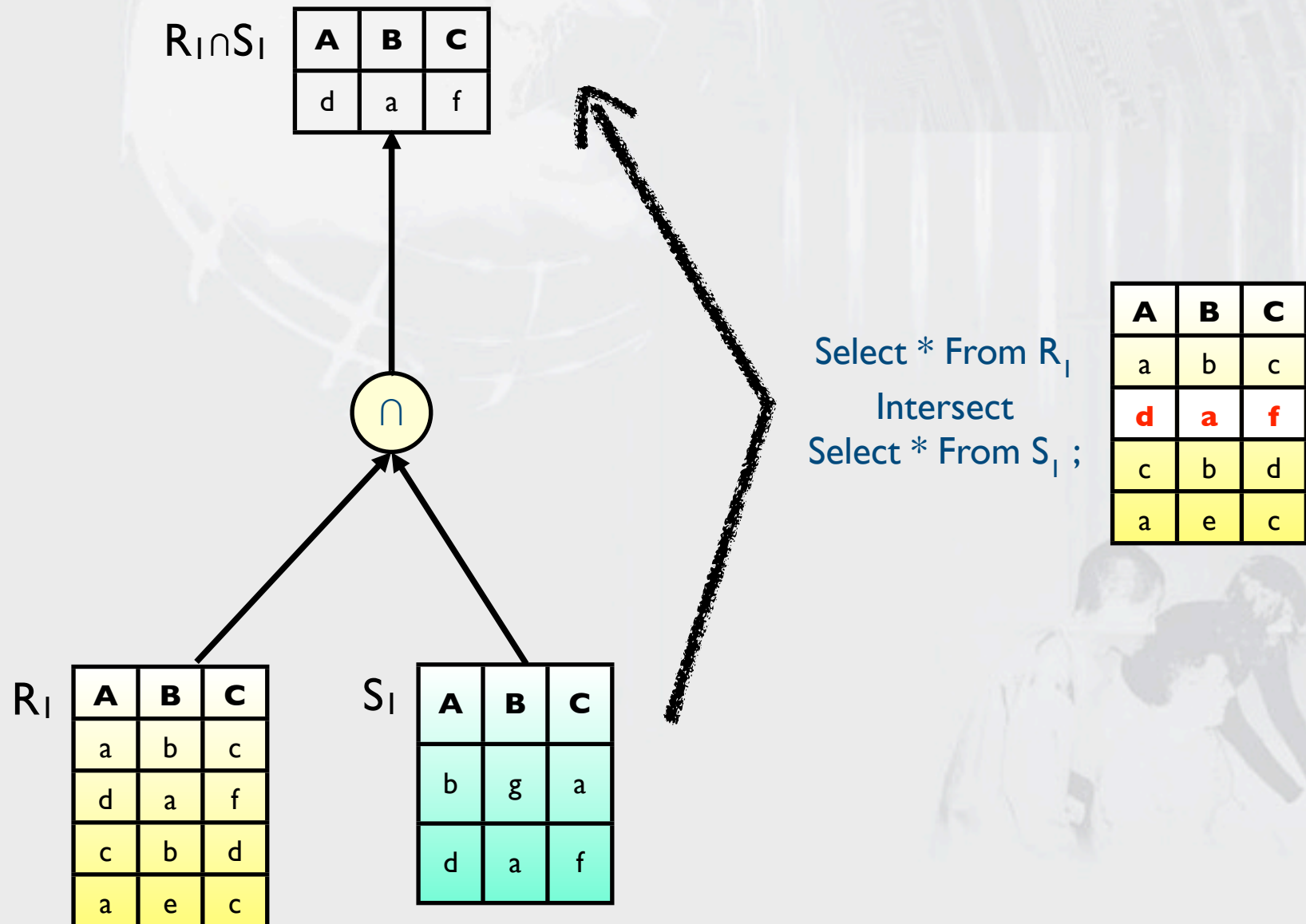
S_I

A	B	C
b	g	a
d	a	f

Exemple d'intersection



Exemple d'intersection



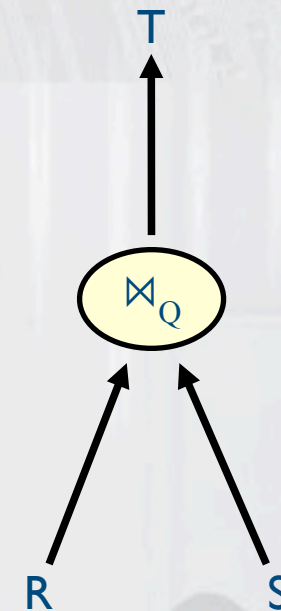
Le complément

Arguments	$R(a_1, a_2 \dots a_n)$
Notation	$T = \neg R = \text{Comp}(R)$
Schéma	$T(a_1, a_2 \dots a_n)$
Valeur	Complémentaire ensembliste sur $D_1 \times D_2 \dots \times D_n$ tel que $T = \{t : t \notin R\}$ $= (D_1 \times D_2 \dots \times D_n) - R$
SQL	
Remarque	$\delta(\neg R) = \delta(R)$



La jointure interne

Arguments	$R(a_1, a_2 \dots a_n)$ $S(b_1, b_2 \dots b_m)$
Notation	$T = R \bowtie_Q S = \text{Join}(R, S, Q) = \text{Join}_Q(R, S)$ avec Q une combinaison logique de termes de la forme $a_i \text{ Op } b_j$
Schéma	$T(a_1, a_2 \dots a_n, b_1, b_2 \dots b_m)$
Valeur	Ensemble de tous les t-uples ayant $n+m$ attributs dont les n premiers forment un t-uple de R et les m derniers un t-uple de S vérifiant la qualification Q $T = \sigma_Q(R \times S)$
SQL	Select * From R Inner Join S on Q ou Select * From R Join S on Q
Remarque	$\delta(R \bowtie_Q S) = \delta(R) + \delta(S)$ et $ R \bowtie_Q S \leq R * S $ $R \bowtie_Q R$: «auto-jointure»



Exemple de jointure

R_3

A	B	C
1	2	3
4	5	6
7	8	9

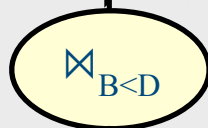
S_3

D	E
3	1
6	9

Exemple de jointure

$R_3 \bowtie_{B < D} S_3$

A	B	C	D	E
1	2	3	3	1
1	2	3	6	9
4	5	6	6	9



R_3

A	B	C
1	2	3
4	5	6
7	8	9

S_3

D	E
3	1
6	9

Exemple de jointure

$R_3 \bowtie_{B < D} S_3$

A	B	C	D	E
1	2	3	3	1
1	2	3	6	9
4	5	6	6	9

$\bowtie_{B < D}$

R_3

A	B	C
1	2	3
4	5	6
7	8	9

S_3

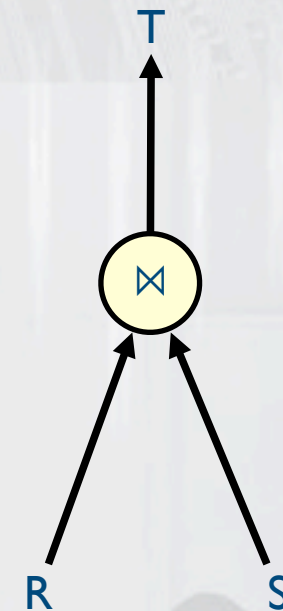
D	E
3	1
6	9

Select * From R_3 Join S_3 On $B < D$;

A	B	C	D	E
1	2	3	3	1
4	5	6	3	1
7	8	9	3	1
1	2	3	6	9
4	5	6	6	9
7	8	9	6	9

La jointure naturelle

Arguments	$R(x_1, x_2, \dots, x_p, a_1, a_2, \dots, a_n)$ $S(x_1, x_2, \dots, x_p, b_1, b_2, \dots, b_m)$
Notation	$T = R \bowtie S$
Schéma	$T(x_1, x_2, \dots, x_p, a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_m)$
Valeur	Ensemble de tous les t-uples ayant $n+m+p$ attributs dont les $n+p$ premiers forment un t-uple de R et les p premiers et les m derniers un t-uple de S.
SQL	Select * From R Natural Join S
Remarque	C'est la jointure la plus utilisée $R \bowtie S = \Pi_{x_1, x_2, \dots, x_p, a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_m}(\sigma_{\forall i \in [1..p] R_{xi} = S_{xi}}(R \times S))$ $R \bowtie S = R \bowtie_{\forall i \in [1..p] R_{xi} = S_{xi}} S$



Exemple de jointure naturelle

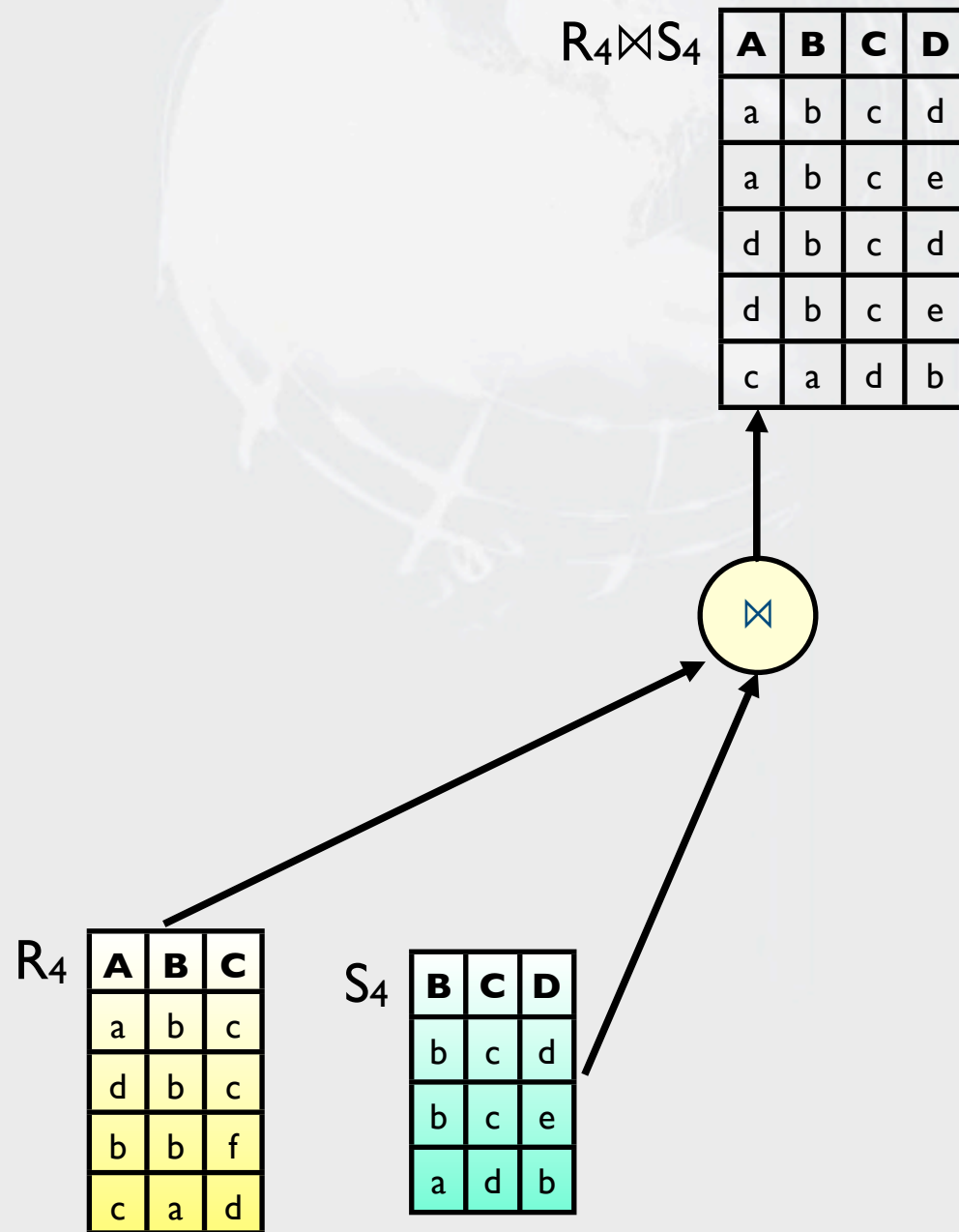
R_4

A	B	C
a	b	c
d	b	c
b	b	f
c	a	d

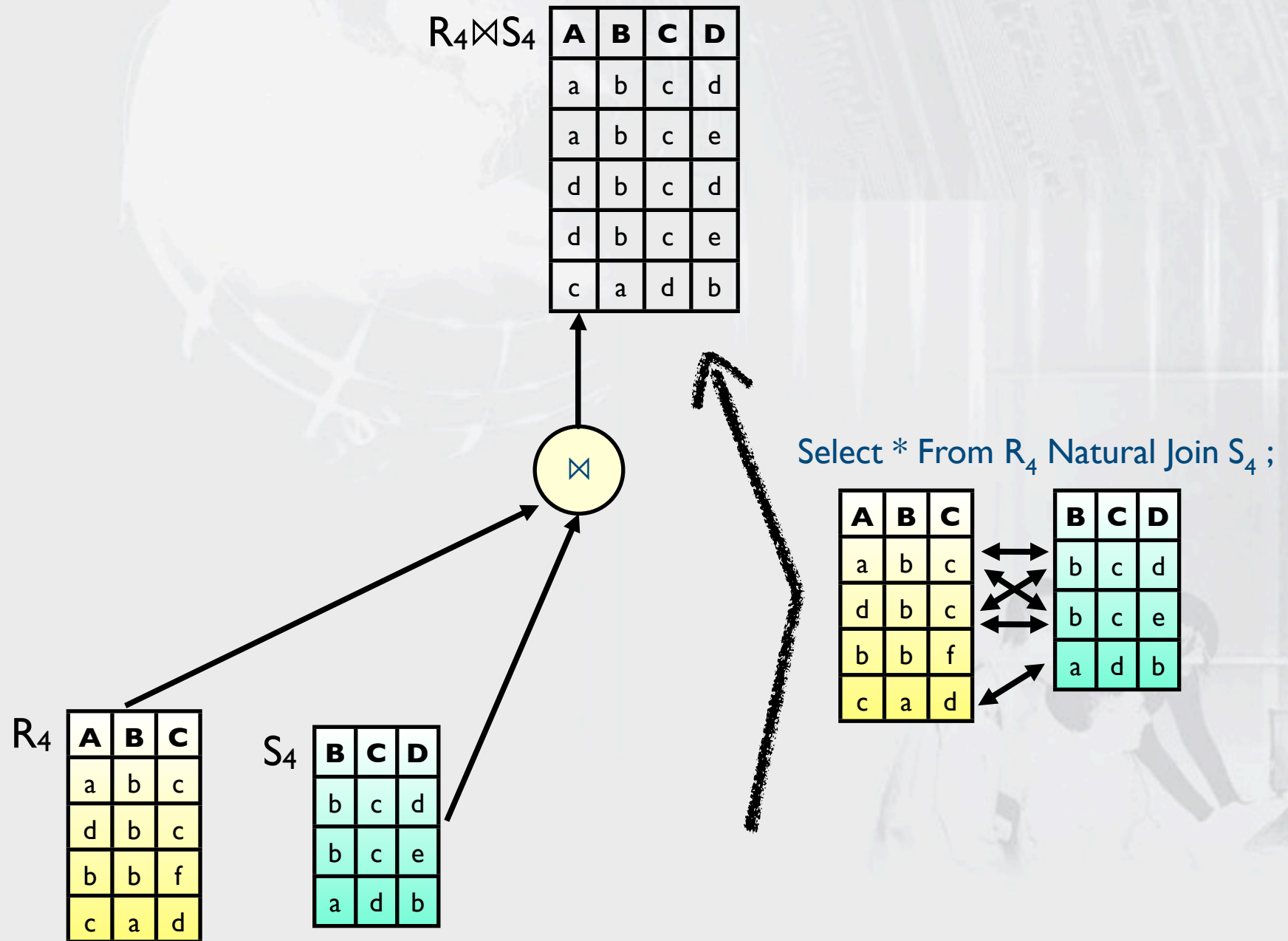
S_4

B	C	D
b	c	d
b	c	e
a	d	b

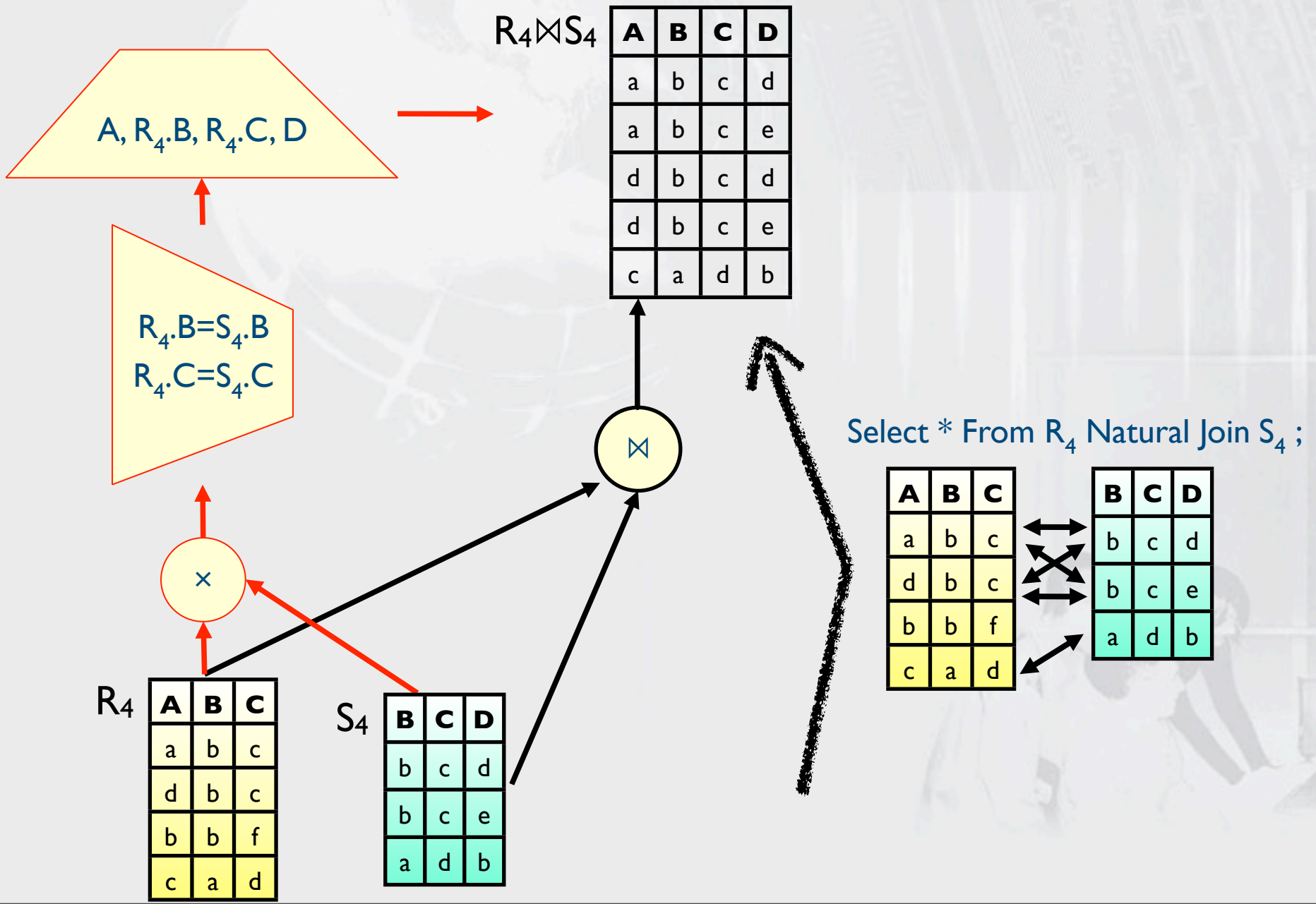
Exemple de jointure naturelle



Exemple de jointure naturelle

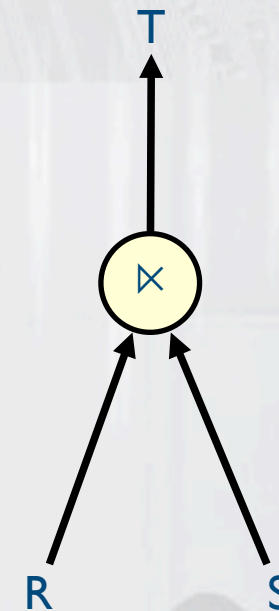


Exemple de jointure naturelle



La semi-jointure

Arguments	$R(x_1, x_2, \dots, x_p, a_1, a_2, \dots, a_n)$ $S(x_1, x_2, \dots, x_p, b_1, b_2, \dots, b_m)$
Notation	$T = R \bowtie S$
Schéma	$T(x_1, x_2, \dots, x_p, a_1, a_2, \dots, a_n)$
Valeur	Projection sur les attributs de R de la jointure naturelle de R avec S
SQL	Select $x_1, x_2, \dots, x_p, a_1, a_2, \dots, a_n$ From R Natural Join S
Remarque	$R \bowtie S = \Pi_{x_1, x_2, \dots, x_p, a_1, a_2, \dots, a_n}(\sigma_{\forall i \in [1..p] R.x_i = S.x_i}(R \times S))$



Exemple de semi-jointure

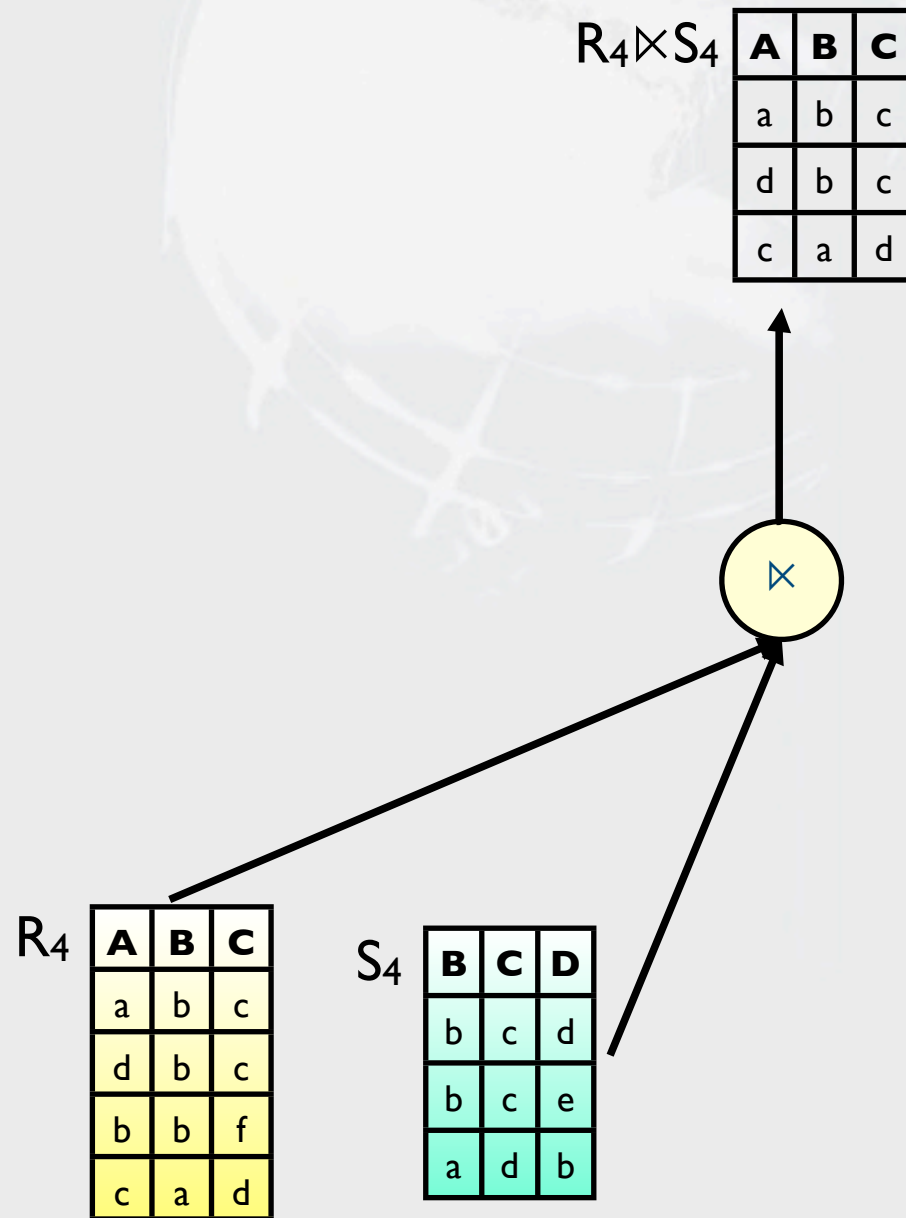
R_4

A	B	C
a	b	c
d	b	c
b	b	f
c	a	d

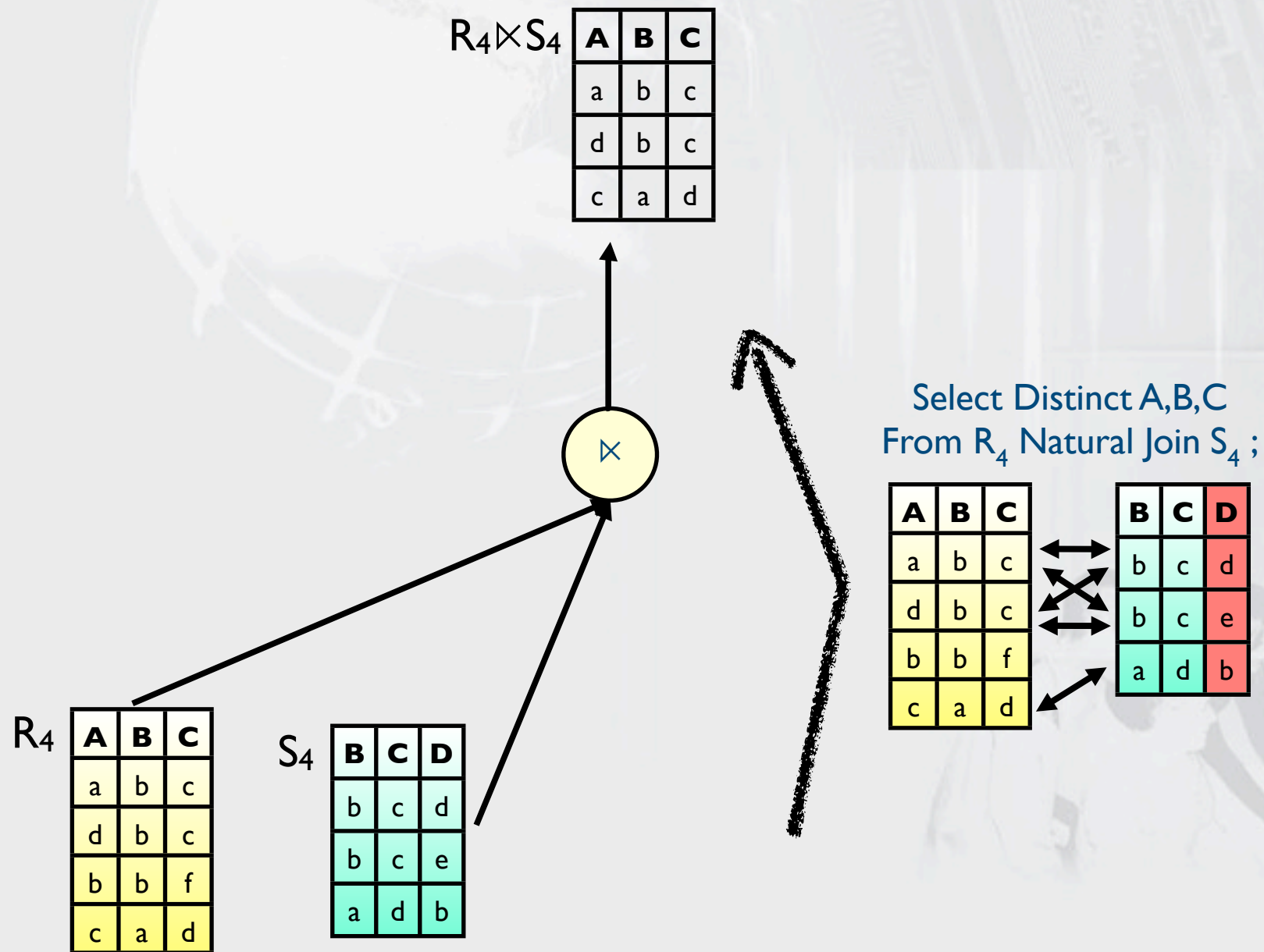
S_4

B	C	D
b	c	d
b	c	e
a	d	b

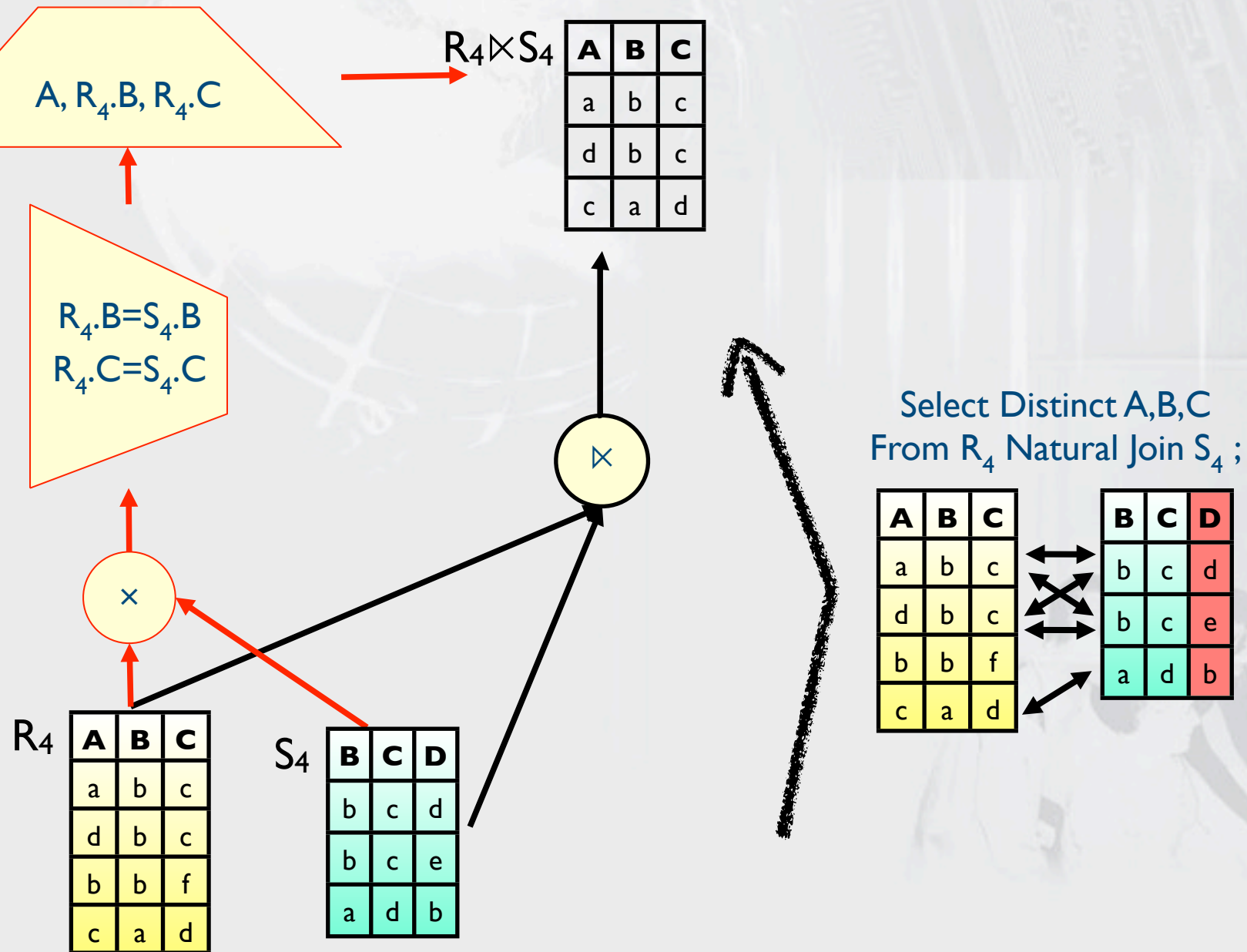
Exemple de semi-jointure



Exemple de semi-jointure

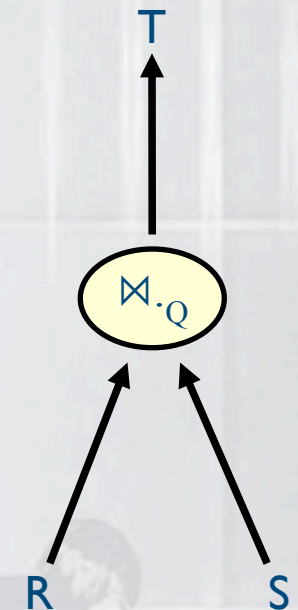


Exemple de semi-jointure



La jointure externe

Arguments	$R(a_1, a_2 \dots a_n)$ $S(b_1, b_2 \dots b_m)$
Notation	$T = R \bowtie_{\oplus Q} S$ $T = R_{\oplus} \bowtie_Q S ; T = R_{\oplus} \bowtie_{\oplus Q} S$
Schéma	$T(a_1, a_2 \dots a_n, b_1, b_2 \dots b_m)$
Valeur	Ensemble de tous les t-uples ayant $n+m$ attributs dont les n premiers forment un t-uple de R et les m derniers un t-uple de S vérifiant la qualification Q + Ajout des t-uples de la relation (appelée directrice ne vérifiant pas la condition en complétant les attribut de l'autre relation par des «null»)
SQL	Select * From R Left Outer Join S on Q Select * From R Right Outer Join S on Q Select * From R Full Outer Join S on Q
Remarque	



Exemple de jointure externe

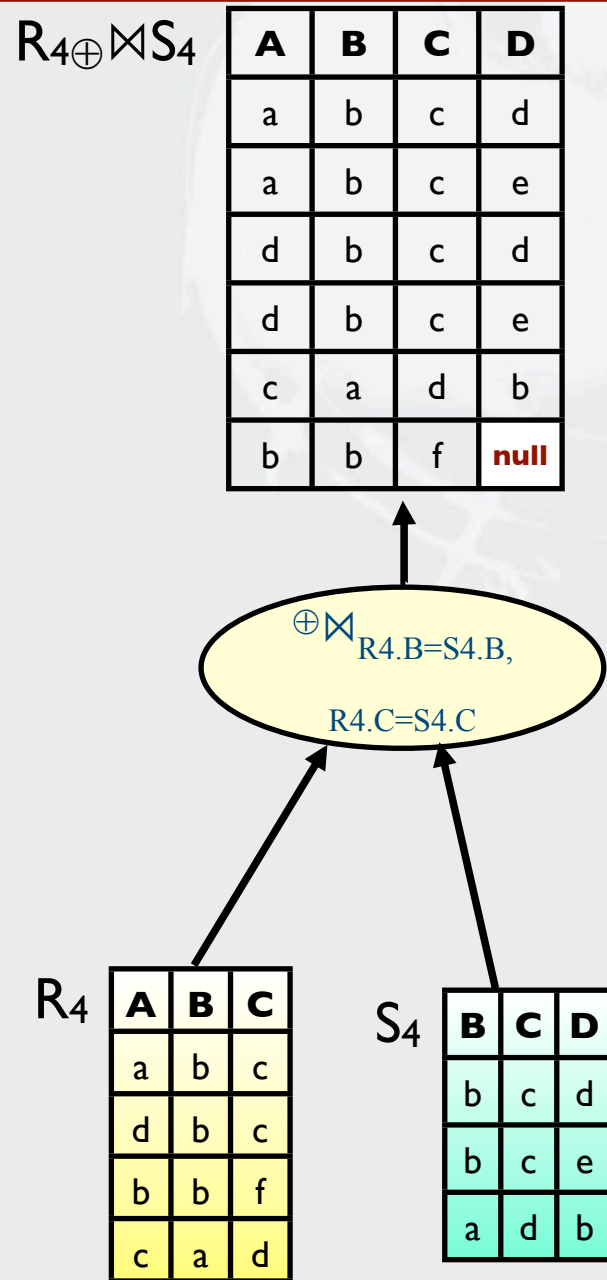
R_4

A	B	C
a	b	c
d	b	c
b	b	f
c	a	d

S_4

B	C	D
b	c	d
b	c	e
a	d	b

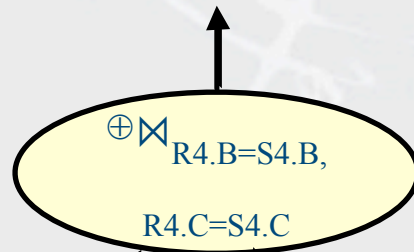
Exemple de jointure externe



Exemple de jointure externe

$R_4 \oplus \bowtie S_4$

A	B	C	D
a	b	c	d
a	b	c	e
d	b	c	d
d	b	c	e
c	a	d	b
b	b	f	null



R_4

A	B	C
a	b	c
d	b	c
b	b	f
c	a	d

S_4

B	C	D
b	c	d
b	c	e
a	d	b

Select * From R_4 Left Outer Join S_4
On $R_4.B=S_4.B$ and $R_4.C=S_4.C$;

Select * From R_4, S_4
Where $\oplus R_4.B=S_4.B$ and $\oplus R_4.C=S_4.C$;

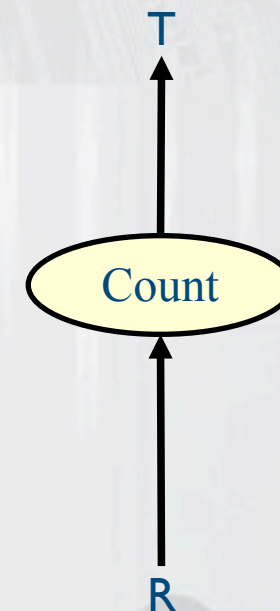
Opérations de comptage

- Opérations "outils" qui sont des extensions des opérateurs de base et non fonction de ces opérateurs.
 - Reste cependant des opérateurs ensemblistes puisqu'ils prennent des ensembles en entrée et retournent un ensemble.
- Comptage
- Somme
- Moyenne
- Minimum
- Maximum...



Comptage

Arguments	$R(a_1, a_2 \dots a_n)$
Notation	$T = \text{Compte}_{x_1, x_2 \dots x_p}(R)$ avec $\forall i \in [1..p] \ x_i \in \{a_1, a_2 \dots a_n\}$ ou $T = \text{Compte}(R)$
Schéma	$T(x_1, x_2 \dots x_p, \text{Cpt})$ ou $T(\text{Cpt})$
Valeur	Dénombrer les t-uples qui ont une même valeur d'attributs en commun. S'il n'y en a pas, c'est le nombre total de t-uples (cardinalité)
SQL	Select Count(*) From R [Group by ...]
Remarque	$\delta(\text{Count}(R)) = 1, \text{Count}(R) = 1$ $\text{Count}(R) = R $ $\delta(\text{Compte}_{x_1, x_2 \dots x_p}(R)) = p+1$ $ \text{Compte}_{x_1, x_2 \dots x_p}(R) = \pi_{x_1, x_2 \dots x_p}(R) $



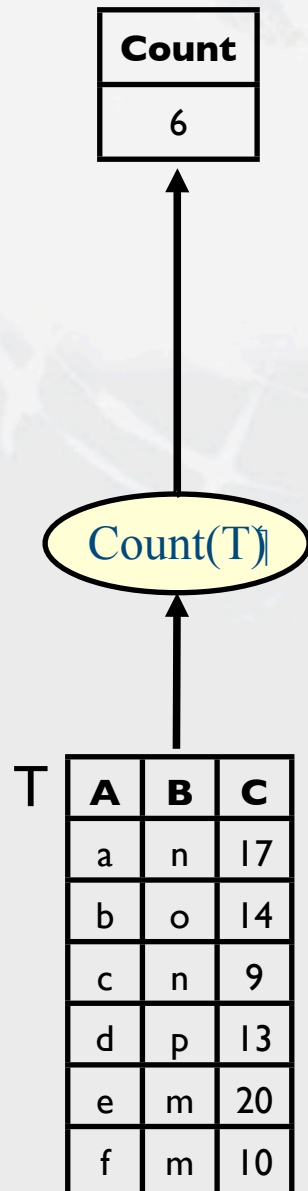
Exemples de comptage

T

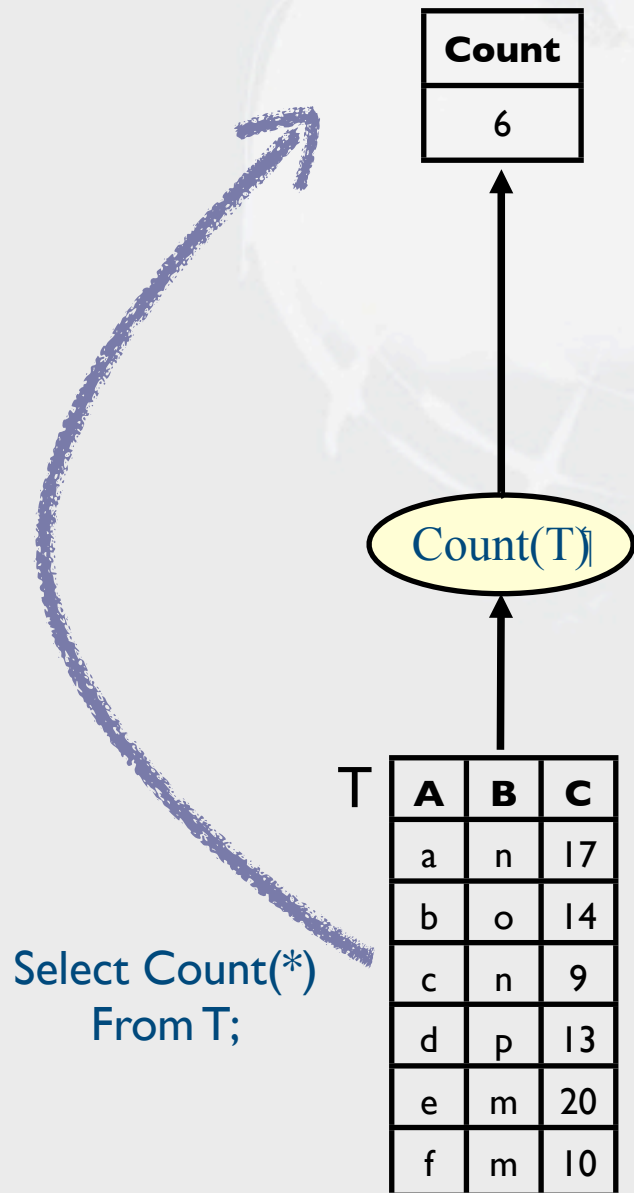
A	B	C
a	n	17
b	o	14
c	n	9
d	p	13
e	m	20
f	m	10



Exemples de comptage



Exemples de comptage

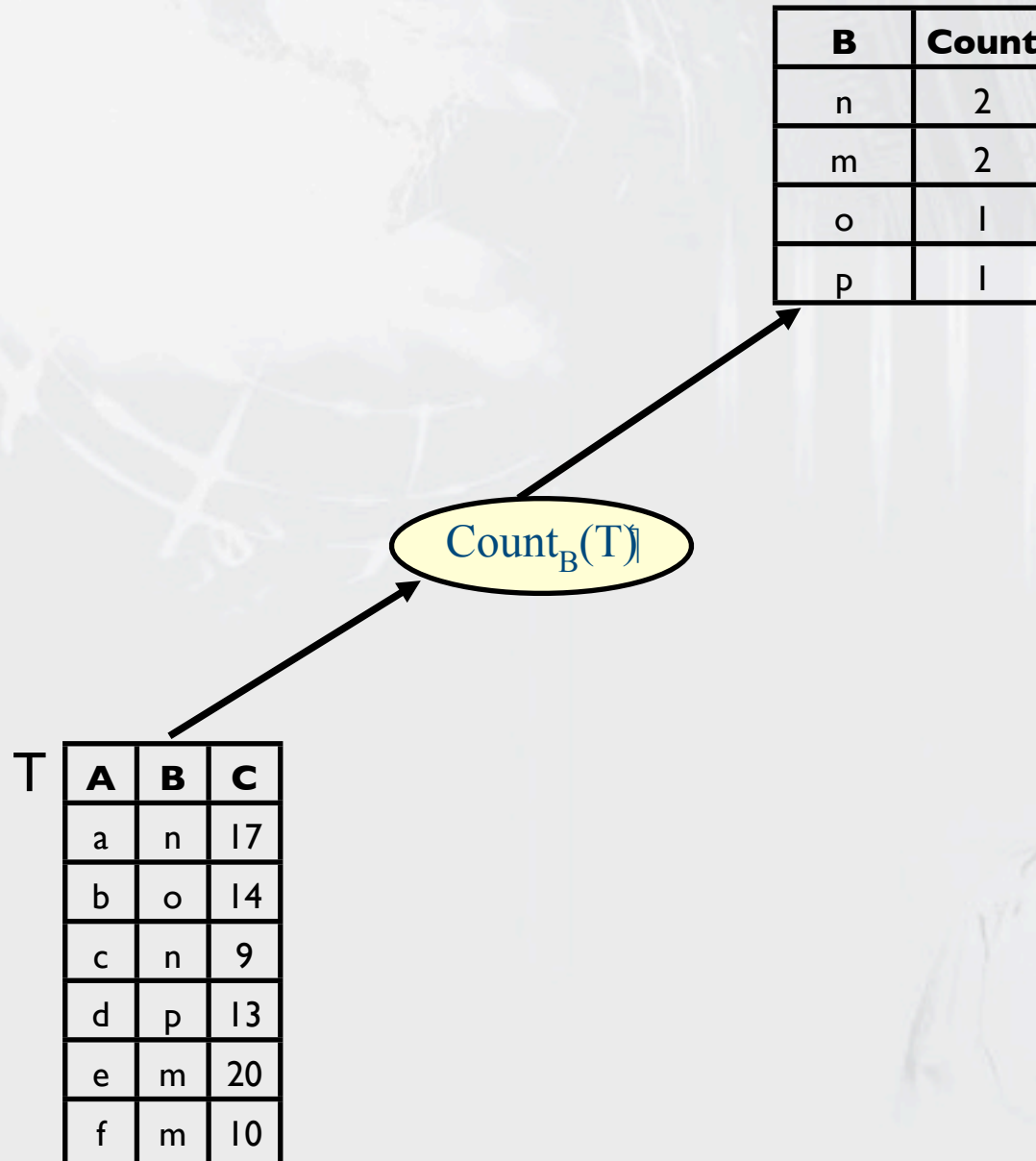


Exemples de comptage

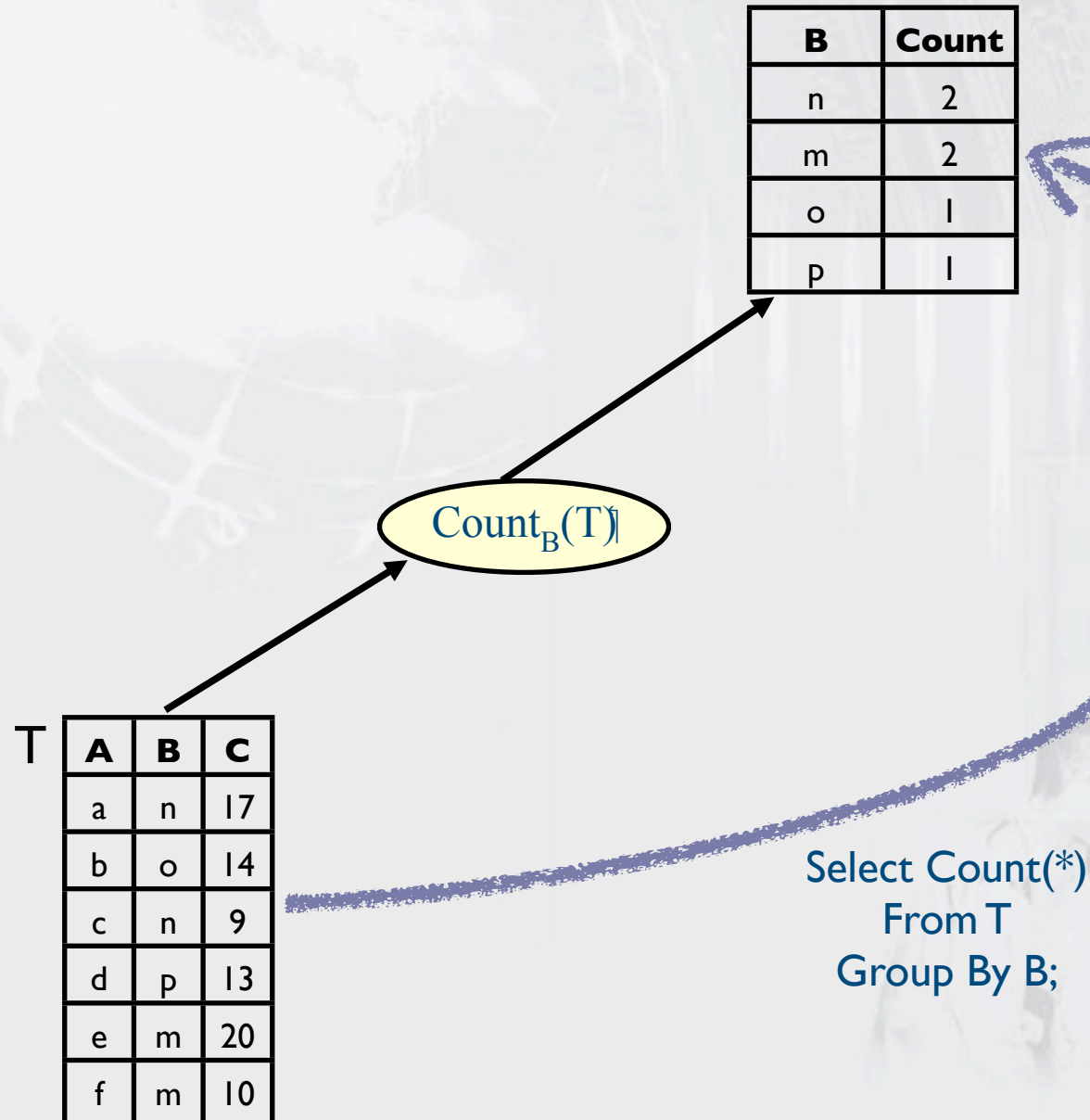
T

A	B	C
a	n	17
b	o	14
c	n	9
d	p	13
e	m	20
f	m	10

Exemples de comptage

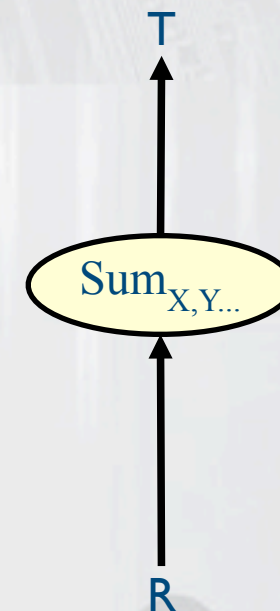


Exemples de comptage



Somme

Arguments	$R(a_1, a_2 \dots a_n)$
Notation	$T = \text{Somme}_{x_1, x_2 \dots x_p}(R, y)$ avec $\forall i \in [1..p] \ x_i \in \{a_1, a_2 \dots a_n\}$ et $y \in \{a_1, a_2 \dots a_n\}$
Schéma	$T(x_1, x_2 \dots x_p, \text{Cpt})$ ou $T(\text{Cpt})$
Valeur	Somme cumulée des valeurs de l'attribut «y» pour chacune des valeurs différentes des attributs. S'il n'y en a pas, c'est la somme sur toute la table des «y».
SQL	Select Sum(...) From R [Group by ...]
Remarque	On peut généraliser aux calculs de moyenne (avg), minimum (min), maximum (max)...



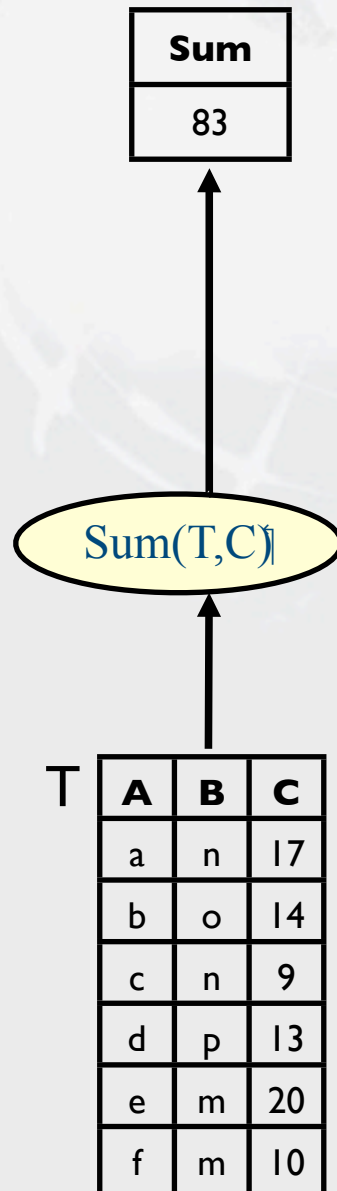
Exemples de sommes

T

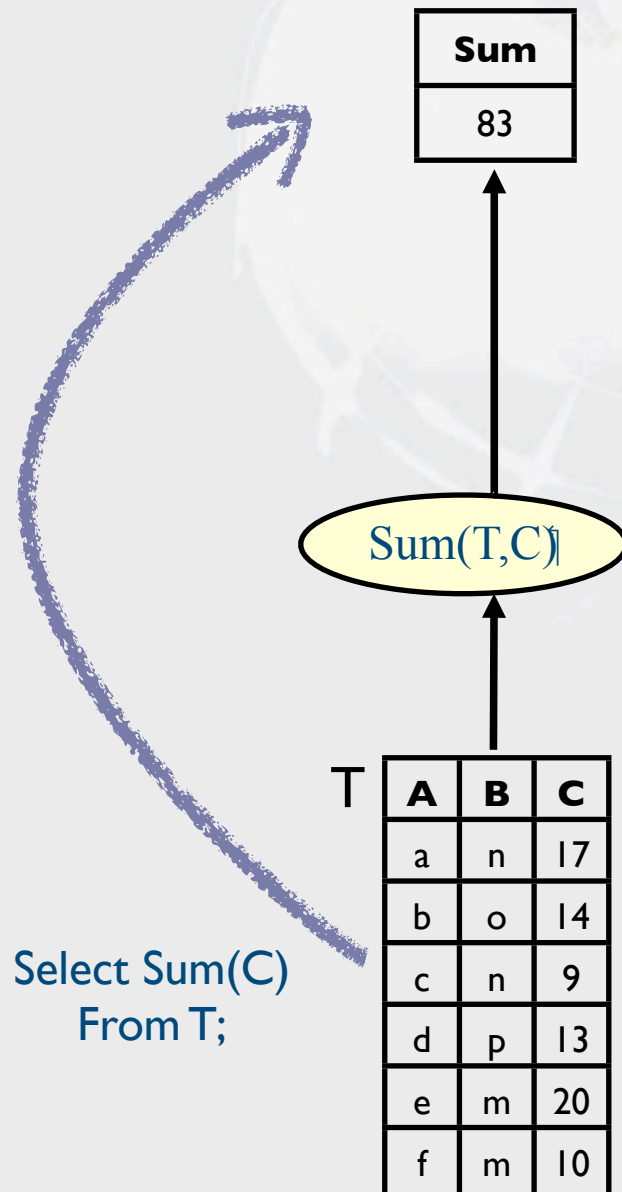
A	B	C
a	n	17
b	o	14
c	n	9
d	p	13
e	m	20
f	m	10



Exemples de sommes



Exemples de sommes

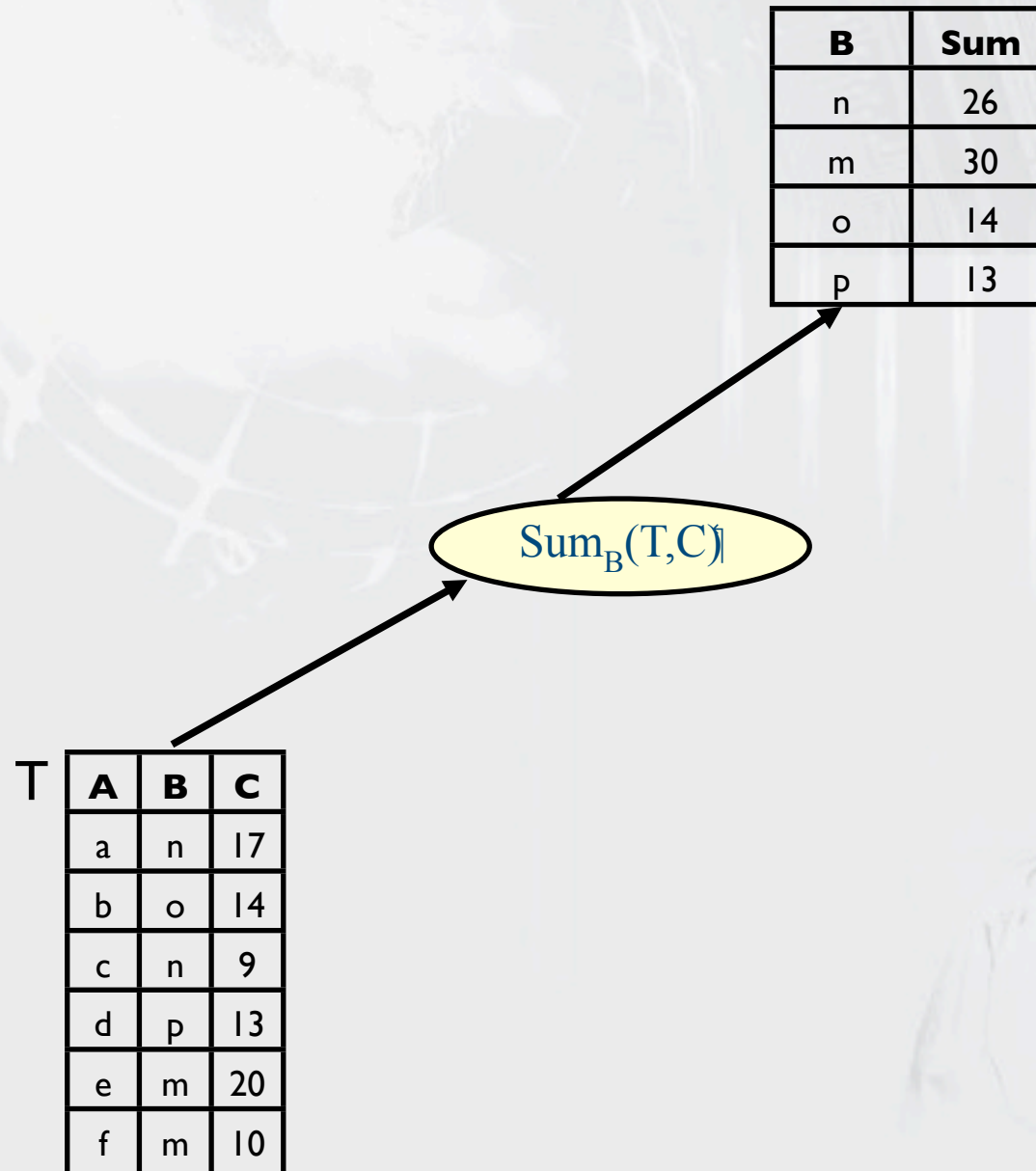


Exemples de sommes

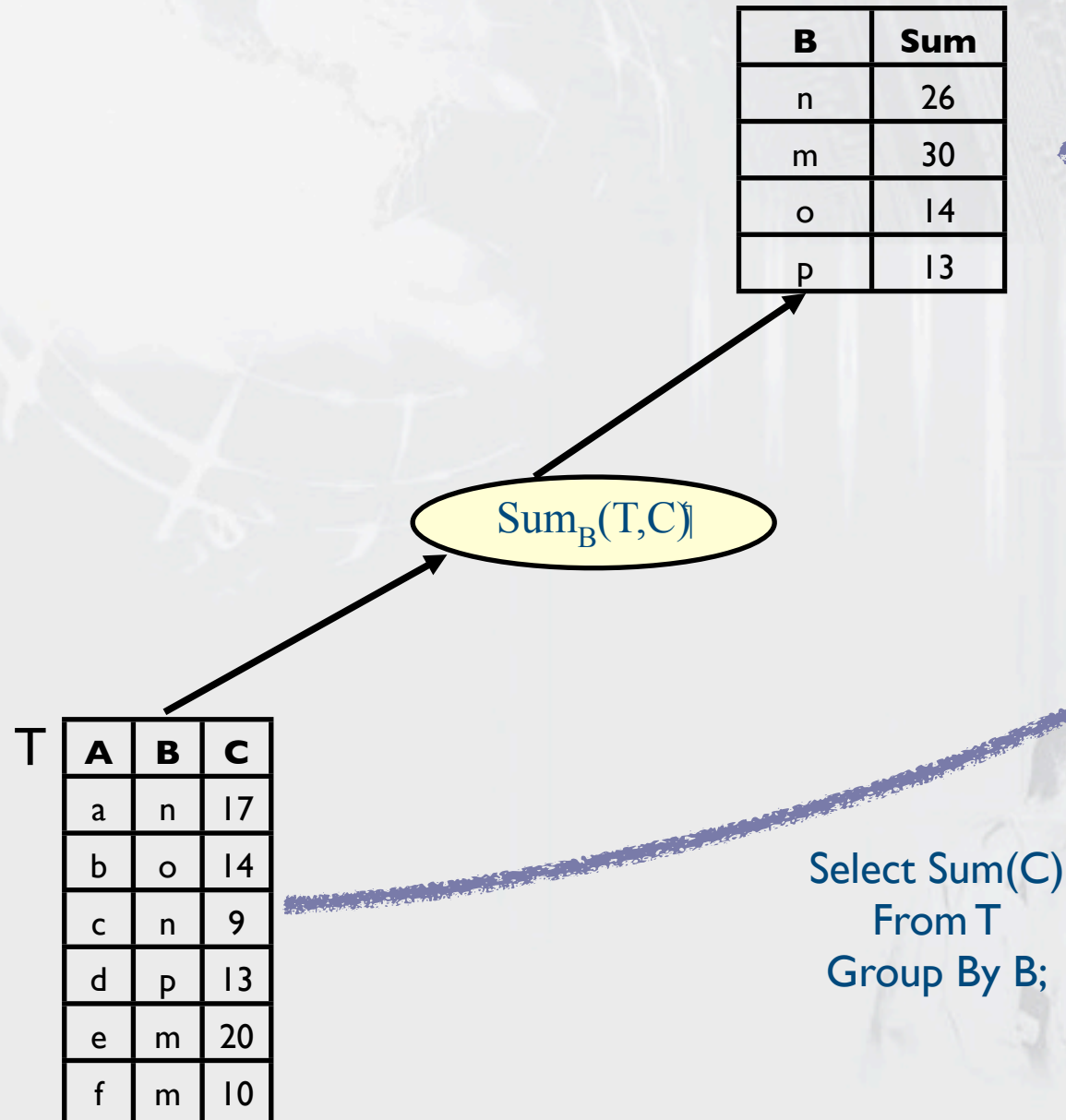
T

A	B	C
a	n	17
b	o	14
c	n	9
d	p	13
e	m	20
f	m	10

Exemples de sommes



Exemples de sommes



Expression

- Une **expression de l'algèbre relationnelle** est une expression composée de relations et d'opérateurs de l'algèbre relationnelle (de base ou non).
 - Une expression peut être représentée à l'aide d'un arbre syntaxique.
- Le résultat d'une expression est une relation sur laquelle on peut faire une autre opération.
- Remarque : en SQL, une table peut être une donnée de base ou le résultat d'une expression.

Exemples d'expressions

R_1

A	B	C
a	b	c
d	a	f
c	b	d
a	e	c

$\pi_A(\sigma_{C='d'}(\sigma_{B='b'}(R_1)))$

R_1

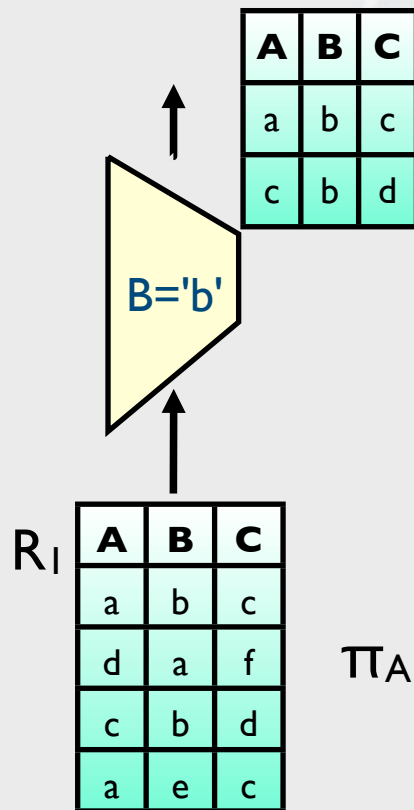
A	B	C
a	b	c
d	a	f
c	b	d
a	e	c

S_1

A	B	C
b	g	a
d	a	f

$\pi_{R.A, S.C}(\sigma_{S.B='g'}(R_1 \bowtie_{B=A} S_1))$

Exemples d'expressions



$\pi_A(\sigma_{C='d'}(\sigma_{B='b'}(R_1)))$

R_1

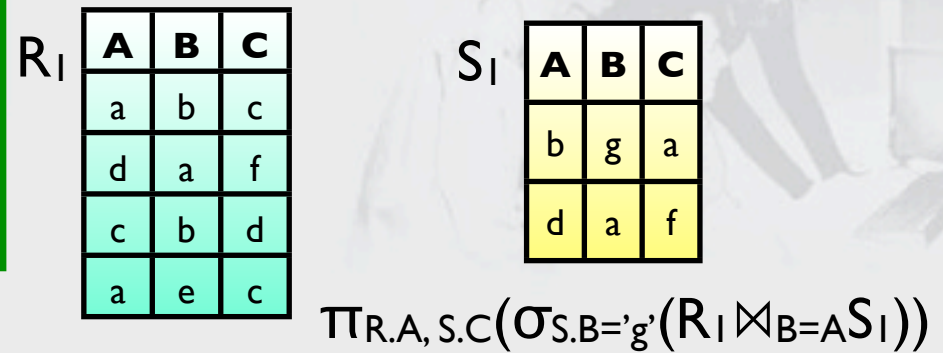
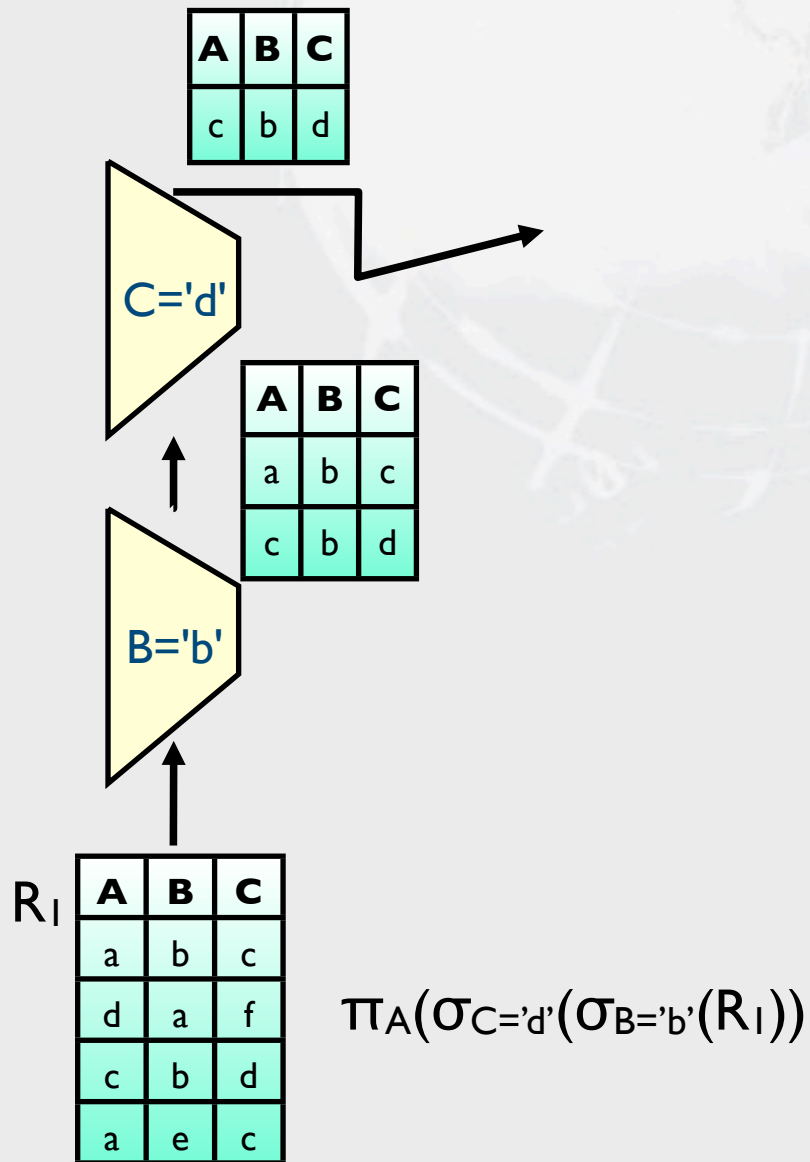
A	B	C
a	b	c
d	a	f
c	b	d
a	e	c

S_1

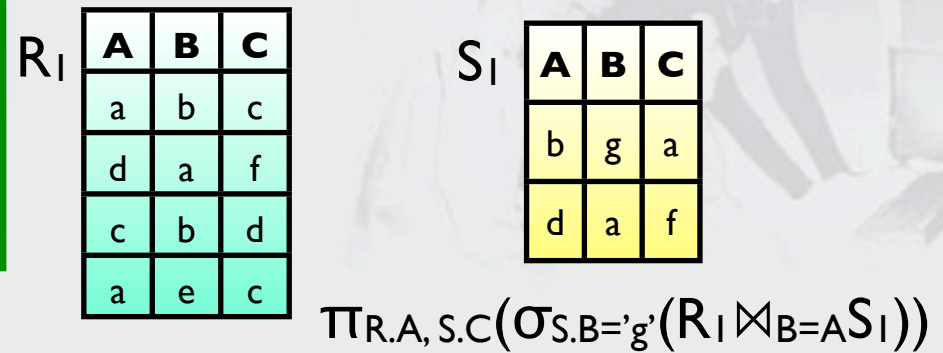
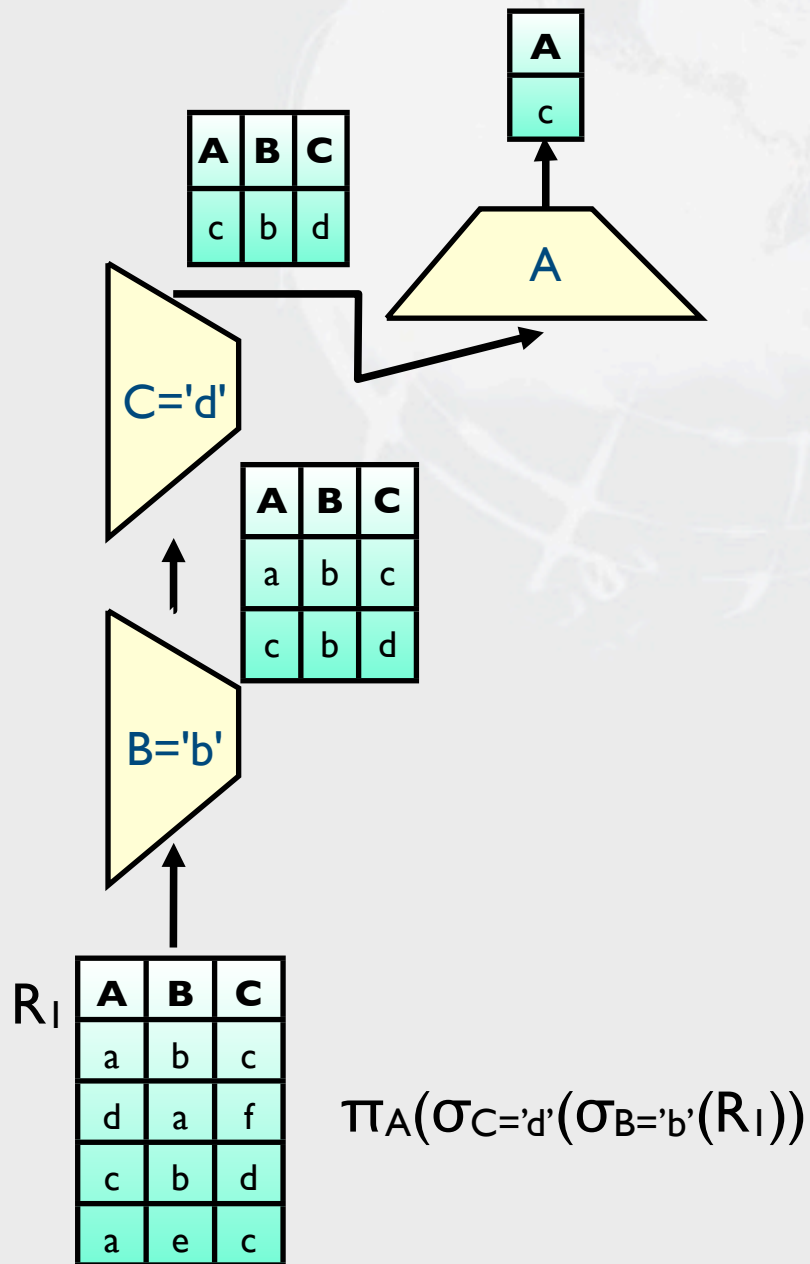
A	B	C
b	g	a
d	a	f

$\pi_{R.A, S.C}(\sigma_{S.B='g'}(R_1 \bowtie_{B=A} S_1))$

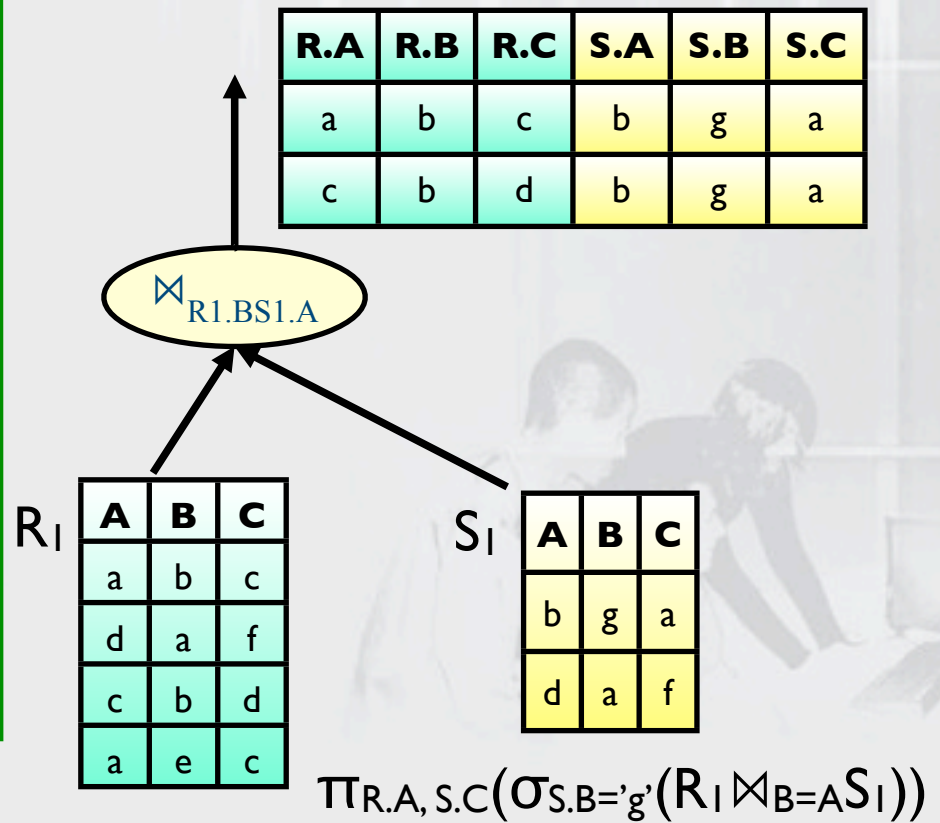
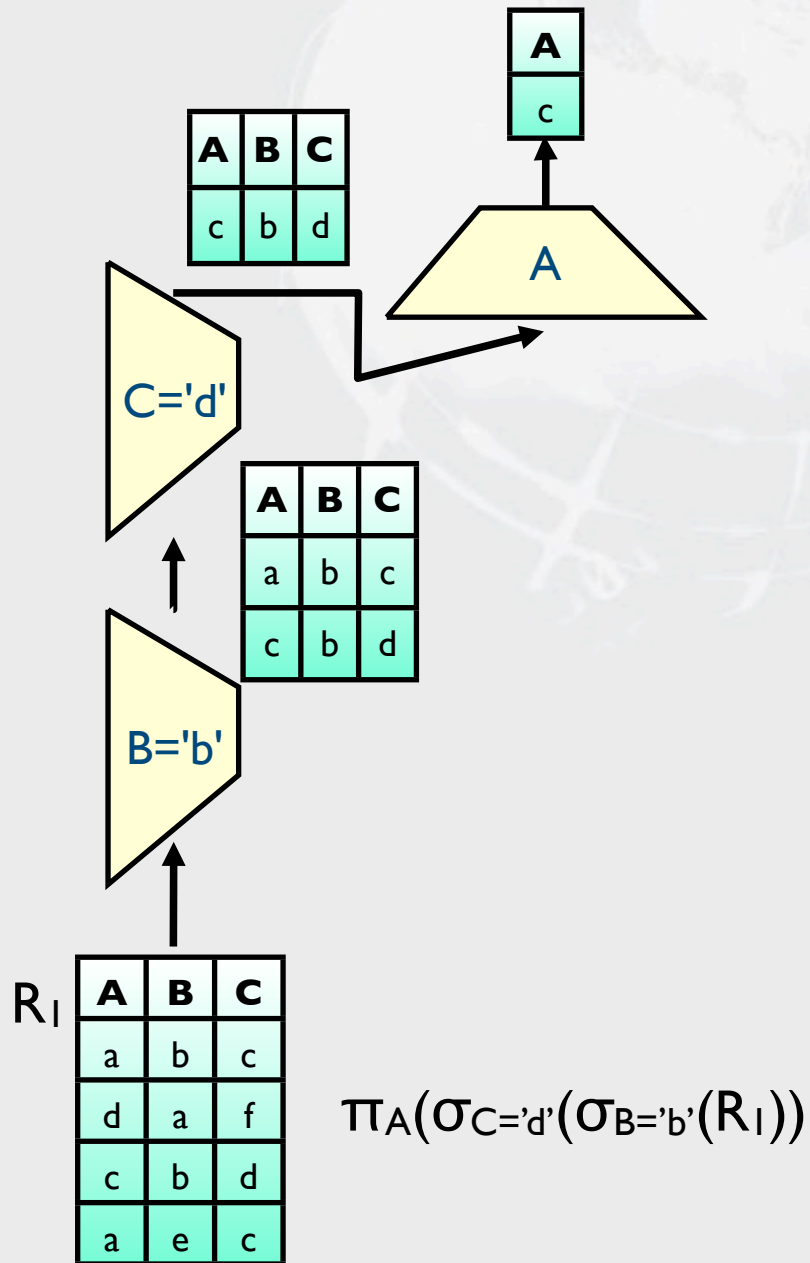
Exemples d'expressions



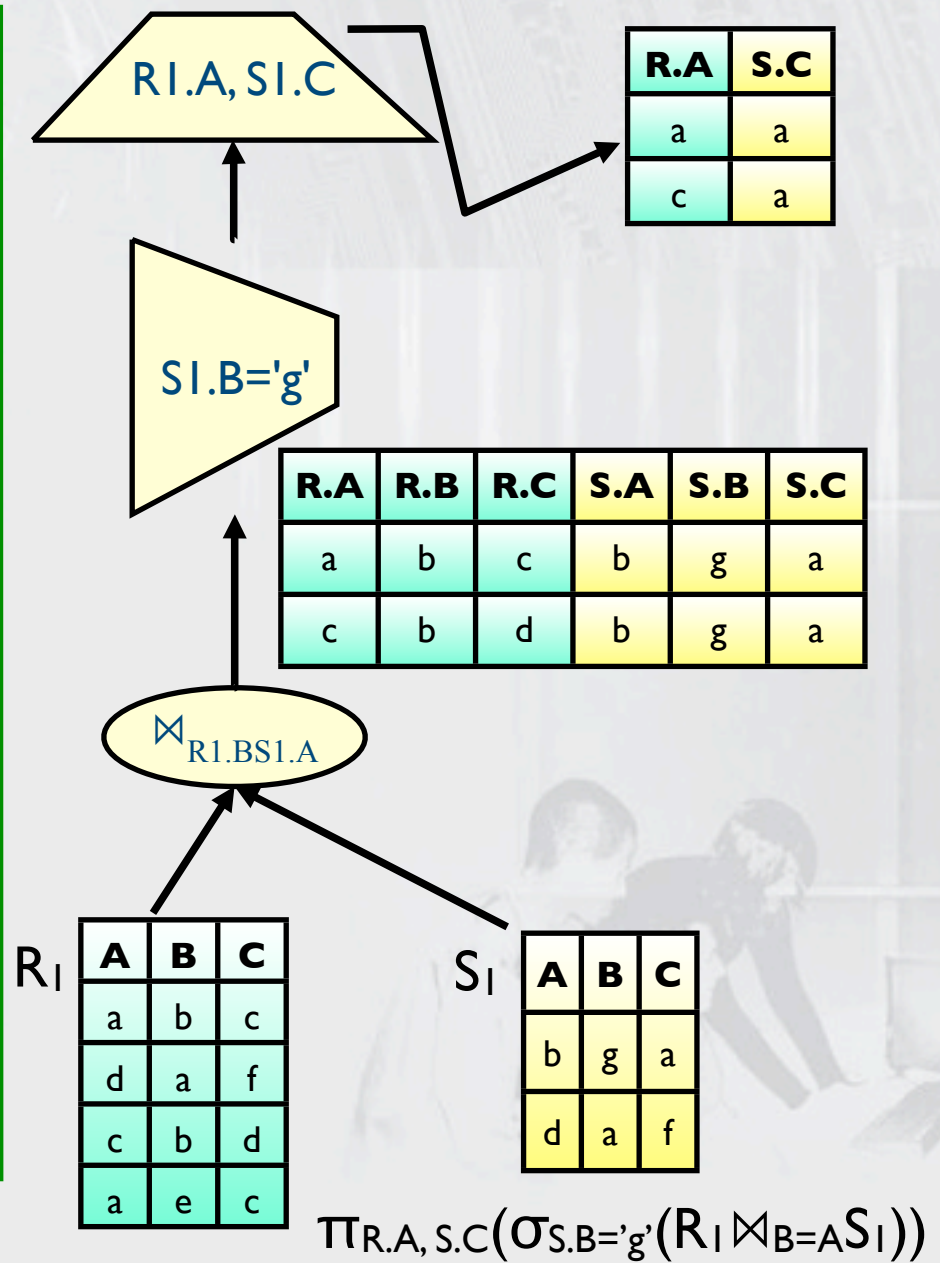
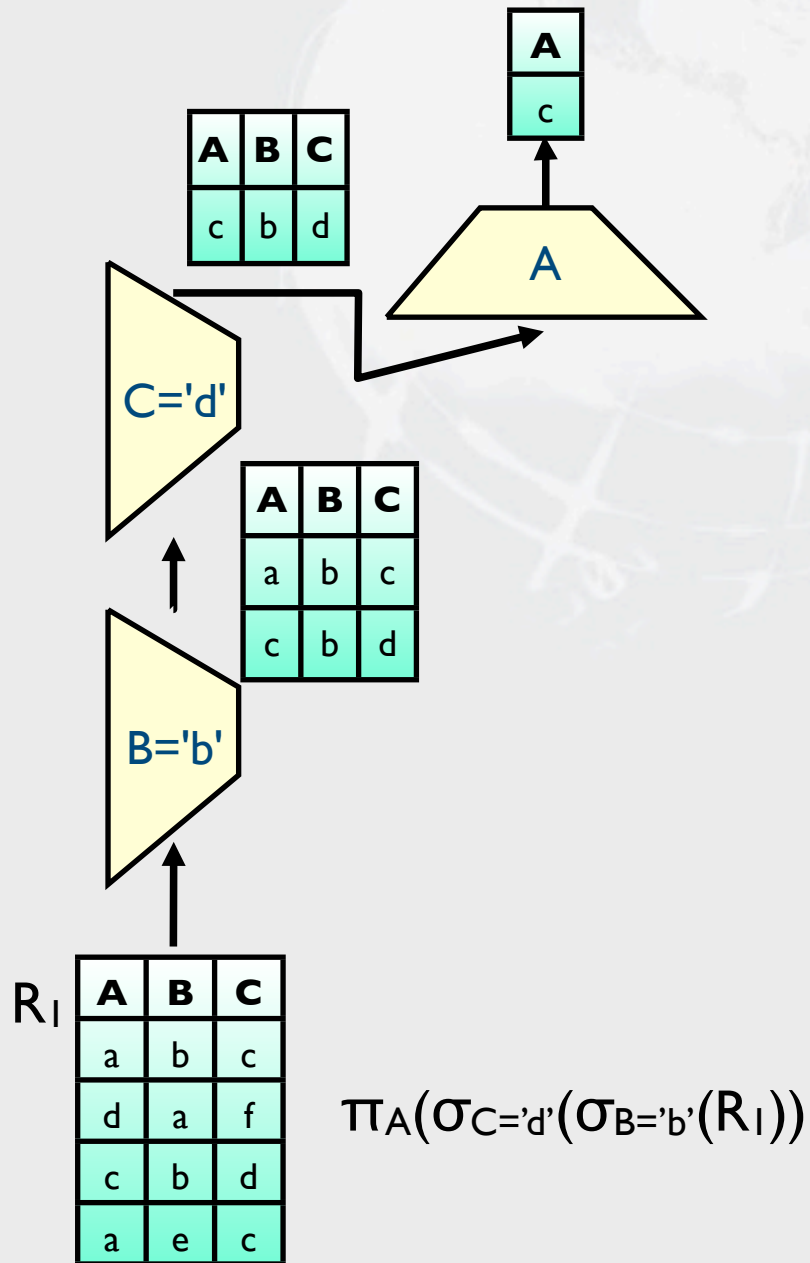
Exemples d'expressions



Exemples d'expressions



Exemples d'expressions



Propriétés

- θ -jointure, Union et Intersection sont commutatives
- θ -jointure, Union et Intersection sont associatives
- $\sigma_{Q_2}(\sigma_{Q_1}(R)) = \sigma_{Q_1 \wedge Q_2}(R)$
- $\sigma_{Q_1}(R) \cup \sigma_{Q_2}(R) = \sigma_{Q_1 \vee Q_2}(R)$
- $\Pi_{A_1, \dots, A_p}(\sigma_Q(R)) = \sigma_Q(\Pi_{A_1, \dots, A_p}(R))$
 $\iff \forall A_i \in Q, A_i \in \{A_1, \dots, A_p\}$
- Si $\exists A_j \in Q / A_j \notin \{A_1, \dots, A_p\}$, alors
 $\Pi_{A_1, \dots, A_p}(\sigma_Q(R)) = \Pi_{A_1, \dots, A_p}(\sigma_Q(\Pi_{A_1, \dots, A_p, A_j}(R)))$

Propriétés

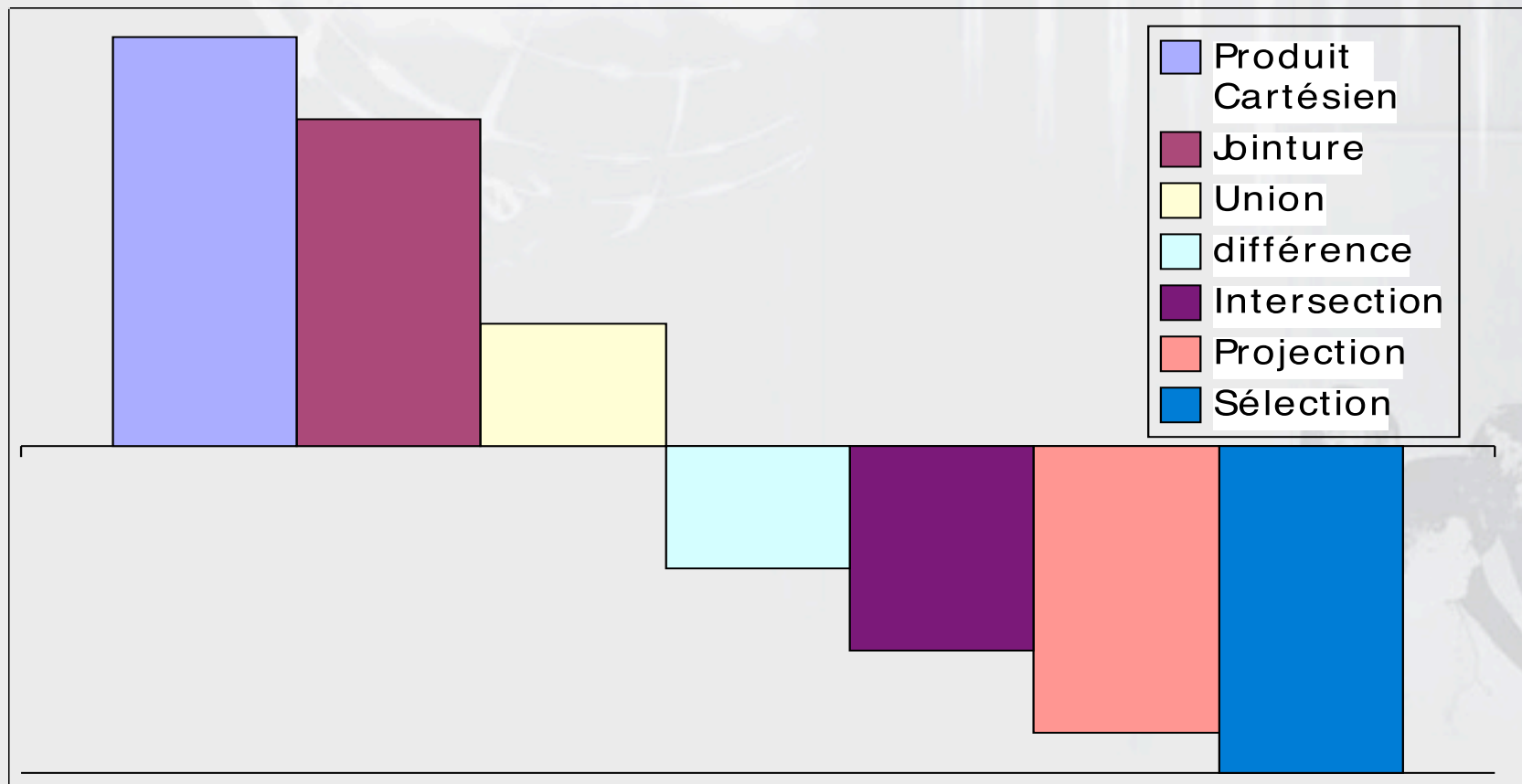
- $\sigma_{Q(A_i) \wedge Q'(B_j)}(R \bowtie_{Q''} S) = \sigma_{Q(A_i)}(R) \bowtie_{Q''} \sigma_{Q'(B_j)}(S)$
- $\sigma_Q(R \cup S) = \sigma_Q(R) \cup \sigma_Q(S)$
- $\Pi_V(R \bowtie_Q S) =$

$$\begin{cases} \Pi_{V(R)}(R) \bowtie_Q \Pi_{V(S)}(S) & \text{si } \forall X_j \in Q, X_j \in \{A_i\} \cup \{B_i\} \\ \Pi_V(\Pi_{V(R) \cup \{A_i\}}(R) \bowtie_Q \Pi_{V(S) \cup \{B_i\}}(S)) & \text{sinon} \end{cases}$$

$$\{A_i\} \cup \{B_j\} \subseteq Q$$
- $\Pi_V(R \cup S) = \Pi_{V(R)}(R) \cup \Pi_{V(S)}(S)$

Remarque

- Coûts relatifs des opérateurs en terme de quantité d'informations produites (cas général) □
 - Taille des tables produites



Remarque

- Coût des opérateurs parfois importants.
- Possibilité de permuter l'ordre des opérations (en fonction des propriétés).
 - Possibilité (nécessité) de réduire considérablement la taille des relations intermédiaires
 - moins de place
 - plus rapide
 - Il existe plusieurs solutions pour obtenir une relation : choisir la plus efficace.
- Piste : en général, cherche à appliquer les opérateurs de sélection et de projection le plus tôt possible.

Un petit exemple

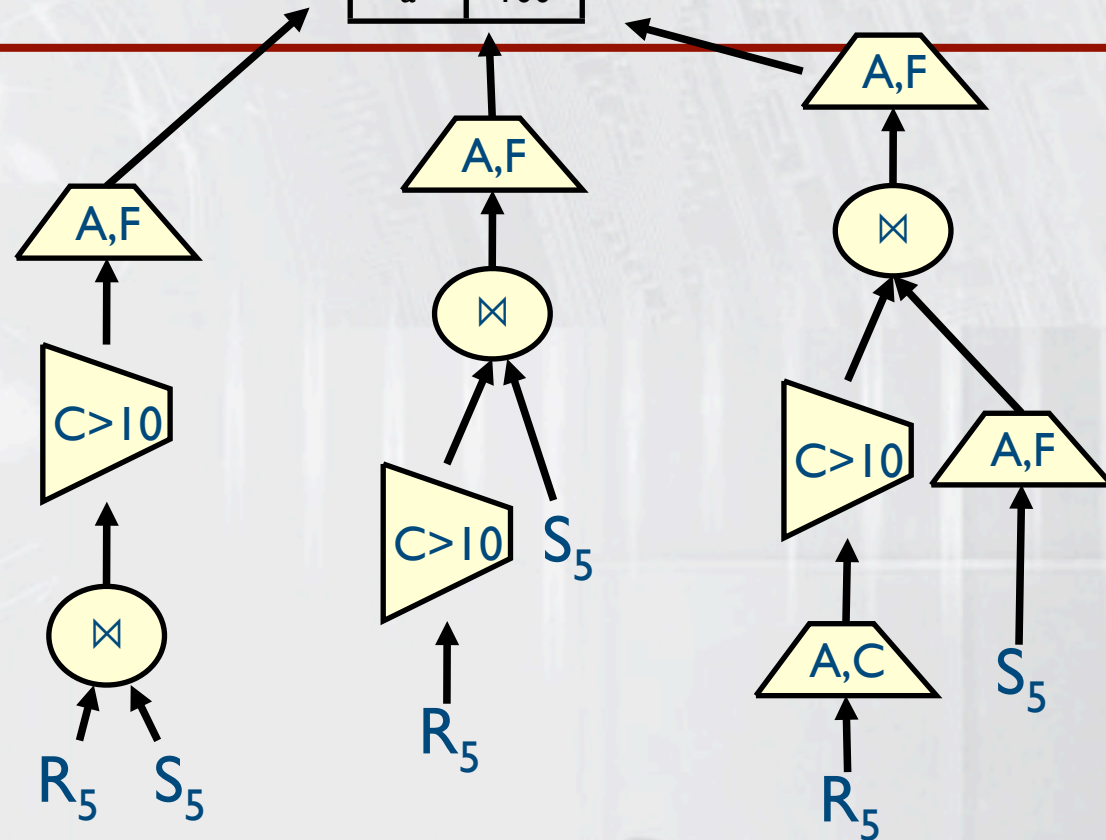
R_5

A	B	C
a	b	2
b	b	12
a	c	20
c	c	1
b	c	18

S_5

A	D	E	F
a	k	f	10
b	b	b	2
b	a	b	18

- (1) : $\pi_{A,F}(\sigma_{C>10}(R_5 \bowtie S_5))$
 (2) : $\pi_{A,F}(\sigma_{C>10}(R_5) \bowtie S_5)$
 (3) : $\pi_{A,F}(\sigma_{C>10}(\pi_{A,C}(R_5)) \bowtie \pi_{A,F}(S_5))$



(1) Taille

30		24	54
	6*7*2		84
	5*7*2		70
	5*2*2		20

Max

84

Gain 61,90 %

(2) Taille

30		24	54
3*3*2		24	42
	5*7*2		70
	20		20

70

(3) Taille

30		24	54
5*2*2		2*2*3	32
3*2*2		2*2*3	24
	5*3*2		30
	20		20

32(54)



Dépendances fonctionnelles



Dépendance fonctionnelle

- Possibilité de «calculer» les clés candidates, voire les clés minimales
 - besoin d'outils théoriques
- Soit une relation $R(X,Y,Z)$,
 X , Y et Z étant des ensembles d'attributs,
- **une dépendance** entre X et Y
signifie que **leurs valeurs sont liées**
 - Nature : fonctionnelle, multi-valuée, produit...

Dépendance fonctionnelle

R(X,Y,Z)

- Il existe une **dépendance fonctionnelle** entre X et Y si, à tout valeur de X, correspond une et une seule valeur de Y
 - le lien de dépendance entre X et Y est une fonction
 - X : la source de la df
 - Y : la cible de la df
 - notée : $X \rightarrow Y$ c'est-à-dire «X détermine Y»
 - $\forall t \in O_R, \forall t' \in O_R,$
 $\Pi_X(t) = \Pi_X(t') \Rightarrow \Pi_Y(t) = \Pi_Y(t')$

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

Dépendance fonctionnelle

R(X,Y,Z)

- Il existe une **dépendance fonctionnelle** entre X et Y si, à tout valeur de X, correspond une et une seule valeur de Y
 - le lien de dépendance entre X et Y est une fonction
 - X : la source de la df
 - Y : la cible de la df
 - notée : $X \rightarrow Y$ c'est-à-dire «X détermine Y»
 - $\forall t \in O_R, \forall t' \in O_R,$
 $\Pi_X(t) = \Pi_X(t') \Rightarrow \Pi_Y(t) = \Pi_Y(t')$

noetu → noetu

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

Dépendance fonctionnelle

R(X,Y,Z)

- Il existe une **dépendance fonctionnelle** entre X et Y si, à tout valeur de X, correspond une et une seule valeur de Y
 - le lien de dépendance entre X et Y est une fonction
 - X : la source de la df
 - Y : la cible de la df
 - notée : $X \rightarrow Y$ c'est-à-dire «X détermine Y»
 - $\forall t \in O_R, \forall t' \in O_R,$
 $\Pi_X(t) = \Pi_X(t') \Rightarrow \Pi_Y(t) = \Pi_Y(t')$

noetu $\rightarrow$ noetu
noetu $\rightarrow$ nom

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

Dépendance fonctionnelle

R(X,Y,Z)

- Il existe une **dépendance fonctionnelle** entre X et Y si, à tout valeur de X, correspond une et une seule valeur de Y
 - le lien de dépendance entre X et Y est une fonction
 - X : la source de la df
 - Y : la cible de la df
 - notée : $X \rightarrow Y$ c'est-à-dire «X détermine Y»
 - $\forall t \in O_R, \forall t' \in O_R,$
 $\Pi_X(t) = \Pi_X(t') \Rightarrow \Pi_Y(t) = \Pi_Y(t')$

noetu $\rightarrow$ noetu

noetu $\rightarrow$ nom

login, noetu $\rightarrow$ mdp, nom, prénom

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

Dépendance fonctionnelle

R(X,Y,Z)

- Il existe une **dépendance fonctionnelle** entre X et Y si, à tout valeur de X, correspond une et une seule valeur de Y
 - le lien de dépendance entre X et Y est une fonction
 - X : la source de la df
 - Y : la cible de la df
 - notée : $X \rightarrow Y$ c'est-à-dire «X détermine Y»
 - $\forall t \in O_R, \forall t' \in O_R,$
 $\Pi_X(t) = \Pi_X(t') \Rightarrow \Pi_Y(t) = \Pi_Y(t')$

noetu $\rightarrow$ noetu

noetu $\rightarrow$ nom

login, noetu $\rightarrow$ mdp, nom, prénom

login, nom $\rightarrow$ mdp, noetu, prénom

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

Dépendance fonctionnelle

R(X,Y,Z)

- Il existe une **dépendance fonctionnelle** entre X et Y si, à tout valeur de X, correspond une et une seule valeur de Y
- le lien de dépendance entre X et Y est une fonction
 - X : la source de la df
 - Y : la cible de la df
- notée : $X \rightarrow Y$ c'est-à-dire «X détermine Y»
- $\forall t \in O_R, \forall t' \in O_R,$
 $\Pi_X(t) = \Pi_X(t') \Rightarrow \Pi_Y(t) = \Pi_Y(t')$

noetu $\rightarrow$ noetu

noetu $\rightarrow$ nom

login, noetu $\rightarrow$ mdp, nom, prénom

login, nom $\rightarrow$ mdp, noetu, prénom

noetu, mdp $\rightarrow$ login, nom, prénom

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

Dépendance fonctionnelle

R(X,Y,Z)

- Il existe une **dépendance fonctionnelle** entre X et Y si, à tout valeur de X, correspond une et une seule valeur de Y
- le lien de dépendance entre X et Y est une fonction
 - X : la source de la df
 - Y : la cible de la df
- notée : $X \rightarrow Y$ c'est-à-dire «X détermine Y»
- $\forall t \in O_R, \forall t' \in O_R,$
 $\Pi_X(t) = \Pi_X(t') \Rightarrow \Pi_Y(t) = \Pi_Y(t')$

noetu $\rightarrow$ noetu

noetu $\rightarrow$ nom

login, noetu $\rightarrow$ mdp, nom, prénom

login, nom $\rightarrow$ mdp, noetu, prénom

noetu, mdp $\rightarrow$ login, nom, prénom

nom $\rightarrow$ prénom

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

Dépendance fonctionnelle

$R(X,Y,Z)$

- Il existe une **dépendance fonctionnelle** entre X et Y si, à tout valeur de X , correspond une et une seule valeur de Y
 - le lien de dépendance entre X et Y est une fonction
 - X : la source de la df
 - Y : la cible de la df
 - notée : $X \rightarrow Y$ c'est-à-dire « X détermine Y »
 - $\forall t \in O_R, \forall t' \in O_R,$
 $\Pi_X(t) = \Pi_X(t') \Rightarrow \Pi_Y(t) = \Pi_Y(t')$

noetu $\rightarrow$ noetu

noetu $\rightarrow$ nom

login, noetu $\rightarrow$ mdp, nom, prénom

login, nom $\rightarrow$ mdp, noetu, prénom

noetu, mdp $\rightarrow$ login, nom, prénom

nom $\leftrightarrow$ prénom

prénom $\leftrightarrow$ nom

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

Dépendance fonctionnelle

R(X,Y,Z)

- Il existe une **dépendance fonctionnelle** entre X et Y si, à tout valeur de X, correspond une et une seule valeur de Y
- le lien de dépendance entre X et Y est une fonction
 - X : la source de la df
 - Y : la cible de la df
- notée : $X \rightarrow Y$ c'est-à-dire «X détermine Y»
- $\forall t \in O_R, \forall t' \in O_R,$
 $\Pi_X(t) = \Pi_X(t') \Rightarrow \Pi_Y(t) = \Pi_Y(t')$

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

noetu $\rightarrow$ noetu

noetu $\rightarrow$ nom

login, noetu $\rightarrow$ mdp, nom, prénom

login, nom $\rightarrow$ mdp, noetu, prénom

noetu, mdp $\rightarrow$ login, nom, prénom

nom $\nrightarrow$ prénom

prénom $\nrightarrow$ nom

nom $\nrightarrow$ mdp (?)

Clé d'une relation (v3)

- **Clé d'une relation :**
X est clé de la relation $R(X,Y,Z)$
si et seulement si
 $X \rightarrow Y,Z$
- clé : partition de l'ensemble des attributs

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

Clé d'une relation (v3)

- **Clé d'une relation :**
 X est clé de la relation $R(X,Y,Z)$
 si et seulement si
 $X \rightarrow Y,Z$
- clé : partition de l'ensemble des attributs

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

noetu $\rightarrow$ login, mdp, nom, prénom

Clé d'une relation (v3)

- **Clé d'une relation :**
X est clé de la relation $R(X,Y,Z)$
si et seulement si
 $X \rightarrow Y,Z$
- clé : partition de l'ensemble des attributs

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

noetu $\rightarrow$ login, mdp, nom, prénom

login $\rightarrow$ noetu, mdp, nom, prénom

Graphe de dépendance

						Matières	codemat	titre	responsable	diplôme
Etudiants	noetu	nom	prénom	login	mdp		MIAS2I5	I5	E238	Deug MIAS
	28936E	Dupont	Franck	dupont-f	fhgtrzv		MIAS2I6	I6	E426	Deug MIAS
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez		IUP2MA	Automates	E238	Licence IUP-MIAGE
	86719E	Martin	Adrien	martin-a	rynjerz		LIL6	Systèmes	E236	Licence Informatique
	99628C	Robert	Adrien	robert-a	mpoutyt		IUP2IS	Systèmes	E526	Licence IUP-MIAGE
	99321C	Denou	Michelle	denou-m	zaazraz		MIAS2I3	Math-Info	E426	Deug MIAS
	99322C	Dupont	Isabelle	dupont-is	khjgktyd		LIL5	Algo	E426	Licence Informatique

Étudiants

Notes	noetu	codemat	noteex	notecc
	99628C	MIAS2I5	12	15,5
	46283B	MIAS2I6	8	11
	46283B	MIAS2I5	9,5	2
	86719E	IUP2MA	12	5,5
	99321C	LIL6	18	16,5
	28936E	MIAS2I5	13,5	13,5
	86719E	IUP2IS	8,5	10
	99628C	MIAS2I6	3	6
	99321C	LIL5	15	14,5
	99322C	MIAS2I5	12	15,5
	28936E	MIAS2I6	12	null

Matières

Notes

Graphe de dépendance

						Matières	codemat	titre	responsable	diplôme
Etudiants	noetu	nom	prénom	login	mdp		MIAS2I5	I5	E238	Deug MIAS
	28936E	Dupont	Franck	dupont-f	fhgtrzv		MIAS2I6	I6	E426	Deug MIAS
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez		IUP2MA	Automates	E238	Licence IUP-MIAGE
	86719E	Martin	Adrien	martin-a	rynjerz		LIL6	Systèmes	E236	Licence Informatique
	99628C	Robert	Adrien	robert-a	mpoutyt		IUP2IS	Systèmes	E526	Licence IUP-MIAGE
	99321C	Denou	Michelle	denou-m	zaazraz		MIAS2I3	Math-Info	E426	Deug MIAS
	99322C	Dupont	Isabelle	dupont-is	khjgktyd		LIL5	Algo	E426	Licence Informatique



Notes	noetu	codemat	noteex	notecc
	99628C	MIAS2I5	12	15,5
	46283B	MIAS2I6	8	11
	46283B	MIAS2I5	9,5	2
	86719E	IUP2MA	12	5,5
	99321C	LIL6	18	16,5
	28936E	MIAS2I5	13,5	13,5
	86719E	IUP2IS	8,5	10
	99628C	MIAS2I6	3	6
	99321C	LIL5	15	14,5
	99322C	MIAS2I5	12	15,5
	28936E	MIAS2I6	12	null

Notes

Matières

Graphe de dépendance

						Matières	codemat	titre	responsable	diplôme
Etudiants	noetu	nom	prénom	login	mdp		MIAS2I5	I5	E238	Deug MIAS
	28936E	Dupont	Franck	dupont-f	fhgtrzv		MIAS2I6	I6	E426	Deug MIAS
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez		IUP2MA	Automates	E238	Licence IUP-MIAGE
	86719E	Martin	Adrien	martin-a	rynjerz		LIL6	Systèmes	E236	Licence Informatique
	99628C	Robert	Adrien	robert-a	mpoutyt		IUP2IS	Systèmes	E526	Licence IUP-MIAGE
	99321C	Denou	Michelle	denou-m	zaazraz		MIAS2I3	Math-Info	E426	Deug MIAS
	99322C	Dupont	Isabelle	dupont-is	khjgktyd		LIL5	Algo	E426	Licence Informatique



Notes	noetu	codemat	noteex	notecc
	99628C	MIAS2I5	12	15,5
	46283B	MIAS2I6	8	11
	46283B	MIAS2I5	9,5	2
	86719E	IUP2MA	12	5,5
	99321C	LIL6	18	16,5
	28936E	MIAS2I5	13,5	13,5
	86719E	IUP2IS	8,5	10
	99628C	MIAS2I6	3	6
	99321C	LIL5	15	14,5
	99322C	MIAS2I5	12	15,5
	28936E	MIAS2I6	12	null

Notes

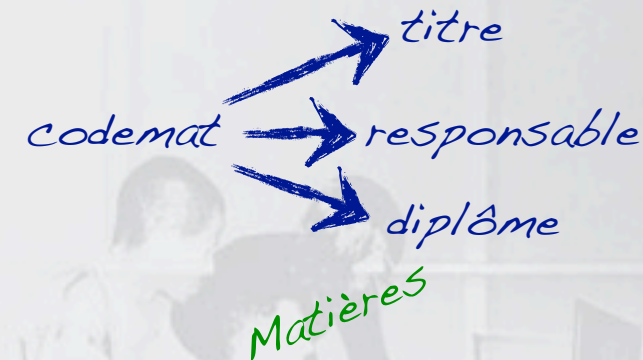
Matières

Graphe de dépendance

						Matières	codemat	titre	responsable	diplôme
Etudiants	noetu	nom	prénom	login	mdp		MIAS2I5	I5	E238	Deug MIAS
	28936E	Dupont	Franck	dupont-f	fhgtrzv		MIAS2I6	I6	E426	Deug MIAS
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez		IUP2MA	Automates	E238	Licence IUP-MIAGE
	86719E	Martin	Adrien	martin-a	rynjerz		LIL6	Systèmes	E236	Licence Informatique
	99628C	Robert	Adrien	robert-a	mpoutyt		IUP2IS	Systèmes	E526	Licence IUP-MIAGE
	99321C	Denou	Michelle	denou-m	zaazraz		MIAS2I3	Math-Info	E426	Deug MIAS
	99322C	Dupont	Isabelle	dupont-is	khjgktyd		LIL5	Algo	E426	Licence Informatique



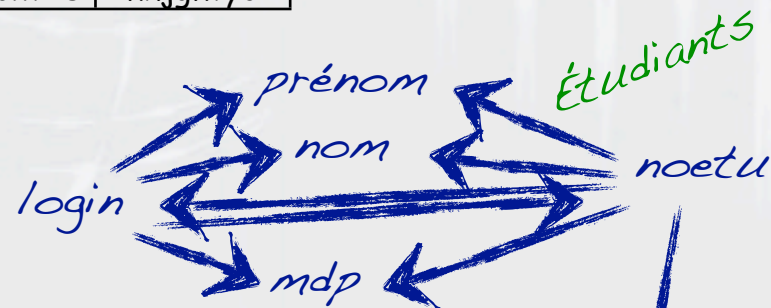
Notes	noetu	codemat	noteex	notecc
	99628C	MIAS2I5	12	15,5
	46283B	MIAS2I6	8	11
	46283B	MIAS2I5	9,5	2
	86719E	IUP2MA	12	5,5
	99321C	LIL6	18	16,5
	28936E	MIAS2I5	13,5	13,5
	86719E	IUP2IS	8,5	10
	99628C	MIAS2I6	3	6
	99321C	LIL5	15	14,5
	99322C	MIAS2I5	12	15,5
	28936E	MIAS2I6	12	null



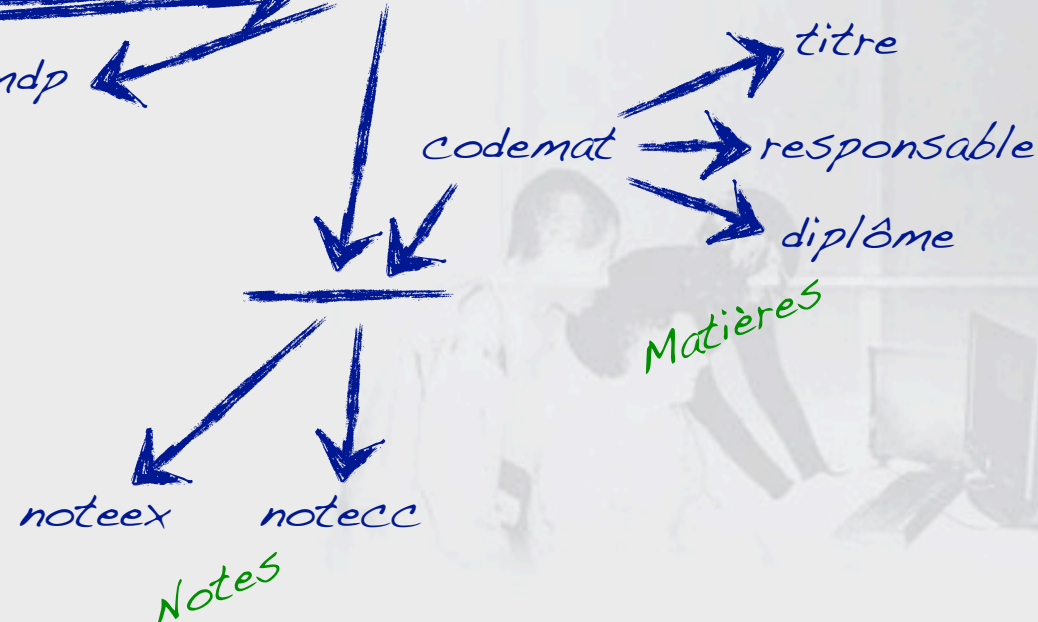
Notes

Graphe de dépendance

						Matières	codemat	titre	responsable	diplôme
Etudiants	noetu	nom	prénom	login	mdp		MIAS2I5	I5	E238	Deug MIAS
	28936E	Dupont	Franck	dupont-f	fhgtrzv		MIAS2I6	I6	E426	Deug MIAS
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez		IUP2MA	Automates	E238	Licence IUP-MIAGE
	86719E	Martin	Adrien	martin-a	rynjerz		LIL6	Systèmes	E236	Licence Informatique
	99628C	Robert	Adrien	robert-a	mpoutyt		IUP2IS	Systèmes	E526	Licence IUP-MIAGE
	99321C	Denou	Michelle	denou-m	zaazraz		MIAS2I3	Math-Info	E426	Deug MIAS
	99322C	Dupont	Isabelle	dupont-is	khjgktyd		LIL5	Algo	E426	Licence Informatique



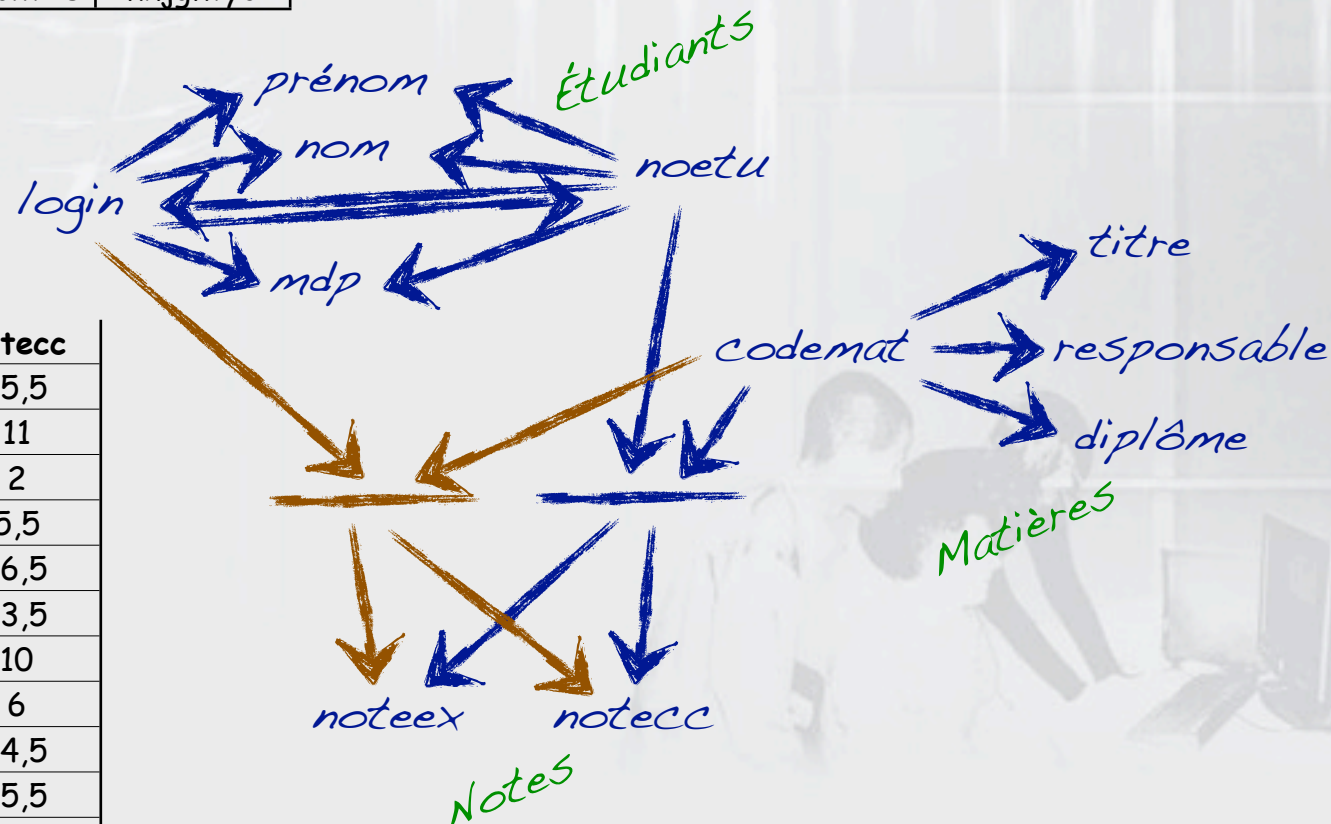
Notes	noetu	codemat	noteex	notecc
	99628C	MIAS2I5	12	15,5
	46283B	MIAS2I6	8	11
	46283B	MIAS2I5	9,5	2
	86719E	IUP2MA	12	5,5
	99321C	LIL6	18	16,5
	28936E	MIAS2I5	13,5	13,5
	86719E	IUP2IS	8,5	10
	99628C	MIAS2I6	3	6
	99321C	LIL5	15	14,5
	99322C	MIAS2I5	12	15,5
	28936E	MIAS2I6	12	null



Graphe de dépendance

						Matières	codemat	titre	responsable	diplôme
Etudiants										
noetu	nom	prénom	login	mdp						
28936E	Dupont	Franck	dupont-f	fhgtrzv		MIAS2I5	I5	E238		Deug MIAS
46283B	Dupont	Isabelle	dupont-i	ebnhertbez		MIAS2I6	I6	E426		Deug MIAS
86719E	Martin	Adrien	martin-a	rynjerz		IUP2MA	Automates	E238		Licence IUP-MIAGE
99628C	Robert	Adrien	robert-a	mpoutyt		LIL6	Systèmes	E236		Licence Informatique
99321C	Denou	Michelle	denou-m	zaazraz		IUP2IS	Systèmes	E526		Licence IUP-MIAGE
99322C	Dupont	Isabelle	dupont-is	khjgktyd		MIAS2I3	Math-Info	E426		Deug MIAS
						LIL5	Algo	E426		Licence Informatique

Notes				
noetu	codemat	noteex	notecc	
99628C	MIAS2I5	12	15,5	
46283B	MIAS2I6	8	11	
46283B	MIAS2I5	9,5	2	
86719E	IUP2MA	12	5,5	
99321C	LIL6	18	16,5	
28936E	MIAS2I5	13,5	13,5	
86719E	IUP2IS	8,5	10	
99628C	MIAS2I6	3	6	
99321C	LIL5	15	14,5	
99322C	MIAS2I5	12	15,5	
28936E	MIAS2I6	12	null	



Dépendance fonctionnelle

- **Schéma d'une relation** (ou schéma relationnel) : $\langle U, F \rangle$
 - U : ensemble des attributs
 - F : ensemble des dépendances fonctionnelles
- Attention, une DF est valable sur toutes les valeurs possibles de t-uples et pas seulement sur celles présentes à un moment donné.

Schéma de la relation

Étudiants :

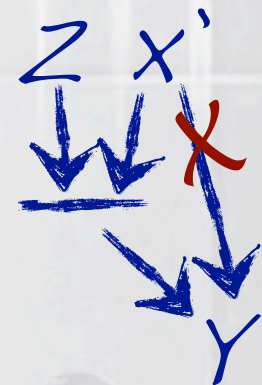
$\langle \{noetu, nom, prénom, mdp, login\}, \{noetu \rightarrow nom, prénom ; login \rightarrow mdp ; noetu \rightarrow login\} \rangle$

Dépendance fonctionnelle

- Soit $\langle U, F \rangle$ le schéma d'une relation R et $X \rightarrow Y$ ($X \subseteq U$ et $Y \subseteq U$) une dépendance fonctionnelle de F :
 X est en **dépendance fonctionnelle élémentaire** avec Y si $\nexists X' \subset X$ tq $X' \rightarrow Y \in F$
 - si X n'est formée que d'un seul attribut alors la dépendance fonctionnelle est nécessairement élémentaire
- X est en **dépendance fonctionnelle directe** avec Y si $\nexists Z \subseteq U$ ($Z \neq X$ et $Z \neq Y$) tq $X \rightarrow Z \wedge Z \rightarrow Y$

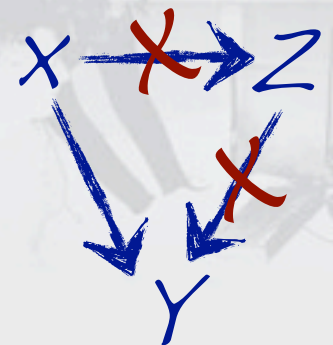
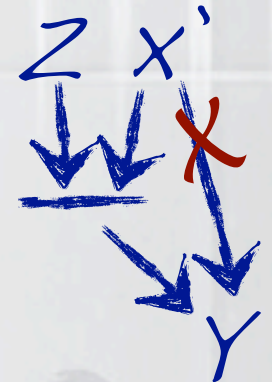
Dépendance fonctionnelle

- Soit $\langle U, F \rangle$ le schéma d'une relation R et $X \rightarrow Y$ ($X \subseteq U$ et $Y \subseteq U$) une dépendance fonctionnelle de F :
 X est en **dépendance fonctionnelle élémentaire** avec Y si $\nexists X' \subset X$ tq $X' \rightarrow Y \in F$
 - si X n'est formée que d'un seul attribut alors la dépendance fonctionnelle est nécessairement élémentaire
- X est en **dépendance fonctionnelle directe** avec Y si $\nexists Z \subseteq U$ ($Z \neq X$ et $Z \neq Y$) tq $X \rightarrow Z \wedge Z \rightarrow Y$



Dépendance fonctionnelle

- Soit $\langle U, F \rangle$ le schéma d'une relation R et $X \rightarrow Y$ ($X \subseteq U$ et $Y \subseteq U$) une dépendance fonctionnelle de F :
 X est en **dépendance fonctionnelle élémentaire** avec Y si $\nexists X' \subset X$ tq $X' \rightarrow Y \in F$
 - si X n'est formée que d'un seul attribut alors la dépendance fonctionnelle est nécessairement élémentaire
- X est en **dépendance fonctionnelle directe** avec Y si $\nexists Z \subseteq U$ ($Z \neq X$ et $Z \neq Y$) tq $X \rightarrow Z \wedge Z \rightarrow Y$



noetu, login $\rightarrow$ mdp, nom, prénom

Manipulation des dépendances fonctionnelles

- Soit $R \langle U, F \rangle$
 $W, X, Y, Z \subseteq U$

$$\begin{aligned} WX &\equiv W, X \equiv W \cup X \\ \text{donc } WX &= XW \\ \text{et } XX &= X \end{aligned}$$

- Règles de Armstrong (1)

- 3 axiomes

- Ar1 - **Réflexivité** : $Y \subseteq X \Rightarrow X \rightarrow Y$

- donc $X \rightarrow X$ (dépendance fonctionnelle triviale)

- Ar2 - **Augmentation** : $X \rightarrow Y, Z \subseteq W \Rightarrow WX \rightarrow ZY$

- donc $X \rightarrow Y \Rightarrow XW \rightarrow YW$

- Ar3 - **Transitivité** : $X \rightarrow Y \wedge Y \rightarrow Z \Rightarrow X \rightarrow Z$

Manipulation des dépendances fonctionnelles

- Exemple :
 - Soit $R\langle\{A,B,C,D,E\},$
 $\{A\rightarrow B ; BC\rightarrow D ; AC\rightarrow D ; D\rightarrow E ; AC\rightarrow E\}\rangle$
 - Montrer que $AD\rightarrow BE$!



Manipulation des dépendances fonctionnelles

- Exemple :
 - Soit $R\langle\{A,B,C,D,E\},$
 $\{A\rightarrow B ; BC\rightarrow D ; AC\rightarrow D ; D\rightarrow E ; AC\rightarrow E\}\rangle$
 - Montrer que $AD\rightarrow BE$!

1) $A\rightarrow B \Rightarrow AD\rightarrow BD$ (aug. avec D)



Manipulation des dépendances fonctionnelles

- Exemple :
 - Soit $R\langle\{A,B,C,D,E\},$
 $\{A\rightarrow B ; BC\rightarrow D ; AC\rightarrow D ; D\rightarrow E ; AC\rightarrow E\}\rangle$
 - Montrer que $AD\rightarrow BE$!

1) $A\rightarrow B \Rightarrow AD\rightarrow BD$ (aug. avec D)

2) $D\rightarrow E \Rightarrow BD\rightarrow BE$ (aug. avec B)

Manipulation des dépendances fonctionnelles

- Exemple :
 - Soit $R\langle\{A,B,C,D,E\},$
 $\{A\rightarrow B ; BC\rightarrow D ; AC\rightarrow D ; D\rightarrow E ; AC\rightarrow E\}\rangle$
 - Montrer que $AD\rightarrow BE$!

1) $A\rightarrow B \Rightarrow AD\rightarrow BD$ (aug. avec D)

2) $D\rightarrow E \Rightarrow BD\rightarrow BE$ (aug. avec B)

3) 1+2 $\Rightarrow AD\rightarrow BE$ (trans.)

Manipulation des dépendances fonctionnelles

- Exemple :
 - Soit $R\langle\{A,B,C,D,E\},$
 $\{A\rightarrow B ; BC\rightarrow D ; AC\rightarrow D ; D\rightarrow E ; AC\rightarrow E\}\rangle$
 - Montrer que $AD\rightarrow BE$!

1) $A\rightarrow B \Rightarrow AD\rightarrow BD$ (aug. avec D)

2) $D\rightarrow E \Rightarrow BD\rightarrow BE$ (aug. avec B)

3) 1+2 $\Rightarrow AD\rightarrow BE$ (trans.)

CQFD !

Manipulation des dépendances fonctionnelles

- Soit $R \langle U, F \rangle$
 $W, X, Y, Z \subseteq U$
- Règles de Armstrong (2)
 - 3 règles dérivées des axiomes
 - Ar4 - **Union** : $X \rightarrow Y \wedge X \rightarrow Z \Rightarrow X \rightarrow YZ$
 - Ar5 - **Décomposition** : $X \rightarrow YZ \Rightarrow X \rightarrow Y \wedge X \rightarrow Z$
 - Ar6 - **Pseudo-transitivité** : $X \rightarrow Y \wedge WY \rightarrow Z \Rightarrow WX \rightarrow Z$

Manipulation des dépendances fonctionnelles

- Soit $R \langle U, F \rangle$.

Union

$$1) X \rightarrow Y \Rightarrow X \rightarrow XY \text{ (aug. } X)$$

$$2) X \rightarrow Z \Rightarrow XY \rightarrow ZY \text{ (aug. } Y)$$

$$\text{Trans. } 1+2 \Rightarrow X \rightarrow YZ$$

CQFD !

(2)

3 règles dérivées des axiomes

- Ar4 - **Union** : $X \rightarrow Y \wedge X \rightarrow Z \Rightarrow X \rightarrow YZ$
- Ar5 - **Décomposition** : $X \rightarrow YZ \Rightarrow X \rightarrow Y \wedge X \rightarrow Z$
- Ar6 - **Pseudo-transitivité** : $X \rightarrow Y \wedge WY \rightarrow Z \Rightarrow WX \rightarrow Z$

Manipulation des dépendances fonctionnelles

- Soit D :

Décomposition

- 1) $YZ \rightarrow Z$ (réflexivité)
- 2) $YZ \rightarrow Y$ (réflexivité)
- 3) $X \rightarrow YZ$ (hypothèse)

Trans. 1+3 $\Rightarrow X \rightarrow Z$

Trans. 2+3 $\Rightarrow X \rightarrow Y$

CQFD !

et dans (2)

des axiomes

$$X \rightarrow Y \wedge X \rightarrow Z \Rightarrow X \rightarrow YZ$$

- Ar5 - **Décomposition** : $X \rightarrow YZ \Rightarrow X \rightarrow Y \wedge X \rightarrow Z$

- Ar6 - **Pseudo-transitivité** : $X \rightarrow Y \wedge WY \rightarrow Z \Rightarrow WX \rightarrow Z$

Manipulation des dépendances fonctionnelles

- Soit $R \langle U, F \rangle$
 $W, X, Y, Z \subseteq U$
- Règles de Armstrong (2)

Pseudo-transitivité
 1) $X \rightarrow Y \Rightarrow WX \rightarrow WY$ (aug. W)
 2) $WY \rightarrow Z$ (hypothèse)
 Trans. 1+2 $\Rightarrow WX \rightarrow Z$
 CQFD !

des axiomes

$$X \rightarrow Y \wedge X \rightarrow Z \Rightarrow X \rightarrow YZ$$

$$X \rightarrow YZ \Rightarrow X \rightarrow Y \wedge X \rightarrow Z$$

- Ar6 - **Pseudo-transitivité** : $X \rightarrow Y \wedge WY \rightarrow Z \Rightarrow WX \rightarrow Z$

Manipulation des dépendances fonctionnelles

- Tout ensemble de DF peut être représenté sous **forme canonique**
- la partie droite (cible) des DF n'est formée que d'un seul attribut (conséquence de l'union et la décomposition).

Fermeture (clôture)

Soit $R \langle U, F \rangle$

- La **fermeture** : l'ensemble noté F^+ de toutes les DF qui sont conséquences de F par l'application des règles d'Armstrong.
 - Si seulement la transitivité et la pseudo-transitivité sont utilisées : **fermeture transitive**.
 - NB : souvent lourd à calculer !
- Soient F et F' deux ensembles de DF
 $F \equiv F' \Leftrightarrow F^+ = F'^+$

$$\{A \rightarrow B ; B \rightarrow C ; B \rightarrow D\} \equiv \{A \rightarrow B ; A \rightarrow D ; B \rightarrow CD\}$$

Fermeture

Soit $R \langle U, F \rangle$

- $\forall F, \exists F'$ ensemble de DF élémentaires non triviales tq $F \equiv F'$.
- X **clé candidate** : si $X \rightarrow U \in F^+$.
- X **clé minimale** : si $X \rightarrow U \in F^+$,
et $\nexists Y \subset X : Y \rightarrow U \in F^+$.
- **fermeture (transitive) d'un ensemble d'attributs X , notée X^+** , est l'ensemble des attributs atteignables par F ($R \langle U, F \rangle, X \subset U$) en appliquant la transitivité ou la pseudo-transitivité.

Couverture minimale

Soit $R \langle U, F \rangle$

- Objectif : gérer les DF
 - simplifier, supprimer les redondances...
 $\Rightarrow$ obtenir un ensemble de DF équivalent plus pertinent.
- **Dépendance fonctionnelle redondante** :
 $f \in F$ redondante si $f \in (F - \{f\})^+$
- **Couverture irredondante** (ou **couverture minimale**),
 notée $I(F)$ ou I^F :
 - $I^F \subseteq F : I^F \equiv F$ (autrement dit $I^{F+} = F^+$)
 et $\nexists F' \subset I^F$ tq $F' \equiv F$
 - Aucune DF de I^F n'est redondante

Couverture minimale

Soit $R \langle U, F \rangle$

- Cette couverture irredondante n'est pas unique
- Algorithme de construction d'une I^F possible :
 - mettre F sous forme canonique,
 - c'est-à-dire «simplifier» les parties droites en appliquant la règle de décomposition (les DF d'origines sont supprimées)
 - supprimer les DF que l'on peut obtenir par calcul à partir des autres

Couverture minimale

- Exemple : soit $R\langle\{A,B,C,D,E\},F\rangle$,
 - avec $F = \{A \rightarrow B ; BC \rightarrow D ; D \rightarrow E ; AC \rightarrow DE\}$
 - donc $\text{Canonique}(F) = \{A \rightarrow B ; BC \rightarrow D ; D \rightarrow E ; AC \rightarrow D ; AC \rightarrow E\}$
 - $AC \rightarrow D$?
 - $A \rightarrow B \Rightarrow AC \rightarrow BC$ (aug. C)
 - $BC \rightarrow D \Rightarrow AC \rightarrow D$ (trans.)
 - CQFD !
 - $AC \rightarrow E$?
 - $AC \rightarrow D$ et $D \rightarrow E \Rightarrow AC \rightarrow E$ (trans.)
 - CQFD !
 - $I(F) = \{A \rightarrow B ; BC \rightarrow D ; D \rightarrow E\}$



Normalisation

- Complexité de construire des schémas «propres»
 - limiter la redondance
 - éviter les problèmes de reconstruction et de stockage (gestion des ajouts/modifications/suppressions)



Normalisation

- Complexité de construire des schémas «propres»
 - limiter la redondance
 - éviter les problèmes de reconstruction et de stockage (gestion des ajouts/modifications/suppressions)

Notes	noe	nom	prénom	login	mdp	codemat	titre	responsable	diplôme	noteex	notecc
1	99628C	Robert	Adrien	robert-a	mpoutyt	MIAS2I5	I5	E238	Deug MIAS	12	15,5
2	46283B	Dupont	Isabelle	dupont-i	ebnhertbez	MIAS2I6	I6	E426	Deug MIAS	8	11
3	46283B	Dupont	Isabelle	dupont-i	ebnhertbez	MIAS2I5	I5	E238	Deug MIAS	9,5	2
4	86719E	Martin	Adrien	martin-a	ryrjerz	IUP2MA	Automates	E238	Licence IUP-MIAGE	12	5,5
5	99321C	Denou	Michelle	denou-m	zaazraz	LIL6	Systèmes	E236	Licence Informatiqu	18	16,5
6	28936E	Dupont	Franck	dupont-f	fhgtrzv	MIAS2I5	I5	E238	Deug MIAS	13,5	13,5
7	86719E	Martin	Adrien	martin-a	ryrjerz	IUP2IS	Systèmes	E526	Licence IUP-MIAGE	8,5	10
8	99628C	Robert	Adrien	robert-a	mpoutyt	MIAS2I6	I6	E426	Deug MIAS	3	6
9	99321C	Denou	Michelle	denou-m	zaazraz	LIL5	Algo	E426	Licence Informatiqu	15	14,5
10	99322C	Dupont	Isabelle	dupont-is	khjgktyd	MIAS2I5	I5	E238	Deug MIAS	12	15,5
11	28936E	Dupont	Franck	dupont-f	fhgtrzv	MIAS2I6	I6	E426	Deug MIAS	12	null

Normalisation

- Complexité de construire des schémas «propres»
 - limiter la redondance
 - éviter les problèmes de reconstruction et de stockage (gestion des ajouts/modifications/suppressions)

Notes	noe	nom	prénom	login	mdp	codemat	titre	responsable	diplôme	noteex	notecc
1	99628C	Robert	Adrien	robert-a	mpoutyt	MIAS2I5	I5	E238	Deug MIAS	12	15,5
2	46283B	Dupont	Isabelle	dupont-i	ebnhertbez	MIAS2I6	I6	E426	Deug MIAS	8	11
3	46283B	Dupont	Isabelle	dupont-i	ebnhertbez	MIAS2I5	I5	E238	Deug MIAS	9,5	2
4	86719E	Martin	Adrien	martin-a	ryrjerz	IUP2MA	Automates	E238	Licence IUP-MIAGE	12	5,5
5	99321C	Denou	Michelle	denou-m	zaazraz	LIL6	Systèmes	E236	Licence Informatiqu	18	16,5
6	28936E	Dupont	Franck	dupont-f	fhgtrzv	MIAS2I5	I5	E238	Deug MIAS	13,5	13,5
7	86719E	Martin	Adrien	martin-a	ryrjerz	IUP2IS	Systèmes	E526	Licence IUP-MIAGE	8,5	10
8	99628C	Robert	Adrien	robert-a	mpoutyt	MIAS2I6	I6	E426	Deug MIAS	3	6
9	99321C	Denou	Michelle	denou-m	zaazraz	LIL5	Algo	E426	Licence Informatiqu	15	14,5
10	99322C	Dupont	Isabelle	dupont-is	khjgktyd	MIAS2I5	I5	E238	Deug MIAS	12	15,5
11	28936E	Dupont	Franck	dupont-f	fhgtrzv	MIAS2I6	I6	E426	Deug MIAS	12	null

beaucoup de redondance

Normalisation

- Complexité de construire des schémas «propres»
 - limiter la redondance
 - éviter les problèmes de reconstruction et de stockage (gestion des ajouts/modifications/suppressions)

Notes	noe	nom	prénom	login	mdp	codemat	titre	responsable	diplôme	noteex	notecc
1	99628C	Robert	Adrien	robert-a	mpoutyt	MIAS2I5	I5	E238	Deug MIAS	12	15,5
2	46283B	Dupont	Isabelle	dupont-i	ebnhertbez	MIAS2I6	I6	E426	Deug MIAS	8	11
3	46283B	Dupont	Isabelle	dupont-i	ebnhertbez	MIAS2I5	I5	E238	Deug MIAS	9,5	2
4	86719E	Martin	Adrien	martin-a	ryrjerz	IUP2MA	Automates	E238	Licence IUP-MIAGE	12	5,5
5	99321C	Denou	Michelle	denou-m	zaazraz	LIL6	Systèmes	E236	Licence Informatiqu	18	16,5
6	28936E	Dupont	Franck	dupont-f	fhgtrzv	MIAS2I5	I5	E238	Deug MIAS	13,5	13,5
7	86719E	Martin	Adrien	martin-a	ryrjerz	IUP2IS	Systèmes	E526	Licence IUP-MIAGE	8,5	10
8	99628C	Robert	Adrien	robert-a	mpoutyt	MIAS2I6	I6	E426	Deug MIAS	3	6
9	99321C	Denou	Michelle	denou-m	zaazraz	LIL5	Algo	E426	Licence Informatiqu	15	14,5
10	99322C	Dupont	Isabelle	dupont-is	khjgktyd	MIAS2I5	I5	E238	Deug MIAS	12	15,5
11	28936E	Dupont	Franck	dupont-f	fhgtrzv	MIAS2I6	I6	E426	Deug MIAS	12	null

*beaucoup de redondance
modifications très complexes*

Normalisation

- Complexité de construire des schémas «propres»
 - limiter la redondance
 - éviter les problèmes de reconstruction et de stockage (gestion des ajouts/modifications/suppressions)

Notes	noe	nom	prénom	login	mdp	codemat	titre	responsable	diplôme	noteex	notecc
1	99628C	Robert	Adrien	robert-a	mpoutyt	MIAS2I5	I5	E238	Deug MIAS	12	15,5
2	46283B	Dupont	Isabelle	dupont-i	ebnhertbez	MIAS2I6	I6	E426	Deug MIAS	8	11
3	46283B	Dupont	Isabelle	dupont-i	ebnhertbez	MIAS2I5	I5	E238	Deug MIAS	9,5	2
4	86719E	Martin	Adrien	martin-a	ryrjerz	IUP2MA	Automates	E238	Licence IUP-MIAGE	12	5,5
5	99321C	Denou	Michelle	denou-m	zaazraz	LIL6	Systèmes	E236	Licence Informatiqu	18	16,5
6	28936E	Dupont	Franck	dupont-f	fhgtrzv	MIAS2I5	I5	E238	Deug MIAS	13,5	13,5
7	86719E	Martin	Adrien	martin-a	ryrjerz	IUP2IS	Systèmes	E526	Licence IUP-MIAGE	8,5	10
8	99628C	Robert	Adrien	robert-a	mpoutyt	MIAS2I6	I6	E426	Deug MIAS	3	6
9	99321C	Denou	Michelle	denou-m	zaazraz	LIL5	Algo	E426	Licence Informatiqu	15	14,5
10	99322C	Dupont	Isabelle	dupont-is	khjgktyd	MIAS2I5	I5	E238	Deug MIAS	12	15,5
11	28936E	Dupont	Franck	dupont-f	fhgtrzv	MIAS2I6	I6	E426	Deug MIAS	12	null

beaucoup de redondance
modifications très complexes
existence dépendante des données (MIAS2I3 ?)

Normalisation

- Complexité de construire des schémas «propres»
 - limiter la redondance
 - éviter les problèmes de reconstruction et de stockage (gestion des ajouts/modifications/suppressions)

Notes	noe	nom	prénom	login	mdp	codemat	titre	responsable	diplôme	noteex	notecc
1	99628C	Robert	Adrien	robert-a	mpoutyt	MIAS2I5	I5	E238	Deug MIAS	12	15,5
2	46283B	Dupont	Isabelle	dupont-i	ebnhertbez	MIAS2I6	I6	E426	Deug MIAS	8	11
3	46283B	Dupont	Isabelle	dupont-i	ebnhertbez	MIAS2I5	I5	E238	Deug MIAS	9,5	2
4	86719E	Martin	Adrien	martin-a	ryrjerz	IUP2MA	Automates	E238	Licence IUP-MIAGE	12	5,5
5	99321C	Denou	Michelle	denou-m	zaazraz	LIL6	Systèmes	E236	Licence Informatiqu	18	16,5
6	28936E	Dupont	Franck	dupont-f	fhgtrzv	MIAS2I5	I5	E238	Deug MIAS	13,5	13,5
7	86719E	Martin	Adrien	martin-a	ryrjerz	IUP2IS	Systèmes	E526	Licence IUP-MIAGE	8,5	10
8	99628C	Robert	Adrien	robert-a	mpoutyt	MIAS2I6	I6	E426	Deug MIAS	3	6
9	99321C	Denou	Michelle	denou-m	zaazraz	LIL5	Algo	E426	Licence Informatiqu	15	14,5
10	99322C	Dupont	Isabelle	dupont-is	khjgktyd	MIAS2I5	I5	E238	Deug MIAS	12	15,5
11	28936E	Dupont	Franck	dupont-f	fhgtrzv	MIAS2I6	I6	E426	Deug MIAS	12	null

beaucoup de redondance
 modifications très complexes
 existence dépendante des données (MIAS2I3 ?)
 IHM ?

Normalisation

Décomposition

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

Matières	codemat	titre	responsable	diplôme
	MIAS2I5	I5	E238	Deug MIAS
	MIAS2I6	I6	E426	Deug MIAS
	IUP2MA	Automates	E238	Licence IUP-MIAGE
	LIL6	Systèmes	E236	Licence Informatique
	IUP2IS	Systèmes	E526	Licence IUP-MIAGE
	MIAS2I3	Math-Info	E426	Deug MIAS
	LIL5	Algo	E426	Licence Informatique

Notes	noetu	codemat	noteex	notecc
	99628C	MIAS2I5	12	15,5
	46283B	MIAS2I6	8	11
	46283B	MIAS2I5	9,5	2
	86719E	IUP2MA	12	5,5
	99321C	LIL6	18	16,5
	28936E	MIAS2I5	13,5	13,5
	86719E	IUP2IS	8,5	10
	99628C	MIAS2I6	3	6
	99321C	LIL5	15	14,5
	99322C	MIAS2I5	12	15,5
	28936E	MIAS2I6	12	null

Normalisation

Notes	noe	nom	prénom	login	mdp	codemat	titre	responsable	diplôme	noteex	notecc
1	99628C	Robert	Adrien	robert-a	mpoutyt	MIAS2I5	I5	E238	Deug MIAS	12	15,5
2	46283B	Dupont	Isabelle	dupont-i	ebnhertbez	MIAS2I6	I6	E426	Deug MIAS	8	11
3	46283B	Dupont	Isabelle	dupont-i	ebnhertbez	MIAS2I5	I5	E238	Deug MIAS	9,5	2
4	86719E	Martin	Adrien	martin-a	ryrjerz	IUP2MA	Automates	E238	Licence IUP-MIAGE	12	5,5
5	99321C	Denou	Michelle	denou-m	zaazraz	LIL6	Systèmes	E236	Licence Informatique	18	16,5
6	28936E	Dupont	Franck	dupont-f	fhgtrzv	MIAS2I5	I5	E238	Deug MIAS	13,5	13,5
7	86719E	Martin	Adrien	martin-a	ryrjerz	IUP2IS	Systèmes	E526	Licence IUP-MIAGE	8,5	10
8	99628C	Robert	Adrien	robert-a	mpoutyt	MIAS2I6	I6	E426	Deug MIAS	3	6
9	99321C	Denou	Michelle	denou-m	zaazraz	LIL5	Algo	E426	Licence Informatique	15	14,5
10	99322C	Dupont	Isabelle	dupont-is	khjgktyd	MIAS2I5	I5	E238	Deug MIAS	12	15,5
11	28936E	Dupont	Franck	dupont-f	fhgtrzv	MIAS2I6	I6	E426	Deug MIAS	12	null

Décomposition

Etudiants	noetu	nom	prénom	login	mdp
	28936E	Dupont	Franck	dupont-f	fhgtrzv
	46283B	Dupont	Isabelle	dupont-i	ebnhertbez
	86719E	Martin	Adrien	martin-a	ryrjerz
	99628C	Robert	Adrien	robert-a	mpoutyt
	99321C	Denou	Michelle	denou-m	zaazraz
	99322C	Dupont	Isabelle	dupont-is	khjgktyd

Matières	codemat	titre	responsable	diplôme
	MIAS2I5	I5	E238	Deug MIAS
	MIAS2I6	I6	E426	Deug MIAS
	IUP2MA	Automates	E238	Licence IUP-MIAGE
	LIL6	Systèmes	E236	Licence Informatique
	IUP2IS	Systèmes	E526	Licence IUP-MIAGE
	MIAS2I3	Math-Info	E426	Deug MIAS
	LIL5	Algo	E426	Licence Informatique

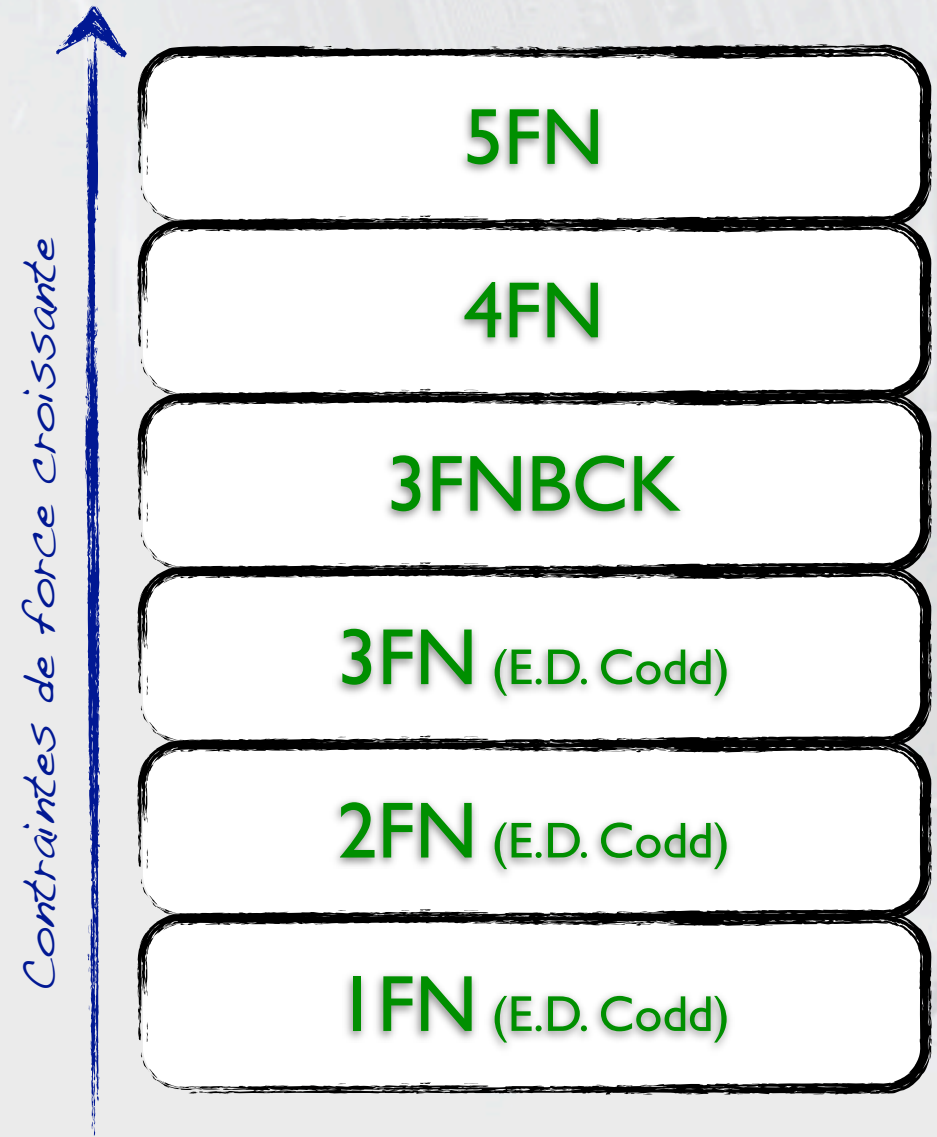
Notes	noetu	codemat	noteex	notecc
	99628C	MIAS2I5	12	15,5
	46283B	MIAS2I6	8	11
	46283B	MIAS2I5	9,5	2
	86719E	IUP2MA	12	5,5
	99321C	LIL6	18	16,5
	28936E	MIAS2I5	13,5	13,5
	86719E	IUP2IS	8,5	10
	99628C	MIAS2I6	3	6
	99321C	LIL5	15	14,5
	99322C	MIAS2I5	12	15,5
	28936E	MIAS2I6	12	null

Normalisation

- **Analyser les dépendances** entre les attributs qui sont à l'origine des phénomènes redondances ;
- Poser des méthodes systématiques pour **décomposer une relation** en plusieurs relations plus adaptées pour la gestion du problème
→ La relation d'origine doit pouvoir être retrouvée par jointures
- NB : bien de normaliser, mais il est parfois préférable de ne pas le faire pour des raisons de performance (efficacité) !

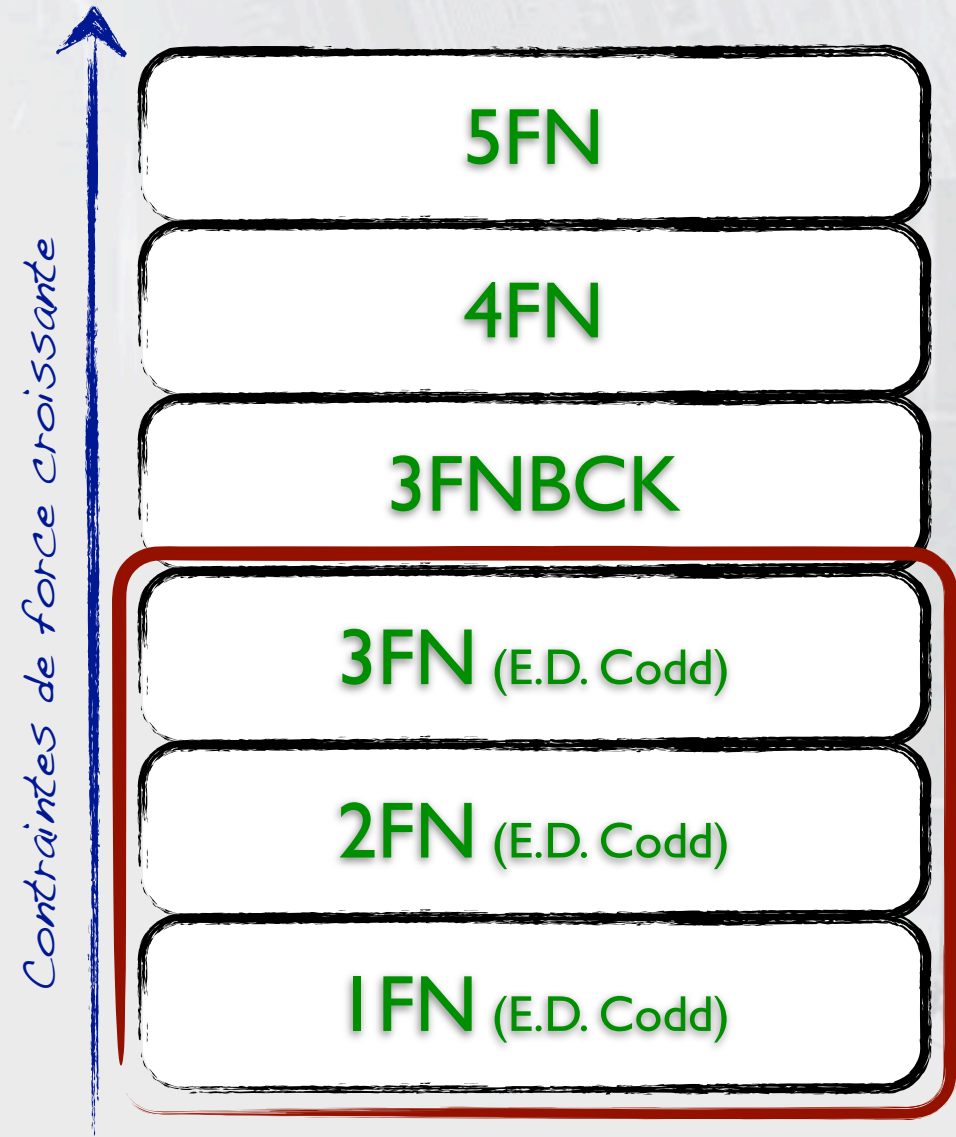
Formes normales

- Concept introduit par E.D. Codd
- Formes normales : classification des relations selon 3 catégories
- actuellement, jusqu'à 6 niveaux



Formes normales

- Concept introduit par E.D. Codd
- Formes normales : classification des relations selon 3 catégories
- actuellement, jusqu'à 6 niveaux



Formes normales

- 1FN : toute relation dont les attributs sont élémentaires (atomique)
 - Les attributs ne peuvent pas être décomposés du point de vue du contexte dans lequel est envisagé la relation.

notes	matiere	etudiant	notes
I5		Albert Paul	8, 12.5
I5		Duplo Bertrand	2.5, 0, 18

Formes normales

- 1FN : toute relation dont les attributs sont élémentaires (atomique)
 - Les attributs ne peuvent pas être décomposés du point de vue du contexte dans lequel est envisagé la relation.

notes	matiere	etudiant	notes
I5		Albert Paul	8, 12.5
I5		Duplo Bertrand	2.5, 0, 18



*notes(matiere, nom,
prenom, note1, note2, note3)*

Formes normales

- 2FN :
 - 1FN et tout attribut n'appartenant pas à une clé soit en DF élémentaire avec toutes les clés.
 - possibilité de résoudre le problème : découpage de la relation
 - $R \langle \{W, \underline{X}, \underline{Y}, Z\}, \{XY \rightarrow Z ; Y \rightarrow Z ; XY \rightarrow W\} \rangle$
 - $R' \langle \{W, \underline{X}, \# \underline{Y}\}, \{XY \rightarrow W\} \rangle$ et $R'' \langle \{\underline{Y}, Z\}, \{Y \rightarrow Z\} \rangle$
 - avec $R = R' \bowtie R''$
 ($\bowtie$: jointure = produit cartésien et sélection)

Formes normales

- 2FN :
 - exemple : $R \langle \{\underline{A}, \underline{B}, C\}, \{AB \rightarrow C ; B \rightarrow C\} \rangle$
pas en 2FN !
 - Pour passer en 2FN, il suffit d'appliquer une décomposition de la relation basée sur « $B \rightarrow C$ »
 - D'où : $R' \langle \{\underline{A}, \# \underline{B}\}, \{\} \rangle$ et $R'' \langle \{\underline{B}, C\}, \{B \rightarrow C\} \rangle$ avec perte de « $AB \rightarrow C$ »

Formes normales

- 3FN :
 - 2FN et tout attribut n'appartenant pas à une clé ne dépend pas d'un attribut non clé.
 - Toutes les DF sont directes.
 - Tout attribut n'appartenant pas à une clé est en DF élémentaire directe avec la clé.
 - Dès que deux attributs sont en DF et qu'ils ne font pas partie d'une clé alors ils ne sont pas en 3FN.
 - possibilité de résoudre le problème : découpage de la relation
 - $R \langle \{W, \underline{X}, \underline{Y}, Z\}, \{XY \rightarrow Z ; W \rightarrow Z ; XY \rightarrow W\} \rangle$
 - $R' \langle \{\#W, \underline{X}, \underline{Y}\}, \{XY \rightarrow W\} \rangle$ et $R'' \langle \{\underline{W}, Z\}, \{W \rightarrow Z\} \rangle$
 - avec $R = R' \bowtie R''$
 ($\bowtie$: jointure = produit cartésien et sélection)

Formes normales

- 3FN :
 - exemple : $R \langle \{\underline{A}, B, C\}, \{A \rightarrow BC ; B \rightarrow C\} \rangle$
n'est pas en 3FN !
 - Pour passer en 3FN, il suffit de décomposer la relation avec $B \rightarrow C$
 - D'où : $R' \langle \{\underline{A}, \#B\}, \{A \rightarrow B\} \rangle$ et $R'' \langle \{\underline{B}, C\}, \{B \rightarrow C\} \rangle$
 - Toute relation admet une décomposition en 3FN à jonction conservatrice (sans perte) et avec préservation des DF.

Formes normales

- Soit $R\langle U, F \rangle$ une relation
- Méthode pour déterminer la forme normale
 1. calculer $I(F)$
 2. déterminer les clés candidates
 3. choisir X , la clé de la relation
 4. calculer $F' = I(F) \cup \{X \rightarrow y : y \in U - X\}$
 5. Analyser les DF de F' au regard des contraintes indiquées en fonction de la clé choisie
 - Il existe des DF non-élémentaires : R est en 1FN
 - Il existe des DF non directes : R est en 2FN
 - toutes les DF sont élémentaires et directes : R est en 3FN

Dépendance fonctionnelle multi-valuée

- Généralisation de la notion de DF : **dépendances multi-valuées**,
 - notée $X \twoheadrightarrow Y$.
 - A chaque valeur de X n'est pas associée une valeur de Y mais un ensemble de valeurs...
 - généralisation des dépendances fonctionnelles
- Exemple :
 - pour une matière, les enseignants qui y participent
 - $\text{codemat} \twoheadrightarrow \text{no_enseignant}$

Dépendance fonctionnelle multi-valuée

- Règles
 - Ar7 - Complémentation : $X \twoheadrightarrow Y \Rightarrow X \twoheadrightarrow U-(XY)$
 - Ar8 - Réflexivité 2 : $Y \subseteq X \Rightarrow X \twoheadrightarrow Y$
 - Ar9 - Augmentation 2 : $X \twoheadrightarrow Y, Z \subseteq W \Rightarrow WX \twoheadrightarrow ZY$
 - Ar10 - Transitivité 2 : $X \twoheadrightarrow Y \wedge Y \twoheadrightarrow Z \Rightarrow X \twoheadrightarrow Z$
 - Ar11 - Pseudo-transitivité 2 : $X \twoheadrightarrow Y \wedge WY \twoheadrightarrow Z \Rightarrow WX \twoheadrightarrow Z-(YW)$
 - Ar12 - Union 2 : $X \twoheadrightarrow Y \wedge X \twoheadrightarrow Z \Rightarrow X \twoheadrightarrow YZ$
 - Ar 13 - Décomposition :
 $X \twoheadrightarrow YZ \Rightarrow X \twoheadrightarrow Y \cap Z \wedge X \twoheadrightarrow Z-Y \wedge X \twoheadrightarrow Y-Z$

Dépendance fonctionnelle multi-valuée

- Règles

- Ar14 : $X \rightarrow Y \Rightarrow X \twoheadrightarrow Y$

- Ar15 : $X \twoheadrightarrow Y \wedge Y \rightarrow Z', Z' \subseteq E \text{ et } Y \cap Z = \emptyset$
 $\Rightarrow X \twoheadrightarrow Z'$

- Ar16 : $X \twoheadrightarrow Y \wedge XY \rightarrow Z \Rightarrow X \twoheadrightarrow Z - Y$



Décomposition

- Objectif : normaliser une relation qui n'est pas en 3FN
- Procédé : la «découper» en plusieurs relations qui seront en 3FN
- Moyen : deux algorithmes
 - synthèse
 - prend en compte uniquement les DF
 - obtention de relations en 3FN (parfois 3FNBCK)
 - préservation des DF, mais pas toujours du contenu !
 - décomposition
 - prend en compte DF, dépendances multi-valuées...
 - obtention de relations en 3FNBCK et 4FN
 - préservation du contenu, mais pas toujours des dépendances

Algorithme de synthèse

Algorithme de Bernstein

- Soit $R \langle U, F \rangle$
 1. calculer $I(F)$.
 2. partitionner $I(F)$ en mettant ensemble les DF ayant la même source (partie gauche) : $I_1, I_2, \dots, I_n$.
 3. pour chaque I_k , construire $R_k \langle U_k, I_k \rangle$, telle que U_k contient les attributs présents dans I_k . Les I_k sont retirés de $I(F)$.
 4. les attributs ne se trouvant pas encore dans une table sont regroupés dans une dernière table.
 5. si la clé ne se trouve pas dans son intégralité dans une table, créer une nouvelle table pour la clé.

Exemple

$\mathcal{U} = \{\text{matiere}, \text{nom}, \text{prenom}, \text{noe}, \text{notecc}, \text{noteex}, \text{responsable}, \text{nobureau}\}$

1. $\text{matiere} \rightarrow \text{responsable}$
2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$
3. $\text{responsable} \rightarrow \text{nobureau}$
4. $\text{matiere}, \text{noe} \rightarrow \text{notecc}$
5. $\text{matiere}, \text{noe} \rightarrow \text{noteex}$



Exemple

$U = \{\text{matiere}, \text{nom}, \text{prenom}, \text{noe}, \text{notecc}, \text{noteex}, \text{responsable}, \text{nobureau}\}$

1. matiere $\rightarrow$ responsable

2. noe $\rightarrow$ nom, prenom

3. responsable $\rightarrow$ nobureau

4. matiere, noe $\rightarrow$ notecc

5. matiere, noe $\rightarrow$ noteex

Matiere(matiere, #responsable)

1. matiere $\rightarrow$ responsable



Exemple

$U = \{\text{matiere}, \text{nom}, \text{prenom}, \text{noe}, \text{notecc}, \text{noteex}, \text{responsable}, \text{nobureau}\}$

1. $\text{matiere} \rightarrow \text{responsable}$

2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$

3. $\text{responsable} \rightarrow \text{nobureau}$

4. $\text{matiere}, \text{noe} \rightarrow \text{notecc}$

5. $\text{matiere}, \text{noe} \rightarrow \text{noteex}$

$\text{Matiere}(\text{matiere}, \# \text{responsable})$

1. $\text{matiere} \rightarrow \text{responsable}$

$\text{Etudiant}(\text{nom}, \text{prenom}, \text{noe})$

1. $\text{noe} \rightarrow \text{nom}, \text{prenom}$

Exemple

$U = \{\text{matiere}, \text{nom}, \text{prenom}, \text{noe}, \text{notecc}, \text{noteex}, \text{responsable}, \text{nobureau}\}$

1. $\text{matiere} \rightarrow \text{responsable}$

2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$

3. $\text{responsable} \rightarrow \text{nobureau}$

4. $\text{matiere}, \text{noe} \rightarrow \text{notecc}$

5. $\text{matiere}, \text{noe} \rightarrow \text{noteex}$

$\text{Matiere}(\text{matiere}, \# \text{responsable})$

1. $\text{matiere} \rightarrow \text{responsable}$

$\text{Etudiant}(\text{nom}, \text{prenom}, \text{noe})$

1. $\text{noe} \rightarrow \text{nom}, \text{prenom}$

$\text{Enseignant}(\text{responsable}, \text{nobureau})$

1. $\text{responsable} \rightarrow \text{nobureau}$

Exemple

$U = \{\text{matiere}, \text{nom}, \text{prenom}, \text{noe}, \text{notecc}, \text{noteex}, \text{responsable}, \text{nobureau}\}$

1. $\text{matiere} \rightarrow \text{responsable}$

2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$

3. $\text{responsable} \rightarrow \text{nobureau}$

4. $\text{matiere}, \text{noe} \rightarrow \text{notecc}$

5. $\text{matiere}, \text{noe} \rightarrow \text{noteex}$

$\text{Matiere}(\text{matiere}, \# \text{responsable})$

1. $\text{matiere} \rightarrow \text{responsable}$

$\text{Enseignant}(\text{responsable}, \text{nobureau})$

1. $\text{responsable} \rightarrow \text{nobureau}$

$\text{Etudiant}(\text{nom}, \text{prenom}, \text{noe})$

1. $\text{noe} \rightarrow \text{nom}, \text{prenom}$

$\text{Notes}(\# \text{matiere}, \# \text{noe}, \text{notecc}, \text{noteex})$

1. $\text{matiere}, \text{noe} \rightarrow \text{notecc}$

2. $\text{matiere}, \text{noe} \rightarrow \text{noteex}$

Algorithme de décomposition

- Soit $R \langle U, F \rangle$ munie de la clé X
 - On peut toujours décomposer une relation suivant une DF
 - On ne peut pas décomposer une relation sans DF
 - La décomposition suivant une DF ne perd pas d'information
- Principe :
 - calcul d' $I(F)$
 - décomposition progressive des relations sur les DF posant des problèmes de normalisation, les DF concernant les attributs extraits sont associées à la nouvelle relation.
 - commencer par les DF non-directes puis prendre les DF non-élémentaires ;
 - prendre aussi les attributs (et les DF concernées) obtenus par la fermeture transitive de la partie gauche ; Reprendre les DF dont certains attributs partent (à partir de F^+)
 - la fin des décompositions est détectée lorsque soit toutes les DF ont été utilisées soit toutes les relations sont en 3FN.
 - nommage des relations obtenues si nécessaire.

Exemple (version 1)

Notes(matiere, nom, prenom, noe, notecc, noteex, responsable, nobureau)

1. *matiere* → responsable
2. *noe* → nom, prenom
3. *responsable* → nobureau
4. *matiere, noe* → notecc, noteex
5. *matiere, noe* → nom, prenom
6. *matiere, noe* → responsable
7. *matiere, noe* → nobureau

Issues de la clé



Exemple (version 1)

Notes(matiere, nom, prenom, noe, notecc, noteex, responsable, nobureau)

1. *matiere* → responsable
2. *noe* → nom, prenom
3. responsable → nobureau
4. *matiere*, *noe* → notecc, noteex
5. *matiere*, *noe* → nom, prenom
6. *matiere*, *noe* → responsable
7. *matiere*, *noe* → nobureau

Pas en 2FN

Issues de la clé

Exemple (version 1)

1. $\text{matiere} \rightarrow \text{responsable}$
2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$
3. $\text{responsable} \rightarrow \text{nobureau}$
4. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$
5. $\text{matiere}, \text{noe} \rightarrow \text{nom}, \text{prenom}$
6. $\text{matiere}, \text{noe} \rightarrow \text{responsable}$
7. $\text{matiere}, \text{noe} \rightarrow \text{nobureau}$

$\text{Notes}(\text{matiere}, \text{nom}, \text{prenom}, \text{noe}, \text{notecc}, \text{noteex}, \text{responsable}, \text{nobureau})$

1. $\text{matiere} \rightarrow \text{responsable}$
2. $\text{responsable} \rightarrow \text{nobureau}$
3. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$
4. $\text{matiere}, \text{noe} \rightarrow \text{responsable}$
5. $\text{matiere}, \text{noe} \rightarrow \text{nobureau}$

$\text{Etudiant}(\text{nom}, \text{prenom}, \text{noe})$

6. $\text{noe} \rightarrow \text{nom}, \text{prenom}$



Exemple (version 1)

1. $\text{matiere} \rightarrow \text{responsable}$
2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$
3. $\text{responsable} \rightarrow \text{nobureau}$
4. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$
5. $\text{matiere}, \text{noe} \rightarrow \text{nom}, \text{prenom}$
6. $\text{matiere}, \text{noe} \rightarrow \text{responsable}$
7. $\text{matiere}, \text{noe} \rightarrow \text{nobureau}$

$\text{Notes}(\text{matiere}, \text{nom}, \text{prenom}, \text{noe}, \text{notecc}, \text{noteex}, \text{responsable}, \text{nobureau})$

1. $\text{matiere} \rightarrow \text{responsable}$
2. $\text{responsable} \rightarrow \text{nobureau}$
3. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$
4. $\text{matiere}, \text{noe} \rightarrow \text{responsable}$
5. $\text{matiere}, \text{noe} \rightarrow \text{nobureau}$

Pas en 2FN

$\text{Etudiant}(\text{nom}, \text{prenom}, \text{noe})$

6. $\text{noe} \rightarrow \text{nom}, \text{prenom}$

Exemple (version 1)

1. $\text{matiere} \rightarrow \text{responsable}$
2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$
3. $\text{responsable} \rightarrow \text{nobureau}$
4. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$
5. $\text{matiere}, \text{noe} \rightarrow \text{nom}, \text{prenom}$
6. $\text{matiere}, \text{noe} \rightarrow \text{responsable}$
7. $\text{matiere}, \text{noe} \rightarrow \text{nobureau}$

$\text{Notes}(\# \text{matiere}, \# \text{noe}, \text{notecc}, \text{noteex}, \text{nobureau})$

1. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$
2. $\text{responsable} \rightarrow \text{nobureau}$
3. $\text{matiere}, \text{noe} \rightarrow \text{nobureau}$

$\text{Etudiant}(\text{nom}, \text{prenom}, \text{noe})$

4. $\text{noe} \rightarrow \text{nom}, \text{prenom}$

$\text{Matiere}(\text{matiere}, \text{responsable})$

5. $\text{matiere} \rightarrow \text{responsable}$

Exemple (version 1)

1. $\text{matiere} \rightarrow \text{responsable}$
2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$
3. $\text{responsable} \rightarrow \text{nobureau}$
4. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$
5. $\text{matiere}, \text{noe} \rightarrow \text{nom}, \text{prenom}$
6. $\text{matiere}, \text{noe} \rightarrow \text{responsable}$
7. $\text{matiere}, \text{noe} \rightarrow \text{nobureau}$

$\text{Notes}(\# \text{matiere}, \# \text{noe}, \text{notecc}, \text{noteex}, \text{nobureau})$

1. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$
2. $\text{responsable} \rightarrow \text{nobureau}$
3. $\text{matiere}, \text{noe} \rightarrow \text{nobureau}$

$\text{Etudiant}(\text{nom}, \text{prenom}, \text{noe})$

4. $\text{noe} \rightarrow \text{nom}, \text{prenom}$

$\text{Matiere}(\text{matiere}, \text{responsable})$

5. $\text{matiere} \rightarrow \text{responsable}$

En 3FN !!!

Exemple (version 2)

Notes(matiere, nom, prenom, noe, notecc, noteex, responsable, nobureau)

1. *matiere* → responsable
2. *noe* → nom, prenom
3. *responsable* → nobureau
4. *matiere, noe* → notecc, noteex
5. *matiere, noe* → nom, prenom
6. *matiere, noe* → responsable
7. *matiere, noe* → nobureau

Issues de la clé



Exemple (version 2)

Notes(matiere, nom, prenom, noe, notecc, noteex, responsable, nobureau)

1. *matiere* → responsable
2. *noe* → nom, prenom
3. responsable → nobureau
4. *matiere*, *noe* → notecc, noteex
5. *matiere*, *noe* → nom, prenom
6. *matiere*, *noe* → responsable
7. *matiere*, *noe* → nobureau

*Pas en 2FN
Cf. Version 1*

Issues de la clé

Exemple (version 2)

1. $\text{matiere} \rightarrow \text{responsable}$
2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$
3. $\text{responsable} \rightarrow \text{nobureau}$
4. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$
5. $\text{matiere}, \text{noe} \rightarrow \text{nom}, \text{prenom}$
6. $\text{matiere}, \text{noe} \rightarrow \text{responsable}$
7. $\text{matiere}, \text{noe} \rightarrow \text{nobureau}$

$\text{Notes}(\# \text{matiere}, \# \text{noe}, \text{notecc}, \text{noteex})$

1. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$

$\text{Etudiant}(\text{nom}, \text{prenom}, \text{noe})$

2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$

$\text{Matiere}(\text{matiere}, \text{responsable}, \text{nobureau})$

3. $\text{matiere} \rightarrow \text{responsable}$
4. $\text{responsable} \rightarrow \text{nobureau}$

Obtenue en ajoutant la fermeture transitive de « matiere » puis en ne gardant que les DF concernées par les attributs

Exemple (version 2)

1. $\text{matiere} \rightarrow \text{responsable}$
2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$
3. $\text{responsable} \rightarrow \text{nobureau}$
4. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$
5. $\text{matiere}, \text{noe} \rightarrow \text{nom}, \text{prenom}$
6. $\text{matiere}, \text{noe} \rightarrow \text{responsable}$
7. $\text{matiere}, \text{noe} \rightarrow \text{nobureau}$

$\text{Notes}(\# \text{matiere}, \# \text{noe}, \text{notecc}, \text{noteex})$

1. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$

$\text{Etudiant}(\text{nom}, \text{prenom}, \text{noe})$

2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$

$\text{Matiere}(\text{matiere}, \text{responsable}, \text{nobureau})$

3. $\text{matiere} \rightarrow \text{responsable}$

4. $\text{responsable} \rightarrow \text{nobureau}$

Pas en 3FN

Obtenue en ajoutant la fermeture transitive de «matiere» puis en ne gardant que les DF concernées par les attributs

Exemple (version 2)

1. $matiere \rightarrow responsable$
2. $noe \rightarrow nom, prenom$
3. $responsable \rightarrow nobureau$
4. $matiere, noe \rightarrow notecc, noteex$
5. $matiere, noe \rightarrow nom, prenom$
6. $matiere, noe \rightarrow responsable$
7. $matiere, noe \rightarrow nobureau$

$Notes(\#matiere, \#noe, notecc, noteex)$

1. $matiere, noe \rightarrow notecc, noteex$

$Etudiant(nom, prenom, noe)$

2. $noe \rightarrow nom, prenom$

$Matiere(matiere, \#responsable)$

3. $matiere \rightarrow responsable$

$Responsable(responsable, nobureau)$

4. $responsable \rightarrow nobureau$

Exemple (version 2)

1. $\text{matiere} \rightarrow \text{responsable}$
2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$
3. $\text{responsable} \rightarrow \text{nobureau}$
4. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$
5. $\text{matiere}, \text{noe} \rightarrow \text{nom}, \text{prenom}$
6. $\text{matiere}, \text{noe} \rightarrow \text{responsable}$
7. $\text{matiere}, \text{noe} \rightarrow \text{nobureau}$

$\text{Notes}(\# \text{matiere}, \# \text{noe}, \text{notecc}, \text{noteex})$

1. $\text{matiere}, \text{noe} \rightarrow \text{notecc}, \text{noteex}$

$\text{Etudiant}(\text{nom}, \text{prenom}, \text{noe})$

2. $\text{noe} \rightarrow \text{nom}, \text{prenom}$

$\text{Matiere}(\text{matiere}, \# \text{responsable})$

3. $\text{matiere} \rightarrow \text{responsable}$

$\text{Responsable}(\text{responsable}, \text{nobureau})$

4. $\text{responsable} \rightarrow \text{nobureau}$

En 3FN !!!

Exemple (version 3)

Notes(matiere, nom, prenom, noe, notecc, noteex, responsable, nobureau)

1. *matiere* → *responsable*
2. *noe* → *nom*, *prenom*
3. *responsable* → *nobureau*
4. *matiere*, *noe* → *notecc*, *noteex*
5. *matiere*, *noe* → *nom*, *prenom*
6. *matiere*, *noe* → *responsable*
7. *matiere*, *noe* → *nobureau*

Issues de la clé



Exemple (version 3)

Notes(matiere, nom, prenom, noe, notecc, noteex, responsable, nobureau)

1. *matiere* → *responsable*
2. *noe* → *nom*, *prenom*
3. *responsable* → *nobureau*
4. *matiere*, *noe* → *notecc*, *noteex*
5. *matiere*, *noe* → *nom*, *prenom*
6. *matiere*, *noe* → *responsable*
7. *matiere*, *noe* → *nobureau*

Pas en 3FN

Issues de la clé

Exemple (version 3)

Notes(matiere, nom, prenom, noe, notecc, noteex, #responsable)

1. *matiere $\rightarrow$ responsable*
2. *noe $\rightarrow$ nom, prenom*
3. *matiere, noe $\rightarrow$ notecc, noteex*
4. *matiere, noe $\rightarrow$ nom, prenom*
5. *matiere, noe $\rightarrow$ responsable*

Enseignant(responsable, nobureau)

6. *responsable $\rightarrow$ nobureau*



Exemple (version 3)

Notes(matiere, nom, prenom, noe, notecc, noteex, #responsable)

1. *matiere* → *responsable*

2. *noe* → *nom*, *prenom*

3. *matiere*, *noe* → *notecc*, *noteex*

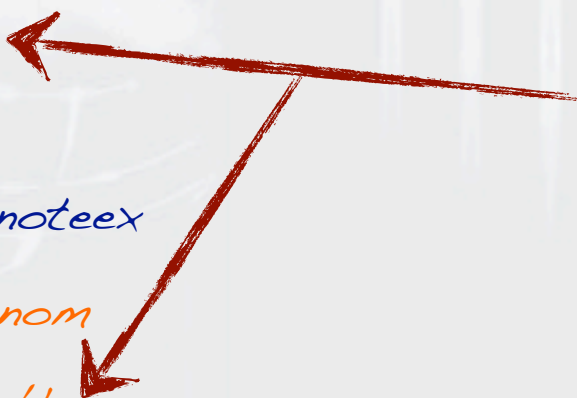
4. *matiere*, *noe* → *nom*, *prenom*

5. *matiere*, *noe* → *responsable*

Enseignant(responsable, nobureau)

6. *responsable* → *nobureau*

Pas en 2FN



Exemple (version 3)

Notes(#matiere, nom, prenom, noe, notecc, noteex)

1. *noe $\rightarrow$ nom, prenom*
2. *matiere, noe $\rightarrow$ notecc, noteex*
3. *matiere, noe $\rightarrow$ nom, prenom*

Module(matiere, #responsable)

4. *matiere $\rightarrow$ responsable*

Enseignant(responsable, nobureau)

5. *responsable $\rightarrow$ nobureau*



Exemple (version 3)

Notes(#matiere, nom, prenom, noe, notecc, noteex)

1. *noe* → *nom*, *prenom*

2. *matiere*, *noe* → *notecc*, *noteex*

3. *matiere*, *noe* → *nom*, *prenom*

Pas en 2FN

Module(matiere, #responsable)

4. *matiere* → *responsable*

Enseignant(responsable, nobureau)

5. *responsable* → *nobureau*

Exemple (version 3)

Notes(#matiere, #noe, notecc, noteex,)

1. *matiere, noe $\rightarrow$ notecc, noteex*

Etudiant(noe, nom, prenom)

2. *noe $\rightarrow$ nom, prenom*

Module(matiere, #responsable)

3. *matiere $\rightarrow$ responsable*

Enseignant(responsable, nobureau)

4. *responsable $\rightarrow$ nobureau*



Exemple (version 3)

Notes(#matiere, #noe, notecc, noteex,)

1. *matiere, noe $\rightarrow$ notecc, noteex*

Etudiant(noe, nom, prenom)

2. *noe $\rightarrow$ nom, prenom*

Module(matiere, #responsable)

3. *matiere $\rightarrow$ responsable*

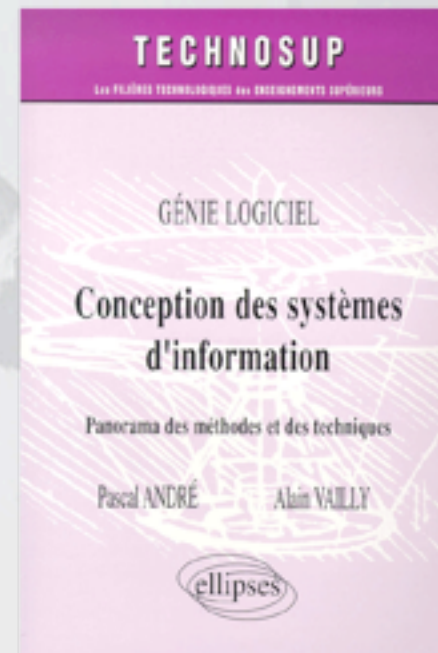
Enseignant(responsable, nobureau)

4. *responsable $\rightarrow$ nobureau*

En 3FN !!!

Mentions

- La structure de ce cours, certains exemples voire certains transparents sont issus du cours d'Alain Vailly pour le modules S21I030 de l'année 2009-2010 (avec son aimable autorisation)
 - Cours en ligne e-miage : <http://pagesperso.lina.univ-nantes.fr/info/perso/permanents/vailly/Enseignement/EMIAGE/EnDev/ModuleB302EMiageV2.3/b302.html>
 - Transparents du module S21I030 pour l'année 2009-2010
 - Livre : "Conception des systèmes d'information", P. André & A. Vailly, Ellipse, 2001
 - Fiches de TD et TP du module S21I030 pour l'année 2009-2010



Cette création est mise à disposition selon le Contrat
Paternité-Pas d'Utilisation Commerciale-Pas de Modification 2.0 France
disponible en ligne <http://creativecommons.org/licenses/by-nc-nd/2.0/fr/>

ou par courrier postal à Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.