

G1: MODULE MSI-MI MODÉLISATION DES SYSTÈMES D'INFORMATION- MICRO INFORMATIQUE

ANNÉE SCOLAIRE 2015-2016

SÉANCE 4 : COURS

DIDIER CORBEEL & JEAN-PIERRE BOUREY, ÉCOLE CENTRALE DE LILLE



Cette œuvre est mise à disposition selon les termes de la Licence Creative Commons Attribution – Pas d'Utilisation Commerciale – Partage dans les Mêmes Conditions 3.0 France disponible en ligne <http://creativecommons.org/licenses/by-nc-sa/3.0/fr/> ou par courrier postal à Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA



EC-Lille, Didier Corbeel & Jean-Pierre
Bourey, Module MSI-MI G1

31/05/2016

1



- ▶ Cette œuvre est mise à disposition selon les termes de la Licence Creative Commons Attribution – Pas d'Utilisation Commerciale – Partage dans les Mêmes Conditions 3.0 France disponible en ligne <http://creativecommons.org/licenses/by-nc-sa/3.0/fr/>
- ▶ This work is licensed under Creative Commons Attribution-NonCommercial-ShareAlike 3.0 available online at <http://creativecommons.org/licenses/by-nc-sa/3.0/> or by regular mail at Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.



EC-Lille, Didier Corbeel & Jean-Pierre
Bourey, Module MSI-MI G1

31/05/2016

2

PLAN DES SÉANCES

N°	Contenu
1	Présentation du module Systèmes d'Information (SI) et base de données relationnelle (BDR) Démarche de développement Diagrammes de classes Diagrammes d'objets
2	Agrégation, composition Association-classe Types de données et énumérations Diagramme de paquetages
3	Généralisation spécialisation Classe abstraite Généralisation multiple
4	Modèle relationnel, dépendances fonctionnelles, normalisation
5	Passage d'un modèle conceptuel à un script pour une BDR
6	Travail en autonomie



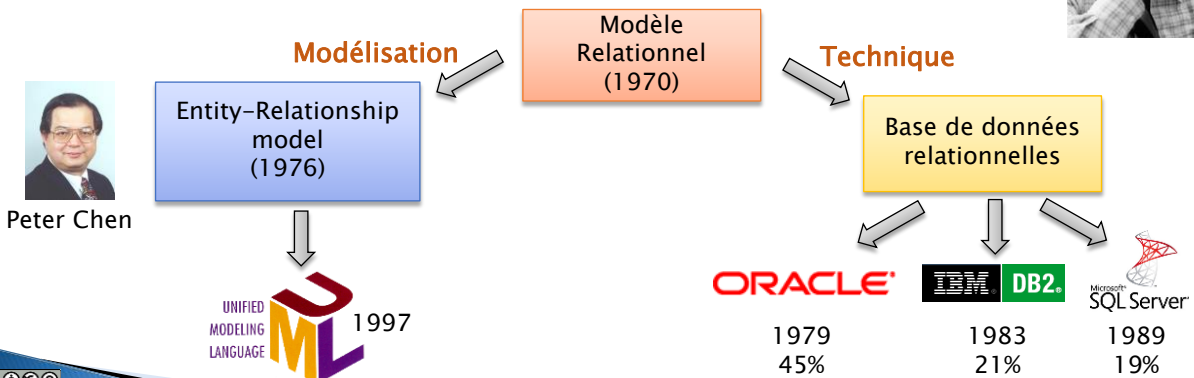
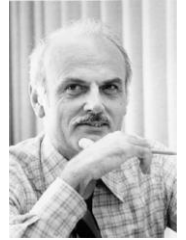
AGENDA

- Définition d'un modèle relationnel
- Les opérateurs
- Pourquoi la normalisation ?
- Les bases de la normalisation : les dépendances fonctionnelles
- Algorithme de normalisation
- Quel modèle choisir ?



POURQUOI CETTE PARTIE ?

- ▶ Modèle relationnel défini par Edgar Frank Codd (1923–2003) en 1970
- ▶ Basé sur le concept de relation introduit par la théorie de **l'algèbre relationnelle**
- ▶ A l'origine de deux axes importants et complémentaires de développement



LE MODÈLE RELATIONNEL : DÉFINITIONS



LE MODÈLE RELATIONNEL ...

- Hypothèse : Chaque occurrence de phénomène du monde réel peut être décrit par une relation.
- 1. Le concept de **relation**.

Soient n ensembles D_1, D_2, \dots, D_n

Une relation R sur ces ensembles : $R(D_1, D_2, \dots, D_n) \subset D_1 \times D_2 \times \dots \times D_n$

R est un ensemble de **n-uplets** (ou **tuples**) de la forme

$\langle d_1, d_2, \dots, d_n \rangle$ tels que $d_1 \in D_1, d_2 \in D_2, \dots, d_n \in D_n$



... LE MODÈLE RELATIONNEL ...

- Transposition au domaine des données.
- D_i : **domaines de valeur** des propriétés

DOMAINE : ensemble de valeurs.

- Exemple 1
 - D_1 : ensemble des chaînes de caractères de longueur au plus 20.
 - D_3 : ensemble des entiers
 - D_4 : ensemble des réels.



... LE MODÈLE RELATIONNEL ...

ATTRIBUT : variable prenant ses valeurs dans un domaine.

► Exemple 2

- l'attribut **nom** prend ses valeurs dans D_1 (ensemble des chaînes de caract. de longueur max 20)
- l'attribut **ville** prend ses valeurs dans $D_2 = D_1$
- l'attribut **num** prend ses valeurs dans D_3 (ensemble des entiers)
- l'attribut **salaire** prend ses valeurs dans D_4 (ensemble des réels)



... LE MODÈLE RELATIONNEL ...

► Relation

Une **RELATION** sur les attributs A_1, A_2, \dots, A_n de domaines respectifs D_1, D_2, \dots, D_n est tout sous-ensemble du produit cartésien de $D_1 \times D_2 \times \dots \times D_n$

► Exemple 3

- une relation **R** sur **NOM**, **VILLE**, **NUM**, **SALAIRE** peut être décrite comme un ensemble de quadruplets
- $R = \{ (\text{Dupont}, \text{Paris}, 2140, 1200.50),$
 $(\text{Durand}, \text{Orsay}, 1128, 2000.50),$
 $(\text{Dubois}, \text{Orsay}, 3213, 1500.99)$
 $\}$



... LE MODÈLE RELATIONNEL ...

- Une relation peut être représentée sous forme de tableaux dont les noms de colonnes sont ceux des attributs correspondants.
- Exemple 4 :
 - La relation **R** peut être écrite sous la forme :

NOM	VILLE	NUM	SALAIRE
Dupont	Paris	2140	1200.50
Durand	Orsay	1128	2000.50
Dubois	Orsay	3213	1500.99



... LE MODÈLE RELATIONNEL ...

- Remarques :
 - Un tableau ne peut représenter une relation que s'il ne contient **pas deux lignes identiques**.
 - **L'ordre des lignes n'a pas d'importance.**
 - **L'ordre des colonnes n'a pas d'importance** à partir du moment où chacune possède un nom.
 - **Chaque case** du tableau **ne contient qu'une seule valeur**.



... LE MODÈLE RELATIONNEL ...

► Schéma de relation R

Un **SCHÉMA DE RELATION R** est la liste des attributs de la relation **R** avec leur DOMAINE

► Exemple 5 :

- La relation **R** de l'exemple 3 a pour schéma : **R = [NOM:D₁, VILLE:D₁, NUM:D₃, SALAIRE:D₄]**



... LE MODÈLE RELATIONNEL ...

► Le schéma de relation

Employe = [NOM:D₁, VILLE:D₁, NUM:D₃, SALAIRE:D₄]

► sera converti en une TABLE Relationnelle (Oracle)

```
create table EMPLOYE (NOM          varchar2(20) ,
                     VILLE        varchar2(20) ,
                     NUM           integer,
                     SALAIRE       real
                     ) ;
```



... LE MODÈLE RELATIONNEL ...

▸ **Prédicats** associés à une relation.

- Une relation $R(D_1, D_2, \dots, D_n)$ est une partie du produit cartésien $D_1 \times D_2 \times \dots \times D_n$.
- Toutes les valeurs possibles de l'un quelconque des domaines D_i ne figurent pas dans l'ensemble des n-uplets de la relation.
- Pour des raisons de cohérence de données, on associe un ou n prédicats qui précisent quand les valeurs d'un n-uplet sont acceptables.

▸ **CONTRAINTES D'INTÉGRITÉ**



... LE MODÈLE RELATIONNEL ...

▸ Exemple 6 : relation EMPLOYÉ

- Prédicat 1 : tout employé a un salaire supérieur à 1000 euros
- Prédicat 2 : le numéro de matricule d'un employé est OBLIGATOIRE et UNIQUE
- Prédicat 3 : tout employé a un nom
- Prédicat 4 : la ville doit être donnée pour chaque employé



... LE MODÈLE RELATIONNEL ...

- Transcription des prédicats dans le modèle Physique (cf. séance 5)

```
create table EMPLOYE (
  NOM      varchar2(20) not null,
  VILLE    varchar2(20) not null,
  NUM      integer
  SALAIRE  number(5,2)
);
```

constraint PK_EMPLOYE primary key,
constraint CK_EMPLOYE_SALAIRE check (salaire > 1000)

Prédicat 3

Prédicat 4

Prédicat 2

Prédicat 1



... LE MODÈLE RELATIONNEL ...

- Définitions complémentaires et importantes :
 - **Degré de la relation**
 - nombre de ses attributs.
 - Exemple 7 : La relation **R** est de degré 4.
 - **Cardinalité de la relation** :
 - nombre de n-uplets
 - Dans l'exemple 4, la cardinalité de la relation **R** est 3
 - **Identifiant** :
 - Un attribut ou un ensemble d'attributs dont les valeurs déterminent de façon univoque un n-uplet de la relation.
 - Une relation peut comporter plusieurs identifiants.
 - Une relation possède au moins un identifiant : la liste de tous ses attributs.
 - Notation : dans une relation, on souligne les constituants de l'identifiant



... LE MODÈLE RELATIONNEL ...

- ▶ Exemple 8 : Relation **EMPLOYE (NOM, PRENOM, ADR, SALAIRE, MATRICULE)**
 - Identifiant1 : **MATRICULE**
 - Identifiant2 : **<NOM, PRENOM, VILLE>**
- ▶ Parmi les n identifiants d'une relation, on en choisit UN arbitrairement qui sera la **CLÉ PRIMAIRE** de la relation (exemple : **MATRICULE**)
- ▶ **EMPLOYE (NOM, PRENOM, VILLE, SALAIRE, MATRICULE)**
- ▶ Les autres identifiants possibles seront dits
 - **CLÉS SECONDAIRES**
 - ou **CLÉS CANDIDATES**
 - ou **CLÉS ALTERNATIVES**



... LE MODÈLE RELATIONNEL ...

- ▶ Transcription dans le modèle Physique

```
create table EMPLOYE (
  MATRICULE integer constraint PK_EMPLOYE primary key
  NOM varchar2(20) not null,
  PRENOM varchar2(20) not null,
  VILLE varchar2(20) not null,
  SALAIRE numeric(5,2),
  constraint UK_EMP_NOM_PRENOM_VILLE unique(nom, prenom, ville)
);
```

Identifiant défini par la clé primaire

Clé secondaire ou alternative



VOCABULAIRE DU MODÈLE RELATIONNEL (1)

Nom de la table :
Nom de la relation

Clé primaire : identifiant
de la relation

Nom des colonnes :
Attributs de la
relation

Cardinalité :
Nb de lignes

Ligne
ou n-uplet ou tuple

Valeur non définie
NULL

Degré :
Nb de colonnes

Commande	NumCde	DateBCde	CodeFou	TotalCde
	1050	26/10/2015	TH008	1548,50
	1025	19/09/2015	TH009	17235,45
	1026	23/09/2015	MN010	38756,25
	1021	03/07/2015	XA11	49625,23
	1048	10/10/2015	MB117	2210,45
	1055	10/11/2015	BG210	9524,78
	2075	24/11/2015	MN010	789,56
	3140	24/07/2015	BS120	751,25
	1059		MB210	33280,85
	1045	01/10/2015	TH009	45698,54



DÉFINITIONS

► Clé primaire

- Identifiant d'un n-uplet.
- Il ne peut y avoir deux n-uplets ayant la même valeur de clé primaire (unicité)
- Elle doit être obligatoirement renseignée (pas de valeur **NULL**).
- Peut être définie par une combinaison des valeurs de plusieurs attributs d'un n-uplet.

► Clé étrangère

- identifie une colonne ou un ensemble de colonnes d'une table (de départ) comme référençant une colonne ou un ensemble de colonnes d'une table référencée (autre table ou la même)
- Référence à une clé primaire (le plus souvent) ou clé unique
- Sert pour l'intégrité référentielle
- Une valeur de la table référencée ne peut être supprimée que s'il n'existe plus de n-uplets dans la table de départ la référençant
- Par défaut, elle autorise les valeurs non renseignées (**NULL**) mais on peut les interdire en ajoutant une contrainte **NOT NULL**



RÉSUMÉ SUR LES CLÉS



Type de clé	Caractéristiques des valeurs	Pour quoi faire ?	Combien par relation ?	Conseil
Primaire	<ul style="list-style-type: none"> Pas de doublon Obligatoires 	<ul style="list-style-type: none"> Identifiant d'un n-uplet Peut être référencée par une clé étrangère 	1 .. 1	Choisir une clé primaire n'ayant pas de sens métier
Unique	<ul style="list-style-type: none"> Pas de doublon Peuvent être non définies 	<ul style="list-style-type: none"> Peut être référencée par une clé étrangère 	0 .. *	
Métier (clé secondaire, alternative)	<ul style="list-style-type: none"> Pas de doublon Obligatoires Ont un sens pour le métier 	<ul style="list-style-type: none"> Identifiant d'un n-uplet Peut être référencée par une clé étrangère 	1 .. *	Éviter de prendre une clé métier pour clé primaire pour faciliter les modifications (cf. exemple diapo suivante)
Etrangère	<ul style="list-style-type: none"> Référentent une clé primaire, secondaire ou unique Peuvent être non définies 	<ul style="list-style-type: none"> Intégrité référentielle 	0 .. *	Choisir comme référence une clé n'ayant pas de sens métier pour faciliter les modifications



VOCABULAIRE DU MODÈLE RELATIONNEL (2)

Domaine de valeur des n° de fournisseurs

TH008 MB210
MN010
MB117 XA011
BG210
TH009 BS120

Intégrité de domaine

Intégrité de domaine

Clé primaire

Fournisseur

NoFour	AdrFour	----
TH008	Paris	----
TH009	Toulouse	--
MN010	Grenoble	--
XA11	Lille	
MB117	P	
BG210		

Intégrité référentielle

Commande	NumCde	DateBCde	CodeFou	TotalCde
	1050	26/10/2015	TH008	1548,
	1025	19/09/2015	TH009	17235
	1026	23/09/2015	MN010	3875
	1021	03/07/2015	XA11	
	1048	10/10/2		

Clé étrangère



LES OPÉRATEURS

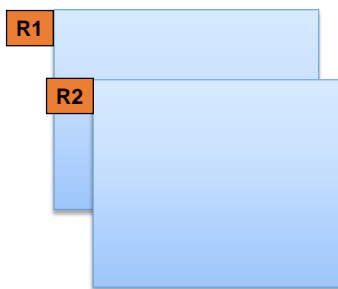


EC-Lille, Didier Corbeel & Jean-Pierre
Bourey, Module MSI-MI G1

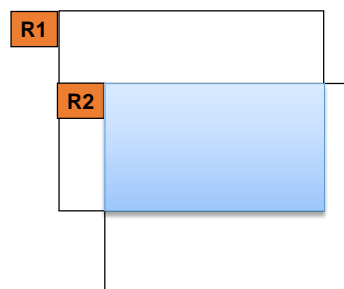
31/05/2016

25

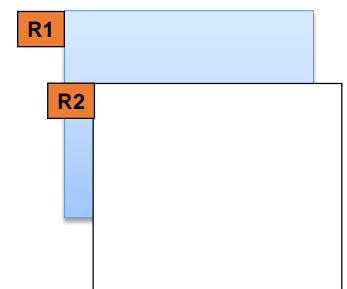
OPÉRATEURS DE L'ALGÈBRE RELATIONNELLE (1)



UNION
 $R1 \cup R2$



INTERSECTION
 $R1 \cap R2$



DIFFERENCE
 $R1 - R2$

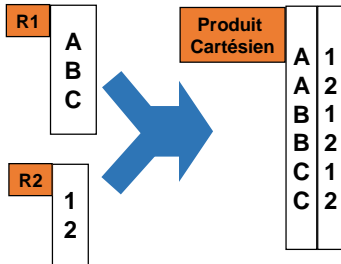


EC-Lille, Didier Corbeel & Jean-Pierre
Bourey, Module MSI-MI G1

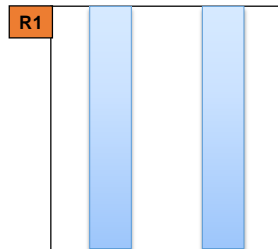
31/05/2016

26

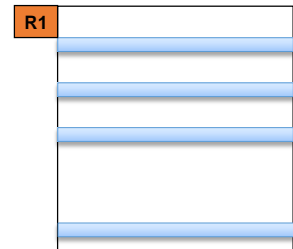
OPÉRATEURS DE L'ALGÈBRE RELATIONNELLE (2)



PRODUIT
 $R1 * R2$



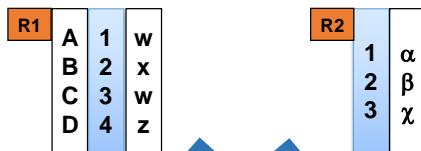
PROJECTION
 $\Pi_{A_i, A_k}(R1)$



RESTRICTION
 $\sigma_{condition}(R1)$



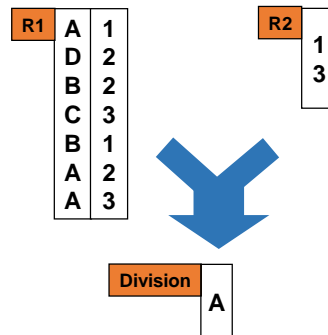
OPÉRATEURS DE L'ALGÈBRE RELATIONNELLE (3)



(equi)Jointure sur colonne "chiffre"

A	w	1	α
B	x	2	β
C	w	3	χ

JOINTURE
 $R1 \bowtie_{chiffre} R2$



DIVISION
 $R1 / R2$



OPÉRATEURS DE L'ALGÈBRE RELATIONNELLE (4)

Tous ces opérateurs s'appliquent à l'ensemble des tuples des relations.

Le résultat d'une opération est une nouvelle relation sur laquelle d'autres opérations vont pouvoir être réalisées.



OPÉRATEURS DE L'ALGÈBRE RELATIONNELLE (5)

► Les trois **opérateurs principaux** de l'algèbre relationnelle sont :

- **La projection**
- **La restriction**
- **La jointure** (ou composition)

► Pour information, un opérateur de l'algèbre relationnelle a été **oublié** dans le standard du langage SQL

- **La division**



OPÉRATEURS DE L'ALGÈBRE RELATIONNELLE (6)

► Soit la relation **EMPLOYE** (NUM, NOM, VILLE, SALAIRE)

NUM	NOM	VILLE	SALAIRE
123	Martin	Lille	1200.00
345	Dupond	Roubaix	1500.00
567	Racine	Lille	1400.00
782	Moliere	Lille	1750.00

► Exemple de **PROJECTION**:

- Quelles sont les villes dans lesquelles habitent les employés ?

Ville
Lille
Roubaix

▪ Notation : $R_1 = \Pi_{A_1, A_2} (R)$

Exemple : $LesVilles = \Pi_{ville} (Employe)$



OPÉRATEURS DE L'ALGÈBRE RELATIONNELLE (7)

► Soit la relation **EMPLOYE** (NUM, NOM, VILLE, SALAIRE)

NUM	NOM	VILLE	SALAIRE
123	Martin	Lille	1200.00
345	Dupond	Roubaix	1500.00
567	Racine	Lille	1400.00
782	Moliere	Lille	1750.00

► Exemple de **RESTRICTION** :

- Quels sont les employés qui gagnent entre 1300 et 1600 euros ?

NUM	NOM	VILLE	SALAIRE
345	Dupond	Roubaix	1500.00
567	Racine	Lille	1400.00

► Notation : $R_1 = \sigma_{condition} (R)$

► Exemple : $EmployesSalairesIntermediaires = \sigma_{salaire > 1300 \text{ et } salaire < 1600} (Employe)$



OPÉRATEURS DE L'ALGÈBRE RELATIONNELLE (8)

- Soient les relations : **DEPARTEMENT**(deptno, nom) et **EMPLOYE**(num, nom, deptno)

Departement	deptno	nom	Employe	num	Nom	deptno
	1	direction		11	King	1
	2	RD		23	Martin	2
	3	finance		34	Moliere	2

- Exemple de JOINTURE: quels sont les employés des départements?

deptno	nom	num	nom	deptno
1	direction	11	King	1
2	RD	23	Martin	2
2	RD	34	Moliere	2

- Notation : $R_3 = R_1 \bowtie_{\text{condition}} R_2$
- Exemple :
- $\text{EmployesDesDepartements} = \text{Departement} \bowtie_{\text{Departement.deptno}=\text{Employe.deptno}} \text{Employe}$



OPÉRATEURS DE L'ALGÈBRE RELATIONNELLE (9)

- Propriétés de ces trois opérateurs

- La **projection réduit le degré** d'une relation
- La **restriction conserve le degré** d'une relation
- La **jointure augmente le degré** d'une relation : $\text{degré}_{\text{résultat}} = \sum \text{degré}_{\text{opérande}}$



POURQUOI LA NORMALISATION ?



MISES À JOUR ET COHÉRENCE

- ▶ But d'un modèle relationnel : décrire une base de données qui va être effectivement utilisée
 - Chargée, consultée, mise à jour (maj)
- ▶ Les maj (insertions, suppressions, modifications) doivent **conserver la cohérence** de la base de données
 - **Intégrité référentielle**
 - Toute contrainte d'intégrité
 - En particulier les dépendances entre attributs
- ▶ Selon le modèle relationnel c'est plus ou moins facile
 - Plus la base de données contient de dépendances, plus les maj avec maintien de la cohérence sont difficiles



EXEMPLE D'ANOMALIES DE MISE À JOUR

► Relation : **Livraison**

Livraison	NFourn	adrFour	NProd	Prix	Qté
	3	Paris	4	40	2
	7	Lille	1	25	1
	5	Ascq	4	40	3
	3	Paris	2	30	7
	3	Lille	8	70	8

- Si un fournisseur change d'adresse et qu'un seul n-uplet est mis à jour => **incohérence**
- Si un nouveau n-uplet est inséré pour un fournisseur connu, avec une adresse différente => **incohérence**
- Si un fournisseur n'a pas de livraison en cours, son adresse est **perdue** ...



QU'EST-CE QU'UNE BD RELATIONNELLE "INCORRECTE" ?

- Une **relation n'est pas correcte** si:
 - Elle implique des **répétitions** au niveau de sa population : **redondances**
 - Elle pose des **problèmes lors des maj** (insertions, modifications et suppressions)
- Les conditions pour qu'une relation soit correcte peuvent être définies formellement :

=> **règles de normalisation**



EXEMPLE (SUITE)

► Relation : **Livraison**

Livraison	NFourn	adrFour	NProd	Prix	Qté
	3	Paris	4	40	2
	7	Lille	1	25	1
	5	Ascq	4	40	3
	3	Paris	2	30	7
	3	Lille	8	70	8

- L'adresse du fournisseur ne dépend que du fournisseur et pas du produit
- Le prix du produit ne dépend que du produit et pas du fournisseur
 - => **Redondances**
 - => **Anomalie de mise à jour**
- Cette relation n'est pas correcte. Il faut la normaliser



NORMALISATION D'UN SCHÉMA

- Processus de transformation d'un schéma S1 pour obtenir un schéma S2:
 - Qui est **équivalent** (même contenu)
 - Dont les **mises à jour sont simples**
- Mise à jour simple :
 - 1 changement élémentaire dans le monde réel se traduit par UNE mise à jour d'UN n-uplet
- Exemples de changements élémentaires
 - Un fournisseur change d'adresse
 - Un produit change de prix
 - Dans l'exemple de la relation **LIVRAISON**, les mises à jour sont complexes



NORMALISATION D'UNE RELATION

- Processus de décomposition d'une relation à maj complexes en plusieurs relations à maj simples
- Exemple : La relation `LIVRAISON(n_fourn, adrF, n_prod, prix, qté)` sera décomposée en :
 - `FOURNISSEUR(n_fourn, adrF)`
 - `PRODUIT(n_prod, prix)`
 - `LIVRAISON(n_fourn, n_prod, qté)`



LES DÉPENDANCES FONCTIONNELLES



DÉPENDANCES FONCTIONNELLES ...

► Notion de dépendance fonctionnelle

Un attribut **B** est en **DÉPENDANCE FONCTIONNELLE** d'un sous ensemble d'attributs **A** dans un schéma de relation **R(A, B, C, D)**

SSI

à toute valeur de $a \in A$ n'est associée qu'une seule valeur $b \in B$

Notation : $A \rightarrow B$



DÉPENDANCES FONCTIONNELLES ...

► Notion de dépendance fonctionnelle : Exemple

Un attribut **NOM** est en **DÉPENDANCE FONCTIONNELLE** d'un sous ensemble d'attributs **NUM** dans un schéma de relation **EMPLOYE(NUM, NOM, VILLE)**

SSI

à toute valeur de $a \in \text{NUM}$ n'est associée qu'une seule valeur $b \in \text{NOM}$

Notation : $\text{NUM} \rightarrow \text{NOM}$

Le numéro d'un employé
détermine son nom



RELATION ENTRE DF ET OPÉRATEUR ALGÈBRE

- Soit $R(A, B, X, Y, Z)$
- Soient t_1 et t_2 2 n-uplet quelconques de R
- $X \rightarrow Y$ si $(\Pi_X(\sigma_{t_1}(R)) = \Pi_X(\sigma_{t_2}(R)) \Rightarrow (\Pi_Y(\sigma_{t_1}(R)) = \Pi_Y(\sigma_{t_2}(R)))$
- (Y est en dépendance fonctionnelle de X si la projection de t_1 sur X est égale à la projection de t_2 sur X implique que la projection de t_1 sur Y doit être égale à la projection de t_2 sur Y)



... DÉPENDANCES FONCTIONNELLES .

- Exemple 10 :

`COMMANDE(Produit, Client, AdresseClient, Quantité, Montant, Date)`

Si , dans notre système, on sait que :

- A un instant donné, un client n'a qu'une seule adresse.
- Un produit commandé par un Client un jour donné correspond à un montant déterminé et unique.
- Un produit commandé par un Client un jour donné correspond à une quantité déterminée et unique.

- ...Dépendances fonctionnelles sont alors :

`Client` → `AdresseClient`
`(Date, Produit, Client)` → `Montant`
`(Date, Produit, Client)` → `Quantité`



... DÉPENDANCES FONCTIONNELLES ...

Commande	Produit	Client	Adresse	NBProduit	DateCde	Montant
	Lessive	Cora	Ascq	200	12/01/2016	3456,58
	Lessive	Cora	Ascq	230	01/02/2016	3900,12
	Yahourt	Cora	Ascq	120	14/02/2016	150,69



... DÉPENDANCES FONCTIONNELLES ...

- Utilisation des DF pour éliminer des redondances

- Remplacer la relation **Commande** par deux relations :

- InformationClient**
- DetailCommande**

InformationClient	Client	Adresse
	Cora	Ascq

Clé étrangère

DetailCommande	Produit	Client	NBProduit	DateCde	Montant
	Lessive	Cora	200	12/01/2016	3456,58
	Lessive	Cora	230	01/02/2016	3900,12
	Yahourt	Cora	120	14/02/2016	150,69



DÉPENDANCE FONCTIONNELLE ÉLÉMENTAIRE ...

Un attribut **B** est en

DÉPENDANCE FONCTIONNELLE ÉLÉMENTAIRE

d'un sous-ensemble d'attributs **A**

- s'il est fonctionnellement dépendant de **A**,
- s'il n'est pas fonctionnellement dépendant d'une partie de **A**.



... DÉPENDANCE FONCTIONNELLE ÉLÉMENTAIRE

Exemple 11 :

FOURNISSEUR (NumFournisseur, NumProduit, Date, Quantité, VilleFourn, CodePostal, Département)

Dépendances fonctionnelles **élémentaires** :

- NumFournisseur, NumProduit, Date → Quantité
- NumFournisseur → VilleFourn

Dépendance fonctionnelle **NON élémentaire** :

- CodePostal → Département
- (car Département, BureauDistrib → Département)



DÉPENDANCE FONCTIONNELLE DIRECTE ...

Un attribut **B** est en

DÉPENDANCE FONCTIONNELLE DIRECTE

d'un sous ensemble d'attributs **A** d'une relation **R(A, B, C, D)**
s'il n'existe pas de sous ensemble d'attributs **C** tel que

$$A \rightarrow C$$

et

$$C \rightarrow B$$



... DÉPENDANCE FONCTIONNELLE DIRECTE.

► Exemple 12 :

EMPLOYÉ (Matricule, NumProjet, BudgetProjet)

► La DF **Matricule** \rightarrow **BudgetProjet** n'est pas directe car

Matricule	\rightarrow	NumProjet
NumProjet	\rightarrow	BudgetProjet



FORME DES RELATIONS ...

- Forme quelconque (0-FN)
(C'est la forme la plus générale)
- **Première forme normale** (1-FN)
 - Une relation est 1-FN si
 - chacun des attributs appartient à un domaine élémentaire
 - toutes les données sont atomiques
 - sont constants dans le temps (**age** vs **dateDeNaissance**)
- Exemple 13 :
 - **PRODUIT**(**CodeProduit**, **Libellé**, **PrixUnit**) est en 1-FN
- Intérêt
 - Pas de mises à jour régulières
 - Recherches plus rapides : pas d'analyse des contenus d'attributs



... FORMES DES RELATIONS ...

- **Deuxième forme normale** (2-FN)
 - Une relation est 2-FN si et seulement si
 - elle est en 1-FN
 - les attributs qui ne font pas partie de l'identifiant sont en dépendance fonctionnelle élémentaire de l'identifiant.
- Exemple 14 :
 - COMMANDE** (**Produit**, **Client**, **Date**, **Quantité**, **Montant**) est en 2-FN
 - COM** (**Produit**, **LibelléProduit**, **Client**, **Date**, **Quantité**, **Montant**, **AdrClient**)
 - n'est pas 2-FN car **Produit** → **LibelléProduit** et **Client** → **AdrClient**
- Intérêt
 - Limite la redondance des données



... FORMES DES RELATIONS.

► Troisième forme normale (3-FN)

Pas de DF d'un autre attribut que l'identifiant

► Une relation est 3-FN si et seulement si

- Elle est en 2-FN (i.e. dépendance fonctionnelle élémentaire)
- les attributs qui ne font pas partie de l'identifiant sont en **dépendance fonctionnelle directe** de l'identifiant.

► Exemple 15 :

- **COMMANDE** (Produit, Client, Date, Quantité, Montant) est 3-FN.
- **SERVICE** (CodeService, Libellé, Budget, Matricule, Nom, Adr, Salaire) n'est pas 3-FN.



Solution :

SERVICE (CodeService, Libellé, Budget)

SERV_EMPL (CodeService#, Matricule#)

EMPLOYÉ (Matricule, Nom, Adr, Salaire)



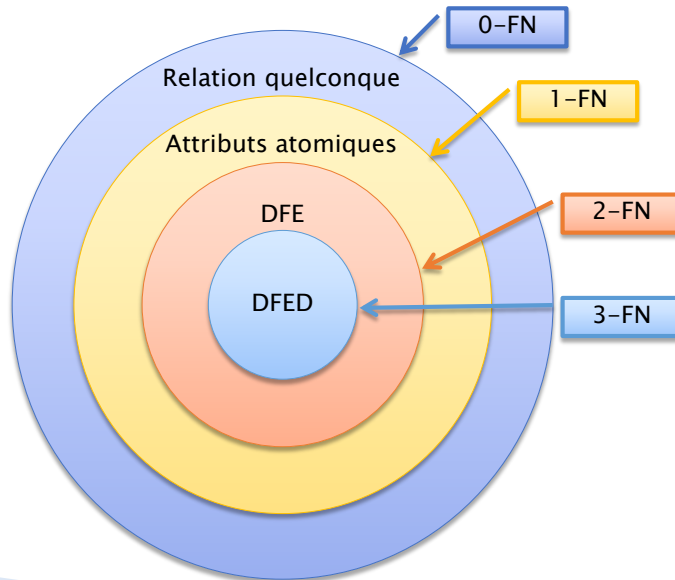
POUR INFORMATION

► Il existe des formes normales plus sophistiquées

- Forme normale de Boyce-Codd (FNBC)
 - 3^{ème} forme normale pour laquelle tous les attributs non-identifiants ne sont pas source de dépendance fonctionnelle (DF) vers une partie de l'identifiant.
- 4^{ème}, 5^{ème}, 6^{ème} formes normales



RÉSUMÉ



POUR RETENIR...MÊME SI CE N'EST PAS TOUT À FAIT EXACT



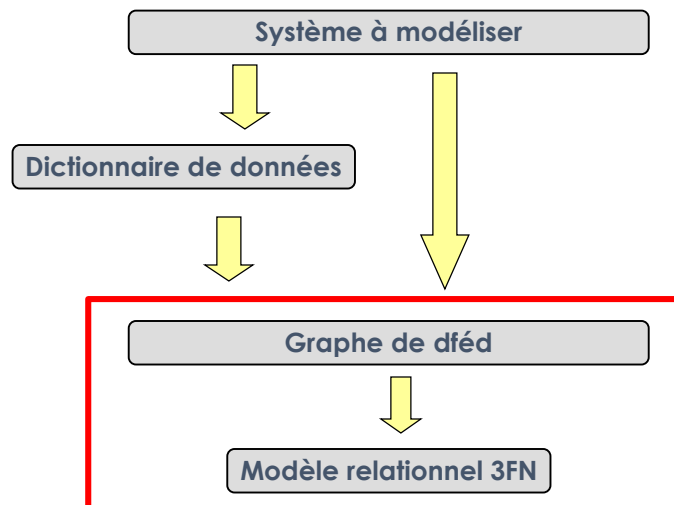
Je jure que dans mes relations,
les attributs dépendent
1) de l'identifiant (1-FN)
2) de tout l'identifiant (2-FN)
3) rien que de l'identifiant (3-FN)



NORMALISATION



PRINCIPES



SYSTÈME À MODÉLISER

▸ Exemple "Cinéma"

- Un film est caractérisé par son titre, son réalisateur, son budget, son année de sortie et le salaire du réalisateur
- Un acteur est caractérisé par son nom, son prénom, sa nationalité, sa date de naissance
- Un acteur est rémunéré pour sa participation à un film

- Les films, acteurs et réalisateurs seront identifiés par un numéro (nfilm, ...)



DICTIONNAIRE DE DONNÉES

- Un **dictionnaire de données**
 - Regroupe les caractéristiques essentielles des données du SI

- Pour chaque donnée pertinente sont définis
 - Le nom
 - Le domaine (numérique, texte ou date/heure)
 - Les règles de validations
 - Les synonymes



EXEMPLE DE DICT. DE DONNÉES (SI CINÉMA)

Nom	Domaine	Règles de validation	Synonymes
Nfilm	Numérique	Identifie un film	
Titre	Texte	De longueur ≤ 80	
Budget	Numérique	Positif ou nul	
Sortie	Numérique	> 1900	
Réalisateur	Numérique	Identifie une personne	npersonne
Salaire_réalisateur	Numérique	Positif ou nul	
Nacteur	Numérique	Identifie un acteur	npersonne
Nom	Texte	De longueur ≤ 30	
Prénom	Texte	De longueur ≤ 16	
Nationalité	Texte	De longueur ≤ 20	
Naissance	Date	$\geq 1/1/1900$	
Salaire_Acteur	Numérique	Positif ou nul	

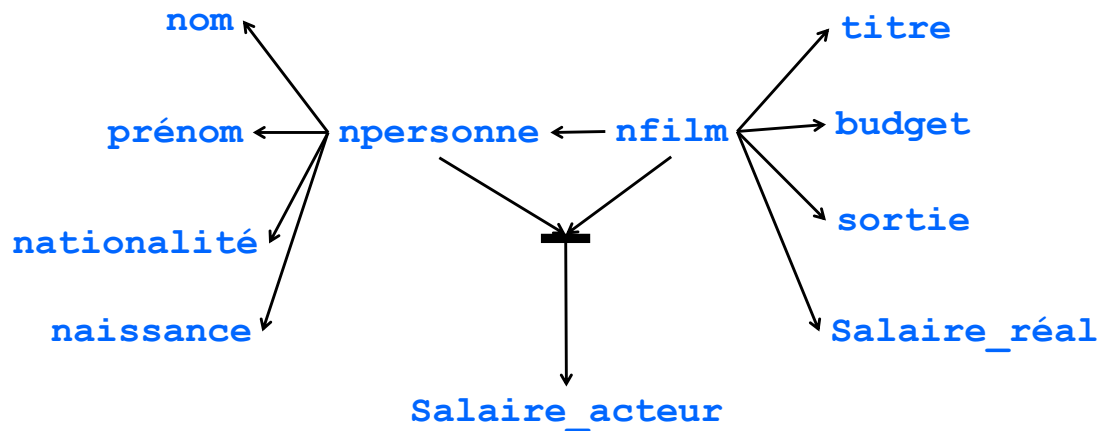


GRAPHE DE DÉPENDANCE FONCTIONNELLE ÉLÉMENTAIRES ET DIRECTES

- Un sommet représente un ensemble de données du dictionnaire :
 - `<nacteur, nfilm>`
 - `<nom>` ou `nom`
- Un arc est orienté et représente un DFED entre deux ensembles de données :
 - `<nfilm> → <titre>`



EXEMPLE DE GRAPHE DE DFED



ALGORITHME DE NORMALISATION DIRECTE

- Cet algorithme
 - se sert du graphe de DFED
 - est itératif
- Chaque itération se décompose en 4 étapes
 1. Dénombrer les relations en 3-FN
 2. Choisir la relation en 3-FN que cette étape va caractériser
 3. Construire cette relation en 3-FN
 4. Éliminer les arcs et sommets inutiles du graphe des DFED

ETAPE 1 : DÉNOMBRER LES RELATIONS ...

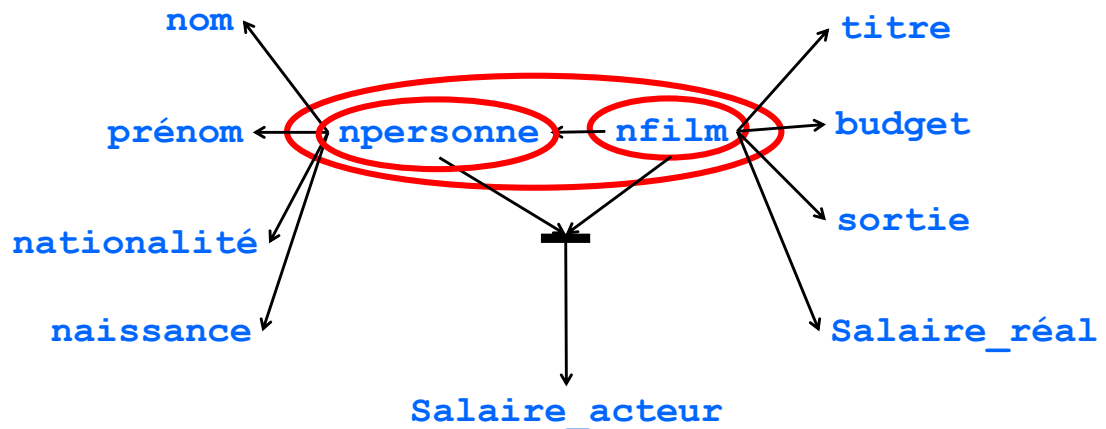
- Dénombrer les relations en 3FN
 - Chaque relation 3FN nécessite la connaissance de son identifiant ou clé primaire
 - Le graphe permet de trouver les identifiants
 - Ce sont les **SOURCES** (sommet dont partent des arcs) **de dféd**

Exemple pour le SI Cinéma

- `<nfilm>`
- `<npersonne>`
- `<npersonne, nfilm>`
- L'algorithme aura TROIS itérations



ETAPE 1 : DÉNOMBRER LES RELATIONS



ETAPE 2 : CHOISIR LA PROCHAINE RELATION À CONSTRUIRE



- L'étape 1 permet de dénombrer les relations, il nous reste à CHOISIR la relation à définir
- Méthode : Critère de choix ou PONDERATION
 - Pondérer chaque source de dféd (trouvée étape 1) par la cardinalité de cette source
 - Choisir le poids maximum (ou l'un des poids maximum)
- Exemple du SI Cinéma


▪ <nfilm>	→ 1
▪ <npersonne>	→ 1
▪ <npersonne, nfilm>	→ 2



ETAPE 3 : CONSTRUIRE UNE RELATION EN 3-FN



- Rappel : Une relation en 3-FN pour un identifiant, est une relation dont les attributs
 - qui ne font pas partie de l'identifiant
 - sont en **Dépendance Fonctionnelle Élémentaire et Directe** de cet identifiant
- L'étape 2 nous donne un identifiant, en regroupant les sommets directement accessibles de cet identifiant nous obtenons par construction, une relation 3-FN
- Exemple : SI Cinéma
 - R1 (npersonne, nfilm, salaire_acteur)


 Sources de la DFED

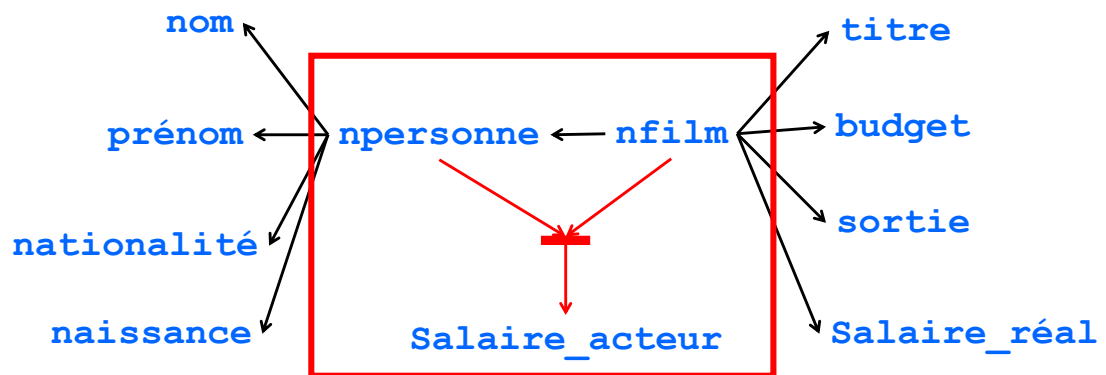


ETAPE 4 : "NETTOYER" LE GRAPHE

- Afin que l'algorithme converge, il nous faut éliminer
 - les arcs utilisés par l'étape 3
 - Puis les sommets isolés



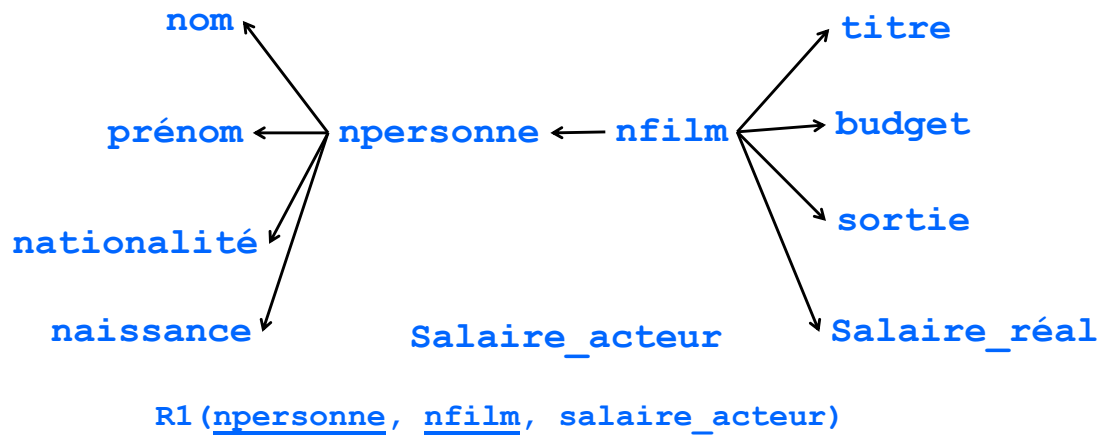
ETAPE 4 : SUPPRIMER LES ARCS UTILISÉS PAR L'ÉTAPE 3



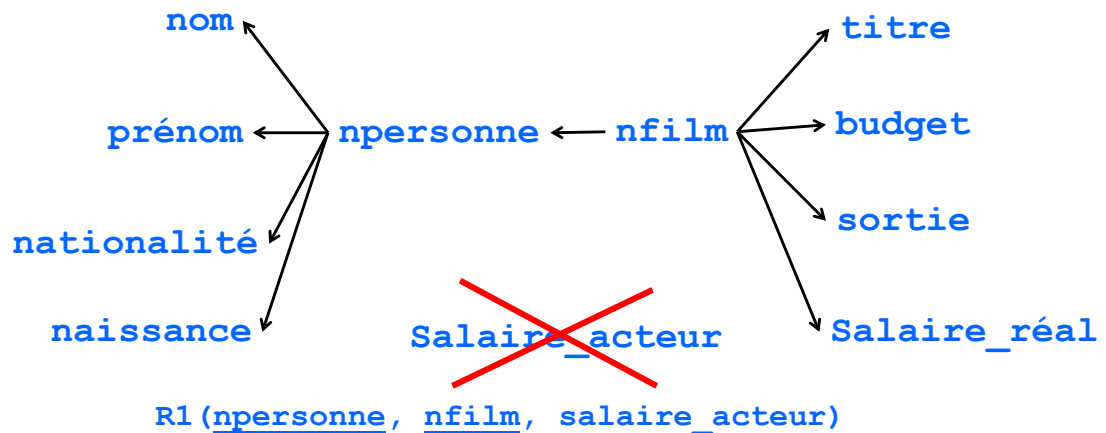
$R1(\underline{npersonne}, \underline{nfilm}, \text{salaire_acteur})$



ETAPE 4 : SUPPRIMER LES ARCS UTILISÉS PAR L'ÉTAPE 3



ETAPE 4 : SUPPRIMER LES SOMMETS ISOLÉS

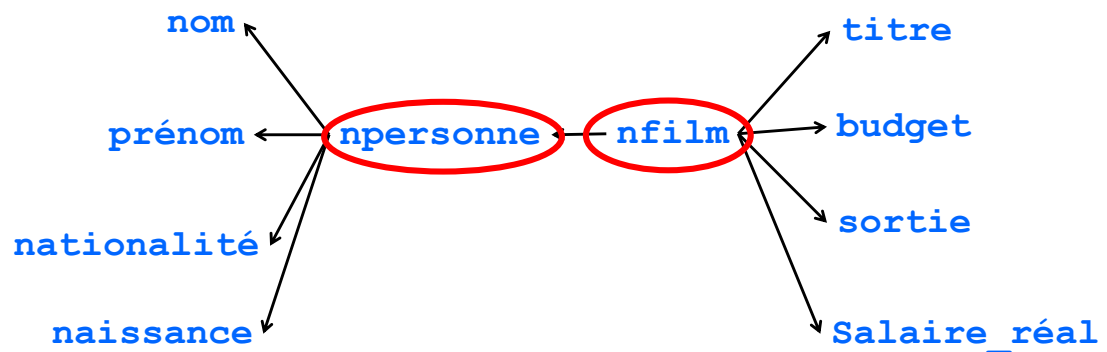


ITÉRATIONS 2 ET 3

- Il nous reste deux sources de DFED de poids 1
 - <npersonne>
 - <nfilm>
- Itération 2 : choix <nfilm>
 - Relation R2(nfilm, titre, budget, sortie, salaire_real, npersonne)
- Itération 3 : choix <npersonne>
 - Relation R3(npersonne, nom, prénom, nationalité, naissance)



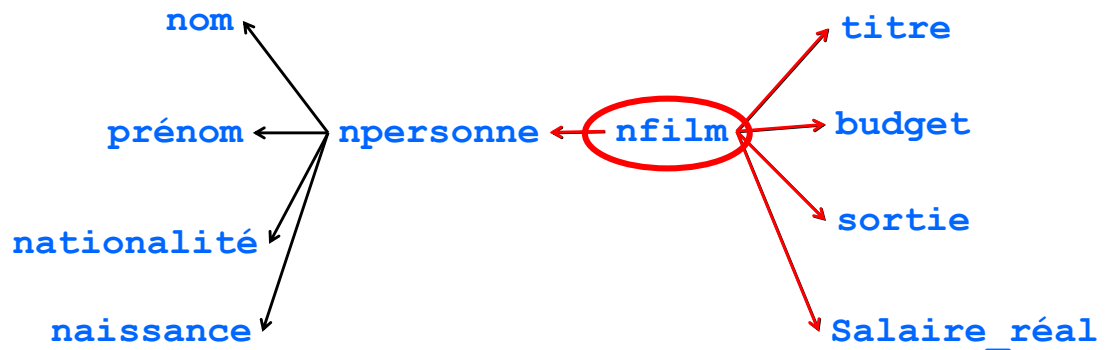
2ÈME ITÉRATION



R1(npersonne, nfilm, salaire_acteur)



2^{ÈME} ITÉRATION : EDITION DE LA RELATION

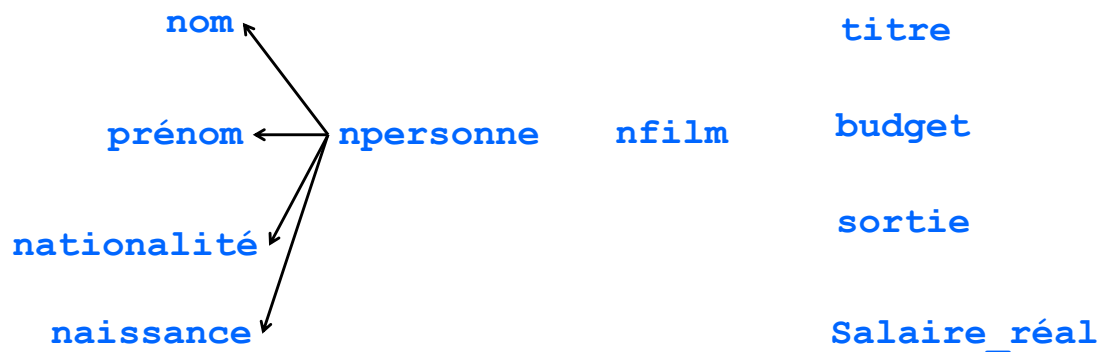


R1(npersonne, nfilm, salaire_acteur)

R2(nfilm, titre, budget, sortie, salaire_real, npersonne)



2^{ÈME} ITÉRATION : SUPPRESSION DES ARCS

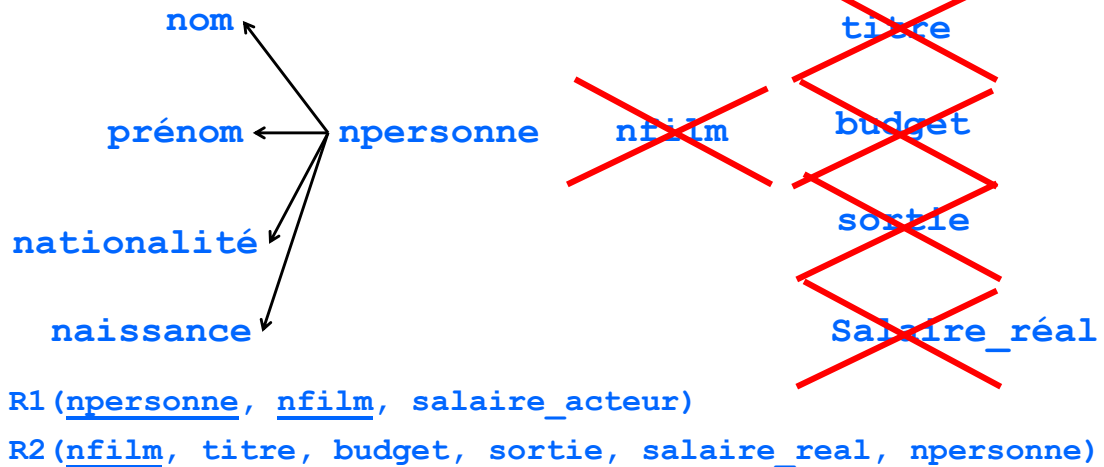


R1(npersonne, nfilm, salaire_acteur)

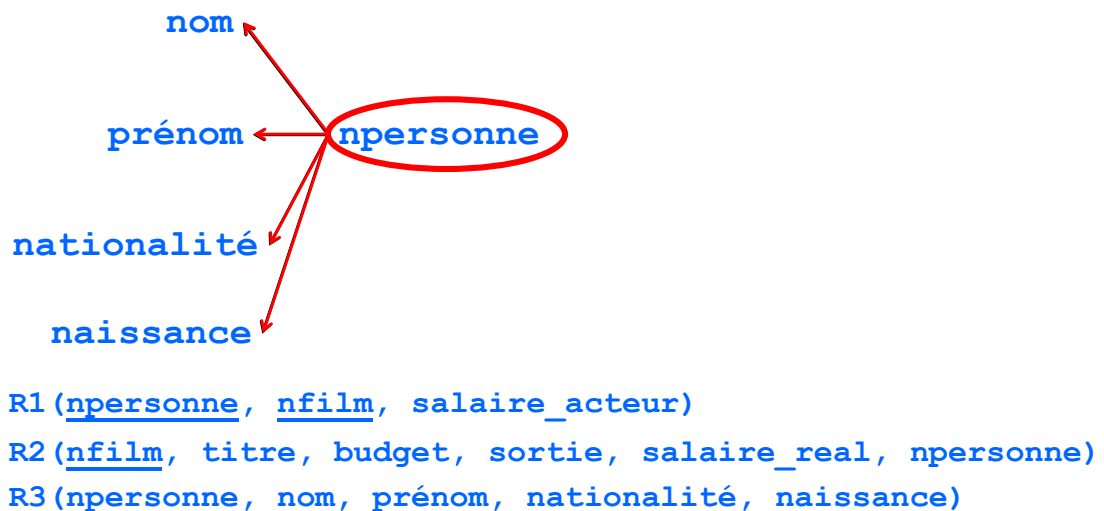
R2(nfilm, titre, budget, sortie, salaire_real, npersonne)



2^{ÈME} ITÉRATION : SUPPRESSION DES SOMMETS ISOLÉS



3^{ÈME} ITÉRATION : ÉDITION DE LA RELATION



MISE EN FORME DU RÉSULTAT

- Renommer les relations
- Renommer certain(s) attribut(s)
- Exemple SI Cinéma

▪ **FILM** (nfilm, titre, budget, sortie, nrealisateur#, salaire_real)

▪ **PERSONNE** (npersonne, nom, prenom, nationalite, naissance)

▪ **CASTING** (npersonne#, nfilm#, salaire_acteur)



LE MODÈLE RELATIONNEL ET MODÈLE UML

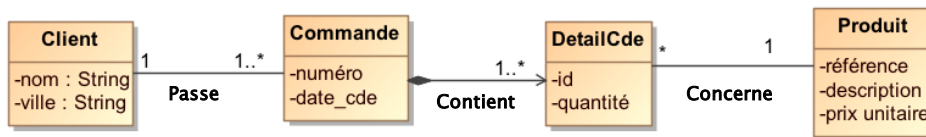


MODÉLISATION DU S.I.

► Description simplifiée d'un S.I.

- Le client Martin habitant Lille commande le 8 mars 2010 une imprimante de marque Epson coûtant 59.99 € et 4 cartouches d'encre coûtant 9.99 € pièce.

► Modélisation UML 2.x



MODÉLISATION DU S.I.

► Modèle relationnel

- CLIENT (numéro_client, nom, ville)
 - COMMANDE (numéro_cde, numéro_client#, date_commande)
 - DETAIL_COMMANDE (id, numéro_cde#, référence_produit#, quantité)
 - PRODUIT (référence, description, prix_unitaire)
- Blue arrows indicate foreign key relationships: from 'numéro_client#' in COMMANDE to 'numéro_client' in CLIENT; from 'numéro_cde#' in DETAIL_COMMANDE to 'numéro_cde' in COMMANDE; from 'référence_produit#' in DETAIL_COMMANDE to 'référence' in PRODUIT.



COMPARAISON DES DEUX MODÈLES

Modèle	Avantages	Inconvénients
UML 2.x	<ul style="list-style-type: none"> . Communication aisée (car graphique) . Quelques règles de gestion (multiplicité des associations) 	Redondances d'information si on y prend pas garde
Relationnel	<ul style="list-style-type: none"> . Détection et élimination des redondances d'information (alg. de normalisation) . Modèle orienté "informatique" (bien adapté aux BD relationnelles) 	Communication difficile (illisible ☹)



COMPARAISON DES DEUX MODÈLES

- ▶ Les deux modèles ne sont pas équivalents mais **COMPLEMENTAIRES**
- ▶ Le premier modèle à utiliser est le modèle UML 2.x car il est proche des "clients"
- ▶ Le second modèle est utilisé essentiellement par les informaticiens devant mettre en place le modèle physique de données (cf. séance 5)
 - Bases de données
 - Applications
 - ...



CE QUE L'ON A ABORDÉ DURANT CETTE SÉANCE

- ▶ Le modèle relationnel (*pour définir les bases*)
 - Relation, domaine, attributs, identifiant
- ▶ Les dépendances fonctionnelles (*pour caractériser les dépendances entre attributs*)
 - Dépendance fonctionnelle
 - Dépendance fonctionnelle élémentaire
 - Dépendance fonctionnelle élémentaire directe
- ▶ Les formes normales (*pour caractériser des relations réduisant les redondances et facilitant les mises à jour*)
 - Première, deuxième et troisième forme normales
- ▶ Une méthode de normalisation (*pour passer à un ensemble de relations en 3-FN*)
 - Basée sur le graphe des dépendances fonctionnelles

