

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

généralisation /
spécialisation

Spécification des
diagrammes EA

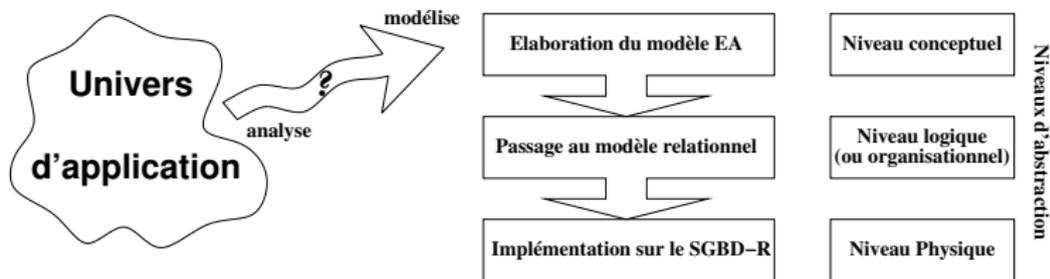
Contraintes
d'intégrité

Normalisation
d'une relation

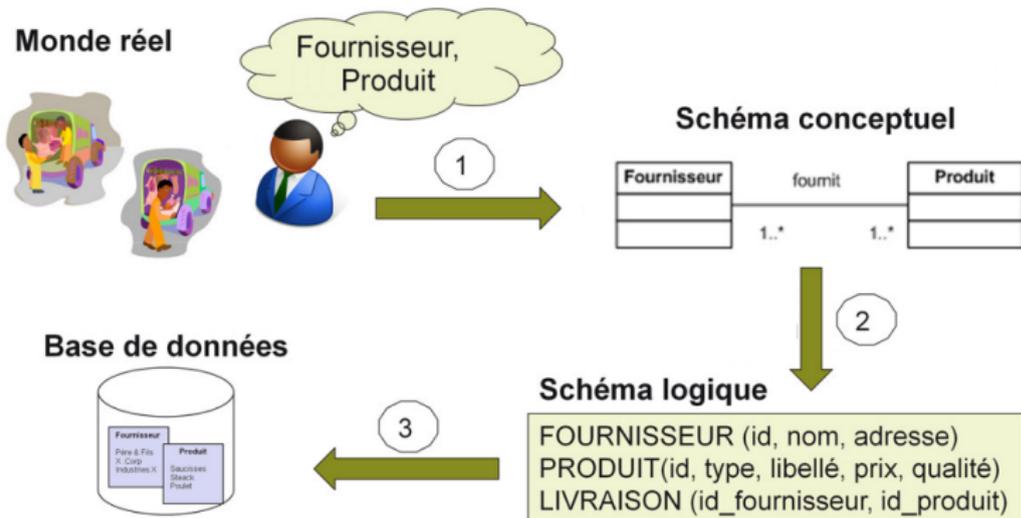
- 1 Les concepts généraux
- 2 Notions essentielles pour le modèle relationnel
- 3 Algèbre Relationnelle et SQL
- 4 Le modèle conceptuel Entités-Associations
 - Concepts de base et Diagramme EA
 - Représentations multiples : généralisation/spécialisation
 - La spécification des diagrammes EA
 - Contraintes d'intégrité
- 5 Normalisation d'une relation

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsConcepts de base et
Diagramme EAgénéralisation /
spécialisationSpécification des
diagrammes EAContraintes
d'intégritéNormalisation
d'une relation

- Les BD constituent le cœur du système d'information.
- La conception de ces bases de données est la tâche la plus ardue du processus de développement du système d'information. :
 - conception bien gérée ⇒ développement facile.
 - si des difficultés non révélées ⇒ la phase de développement sera obligée de reprendre des points non éclaircis lors de la phase de conception. De nombreux bugs verront le jour.
- Méthodes de conception pour assister la démarche de conception.
3 étapes principales ~ 3 niveaux d'abstraction différents :



- Les BD constituent le cœur du système d'information.
- La conception de ces bases de données est la tâche la plus ardue du processus de développement du système d'information. :
 - conception bien gérée \Rightarrow développement facile.
 - si des difficultés non révélées \Rightarrow la phase de développement sera obligée de reprendre des points non éclaircis lors de la phase de conception. De nombreux bugs verront le jour.
- Méthodes de conception pour assister la démarche de conception.
3 étapes principales \sim 3 niveaux d'abstraction différents :



Introduction aux BDR

Jean-Paul Comet,
Nadia Abchiche-Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

le modèle Entités-Associations

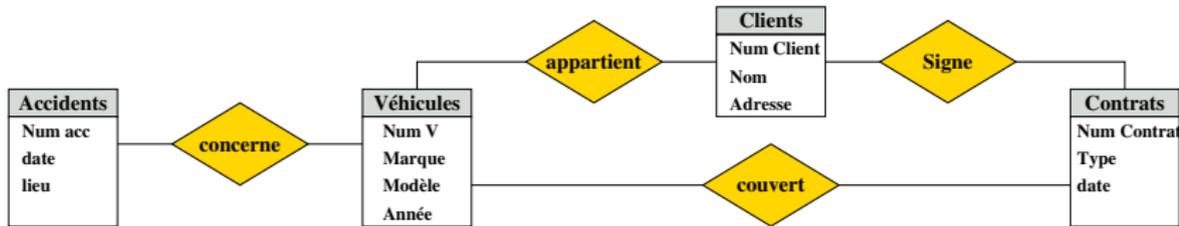
Concepts de base et Diagramme EA

généralisation / spécialisation

Spécification des diagrammes EA

Contraintes d'intégrité

Normalisation d'une relation



- Le modèle **entités-associations** (EA), aussi appelé **entités-relations** (ER) est un modèle de données de type conceptuel.
- **But** : **exprimer une partie du monde réel indépendamment de toute contingence informatique.**
Utilisé dans des méthodes et outils d'aide à la conception (Merise, par ex.).
Se limite à une description statique : son but est d'aider à la conception des structures de données d'une application.
- **Idée sous-jacente** : représentation des **entités** du monde réel que l'observateur peut identifier de manière distincte.

On distingue alors

- les **entités**
(personne, voitures, contracts, accidents...),
- les propriétés observables de ces entités, appelées **attributs**
(taille, couleur, date de signature, lieu de l'accident...),
- et les **associations** qui relient les entités
(une personne est propriétaire d'une voiture).

- Les **entités** représentent des objets du monde réel, (concrets ou abstraits)
- Les **types d'entités** représentent les ensembles d'entités perçues comme similaires et ayant des caractéristiques comparables.

Clients	Véhicules	Contrats	Accidents
Num Client Nom Adresse	Num V Marque Modèle Année	Num Contrat Type date	Num acc date lieu

- **occurrence** d'un type d'entités / **population** d'un type d'entités.
- Une occurrence d'un type d'entités est vue par l'utilisateur comme un ensemble de valeurs : une valeur est associée à chacun des attributs
- Les **attributs** : représentations des propriétés associées à un type d'entités.
- **attribut composé** \equiv attribut décomposé en attributs plus simples.
Exemple : date \equiv jour + mois + année

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

généralisation /
spécialisation

Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation



Les **associations** représentent les liens existants entre les entités. Les associations sont caractérisées, comme les entités, par un nom et éventuellement des attributs.

- **Types d'associations** : ensembles d'associations perçues comme similaires et ayant des caractéristiques qui peuvent être comparées.
- **Occurrences** d'un type d'associations : instances d'un type d'associations.
- **Population** d'un type d'associations : ensemble d'occurrences d'un type d'associations.

Cardinalité : la description complète d'une association nécessite la définition précise de la participation des entités. La cardinalité est le nombre de participation d'une entité à une association.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

généralisation /
spécialisation

Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

- **Cardinalité un à un :**

- **Cardinalité un à plusieurs :**

- **Cardinalité plusieurs à plusieurs :**

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

généralisation /
spécialisation

Spécification des
diagrammes EA

Contraintes
d'intégrité

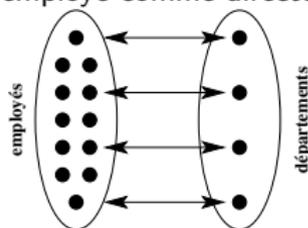
Normalisation
d'une relation

- **Cardinalité un à un** : un employé ne peut être directeur que dans un seul département et un département n'a qu'un seul employé comme directeur.

- **Cardinalité un à plusieurs** :

- **Cardinalité plusieurs à plusieurs** :

- **Cardinalité un à un** : un employé ne peut être directeur que dans un seul département et un département n'a qu'un seul employé comme directeur.



- **Cardinalité un à plusieurs** :
- **Cardinalité plusieurs à plusieurs** :

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

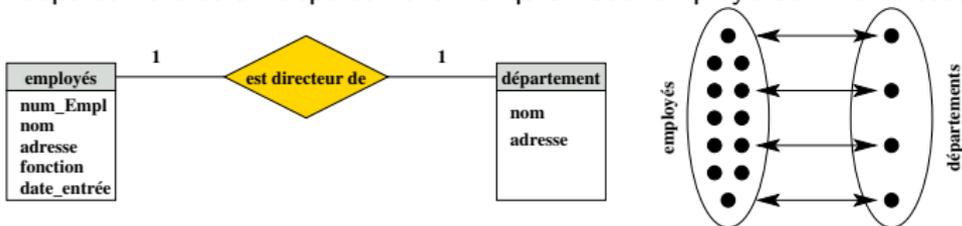
généralisation /
spécialisation

Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

- **Cardinalité un à un** : un employé ne peut être directeur que dans un seul département et un département n'a qu'un seul employé comme directeur.



- **Cardinalité un à plusieurs** : un département emploie plusieurs personnes (employés) mais chaque employé ne fait partie que d'un seul département.

- **Cardinalité plusieurs à plusieurs** :

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

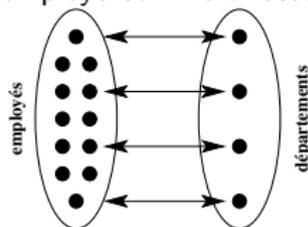
généralisation /
spécialisation

Spécification des
diagrammes EA

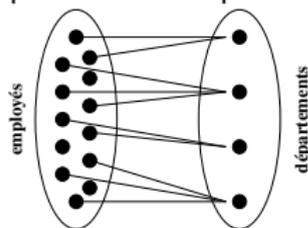
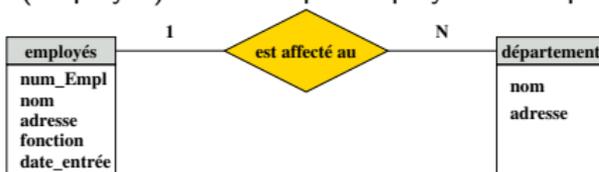
Contraintes
d'intégrité

Normalisation
d'une relation

- **Cardinalité un à un** : un employé ne peut être directeur que dans un seul département et un département n'a qu'un seul employé comme directeur.

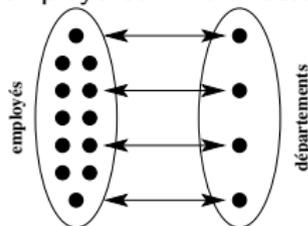


- **Cardinalité un à plusieurs** : un département emploie plusieurs personnes (employés) mais chaque employé ne fait partie que d'un seul département.

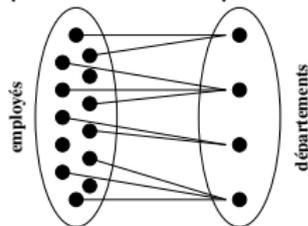


- **Cardinalité plusieurs à plusieurs** :

- **Cardinalité un à un** : un employé ne peut être directeur que dans un seul département et un département n'a qu'un seul employé comme directeur.

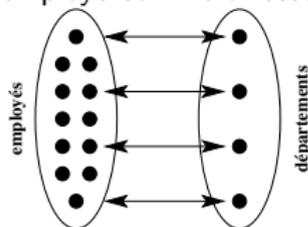


- **Cardinalité un à plusieurs** : un département emploie plusieurs personnes (employés) mais chaque employé ne fait partie que d'un seul département.

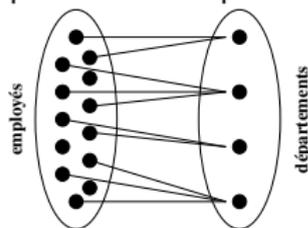


- **Cardinalité plusieurs à plusieurs** : un type de produit peut être fabriqué dans plusieurs usines et une usine donnée peut fabriquer plusieurs produits.

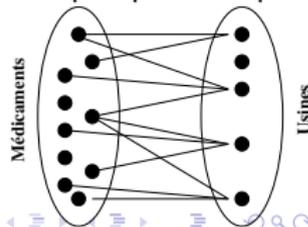
- **Cardinalité un à un** : un employé ne peut être directeur que dans un seul département et un département n'a qu'un seul employé comme directeur.



- **Cardinalité un à plusieurs** : un département emploie plusieurs personnes (employés) mais chaque employé ne fait partie que d'un seul département.



- **Cardinalité plusieurs à plusieurs** : un type de produit peut être fabriqué dans plusieurs usines et une usine donnée peut fabriquer plusieurs produits.



Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentielsAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsConcepts de base et
Diagramme EAgénéralisation /
spécialisationSpécification des
diagrammes EAContraintes
d'intégritéNormalisation
d'une relation

Cardinalités maximales : nb max de participations d'une entité à une association.

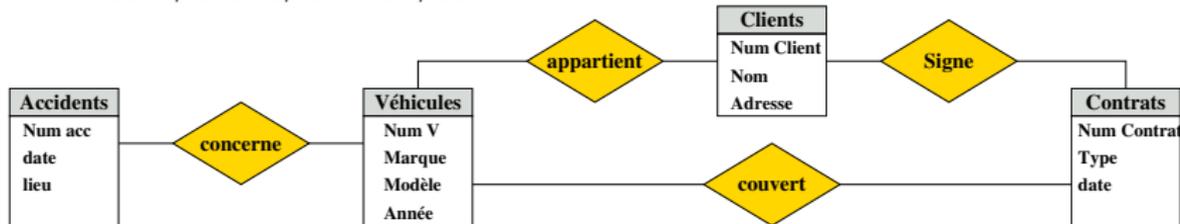
Cardinalités minimales : nb min de participations d'une entité à une association (souvent 0 ou 1).

Les cardinalités max et min traduisent les contraintes propres aux entités et associations.

Notations :

0-1	aucune ou une seule
1-1	une et une seule
0-N	aucune ou plusieurs
1-N	une ou plusieurs

Exercice : Faire le schéma Entités-Associations correspondant aux entités Véhicules, Clients, Contrats, accidents.



Cardinalités maximales : nb max de participations d'une entité à une association.

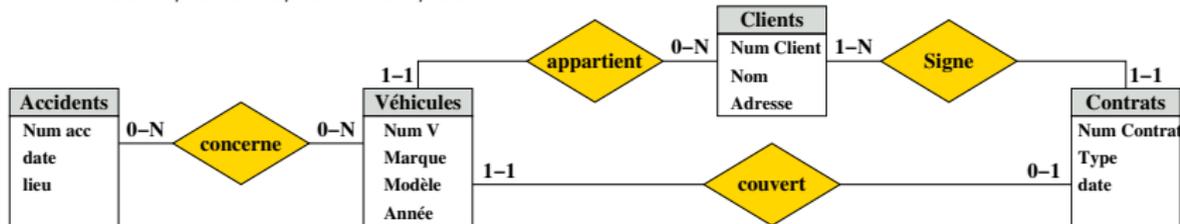
Cardinalités minimales : nb min de participations d'une entité à une association (souvent 0 ou 1).

Les cardinalités max et min traduisent les contraintes propres aux entités et associations.

Notations :

0-1	aucune ou une seule
1-1	une et une seule
0-N	aucune ou plusieurs
1-N	une ou plusieurs

Exercice : Faire le schéma Entités-Associations correspondant aux entités Véhicules, Clients, Contrats, accidents.



Notion de rôles.

- Une association permet de lier plusieurs entités entre elles.
- Chaque entité va donc jouer un **rôle** vis à vis de la sémantique de l'association considérée.
- Si un type d'associations est relié à un même type d'entités par plusieurs liens différents, le rôle de chacun des types d'entités est primordial (indispensable), car c'est lui qui va distinguer les différents cas possibles.

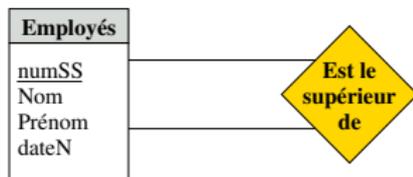
Exemple : les liens hiérarchiques dans une entreprise

2 rôles :

- un rôle pour exprimer le chef.
- un rôle pour exprimer le ou les subordonnés,

Les cardinalités :

- 0-N pour le rôle « *chef* »,
- 0-1 pour le rôle « *subordonné* ».



Une **occurrence d'un type d'associations** est vue par l'utilisateur comme

- un ensemble de valeurs : une valeur pour chaque attribut du type d'associations (s'il en existe)
- un ensemble d'occurrences de types d'entités : pour chaque rôle associé au type d'associations, une occurrence du type d'entités qui joue ce rôle.

Notion de rôles.

- Une association permet de lier plusieurs entités entre elles.
- Chaque entité va donc jouer un **rôle** vis à vis de la sémantique de l'association considérée.
- Si un type d'associations est relié à un même type d'entités par plusieurs liens différents, le rôle de chacun des types d'entités est primordial (indispensable), car c'est lui qui va distinguer les différents cas possibles.

Exemple : les liens hiérarchiques dans une entreprise
2 rôles :

- un rôle pour exprimer le chef.
- un rôle pour exprimer le ou les subordonnés,

Les cardinalités :

- 0-N pour le rôle « *chef* »,
- 0-1 pour le rôle « *subordonné* ».



Une **occurrence d'un type d'associations** est vue par l'utilisateur comme

- un ensemble de valeurs : une valeur pour chaque attribut du type d'associations (s'il en existe)
- un ensemble d'occurrences de types d'entités : pour chaque rôle associé au type d'associations, une occurrence du type d'entités qui joue ce rôle.

Notion de rôles.

- Une association permet de lier plusieurs entités entre elles.
- Chaque entité va donc jouer un **rôle** vis à vis de la sémantique de l'association considérée.
- Si un type d'associations est relié à un même type d'entités par plusieurs liens différents, le rôle de chacun des types d'entités est primordial (indispensable), car c'est lui qui va distinguer les différents cas possibles.

Exemple : les liens hiérarchiques dans une entreprise
2 rôles :

- un rôle pour exprimer le chef.
- un rôle pour exprimer le ou les subordonnés,

Les cardinalités :

- 0-N pour le rôle « *chef* »,
- 0-1 pour le rôle « *subordonné* ».



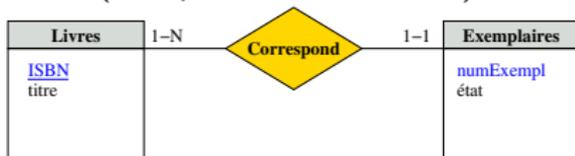
Une **occurrence d'un type d'associations** est vue par l'utilisateur comme

- un ensemble de valeurs : une valeur pour chaque attribut du type d'associations (s'il en existe)
- un ensemble d'occurrences de types d'entités : pour chaque rôle associé au type d'associations, une occurrence du type d'entités qui joue ce rôle.

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsConcepts de base et
Diagramme EAgénéralisation /
spécialisationSpécification des
diagrammes EAContraintes
d'intégritéNormalisation
d'une relation

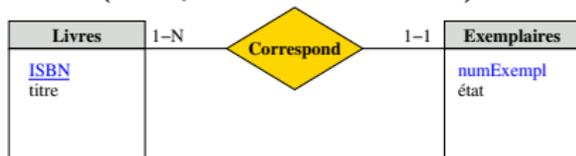
- Un **identifiant** est un attribut ou un *ensemble d'attributs* permettant de déterminer une et une seule entité à l'intérieur de la population.
 - Un **identifiant** est un ensemble minimum d'attributs tel qu'il n'existe pas deux occurrences du TE qui ont la même valeur pour ces attributs.
- ⇒ s'il y a un identifiant composé d'au moins 2 attributs, bien le distinguer de deux identifiants simples.

Exemple. Dans une bibliothèque, on distingue le livre (œuvre protégée) de l'exemplaire de cet œuvre (exemplaire numéro 1, 2 ...).



- Un **identifiant** est un attribut ou un *ensemble d'attributs* permettant de déterminer une et une seule entité à l'intérieur de la population.
 - Un **identifiant** est un ensemble minimum d'attributs tel qu'il n'existe pas deux occurrences du TE qui ont la même valeur pour ces attributs.
- ⇒ s'il y a un identifiant composé d'au moins 2 attributs, bien le distinguer de deux identifiants simples.

Exemple. Dans une bibliothèque, on distingue le livre (œuvre protégée) de l'exemplaire de cet œuvre (exemplaire numéro 1, 2 ...).



- on dit que le TE *Exemplaires* est **faible**.
- Un TE est dit **faible** lorsque :
 - aucun sous-ensemble de ses attributs constitue un identifiant, et
 - un identifiant peut être défini en intégrant un identifiant d'un autre TE' qui lui est lié par un TA binaire de cardinalité (1,1) pour TE.On dit dans ce cas que TE **dépend** de TE'.
- le TE *Exemplaires* dépend du TE *Livres*, car tout exemplaire est rattaché à un livre.
L'identifiant est (**Livres.ISBN, numExempl**).

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

généralisation /
spécialisation

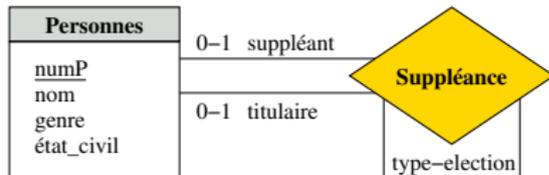
Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

La définition des identifiants est la même que pour les types d'entités.

Exemples.



- Une occurrence du TA Suppléance est un triplet.
numP est un identifiant de Personnes, la cardinalité des deux rôles est 0-1

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

généralisation /
spécialisation

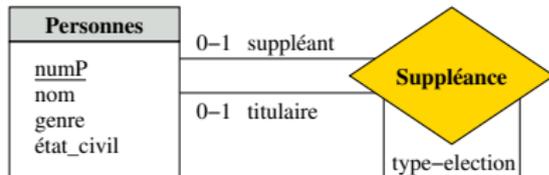
Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

La définition des identifiants est la même que pour les types d'entités.

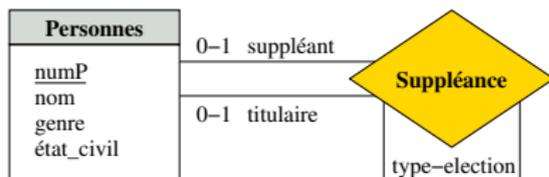
Exemples.



- Une occurrence du TA Suppléance est un triplet.
numP est un identifiant de Personnes, la cardinalité des deux rôles est 0-1
⇒ Suppléance possède 2 identifiants :
(suppléant.numP, type_election) et
(titulaire.numP, type_election).

La définition des identifiants est la même que pour les types d'entités.

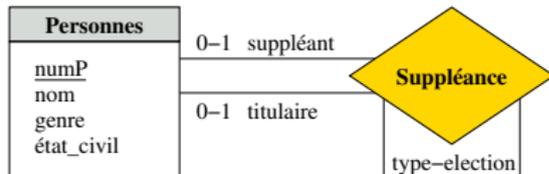
Exemples.



- Une occurrence du TA Suppléance est un triplet.
numP est un identifiant de Personnes, la cardinalité des deux rôles est 0-1
⇒ Suppléance possède 2 identifiants :
(suppléant.numP, type_election) et
(titulaire.numP, type_election).
- Ce raisonnement est bon uniquement si la base de données ne contient que les suppléances en cours. Si on mémorisait toutes les suppléances, les cardinalités des rôles suppléant et titulaire seraient 0-N, et les identifiants de Suppléance seraient :

La définition des identifiants est la même que pour les types d'entités.

Exemples.



- Une occurrence du TA Suppléance est un triplet.
numP est un identifiant de Personnes, la cardinalité des deux rôles est 0-1
⇒ Suppléance possède 2 identifiants :
(suppléant.numP, type_election) et
(titulaire.numP, type_election).
- Ce raisonnement est bon uniquement si la base de données ne contient que les suppléances en cours. Si on mémorisait toutes les suppléances, les cardinalités des rôles suppléant et titulaire seraient 0-N, et les identifiants de Suppléance seraient :
(suppléant.numP, type_election, date) et
(titulaire.numP, type_election, date).

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

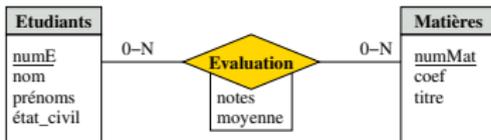
généralisation /
spécialisation

Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

Exercice : Donner les identifiants des types d'associations suivants.



Correction :

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

généralisation /
spécialisation

Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

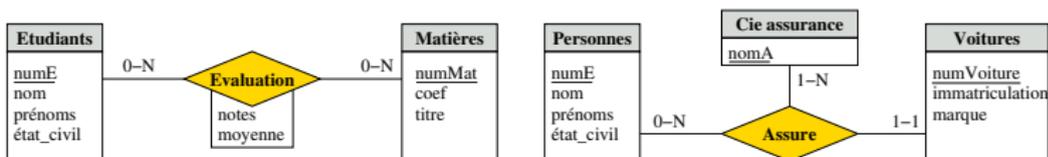
Exercice : Donner les identifiants des types d'associations suivants.



Correction :

Dans le 1er cas, l'identifiant est (`Etudiant.numE, Matière.numMat`).

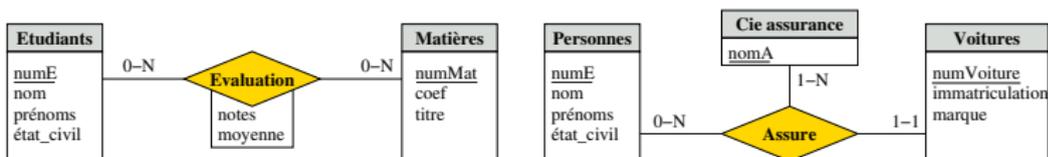
Exercice : Donner les identifiants des types d'associations suivants.



Correction :

Dans le 1er cas, l'identifiant est (**Etudiant.numE, Matière.numMat**).

Exercice : Donner les identifiants des types d'associations suivants.



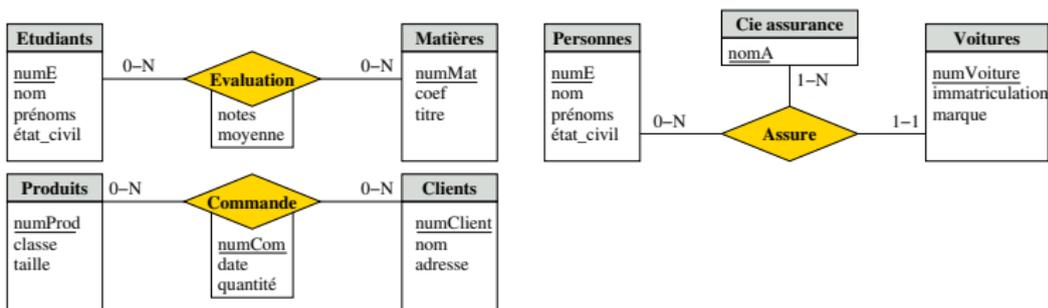
Correction :

Dans le 1er cas, l'identifiant est **(Etudiant.numE, Matière.numMat)**.

Dans le 2ème cas, l'identifiant est **Voitures.numVoiture**.

Règle 1. Lorsqu'un type associations est lié à un TE par un lien de cardinalité maximum égale à 1, alors tout identifiant de ce TE fait partie de l'identifiant du TA considéré.

Exercice : Donner les identifiants des types d'associations suivants.



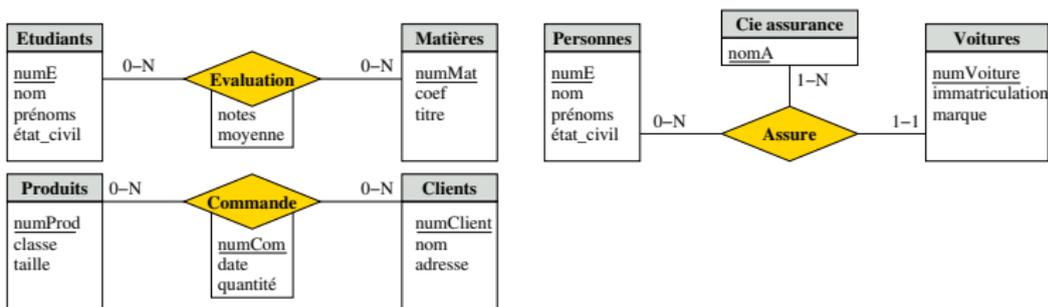
Correction :

Dans le 1er cas, l'identifiant est **(Etudiant.numE, Matière.numMat)**.

Dans le 2ème cas, l'identifiant est **Voitures.numVoiture**.

Règle 1. Lorsqu'un type associations est lié à un TE par un lien de cardinalité maximum égale à 1, alors tout identifiant de ce TE fait partie de l'identifiant du TA considéré.

Exercice : Donner les identifiants des types d'associations suivants.



Correction :

Dans le 1er cas, l'identifiant est **(Etudiant.numE, Matière.numMat)**.

Dans le 2ème cas, l'identifiant est **Voitures.numVoiture**.

Dans le 3ème cas, l'identifiant est

(Produit.numProd, numCom, Client.numClient).

Règle 1. Lorsqu'un type associations est lié à un TE par un lien de cardinalité maximum égale à 1, alors tout identifiant de ce TE fait partie de l'identifiant du TA considéré.

Règle 2. Lorsque plusieurs occurrences d'un TA peuvent lier les mêmes occurrences des TE, alors l'identifiant du TA contient au moins un attribut du TA.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

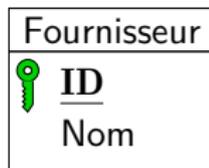
Concepts de base et
Diagramme EA

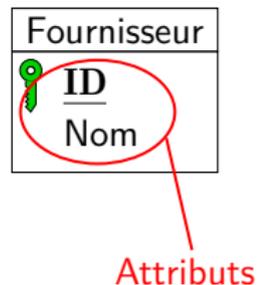
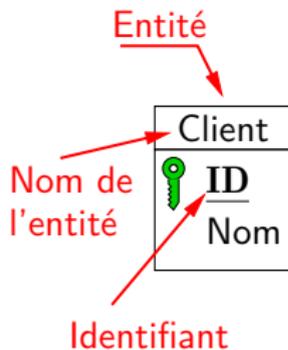
généralisation /
spécialisation

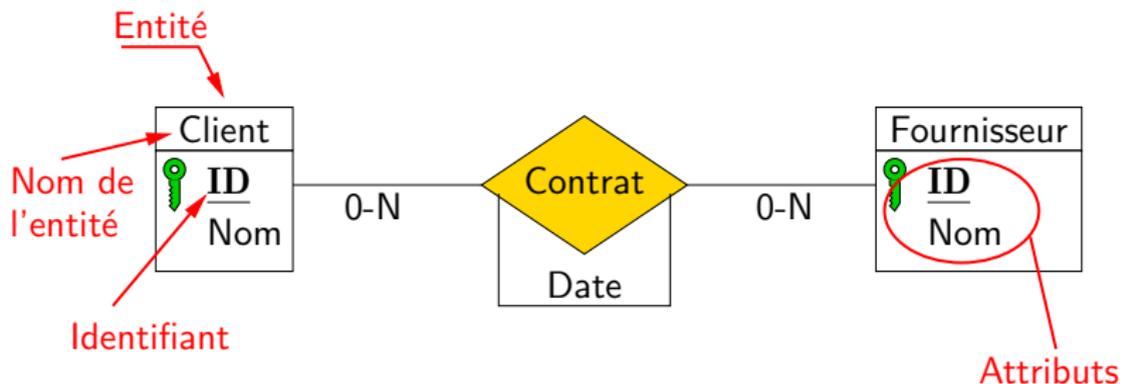
Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation



Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsConcepts de base et
Diagramme EAgénéralisation /
spécialisationSpécification des
diagrammes EAContraintes
d'intégritéNormalisation
d'une relation

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsConcepts de base et
Diagramme EAgénéralisation /
spécialisationSpécification des
diagrammes EAContraintes
d'intégritéNormalisation
d'une relation

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

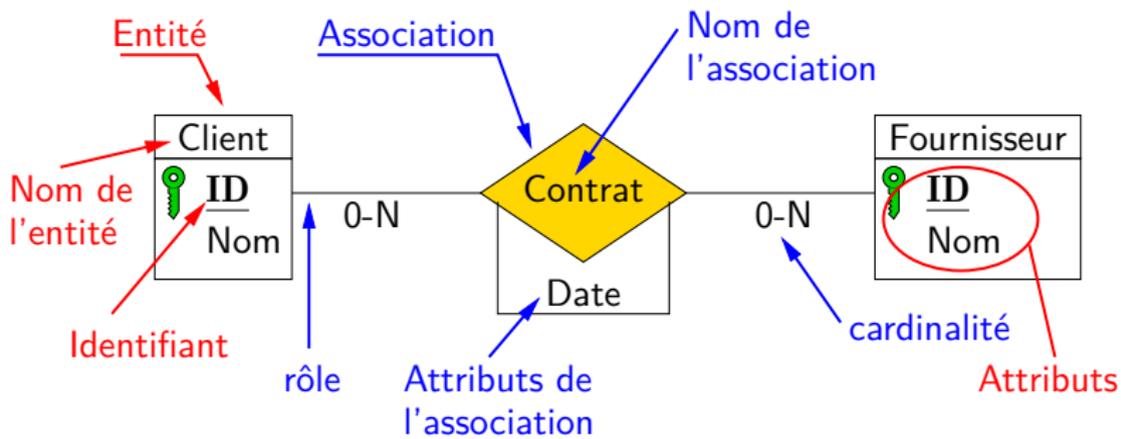
Concepts de base et
Diagramme EA

généralisation /
spécialisation

Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation



Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

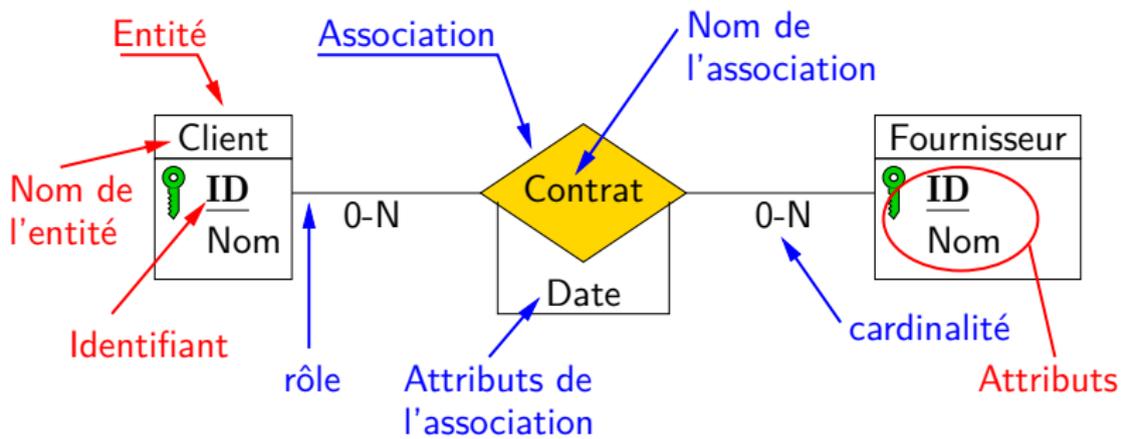
Concepts de base et
Diagramme EA

généralisation /
spécialisation

Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation



Association : Contrat
Dimension (arité) = 2

Les entités Client et Fournisseur
sont les participants de
l'association Contrat

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

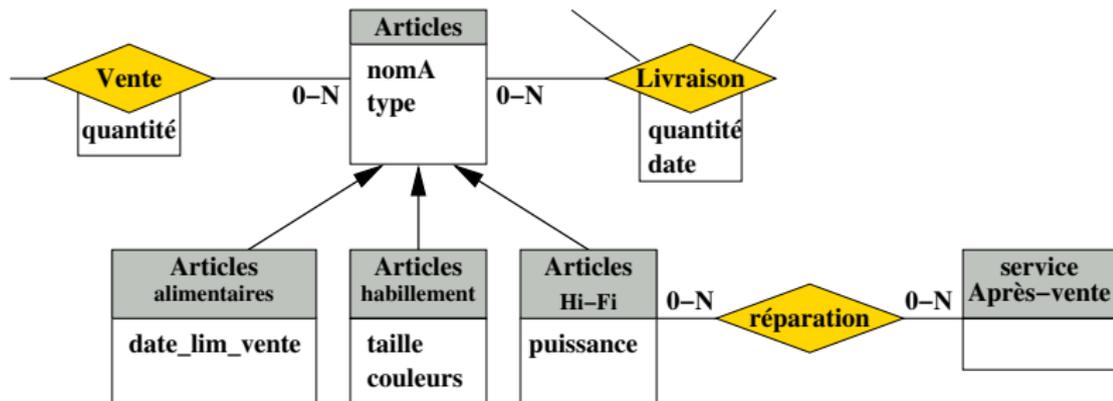
**généralisation/
spécialisation**

Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

- Un type d'entités représente une classe d'objets du monde réel.
 - Ce même ensemble peut être perçu d'un autre point de vue comme plusieurs classes. **Exemple** : articles d'un supermarché
 - pour certains, tous les articles ont le même statut,
 - pour d'autres, articles d'alimentation, articles d'habillement... (traitements différents)
 - Comment définir un TE générique `Article`, et des TE spécialisés représentant des sous-classes spécialisées ?
Les classes d'objets *ne sont donc plus distinctes*.
- 1 **Généralisation** : TE spécialisé (*spécifique*) \longrightarrow TE générique
Spécialisation : TE générique \longrightarrow TE spécialisé (*spécifique*)
« à toute entité d'un TE spécialisé correspond une et une seule entité du TE générique »
« à toute entité du TE générique correspond au plus une entité d'un TE spécialisé »
Lien « est un » (ou « IS A »)
le TE spécialisé est appelé **sous-type** du TE générique
le TE générique est appelé **sur-type** du TE spécialisé.
Lorsqu'il y a plusieurs niveaux, **hiérarchie de généralisations**.
- 2 les attributs qui sont communs à tous les sous-types, sont rattachés au type d'entités générique : les sous-types **héritent** de ces attributs.



- Il n'est pas nécessaire que l'ensemble des sous-types constituent une partition du sur-type.
- Les TE spécifiques peuvent représenter des populations qui ne sont pas disjointes.
- **généralisation multiple** lorsqu'un TE spécialisé est sous-type de plusieurs autres TE. (Cf exemple suivant)
- **Identifiant d'un TE sous-type.** Soit E un TE sous-type d'un TE E'. Tout identifiant de E' est aussi identifiant de E. E n'a pas nécessairement d'identifiant qui lui soit propre.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

**généralisation /
spécialisation**

Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

Exercice : Décrire par un diagramme EA l'ensemble des personnes travaillant dans une université. Chaque individu est une personne, la population des personnes est partagée en étudiants et employés. La population des personnels est découpée en trois : les enseignants (MCF, Pr, et autres), les techniciens et les administratifs. Parmi les étudiants, on particularise les doctorants, parmi lesquels certains sont moniteurs, c'est-à-dire héritent de la population des enseignants.

Les attributs associés aux personnes sont : n^o d'identifiant, nom, prénom. L'étudiant a comme attribut le dépt, le doctorant a un sujet de thèse, et l'employé a pour attribut sa classe (A,B,C).

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

**généralisation /
spécialisation**

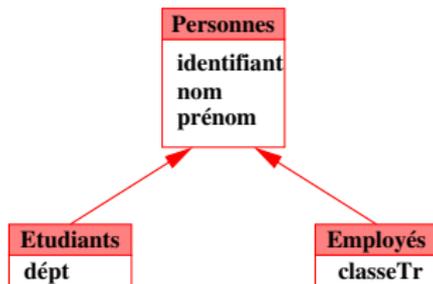
Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

Exercice : Décrire par un diagramme EA l'ensemble des personnes travaillant dans une université. **Chaque individu est une personne, la population des personnes est partagée en étudiants et employés.** La population des personnels est découpée en trois : les enseignants (MCF, Pr, et autres), les techniciens et les administratifs. Parmi les étudiants, on particularise les doctorants, parmi lesquels certains sont moniteurs, c'est-à-dire héritent de la population des enseignants.

Les attributs associés aux personnes sont : n^o d'identifiant, nom, prénom. L'étudiant a comme attribut le dépt, le doctorant a un sujet de thèse, et l'employé a pour attribut sa classe (A,B,C).



Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

**généralisation /
spécialisation**

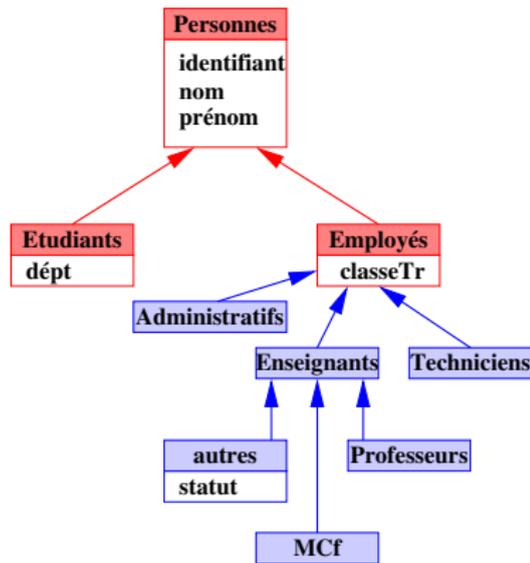
Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

Exercice : Décrire par un diagramme EA l'ensemble des personnes travaillant dans une université. **Chaque individu est une personne, la population des personnes est partagée en étudiants et employés.** La population des personnels est découpée en trois : les enseignants (MCF, Pr, et autres), les techniciens et les administratifs. Parmi les étudiants, on particularise les doctorants, parmi lesquels certains sont moniteurs, c'est-à-dire héritent de la population des enseignants.

Les attributs associés aux personnes sont : n^o d'identifiant, nom, prénom. L'étudiant a comme attribut le dépt, le doctorant a un sujet de thèse, et l'employé a pour attribut sa classe (A,B,C).



Introduction aux BDR

Jean-Paul Comet,
Nadia Abchiche-Mimouni

Les concepts généraux

Notions essentielles

Algèbre Relationnelle + SQL

le modèle Entités-Associations

Concepts de base et Diagramme EA

généralisation / spécialisation

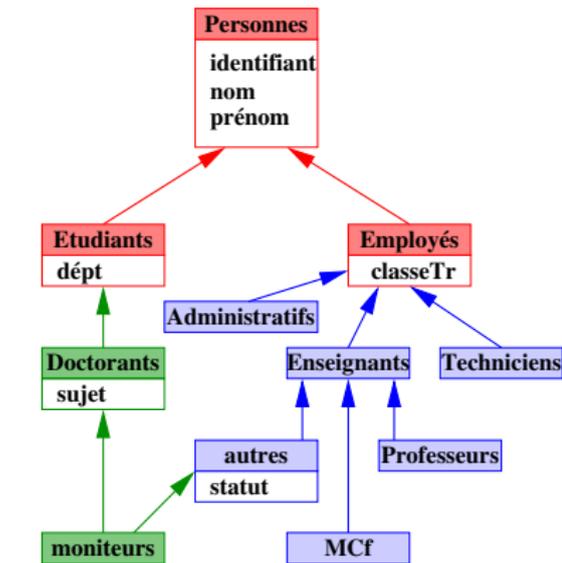
Spécification des diagrammes EA

Contraintes d'intégrité

Normalisation d'une relation

Exercice : Décrire par un diagramme EA l'ensemble des personnes travaillant dans une université. **Chaque individu est une personne, la population des personnes est partagée en étudiants et employés.** La population des personnels est découpée en trois : les enseignants (MCF, Pr, et autres), les techniciens et les administratifs. Parmi les étudiants, on particularise les doctorants, parmi lesquels certains sont moniteurs, c'est-à-dire héritent de la population des enseignants.

Les attributs associés aux personnes sont : n^o d'identifiant, nom, prénom. L'étudiant a comme attribut le dépt, le doctorant a un sujet de thèse, et l'employé a pour attribut sa classe (A,B,C).

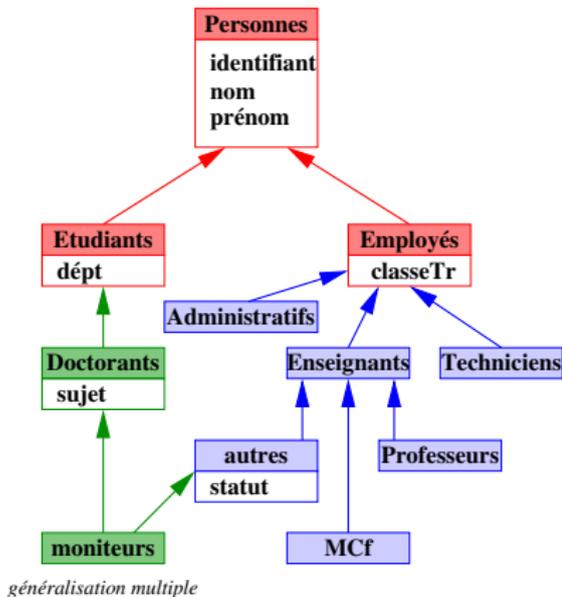


généralisation multiple

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsConcepts de base et
Diagramme EAgénéralisation/
spécialisationSpécification des
diagrammes EAContraintes
d'intégritéNormalisation
d'une relation

Exercice : Décrire par un diagramme EA l'ensemble des personnes travaillant dans une université. **Chaque individu est une personne, la population des personnes est partagée en étudiants et employés.** La population des personnels est découpée en trois : les enseignants (MCF, Pr, et autres), les techniciens et les administratifs. Parmi les étudiants, on particularise les doctorants, parmi lesquels certains sont moniteurs, c'est-à-dire héritent de la population des enseignants.

Les attributs associés aux personnes sont : n^o d'identifiant, nom, prénom. L'étudiant a comme attribut le dépt, le doctorant a un sujet de thèse, et l'employé a pour attribut sa classe (A,B,C).



Problèmes liés à l'héritage

- les sous-types peuvent hériter deux fois d'un ancêtre commun
- conflit d'héritage lorsqu'un sous-type hérite de 2 TE ayant un nom d'attribut en commun.
- Ces problèmes doivent être résolus par le concepteur : préférence d'héritage par exemple.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

généralisation /
spécialisation

Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

Les Types d'Entités (TE)

- nom du type,
- nom des sur-types s'il en existe,
- la définition libre (commentaire) précisant la sémantique du TE,
- la description des attributs,
- les identifiants.

Deux TE différents ne peuvent pas avoir le même nom.

Les Types d'Associations (TA)

- nom du type,
- la définition libre (commentaire) précisant la sémantique du TA,
- le nom des TE avec le nom du rôle les associant au TA (en pratique, ce n'est nécessaire que pour les associations cycliques),
- pour chacun des rôles, ses cardinalités ; si la cardinalité max est supérieure à 1, on précise si les rôles liant une entité constitue une liste ou un ensemble (valeur par défaut),
- la description des attributs,
- la composition des identifiants, s'il en existe.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

généralisation /
spécialisation

Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

Les attributs : plusieurs types :

- **attributs simples** (attributs non composés),
- **attributs complexes** (attributs composés),
- **attributs monovalués** (cardinalité $\max=1$),
- **attributs multivalués** (cardinalité $\max>1$),
- **attributs obligatoires** (cardinalité $\min=1$),
- **attributs facultatifs** (cardinalité $\min=0$).

Un attribut est décrit par les spécifications suivantes :

- nom de l'attribut,
- définition libre,
- cardinalité (min :max)
Si la cardinalité est supérieure à 1, on précise si les valeurs de l'attribut constituent une liste ou un ensemble (valeur par défaut)
- si l'attribut n'est pas composé d'autres attributs : domaine de valeurs définissant les valeurs possibles,
- si l'attribut est composé d'autres attributs : description des attributs composants.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

généralisation /
spécialisation

Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

Exemple : On considère un type d'entités *employés* qui doit mémoriser en plus des identifiants, noms et prénoms de la personne, les formations validées, ainsi que l'ensemble des postes occupées par l'employé en question.

employés	cardinalité	type d'attribut
identifiantE	1:1	simple, monovalué, obligatoire
Nom	1:1	simple, monovalué, obligatoire
prénoms	1:N	simple, multivalué, obligatoire
formation Validée	0:N	complexe, multivalué, facultatif
diplome	1:1	
année	1:1	
poste	1:N	complexe, multivalué, obligatoire
intitule	1:1	
date-debut	1:1	
date_fin	1:1	
salaire	1:N	
montant	1:1	
date	1:1	
année	1:1	
mois	1:1	

Considérons le type d'associations Mariages et le type d'entités Personnes.

- si une personne participe à une association du type Mariages son état civil doit être marié.

$$\forall x, y \in \text{Personnes}, (x, y) \in \text{Mariages} \implies \begin{cases} x.\text{état} = \text{"Marié"} \wedge \\ y.\text{état} = \text{"Marié"} \end{cases}$$

- si l'état civil d'une personne est "Marié", il ne peut être changé par la suite en "Célibataire"

$$\forall x \in \text{Personnes}, \forall t_1, t_2 \in \text{Temps}, \\ (t_1 < t_2) \wedge x(t_1).\text{état} = \text{"Marié"} \implies x(t_2).\text{état} \neq \text{"Célibataire"}$$

Autre exemple : tournois de doubles mixtes.

- seuls les hommes (resp. femmes) participent à l'association Equipe dans le rôle d'homme (resp. de femme) :

$$\begin{aligned} \forall x, y \in \text{Personnes}, (\text{homme} : x, y) \in \text{Equipe} &\implies x.\text{genre} = \text{"M"} \\ \forall x, y \in \text{Personnes}, (\text{femme} : x, y) \in \text{Equipe} &\implies x.\text{genre} = \text{"F"} \end{aligned}$$

Ces règles sont décrites à l'aide de **règles d'intégrité**. Si ces règles ne sont pas satisfaites par la BD, la base est dite **incohérente**.

L'implémentation se fait par des prédicats de contraintes (**CHECK**), par des procédures déclenchées automatiquement (voir les **TRIGGERS**) ou par des procédures associées aux schémas (voir les stored procedure).

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Concepts de base et
Diagramme EA

généralisation /
spécialisation

Spécification des
diagrammes EA

Contraintes
d'intégrité

Normalisation
d'une relation

Sur les attributs : Les contraintes les plus fréquentes cherchent à réduire les valeurs possibles d'un attribut à certaines valeurs du domaine sous-jacent.

Exemple :

- un mois dans une date sera un entier compris entre 1 et 12 :
 $\text{mois} \in [1 : 12]$.

Ces contraintes peuvent aussi dépendre d'un contexte :

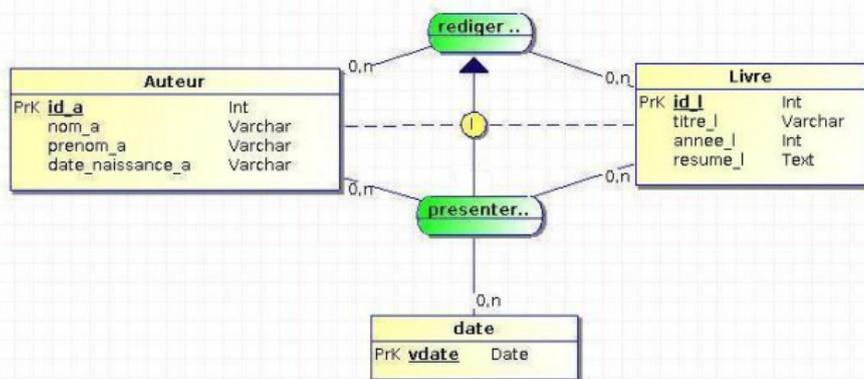
- si $\text{mois} \in \{4, 6, 9, 11\}$ alors $\text{jour} \in [1 : 30]$, sinon si $\text{mois} = 2$ alors $\text{jour} \in [1 : 29]$, sinon $\text{jour} \in [1 : 31]$.
- si une personne participe à l'association Mariages, alors son état civil est marié.

Sur les cardinalités : Prenons un exemple :

- le TA "est_parent_de"
- le TE Parents comprend l'attribut "nb_enfants".

⇒ Contrainte d'intégrité :

« le "nb_enfants" d'un parent est égal au nombre d'occurrences dans le TA "est_parent_de" qui lie ce parent. »

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsConcepts de base et
Diagramme EAgénéralisation /
spécialisationSpécification des
diagrammes EAContraintes
d'intégritéNormalisation
d'une relation

"Inclusion" : si un auteur présente son livre, c'est qu'il l'a écrit !

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

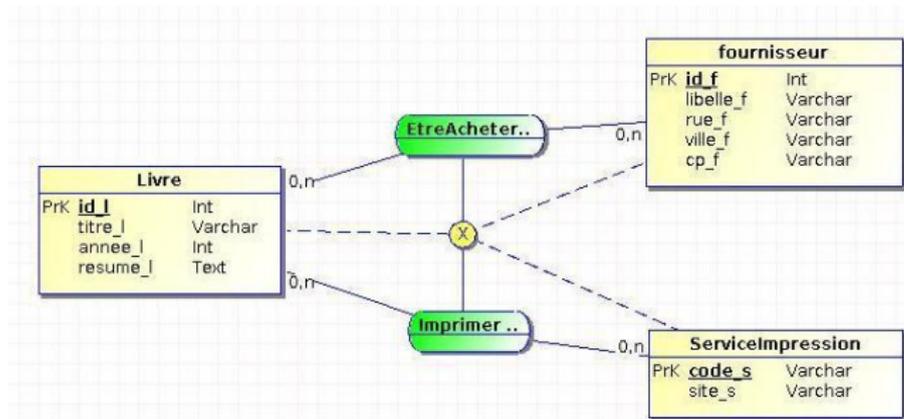
Concepts de base et
Diagramme EA

généralisation /
spécialisation

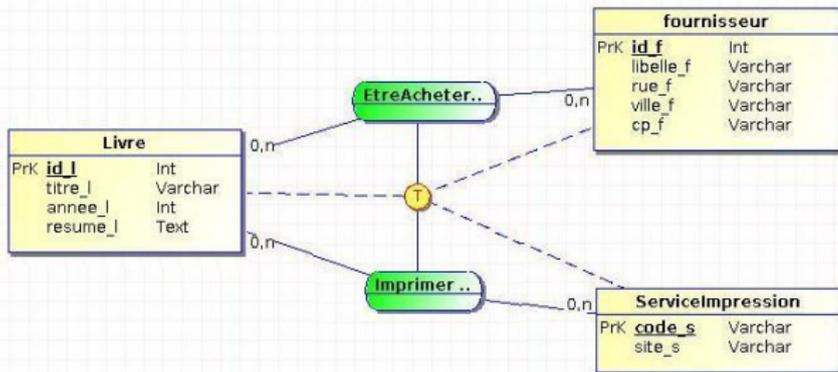
Spécification des
diagrammes EA

Contraintes
d'intégrité

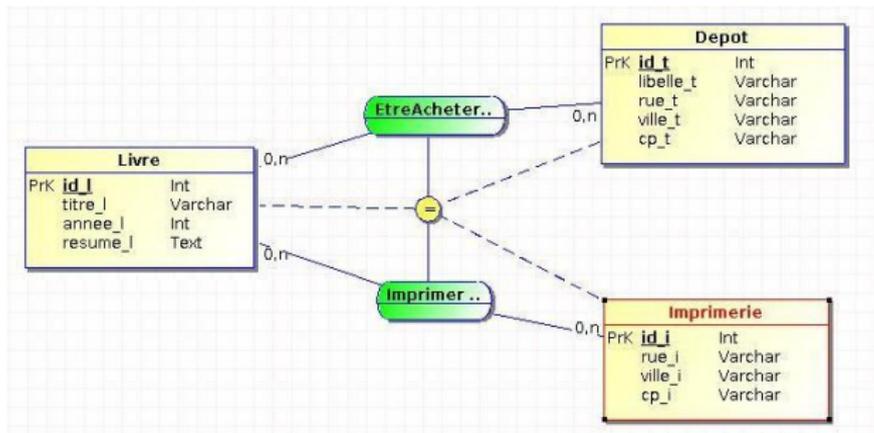
Normalisation
d'une relation



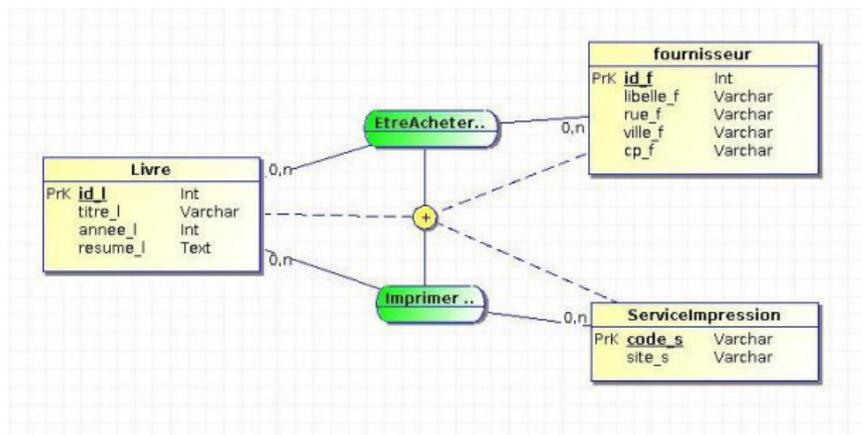
"ou exclusif" : si un livre est présent, soit il a été acheté à un fournisseur, soit il a été imprimé dans un des services d'impression du groupe.

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsConcepts de base et
Diagramme EAgénéralisation /
spécialisationSpécification des
diagrammes EAContraintes
d'intégritéNormalisation
d'une relation

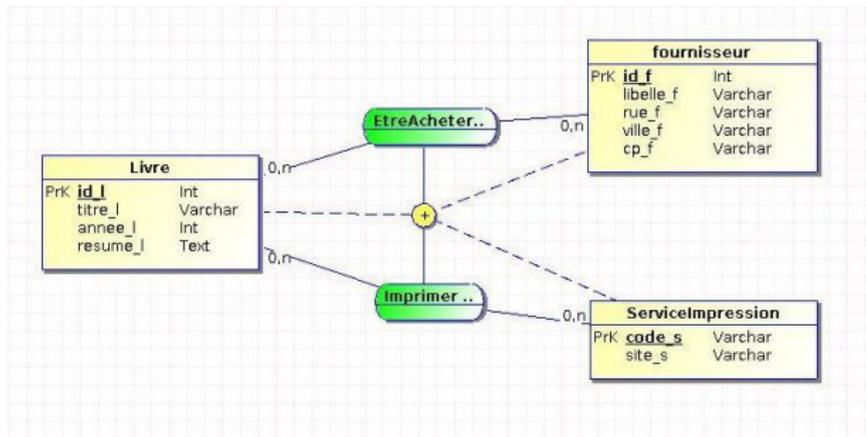
"Totalité" : il n'y a pas d'autre choix...



"Egalité" : Un livre (vu par une librairie) acheté dans un dépôt est également imprimé dans une imprimerie et vice et versa.



"Totalité + ou exclusif" noté "TX" ou "+" : le livre sera soit imprimé dans un service interne, soit acheté.

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsConcepts de base et
Diagramme EAgénéralisation /
spécialisationSpécification des
diagrammes EAContraintes
d'intégritéNormalisation
d'une relation

"Totalité + ou exclusif" noté "TX" ou "+" : le livre sera soit imprimé dans un service interne, soit acheté.

Pour conclure sur le modèle Entités-Associations : un schéma conceptuel Entités-Associations est un ensemble de descriptions de types d'entités et de types d'associations (attributs, liens de généralisations...) et de contraintes d'intégrités.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphe des DF
Couverture minimale
Décomposition
Les formes normales

- 1 Les concepts généraux
- 2 Notions essentielles pour le modèle relationnel
- 3 Algèbre Relationnelle et SQL
- 4 Le modèle conceptuel Entités-Associations
- 5 Normalisation d'une relation
 - Graphe des dépendances fonctionnelles
 - Couverture minimale
 - Décomposition d'une relation
 - Les formes normales

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphe des DF
Couverture minimale
Décomposition
Les formes normales

- La normalisation des relations a pour but d'aider la conception de schémas relationnels.
- À travers cette normalisation, on va pouvoir améliorer, selon certains critères, la qualité d'un schéma relationnel.

Exemple. Lequel de ces deux schémas relationnels est le meilleur ? pourquoi ?

Produit (NP, NomP, Couleur, Poids)

Fournisseur (NF, NomF, Adr, Tel)

Livraison (NP, NF, Date, Qté)

Produit (NP, NomP, Couleur, Poids)

Fournisseur (NF, NomF, Adr)

Livraison (NP, NF, Date, TelF, Qté)

- La normalisation des relations a pour but d'aider la conception de schémas relationnels.
- À travers cette normalisation, on va pouvoir améliorer, selon certains critères, la qualité d'un schéma relationnel.

Exemple. Lequel de ces deux schémas relationnels et le meilleur ? pourquoi ?

Produit (NP, NomP, Couleur, Poids)

Fournisseur (NF, NomF, Adr, Tel)

Livraison (NP, NF, Date, Qté)

Produit (NP, NomP, Couleur, Poids)

Fournisseur (NF, NomF, Adr)

Livraison (NP, NF, Date, TelF, Qté)

Le second soulève les problèmes suivants :

- Si un fournisseur ne livre plus, on perd son n^o de téléphone
- Si un fournisseur livre beaucoup, on duplique l'information du n^o de tél.
- Pour enregistrer une nouvelle livraison, il faut recopier le n^o de tél.
- Si un fournisseur change de n^o de tél, il faut propager la modification sur toutes les livraisons de ce fournisseur.

La *qualité* de la relation peut être évaluée par son *degré de normalisation* : une relation peut être

- 1^{ère}, 2^{ème}, 3^{ème} forme normale,
- en forme normale de Boyce-Codd ou
- en 4^{ème} forme normale.

Plus le degré de normalisation est élevé, plus la relation a de *bonnes propriétés*.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Grappe des DF

Couverture minimale

Décomposition

Les formes normales

Etant donné une relation $R(X, Y, Z)$, il existe une **dépendance fonctionnelle**, ou DF, de Y vers Z , (Y et Z étant soit des attributs soit des ensembles d'attributs) souvent notée $Y \rightarrow Z$, si : pour tout couple *envisageable* de n -uplets de R , s'ils ont même valeurs pour l'attribut Y , ils ont aussi même valeur pour Z .
On appelle Y la *source* de la dépendance fonctionnelle et Z la *cible*.

Exemple. Pour la relation $\text{Produit}(\text{NP}, \text{NomP}, \text{Couleur}, \text{Poids})$, les dépendances sont les suivantes, si on considère que deux produits différents ne peuvent pas avoir le même nom :

- $\text{NP} \rightarrow \text{NomP}, \quad \text{NP} \rightarrow \text{Poids}, \quad \text{NP} \rightarrow \text{Couleur},$
- $\text{NomP} \rightarrow \text{NP}, \quad \text{NomP} \rightarrow \text{Couleur}, \quad \text{NomP} \rightarrow \text{Poids},$
- $\text{NP} \rightarrow (\text{NomP}, \text{Poids}, \text{Couleur})$
- $(\text{NP}, \text{NomP}) \rightarrow \text{Poids}, \quad (\text{NP}, \text{NomP}) \rightarrow \text{Couleur},$
- ...

Une **dépendance fonctionnelle** $X \rightarrow B$ est **élémentaire** (on dit parfois complète, pleine) si, X et B étant resp. un groupe d'attributs et un attribut d'une même relation, B n'est pas fonctionnellement dépendant d'un sous-ensemble de X .

Exemple. Pour les relations $\left\{ \begin{array}{l} \text{Produit}(\text{NP}, \text{NomP}, \text{Couleur}, \text{Poids}) \\ \text{Livraison}(\text{NP}, \text{NF}, \text{Date}, \text{Qté}) \end{array} \right.$

- Les DF $\left\{ \begin{array}{l} \text{NP} \rightarrow (\text{Poids}, \text{Couleur}) \\ (\text{NP}, \text{NomP}) \rightarrow \text{Poids} \end{array} \right.$ ne sont pas élémentaires.
- La DF $(\text{NP}, \text{NF}, \text{date}) \rightarrow \text{Qté}$ est élémentaire (si un même fournisseur ne se déplace qu'une seule fois par jour pour un client et un produit).

Chaque DF exprime un fait du monde réel.

- Les DFs élémentaires expriment des faits élémentaires.
 $\text{NP} \rightarrow \text{Couleur}$: chaque produit, identifié par un n^o , a une couleur bien caractérisée.
 $(\text{NP}, \text{NF}, \text{date}) \rightarrow \text{Qté}$: un fournisseur ne peut pas livrer le même jour le même produit au même client avec des quantités différentes.

Propriétés :

- Si pour une relation donnée, on a les dépendances fonctionnelles $X \rightarrow Y$ et $Y \rightarrow Z$, alors on a aussi la dépendance fonctionnelle $X \rightarrow Z$ qui est dite **déduite** des deux autres.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphes des DF
Couverture minimale
Décomposition
Les formes normales

Etant donné une relation et un ensemble F de DF portant sur les attributs de cette relation, on appelle **graphe minimum** des DF de la relation, tout graphe (S, A) dont

- S est l'ensemble des attributs de la relation et
- A est un sous-ensemble de F de DF élémentaires non déduites, à partir desquelles toute dépendance de F peut être déduite.

Il est facile de savoir si une DF est déduite des autres ou non.

On construit le graphe de toutes les dépendances fonctionnelles.

Une DF $X \rightarrow Y$ est déduite s'il existe un autre chemin allant de X à Y .

Remarques.

- 1 Une **clé** est un sous-ensemble d'attributs respectant les 2 contraintes :
 - **Unicité** : 2 n -uplets distincts ne peuvent avoir la même valeur pour ces attributs
 - **Irréductibilité** : il n'existe pas de sous-ensemble strict de la clé garantissant la règle d'unicité.
- 2 Tout graphe de DF (minimum ou non) peut être utilisé pour la recherche des identifiants : les identifiants correspondent aux ensembles minimaux d'attributs X pour lesquels les chemins, partant de ces attributs, atteignent *tous* les autres attributs du graphe.
- 3 **Concept de sur-clé** :
une surclé est un sous-ensemble d'attributs de la relation, respectant uniquement la règle d'unicité

- Une couverture minimale d'un ensemble de DF est un sous ensemble minimum de dépendances fonctionnelles élémentaires (DFE) qui permettent de générer toutes les autres.
- Tout ensemble de dépendances fonctionnelles possède une couverture minimale (pas forcément unique) composée de dépendances fonctionnelles dont les parties droites contiennent 1 seul attribut.

1^{er} Algorithme de calcul de couverture minimale

- 1 Décomposition de chaque DF pour avoir un seul attribut à droite
- 2 Suppression des attributs en surnombre à gauche : Pour tout $X \rightarrow Y$, s'il existe un $Z \subset X$ tel que $Z \rightarrow Y$ alors remplacer $X \rightarrow Y$ par $Z \rightarrow Y$
- 3 Suppression des DF redondantes (qu'on peut obtenir par transitivité)

Exemple : $A \rightarrow B \mid B, C \rightarrow D \mid A, C \rightarrow B, D, E \mid D \rightarrow E$

- 1 Etape 1 : décomposition
- 2 Etape 2 : Suppression des attributs en surnombre à gauche
- 3 Etape 3 : Supprimer la redondance

⇒ la couverture minimale :

- Une couverture minimale d'un ensemble de DF est un sous ensemble minimum de dépendances fonctionnelles élémentaires (DFE) qui permettent de générer toutes les autres.
- Tout ensemble de dépendances fonctionnelles possède une couverture minimale (pas forcément unique) composée de dépendances fonctionnelles dont les parties droites contiennent 1 seul attribut.

1^{er} Algorithme de calcul de couverture minimale

- 1 Décomposition de chaque DF pour avoir un seul attribut à droite
- 2 Suppression des attributs en surnombre à gauche : Pour tout $X \rightarrow Y$, s'il existe un $Z \subset X$ tel que $Z \rightarrow Y$ alors remplacer $X \rightarrow Y$ par $Z \rightarrow Y$
- 3 Suppression des DF redondantes (qu'on peut obtenir par transitivité)

Exemple : $A \rightarrow B \mid B, C \rightarrow D \mid A, C \rightarrow B, D, E \mid D \rightarrow E$

- 1 Etape 1 : décomposition
 $A \rightarrow B \mid B, C \rightarrow D \mid A, C \rightarrow B \mid A, C \rightarrow D \mid A, C \rightarrow E \mid D \rightarrow E$
- 2 Etape 2 : Suppression des attributs en surnombre à gauche
- 3 Etape 3 : Supprimer la redondance

⇒ la couverture minimale :

- Une couverture minimale d'un ensemble de DF est un sous ensemble minimum de dépendances fonctionnelles élémentaires (DFE) qui permettent de générer toutes les autres.
- Tout ensemble de dépendances fonctionnelles possède une couverture minimale (pas forcément unique) composée de dépendances fonctionnelles dont les parties droites contiennent 1 seul attribut.

1^{er} Algorithme de calcul de couverture minimale

- 1 Décomposition de chaque DF pour avoir un seul attribut à droite
- 2 Suppression des attributs en surnombre à gauche : Pour tout $X \rightarrow Y$, s'il existe un $Z \subset X$ tel que $Z \rightarrow Y$ alors remplacer $X \rightarrow Y$ par $Z \rightarrow Y$
- 3 Suppression des DF redondantes (qu'on peut obtenir par transitivité)

Exemple : $A \rightarrow B \mid B, C \rightarrow D \mid A, C \rightarrow B, D, E \mid D \rightarrow E$

- 1 Etape 1 : décomposition
 $A \rightarrow B \mid B, C \rightarrow D \mid A, C \rightarrow B \mid A, C \rightarrow D \mid A, C \rightarrow E \mid D \rightarrow E$
- 2 Etape 2 : Suppression des attributs en surnombre à gauche
 $A \rightarrow B \mid B, C \rightarrow D \mid A, C \rightarrow D \mid A, C \rightarrow E \mid D \rightarrow E$
- 3 Etape 3 : Supprimer la redondance

⇒ la couverture minimale :

- Une couverture minimale d'un ensemble de DF est un sous ensemble minimum de dépendances fonctionnelles élémentaires (DFE) qui permettent de générer toutes les autres.
- Tout ensemble de dépendances fonctionnelles possède une couverture minimale (pas forcément unique) composée de dépendances fonctionnelles dont les parties droites contiennent 1 seul attribut.

1^{er} Algorithme de calcul de couverture minimale

- 1 Décomposition de chaque DF pour avoir un seul attribut à droite
- 2 Suppression des attributs en surnombre à gauche : Pour tout $X \rightarrow Y$, s'il existe un $Z \subset X$ tel que $Z \rightarrow Y$ alors remplacer $X \rightarrow Y$ par $Z \rightarrow Y$
- 3 Suppression des DF redondantes (qu'on peut obtenir par transitivité)

Exemple : $A \rightarrow B \mid B, C \rightarrow D \mid A, C \rightarrow B, D, E \mid D \rightarrow E$

- 1 Etape 1 : décomposition
 $A \rightarrow B \mid B, C \rightarrow D \mid A, C \rightarrow B \mid A, C \rightarrow D \mid A, C \rightarrow E \mid D \rightarrow E$
- 2 Etape 2 : Suppression des attributs en surnombre à gauche
 $A \rightarrow B \mid B, C \rightarrow D \mid A, C \rightarrow D \mid A, C \rightarrow E \mid D \rightarrow E$
- 3 Etape 3 : Supprimer la redondance
($A, C \rightarrow D$) redondante : se déduit de ($A \rightarrow B$) et ($B, C \rightarrow D$).
($A, C \rightarrow E$) redondante : se déduit de ($A \rightarrow B$), ($B, C \rightarrow D$) et ($D \rightarrow E$).

⇒ la couverture minimale :

- Une couverture minimale d'un ensemble de DF est un sous ensemble minimum de dépendances fonctionnelles élémentaires (DFE) qui permettent de générer toutes les autres.
- Tout ensemble de dépendances fonctionnelles possède une couverture minimale (pas forcément unique) composée de dépendances fonctionnelles dont les parties droites contiennent 1 seul attribut.

1^{er} Algorithme de calcul de couverture minimale

- 1 Décomposition de chaque DF pour avoir un seul attribut à droite
- 2 Suppression des attributs en surnombre à gauche : Pour tout $X \rightarrow Y$, s'il existe un $Z \subset X$ tel que $Z \rightarrow Y$ alors remplacer $X \rightarrow Y$ par $Z \rightarrow Y$
- 3 Suppression des DF redondantes (qu'on peut obtenir par transitivité)

Exemple : $A \rightarrow B \mid B, C \rightarrow D \mid A, C \rightarrow B, D, E \mid D \rightarrow E$

- 1 Etape 1 : décomposition
 $A \rightarrow B \mid B, C \rightarrow D \mid A, C \rightarrow B \mid A, C \rightarrow D \mid A, C \rightarrow E \mid D \rightarrow E$
- 2 Etape 2 : Suppression des attributs en surnombre à gauche
 $A \rightarrow B \mid B, C \rightarrow D \mid A, C \rightarrow D \mid A, C \rightarrow E \mid D \rightarrow E$
- 3 Etape 3 : Supprimer la redondance
($A, C \rightarrow D$) redondante : se déduit de ($A \rightarrow B$) et ($B, C \rightarrow D$).
($A, C \rightarrow E$) redondante : se déduit de ($A \rightarrow B$), ($B, C \rightarrow D$) et ($D \rightarrow E$).

⇒ la couverture minimale : $A \rightarrow B \mid B, C \rightarrow D \mid D \rightarrow E$.

- \bullet *décomposer* d'une relation \equiv
 $\left\{ \begin{array}{l} \text{transformer la relation en un ensemble} \\ \text{de relations satisfaisantes qui} \\ \text{contiennent les mêmes informations.} \end{array} \right.$
- \bullet On peut décomposer une relation en un ensemble de relations projetées, si à partir des relations projetées, on peut retrouver la relation initiale par jointure.

Une décomposition d'une relation $R(X, Y, Z)$ en deux relations $R_1 = \pi[X, Y]R$ et $R_2 = \pi[X, Z]R$ est dite **sans perte d'information** si $R = R_1 \bowtie R_2$

Exemple : Soit la relation $\text{pers}(\underline{\text{numPers}}, \text{nom}, \text{adresse})$.

- \bullet La décomposition de la relation pers en deux relations $\pi \left[\begin{array}{c} \text{numPers,} \\ \text{nom} \end{array} \right] \text{pers}$ et $\pi \left[\begin{array}{c} \text{numPers,} \\ \text{adresse} \end{array} \right] \text{pers}$, est sans perte d'information.

Soit la relation pers :

1234	Comet	Nice
2345	Comet	Paris

$$\pi \left[\begin{array}{c} \text{numPers,} \\ \text{nom} \end{array} \right] \text{pers} =$$

1234	Comet
2345	Comet

$$\pi \left[\begin{array}{c} \text{numPers,} \\ \text{adresse} \end{array} \right] \text{pers} =$$

1234	Nice
2345	Paris

$$\pi \left[\begin{array}{c} \text{numPers,} \\ \text{nom} \end{array} \right] \text{pers} \bowtie \pi \left[\begin{array}{c} \text{numPers,} \\ \text{adresse} \end{array} \right] \text{pers} =$$

1234	Comet	Nice
2345	Comet	Paris

- \bullet *décomposer* d'une relation \equiv
 $\left\{ \begin{array}{l} \text{transformer la relation en un ensemble} \\ \text{de relations satisfaisantes qui} \\ \text{contiennent les mêmes informations.} \end{array} \right.$
- \bullet On peut décomposer une relation en un ensemble de relations projetées, si à partir des relations projetées, on peut retrouver la relation initiale par jointure.

Une décomposition d'une relation $R(X, Y, Z)$ en deux relations $R_1 = \pi[X, Y]R$ et $R_2 = \pi[X, Z]R$ est dite **sans perte d'information** si $R = R_1 \bowtie R_2$

Exemple : Soit la relation pers(numPers, nom, adresse).

- \bullet La décomposition de la relation pers en deux relations $\pi \left[\begin{array}{c} \text{numPers,} \\ \text{nom} \end{array} \right] \text{pers}$

et $\pi \left[\begin{array}{c} \text{nom,} \\ \text{adresse} \end{array} \right] \text{pers}$, est **avec perte d'information**.

Soit la relation pers :

1234	Comet	Nice
2345	Comet	Paris

$\pi \left[\begin{array}{c} \text{numPers,} \\ \text{nom} \end{array} \right] \text{pers} =$

1234	Comet
2345	Comet

$\pi \left[\begin{array}{c} \text{nom,} \\ \text{adresse} \end{array} \right] \text{pers} =$

Comet	Nice
Comet	Paris

$\pi \left[\begin{array}{c} \text{numPers,} \\ \text{nom} \end{array} \right] \text{pers} \bowtie \pi \left[\begin{array}{c} \text{nom,} \\ \text{adresse} \end{array} \right] \text{pers} =$

1234	Comet	Nice
1234	Comet	Paris
2345	Comet	Nice
2345	Comet	Paris

- $décomposer$ d'une relation \equiv
 $\left\{ \begin{array}{l} \text{transformer la relation en un ensemble} \\ \text{de relations satisfaisantes qui} \\ \text{contiennent les mêmes informations.} \end{array} \right.$
- On peut décomposer une relation en un ensemble de relations projetées, si à partir des relations projetées, on peut retrouver la relation initiale par jointure.

Une décomposition d'une relation $R(X, Y, Z)$ en deux relations $R_1 = \pi[X, Y]R$ et $R_2 = \pi[X, Z]R$ est dite **sans perte d'information** si $R = R_1 \bowtie R_2$

Exemple : Soit la relation $pers(\underline{numPers}, nom, adresse)$.

- La décomposition de la relation $pers$ en deux relations $\pi \left[\begin{array}{c} numPers, \\ nom \end{array} \right] pers$

et $\pi \left[\begin{array}{c} nom, \\ adresse \end{array} \right] pers$, est **avec perte d'information**.

Si la décomposition est **avec perte d'information**, la relation obtenue par jointure naturelle est composée de plus de n -uplets !

Soit la relation $pers$:

1234	Comet	Nice
2345	Comet	Paris

$\pi \left[\begin{array}{c} numPers, \\ nom \end{array} \right] pers =$

1234	Comet
2345	Comet

$\pi \left[\begin{array}{c} nom, \\ adresse \end{array} \right] pers =$

Comet	Nice
Comet	Paris

$\pi \left[\begin{array}{c} numPers, \\ nom \end{array} \right] pers \bowtie \pi \left[\begin{array}{c} nom, \\ adresse \end{array} \right] pers =$

1234	Comet	Nice
1234	Comet	Paris
2345	Comet	Nice
2345	Comet	Paris

Theoreme de Heath Toute relation $R(X, Y, Z)$ est décomposable sans perte d'information en $R_1 = \pi[X, Y]R$ et $R_2 = \pi[X, Z]R$, s'il y a dans R une dépendance fonctionnelle de X vers Y

Preuve.

① Montrons que $R \subseteq R_1 \bowtie R_2$:

$R_1 \bowtie R_2$ contient au moins tous les n -uplets de R , puisque tout n -uplet xyz de R crée un n -uplet xy dans R_1 et un n -uplet xz dans R_2 . Ces deux n -uplets sont ensuite concaténées dans la jointure naturelle en xyz .

② Montrons par l'absurde que $R_1 \bowtie R_2 \setminus R$ est vide :

Soit xyz un n -uplet de $R_1 \bowtie R_2$ qui n'appartient pas à R .

- xyz provient de deux n -uplets xy de R_1 et xz de R_2 .
- Si xy est dans R_1 , c'est qu'il y avait dans R un n -uplet xyz' .
Si xz est dans R_2 , c'est qu'il y avait dans R un n -uplet $xy'z$.
- comme $xyz \notin R$, on a $z' \neq z$ et $y' \neq y$.

Donc pour un même x , on a deux n -uplets xyz' et $xy'z$ qui ont des valeurs différentes pour Y , ce qui contredit la dépendance fonctionnelle $X \rightarrow Y$.

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsNormalisation
d'une relationGraphe des DF
Couverture minimale
Décomposition
Les formes normales

Une relation est en **première forme normale** si chaque valeur de chaque attribut de chaque n -uplet est une valeur simple (tous les attributs sont simples et monovalués).

Une relation est en **deuxième forme normale** si elle est en première forme normale et si chaque attribut qui ne fait partie d'aucun identifiant, ne dépend pas d'une sous partie stricte d'une clef (c-à-d *dépend de tout identifiant entier*), autrement dit, si toutes les dépendances fonctionnelles entre la clé et les autres attributs sont élémentaires.

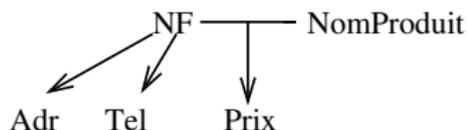
« *Tout l'identifiant est utile.* »

La méthode à suivre pour normaliser une relation est la suivante :

- vérifier que la relation est en première forme normale,
- établir un graphe minimum de dépendances fonctionnelles,
- déterminer tous ses identifiants,
- déterminer à l'aide du graphe sa forme normale,
- si la relation n'est pas (suffisamment) normalisée, décomposer à l'aide du graphe la relation en relations mieux normalisées.

Exemple. Fournisseur1(NF, NomProduit, ADR, Tel, Prix) supposée en 1FN.

- Les DF : $\left\{ \begin{array}{l} (\text{nomProd}, \text{NF}) \rightarrow \text{prix}, \\ \text{NF} \rightarrow \text{Adr et} \\ \text{NF} \rightarrow \text{Tel.} \end{array} \right.$



Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

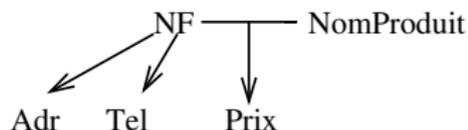
le modèle
Entités-
Associations

Normalisation
d'une relation

Graphe des DF
Couverture minimale
Décomposition
Les formes normales

Exemple. Fournisseur1(NF, NomProduit, Adr, Tel, Prix) supposée en 1FN.

- Les DF : $\left\{ \begin{array}{l} (\text{nomProd}, \text{NF}) \rightarrow \text{prix}, \\ \text{NF} \rightarrow \text{Adr et} \\ \text{NF} \rightarrow \text{Tel.} \end{array} \right.$



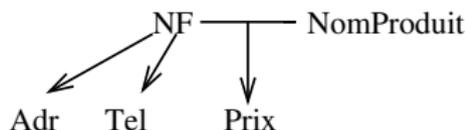
Limitations :

- Si 100 produits pour un fournisseur, on répétera 100 fois NF, Adr et Tel.
- Ajout d'un produit : il faut donner de nouveau Adr et Tel.
Suppression momentanément de tous les produits : perte de Adr et Tel.
- Changement d'adresse d'un fournisseur : le faire pour tous les produits.

Raison : la relation n'est pas en 2^{ème} FN.

Exemple. Fournisseur1(NF, NomProduit, Adr, Tel, Prix) supposée en 1FN.

- Les DF : $\left\{ \begin{array}{l} (\text{nomProd}, \text{NF}) \rightarrow \text{prix}, \\ \text{NF} \rightarrow \text{Adr et} \\ \text{NF} \rightarrow \text{Tel.} \end{array} \right.$



Limitations :

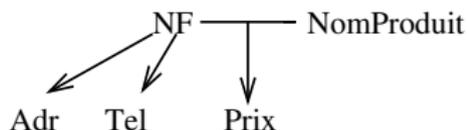
- Si 100 produits pour un fournisseur, on répétera 100 fois NF, Adr et Tel.
- Ajout d'un produit : il faut donner de nouveau Adr et Tel.
Suppression momentanément de tous les produits : perte de Adr et Tel.
- Changement d'adresse d'un fournisseur : le faire pour tous les produits.

Raison : la relation n'est pas en 2^{ème} FN. **On décompose :**

- il existe 3 DF dont 2 ont pour source un sous-ensemble de la source de l'autre. En effet, on a : $(\text{nomProd}, \text{NF}) \rightarrow \text{prix}$, $\text{NF} \rightarrow \text{Adr}$ et $\text{NF} \rightarrow \text{Tel}$.
- ① Pour chacune de ces sources de DF,
 - on crée une relation ayant pour attributs la source et tous les attributs en DF directe de cette source,
 - en s'assurant qu'une (ou +) de ces deux sources est contenue dans les attributs communs aux 2 relations créées (th. de Heath)

Exemple. Fournisseur1(NF, NomProduit, Adr, Tel, Prix) supposée en 1FN.

- Les DF : $\left\{ \begin{array}{l} (\text{nomProd}, \text{NF}) \rightarrow \text{prix}, \\ \text{NF} \rightarrow \text{Adr et} \\ \text{NF} \rightarrow \text{Tel.} \end{array} \right.$



Limitations :

- Si 100 produits pour un fournisseur, on répétera 100 fois NF, Adr et Tel.
- Ajout d'un produit : il faut donner de nouveau Adr et Tel.
Suppression momentanément de tous les produits : perte de Adr et Tel.
- Changement d'adresse d'un fournisseur : le faire pour tous les produits.

Raison : la relation n'est pas en 2^{ème} FN. **On décompose :**

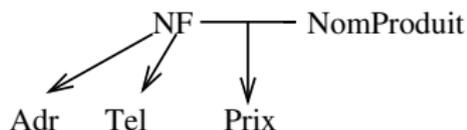
- il existe 3 DF dont 2 ont pour source un sous-ensemble de la source de l'autre. En effet, on a : $(\text{nomProd}, \text{NF}) \rightarrow \text{prix}$, $\text{NF} \rightarrow \text{Adr}$ et $\text{NF} \rightarrow \text{Tel}$.

- 1 Pour chacune de ces sources de DF,
 - on crée une relation ayant pour attributs la source et tous les attributs en DF directe de cette source,
 - en s'assurant qu'une (ou +) de ces deux sources est contenue dans les attributs communs aux 2 relations créées (th. de Heath)

$$\Rightarrow \left\{ \begin{array}{l} \text{Fournisseur}(\underline{\text{NF}}, \text{Adr}, \text{Tel}) \\ \text{Catalogue}(\underline{\text{NF}}, \underline{\text{NomProduit}}, \text{Prix}). \end{array} \right.$$

Exemple. Fournisseur1(NF, NomProduit, Adr, Tel, Prix) supposée en 1FN.

- Les DF : $\left\{ \begin{array}{l} (\text{nomProd}, \text{NF}) \rightarrow \text{prix}, \\ \text{NF} \rightarrow \text{Adr et} \\ \text{NF} \rightarrow \text{Tel.} \end{array} \right.$



Limitations :

- Si 100 produits pour un fournisseur, on répétera 100 fois NF, Adr et Tel.
- Ajout d'un produit : il faut donner de nouveau Adr et Tel.
Suppression momentanément de tous les produits : perte de Adr et Tel.
- Changement d'adresse d'un fournisseur : le faire pour tous les produits.

Raison : la relation n'est pas en 2^{ème} FN. **On décompose :**

- il existe 3 DF dont 2 ont pour source un sous-ensemble de la source de l'autre. En effet, on a : $(\text{nomProd}, \text{NF}) \rightarrow \text{prix}, \text{NF} \rightarrow \text{Adr et} \text{NF} \rightarrow \text{Tel.}$

- 1 Pour chacune de ces sources de DF,
 - on crée une relation ayant pour attributs la source et tous les attributs en DF directe de cette source,
 - en s'assurant qu'une (ou +) de ces deux sources est contenue dans les attributs communs aux 2 relations créées (th. de Heath)

$$\Rightarrow \left\{ \begin{array}{l} \text{Fournisseur}(\underline{\text{NF}}, \text{Adr}, \text{Tel}) \\ \text{Catalogue}(\underline{\text{NF}}, \underline{\text{NomProduit}}, \text{Prix}). \end{array} \right.$$

- On vérifie que cette décomposition est sans perte d'information, et
- sans perte de dépendances fonctionnelles (toutes les DF de la relation initiale se retrouvent, éventuellement pas transitivité, à partir des DF des 2 relations créées).

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphe des DF
Couverture minimale
Décomposition
Les formes normales

Autre exemple : joueur(personne, sport, taille)

- identifiant (personne, sport)
 - La DF personne \rightarrow taille fait que la relation n'est pas en 2^{ème} FN.
- \Rightarrow On décompose la relation en $\left\{ \begin{array}{l} \text{pratique}(\text{personne}, \text{sport}) \\ \text{hauteur}(\text{personne}, \text{taille}). \end{array} \right.$
- La relation joueur est alors retrouvée par une jointure naturelle.

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsNormalisation
d'une relationGraphe des DF
Couverture minimale
Décomposition
Les formes normales

« La 3NF permet d'éliminer les DF transitives. »

Une relation est en **troisième forme normale** si elle est en deuxième forme normale et si chaque attribut qui ne fait partie d'aucun identifiant, *dépend uniquement et directement* de sur-clés entières. (Chaque attribut ne dépend pas d'un ensemble d'attributs qui ne soit pas une sur-clef. La dépendance est directe : elle n'est pas dû à la transitivité des DF).

Remarque : 3NF \Rightarrow 2NF (même sans la première condition)

Exemple : Fournisseur2(NF, Pays, Ville) est en 2FN mais pas 3NF.

- Les DF : $NF \rightarrow Ville$ et $Ville \rightarrow Pays$.
- La DF $NF \rightarrow Pays$ est déduite (si pas d'homonymie entre villes)
- Le graphe minimum des DF de Fournisseur2 est donc : $NF \rightarrow Ville \rightarrow Pays$.

Il y a redondance : le pays d'une ville est répété.

On décompose donc en : $\left\{ \begin{array}{l} \text{Fourn}(\underline{NF}, \text{Ville}) \\ \text{Geo}(\underline{\text{Ville}}, \text{Pays}). \end{array} \right.$

- sans perte d'information (Ville est identifiant pour Géo)
- sans perte de DF (toutes les DF de la relation initiale se retrouvent à partir des DF des deux relations créées).

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphe des DF
Couverture minimale
Décomposition
Les formes normales

Exemple :

Voiture(Imm, Marque, Type, Puissance, Couleur) n'est pas en 3FN.

Type (non clé) détermine les attributs Marque et Puissance.

Il faut décomposer la relation : $\left\{ \begin{array}{l} \text{Voiture (Imm, Type, Couleur)} \\ \text{Modèle (Type, Marque, Puissance)} \end{array} \right.$

Remarque :

- On peut toujours décomposer une relation en un ensemble de relations qui sont chacune en 3FN.
- Ce n'est pas le cas pour les formes normales suivantes. D'où l'intérêt de cette troisième forme normale.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphe des DF
Couverture minimale
Décomposition
Les formes normales

Algorithme de décomposition 3NF (SPI et préservant les DF)

Soit un schéma de relation R , et une couverture minimale F' de dépendances fonctionnelles s'appliquant sur R . Une décomposition SPI et SPDF de R , qu'on appellera D , est construite de la façon suivante :

- 1 pour chaque DF $X \rightarrow A$, on crée une relation $R_i(X, A)$,
- 2 si on a plusieurs DF telles que $X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_n$, alors on regroupe tous ces attributs dans une même relation $R_j(X, A_1, \dots, A_n)$,

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphe des DF
Couverture minimale
Décomposition
Les formes normales

Algorithme de décomposition 3NF (SPI et préservant les DF)

Soit un schéma de relation R , et une couverture minimale F' de dépendances fonctionnelles s'appliquant sur R . Une décomposition SPI et SPDF de R , qu'on appellera D , est construite de la façon suivante :

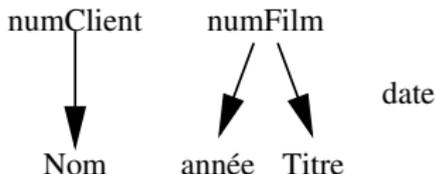
- 1 pour chaque DF $X \rightarrow A$, on crée une relation $R_i(X, A)$,
- 2 si on a plusieurs DF telles que $X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_n$, alors on regroupe tous ces attributs dans une m relation $R_j(X, A_1, \dots, A_n)$,
- 3 Pour avoir une décomposition SPI :
 - il faut s'assurer qu'il y ait au-moins une clé de R dans au moins une des relations de décomposition.
 - Si ce n'est pas le cas, il faut soit ajouter une relation contenant une clé de R , soit ajouter des attributs dans une des relations de décomposition afin de satisfaire cette contrainte.

Exemple.

- $\text{Location}(\underline{\text{numFilm}}, \text{numClient}, \text{date}, \text{titre}, \text{année}, \text{nom})$

Les DF : $\left\{ \begin{array}{l} \text{numFilm} \rightarrow \text{titre}, \\ \text{numFilm} \rightarrow \text{année}, \\ \text{numClient} \rightarrow \text{nom}. \end{array} \right.$

- Le graphe de DF :



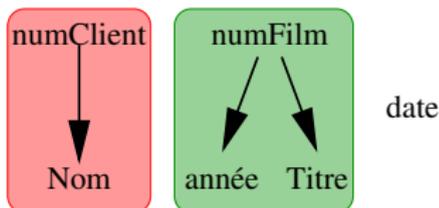
Exemple.

- Location(numFilm, numClient, date, titre, année, nom)

Les DF : $\left\{ \begin{array}{l} \text{numFilm} \rightarrow \text{titre}, \\ \text{numFilm} \rightarrow \text{année}, \\ \text{numClient} \rightarrow \text{nom}. \end{array} \right.$

- Les étapes 1 et 2 donnent 2 tables : $\left\{ \begin{array}{l} R1(\text{numFilm}, \text{titre}, \text{année}) \\ R2(\text{numClient}, \text{nom}). \end{array} \right.$

- Le graphe de DF :



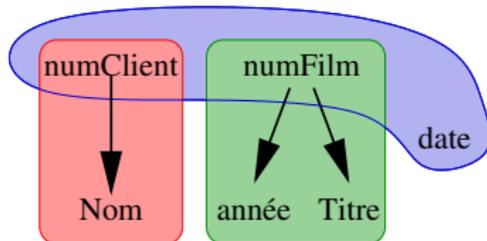
Exemple.

- Location(numFilm, numClient, date, titre, année, nom)

Les DF : $\left\{ \begin{array}{l} \text{numFilm} \rightarrow \text{titre}, \\ \text{numFilm} \rightarrow \text{année}, \\ \text{numClient} \rightarrow \text{nom}. \end{array} \right.$

- Les étapes 1 et 2 donnent 2 tables : $\left\{ \begin{array}{l} R1(\text{numFilm}, \text{titre}, \text{année}) \\ R2(\text{numClient}, \text{nom}). \end{array} \right.$
- L'étape 3 donne une table supplémentaire :
R2(numClient, numFilm, date).

- Le graphe de DF :



Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphes des DF
Couverture minimale
Décomposition
Les formes normales

Remarques.

- 1 L'algorithme a besoin d'une couverture minimale de DF. La recherche d'un graphe de DF minimum peut être compliquée.
- 2 S'il y a dans R des attributs n'intervenant dans aucune DF, alors ils font partie de l'identifiant. Ils seront donc intégrés dans la table créée à l'étape 3.
- 3 L'étape 3 est importante pour pouvoir éliminer les n -uplets excédentaires lors de la jointure pour reconstituer la relation initiale.
- 4 Parfois cet algorithme mène à des décompositions redondantes.

Exemple :

- soit la relation enseignement(numEtud,matière,prof)
chaque professeur n'enseigne qu'une seule matière.
 - $DF : (\text{numEtud}, \text{matiere}) \rightarrow \text{prof} \quad \text{prof} \rightarrow \text{matiere}.$
 - Cette relation est déjà en 3FN.
 - l'algorithme mène à : $\left\{ \begin{array}{l} (\text{numEtud}, \text{matière}, \text{prof}) \\ (\text{prof}, \text{matière}). \end{array} \right.$
La deuxième relation est incluse dans la première.
- 5 Il faut donc ensuite supprimer du résultat les relations qui sont incluses dans d'autres.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphes des DF
Couverture minimale
Décomposition
Les formes normales

Remarques.

- 1 L'algorithme a besoin d'une couverture minimale de DF. La recherche d'un graphe de DF minimum peut être compliquée.
- 2 S'il y a dans R des attributs n'intervenant dans aucune DF, alors ils font partie de l'identifiant. Ils seront donc intégrés dans la table créée à l'étape 3.
- 3 L'étape 3 est importante pour pouvoir éliminer les n -uplets excédentaires lors de la jointure pour reconstituer la relation initiale.
- 4 Parfois cet algorithme mène à des décompositions redondantes.

Exemple :

- soit la relation enseignement(numEtud,matière,prof)
chaque professeur n'enseigne qu'une seule matière.
 - $DF : (\text{numEtud}, \text{matiere}) \rightarrow \text{prof} \quad \text{prof} \rightarrow \text{matiere}.$
 - Cette relation est déjà en 3FN.
 - l'algorithme mène à : $\left\{ \begin{array}{l} (\text{numEtud}, \text{matière}, \text{prof}) \\ (\text{prof}, \text{matière}). \end{array} \right.$
La deuxième relation est incluse dans la première.
- 5 Il faut donc ensuite supprimer du résultat les relations qui sont incluses dans d'autres.



L'algorithme n'est pas applicable si le graphe de DF n'est pas minimal.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

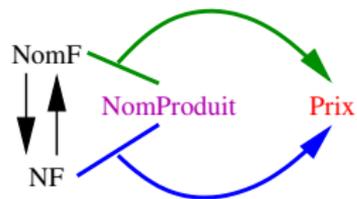
Graphe des DF
Couverture minimale
Décomposition
Les formes normales

2^{ème} algorithme de décomposition 3NF (sans couverture minimale) :

- 1 Repérer dans la relation R une DF $L \rightarrow A$ qui ne satisfait pas aux contraintes de la 3^{ème} forme normale : $A \notin L$, L n'est pas une surclef, et A ne fait pas partie d'un identifiant.
- 2 On projette alors la relation R en 2 tables :
 - une sur les attributs de R privé de A
 - l'autre sur les attributs $\{L, A\}$. Pour cette table, L est identifiant.
- 3 On réitère ce processus (en sachant que toute table binaire est en 3NF).

Une relation est en **forme normale de Boyce-Codd** si elle est en troisième forme normale et si toute source de DF élémentaire est une *sur-clé*, autrement dit, une relation est en **FNBC** si et seulement si les seules dépendances fonctionnelles élémentaires sont celles dans lesquelles une clé (ou sur-clé) détermine un attribut.

Le graphe de DF :



Exemple.

Catalogue3(NF, NomF, NomProduit, Prix)
en 3NF mais pas en FNBC

(pas d'homonymie chez les fournisseurs)

Les identifiants sont : (NF, NomProduit)
(NomF, NomProduit)

Il y a donc deux graphes de DF minimum :

$$NF \leftrightarrow NomF, (NomProduit, NomF) \rightarrow Prix$$

ou alors

$$NF \leftrightarrow NomF, (NomProduit, NF) \rightarrow Prix$$

Une relation est en **forme normale de Boyce-Codd** si elle est en troisième forme normale et si toute source de DF élémentaire est une *sur-clé*, autrement dit, une relation est en **FNBC** si et seulement si les seules dépendances fonctionnelles élémentaires sont celles dans lesquelles une clé (ou sur-clé) détermine un attribut.

Exemple.

Catalogue3(NF, NomF, NomProduit, Prix)
en 3NF mais pas en FNBC

(pas d'homonymie chez les fournisseurs)

Les identifiants sont : (NF, NomProduit)
(NomF, NomProduit)

Le graphe de DF :



Il y a donc deux graphes de DF minimum :

$$NF \leftrightarrow NomF, (NomProduit, NomF) \rightarrow Prix$$

ou alors

$$NF \leftrightarrow NomF, (NomProduit, NF) \rightarrow Prix$$

Une relation est en **forme normale de Boyce-Codd** si elle est en troisième forme normale et si toute source de DF élémentaire est une *sur-clé*, autrement dit, une relation est en **FNBC** si et seulement si les seules dépendances fonctionnelles élémentaires sont celles dans lesquelles une clé (ou sur-clé) détermine un attribut.

Le graphe de DF :

Exemple.

Catalogue3(NF, NomF, NomProduit, Prix)
en 3NF mais pas en FNBC

(pas d'homonymie chez les fournisseurs)

Les identifiants sont : (NF, NomProduit)
(NomF, NomProduit)



Il y a donc deux graphes de DF minimum :

$$NF \leftrightarrow NomF, (NomProduit, NomF) \rightarrow Prix$$

ou alors

$$NF \leftrightarrow NomF, (NomProduit, NF) \rightarrow Prix$$

Une relation est en **forme normale de Boyce-Codd** si elle est en troisième forme normale et si toute source de DF élémentaire est une *sur-clé*, autrement dit, une relation est en **FNBC** si et seulement si les seules dépendances fonctionnelles élémentaires sont celles dans lesquelles une clé (ou sur-clé) détermine un attribut.

Le graphe de DF :

Exemple.

Catalogue3(NF, NomF, NomProduit, Prix)
en 3NF mais pas en FNBC

(pas d'homonymie chez les fournisseurs)

Les identifiants sont : (NF, NomProduit)
(NomF, NomProduit)



Il y a donc deux graphes de DF minimum :

$$NF \leftrightarrow NomF, (NomProduit, NomF) \rightarrow Prix$$

ou alors

$$NF \leftrightarrow NomF, (NomProduit, NF) \rightarrow Prix$$

- La relation Catalogue3 n'est pas en FNBC : l'attribut NF est source complète de DF.
- Dans Catalogue3, il y a redondance entre NF et NomF

Une relation est en **forme normale de Boyce-Codd** si elle est en troisième forme normale et si toute source de DF élémentaire est une *sur-clé*, autrement dit, une relation est en **FNBC** si et seulement si les seules dépendances fonctionnelles élémentaires sont celles dans lesquelles une clé (ou sur-clé) détermine un attribut.

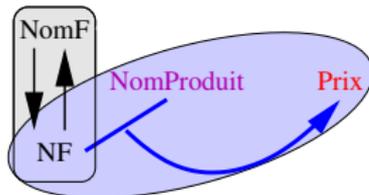
Le graphe de DF :

Exemple.

Catalogue3(NF, NomF, NomProduit, Prix)
en 3NF mais pas en FNBC

(pas d'homonymie chez les fournisseurs)

Les identifiants sont : (NF, NomProduit)
(NomF, NomProduit)



Il y a donc deux graphes de DF minimum :

$$NF \leftrightarrow NomF, (NomProduit, NomF) \rightarrow Prix$$

ou alors

$$NF \leftrightarrow NomF, (NomProduit, NF) \rightarrow Prix$$

- La relation Catalogue3 n'est pas en FNBC : l'attribut NF est source complète de DF.
- Dans Catalogue3, il y a redondance entre NF et NomF
- On décompose donc en : $\left\{ \begin{array}{l} \text{Catalogue}(\underline{NF}, \underline{NomProduit}, \underline{Prix}) \\ \text{Fournisseur}(\underline{NF}, \underline{NomF}). \end{array} \right.$

Autres exemples.

- ① Collège(élève, matière, enseignant, note) avec les DP suivantes :
- $$\begin{cases} (\text{élève}, \text{matière}) \rightarrow \text{enseignant}, \text{note} \\ \text{enseignant} \rightarrow \text{matière}. \end{cases}$$
- Les identifiants : (élève, matière) et (élève, enseignant).
3NF.



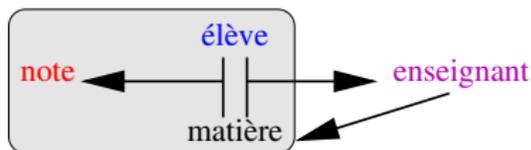
Autres exemples.

1 Collège(élève, matière, enseignant, note) avec les DP suivantes :

$$\left\{ \begin{array}{l} (\text{élève}, \text{matière}) \rightarrow \text{enseignant}, \text{note} \\ \text{enseignant} \rightarrow \text{matière}. \end{array} \right.$$

Les identifiants : (élève, matière) et (élève, enseignant).

3NF.



La relation collège est décomposée en :

$$\left\{ \begin{array}{l} \text{notes}(\underline{\text{élève}}, \underline{\text{matière}}, \text{note}) \\ \text{Enseignement}(\underline{\text{élève}}, \underline{\text{enseignant}}, \text{matière}). \end{array} \right.$$

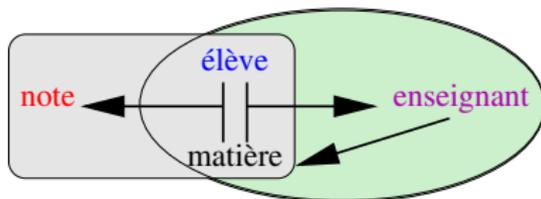
Autres exemples.

1 Collège(élève, matière, enseignant, note) avec les DP suivantes :

$$\begin{cases} (\text{élève}, \text{matière}) \rightarrow \text{enseignant}, \text{note} \\ \text{enseignant} \rightarrow \text{matière}. \end{cases}$$

Les identifiants : (élève, matière) et (élève, enseignant).

3NF.



La relation collège est décomposée en :

$$\begin{cases} \text{notes}(\underline{\text{élève}}, \underline{\text{matière}}, \text{note}) \\ \text{Enseignement}(\underline{\text{élève}}, \underline{\text{enseignant}}, \text{matière}). \end{cases}$$

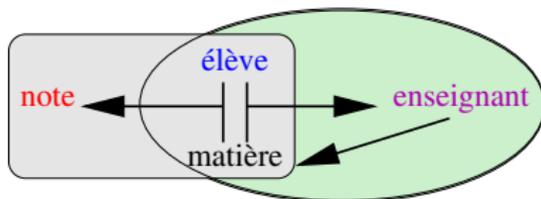
Autres exemples.

1 Collège(élève, matière, enseignant, note) avec les DP suivantes :

$$\begin{cases} (\text{élève}, \text{matière}) \rightarrow \text{enseignant}, \text{note} \\ \text{enseignant} \rightarrow \text{matière}. \end{cases}$$

Les identifiants : (élève, matière) et (élève, enseignant).

3NF.



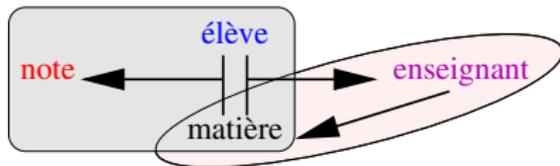
La relation collège est décomposée en :

$$\begin{cases} \text{notes}(\underline{\text{élève}}, \text{matière}, \text{note}) \\ \text{Enseignement}(\underline{\text{élève}}, \underline{\text{enseignant}}, \text{matière}). \end{cases}$$

Enseignement (élève, enseignant, Matière) n'est pas FNBC.

Autres exemples.

- 1 Collège(élève, matière, enseignant, note) avec les DP suivantes :
- $$\left\{ \begin{array}{l} (\text{élève}, \text{matière}) \rightarrow \text{enseignant}, \text{note} \\ \text{enseignant} \rightarrow \text{matière}. \end{array} \right.$$
- Les identifiants : (élève, matière) et (élève, enseignant).
3NF.



La relation collège est décomposée en :

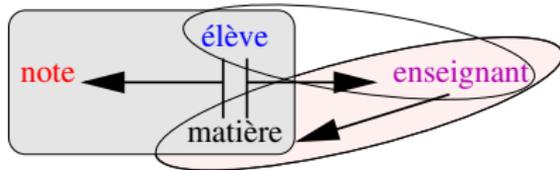
$$\left\{ \begin{array}{l} \text{notes}(\underline{\text{élève}}, \text{matière}, \text{note}) \\ \text{Enseignement}(\underline{\text{élève}}, \underline{\text{enseignant}}, \text{matière}). \end{array} \right.$$

Enseignement (élève, enseignant, Matière) n'est pas FNBC.

$$\text{On la décompose : } \left\{ \begin{array}{l} \text{notes}(\underline{\text{élève}}, \text{matière}, \text{note}), \\ \text{Enseignant}(\underline{\text{enseignant}}, \text{Matière}) \\ \text{Enseignement}(\underline{\text{élève}}, \underline{\text{enseignant}}). \end{array} \right.$$

Autres exemples.

- 1 Collège(élève, matière, enseignant, note) avec les DP suivantes :
- $$\begin{cases} (\text{élève}, \text{matière}) \rightarrow \text{enseignant}, \text{note} \\ \text{enseignant} \rightarrow \text{matière}. \end{cases}$$
- Les identifiants : (élève, matière) et (élève, enseignant).
3NF.



La relation collège est décomposée en :

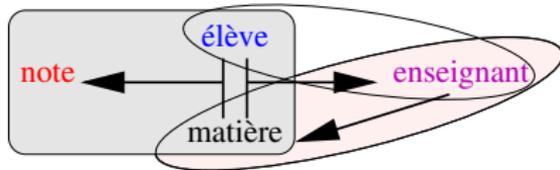
$$\begin{cases} \text{notes}(\underline{\text{élève}}, \text{matière}, \text{note}) \\ \text{Enseignement}(\underline{\text{élève}}, \underline{\text{enseignant}}, \text{matière}). \end{cases}$$

Enseignement (élève, enseignant, Matière) n'est pas FNBC.

$$\text{On la décompose : } \begin{cases} \text{notes}(\underline{\text{élève}}, \text{matière}, \text{note}), \\ \text{Enseignant}(\underline{\text{enseignant}}, \text{Matière}) \\ \text{Enseignement}(\underline{\text{élève}}, \underline{\text{enseignant}}). \end{cases}$$

Autres exemples.

- 1 Collège(élève, matière, enseignant, note) avec les DP suivantes :
- $$\begin{cases} (\text{élève}, \text{matière}) \rightarrow \text{enseignant}, \text{note} \\ \text{enseignant} \rightarrow \text{matière}. \end{cases}$$
- Les identifiants : (élève, matière) et (élève, enseignant).
3NF.



La relation collège est décomposée en :

$$\begin{cases} \text{notes}(\underline{\text{élève}}, \underline{\text{matière}}, \text{note}) \\ \text{Enseignement}(\underline{\text{élève}}, \underline{\text{enseignant}}, \text{matière}). \end{cases}$$

Enseignement (élève, enseignant, Matière) n'est pas FNBC.

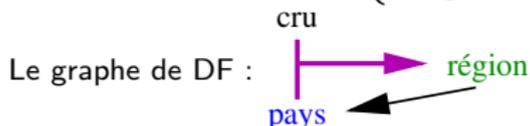
On la décompose : $\begin{cases} \text{notes}(\underline{\text{élève}}, \underline{\text{matière}}, \text{note}), \\ \text{Enseignant}(\underline{\text{enseignant}}, \text{Matière}) \\ \text{Enseignement}(\underline{\text{élève}}, \underline{\text{enseignant}}). \end{cases}$

La dépendance fonctionnelle (élève, matière) → enseignant est perdue.

- ② Vins(Cru, Pays, Région) avec les DF :
 $Region \rightarrow Pays$ et $(Cru, Pays) \rightarrow Region$

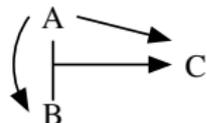
3NF

Vins est décomposée en : $\left\{ \begin{array}{l} \text{Crus } (\underline{Cru}, \text{Région}) \\ \text{Régions } (\underline{\text{Région}}, \text{Pays}). \end{array} \right.$



La dépendance fonctionnelle $(Cru, Pays) \rightarrow Region$ est perdue.

- ③ $R_1(A, B, C)$ avec les DF : $\left\{ \begin{array}{l} A \rightarrow B, C \\ (A, B) \rightarrow C \end{array} \right.$



La DF $A \rightarrow C$ est déduite de $A \rightarrow B$ et de $(A, B) \rightarrow C$.
 L'identifiant est A.

Les DF élémentaires ont donc toutes pour source une sur-clé
 (une clé pour $A \rightarrow B$, une sur-clé strict pour $(A, B) \rightarrow C$).

R_1 est donc en FNBC.

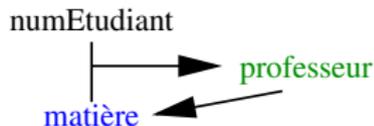
Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsNormalisation
d'une relationGraphe des DF
Couverture minimale
Décomposition
Les formes normales

Théorème de décomposition en FNBC : Toute relation admet au moins une décomposition en FNBC qui est sans perte d'informations ; cependant, une telle décomposition ne préserve généralement pas les dépendances fonctionnelles.

Exemple ne préservant pas les DF.

Enseignement(numEtudiant,
matière,
professeur)

avec les DF ci-contre.



Si on décompose en $\left\{ \begin{array}{l} \text{Enseigne}(\text{professeur}, \text{matière}) \\ \text{suit}(\underline{\text{numEtudiant}}, \text{professeur}) \end{array} \right.$
on ne préserve pas la DF (numEtudiant, matière) \rightarrow professeur.

En effet, on peut dire qu'un étudiant suit 2 cours sur la même matière avec 2 professeurs différents. Cela n'était pas possible dans le schéma initial.

Algorithme de décomposition FNBC (SPI) Soit un schéma de relation R , et un ensemble F de dépendances fonctionnelles s'appliquant sur R . Une décomposition SPI de R , qu'on appellera D , est construite de manière itérative :

- 1 D est initialisée à R ,
- 2 Soit T une relation de D qui ne soit pas FNBC.
→ il y a une DF $X \rightarrow A$ qui n'est pas de la forme $\text{clef} \rightarrow \text{attribut}$.
On décompose alors T en :
 - T_1 contenant A et les attributs de X , et
 - T_2 contenant tous les attributs de T sauf A ,
- 3 Dans D , la relation T est supprimée et on ajoute T_1 et T_2 , et on boucle sur 2 jusqu'à ce que toutes les relations soient FNBC.

Conclusion

- 1 Cet algorithme permet de trouver une décomposition sans perte d'information qui soit FNBC.
- 2 Il en existe généralement plusieurs et le résultat de l'algorithme dépend de l'ordre que l'on choisit pour traiter les dépendances fonctionnelles ne satisfaisant pas la propriété de Boyce-Codd.
- 3 La décomposition ne préserve pas nécessairement les dépendances fonctionnelles.

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphe des DF
Couverture minimale
Décomposition
Les formes normales

Exemple. On considère la relation

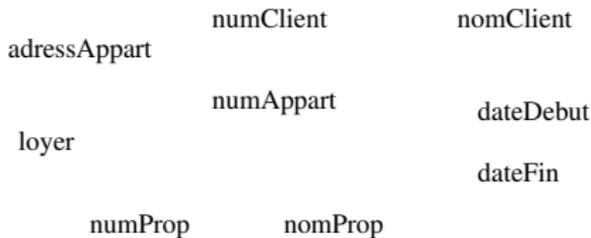
$T(\text{numClient}, \text{nomClient}, \text{numAppart}, \text{adrAppart},$
 $\text{dateDeb}, \text{dateFin}, \text{loyer}, \text{numProp}, \text{nomProp})$

	numClient	nomClient
adressAppart		
	numAppart	dateDebut
loyer		dateFin
	numProp	nomProp

Exemple. On considère la relation et les dépendances fonctionnelles suivantes :

$T(\text{numClient}, \text{nomClient}, \text{numAppart}, \text{adrAppart}, \text{dateDeb}, \text{dateFin}, \text{loyer}, \text{numProp}, \text{nomProp})$

- ① $\text{numClient} \rightarrow \text{nomClient};$
- ② $(\text{numClient}, \text{numAppart}) \rightarrow \text{dateDeb}, \text{dateFin};$
- ③ $\text{numAppart} \rightarrow \begin{cases} \text{adrAppart}, \\ \text{loyer}, \\ \text{numProp}, \\ \text{nomProp} \end{cases}$
- ④ $\text{numProp} \rightarrow \text{nomProp}$



Exemple. On considère la relation et les dépendances fonctionnelles suivantes :

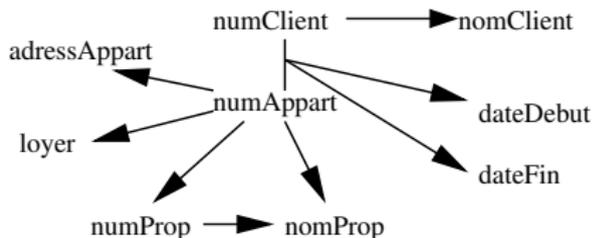
$T(\text{numClient}, \text{nomClient}, \text{numAppart}, \text{adrAppart},$
 $\text{dateDeb}, \text{dateFin}, \text{loyer}, \text{numProp}, \text{nomProp})$

① $\text{numClient} \rightarrow \text{nomClient};$

② $(\text{numClient}, \text{numAppart})$
 $\rightarrow \text{dateDeb}, \text{dateFin};$

③ $\text{numAppart} \rightarrow \left\{ \begin{array}{l} \text{adrAppart}, \\ \text{loyer}, \\ \text{numProp}, \\ \text{nomProp} \end{array} \right.$

④ $\text{numProp} \rightarrow \text{nomProp}$



Il n'y a qu'un seul identifiant : $(\text{numClient}, \text{numAppart})$.

Les DFs 1, 3 et 4 ne satisfont pas la propriété voulue.

Exemple. On considère la relation et les dépendances fonctionnelles suivantes :

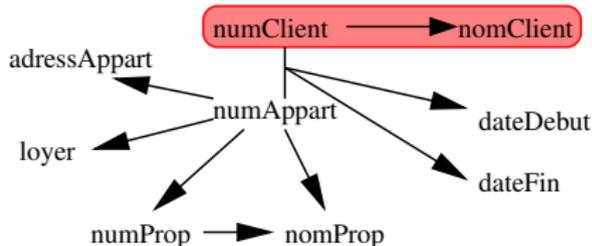
$T(\text{numClient}, \text{nomClient}, \text{numAppart}, \text{adrAppart},$
 $\text{dateDeb}, \text{dateFin}, \text{loyer}, \text{numProp}, \text{nomProp})$

① $\text{numClient} \rightarrow \text{nomClient};$

② $(\text{numClient}, \text{numAppart})$
 $\rightarrow \text{dateDeb}, \text{dateFin};$

③ $\text{numAppart} \rightarrow \left\{ \begin{array}{l} \text{adrAppart}, \\ \text{loyer}, \\ \text{numProp}, \\ \text{nomProp} \end{array} \right.$

④ $\text{numProp} \rightarrow \text{nomProp}$



Il n'y a qu'un seul identifiant : $(\text{numClient}, \text{numAppart})$.

Les DFs 1, 3 et 4 ne satisfont pas la propriété voulue.

● 1ère étape : DF 1. On obtient

$R_1(\text{numClient}, \text{nomClient})$

Exemple. On considère la relation et les dépendances fonctionnelles suivantes :

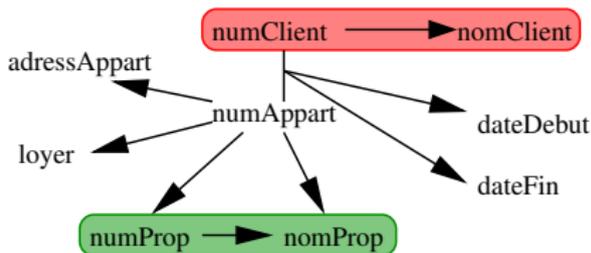
$T(\text{numClient}, \text{nomClient}, \text{numAppart}, \text{adrAppart}, \text{dateDeb}, \text{dateFin}, \text{loyer}, \text{numProp}, \text{nomProp})$

① $\text{numClient} \rightarrow \text{nomClient};$

② $(\text{numClient}, \text{numAppart}) \rightarrow \text{dateDeb}, \text{dateFin};$

③ $\text{numAppart} \rightarrow \left\{ \begin{array}{l} \text{adrAppart}, \\ \text{loyer}, \\ \text{numProp}, \\ \text{nomProp} \end{array} \right.$

④ $\text{numProp} \rightarrow \text{nomProp}$



Il n'y a qu'un seul identifiant : $(\text{numClient}, \text{numAppart})$.

Les DFs 1, 3 et 4 ne satisfont pas la propriété voulue.

● 1ère étape : DF 1. On obtient

$R1(\text{numClient}, \text{nomClient})$

● 2ème étape : DF 4.

$\left\{ \begin{array}{l} R1(\text{numClient}, \text{nomClient}) \\ R2(\text{numProp}, \text{nomProp}) \end{array} \right.$

Exemple. On considère la relation et les dépendances fonctionnelles suivantes :

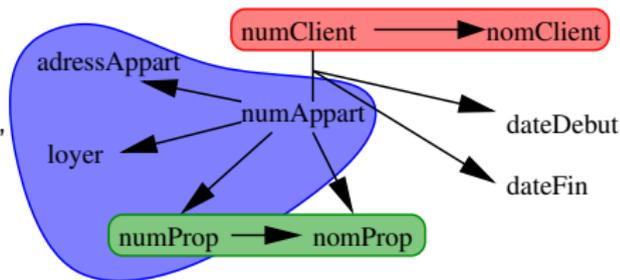
$T(\text{numClient}, \text{nomClient}, \text{numAppart}, \text{adrAppart}, \text{dateDeb}, \text{dateFin}, \text{loyer}, \text{numProp}, \text{nomProp})$

① $\text{numClient} \rightarrow \text{nomClient};$

② $(\text{numClient}, \text{numAppart}) \rightarrow \text{dateDeb}, \text{dateFin};$

③ $\text{numAppart} \rightarrow \left\{ \begin{array}{l} \text{adrAppart}, \\ \text{loyer}, \\ \text{numProp}, \\ \text{nomProp} \end{array} \right.$

④ $\text{numProp} \rightarrow \text{nomProp}$



Il n'y a qu'un seul identifiant : $(\text{numClient}, \text{numAppart})$.

Les DFs 1, 3 et 4 ne satisfont pas la propriété voulue.

● 1ère étape : DF 1. On obtient $R1(\text{numClient}, \text{nomClient})$

● 2ème étape : DF 4. $\left\{ \begin{array}{l} R1(\text{numClient}, \text{nomClient}) \\ R2(\text{numProp}, \text{nomProp}) \end{array} \right.$

● 3ème étape : DF 3. $\left\{ \begin{array}{l} R1(\text{numClient}, \text{nomClient}), \\ R2(\text{numProp}, \text{nomProp}), \\ R3(\text{numAppart}, \text{adrAppart}, \text{loyer}, \text{numProp}) \\ T'''(\text{numClient}, \text{numAppart}, \text{dateD}, \text{dateF}) \end{array} \right.$

Exemple. On considère la relation et les dépendances fonctionnelles suivantes :

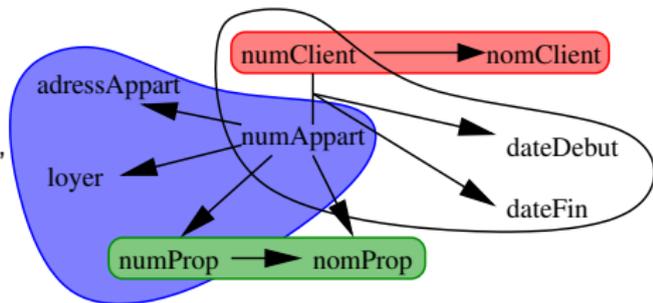
$T(\text{numClient}, \text{nomClient}, \text{numAppart}, \text{adrAppart}, \text{dateDeb}, \text{dateFin}, \text{loyer}, \text{numProp}, \text{nomProp})$

① $\text{numClient} \rightarrow \text{nomClient};$

② $(\text{numClient}, \text{numAppart}) \rightarrow \text{dateDeb}, \text{dateFin};$

③ $\text{numAppart} \rightarrow \left\{ \begin{array}{l} \text{adrAppart}, \\ \text{loyer}, \\ \text{numProp}, \\ \text{nomProp} \end{array} \right.$

④ $\text{numProp} \rightarrow \text{nomProp}$



Il n'y a qu'un seul identifiant : $(\text{numClient}, \text{numAppart})$.

Les DFs 1, 3 et 4 ne satisfont pas la propriété voulue.

● 1ère étape : DF 1. On obtient

$R1(\text{numClient}, \text{nomClient})$

● 2ème étape : DF 4.

$\left\{ \begin{array}{l} R1(\text{numClient}, \text{nomClient}) \\ R2(\text{numProp}, \text{nomProp}) \end{array} \right.$

● 3ème étape : DF 3.

$\left\{ \begin{array}{l} R1(\text{numClient}, \text{nomClient}), \\ R2(\text{numProp}, \text{nomProp}), \\ R3(\text{numAppart}, \text{adrAppart}, \text{loyer}, \text{numProp}) \\ T'''(\text{numClient}, \text{numAppart}, \text{dateD}, \text{dateF}) \end{array} \right.$

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphe des DF
Couverture minimale
Décomposition
Les formes normales

Grappe
de DF

1FN

2FN

3FN

FNBC

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Grappe des DF
Couverture minimale
Décomposition
Les formes normales

Grappe
de DF

Détermination
des identifiants

1FN

2FN

3FN

FNBC

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

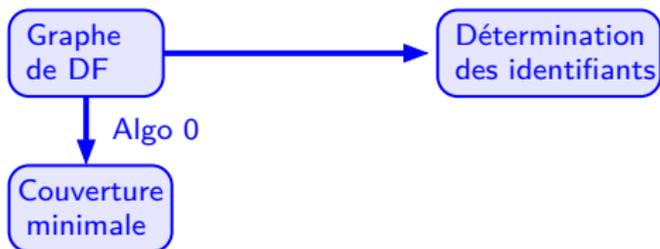
Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphe des DF
Couverture minimale
Décomposition
Les formes normales



1FN

2FN

3FN

FNBC

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

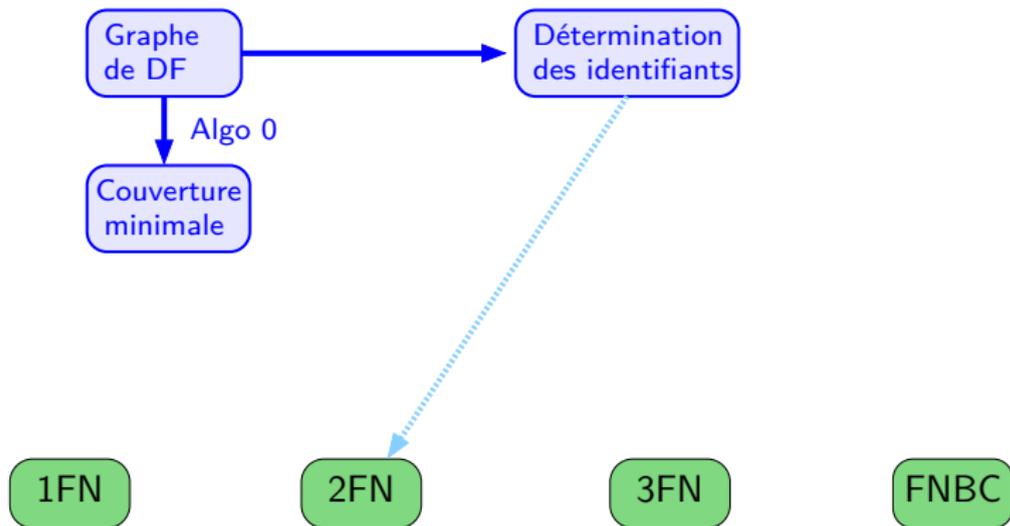
Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphe des DF
Couverture minimale
Décomposition
Les formes normales



Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

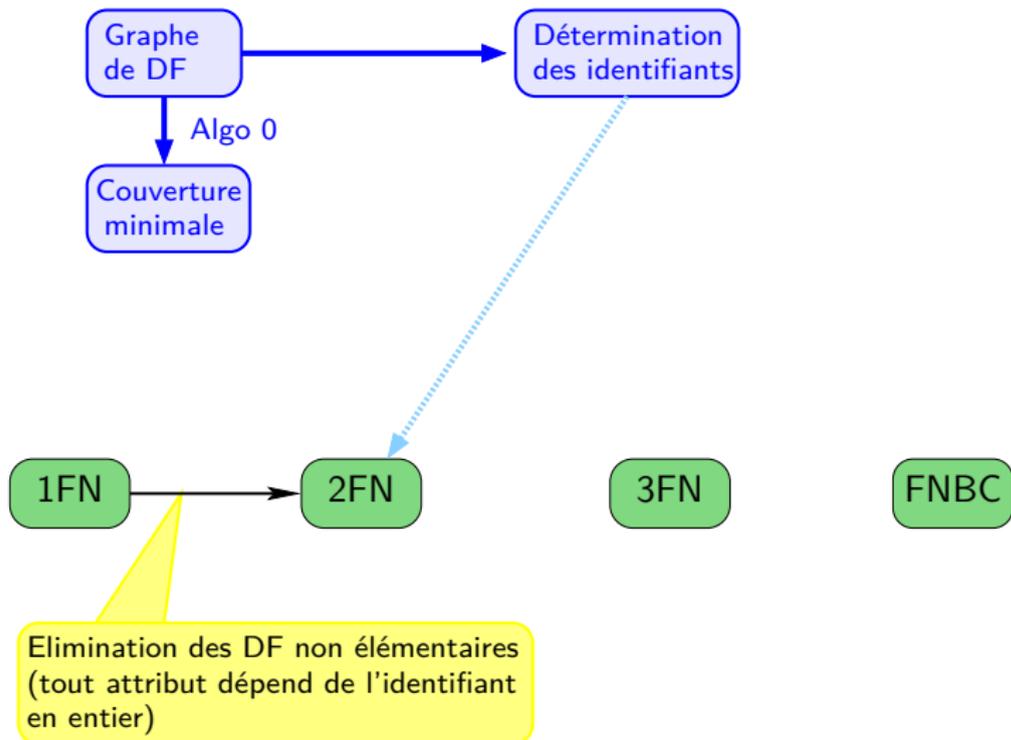
Notions
essentielles

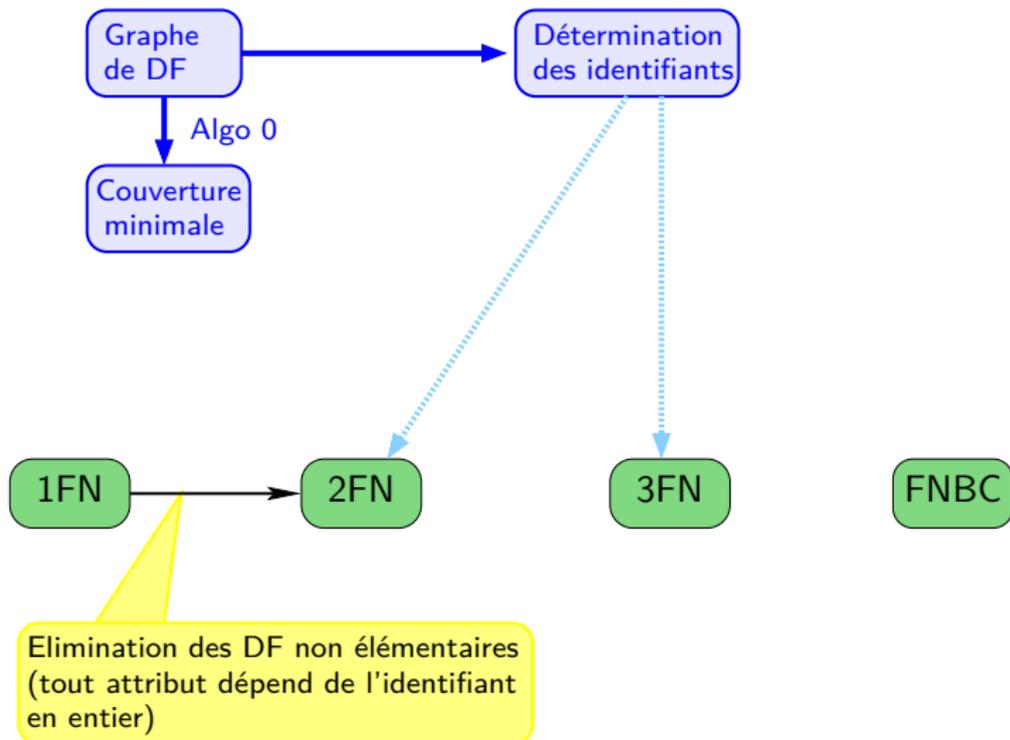
Algèbre
Relationnelle
+ SQL

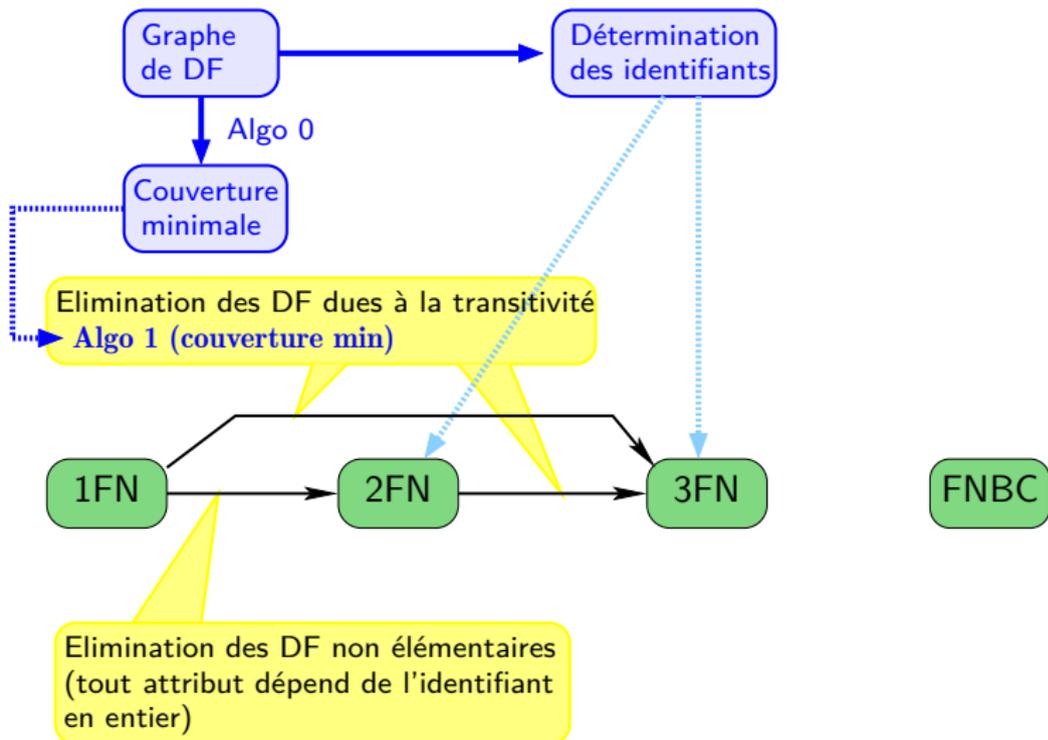
le modèle
Entités-
Associations

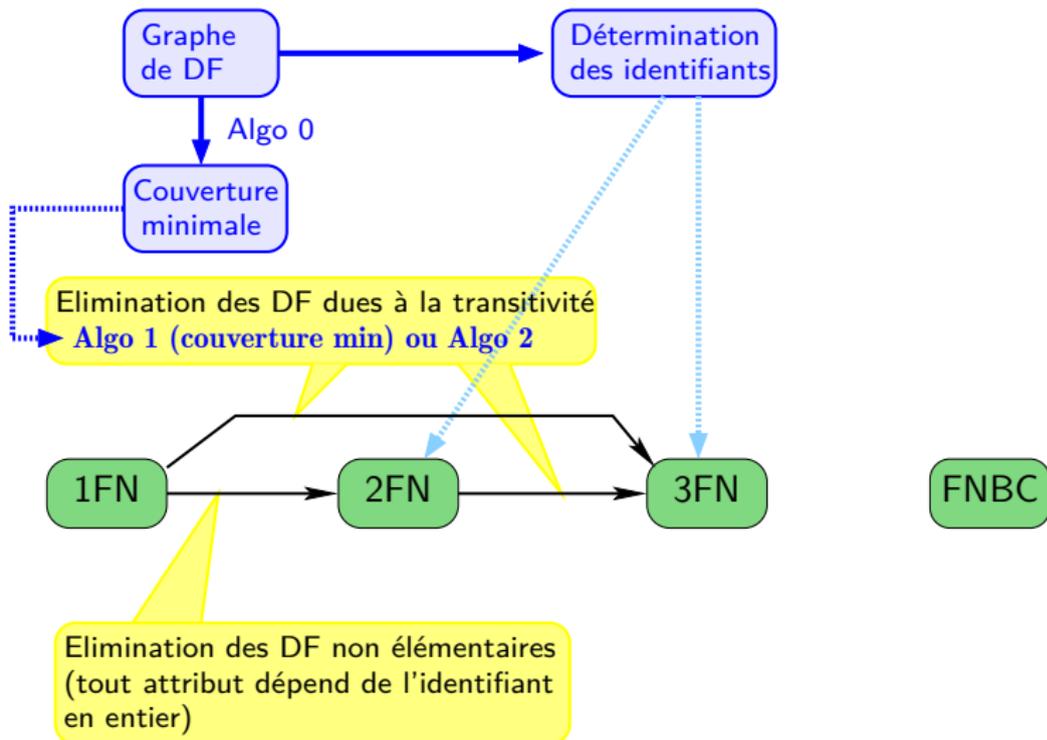
Normalisation
d'une relation

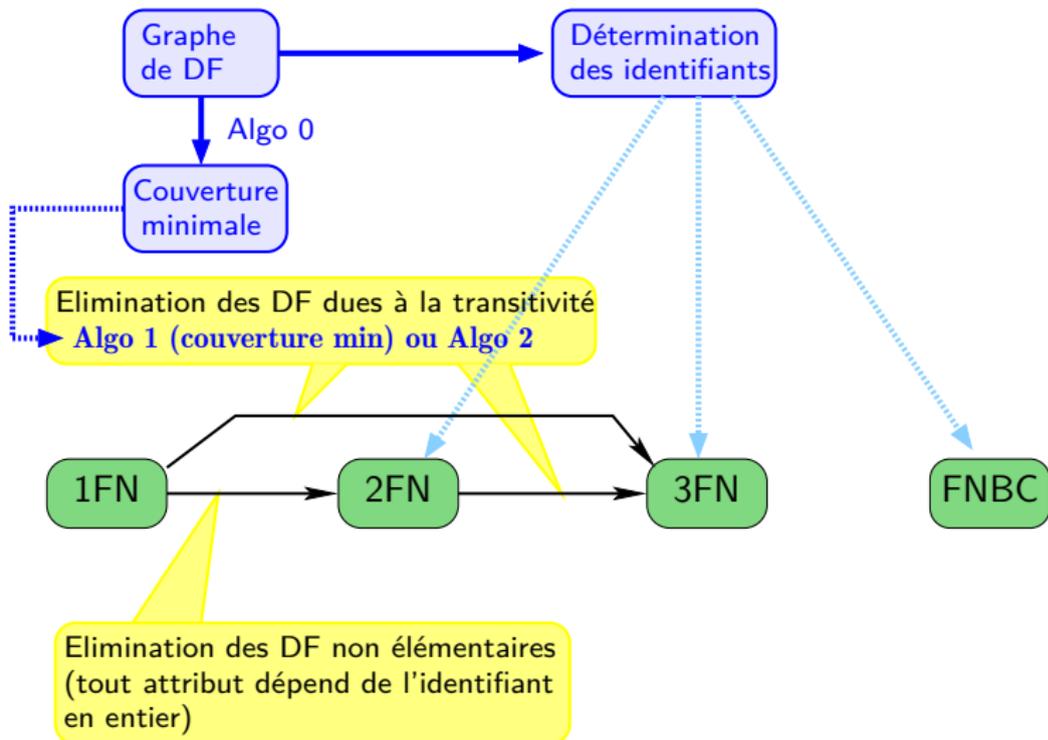
Graphe des DF
Couverture minimale
Décomposition
Les formes normales



Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsNormalisation
d'une relationGraphe des DF
Couverture minimale
Décomposition
Les formes normales

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsNormalisation
d'une relationGraphe des DF
Couverture minimale
Décomposition
Les formes normales

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsNormalisation
d'une relationGraphe des DF
Couverture minimale
Décomposition
Les formes normales

Introduction
aux BDRJean-Paul
Comet,
Nadia
Abchiche-
MimouniLes concepts
générauxNotions
essentiellesAlgèbre
Relationnelle
+ SQLle modèle
Entités-
AssociationsNormalisation
d'une relationGraphe des DF
Couverture minimale
Décomposition
Les formes normales

Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

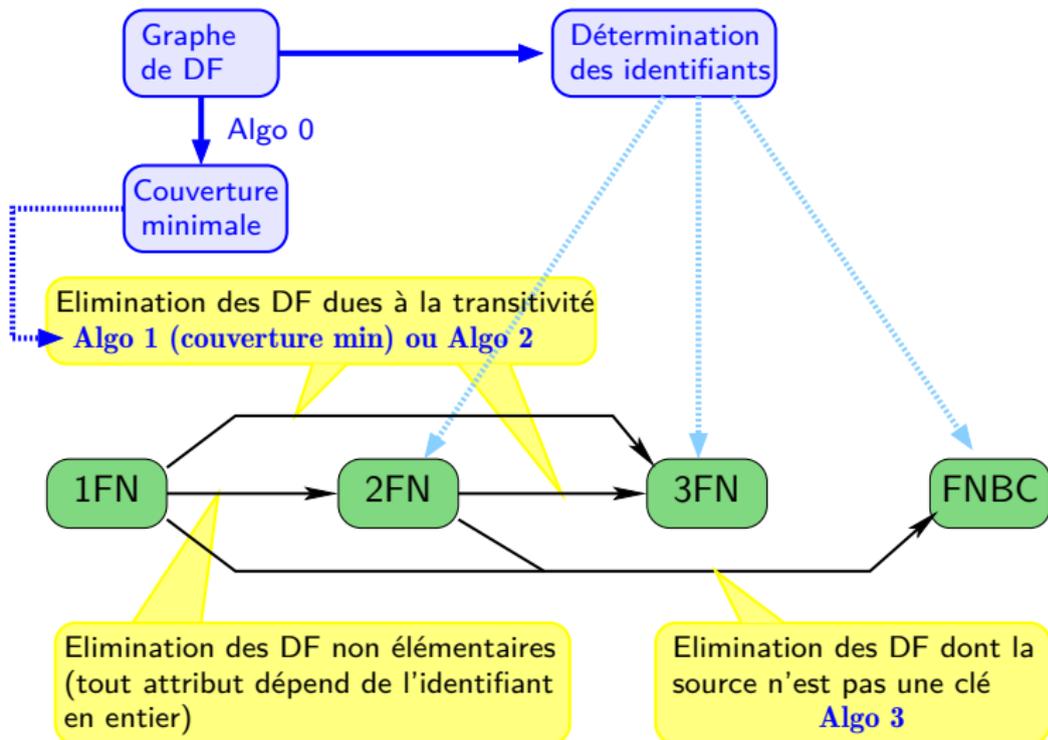
Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphes des DF
Couverture minimale
Décomposition
Les formes normales



Introduction
aux BDR

Jean-Paul
Comet,
Nadia
Abchiche-
Mimouni

Les concepts
généraux

Notions
essentielles

Algèbre
Relationnelle
+ SQL

le modèle
Entités-
Associations

Normalisation
d'une relation

Graphes des DF
Couverture minimale
Décomposition
Les formes normales

