

3. Analyse conceptuelle du domaine d'application

Version 2 - Septembre 2018

Support du chapitre 18, *Analyse conceptuelle du domaine d'application*
de l'ouvrage *Bases de données*, J-L Hainaut, Dunod 2018.

3. Analyse conceptuelle du domaine d'application

Contenu

- 3.1 Introduction**
- 3.2 Analyse des énoncés textuels**
- 3.3 Analyse des documents structurés**
- 3.4 Processus de correction d'un schéma**
- 3.5 Processus de normalisation d'un schéma**
- 3.6 Evaluation d'un schéma**
- 3.7 Intégration de schémas**
- 3.8 Analyse à l'aide d'un AGL**

3.1 Introduction

3.1 Introduction

Objectif

Construire un schéma conceptuel de qualité exprimant formellement les besoins en information des utilisateurs

1. Méthodologie des BD
2. Le modèle Entité-association
3. **Analyse conceptuelle**
4. Conception logique relationnelle

5. Conception physique
6. Production du code
7. Rétro-ingénierie

- 3.1 **Introduction**
- 3.2 Analyse des énoncés textuels
- 3.3 Analyse des documents structurés
- 3.4 Correction d'un schéma

- 3.5 Normalisation d'un schéma
- 3.6 Evaluation d'un schéma
- 3.7 Intégration de schémas
- 3.8 Analyse à l'aide d'un AGL

3.1 Introduction

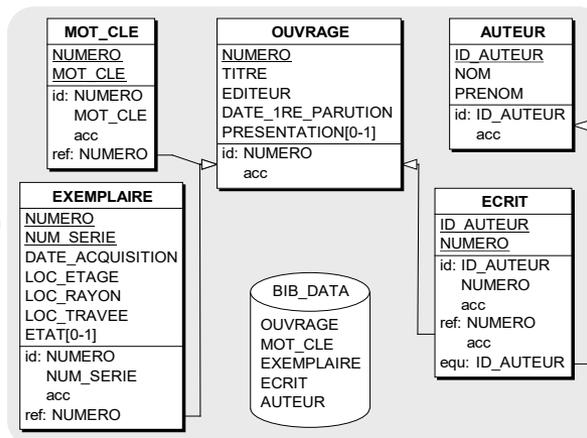
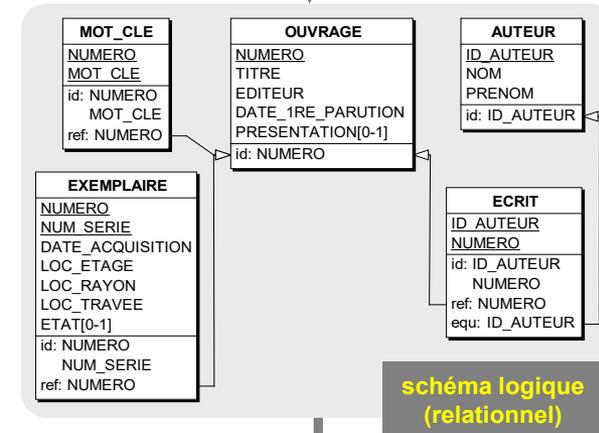
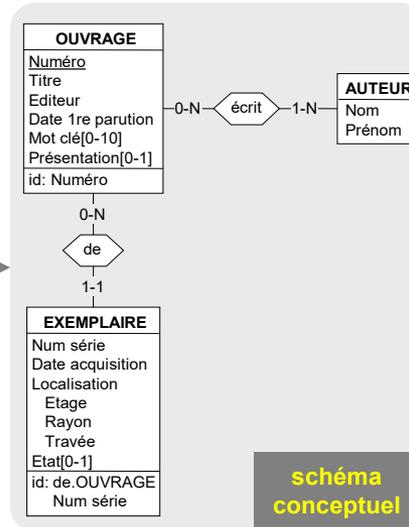
Un ouvrage est une oeuvre littéraire publiée. Il est caractérisé par son numéro identifiant, son titre, son éditeur, sa date de première parution, ses mots-clés (10 au maximum), une brève note de présentation (ces notes sont en cours de constitution), le nom et le prénom de ses auteurs. A un ouvrage correspondent un certain nombre d'exemplaires, qui en sont la matérialisation physique. ...

Conception de base de données

```

create database BIB
create dbspace BIB_DATA;
create table OUVRAGE (
  NUMERO char(18) not null,
  TITRE varchar(60) not null,
  EDITEUR char(32) not null,
  DATE_1RE_PARUTION date not null,
  PRESENTATION varchar(255),
  primary key (NUMERO) in BIB_DATA;
  ...
alter table EXEMPLAIRE add constraint FKDE
foreign key (NUMERO) references OUVRAGE;
...
create unique index IDOUVRAGE
on OUVRAGE (NUMERO);
...

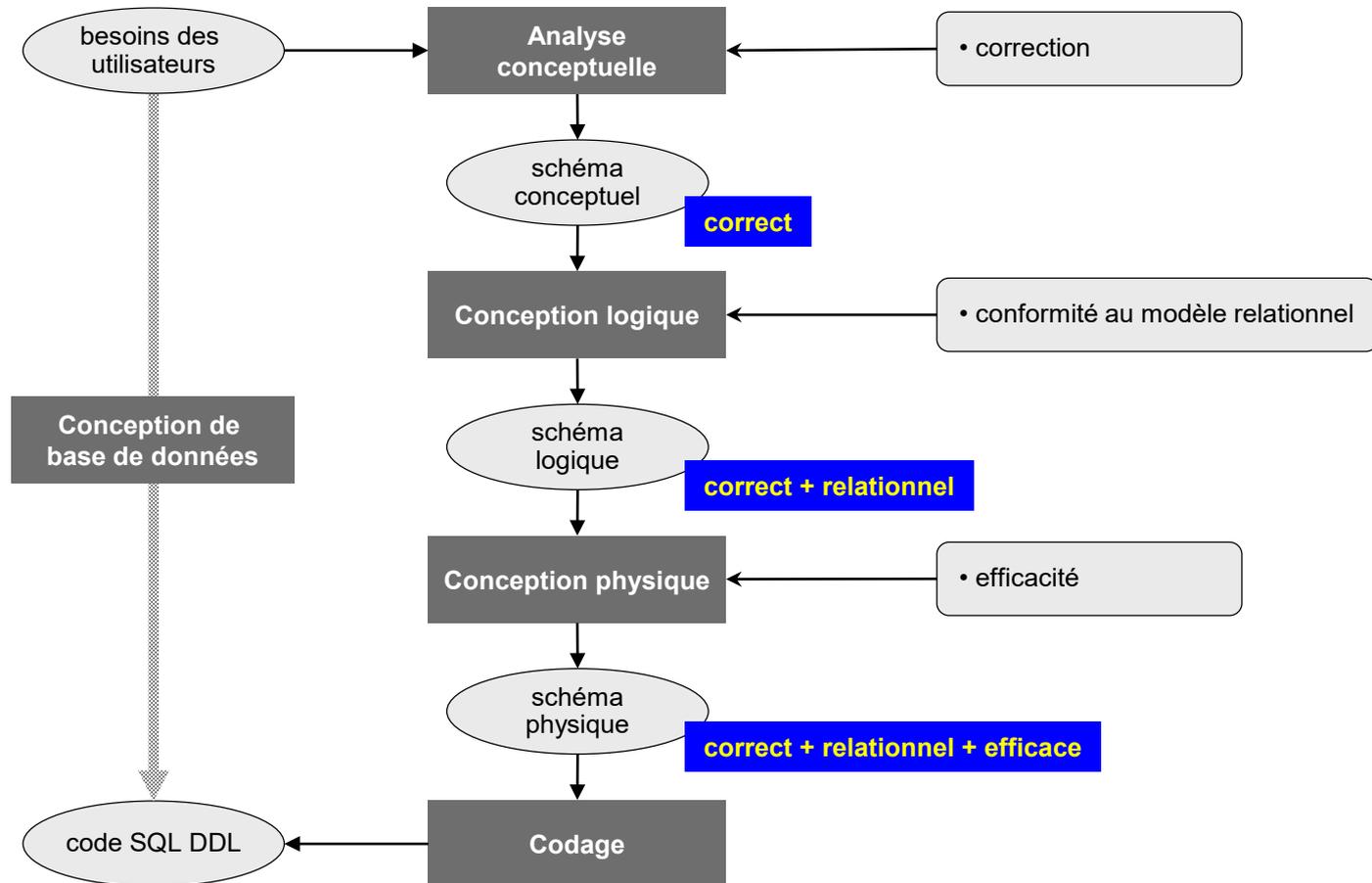
```



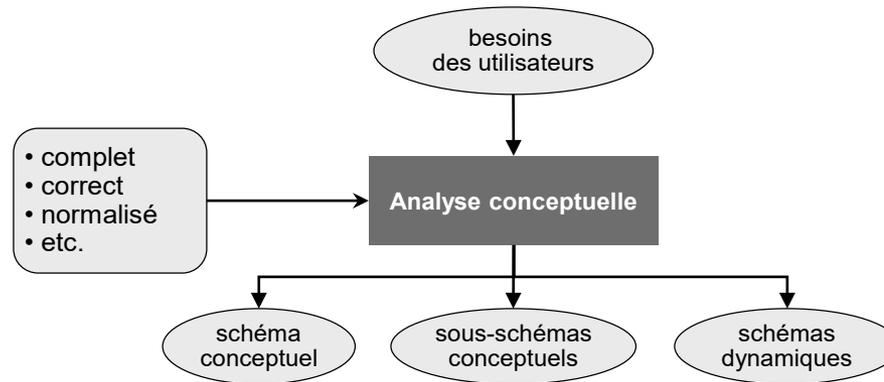
Rappel

3.1 Introduction

Rappel



3.1 Introduction



3.1 Introduction

Rappel

Analyse conceptuelle

Produit en entrée : besoins fonctionnels des utilisateurs

Processus : construction d'un schéma qui exprime formellement et de manière abstraite les besoins fonctionnels des utilisateurs

Produits en sortie : sous-schémas conceptuels; schéma conceptuel (global); diagrammes des traitements, etc.

Critères : complétude, correction, normalisation

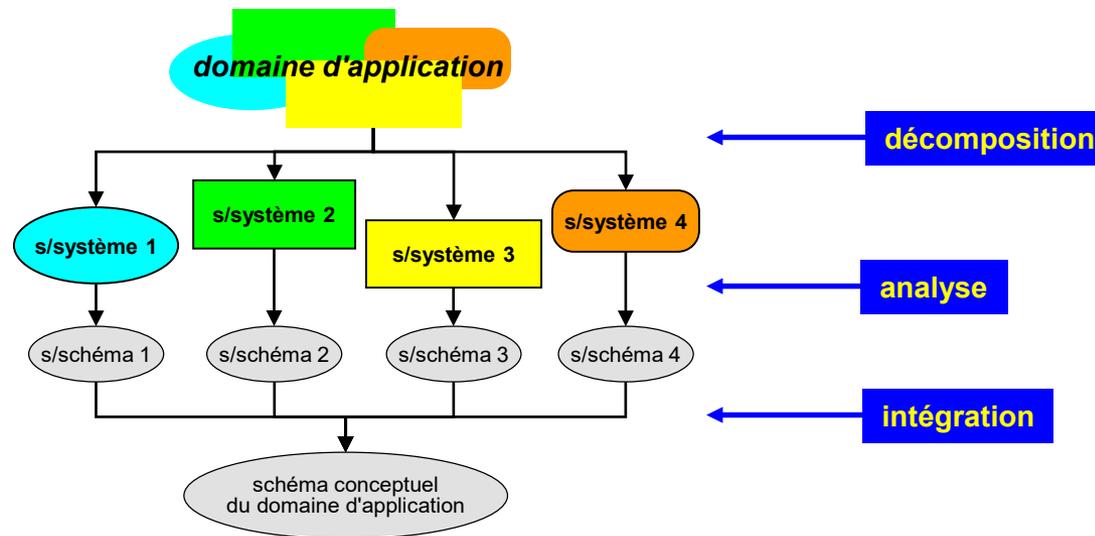
Expression du schéma conceptuel :
modèle Entité-association ou diagrammes de classes UML (faute de mieux)

3.1 Introduction

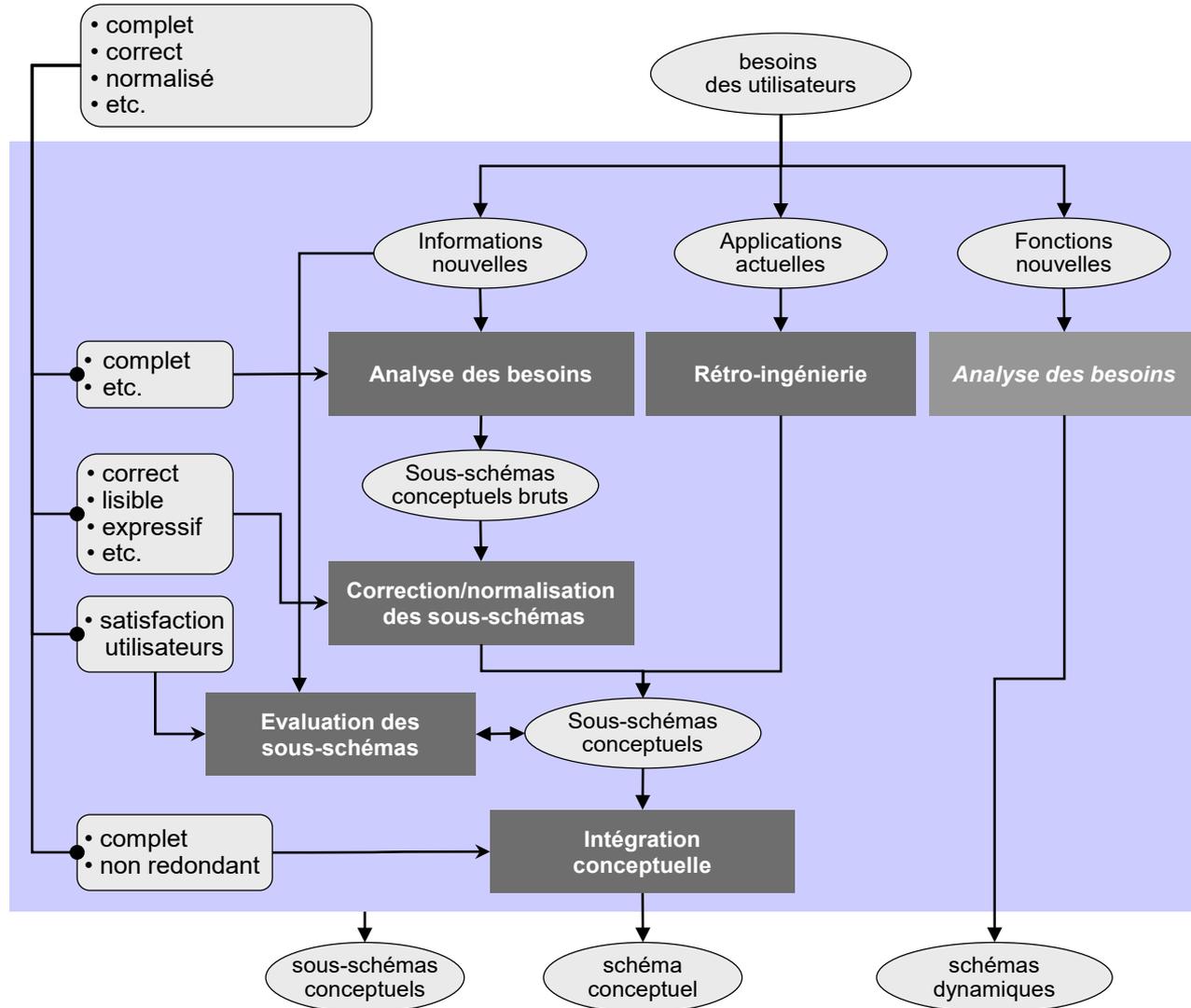
Analyse conceptuelle

Le domaine d'application est le plus souvent composé de sous-systèmes homogènes

- **décomposition** en sous-systèmes
- **analyse** de chaque sous-système sous forme d'un sous-schéma
- **intégration** des sous-schémas



3.1 Introduction



3.1 Introduction

Besoins des utilisateurs : les sources d'information

- **Textes** (*littéraires*, commerciaux, techniques, légaux, etc.)
- **Interviews**
- **Formulaires, documents, messages**
- **Système d'information existant**
 - **documentation**
 - **interface H/M (écrans, dialogues, etc.)**
 - **BD (rétro-ingénierie)**

3.2 Analyse des énoncés textuels

3.2 Analyse des énoncés textuels

Principes

1. Un texte est décomposé en propositions élémentaires

- existentielles
- relationnelles binaires
- relationnelles n-aires

2. L'information véhiculée par une proposition élémentaire est traduite en un fragment de schéma conceptuel

3. Les fragments de schéma sont intégrés progressivement

3.2 Analyse des énoncés textuels - Propositions unaires

Les propositions existentielles (unaires)

On considère les **clients**
de l'entreprise

CLIENT

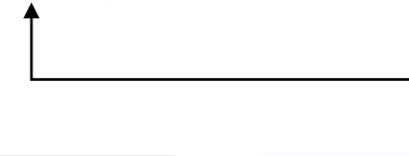
Il existe des **fournisseurs**

FOURNISSEUR

On désire enregistrer des
informations sur les **contrats**

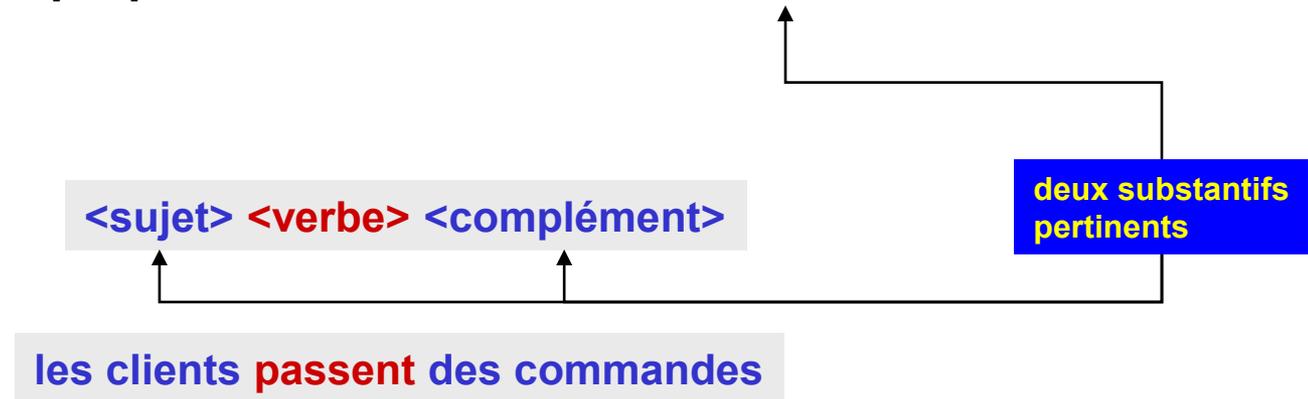
CONTRAT

un seul substantif
pertinent



3.2 Analyse des énoncés textuels - Propositions binaires

Les propositions relationnelles binaires



Selon l'intention du verbe :

- proposition d'action
- proposition de caractérisation
- proposition de classification

3.2 Analyse des énoncés textuels - Propositions binaires

Les propositions relationnelles binaires (**action**)

Les *clients* **passent**
des *commandes*



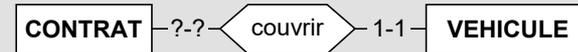
Toute *commande* **est passée**
par un *client*



Le *service* **a** un *directeur*



Chaque *véhicule* **est couvert**
par un *contrat*



3.2 Analyse des énoncés textuels - Propositions binaires

Les propositions relationnelles binaires (**caractérisation**)

Une *commande* **porte** un *numéro*

COMMANDE

Numéro

L'*abonné* **reçoit** un *numéro*
identifiant

ABONNE

Numéro

id: Numéro

Un *code* **est attribué**
à chaque *produit*

PRODUIT

Code

Tout *élève* **a** un ou plusieurs
prénoms

ELEVE

Prénom[1-N]

Les *clients* **sont identifiés**
par un *numéro*

CLIENT

Numéro

id: Numéro

3.2 Analyse des énoncés textuels - Propositions binaires

Les propositions relationnelles binaires (**classification**)

Le *directeur* est un *employé* de l'entreprise

EMPLOYE

DIRECTEUR

Un *client* est une *personne morale* ou une *personne physique*

CLIENT

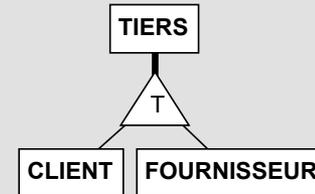
PERS-MORALE

PERS-PHYSIQUE

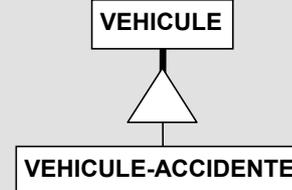
3.2 Analyse des énoncés textuels - Propositions binaires

Les propositions relationnelles binaires (classification)

Les *clients* et les *fournisseurs* constituent les *tiers* de l'entreprise

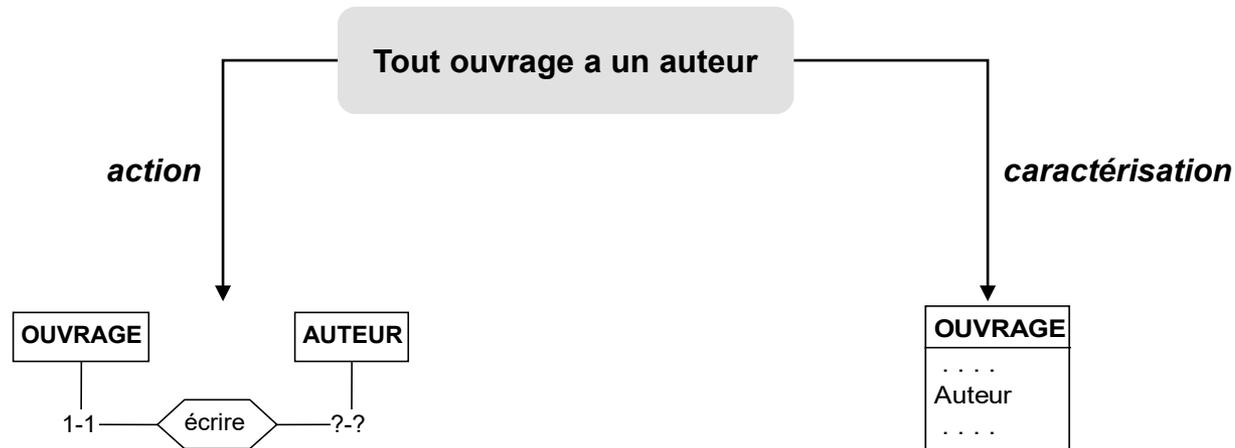


Les *véhicules* impliqués dans un accident



3.2 Analyse des énoncés textuels - Propositions binaires

Quid des ambiguïtés ?



Sans importance à ce stade : équivalence par les transformations

(on peut toujours transformer un attribut en un type d'entités et inversement)

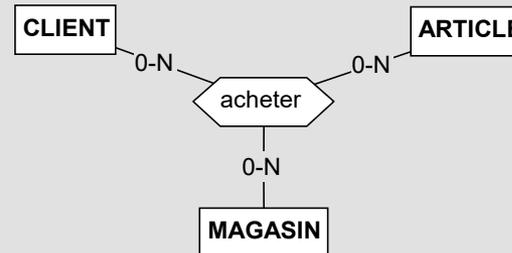
3.2 Analyse des énoncés textuels - Propositions n-aires

Les propositions relationnelles n-aires

plusieurs substantifs
pertinents

< sujet > < verbe > < complément 1 > < complément 2 > etc.

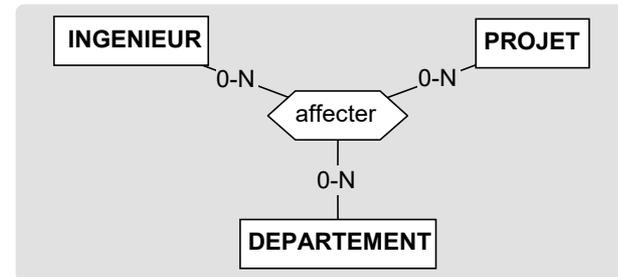
Les *clients* achètent
des *articles*
dans des *magasins*



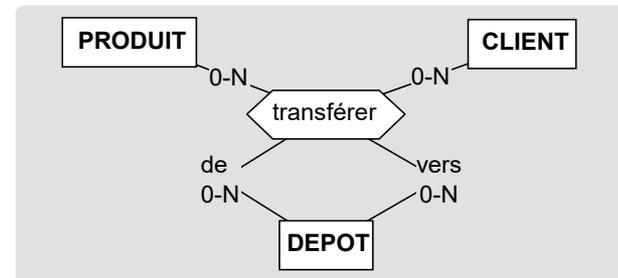
3.2 Analyse des énoncés textuels - Propositions n-aires

Les propositions relationnelles n-aires

Les *ingénieurs* sont affectés
à des *projets*
pour le compte de *départements*



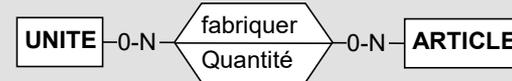
Un *produit* est transféré
d'un *dépôt*
vers un *autre*
à la demande d'un *client*



3.2 Analyse des énoncés textuels - Propositions n-aires

Les propositions relationnelles n-aires (hétérogènes)

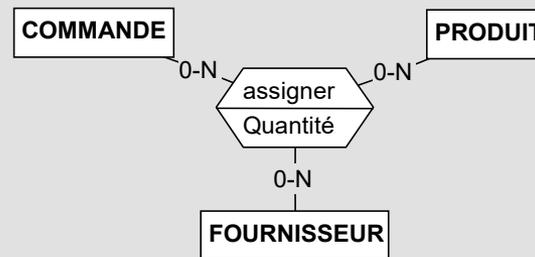
Une *unité* fabrique
un *article*
en quantité déterminée



Un *employé* a été nommé à la direction
du *service*
à une date déterminée



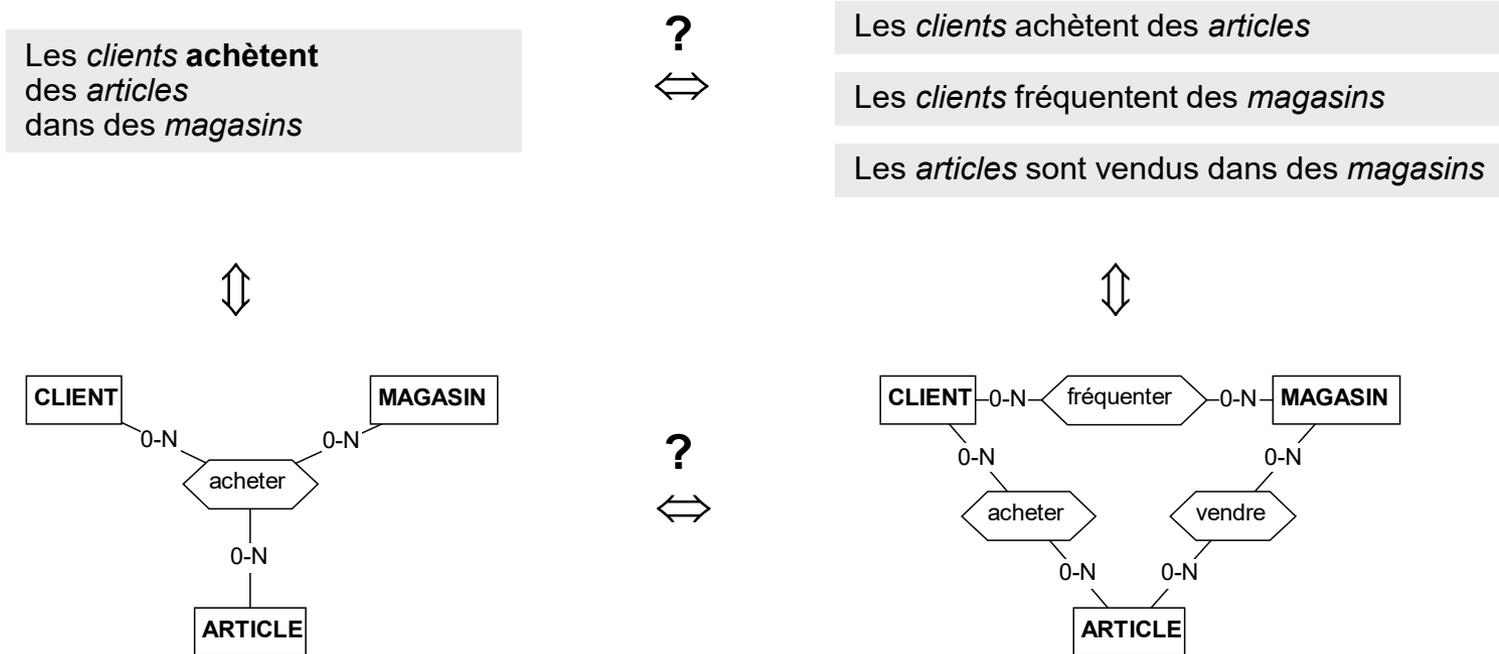
Une *commande* est assignée
à un *fournisseur*
pour une certaine quantité
d'un *produit*



3.2 Analyse des énoncés textuels - Propositions n-aires décomposables

Propositions n-aires décomposables et irréductibles

Une proposition n-aire est-elle équivalente à plusieurs propositions de degré inférieur ?



3.2 Analyse des énoncés textuels - Propositions n-aires décomposables

Propositions n-aires *décomposables* et *irréductibles*

Une proposition est décomposable (sans perte)
si le schéma qui en dérive est décomposable

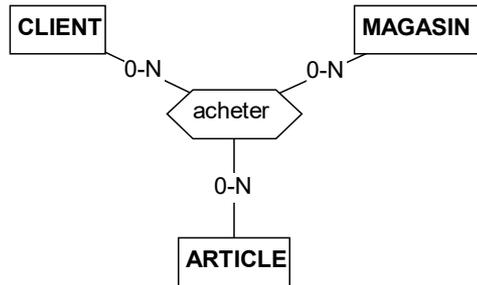
Décomposition = concept de la théorie de la normalisation.
Que nous dit cette théorie ?

"Une relation n'est décomposable sans perte
que selon une dépendance fonctionnelle"

Pas de dépendance fonctionnelle \Rightarrow pas de décomposition possible !

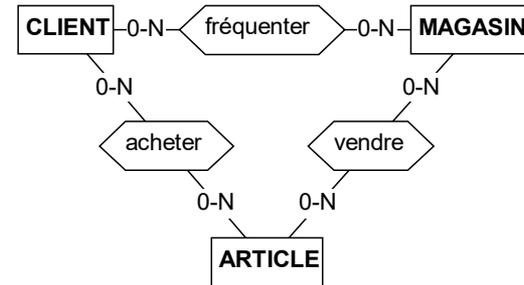
3.2 Analyse des énoncés textuels - Propositions n-aires décomposables

Propositions n-aires décomposables et irréductibles



acheter(CLIENT, ARTICLE, MAGASIN)

pas de dépendance fonctionnelle



acheter(CLIENT, ARTICLE)

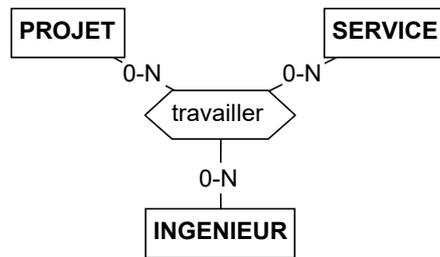
fréquenter(CLIENT, MAGASIN)

vendre(ARTICLE, MAGASIN)

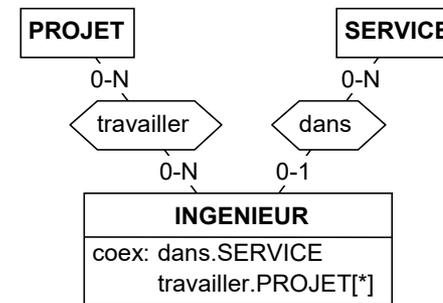


3.2 Analyse des énoncés textuels - Propositions n-aires décomposables

Propositions n-aires décomposables et irréductibles

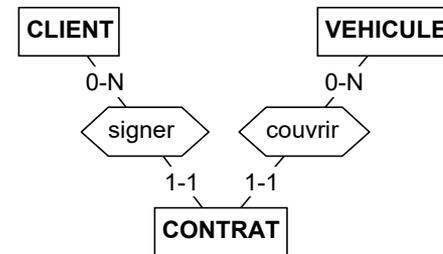
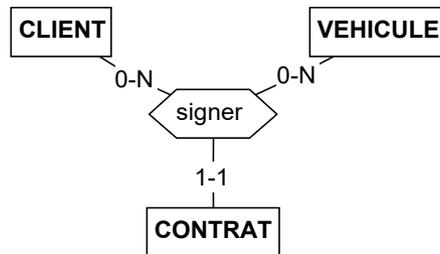


INGENIEUR → SERVICE



à démontrer

comment les noms ont-ils été choisis ?



à démontrer

3.2 Analyse des énoncés textuels - Exemple

Analyse de texte : un exemple

PRODUIT
NumPro
Description
id: NumPro

+

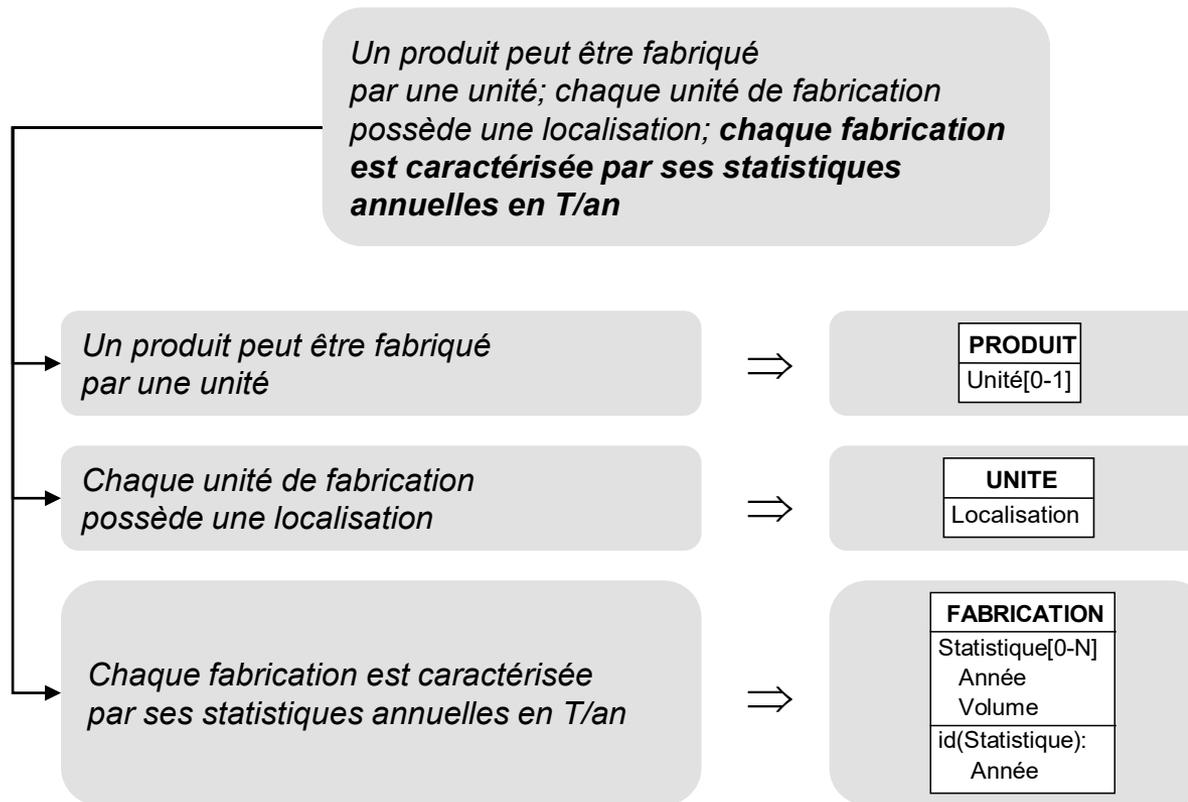
*Un produit peut être fabriqué
par une unité; chaque unité de fabrication
possède une localisation; chaque fabrication
est caractérisée par ses statistiques annuelles
en T/an*

=

?

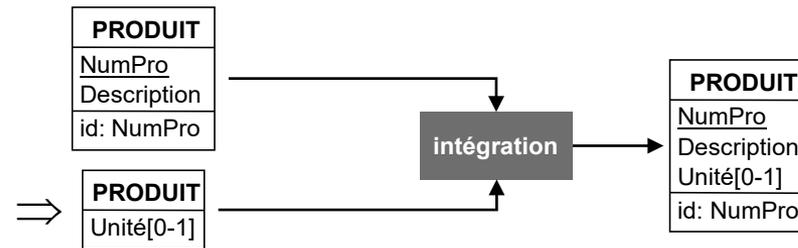
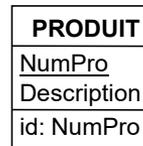
3.2 Analyse des énoncés textuels - Exemple

Exemple : décomposition et analyse



3.2 Analyse des énoncés textuels - Exemple

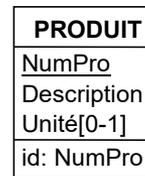
Exemple : intégration



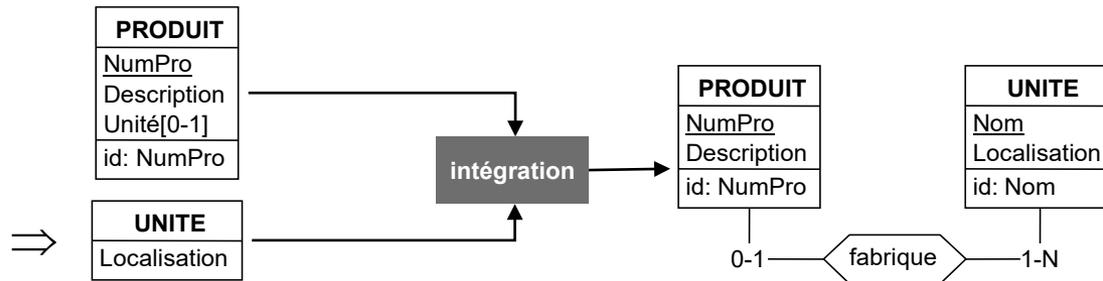
intégration : on anticipe !

3.2 Analyse des énoncés textuels - Exemple

Exemple : intégration

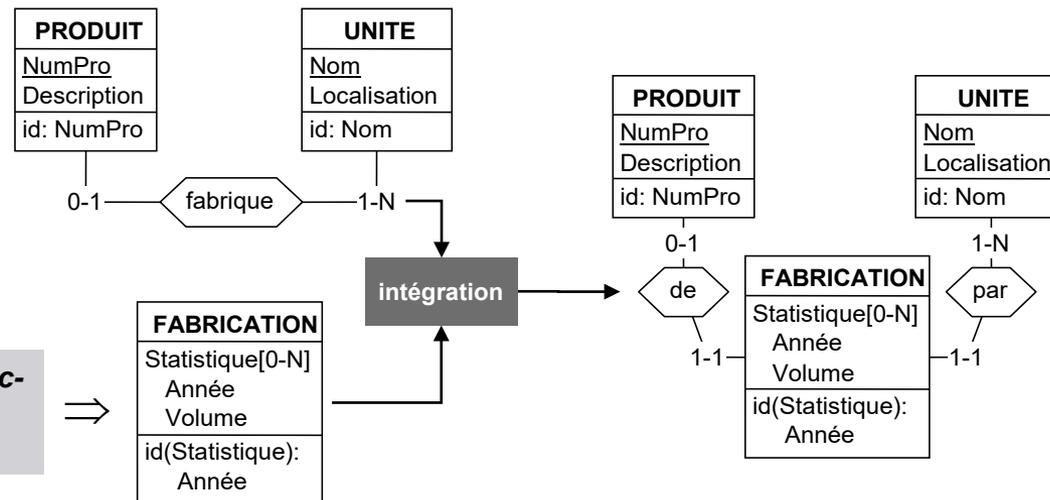
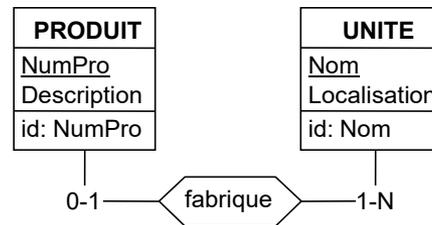


Chaque unité de fabrication possède une localisation



3.2 Analyse des énoncés textuels - Exemple

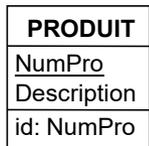
Exemple : intégration



Chaque fabrication est caractérisée par ses statistiques annuelles en T/an

3.2 Analyse des énoncés textuels - Exemple

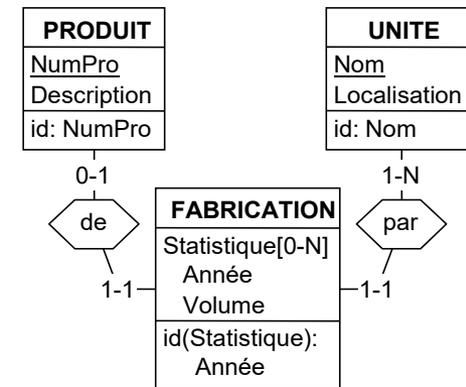
Exemple



+

Un produit peut être fabriqué par une unité; chaque unité de fabrication possède une localisation; chaque fabrication est caractérisée par ses statistiques annuelles en T/an

=



3.2 Analyse des énoncés textuels - Compléments

Compléments : analyse automatique du langage naturel

1. Conversion d'un texte en schéma

2. Extraction d'ontologies

3.2 Analyse des énoncés textuels - Compléments

Analyse automatique du langage naturel : exemple

Texte brut

A book is considered as a piece of literary, a scientific or technical document. It's identified by a number. Each book is characterized by its title, the first published date, keywords (maximum 6), an abstract and its bibliographic references. A book is also characterized by its physical state (new, used, worn, torn, damaged, etc). A technical document is characterized by a mandatory comment.

A book can be written by several authors. An author can have a first name, a birth date, and an origin (i.e., the organization (s)he belongs to when the book was written). For some authors, only the name is known. To be recorded in the database, an author must have written at least one book.

For each book, the library has acquired a certain number (0, 1, or more) of copies. The copies have distinct serial numbers. For each copy, the date it was acquired is known as well as its location in the library.

A copy can be borrowed by only one borrower. Borrowers are identified by a personal id. They are characterized by their phone numbers (at most five). A borrower can borrow at most five books.

[Ph. Thiran, *Acquisition de spécifications à partir du langage naturel*, mémoire de licence en Informatique, FUNDP, Septembre 1997.]

3.2 Analyse des énoncés textuels - Compléments

Analyse automatique du langage naturel : exemple

Texte normalisé

A book can be a literary-document, a scientific-document or a technical-document.

A book is identified by a number. Each book is characterized by its title, the first-published-date, keywords, an abstract and its bibliographic-references. A book can have at most 6 keywords. A book is characterized by its physical-state. A technical-document must have a comment.

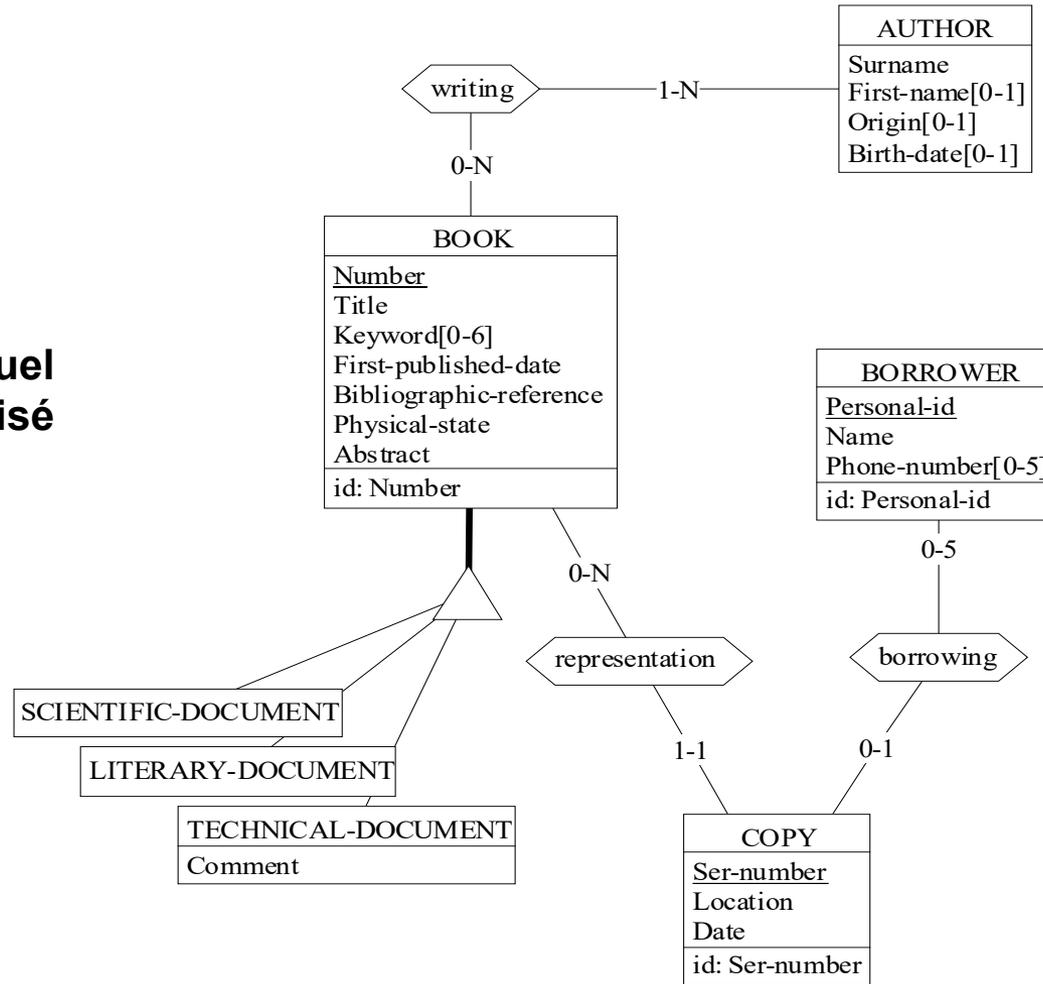
A book can be written by several authors. An author can have a first-name, a birth-date, and an origin. Each author has a surname. An author must write at least 1 book.

A book can be represented by several copies. The copies are identified by their ser-number. Each copy is characterized by its date and its location.

A copy can be borrowed by 1 borrower. Borrowers are identified by a personal-id. Borrowers are characterized by their name. Borrowers can have at most 5 phone-numbers. A borrower can borrow at most 5 books.

3.2 Analyse des énoncés textuels - Compléments

Schéma conceptuel normalisé



3.3 Analyse des documents structurés

3.3 Analyse des documents structurés

Principes

Les documents qui circulent dans l'organisation véhiculent des informations

- **formulaire papier (*)**
- **formulaire/messages électroniques (XML)**
- **écrans de saisie et de présentation (*)**
- **rapports**

⇒ repérage, identification et modélisation de ces informations

3.3 Analyse des documents structurés

Quelles informations tire-t-on d'un document structuré ?

- **les champs et leurs superstructures (agrégats, exclusivité, coexistence, etc.)**
- **les renseignements explicatifs (libellés, syntaxe, unicité, valeurs admissibles, conseils, aide, tutoriel, etc.)**
- **le comportement dynamique (syntaxe, unicité, valeurs admissibles)**
- **les exemples de valeurs (type, règles syntaxiques, interprétation)**

3.3 Analyse des documents structurés - Ecran de saisie

Ecran de saisie / présentation

The screenshot shows a software window titled "Inscription participant". The form is organized into several sections:

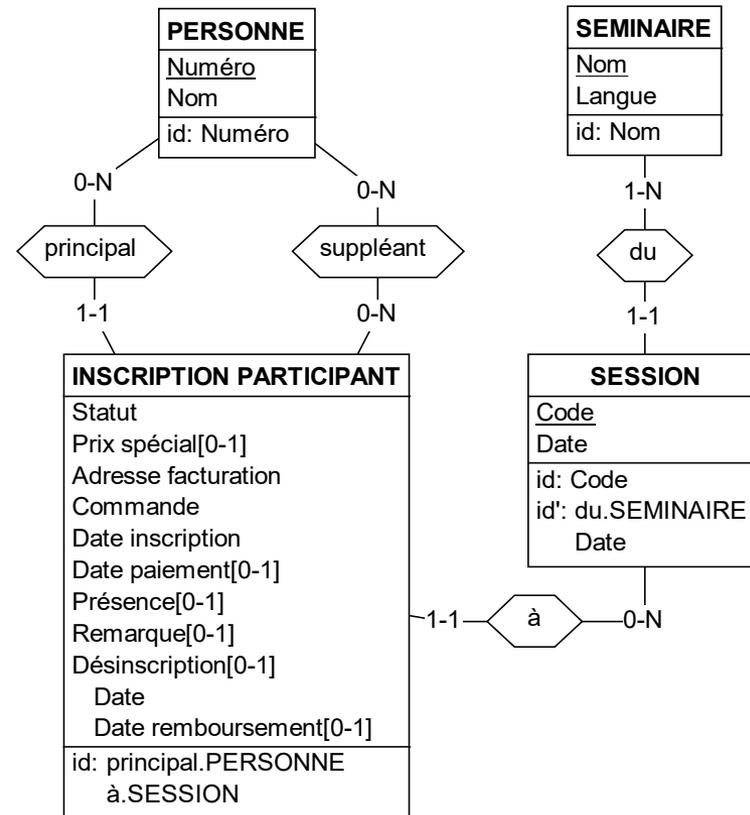
- Session**: Includes a "Séminaire" dropdown, "Nom" and "Langue" text boxes, and "Code" and "Date" text boxes.
- Participant**: Contains a "Principal" section with "Numéro" and "Nom" text boxes, and a "Suppléants" section with "Numéro" and "Nom" text boxes and a dropdown arrow.
- Administrative fields**: "Statut" and "Prix spécial" text boxes, "Adresse facturation" text box, "Commande" text box, "Date inscription" and "Date paiem." text boxes, a "Présence" checkbox, and a "Remarque" text box.
- Désinscription**: "Date" and "Date rembours." text boxes.
- Buttons**: "OK" and "Cancel" buttons at the bottom right.

3.3 Analyse des documents structurés - Ecran de saisie

contenu informationnel

INSCRIPTION PARTICIPANT
Session
Séminaire
Nom
Langue
Code
Date
Participant
Principal
Numéro
Nom
Suppléants[0-N]
Numéro
Nom
Statut
Prix spécial[0-1]
Adresse facturation
Commande
Date inscription
Date paiement[0-1]
Présence[0-1]
Remarque[0-1]
Désinscription[0-1]
Date
Date remboursement[0-1]

conceptualisation



3.3 Analyse des documents structurés - Formulaire papier

Formulaire papier

N.V. BELIMPORT S.A. 123, rue Neuve - 6743 Bertelmont TVA BE 546.901.442 - 059/45.56.67		N° 0030127				
		Date 24-07-2009				
Client G. Desmedt, rue Garamond 104 Verviers						
N° TVA _____		Num. Cmde _____				
DESCRIPTION	CODE	QTE	%TVA	PRIX TVAC	REMISE	TOTAL
BOOKMAN BS 620	NF-00453-34	1	21	65	0	65
Aqua Crystal 12L	FF-10067-01	5	6	25	0	25
Cafe COLOMBO 22	FF-77029-07	12	6	23,2	0	23,2
<i>Dispense de certification et de signature par autorisation 1645 du 04/08/1995. Exonération ecotaxe (Art 372, 373, 378)</i>						
Total montants TVA et marchandise HTVA par taux TVA						
TVA 21% 11,28	TVA 6% 8,37	TVA	TVA	Tot. TVA 19,65	Tot. March. 93,55	TOTAL 113,2
Marchandise 53,72	Marchandise 39,83	Marchandise	Marchandise			Reste à payer 0

1. Méthodologie des BD
2. Le modèle Entité-association
- 3. Analyse conceptuelle**
4. Conception logique relationnelle

5. Conception physique
6. Production du code
7. Rétro-ingénierie

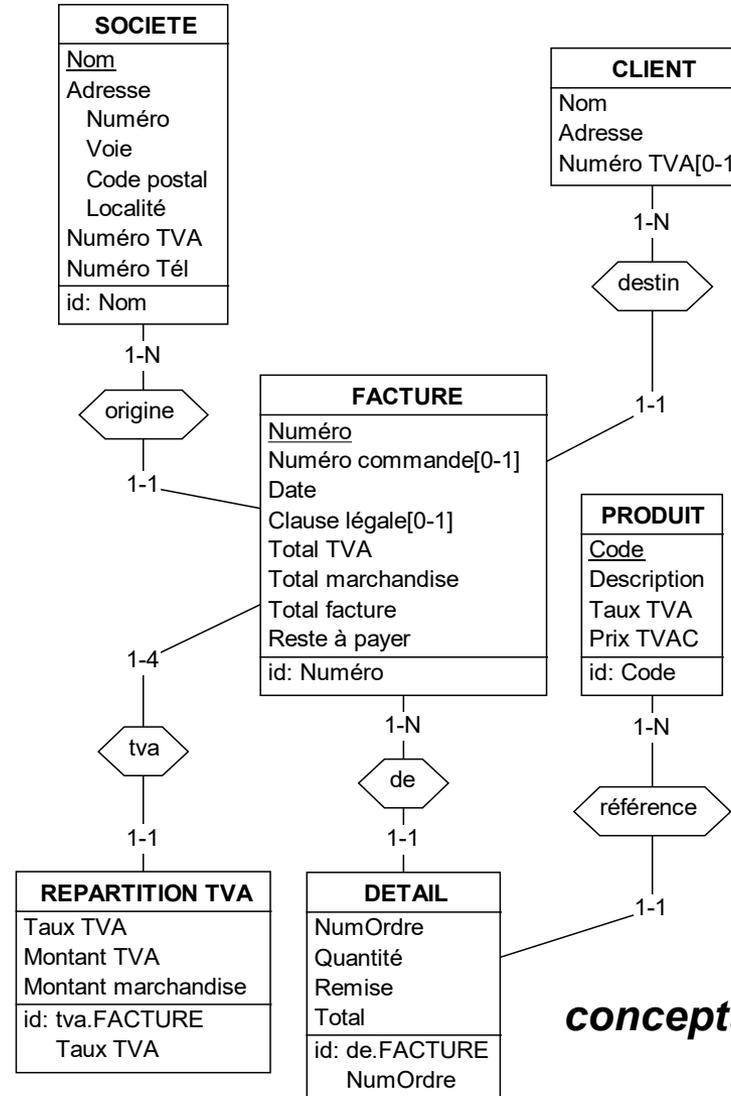
- 3.1 Introduction
- 3.2 Analyse des énoncés textuels
- 3.3 Analyse des documents structurés**
- 3.4 Correction d'un schéma

- 3.5 Normalisation d'un schéma
- 3.6 Evaluation d'un schéma
- 3.7 Intégration de schémas
- 3.8 Analyse à l'aide d'un AGL

3.3 Analyse des documents structurés - Formulaire papier

FACTURE BELIMPORT
<u>Numéro</u>
Date
Société
Nom
Adresse
Numéro
Voie
Code postal
Localité
Numéro TVA
Numéro Tél
Client
Nom
Adresse
Numéro TVA[0-1]
Numéro commande[0-1]
Détail[0-N] list
Description
Code
Quantité
Taux TVA
Prix TVAC
Remise
Total
Clause légale[0-1]
Répartition TVA[1-4]
Taux TVA
Montant TVA
Montant marchandise
Total TVA
Total marchandise
Total facture
Reste à payer
id: Numéro
id(Répartition TVA):
Taux TVA

**contenu
informationnel**



conceptualisation

3.3 Analyse des documents structurés - Compléments

Compléments

- **Utilisation des techniques de rétro-ingénierie**
- **Extension aux documents XML**
- **Boîtes de dialogue riches en information (boutons radio, onglets, etc.)**
- **Automatisation possible**
- **Utilisation des données : observation, induction, contre-exemples, etc.**

3.4 Processus de correction d'un schéma

3.4 Processus de correction d'un schéma

Un schéma conceptuel peut contenir des constructions qui

- **soit empêchent son implémentation**
- **soit empêchent l'utilisation normale de la base de données**

Deux phases

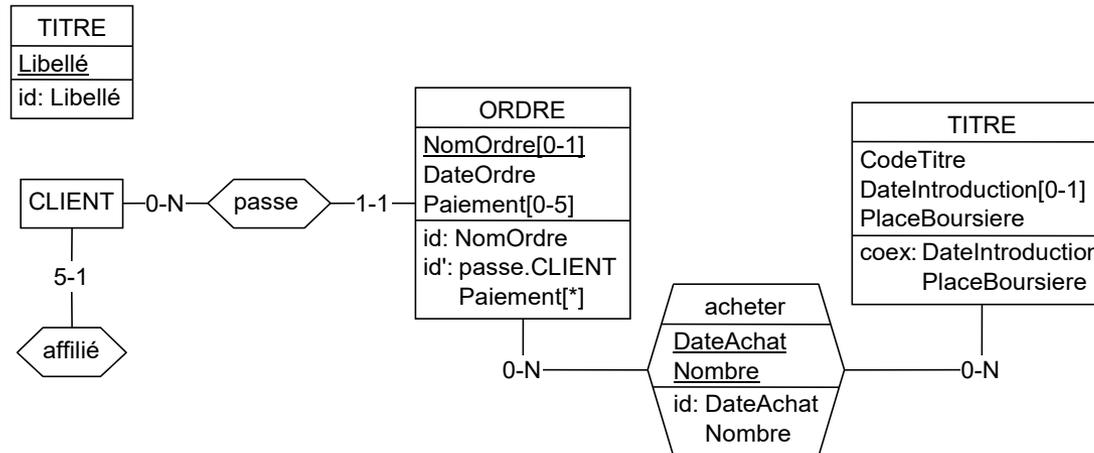
- **identifier les erreurs**
- **corriger les erreurs**

Trois phénomènes différents

- A. les **erreurs syntaxiques**
- B. les **constructions non satisfiables**
- C. les **constructions incohérentes**

3.4 Processus de correction d'un schéma - Erreurs syntaxiques

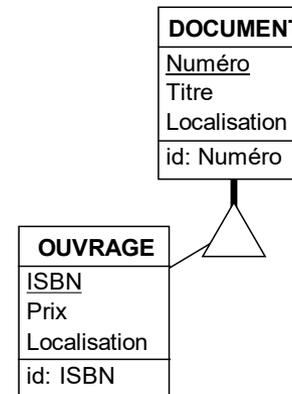
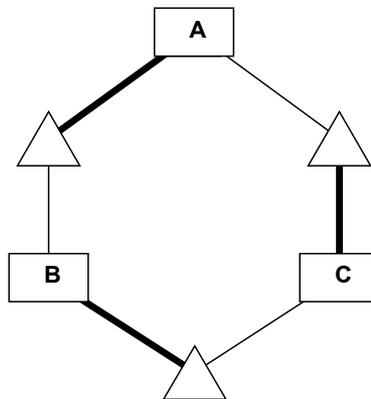
A. Les erreurs syntaxiques



chercher les 7 erreurs !

3.4 Processus de correction d'un schéma - Erreurs syntaxiques

A. Les erreurs syntaxiques



chercher les 3 erreurs

3.4 Processus de correction d'un schéma - *Constructions non satisfiables*

B. Les constructions non satisfiables

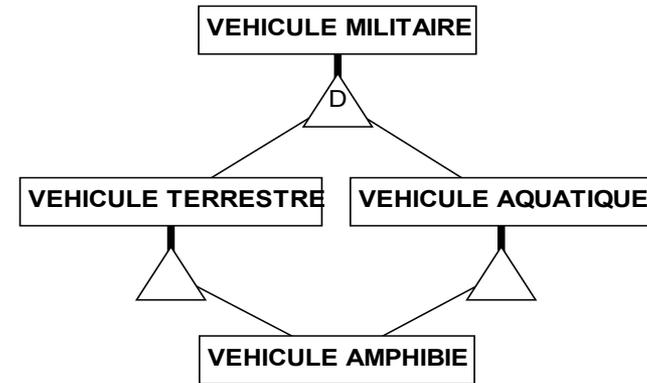
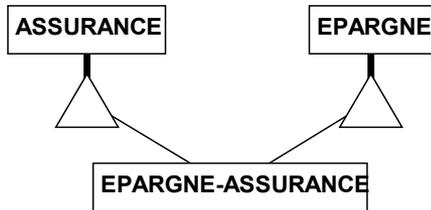
Il est impossible de créer des instances de C

ou

La seule instance valide de C est l'instance vide

3.4 Processus de correction d'un schéma - *Constructions non satisfiables*

B. Les constructions non satisfiables : sous types



3.4 Processus de correction d'un schéma - *Constructions non satisfiables*

B. Les constructions non satisfiables : contraintes d'existence

A
A1
A2[0-1]
A3[0-1]
coex: A2 A3
excl: A2 A3

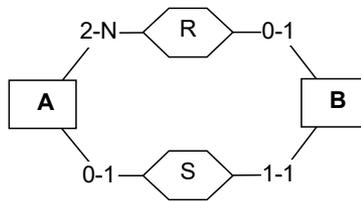
A2 et A3 non satisfiables

B
B1
B2[0-1]
B3[0-1]
B4[0-1]
coex: B2 B4
exact-1: B2 B3 B4
if: B3 B4

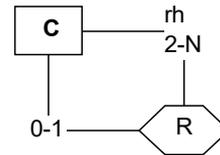
B non satisfiable

3.4 Processus de correction d'un schéma - *Constructions non satisfiables*

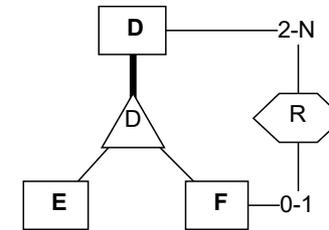
B. Les constructions non satisfiables : cardinalités



A et B non satisfiables



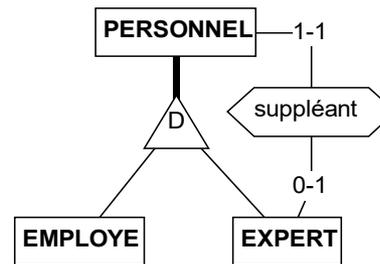
C non satisfiable



D, E, F non satisfiables

3.4 Processus de correction d'un schéma - *Constructions non satisfiables*

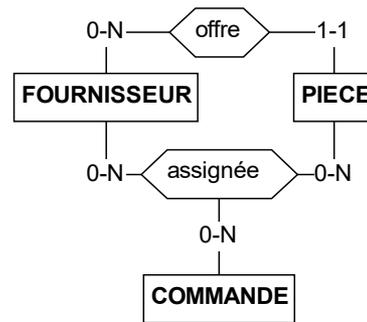
B. Les constructions non satisfiables : cardinalités



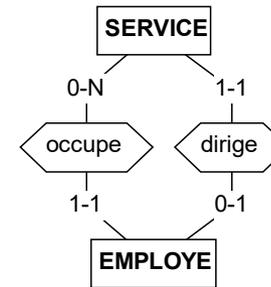
rien que des experts dans cette boîte ?

3.4 Processus de correction d'un schéma - *Constructions non satisfiables*

B. Les constructions non satisfiables : contraintes d'inclusion incohérentes



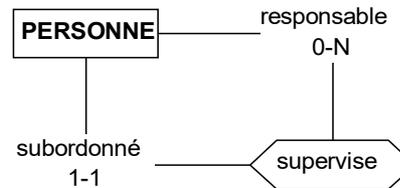
assignée[Fournisseur, Pièce] \subseteq offre



dirige \subseteq occupe

3.4 Processus de correction d'un schéma - *Constructions non satisfiables*

C. Les constructions non initialisables



Heu ... comment démarre-t-on ?

3.5 Processus de normalisation d'un schéma

3.5 Processus de normalisation d'un schéma

Un schéma peut être correct sans pour autant être satisfaisant pour ses utilisateurs.

Il doit en plus être *lisible, compréhensible, conforme aux bonnes pratiques* du métier.

Pourquoi ?

- pour faciliter son **utilisation** (par les programmeurs)
- pour faciliter sa **maintenance**
- pour faciliter son **évolution**

Un schéma lisible, compréhensible, ... est dit normalisé.

3.5 Processus de normalisation d'un schéma

La normalisation conceptuelle est une qualité malaisée à définir (contrairement à la normalisation relationnelle).

On se concentre sur des critères de qualité pour lesquels

- les défauts sont formellement identifiables
- il existe des techniques d'amélioration

On pourrait donc

- automatiser la détection des défauts
- automatiser la correction des défauts

*mais ceci reste
une question de recherche*

3.5 Processus de normalisation d'un schéma

On retient sept critères de normalisation

- 1. Minimalité**
- 2. Expressivité**
- 3. Absence de sur-spécifications**
- 4. Prévisibilité, régularité**
- 5. Absence de redondance**
- 6. Qualité graphique**
- 7. Respect des standards**

Remarque : ces critères sont parfois incompatibles !

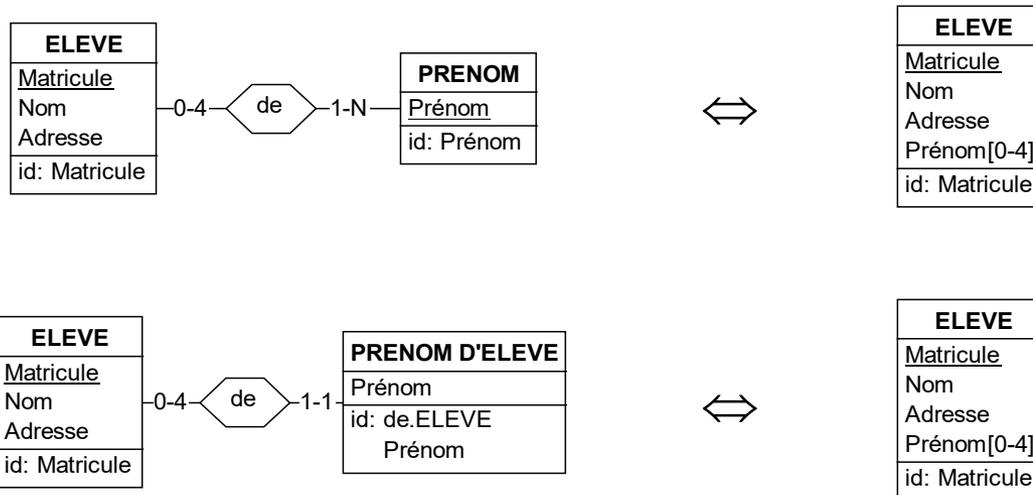
3.5 Processus de normalisation d'un schéma

1. Minimalité

Un type de faits doit être représenté **le plus simplement possible**;
Par exemple, il serait excessif de représenter une simple propriété par un type d'entités.

3.5 Processus de normalisation d'un schéma - *Minimalité*

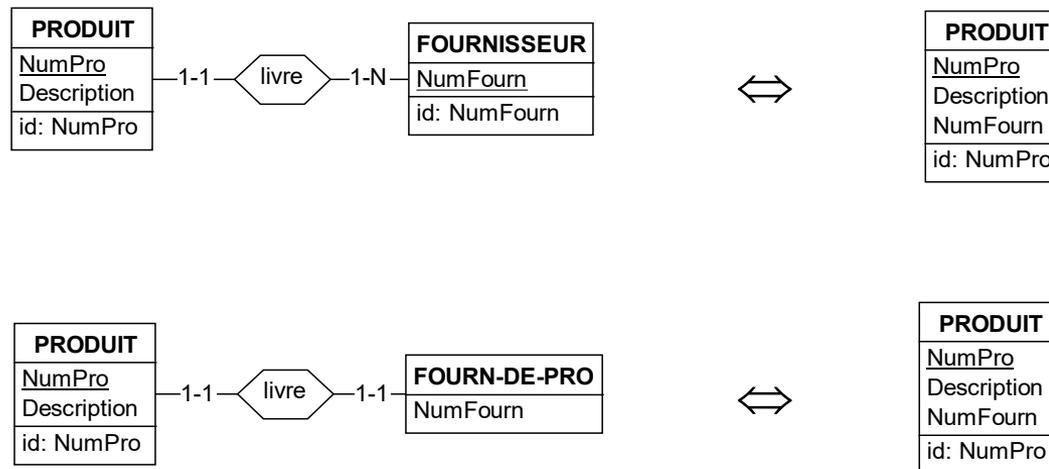
Type d'entités attribut



Type d'entités jouant le rôle d'un simple attribut

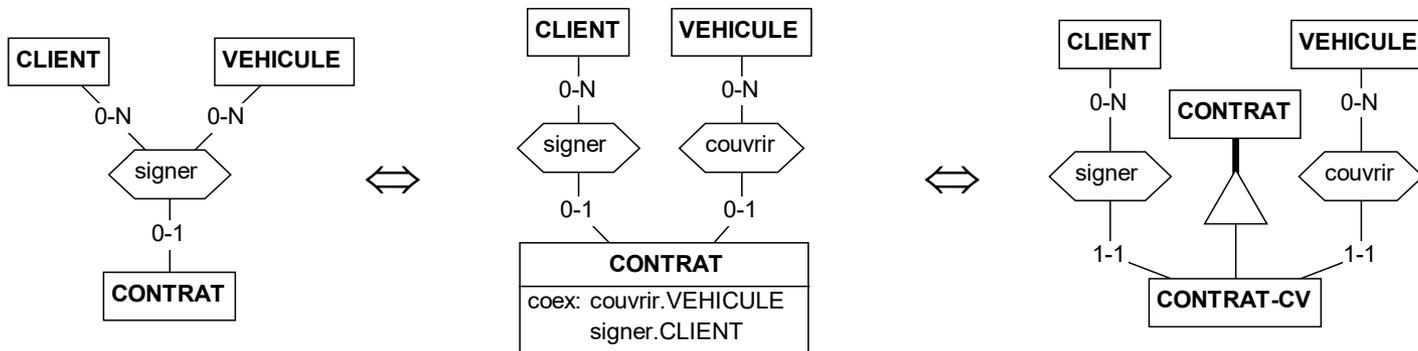
3.5 Processus de normalisation d'un schéma - *Minimalité*

Type d'entités attribut



3.5 Processus de normalisation d'un schéma - *Minimalité*

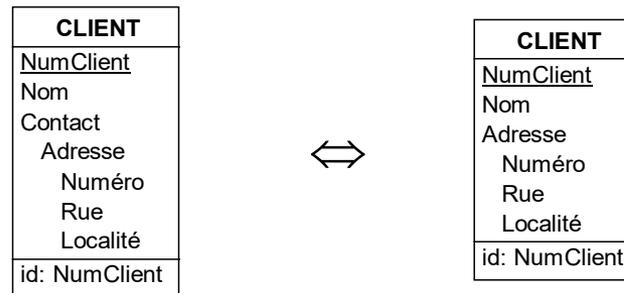
Type d'associations avec rôle [1-1]



Un TA n-aire est plus complexe à comprendre qu'un TA binaire

3.5 Processus de normalisation d'un schéma - *Minimalité*

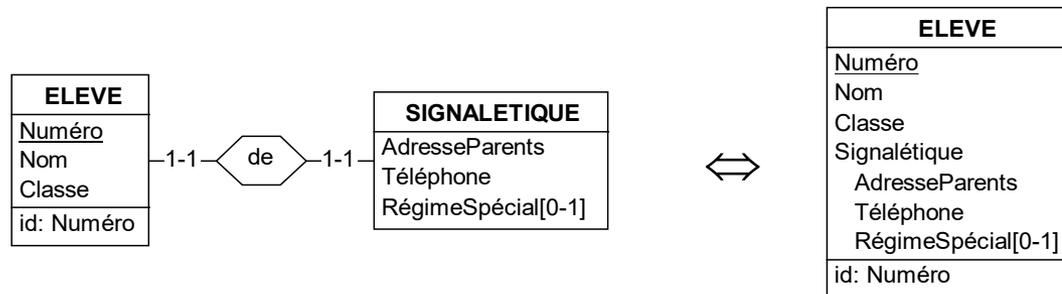
Attribut composé d'un seul composant



Vérifier si tous ces niveaux sont utiles

3.5 Processus de normalisation d'un schéma - *Minimalité*

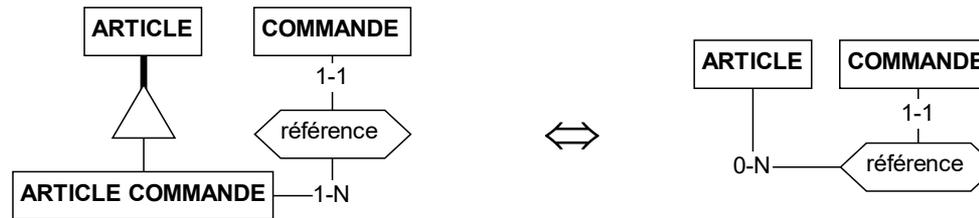
Type d'associations 1:1 à rôles obligatoires



Est-il utile de conserver cette décomposition ?

3.5 Processus de normalisation d'un schéma - *Minimalité*

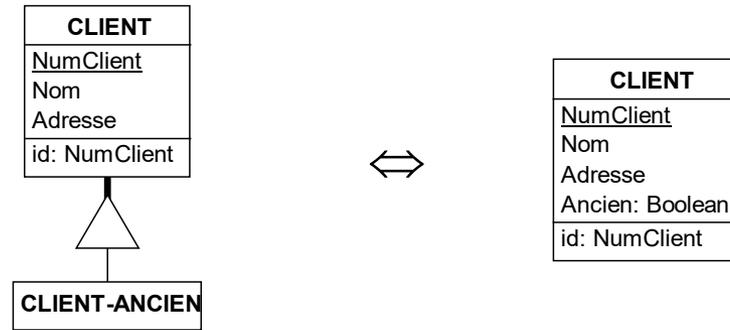
Sous-type faiblement spécifique



Ce sous-type est-il nécessaire ?

3.5 Processus de normalisation d'un schéma - *Minimalité*

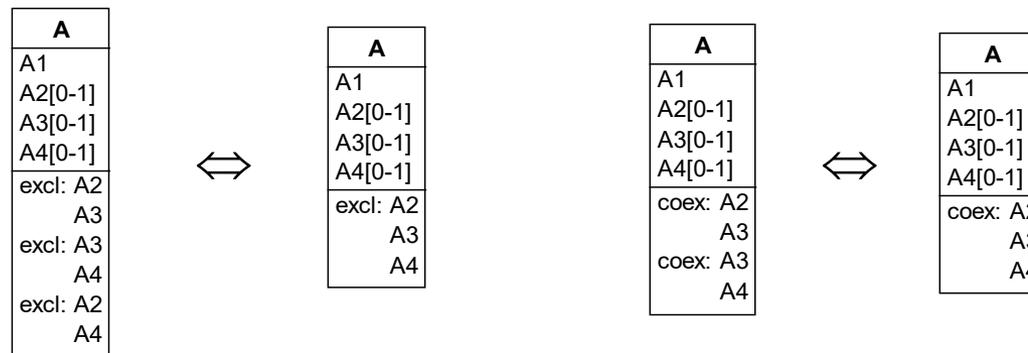
Sous-type unique sans propriété



Ce sous-type est-il nécessaire ?

3.5 Processus de normalisation d'un schéma - *Minimalité*

Contraintes d'existence ternaires décomposées



Décomposition correcte mais inutilement complexe

3.5 Processus de normalisation d'un schéma - *Expressivité*

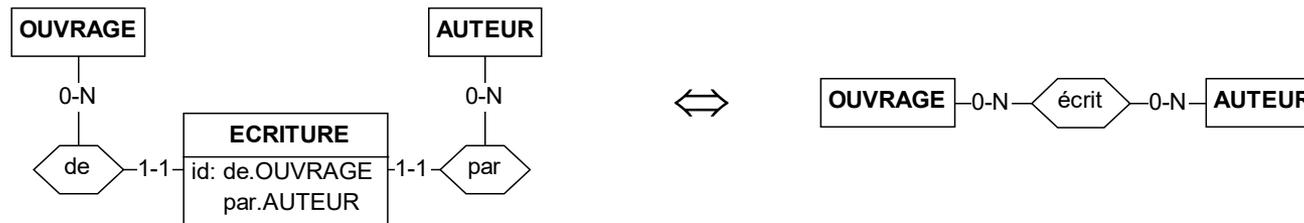
2. Expressivité

Un type de faits doit être représenté par une construction qui évoque naturellement et clairement sa nature.

Par exemple on utilisera une relation *is-a* pour indiquer qu'un type d'entités est un cas particulier d'un autre.

3.5 Processus de normalisation d'un schéma - *Expressivité*

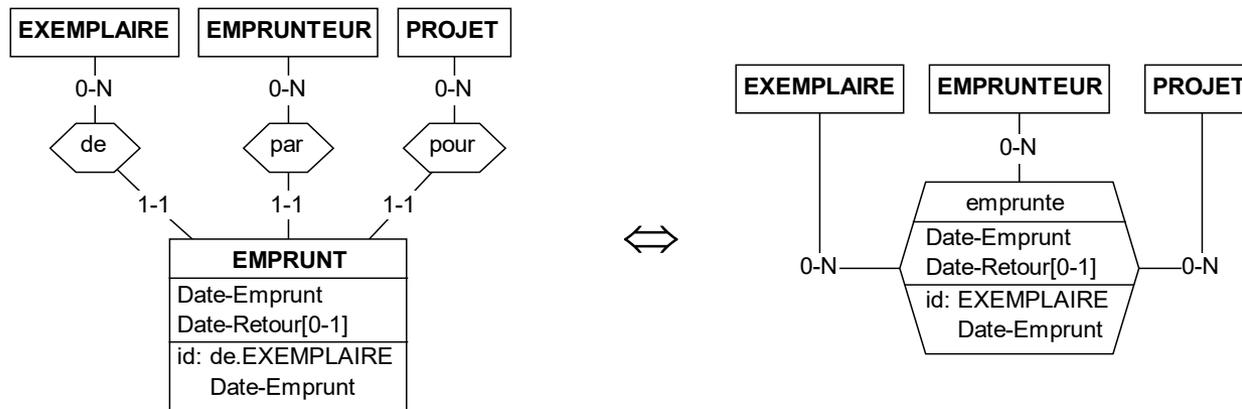
Type d'entités associations



Type d'entités exprimant un type d'associations

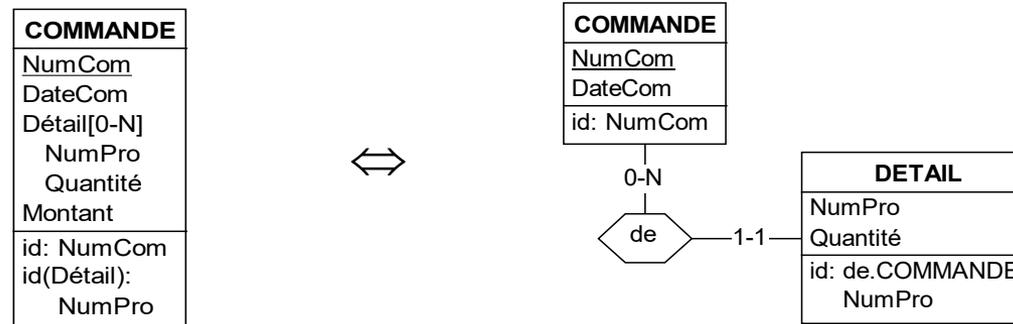
3.5 Processus de normalisation d'un schéma - *Expressivité*

Type d'entités associations



3.5 Processus de normalisation d'un schéma - *Expressivité*

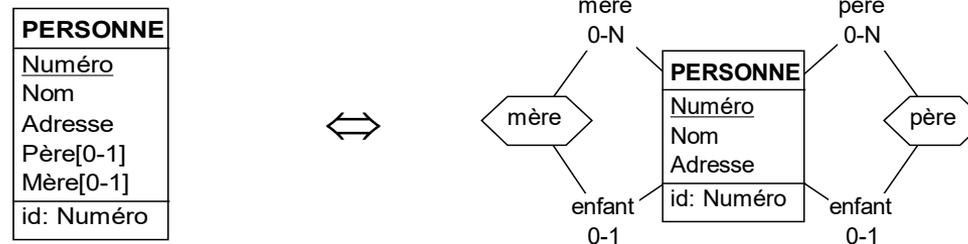
Attribut complexe



Attribut représentant un type d'entités caché

3.5 Processus de normalisation d'un schéma - *Expressivité*

Attribut de référence

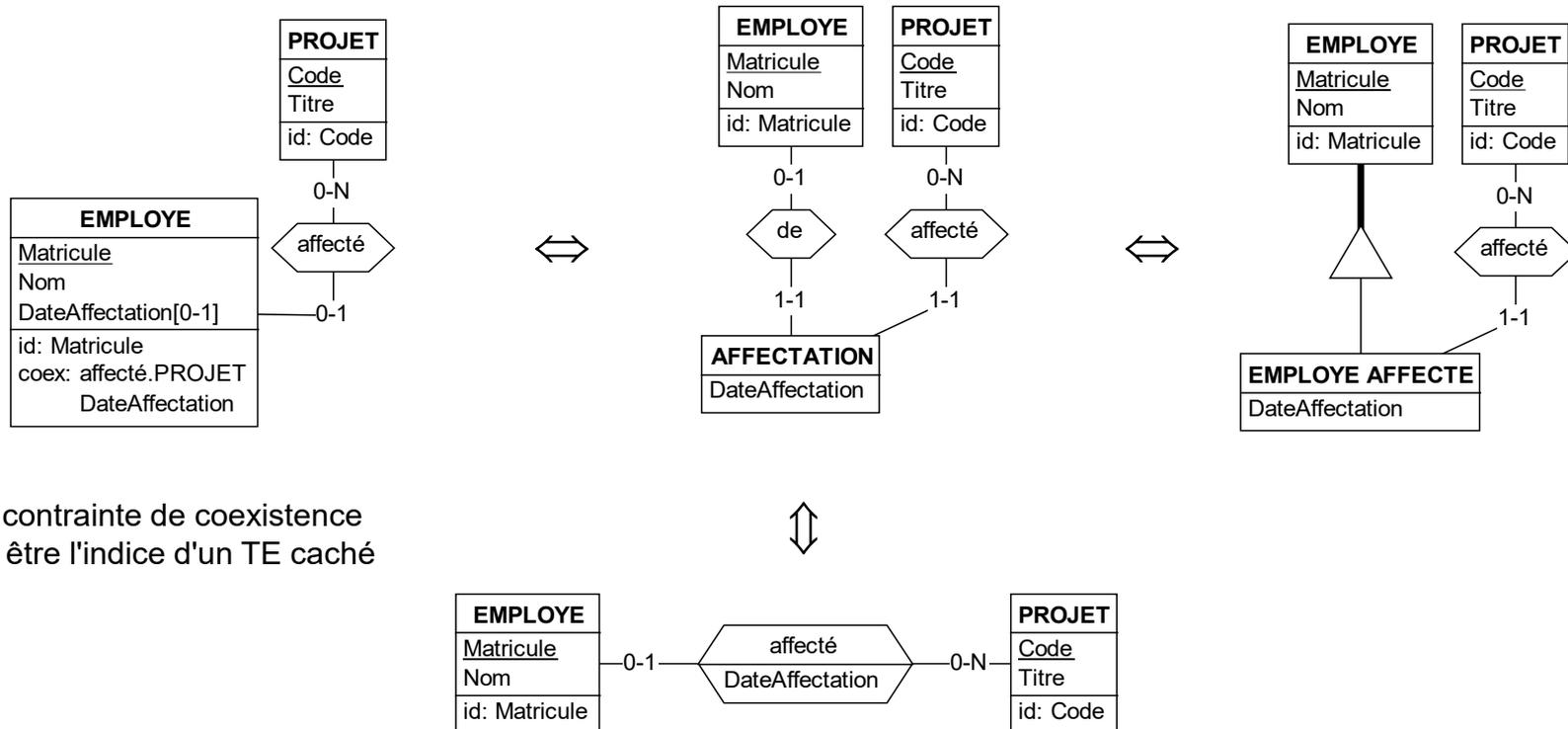


Attribut utilisé pour référencer d'autres entités

= clé étrangère implicite

3.5 Processus de normalisation d'un schéma - Expressivité

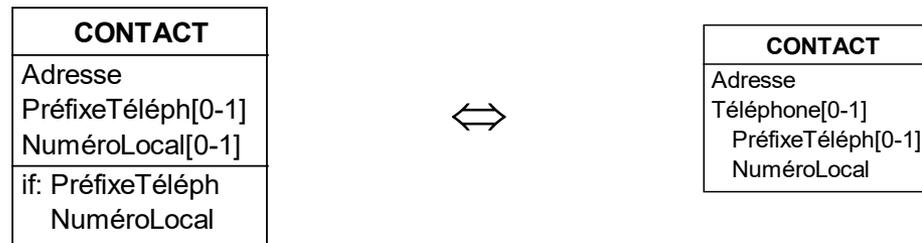
Contrainte de coexistence



Une contrainte de coexistence peut être l'indice d'un TE caché

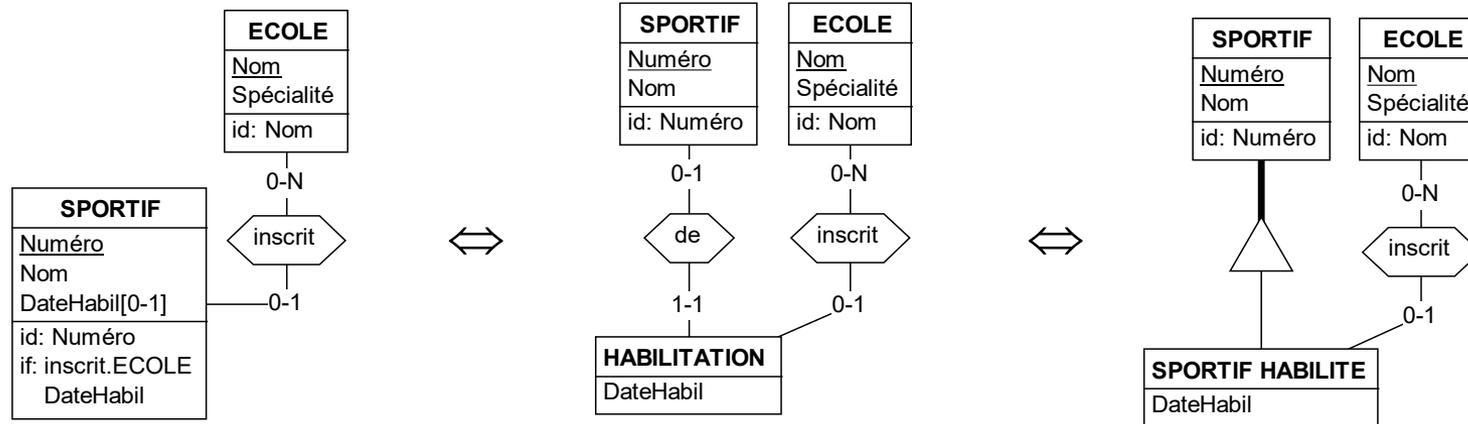
3.5 Processus de normalisation d'un schéma - *Expressivité*

Contrainte d'implication



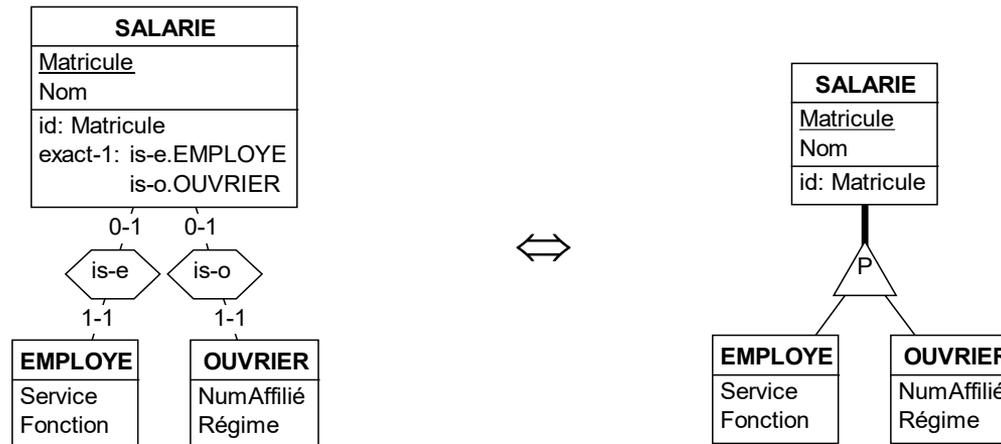
3.5 Processus de normalisation d'un schéma - Expressivité

Contrainte d'implication



3.5 Processus de normalisation d'un schéma - *Expressivité*

Relation *is-a* implicite



- 1. Méthodologie des BD
- 2. Le modèle Entité-association
- 3. Analyse conceptuelle**
- 4. Conception logique relationnelle

- 5. Conception physique
- 6. Production du code
- 7. Rétro-ingénierie

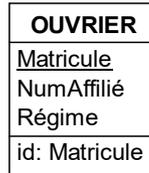
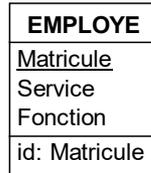
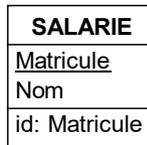
- 3.1 Introduction
- 3.2 Analyse des énoncés textuels
- 3.3 Analyse des documents structurés
- 3.4 Correction d'un schéma

- 3.5 Normalisation d'un schéma**
- 3.6 Evaluation d'un schéma
- 3.7 Intégration de schémas
- 3.8 Analyse à l'aide d'un AGL

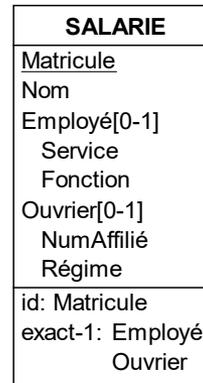
3.5 Processus de normalisation d'un schéma - Expressivité

Relation *is-a* implicite

contraintes manquantes

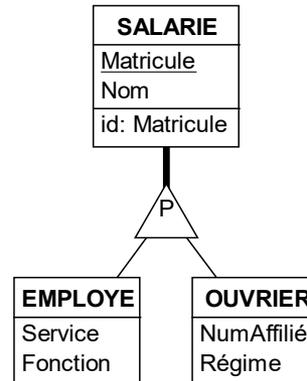
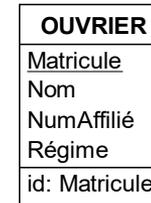
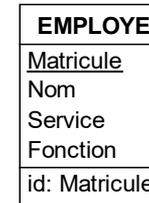


ou



ou

contraintes manquantes



3.5 Processus de normalisation d'un schéma - *Pas de sur-spécification*

3. Absence de sur-spécification

Le schéma représente les types de faits

- raisonnablement utiles
- sans préjuger de la manière dont ils seront implémentés.

3.5 Processus de normalisation d'un schéma - *Pas de sur-spécification*

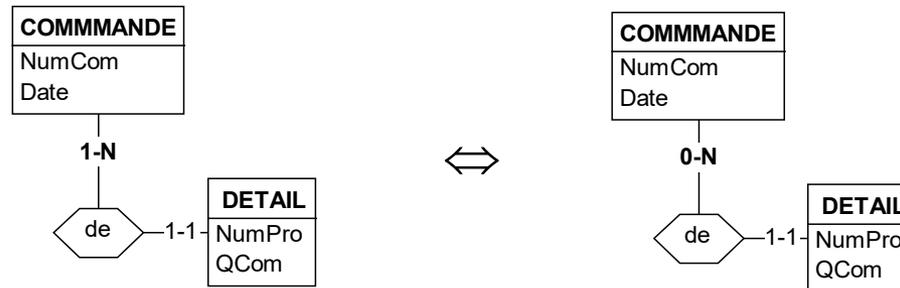
Intégrité et plausibilité

CLIENT
<u>NumCli</u> Nom Prénom Date Naissance Code Postal Rue Numéro Localité Compte
id: NumCli id': Nom Prénom Date Naissance Code Postal Rue Numéro Localité

l'identifiant secondaire est-il vraiment un identifiant ?

3.5 Processus de normalisation d'un schéma - *Pas de sur-spécification*

Relaxation de contraintes trop fortes



Bien sûr, toute commande possède au moins un détail, mais ...

3.5 Processus de normalisation d'un schéma - *Prévisibilité, régularité*

4. Prévisibilité, régularité

Le lecteur ne doit pas être *surpris* par l'usage d'une construction pour exprimer un certain type de faits.

Deux sortes de surprises :

- les constructions *anormales* (cas limites d'utilisation d'une construction, style inhabituel)
- les constructions *irrégulières* (changement injustifié de style, style incohérent).

3.5 Processus de normalisation d'un schéma - *Prévisibilité, régularité*

4.1 Prévisibilité, régularité - Pas de constructions anormales

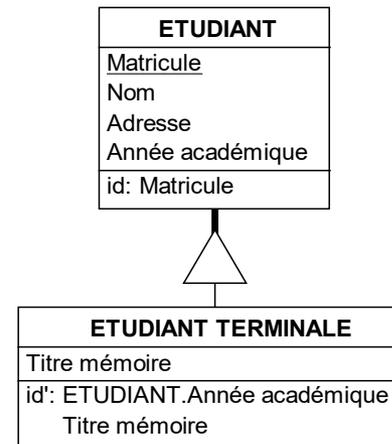
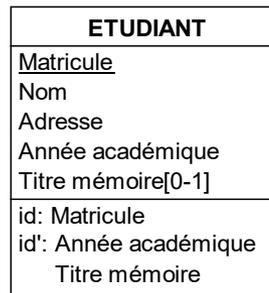
Constructions dégénérées



3.5 Processus de normalisation d'un schéma - *Prévisibilité, régularité*

4.1 Prévisibilité, régularité - Pas de constructions anormales

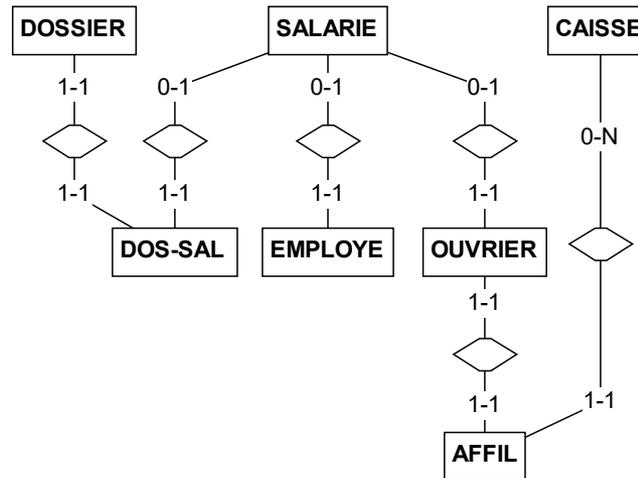
Identifiant semi-obligatoire



3.5 Processus de normalisation d'un schéma - *Prévisibilité, régularité*

4.1 Prévisibilité, régularité - Pas de constructions anormales

Constructions étrangères

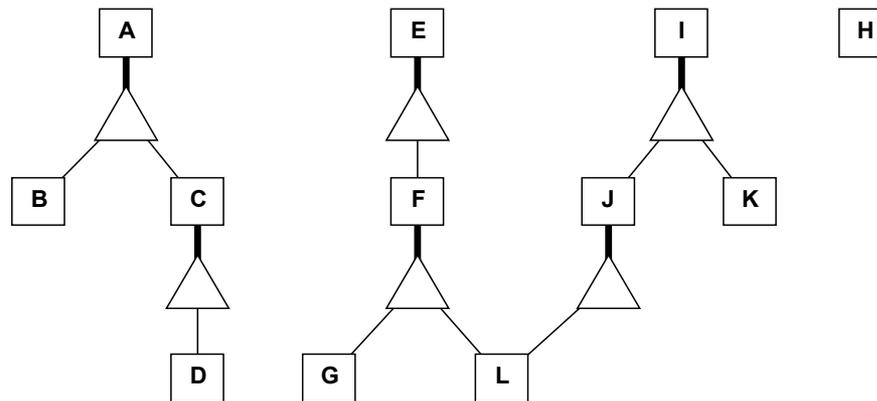


Esprit IMS !

3.5 Processus de normalisation d'un schéma - *Prévisibilité, régularité*

4.1 Prévisibilité, régularité - Pas de constructions anormales

Constructions étrangères



Esprit programmation OO !

3.5 Processus de normalisation d'un schéma - *Prévisibilité, régularité*

4.1 Prévisibilité, régularité - Pas de constructions irrégulières

Tous les types de faits de même nature doivent être représentés par les mêmes constructions.

3.5 Processus de normalisation d'un schéma - Absence de redondance

5. Absence de redondances

Un schéma doit exprimer un type de faits une et une seule fois.
Il en est de même des instances futures ou existantes.

On retrouve une qualité proche de celle qui est recommandée pour les schémas relationnels.

Trois variantes de redondance :

1. Redondance de **contraintes**
2. Redondance **structurelle d'instances**
3. Redondance **interne**

3.5 Processus de normalisation d'un schéma - *Absence de redondance*

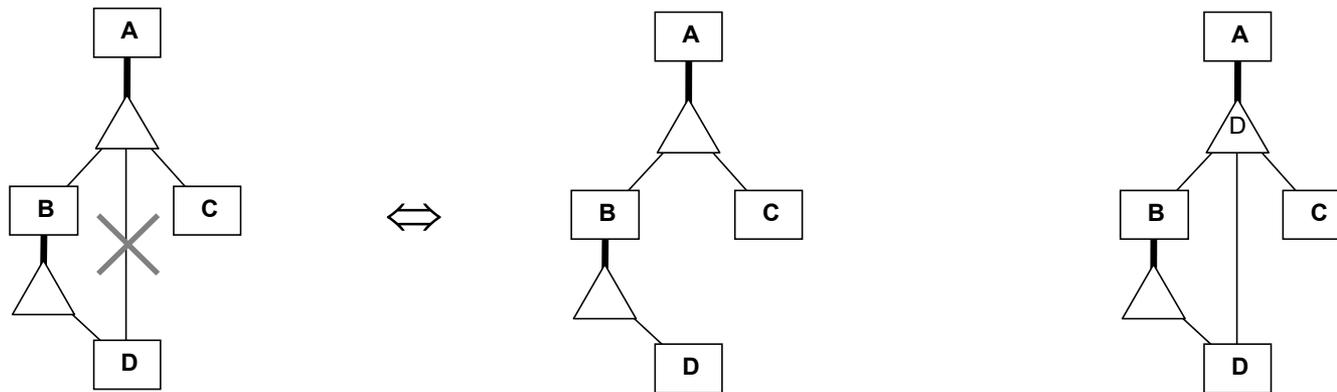
Redondance de contraintes

Une contrainte dérive d'autres constructions du schéma

Rectification : supprimer la contrainte dérivable

3.5 Processus de normalisation d'un schéma - Absence de redondance

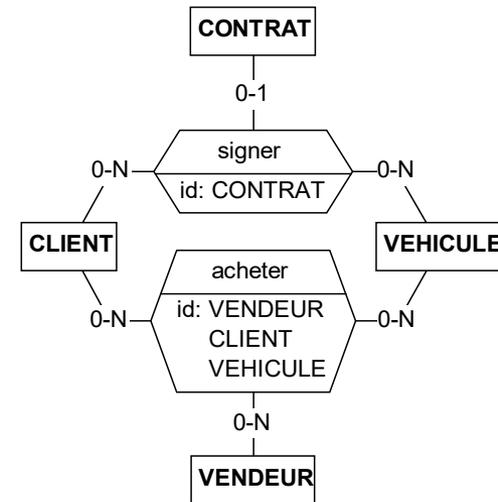
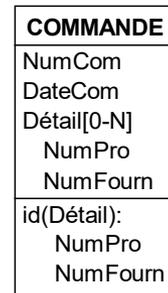
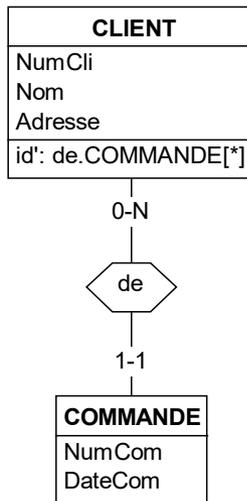
Redondance de contraintes



redundant ET non satisfiable !

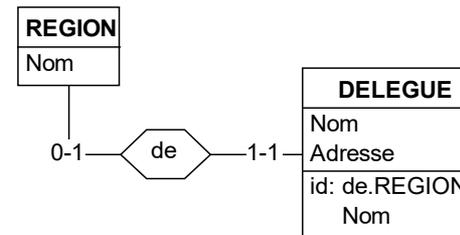
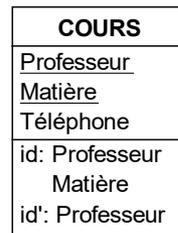
3.5 Processus de normalisation d'un schéma - Absence de redondance

Redondance de contraintes



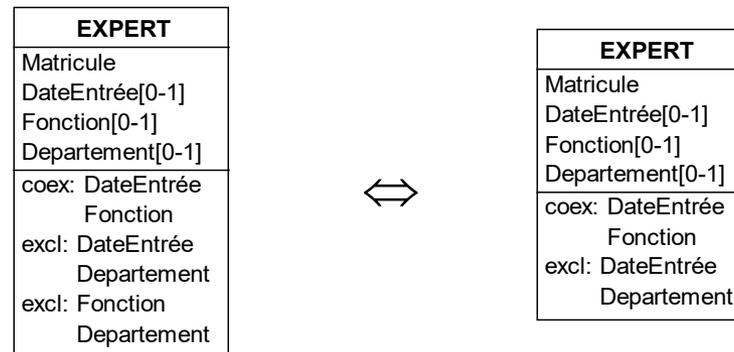
3.5 Processus de normalisation d'un schéma - *Absence de redondance*

Redondance de contraintes



3.5 Processus de normalisation d'un schéma - *Absence de redondance*

Redondance de contraintes



3.5 Processus de normalisation d'un schéma - *Absence de redondance*

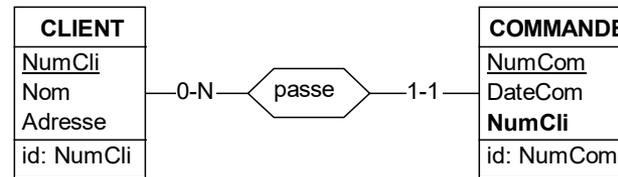
Redondance structurelle d'instances

Une construction possède des instances calculables
à partir d'instances d'autres constructions

Rectification : supprimer la construction redondante

3.5 Processus de normalisation d'un schéma - Absence de redondance

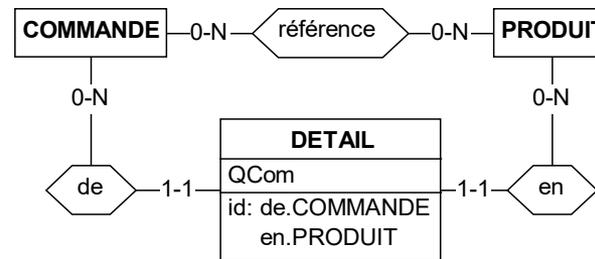
Redondance structurelle d'instances



$\forall c \in \text{COMMANDE}, c.\text{NumCli} = c.\text{passe}.\text{CLIENT}.\text{NumCli}$

3.5 Processus de normalisation d'un schéma - Absence de redondance

Redondance structurelle d'instances



COMMANDE.référence.PRODUIT = COMMANDE.de.DETAIL.en.PRODUIT

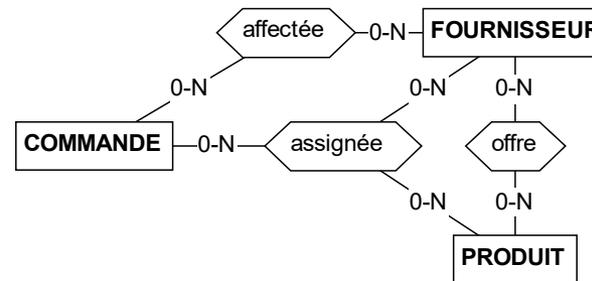
reference [COMMANDE, PRODUIT] = de o en [COMMANDE, PRODUIT]

reference = de o en

comment corriger ?

3.5 Processus de normalisation d'un schéma - Absence de redondance

Redondance structurelle d'instances



$\text{affectée} = \text{assignée}[\text{COMMANDE}, \text{FOURNISSEUR}]$

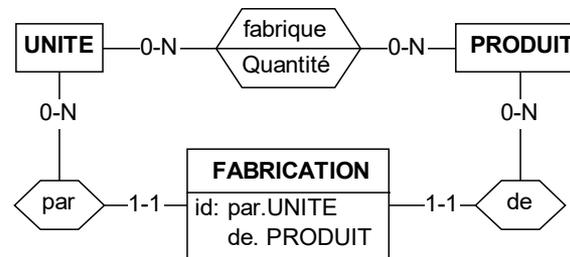
redondance

$\text{assignée}[\text{FOURNISSEUR}, \text{PRODUIT}] \subseteq \text{offre}$

contrainte d'intégrité

3.5 Processus de normalisation d'un schéma - Absence de redondance

Redondance structurelle d'instances



UNITE.fabriquer.PRODUIT = UNITE.par.FABRICATION.de.PRODUIT

comment corriger ?

3.5 Processus de normalisation d'un schéma - *Absence de redondance*

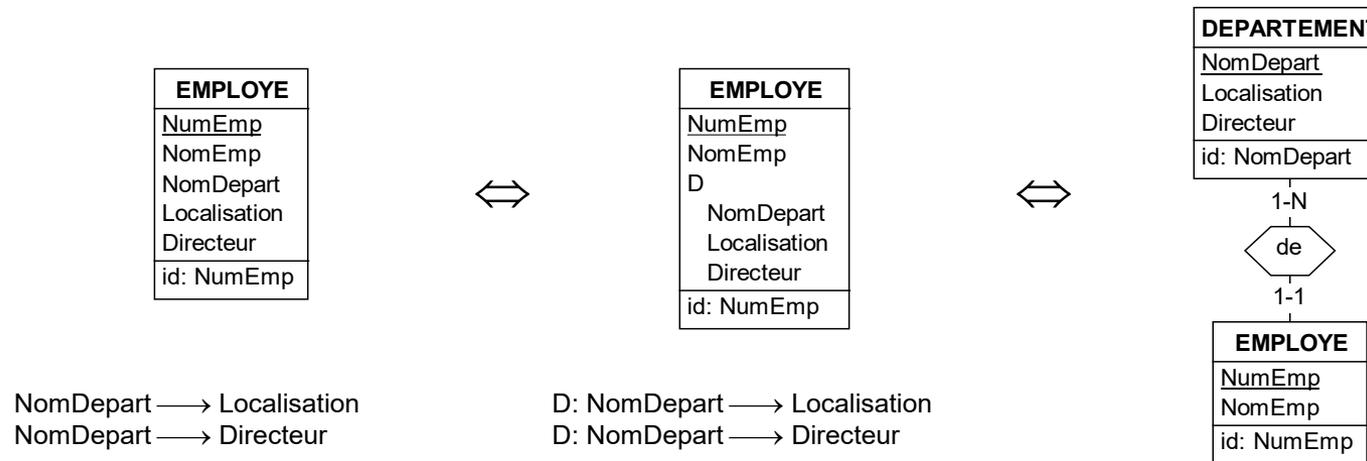
Redondance interne

Une construction n'est pas sous forme normale
au sens de la théorie relationnelle

Rectification : décomposer

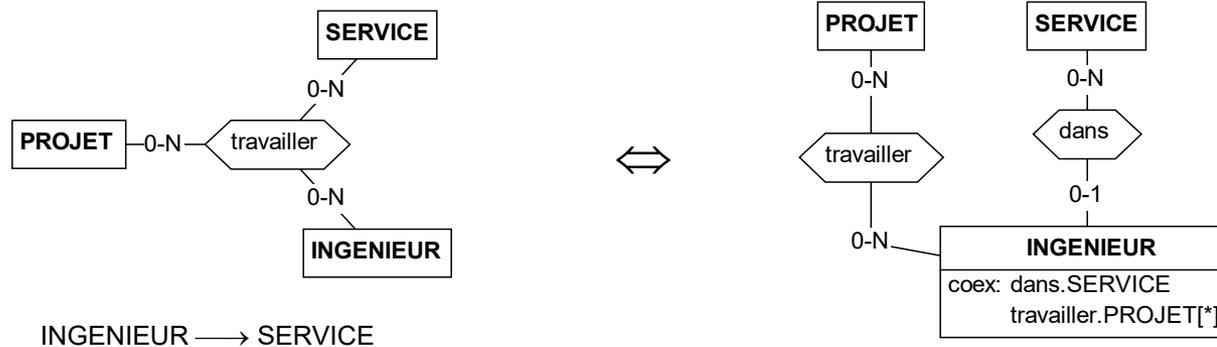
3.5 Processus de normalisation d'un schéma - Absence de redondance

Redondance interne



3.5 Processus de normalisation d'un schéma - Absence de redondance

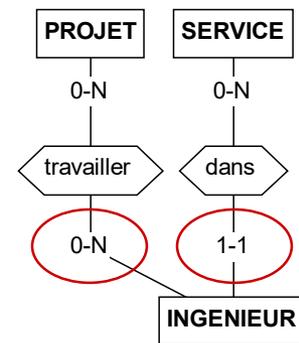
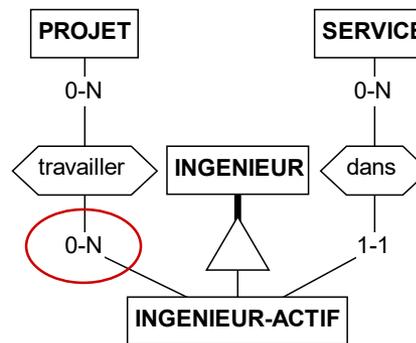
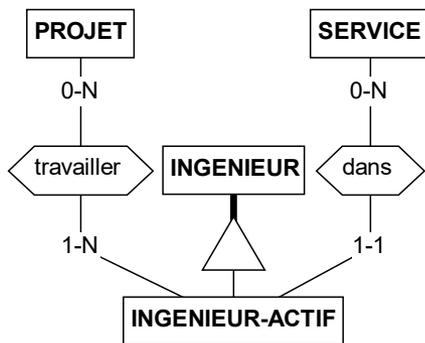
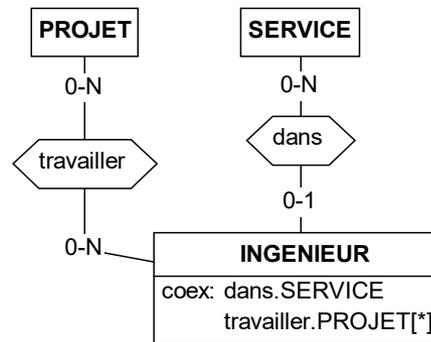
Redondance interne



3.5 Processus de normalisation d'un schéma - Absence de redondance

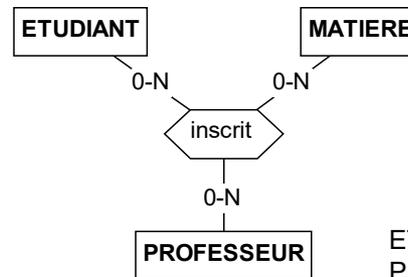
suite ...

Redondance interne

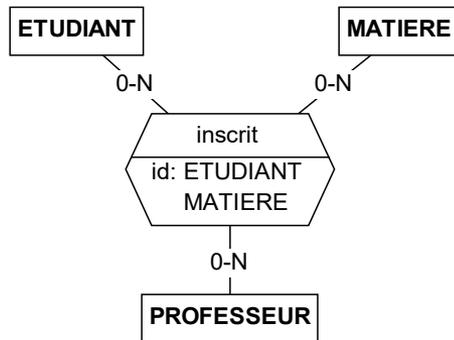


3.5 Processus de normalisation d'un schéma - Absence de redondance

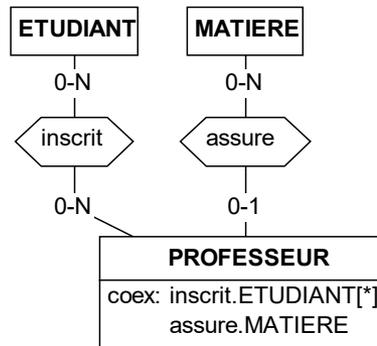
Redondance interne



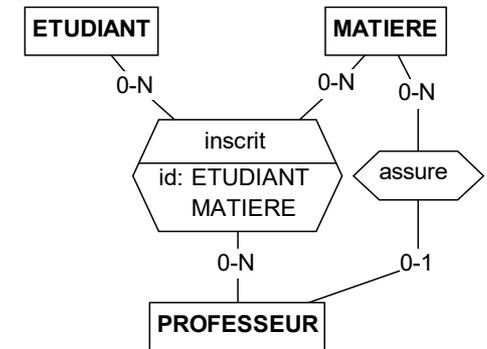
ETUDIANT, MATIERE → PROFESSEUR
 PROFESSEUR → MATIERE



PROFESSEUR → MATIERE



inscrit * assure: ETUDIANT, MATIERE → PROFESSEUR



inscrit[PROFESSEUR, MATIERE] ⊆ assure

3.5 Processus de normalisation d'un schéma - *Qualité graphique*

6. Qualité graphique

La manière dont les constructions d'un schéma sont disposées dans l'espace graphique peut améliorer considérablement sa lisibilité et donc sa maintenabilité et son évolutivité.

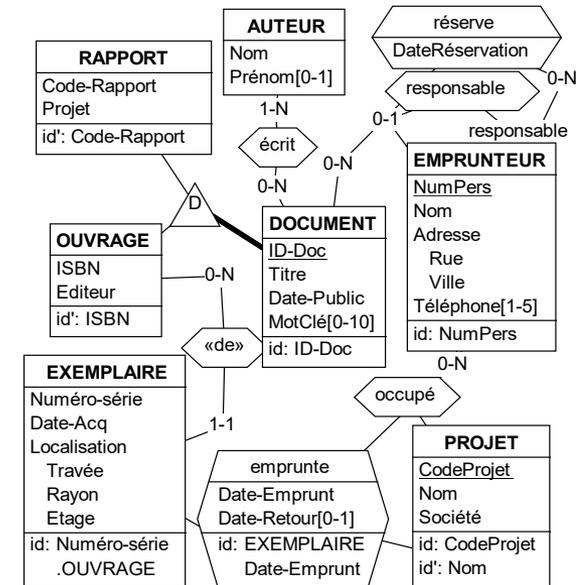
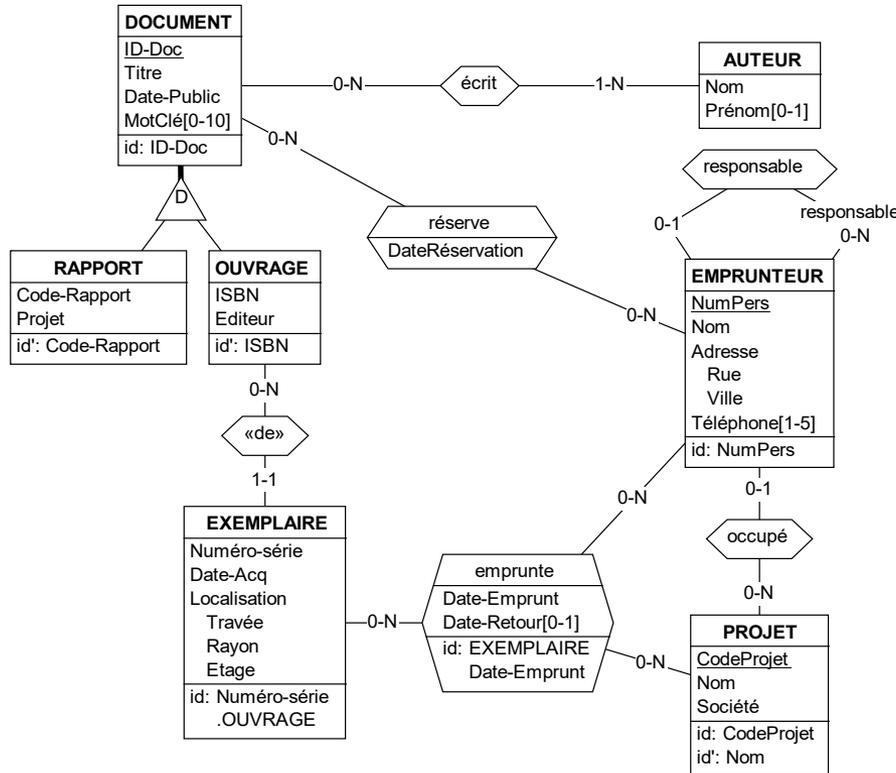
1. Méthodologie des BD
2. Le modèle Entité-association
- 3. Analyse conceptuelle**
4. Conception logique relationnelle

5. Conception physique
6. Production du code
7. Rétro-ingénierie

- 3.1 Introduction
- 3.2 Analyse des énoncés textuels
- 3.3 Analyse des documents structurés
- 3.4 Correction d'un schéma

- 3.5 Normalisation d'un schéma**
- 3.6 Evaluation d'un schéma
- 3.7 Intégration de schémas
- 3.8 Analyse à l'aide d'un AGL

3.5 Processus de normalisation d'un schéma - Qualité graphique



Est-il possible de vérifier si un emprunteur qui réserve un document n'en a pas emprunté auparavant ?

3.5 Processus de normalisation d'un schéma - *Qualité graphique*

Trois classes de règles

1. *Règles de disposition de graphes* : indépendantes de la nature du graphe
2. *Règles de disposition Entité-association* : indépendantes du domaine d'application
3. *Règles de disposition liées au domaine d'application*

3.5 Processus de normalisation d'un schéma - *Qualité graphique*

Règles de disposition de graphes

1. Minimiser les croisements d'arcs
2. En cas de croisement, éviter les angles trop faibles
3. Laisser un espace suffisant entre deux noeuds
4. Ecarter les uns des autres les noeuds directement connectés ...
5. . . . mais pas trop !
6. Ecarter un noeud des arcs dans lesquels il n'intervient pas
etc.

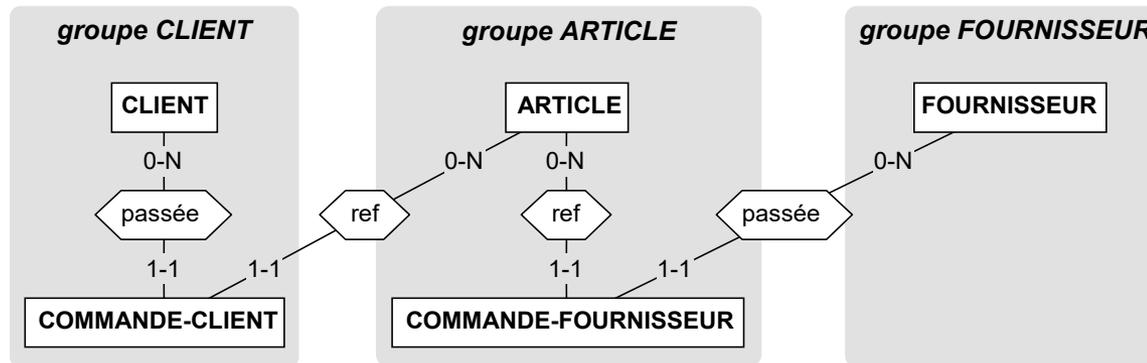
3.5 Processus de normalisation d'un schéma - *Qualité graphique*

Règles de disposition Entité-association

1. Placement relatif des TE connectés par type d'associations 1:N
 2. Placement relatif des TE connectés par type d'associations 1:1 ou N:N
 3. Placement relatif des TE connectés par type d'associations de composition, de matérialisation
 4. Placement relatif des TE connectés par des relations *is-a*
 5. Placement relatif des TE connectés par type d'associations n-aire
 6. Position des attributs de l'identifiant primaire
- etc.*

3.5 Processus de normalisation d'un schéma - *Qualité graphique*

Règles de disposition liées au domaine d'application



Irrégularité justifiée par les règles spécifiques du métier

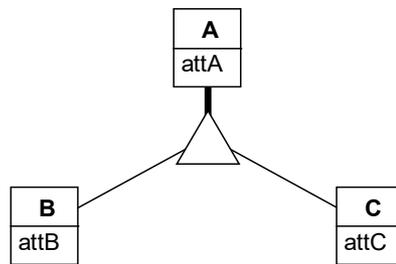
3.5 Processus de normalisation d'un schéma - *Respect des standards*

7. Respect des standards en vigueur

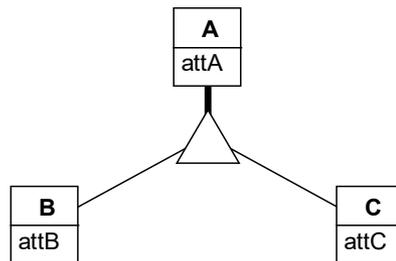
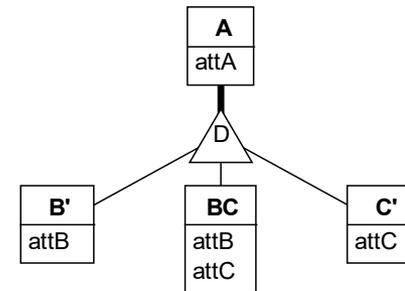
Le schéma doit respecter certaines règles imposées par l'organisme pour lequel (ou dans lequel) il est développé, quand bien même le concepteur les estimerait inutiles, voire absurdes.

3.5 Processus de normalisation d'un schéma - *Respect des standards*

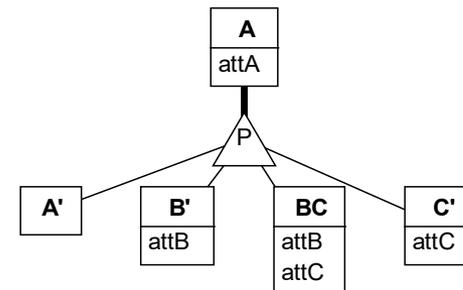
Restrictions sur les relations *is-a*



disjonction obligatoire



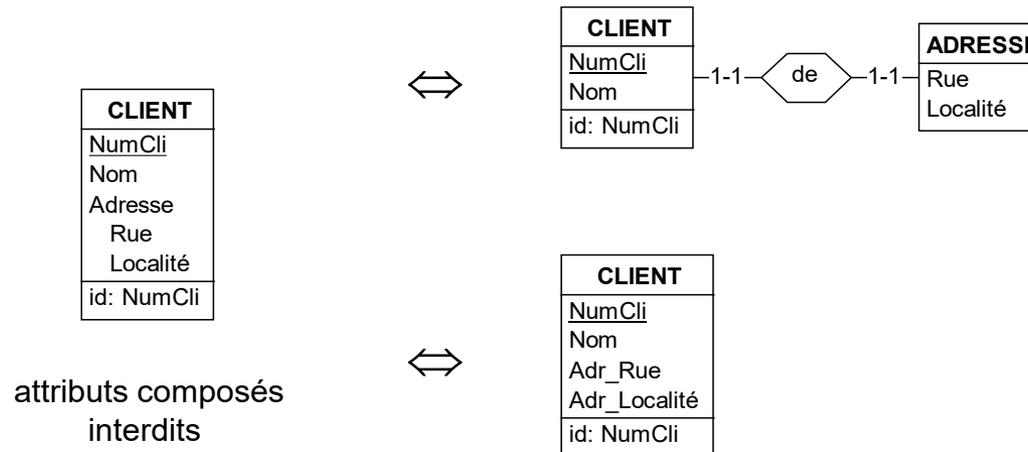
partition obligatoire



Attention : question beaucoup plus complexe qu'il n'y paraît !

3.5 Processus de normalisation d'un schéma - *Respect des standards*

Restrictions sur la structure des attributs



3.5 Processus de normalisation d'un schéma - *Respect des standards*

Restrictions sur les identifiants

RAPPORT
<u>Titre</u>
Auteurs
Statut
Date
id: Titre



RAPPORT
<u>ID_Rapport</u>
Titre
Auteurs
Statut
Date
id: ID_Rapport
id': Titre

id technique
obligatoire

3.5 Processus de normalisation d'un schéma

Restrictions sur les noms

Syntaxe, composition, formes verbales, etc.

Restrictions sur le modèle

Modèle binaire, diagrammes de classes UML, modèle d'un AGL, etc.

3.6 Evaluation d'un schéma

3.6 Evaluation d'un schéma

En principe, les utilisateurs doivent approuver les spécifications fonctionnelles des différents composants du système d'information, . . .

. . . mais il apparaît qu'ils sont incapables de juger de la validité d'un schéma :

- **complexité du modèle EA**
- **taille du schéma**
- **complexité du schéma**

Comment rendre la sémantique d'un schéma accessible à l'utilisateur ?

- 1. paraphrase**
- 2. prototypage**
- 3. inférence structurelle**
- 4. les heuristiques d'erreurs**

3.6 Evaluation d'un schéma

1. Paraphrase

Le schéma conceptuel est traduit sous la forme de propositions élémentaires

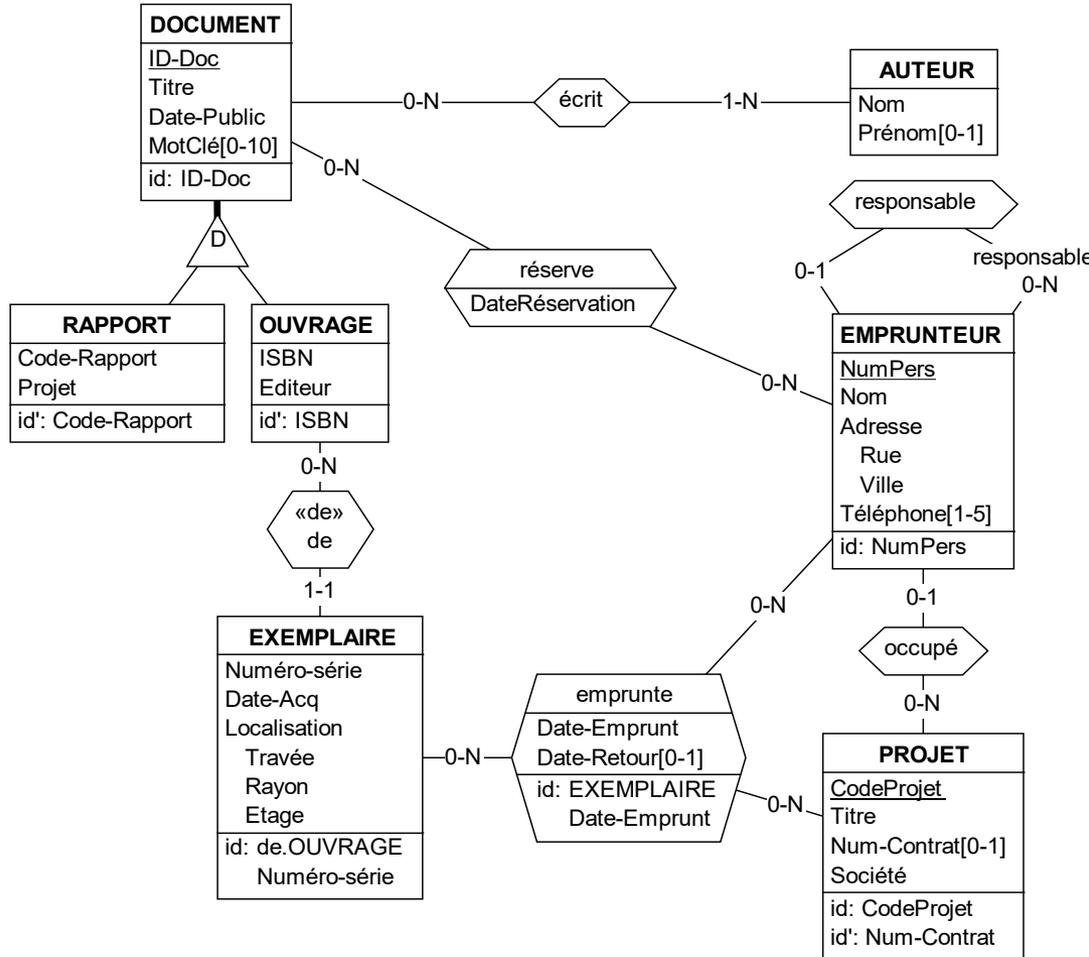
La véracité de chaque proposition est évaluée par les utilisateurs

Les propositions rejetées sont rediscutées

Automatisation partielle possible

3.6 Evaluation d'un schéma

1. Paraphrase



- Le domaine comprend des *documents*, des *rapports*, des *ouvrages*, des *exemplaires*, des *auteurs*, des *emprunteurs* et des *projets*.
- Tout *document* est identifié par son *ID-Doc*. Il est caractérisé par un *Titre*, une *date de publication* et possède de 0 à 10 *mots-clés*.
- Un *document* peut être un *rapport* ou un *ouvrage*. Un *rapport* ne peut pas être un *ouvrage*.
- Un *rapport* est identifié par son *Code-Rapport*. Il est caractérisé par un *Projet*.
- Un *document* peut être écrit par un nombre quelconque d'*auteurs*. Tout *auteur* écrit au moins un *document*.
- Un *document* peut être *réservé* par un nombre quelconque d'*emprunteurs*. Chacun de ces faits est caractérisé par une *date de réservation*.
- etc.*

3.6 Evaluation d'un schéma

2. Prototypage

Prototype :

- **version préliminaire simplifiée d'un système en projet**
- **destiné à l'évaluation de certaines décisions de conception**
- **offre certaines fonctions (besoins fonctionnels)**
- **ne prend généralement pas en compte les besoins non fonctionnels (performances, ergonomie, robustesse, etc.)**

Cette technique n'est viable que si le prototype peut être généré automatiquement.

Alternative : utilisation d'outils RAD (Rapid Application Development)

3.6 Evaluation d'un schéma

2. Prototypage

Prototype de base de données :

- **schéma logique et code SQL produits automatiquement,**
- **interface de saisie et gestion de données produite automatiquement,**
- **interface de consultation de données produite automatiquement,**
- **traitements sous forme de requêtes SQL (développement manuel)**

Exploitation : les utilisateurs *jouent* avec le prototype pour vérifier sa conformité aux besoins exprimés. Les spécifications sont modifiées en fonction des réactions. Versions successives du prototype.

La version finale peut faire partie du cahier des charges pour l'implémentation.

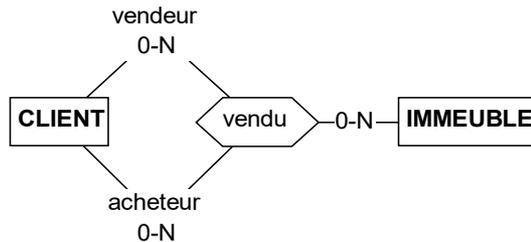
3.6 Evaluation d'un schéma

3. Inférence structurelle

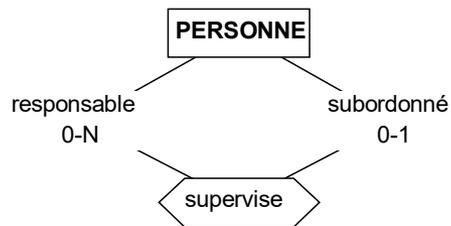
En complément de la paraphrase. On présente à l'utilisateur des propriétés dérivant formellement du schéma conceptuel. L'utilisateur approuve ou conteste ces propriétés.

3.6 Evaluation d'un schéma

3. Inférence structurelle



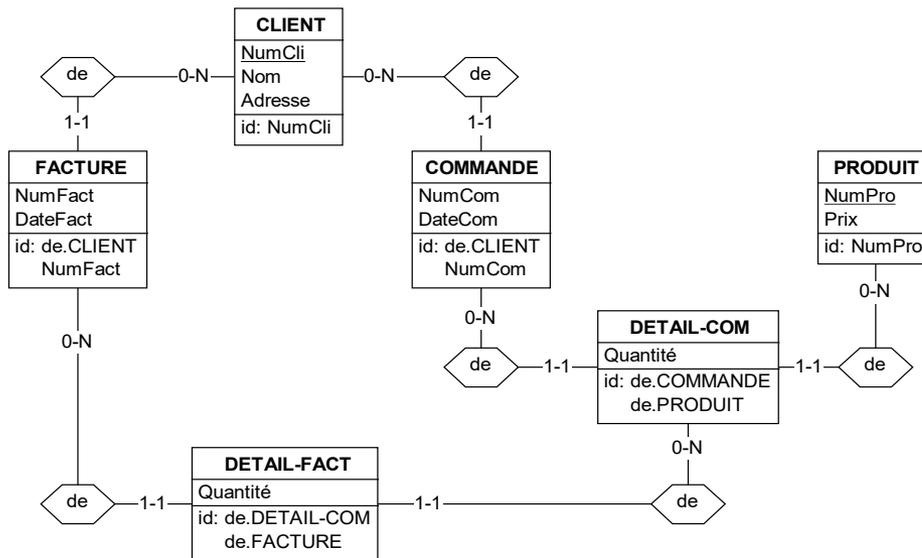
Il est possible qu'un immeuble soit acheté par son vendeur.
D'accord ?



Il est possible qu'une personne soit son propre responsable.
D'accord ?

3.6 Evaluation d'un schéma

3. Inférence structurelle



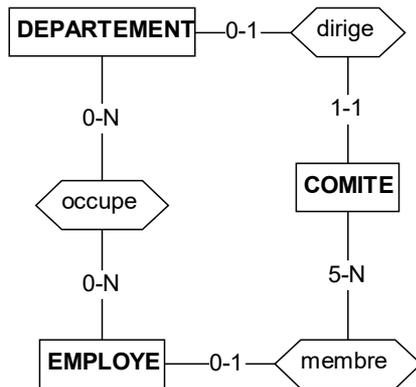
**Pour identifier un détail de facture, il faut préciser deux valeurs de NumCli.
 D'accord ?**

**Pour un détail de facture, le client de la facture et le client de la commande du détail de commande peuvent être différents.
 D'accord ?**

**Une facture peut référencer plusieurs fois le même produit, ce qui n'est pas le cas des commandes.
 D'accord ?**

3.6 Evaluation d'un schéma

3. Inférence structurelle



**Un département dirigé par un comité occupe au moins 5 employés.
 D'accord ?**

DEPARTEMENT.dirige.COMITE.membre.EMPLOYE
 \subseteq
 DEPARTEMENT.occupe.EMPLOYE

3.6 Evaluation d'un schéma

4. Les heuristiques d'erreurs

Observations :

- **Les erreurs apparaissent majoritairement dans certaines constructions dites à *risque*.**
- **Certaines constructions apparaissent rarement dans un schéma.**

Technique :

1. **On identifie les constructions à risque du schéma.**
2. **On valide ces constructions, éventuellement avec l'aide des utilisateurs.**
3. **On corrige si nécessaire.**

→ automatisable

3.6 Evaluation d'un schéma

4. Les heuristiques d'erreurs

Exemples de **constructions à risque** :

- **types d'associations non fonctionnels**
- **types d'associations n-aires**
- **types d'associations avec attributs**
- **attributs complexes**
- **surtypes multiples**
- **sous-types non disjoints**
- **identifiants**

3.6 Evaluation d'un schéma

4. Les heuristiques d'erreurs

Exemples de **constructions rares** :

- types d'associations de degré > 3
- types d'entités sans attribut
- types d'entités sans identifiant
- attributs complexes de plus de 2 niveaux
- identifiants de plus de 2 composants

3.6 Evaluation d'un schéma

5. Cohérence avec les autres composants

1. Les autres diagrammes mentionnent des types de données. Ceux-ci sont-ils repris dans le schéma conceptuel ?
2. Le schéma conceptuel mentionne des types de données. Ceux-ci sont-ils utilisés dans les autres diagrammes ?

automatisable

3.7 Intégration de schémas

3.7 Intégration de schémas

Les schémas des sous-systèmes sont *fusionnés* pour former le schéma global

Problème complexe car divergences de modélisation parmi les schémas sources :

- **différentes manières de modéliser les mêmes types de faits**
- **différentes manières de percevoir les mêmes types de faits**
- **un type de faits peut être absent d'un sous-système**

Problème plus général apparaissant dans d'autres processus, par exemple la *rétro-ingénierie* et l'*intégration de bases de données*.

3.7 Intégration de schémas

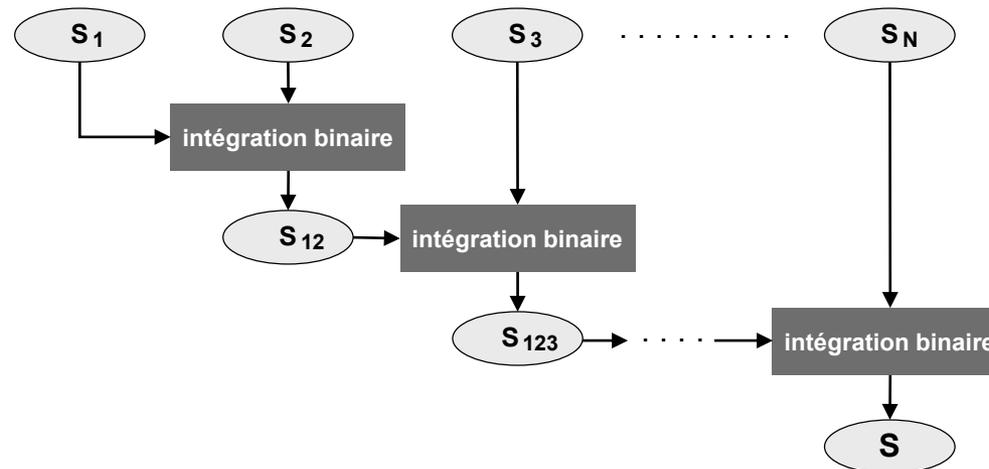
Principes

Soit $SS = \{S1, S2, \dots, SN\}$ un ensemble de N schémas

- Intégrer SS = produire un schéma global unique S qui représente, sans redondance, les concepts des schémas de SS et eux seulement.
- Tout type de faits représenté dans un schéma S_i est aussi représenté dans S .
- Tout type de faits représenté dans S l'est aussi dans au moins un des sous-schémas de SS .

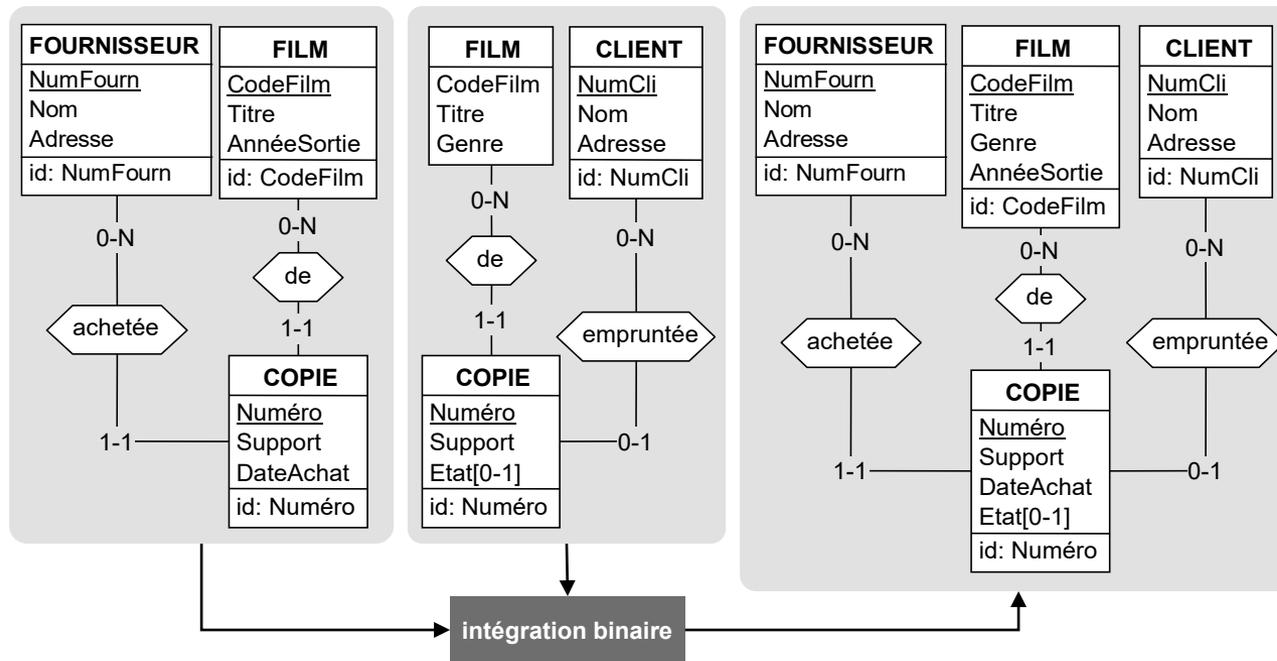
3.7 Intégration de schémas

Scénario d'intégration binaire (en échelle)



3.7 Intégration de schémas

Situation idéale : la fusion simple



Les concepts du domaine d'application (types de faits) identiques sont représentés par des constructions de même nature et de même nom.

3.7 Intégration de schémas

En toute généralité . . .

Procédure générale d'intégration des schémas S1 et S2 :

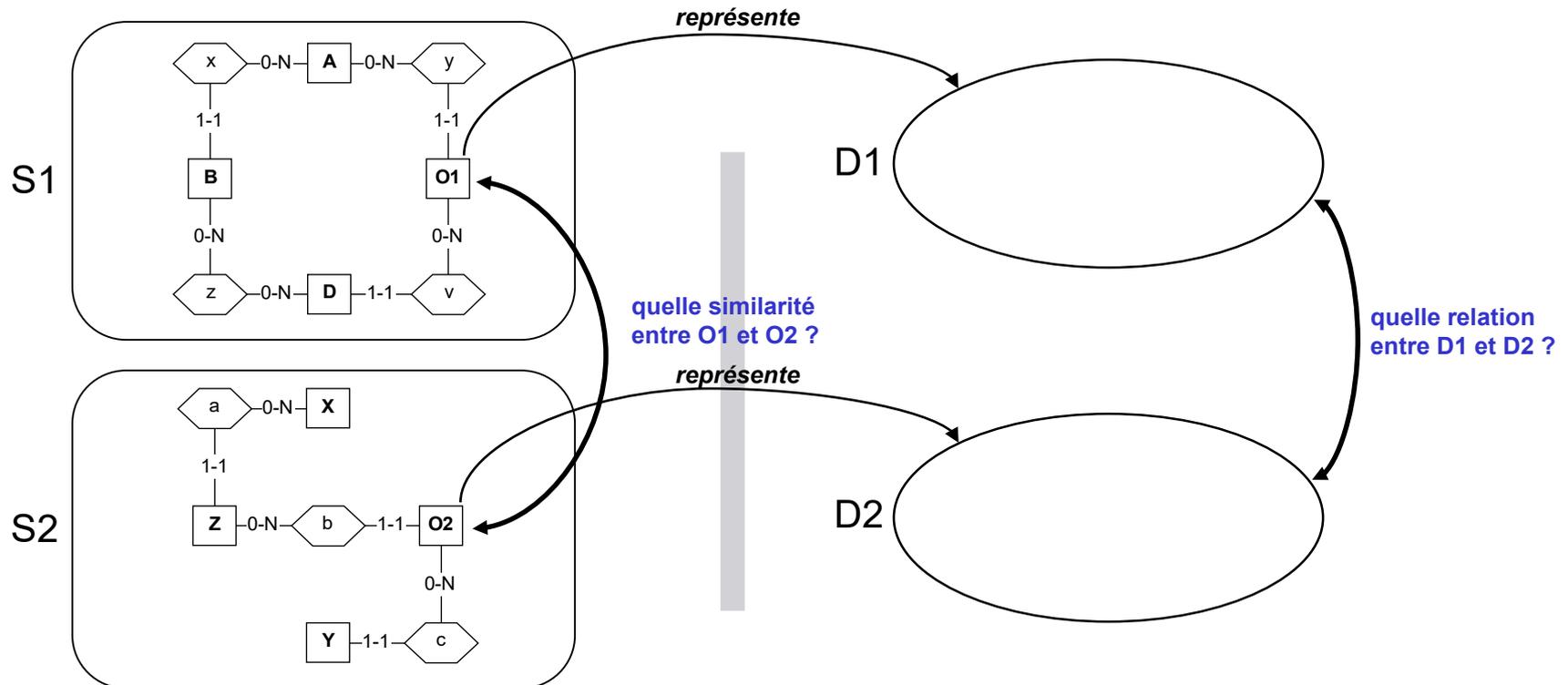
1. ***Recherche des correspondances*** : identifier les relations sémantiques entre les objets du domaine dénotés par les constructions des deux schémas.
2. ***Unification des schémas*** : transformation des constructions en relation pour les rendre conformes.
3. ***Fusion des schémas*** : production du schéma intégré.

3.7 Intégration de schémas - Correspondances

Etape 1 : Recherche des correspondances

Deux constructions $O1 \in S1$ et $O2 \in S2$
 présentent-elles des similarités sémantiques ?

Les objets du domaine d'application D1 et
 D2 qu'ils représentent sont-ils corrélés ?



3.7 Intégration de schémas - Correspondances

Les correspondances

On retient six types de correspondances

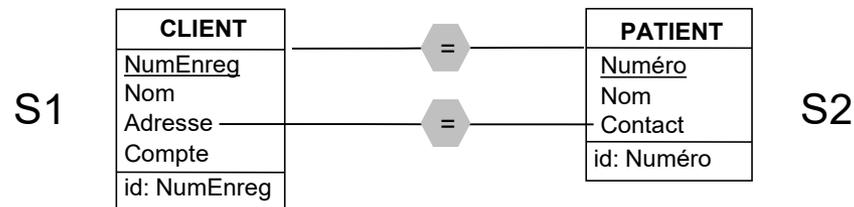
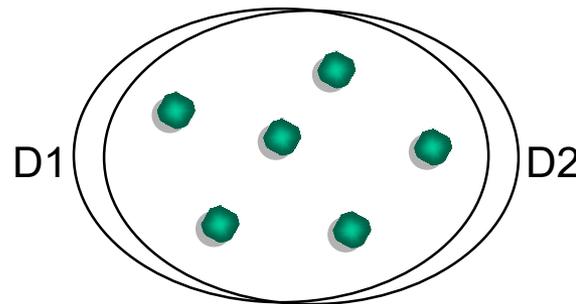
1. égalité
 2. complémentarité
 3. compréhension
 4. union
 5. dérivation
 6. dérivation mutuelle
- + 1 : la différence

3.7 Intégration de schémas - Correspondances

Les correspondances

Correspondance d'égalité

O1 et O2 sont en **correspondance d'égalité** si $D1 = D2$ (synonymie par exemple)

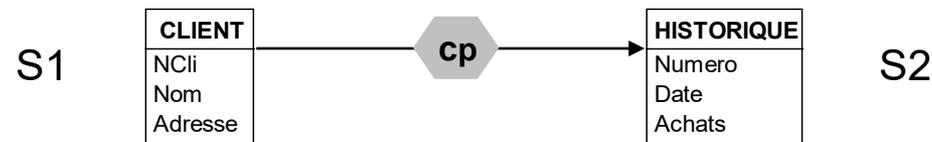
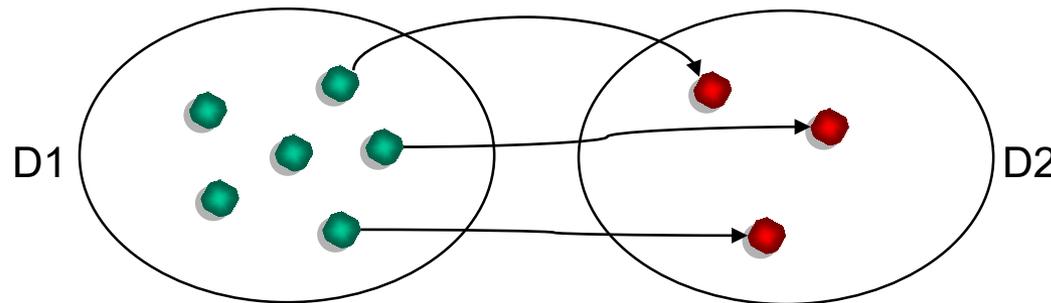


3.7 Intégration de schémas - Correspondances

Les correspondances

Correspondance de complémentarité

O1 et O2 sont en **correspondance de complémentarité** si D2 est un complément de D1

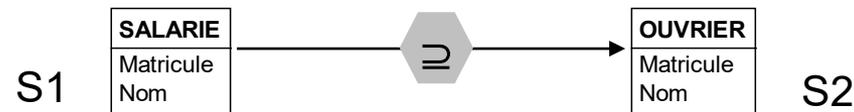
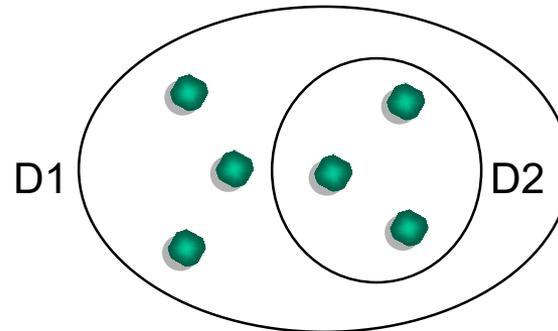


3.7 Intégration de schémas - Correspondances

Les correspondances

Correspondance de compréhension

O1 et O2 sont en **correspondance de compréhension** si $D2 \subseteq D1$

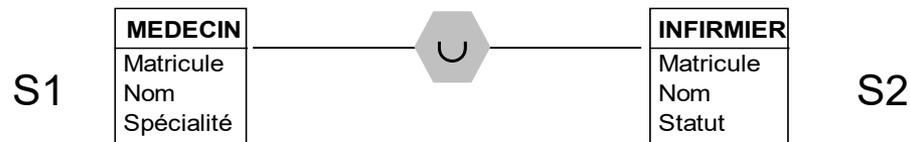
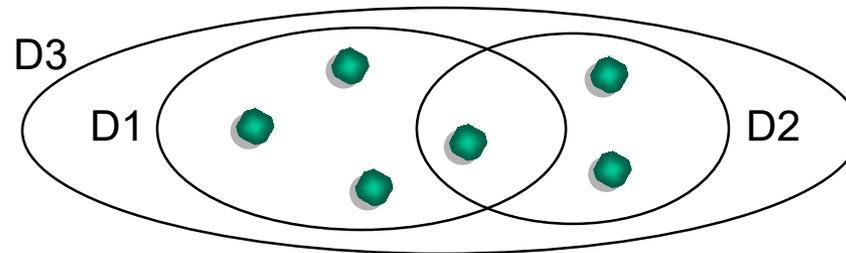


3.7 Intégration de schémas - Correspondances

Les correspondances

Correspondance d'union

O1 et O2 sont en **correspondance d'union** s'il existe $D3 \supseteq D1 \cup D2$

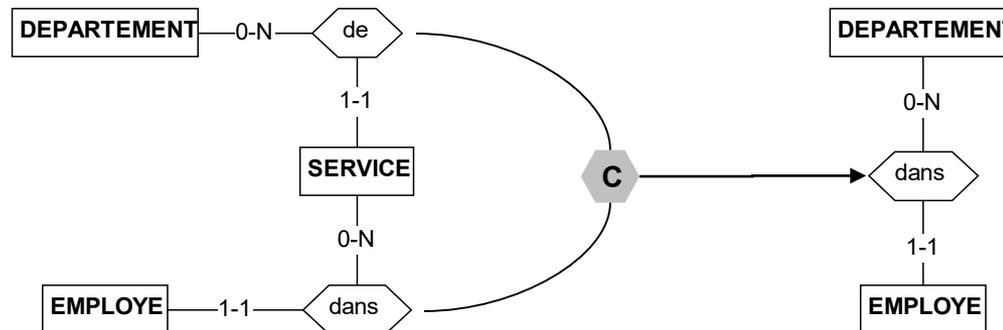
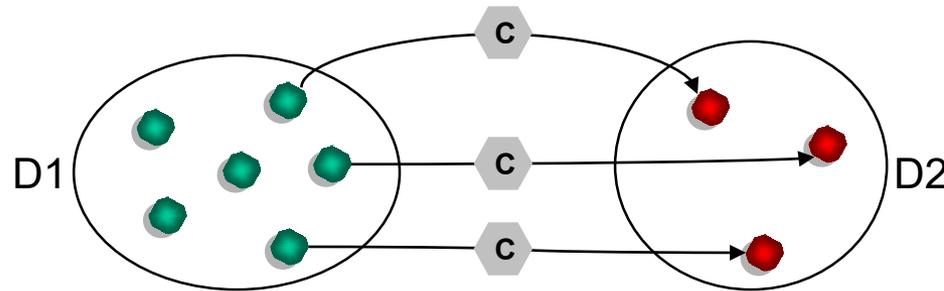


3.7 Intégration de schémas - Correspondances

Les correspondances

Correspondance de dérivation

O1 et O2 sont en **correspondance de dérivation** si $D2 = C(D1)$

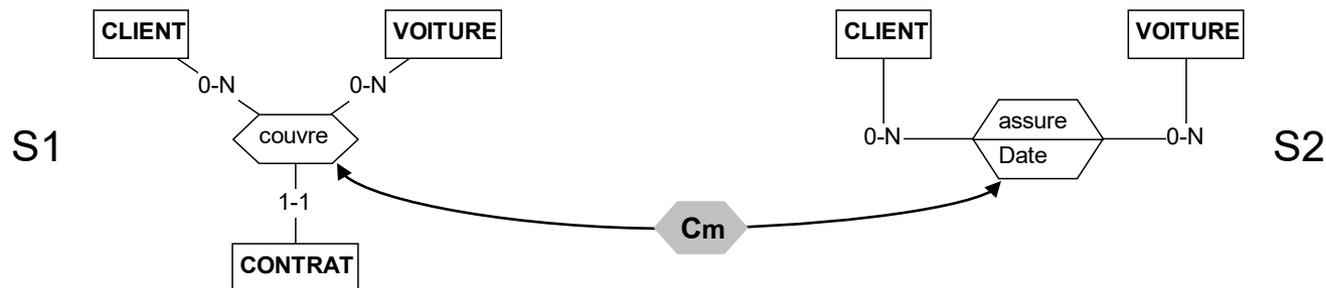
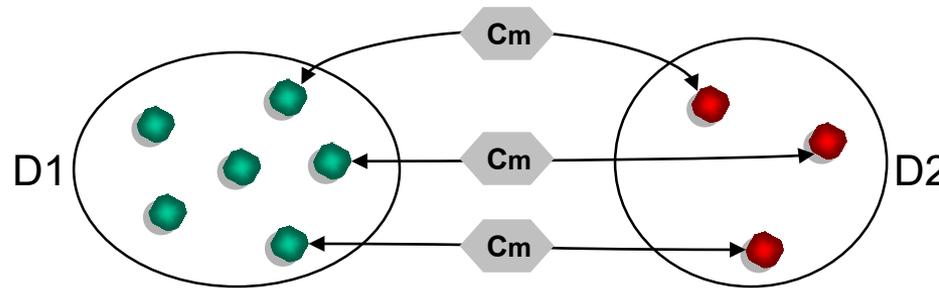


3.7 Intégration de schémas - Correspondances

Les correspondances

Correspondance de dérivation mutuelle

O1 et O2 sont en **correspondance de dérivation mutuelle** si $Cm(D1,D2)$

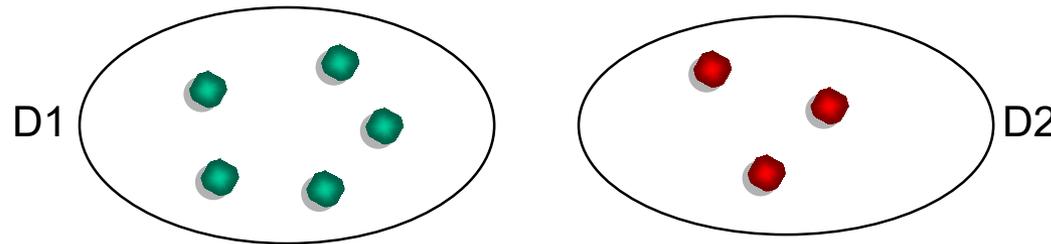


3.7 Intégration de schémas - Correspondances

Les correspondances

Correspondance de différence

O1 et O2 sont en **correspondance de différence** s'il n'existe pas de relation entre D1 et D2. *Intérêt* : infirmer une correspondance apparente (homonymie par exemple).



unités de soins

S1

UNITE
Nom
Localisation
Spécialité

UNITE
Nom
Localisation
Production

S2

unités de fabrication

≠

3.7 Intégration de schémas - Correspondances

Propriétés des correspondances

Correspondances symétriques : égalité, union, dérivation mutuelle (bidirectionnelle), différence

Correspondances non symétriques : complémentarité, compréhension, dérivation
Construction **dominante** vs construction **mineure**

Correspondances homogènes : entre objets de même nature

Correspondances hétérogènes : entre objets de natures différentes

3.7 Intégration de schémas - Correspondances

Comment repérer les objets en correspondance ?

1. Analyse des noms

2. Analyse structurelle

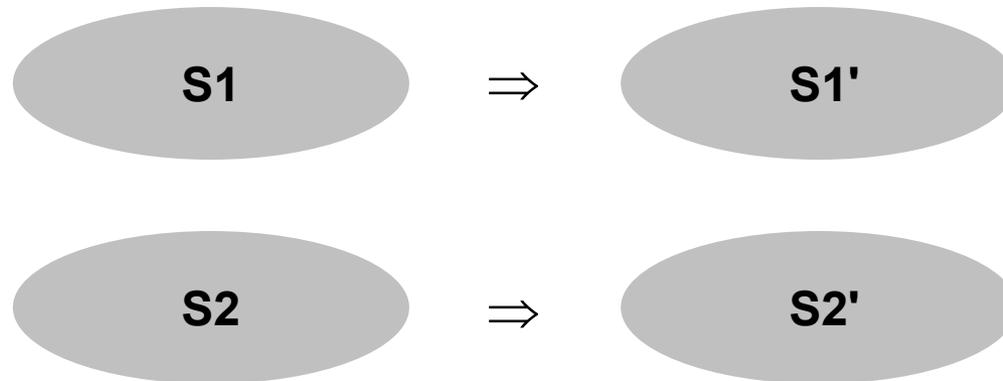
3. Ontologies

3.7 Intégration de schémas - Unification

Etape 2 : Unification des schémas

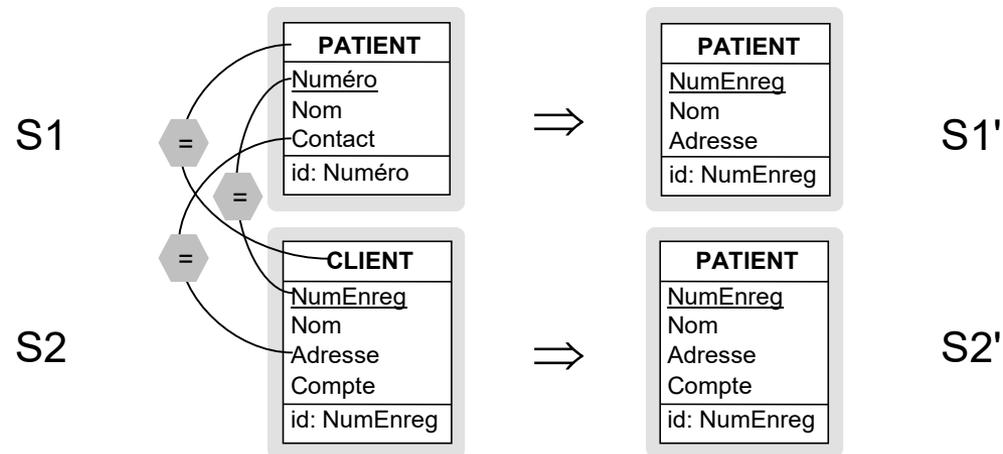
Transformation des constructions en correspondance pour faciliter leur fusion

On distingue les correspondance homogènes et hétérogènes



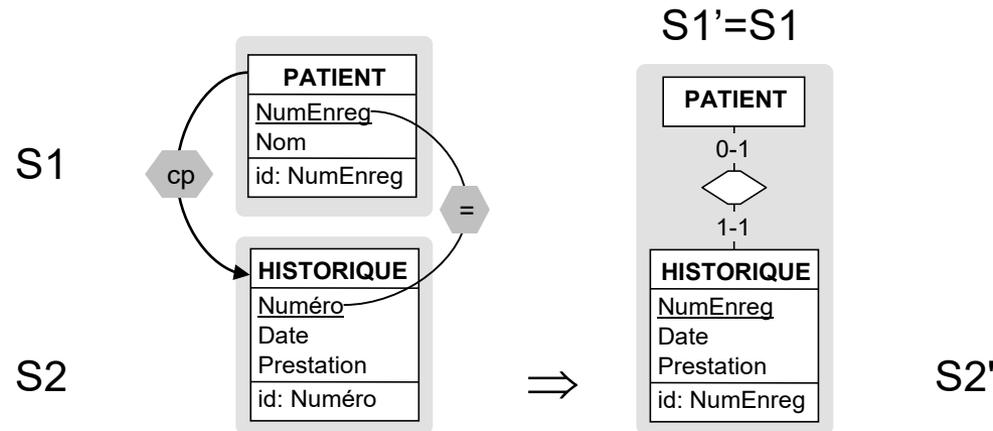
3.7 Intégration de schémas - Unification

Correspondance d'égalité (homogène)



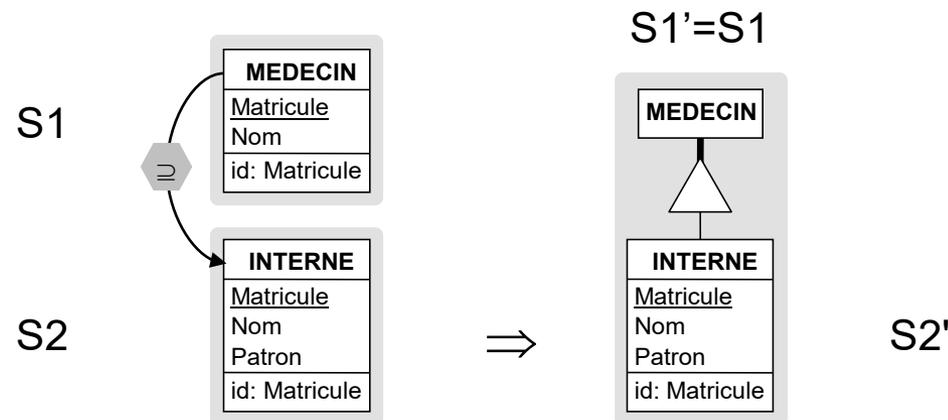
3.7 Intégration de schémas - Unification

Correspondance de complémentarité (homogène)



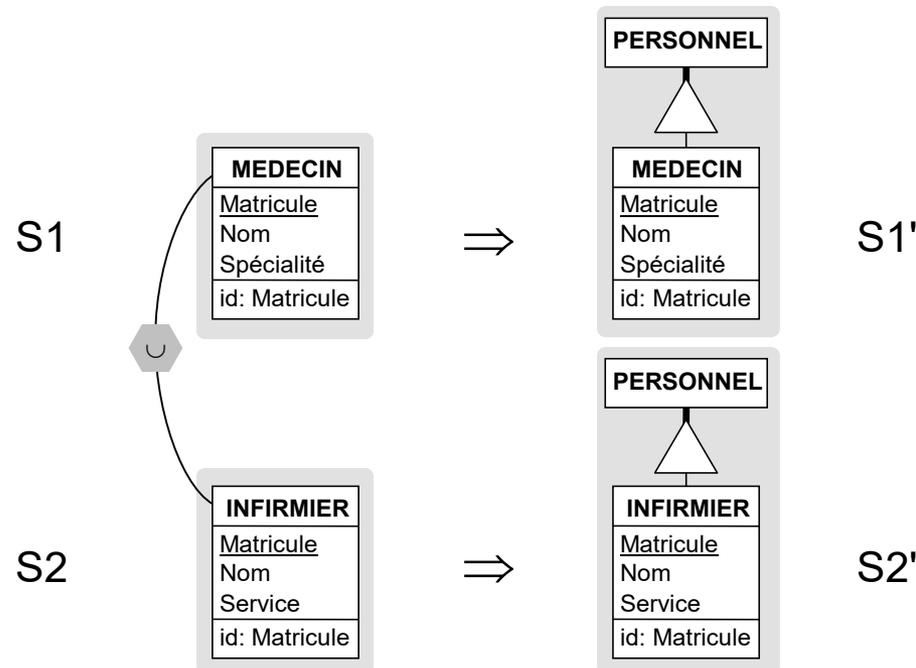
3.7 Intégration de schémas - Unification

Correspondance de compréhension (homogène)



3.7 Intégration de schémas - Unification

Correspondance d'union (homogène)



3.7 Intégration de schémas - Unification

Correspondance de dérivation

Suppression de la construction mineure

Correspondance de dérivation mutuelle

Au cas par cas

Correspondance de différence

Renommage

3.7 Intégration de schémas - Unification

Correspondances hétérogènes

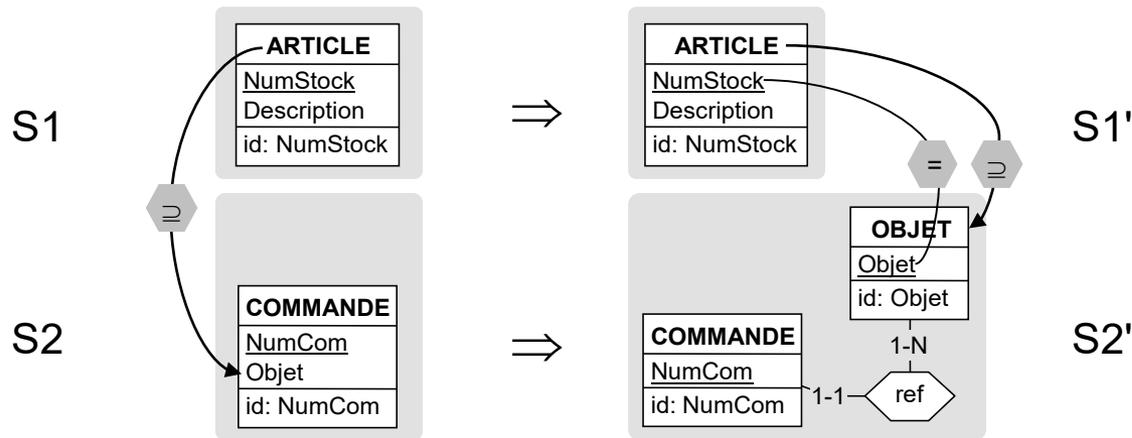
1. Homogénéisation préalable
2. Traitement des correspondances homogènes

Trois classes d'hétérogénéité :

- A. attribut ↔ type d'entités
- B. attribut ↔ rôle
- C. type d'associations ↔ type d'entités

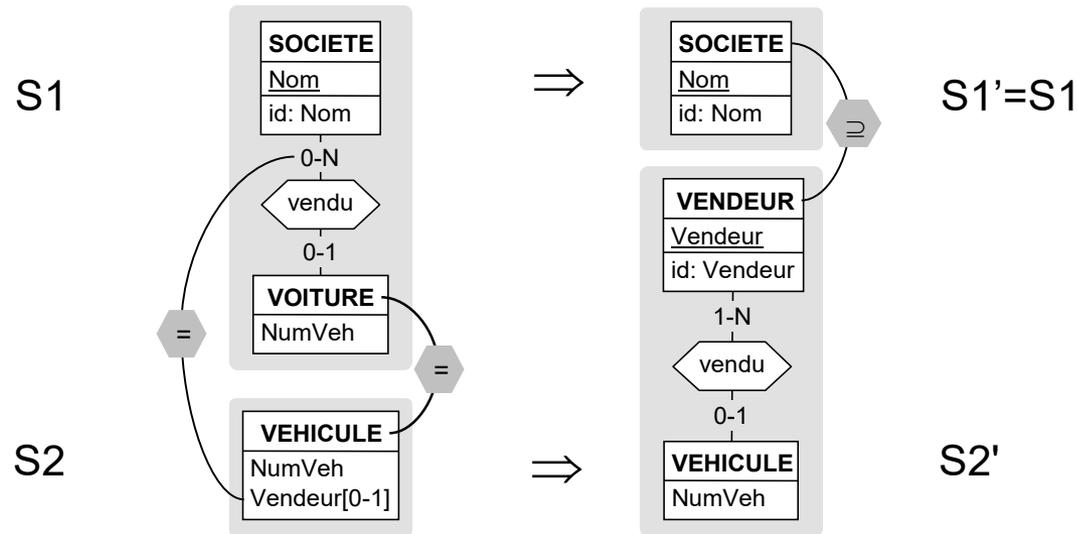
3.7 Intégration de schémas - Unification

A. Correspondances hétérogènes *attribut ↔ type d'entités*



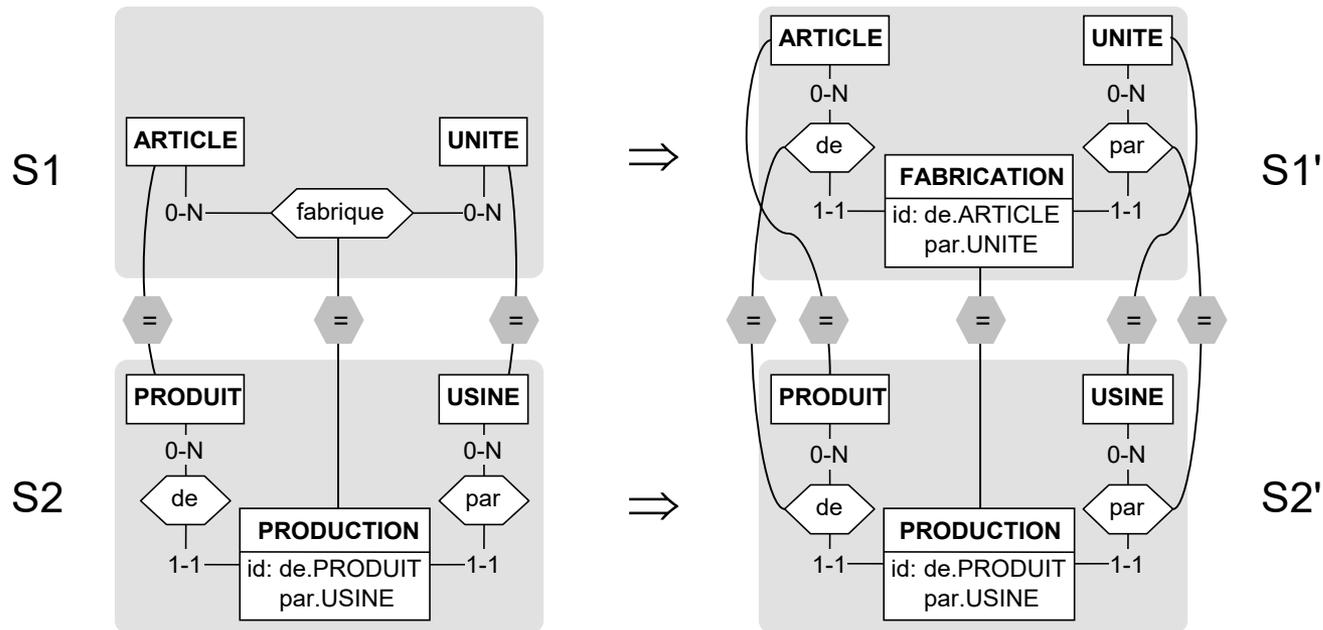
3.7 Intégration de schémas - Unification

B. Correspondances hétérogènes *attribut* ↔ *rôle*



3.7 Intégration de schémas - Unification

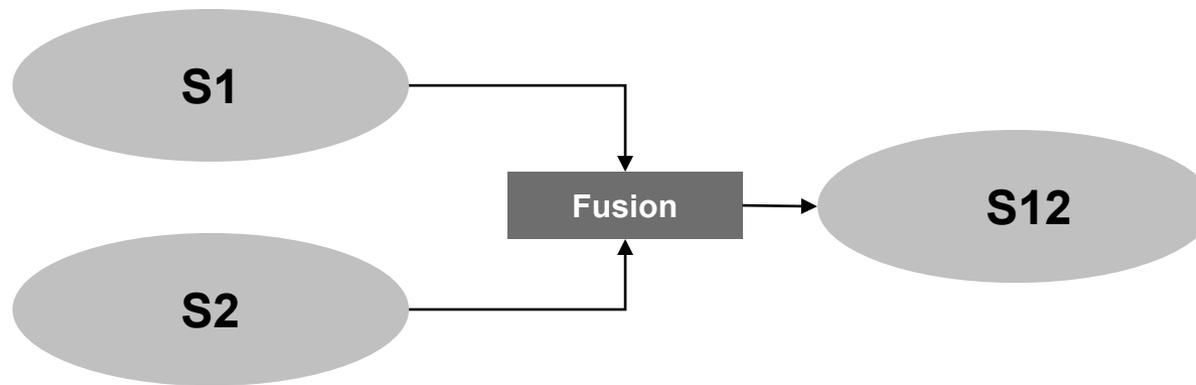
C. Correspondances hétérogènes *type d'associations* ↔ *type d'entités*



3.7 Intégration de schémas - Fusion

Etape 3 : Fusion des schémas unifiés

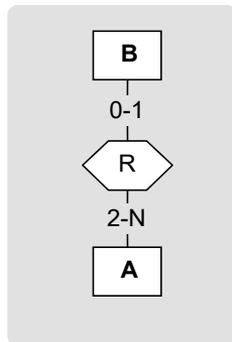
Il reste des correspondances d'égalité implicites (même nature, même nom)



Attention, S12 peut être incorrect et non normalisé

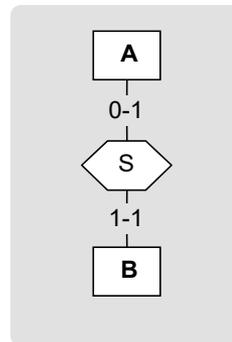
3.7 Intégration de schémas - Fusion

Résolution des conflits sémantiques

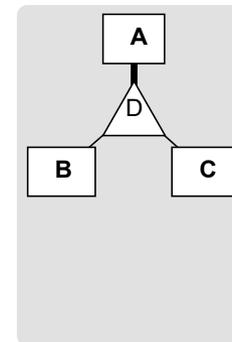


S1

?

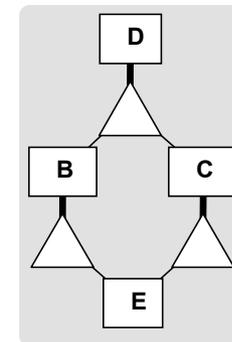


S2



S1

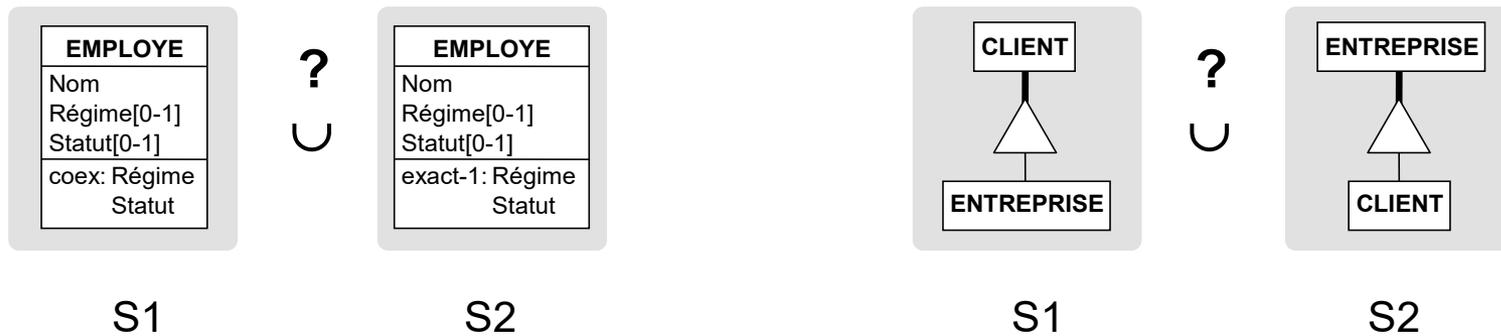
?



S2

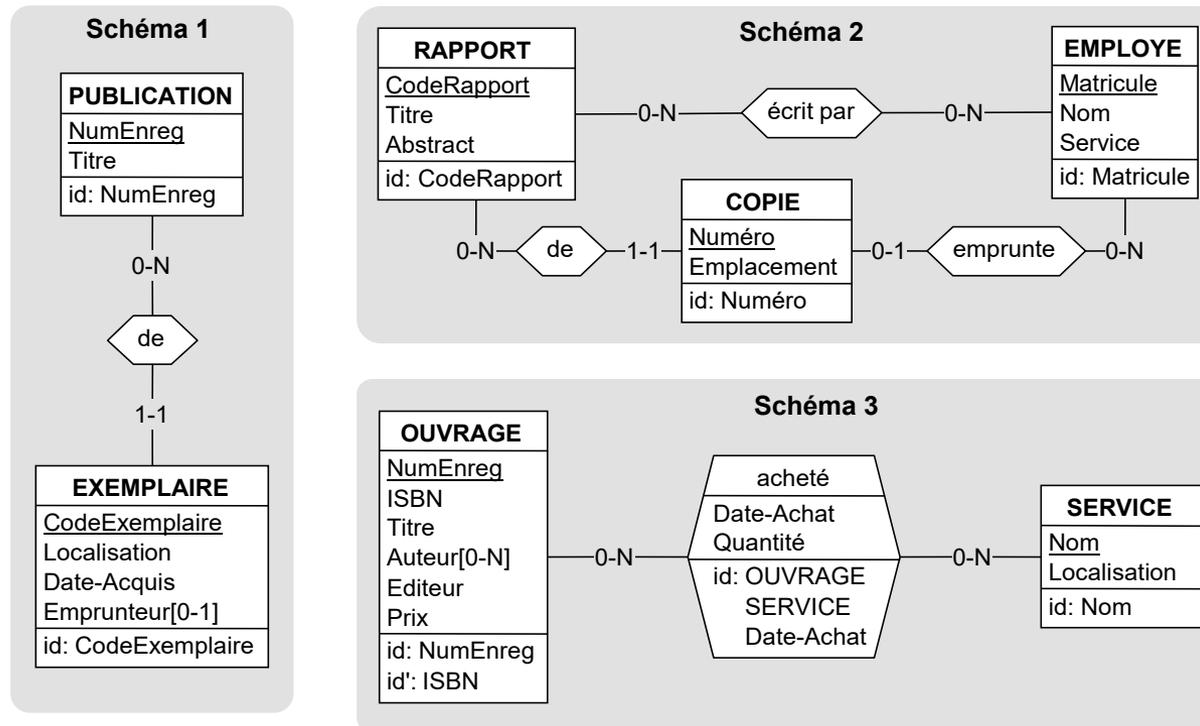
3.7 Intégration de schémas - Fusion

Résolution des conflits sémantiques



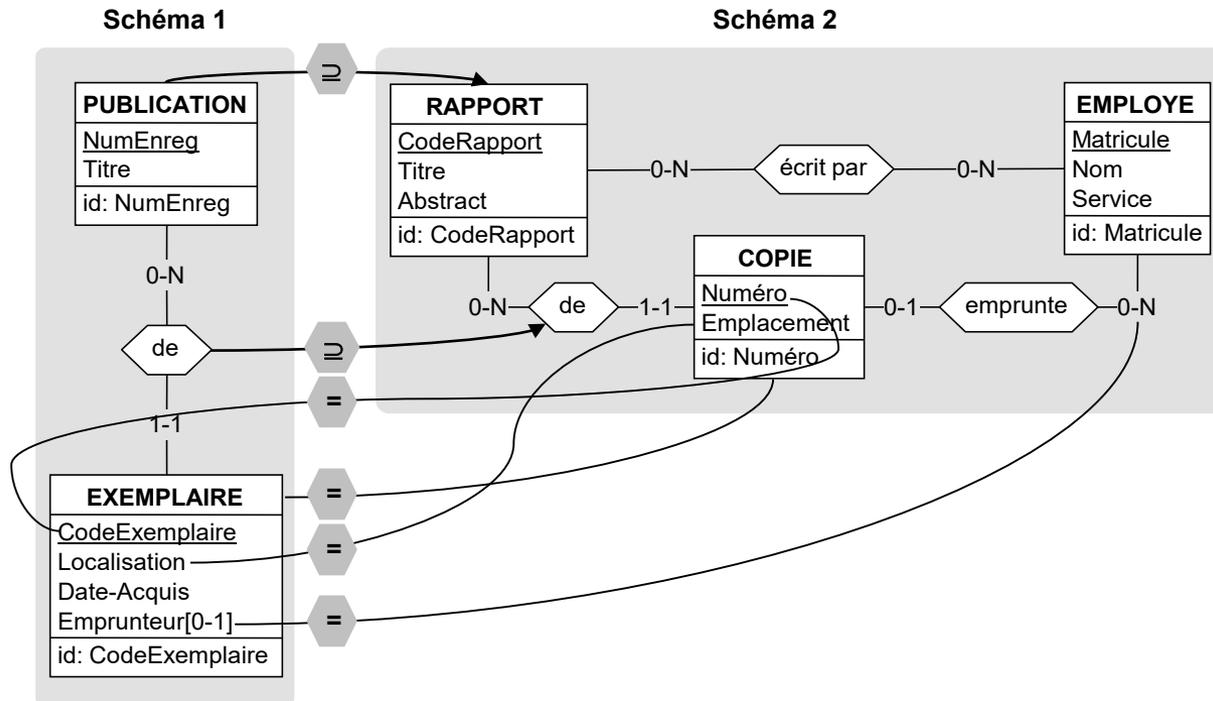
3.7 Intégration de schémas - Exemple

Exemple d'intégration de schémas



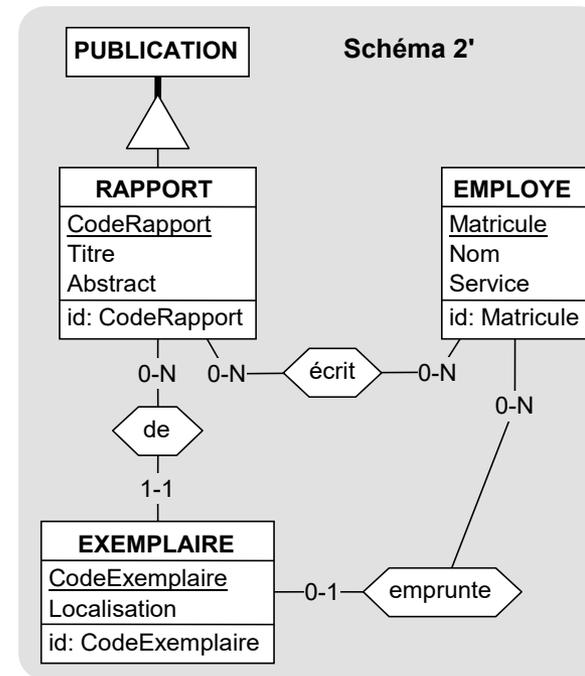
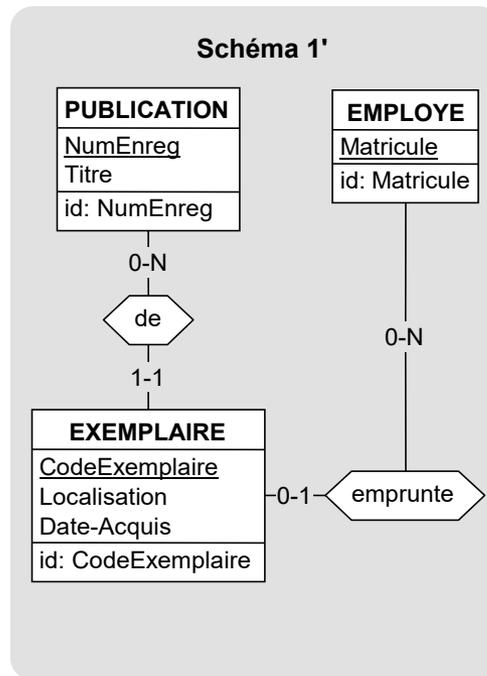
3.7 Intégration de schémas - Exemple

Correspondances entre Schéma 1 et Schéma 2



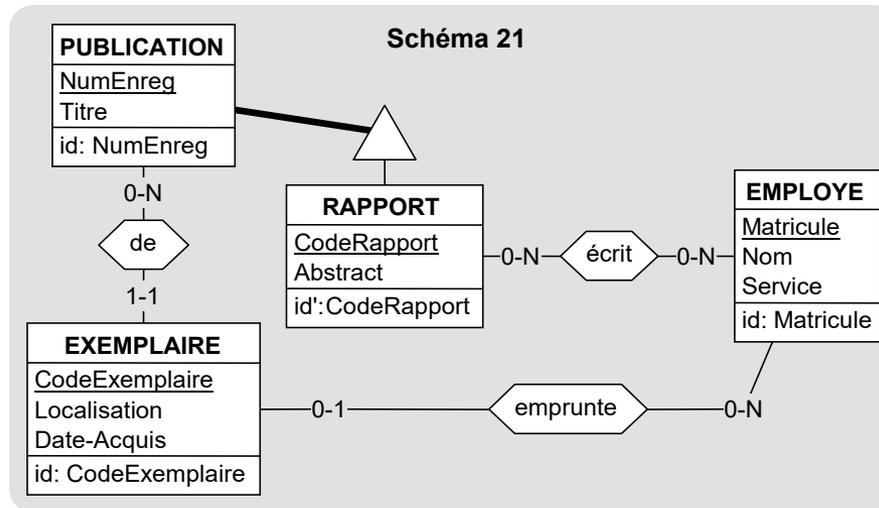
3.7 Intégration de schémas - Exemple

Unification de Schéma 1 et Schéma 2



3.7 Intégration de schémas - Exemple

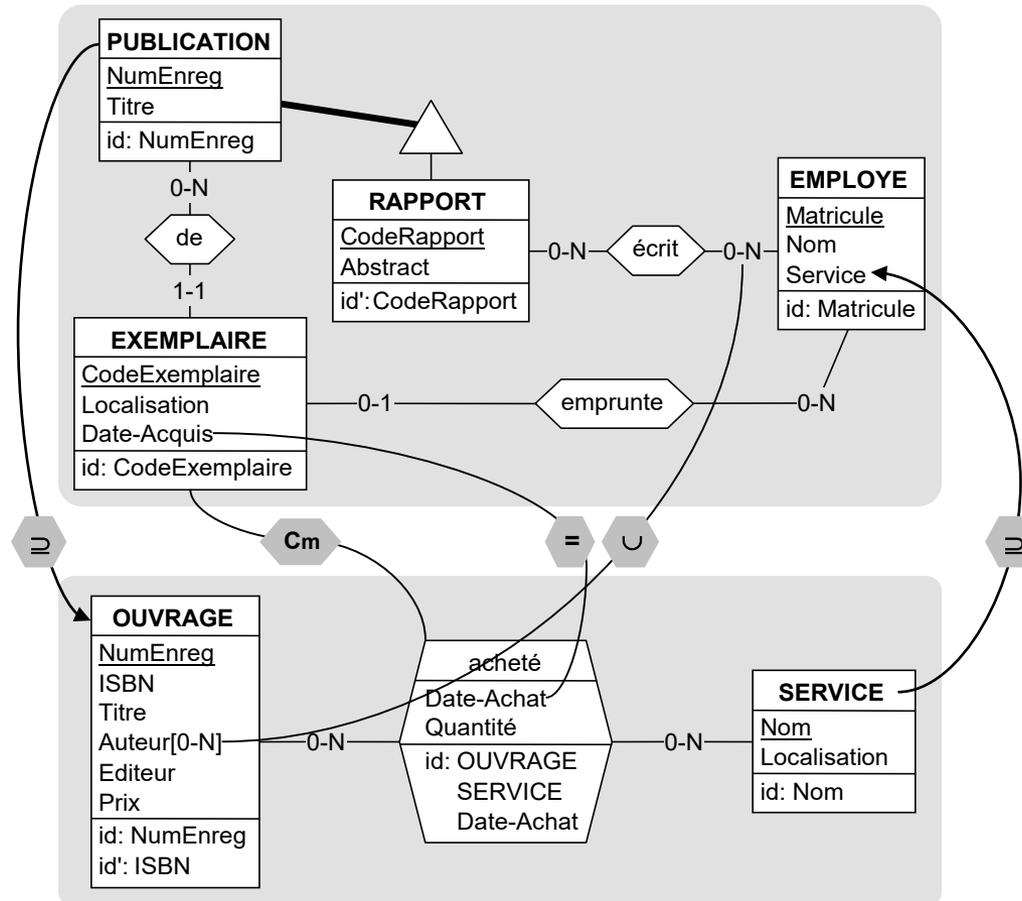
Fusion de Schéma 1' et Schéma 2' en Schéma 21



attention au changement d'identifiant de RAPPORT

3.7 Intégration de schémas - Exemple

Correspondances entre Schéma 21 et Schéma 3



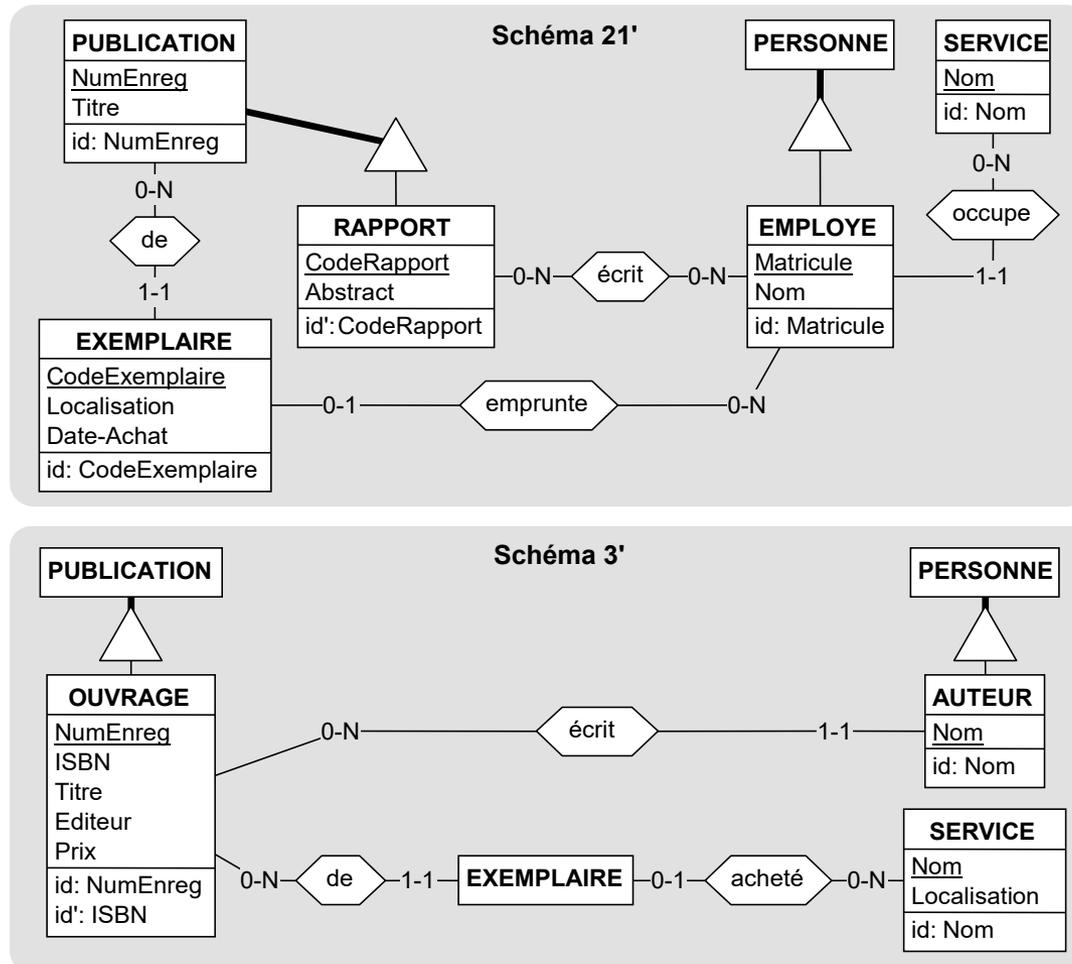
3.7 Intégration de schémas - Exemple

Première tentative d'unification de **acheté** et **EXEMPLAIRE**



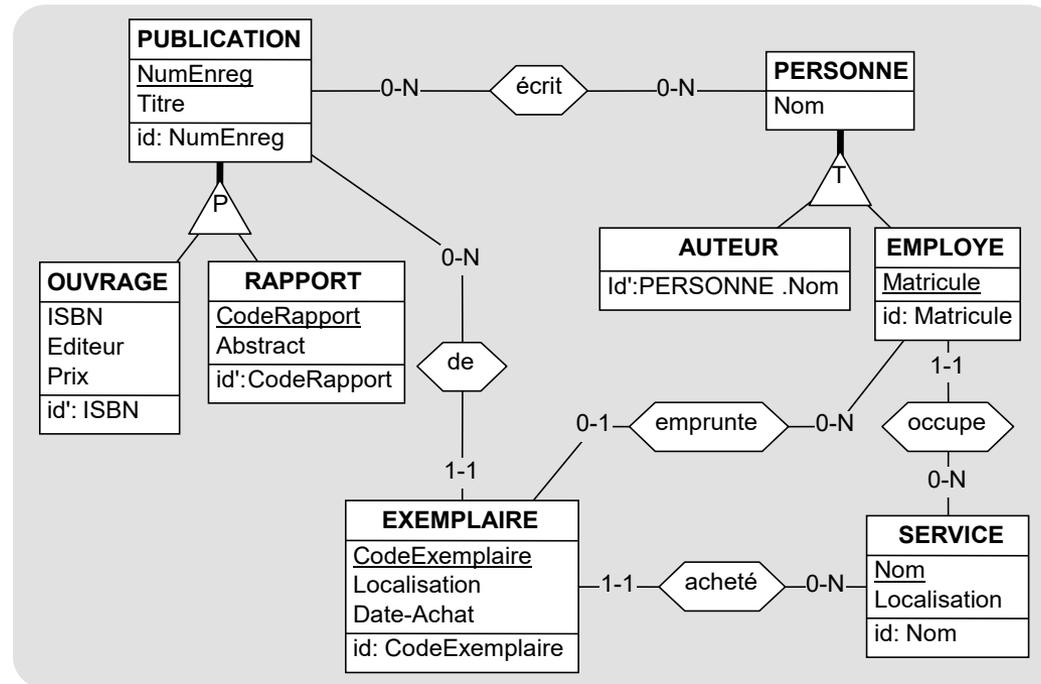
3.7 Intégration de schémas - Exemple

Unification de Schéma 21' et Schéma 3



3.7 Intégration de schémas - Exemple

Fusion de Schéma 21' et Schéma 3' en Schéma 321



L'identifiant primaire de AUTEUR devient secondaire. Pourquoi ?

3.8 Analyse conceptuelle à l'aide d'un AGL

3.8 Analyse conceptuelle à l'aide d'un AGL - Processus

Fonctions de base :

- saisie graphique
- saisie textuelle
- visualisation
- mémorisation

Fonctions avancées :

- interrogation
- correction
- normalisation
- validation
- intégration

3.8 Analyse conceptuelle à l'aide d'un AGL - Fonctions

Interrogation

- langage graphique
- langage textuel

Correction

- repérage de patterns
- correction manuelle

Normalisation

- repérage de patterns
- correction par transformation

Validation

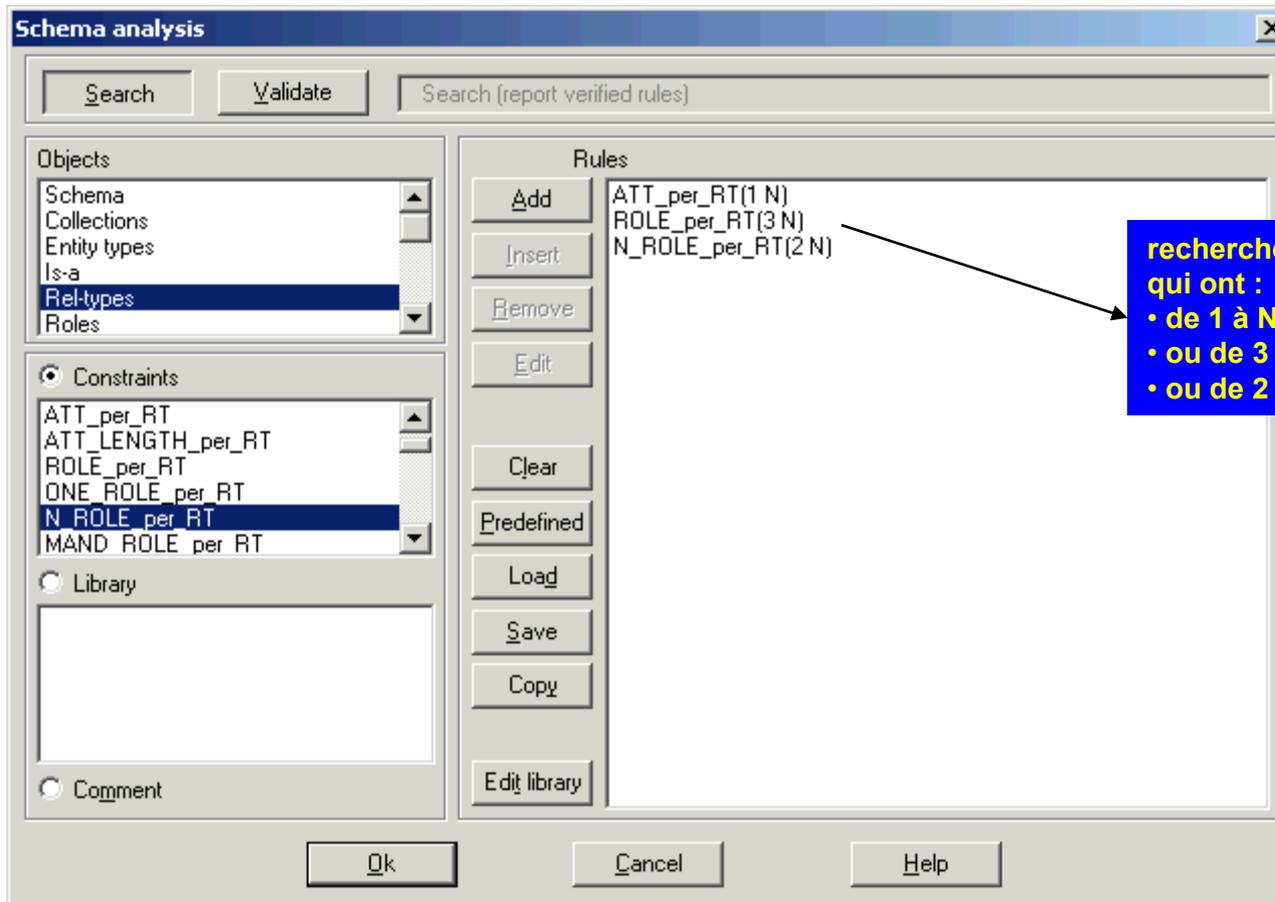
- paraphrase
- génération de prototypes
- repérage de patterns

Intégration

- aide à la détection des correspondances
- résolution ponctuelle (intégration d'objets)
- fusion

3.8 Analyse conceptuelle à l'aide d'un AGL - DB-MAIN

Analyse de schémas - Recherche de patterns

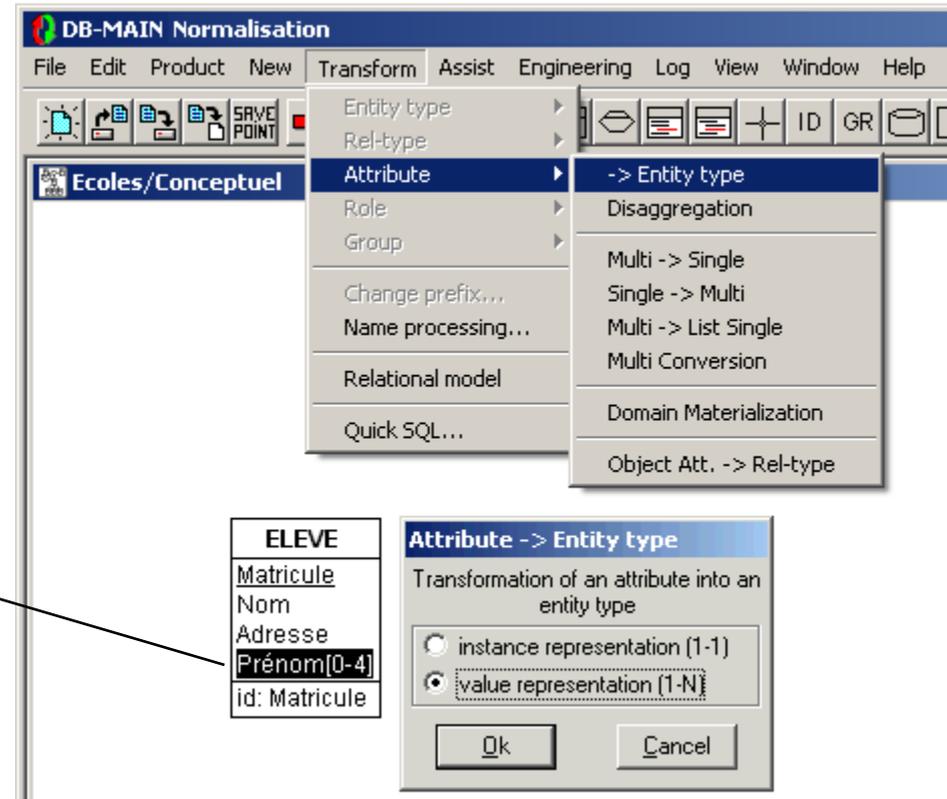


rechercher les TA "complexes", c-à-d qui ont :

- de 1 à N attributs
- ou de 3 à N rôles
- ou de 2 à N rôles "many"

3.8 Analyse conceptuelle à l'aide d'un AGL - DB-MAIN

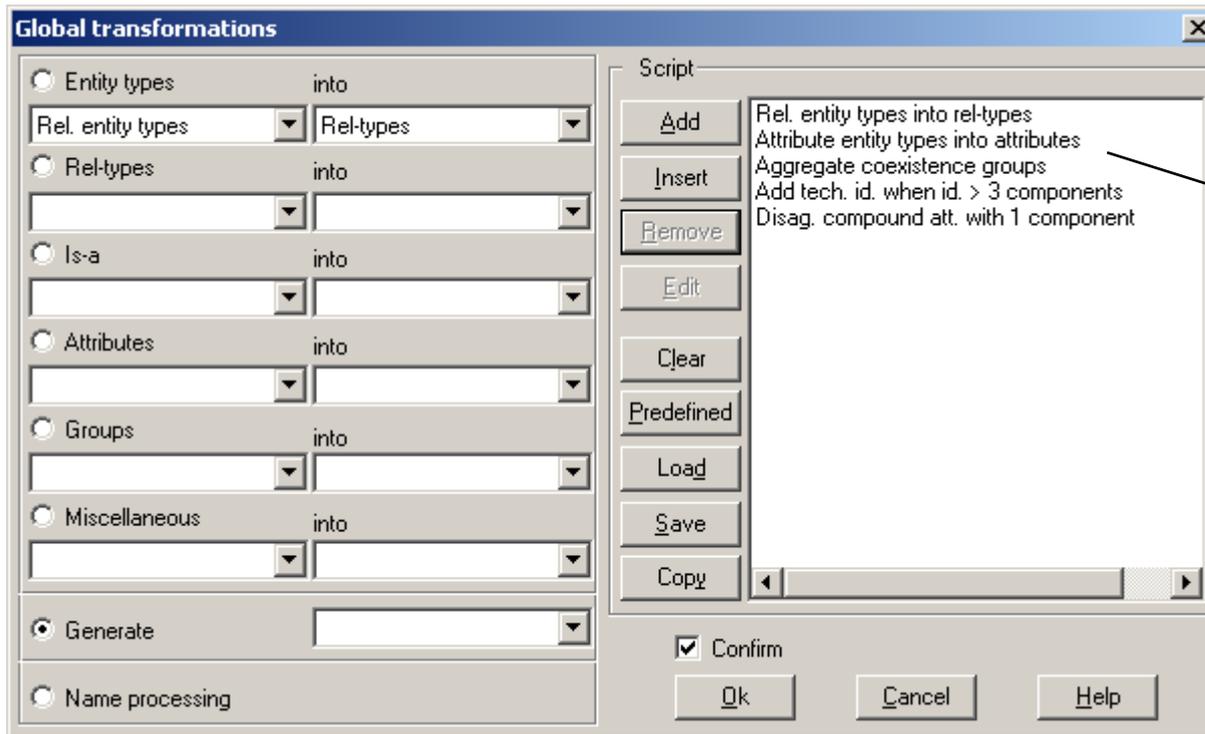
Transformation de schéma élémentaire



transformation d'un attribut multivalué en type d'entités

3.8 Analyse conceptuelle à l'aide d'un AGL - DB-MAIN

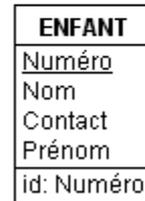
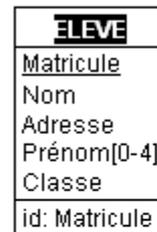
Transformation de schémas - Globale



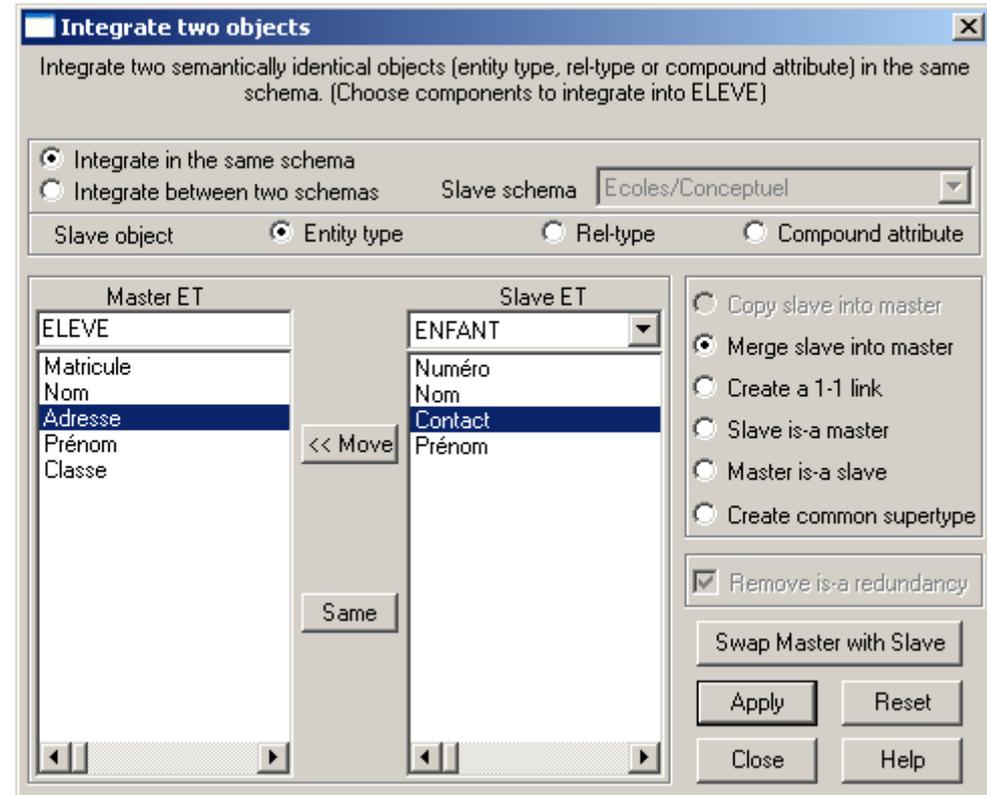
petit script de normalisation

3.8 Analyse conceptuelle à l'aide d'un AGL - DB-MAIN

Intégration de schémas (intégration d'objets)



intégration d'ELEVE et ENFANT en correspondance d'égalité



Fin du module 3

Module suivant :
4. Conception logique relationnelle

1. Méthodologie des BD
2. Le modèle Entité-association
- 3. Analyse conceptuelle**
4. Conception logique relationnelle

5. Conception physique
6. Production du code
7. Rétro-ingénierie