

MERISE

Laurent HUBERT

1 Introduction

Le Système d'Information, appelé aussi SI, est un ensemble de tâches complexes regroupées en modules spécialisés qui composent l'applicatif informatique : le logiciel.

Ces tâches complexes sont généralement un assemblage de tâches plus simples. Ces tâches simples sont les briques de base de l'applicatif.

Si nous devons créer une analogie nous pourrions dire que ces tâches simples sont comme les briques qu'un maçon assemble pour ériger une maison.

Le logiciel, tout comme une maison, a besoin d'un plan de conception réalisée par un architecte.

Une maison conçue sans plan risque de présenter, une fois finie, plus d'une erreur de conception. Il en est de même pour un logiciel. Le logiciel sans études préalables, construit sans méthodologie, risque de surprendre son utilisateur !

La méthode MERISE

MERISE est une méthode créée et développée en France à partir du milieu des années 1970.

Subventionnée et soutenue par l'État, elle fut utilisée et enseignée afin de permettre la communication entre les administrations et ses sous-traitants.

Cette méthode présente une manière rationnelle et claire d'étudier et de formaliser le fonctionnement du Système d'Informations d'une entreprise.

Cette méthode répond à la question déjà posée lors de la session concernant UML :
Comment passer d'un besoin exprimé à un code informatique ?



FIGURE 1.1 – Des Besoins au logiciel

Nous allons donc découvrir dans les sections suivantes la méthode que propose MERISE pour répondre à cette question.

1.1 Quelques références

- ▷ <http://cyril-gruau.developpez.com/merise/>
- ▷ <http://sql.developpez.com/modelisation/merise/>
- ▷ <http://merise.developpez.com/faq/>

1.2 Rôle des Systèmes d'Information

L'information dans une entreprise, c'est :

▷ des données, des clients, des quantité en stock,...

▷ des dépendances entre informations

ex : facture \Rightarrow produit

▷ des échanges et transmissions d'informations

commande → entrepôt → liste produits livrés → service
facturation, ...

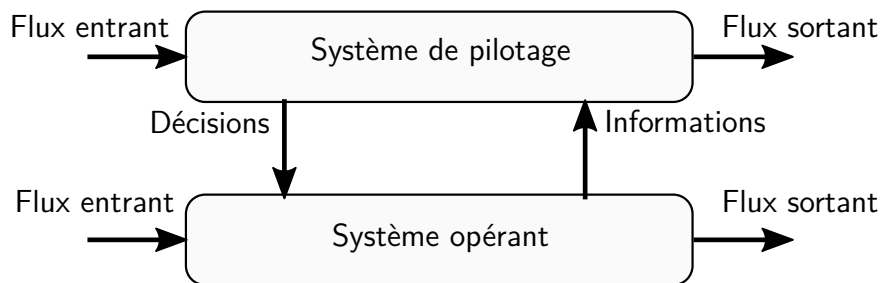
▷ des règles de gestion
facture \Rightarrow client, ...

La méthode Merise se caractérise par :

- ▷ une approche systémique (l'entreprise vue comme un ensemble de systèmes) ;
- ▷ une séparation des données (le côté statique) et des traitements (le côté dynamique) ;
- ▷ une approche par niveaux.

L'entreprise est donc vue comme un système.

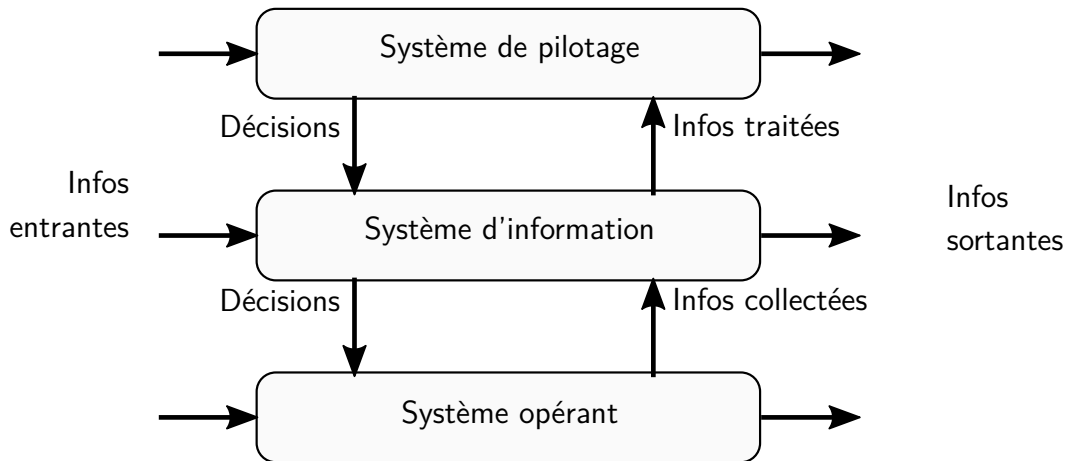
Ce système est lui-même composé de sous-systèmes :



1.3 L'approche systémique

1.3.1 L'information dans l'entreprise (approche Merise)

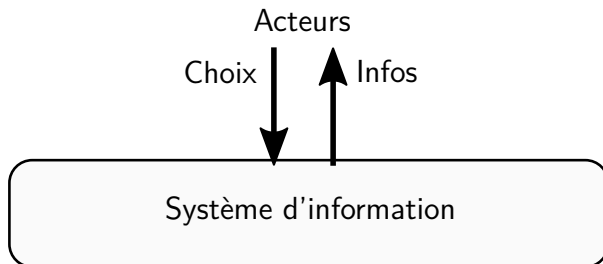
Le système d'information est l'interface entre le système décisionnel et le système opérationnel :



1.3.2 L'information dans l'entreprise (approche UML)

Des données, des traitements, la circulation d'informations, ...

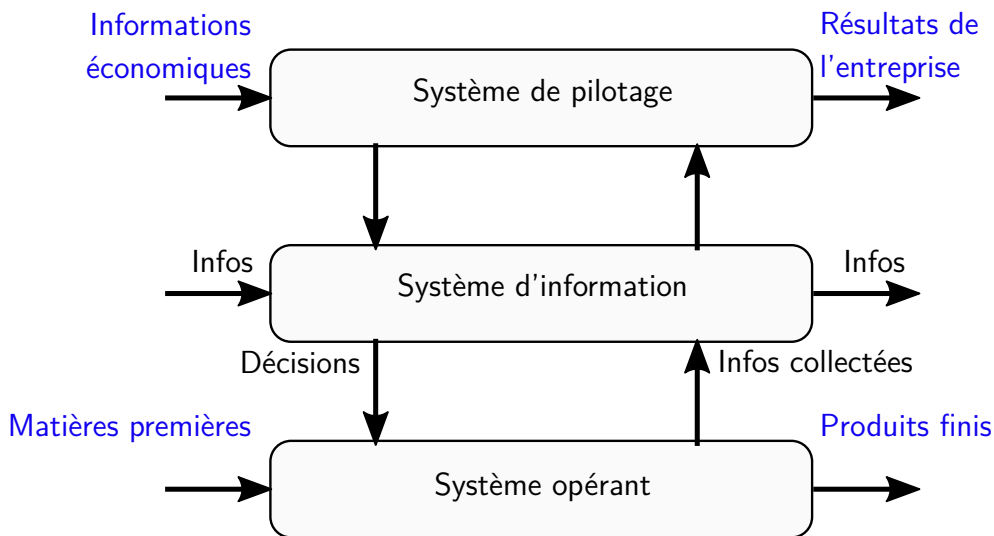
Cette approche est plus orienté sur l'interaction utilisateurs / système d'information :



1.3.3 Une interface entre les sous-systèmes

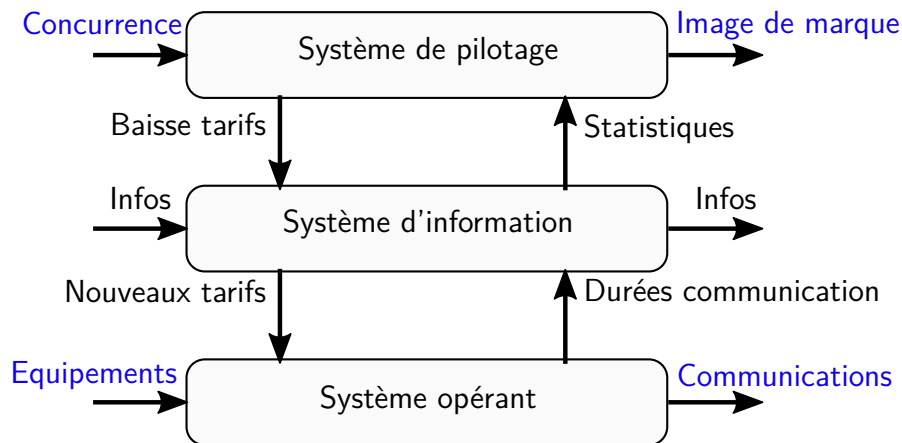
Le système d'information est la mémoire de l'entreprise ;

Il sert d'**interface entre le système décisionnel et le système opérationnel** :



1.4 Exemple

Exemple inspiré d'un opérateur de téléphonie, la société Chausse Mobile :



1. On peut faire une lecture chronologique du schéma précédent :
2. La société Chausse Mobile surveille le comportement de la concurrence ;
3. les durées des communications ont été relevées en permanence et ont alimenté le système d'information ;
4. les têtes pensantes de Gaule Radiocom peuvent interroger le système d'information, faire des statistiques ;
5. le système de pilotage peut alors décider la plus judicieuse baisse de tarifs ;
6. qui sera répercutée sur le système opérant de la facturation...

Ainsi arrive une baisse des tarifs :

Janvier 2001, la société Chausse Mobile baisse ses tarifs !

0.22€ la minute au lieu de 0.28€ ;

0.60€ le crédit-temps* (1mn) au lieu de 0.74€ (3mn)

Une baisse de plus de 20% par minute de communication !

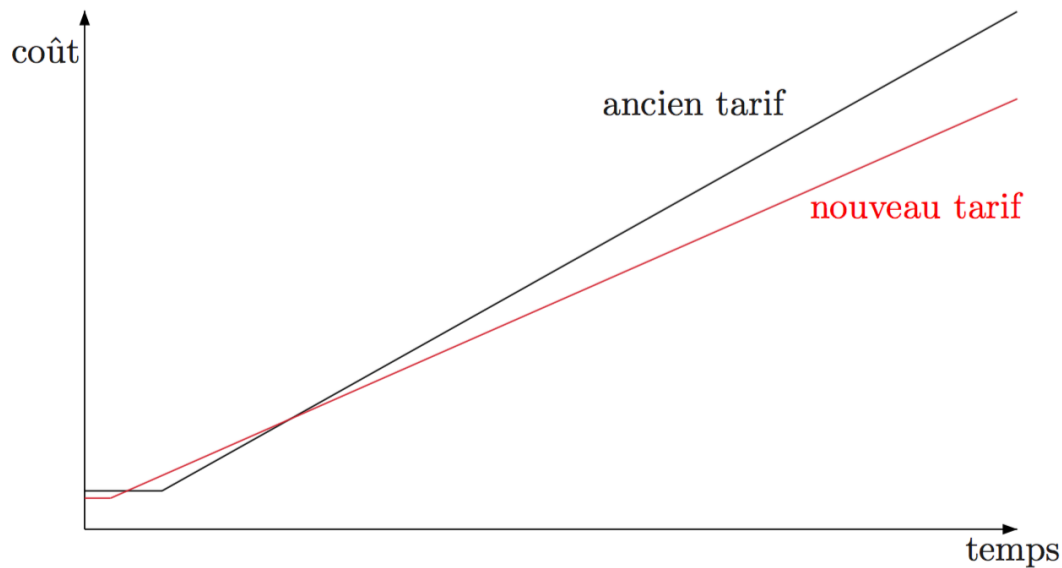
Pourquoi Chausse Mobile se montre-t-elle aussi généreuse ???

* le crédit-temps correspond à la première tranche de facturation. Avant janvier 2001, toute communication impliquait la facturation d'une tranche initiale de 3mn à 0.74€. Passée ces 3mn, la facturation continuait par tranches d'1mn à 0.28€.

Janvier 2001, la société Chausse Mobile baisse ses tarifs !

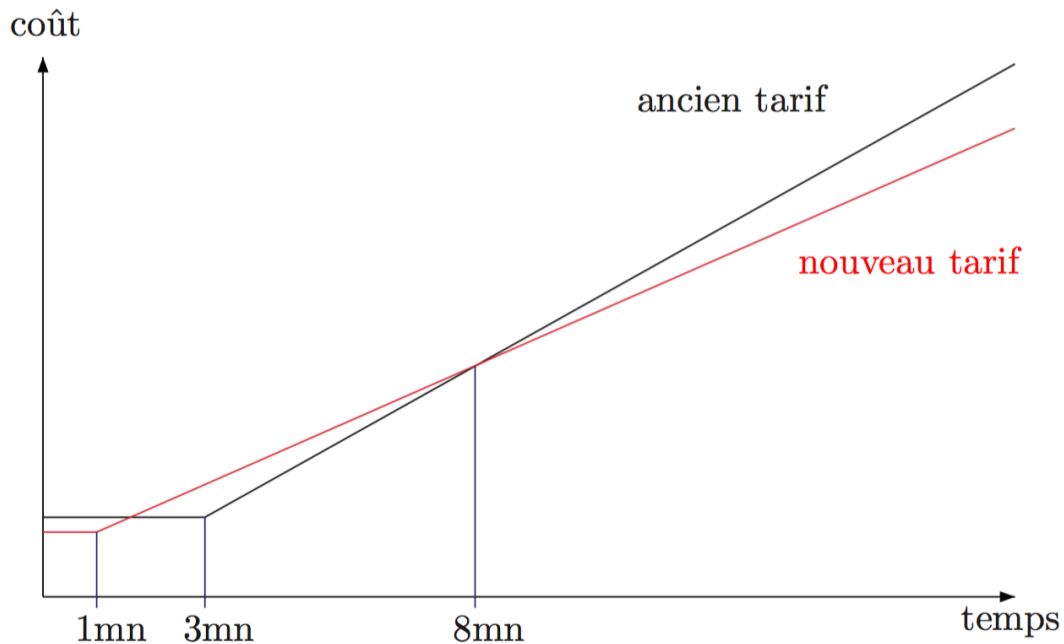
0.22€la minute au lieu de 0.28e;

0.60€le crédit-temps* (1mn) au lieu de 0.74€(3mn)



Parce que Chausse Mobile a bien analysé les statistiques des communications de ses abonnés.

La majorité des communications durent moins de 8 mn !



L'informatique des années 80 était axée sur la production, la mémorisation et le traitement des données de l'activité de l'entreprise : Informatique Opérationnelle.

Les années 90 marquent le développement de l'Informatique Décisionnelle.

Les années 2000 sacrent l'avènement du réseau.

Informatique Opérationnelle : production et traitement de données : paie, comptabilité, commandes, factures, ...

Informatique Décisionnelle : Les données sur l'activité sont filtrées, historisées et traitées pour faciliter la prise de décision. Apparition de Datawarehouse, Data-Center, etc.

1.5 La séparation des Données et des Traitements

1.5.1 Les Données (ou informations)

L'information est l'émission ou la réception de signaux oraux ou écrits, sonores, visuels ou multimédias dont le but est de déclencher les processus alimentant l'échange, base naturelle et indispensable de l'animation de l'organisation.

Les informations se recueillent à l'intérieur du domaine à étudier. La liste d'informations est constituée de plusieurs façons :

- ▷ l'interview ;
- ▷ l'étude des documents internes ;
- ▷ l'étude des documents externes.

L'interview

Une des phases du recueil d'information est un entretien avec les différents acteurs de l'organisation.

Cet entretien permet de définir le périmètre de l'applicatif futur.

Les informations orales sont classées et regroupées en parties distinctes. Ainsi, les informations concernant l'enregistrement des données de l'organisation seront regroupées.

Etudes des documents internes

Les documents internes (factures, bons de livraison, ordres de fabrication) recèlent des informations qui sont souvent omises lors des entretiens.

Ces oublis sont dus au caractère automatique et récurrent de ces informations.

Les personnes qui les manipulent au quotidien oublient souvent de les citer tant elles leur paraissent évidentes.

Etudes des documents externes

L'étude des documents externes (factures des fournisseurs, bons de livraison fournisseurs...) tout comme l'étude des documents internes permet de découvrir des informations oubliées lors des interviews et de découvrir aussi quelques règles de gestion.

Pour ce recueil d'informations, il est nécessaire de respecter certaines règles pour éviter des erreurs futures.

Avant d'ajouter une information, il est impératif de s'assurer qu'elle n'est pas déjà présente. Par exemple, un numéro client peut apparaître sur un bon de livraison et sur une facture. Ce n'est pas la peine de le répertorier deux fois.

De même, une information peut être synonyme d'une autre.

Par exemple sur le bon de livraison il apparaît « Code client » et sur la facture « Numéro Client ». Il est impératif de ne garder qu'une seule des deux informations.

1.5.2 Les différents types d'informations

Les informations élémentaires et mémorissables

Les informations élémentaires sont des informations dont les valeurs **ne peuvent pas être inventées**, elles ne sont pas déductibles d'autres informations.

Par exemple, un nom de client ou sa raison sociale ne peuvent pas être inventés. Une quantité commandée ne peut pas non plus être inventée.

Une information doit être **atomique**, c'est-à-dire non décomposable.

Par exemple si l'information *Adresse* doit contenir « 36, rue de la paix 75000 Paris » celle-ci peut être décomposée en plusieurs informations élémentaires :

- ▷ Adresse ;
- ▷ Code postal ;
- ▷ Ville.

Chaque valeur prise par une information est appelée une **occurrence**. Par exemple, l'information *Nom* peut avoir les occurrences suivantes :

- ▷ Baptiste ;
- ▷ Durand.

Les informations calculées

Les informations calculées sont déductibles des informations élémentaires.

Par exemple, le total d'une ligne de commande est le résultat de la multiplication du prix de vente hors taxe et de la quantité commandée.

Les traitements

Ils sont collectés comme les informations via un processus d'interview et d'étude des documents.

Ils peuvent être de deux sortes :

- ▷ automatiques ;
- ▷ manuels.

Ils sont déclenchés par l'arrivée d'évènements.

La gestion des traitements sert à identifier les fonctionnalités selon une approche qui va du général au particulier et qui définit leur découpage et leur enchaînement.

1.6 Une approche par niveaux

Pour la conception d'un SI, il est nécessaire de considérer quatre niveaux d'étude :

- ▷ Le niveau conceptuel.
- ▷ Le niveau organisationnel.
- ▷ Le niveau logique.
- ▷ Le niveau physique.

1.6.1 Le niveau conceptuel

Le niveau conceptuel consiste à concevoir le SI en faisant abstraction de toutes les contraintes techniques ou organisationnelles et cela tant au niveau des données que des traitements.

Le niveau conceptuel répond à la question **Quoi ?**

(le *quoi* faire, avec *quelles* données).

Le formalisme Merise employé sera :

- ▷ Le Modèle Conceptuel des Données (MCD).
- ▷ Le Modèle Conceptuel des Traitements (MCT).

1.6.2 Le niveau organisationnel

Le niveau organisationnel a comme mission d'intégrer dans l'analyse les critères liés à l'organisation étudiée.

Le niveau organisationnel fera préciser

- ▷ les notions de temporalité,
- ▷ de chronologie des opérations,
- ▷ d'unité de lieu,

Il définira

- ▷ les postes de travail,
- ▷ l'accès aux bases de données...

Les questions posées, au niveau des traitements, sont :

- ▷ Qui ?
- ▷ Où ?
- ▷ Quand ?

Le formalisme Merise employé sera :

- ▷ Le Modèle Organisationnel des Données (MOD).
- ▷ Le Modèle Organisationnel des Traitements (MOT).

1.6.3 Le niveau logique

Le niveau logique est indépendant du matériel informatique, des langages de programmation ou de gestion des données.

C'est la réponse à la question **Avec quoi ?**

Le formalisme sera :

- ▷ Le Modèle Logique des Données (MLD).
- ▷ Le Modèle Logique des Traitements (MLT).

1.6.4 Le niveau physique

Le niveau physique permet de définir l'organisation réelle (physique) des données.

Il apporte les solutions techniques, par exemple sur les méthodes de stockage et d'accès à l'information.

C'est la réponse au Comment ?

Le formalisme employé sera :

- ▷ Le Modèle Physique des Données (MPD).
- ▷ Le Modèle Opérationnel et physique des Traitements (MOpT).

1.6.5 Tableau récapitulatif

Niveaux	Données	Traitements
Conceptuel	Modèle Conceptuel des Données	Modèle Conceptuel des Traitements
Organisationnel	Modèle Organisationnel des Données	Modèle Organisationnel des Traitements
Logique	Modèle Logique des Données	Modèle Logique des traitements
Physique	Modèle Physique des Données	Modèle Opérationnel et Physique des Traitements

2 Démarche de la méthode MERISE

2.1 Analyse Informatique

La conception d'un Système d'Information est une tâche complexe et de haut niveau qui nécessite un bon pouvoir d'abstraction et la prise en compte d'un grand nombre de contraintes et d'un grand nombre de personnes. D'où la nécessité d'utiliser des méthodes ou méthodologies.

La **méthode MERISE** est une méthode (française), développée dans les années 70/80,

- ▷ **formalisée,**
- ▷ **complète,**
- ▷ **détaillée**

Elle garantit (en principe !) une informatisation réussie.

- ▷ **formalisée** : utilisation d'outils logiques : graphes, règles, ...
- ▷ **complète** : de la décision d'informatisation à la mise en œuvre effective,
- ▷ **détaillée** : de la technique d'interview jusqu'au commentaire de programmes...

Les principales caractéristiques de la méthode MERISE sont :

- ▷ Une approche globale menée parallèlement sur les données et les traitements ;
- ▷ Une description du système d'information en trois niveaux :
 - le niveau conceptuel (le **quoi**) ;
 - le niveau organisationnel et logique (**qui fait quoi et où**)
 - le niveau physique (**comment**).

Le processus de développement est découpé en étapes :

- ▷ **l'étude préalable** : elle aboutit à une prise de décision d'informatisation, en cas de décision positive, elle est suivie par
- ▷ **l'étude détaillée** : elle aboutit à un cahier de réalisation avec affectation des tâches
- ▷ **Réalisation** : écriture des programmes et implantation des bases
- ▷ **Mise en œuvre et maintenance.**

La structure de travail comporte

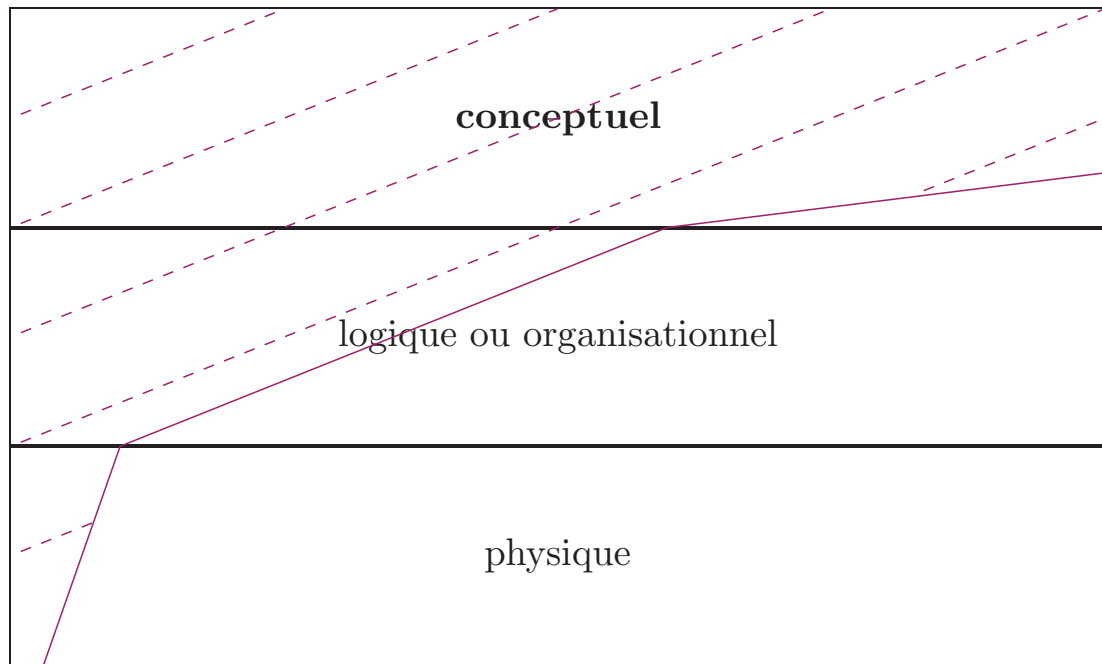
- ▷ des *informaticiens*,
- ▷ des **représentants de la direction**,
- ▷ des **chefs de service**,
- ▷ des **utilisateurs finaux**.

Analyse Informatique : étude préalable

- étude de l'existant ; règles de gestion
- Modèle Conceptuel des Données (MCD) et Modèle Organisationnel des Traitements (MOT) actuels
- diagramme des flux de documents
- proposition d'un MCD et d'un Modèle Conceptuel des Traitements (MCT)
- bilan qualitatif

⇒ cahier des charges

Analyse Informatique : étude préalable

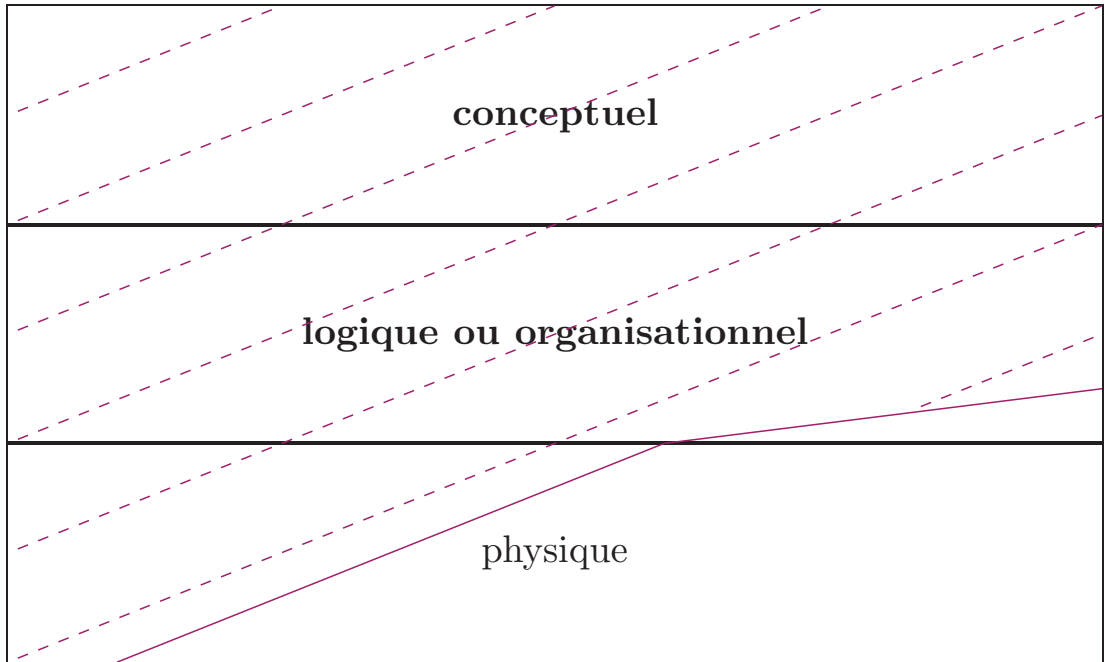


La part hachurée représente ce que couvre chaque étape. L'étude préalable couvre presque tout le niveau conceptuel, mais aussi un peu de niveau logique et physique.

Analyse Informatique : étude détaillée

- choix d'une organisation
- validation MCD, MCT
- proposition d'un MLD et d'un MOT
- optimisation, solutions dégradées
- \implies affectation des tâches de réalisation

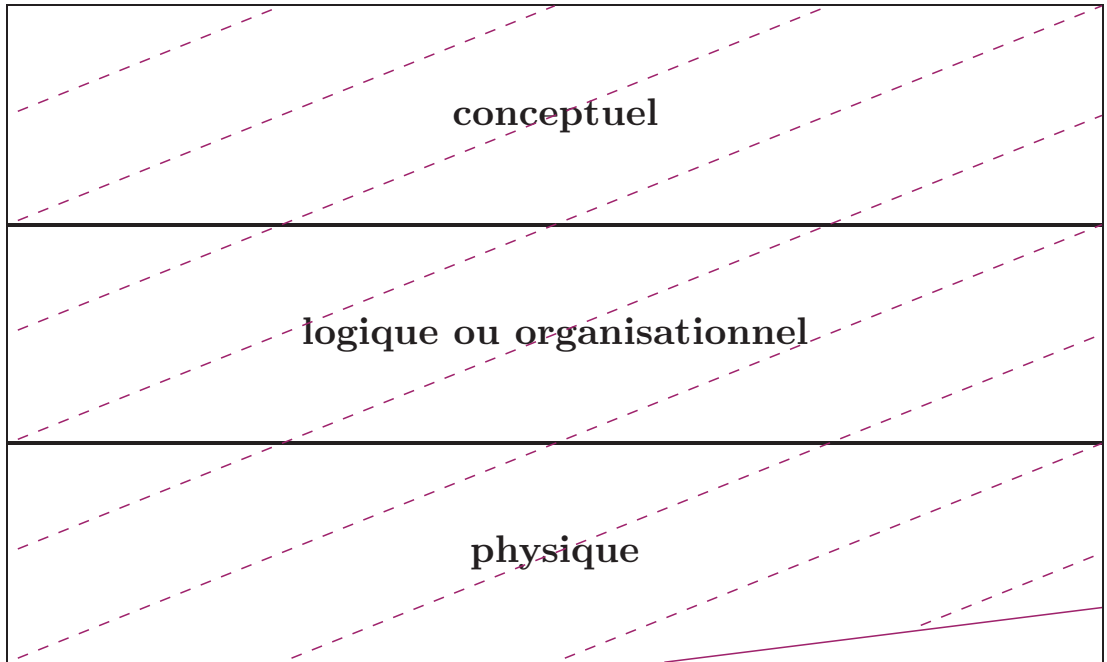
Analyse Informatique : étude détaillée



Analyse Informatique : réalisation

- étude technique et production logicielle
- mise en place et tests
- documentation, formation

Analyse Informatique : réalisation



3 Les dépendances fonctionnelles

3.1 Des données aux dépendances fonctionnelles

Pour être traitées de manière informatisée, les données doivent être décrites dans un formalisme compris par le système informatique qui va les gérer. Voici les formats génériques utilisés :

- ▷ Le type alphabétique (rien que des caractères).
- ▷ Le type alphanumérique (des caractères, des chiffres...).
- ▷ Le type numérique (les nombres).
- ▷ Le type date.
- ▷ Le type logique (0-1, Vrai-Faux, Oui-Non).

Suite à l'interview et la collecte des documents il est nécessaire de centraliser toutes les informations et règles de gestions (calcul d'un taux de remise par exemple) au sein d'un document. Ce document se nomme le dictionnaire des données.

3.1.1 Le dictionnaire des données

Le dictionnaire des données est un document qui permet de recenser, de classer et de trier toutes les informations (les données) collectées lors des entretiens ou de l'étude des documents. Le dictionnaire peut être plus ou moins élaboré selon le niveau de granularité souhaité.

Exemple :

Nom de la donnée	Format	Longueur	Type		Règle de calcul	Règle de gestion	Document
			Élémentaire	Calculé			

Nom de la donnée

Cette cellule recevra une donnée par exemple : Nom client.

Format

Ici sera indiqué le format de la donnée, par exemple : alphabétique.

Longueur

La longueur approximative ou exacte de la donnée sera indiquée, par exemple : 30.

Type

Une croix sera inscrite dans la colonne pour indiquer si la donnée est élémentaire ou calculée.

Exemple :

Nom de la donnée	Format	Longueur	Type		Règle de calcul	Règle de gestion	Document
			Élémentaire	Calculé			

Règle de calcul

Ici sera indiquée de manière claire la formule ou le calcul nécessaire à appliquer pour obtenir la donnée.

Règle de gestion

Dans cette zone sera indiquée, si nécessaire, la règle de gestion inhérente à la donnée.

Document

La rubrique document permet de saisir le document dans lequel a été trouvée la donnée.

Voici ce que pourrait être le dictionnaire :

Nom de la donnée	Format	Longueur	Type		Règle de calcul	Règle de gestion	Document
			E	C			
Nom client	Alphabétique	30	X				Facture

Le nom est au format alphabétique, d'une longueur de 30 caractères, de type élémentaire, il n'y a aucune règle de gestion et le document dans lequel l'information a été trouvée est la facture.

Longueur des champs

La longueur du champ nom a été définie aléatoirement à 30 caractères. Il faut toujours avoir à l'esprit que dans le doute il vaut mieux surdimensionner les tailles. Le proverbe qui s'adapte à la situation est « Qui peut le plus peut le moins ».

Que pensez-vous de cette règle ?

Exemple

Suite à une demande d'un membre de notre famille, président d'une association, nous devons établir le dictionnaire des données de la gestion des adhérents.

Voici une représentation d'une fiche d'adhérent :



Association des Palanges

Fiche Adhérent

Numéro	66
Nom :	BAPTISTE
Prénom :	Jean-Luc
Adresse :	Rue de la forêt
Code Postal :	12000
Ville :	Rodez
Téléphone :	05-65-42-00-00
Mail :	jeanluc.baptiste@btsig.org
Date d'adhésion :	20 décembre 2007

À la lecture de la fiche, nous pouvons déterminer la présence de neuf informations différentes :

- ▷ Le numéro de l'adhérent.
- ▷ Le nom.
- ▷ Le prénom.
- ▷ L'adresse.
- ▷ Le code postal.
- ▷ La ville.
- ▷ Le téléphone.
- ▷ Le mail.
- ▷ La date d'adhésion.

Voici le dictionnaire des données :

Nom	Format	Longueur	Type		Règle de calcul	Règle de gestion	Document
			E	C			
Numéro	Numérique		X				Fiche
Nom	Alphabétique	30	X				//
Prénom	Alphabétique	30	X				//
Adresse	Alphabétique	50	X				//
Code Postal	Alphanumérique	10	X				//
Ville	Alphabétique	50	X				//
Téléphone	Alphanumérique	15	X				//
Mail	Alphanumérique	50	X				//
Date d'adhésion	Date		X				//

Concernant le choix des types de données

Le code postal est alphanumérique et de taille 10. Certaines personnes peuvent considérer qu'il serait plus judicieux de placer le format en numérique.

Or qu'est-ce qui prouve que dans certains pays la règle d'écriture des code postaux est identique à la règle française des 5 chiffres ? En effet, certains pays mélangent des chiffres et des lettres. Le format alphanumérique est le plus approprié dans ce cas-là. De manière générale, il est souhaitable de ne formater en numérique que les champs sur lesquels il va y avoir des calculs. Le raisonnement appliqué est le même pour le champ téléphone.

Que pensez-vous de cette remarque ?

3.2 Les dépendances fonctionnelles

Le rôle de l'établissement des dépendances fonctionnelles est de nous aider à comprendre les liens existants entre chaque donnée. Cette démarche de recherche des dépendances fonctionnelles est la pierre angulaire de toute l'analyse des données. En effet, cette activité étant la première dans l'élaboration de l'analyse, si elle est négligée c'est tout l'ensemble qui en subira les conséquences.

Définition

Une donnée B dépend fonctionnellement (ou est en dépendance fonctionnelle) d'une donnée A lorsque la connaissance de la valeur de la donnée A nous permet la connaissance d'une et au maximum une seule valeur de la donnée B.

Exemple : La connaissance de la valeur d'un numéro de client nous permet de connaître sans ambiguïté la valeur d'un et d'un seul nom de client.

Dans la fiche d'adhérent, l'adhérent numéro 1 a pour nom Baptiste.

Formalisme

Le formalisme de représentation d'une dépendance fonctionnelle est le suivant :

Numéro adhérent → (Nom adhérent, prénom, adresse, code postal, ville, téléphone, mail, date d'adhésion)

Clé primaire

Numéro adhérent sera appelé la clé de la relation ou **clé primaire** ou encore **identifiant** de la relation.

La partie gauche de la dépendance fonctionnelle (ici Numéro adhérent) est aussi appelée source de la dépendance fonctionnelle. La partie droite de la dépendance fonctionnelle est appelée le but de la dépendance fonctionnelle.

3.2.1 Dépendances fonctionnelles composées

Une dépendance fonctionnelle qui comporte plusieurs attributs est dite composée.

Voici un exemple de dépendance fonctionnelle composée :

(Numéro Coureur, Numéro course) \rightarrow (temps)

Interprétation

Connaissant le numéro du coureur et le numéro de la course, nous connaissons de façon certaine le temps chronométré d'un coureur précis sur une course précise.

Autre exemple : (Code athlète, code sport) \rightarrow (année de pratique)

Interprétation

Connaissant le code de l'athlète et le code du sport nous pouvons connaître de façon sûre et unique le nombre d'années de pratique.

Comme nous pouvons le constater la seule connaissance du code d'athlète ne nous permet pas de connaître le nombre d'années de pratique, de la même manière la seule connaissance du code du sport ne permet pas la connaissance pleine et entière des années de pratique.

Structurellement, il est nécessaire d'avoir les deux informations : le code de l'athlète et le code du sport, pour pouvoir connaître les années de pratique d'un sport précis par un athlète précis.

3.2.2 Dépendance fonctionnelle élémentaire

Une dépendance fonctionnelle $A \rightarrow B$ est élémentaire

s'il n'existe pas une donnée C , sous-ensemble de A ,

décrivant une dépendance fonctionnelle de type $C \rightarrow B$

Par exemple :

- ▷ RéférenceProduit \rightarrow Désignation
- ▷ NuméroCommande, RéférenceProduit \rightarrow Quantité
- ▷ NuméroCommande, RéférenceProduit \rightarrow Désignation

La première dépendance fonctionnelle est correcte car ayant deux rubriques elle est élémentaire.

La deuxième dépendance fonctionnelle est correcte également car la connaissance d'un numéro de commande et d'une référence produit nous permet de connaître la quantité commandé du produit. Elle est aussi élémentaire car c'est la connaissance du couple (NuméroCommande, RéférenceProduit) et pas seulement d'un des éléments qui permet la connaissance de la quantité.

La troisième dépendance fonctionnelle n'est pas élémentaire car il existe à l'intérieur d'elle
RéférenceProduit → Désignation qui était déjà une dépendance fonctionnelle élémentaire. Pour
connaître la Désignation, NuméroCommande est dans ce cas superflu.

3.2.3 Dépendance fonctionnelle élémentaire directe

On dit que la dépendance fonctionnelle $A \rightarrow B$ est directe

s'il n'existe aucun attribut C tel que l'on puisse avoir $A \rightarrow C$ et $C \rightarrow B$.

En d'autres termes, cela signifie que la dépendance fonctionnelle entre A et B ne peut pas être obtenue par transitivité.

Exemple :

- ▷ NumClasse \rightarrow NumElève
- ▷ NumEleve \rightarrow NomElève
- ▷ NumClasse \rightarrow NomElève

La troisième dépendance fonctionnelle n'est pas directe car nous pourrions écrire :

NumClasse \rightarrow NumElève \rightarrow NomElève

3.2.4 Méthodologie d'élaboration des dépendances fonctionnelles

L'élaboration des dépendances fonctionnelles est réalisée à l'aide du dictionnaire des données.

La démarche consiste à rechercher :

- ▷ les dépendances fonctionnelles formées par deux rubriques, élémentaires et directes ;
- ▷ les dépendances fonctionnelles composées.

3.3 Cas pratique : camping

Monique, sa fille Rachel et son gendre Marc gèrent un camping dans les Pyrénées orientales. Le camping est ouvert du 1er juin au 30 septembre. Ils disposent de cinquante emplacements sur un terrain d'une superficie totale de quarante hectares.

Ils sont équipés d'un logiciel spécialisé dans la réservation des emplacements qui fonctionne très bien mais qui ne permet pas de gérer les achats de l'épicerie ou du bar selon leurs règles de gestion. En effet, les vacanciers ne payent leurs achats qu'à la fin de leur séjour. Concrètement, les achats sont inscrits manuellement sur une fiche bristol créée pour chaque famille de vacanciers. À la fin du séjour, les cumuls sont réalisés et une facture manuelle concernant les achats est établie. Les propriétaires du camping souhaiteraient disposer d'un logiciel permettant d'automatiser la création de la facture grâce à la saisie journalière des achats.

Voici une représentation de la fiche bristol :



Camping de la source

Liste des Achats

Nom : BAPTISTE
Prénom : Jean-Luc
Adresse : Rue de la forêt
Code Postal : 12000
Ville : Rodez
Téléphone : 05-65-42-00-00

Date	Désignation	Qté	Prix	Total
14/7/08	Repas « Cargolade »	4	22	88
15/7/08	Café	1	1,20	1,20
15/7/08	Glace « Magnum »	2	2,10	4,20
16/7/08	Baguette	1	1,15	1,15

Total dû : 94,55

4 Modèles de données

4.1 Diagramme des flux

Etude préalable

Analyse de l'existant : interviews, étude des documents, MCD et MOT actuels, mise en évidence des règles de gestion^a,...

Diagramme des flux de documents.

Critique de l'existant.

Ebauche de solutions. Proposition du Modèle Conceptuel des Données et du Modèle Conceptuel des Traitements

Rédaction du cahier des charges.

^a les règles de gestion expliquent le fonctionnement de l'entreprise, les contraintes du problème, la méthode actuelle pour le traiter...

Enoncé pour un exemple de diagramme des flux

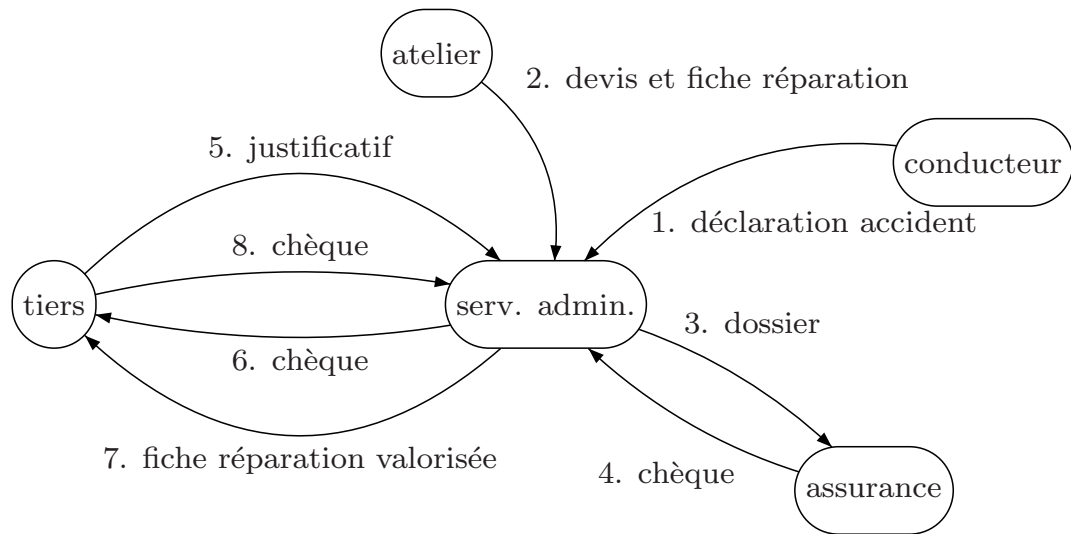
Une société de transport en commun... un accident...

Règles de gestion pour le traitement de l'accident :

- Le conducteur remplit une déclaration d'accident. Il conduit le bus à l'atelier qui établit un devis de réparation.
- Le service administratif définit la responsabilité selon la déclaration d'accident.
- Trois cas sont envisagés
 - prise en charge par les assurances : déclaration et devis sont transmis à l'assurance qui règle la note.
 - responsabilité du conducteur : la société indemnise l'accidenté sur présentation des justificatifs.
 - responsabilité du tiers : c'est lui qui paie l'addition...

Exemple de diagramme des flux

Circulation chronologique des *documents* entre les *intervenants*.



scénario “règlement par l’assurance” : 1,2,3,4

scénario “règlement par la société” : 1,2,5,6

scénario “règlement par le tiers” : 1,2,7,8

4.2 Modèle Conceptuel des Données (MCD)

Modèle conceptuel des données

Objectif : le MCD a pour but de modéliser les données (aspect statique) mémorisées dans le système d'information ;

Caractéristiques : Représentation graphique des données à un niveau conceptuel, c'est-à-dire, sans se préoccuper ni des contraintes d'organisation, ni du gestionnaire de bases de données utilisé, ni des traitements ;

MCD Merise : correspond au **modèle Entité - Association**.

Modèle conceptuel des données

Construction d'un MCD

Quand : dans l'étude préalable : MCD de l'existant et ébauche du MCD de la nouvelle solution ; dans l'étude détaillée : MCD complet de la nouvelle solution.

Préalable : avoir explicité les règles de gestion, avoir établi un diagramme des flux, avoir construit un dictionnaire des données

Modèle conceptuel des données

les objets ou entités

- Une **entité** est la représentation d'un objet matériel ou immatériel pourvu d'une existence propre et conforme aux choix de gestion de l'entreprise.
- Une **occurrence d'une entité** est une instance (un représentant) de l'entité dans le *monde réel*.
- Une **propriété (ou attribut)** est une donnée élémentaire qu'on perçoit sur l'entité.
- L'**identifiant d'une entité** est une propriété (ou un ensemble de plusieurs propriétés) de l'entité telle que, à chaque valeur de l'identifiant, correspond *une et une seule* occurrence de l'entité. Il peut être créé ex nihilo.

MCD : un exemple

Les règles de gestion (simplifiées) :

- Les étudiants ont une identité (nom, prénom, adresse...).
- Les enseignants ont une identité.
- Les étudiants suivent des cours mais ne suivent pas tous les cours.
- Les cours sont hebdomadaires et ont une heure de début, une durée, une heure de fin ; les horaires ne doivent pas se superposer.
- Les cours sont assurés par un seul enseignant.

MCD : suite de l'exemple

Le dictionnaire des données :

- Nom étudiant, texte, **E**lémentaire, 30 caractères,
- Nom jeune fille, texte, **E**, 30 caractères, vide autorisé,
- Numéro étudiant, texte **?**, **E**, 17 caractères,
- ...
- Intitulé cours, texte, **E**, 70 caractères,
- Jour, numérique, **E**, entier, 1 pour lundi, 2 pour mardi...
- Heure Début, numérique, **E**, réel, $7 \leq hd \leq 20$
- Durée, numérique, **E**, réel, $0 \leq d \leq 5$
- Heure Fin, numérique, **C**alculé, réel, $hf = hd + d$
- ...

MCD : suite de l'exemple

ÉTUDIANTS

NoEtudiant

nom

nom jeune fille

prénom

adresse

ENSEIGNANTS

NoEnseignant

nom

prénom

adresse

COURS

NoCours

intitulé

jour

heure début

durée

Modèle conceptuel des données

Règles à vérifier sur les entités (à appliquer avec bon sens)

1. Toute propriété est élémentaire ;
2. Une propriété ne doit pas être “instable”, pas “calculable” ;
3. Toute propriété d’une instance aura au plus une valeur ;
4. Une propriété doit permettre d’éviter la redondance des valeurs sur l’ensemble des instances ;
5. Toute entité possède un identifiant ;
6. Toute propriété dépend (directement) de l’identifiant ;
7. Toute propriété dépend de *tout* l’identifiant ;

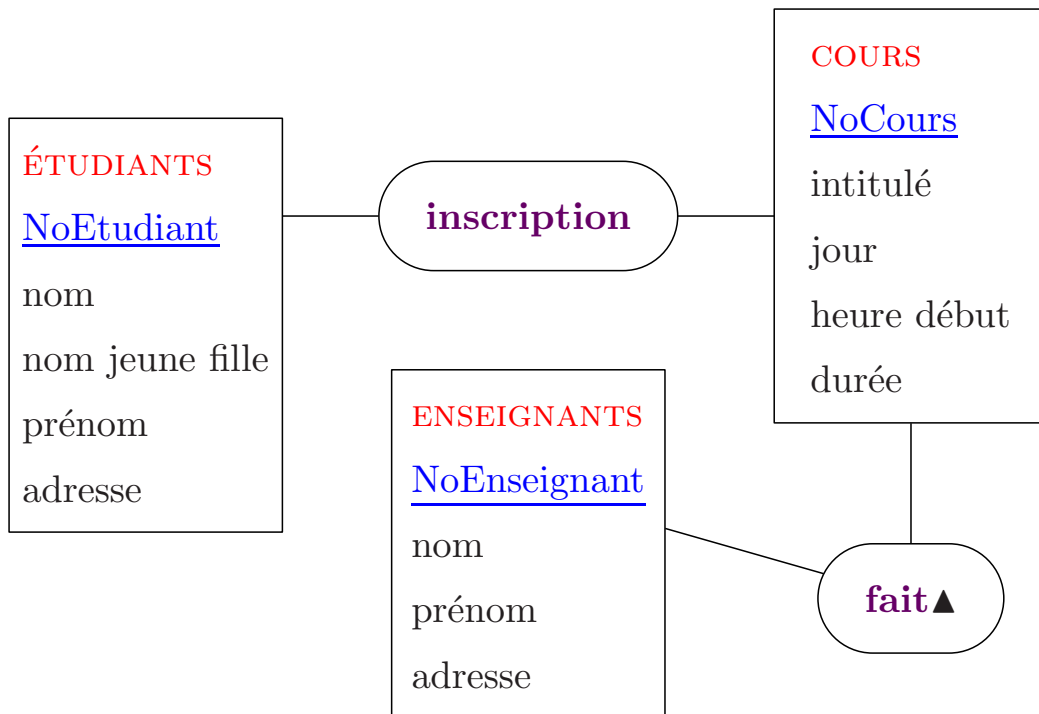
Modèle conceptuel des données les associations

- Une **association** est une relation que les règles de gestion établissent entre deux entités (ou plus).
- Une **occurrence d'une association** est une instance de l'association dans le *monde réel*.
- Une association peut posséder des propriétés.

Règles à vérifier sur les associations : règles des entités.

L'identifiant d'une association est la concaténation des identifiants des entités qu'elle relie.

MCD : suite de l'exemple



Modèle conceptuel des données les cardinalités

La **cardinalité d'une entité par rapport à une association** s'exprime sous forme d'un couple :

(cardinalité minimale : cardinalité maximale)

cardinalité minimale : c'est le nombre minimal de fois où une occurrence d'un objet participe aux occurrences de l'association ; elle vaut très souvent **0 ou 1**.

cardinalité maximale : c'est le nombre maximal de fois où une occurrence d'un objet participe aux occurrences de l'association ; elle vaut **1, un entier fixé ou n** ("n" pour indiquer l'impossibilité de fixer une borne maximale).

Modèle conceptuel des données les cardinalités

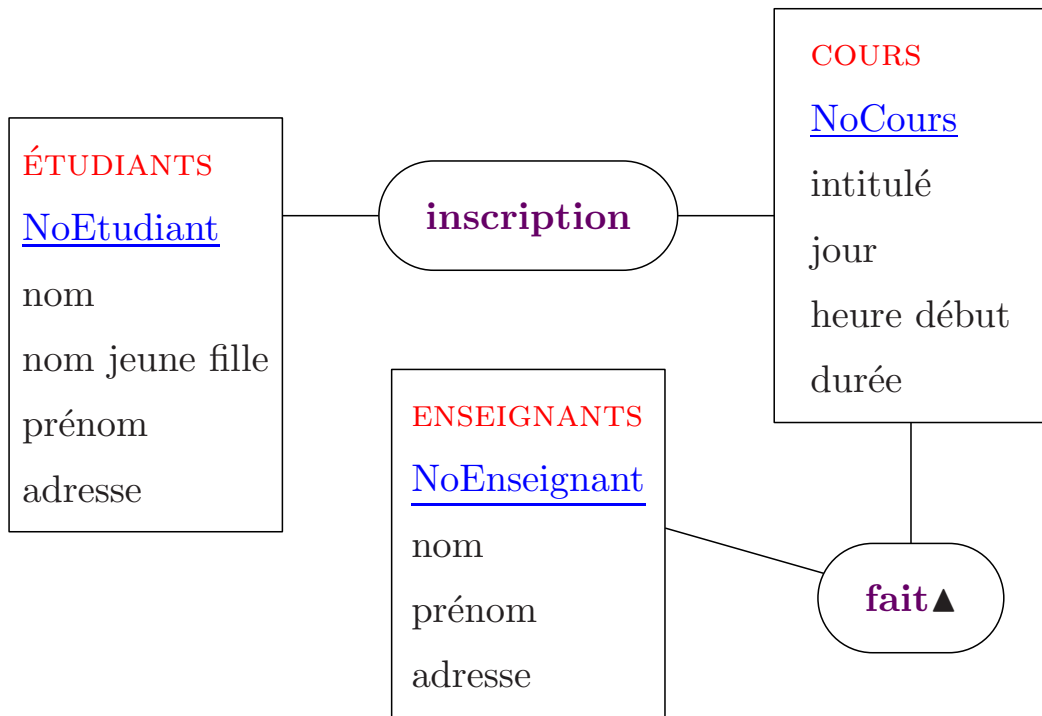


cmA : Vous l'obtenez en répondant à la question *“quand je considère un élément de l'entité A, combien d'éléments de l'entité B lui sont obligatoirement associés ?”*.

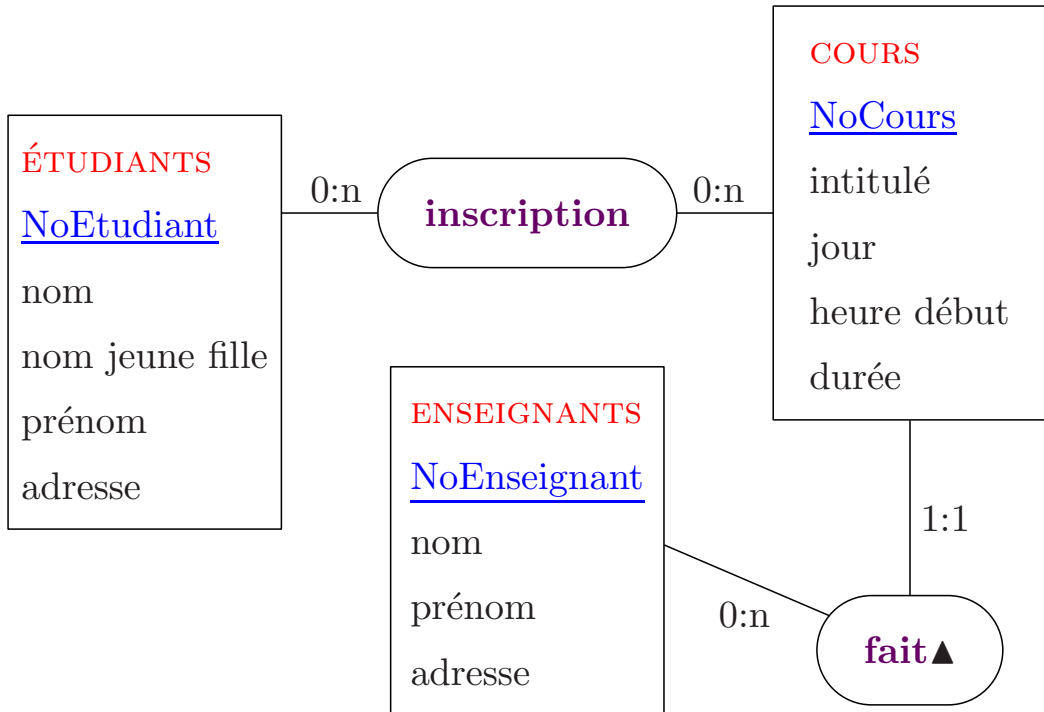
cMA : *“quand je considère un élément de l'entité A, combien d'éléments de l'entité B lui sont associés au maximum ?”*.

cmB et cMB : Symétriquement, *“quand je considère un élément de l'entité B, combien d'éléments de l'entité A lui sont associés, au minimum (cmB) et au maximum (cMB) ?”*.

MCD : suite de l'exemple



MCD : suite de l'exemple



Modèle conceptuel des données

Vérification d'un MCD

Vérifier :

- les règles sur les entités ;
- les règles sur les associations ;
- les règles globales :
 1. Une propriété ne figure qu'une fois dans le MCD ;
 2. Les propriétés calculées ne figurent pas dans le MCD (mais il faut s'assurer qu'avec le MCD on puisse les calculer) ;
 3. On ne fait pas figurer les associations qui se déduisent par transitivité.

MCD : un exercice (gestion de commandes)

Les règles de gestion :

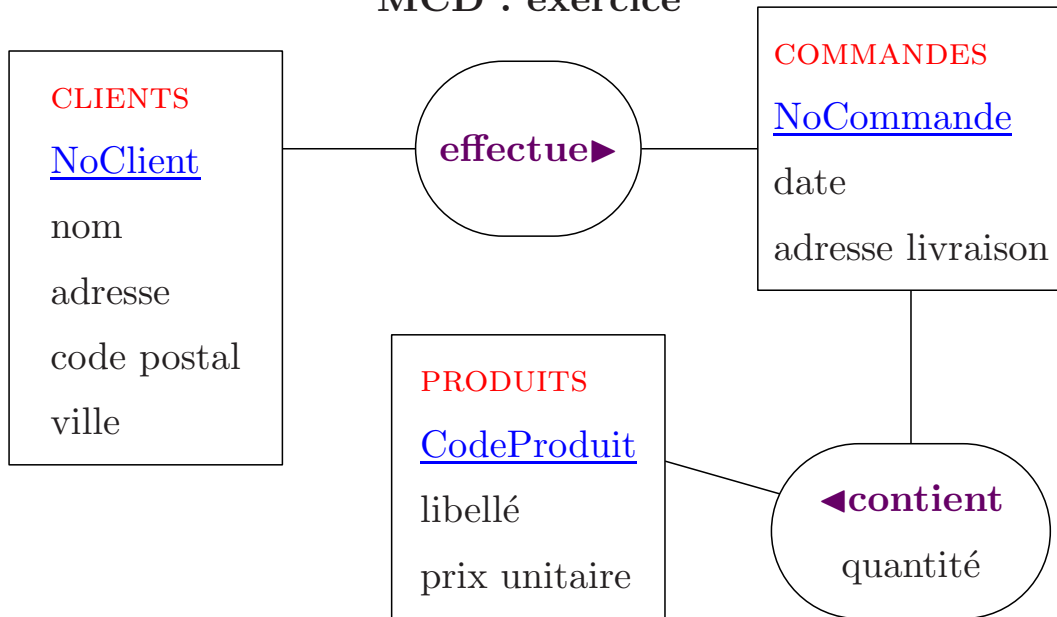
- Le magasin vend des produits à des clients.
- Les produits possèdent une référence (un code), un libellé et un prix unitaire.
- Les clients ont une identité (nom, prénom, adresse...).
- Les clients passent des commandes de produits. On mémorise la date de la commande.
- Pour chaque commande, le client précise une adresse de livraison.
- La commande concerne un certain nombre de produits, en une quantité spécifiée pour chaque produit.

MCD : suite de l'exercice

Le dictionnaire des données :

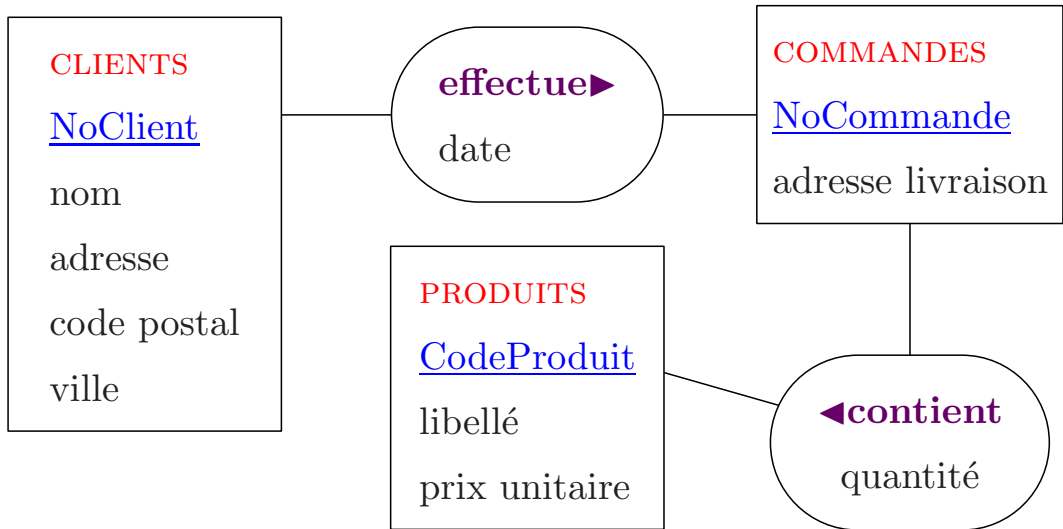
- Code Produit, texte, E, 5 caractères,
- Libellé, texte, E, 40 caractères,
- Prix, numérique, E, réel, $0 \leq \text{prix}$,
- Nom client, texte, E, 30 caractères,
- Adresse, texte, E, 40 caractères,
- ...
- Date Commande, numérique, E, date,
- Adresse livraison, texte, E, 50 caractères,
- Quantité commandée, numérique, E, entier, $0 \leq \text{quantité}$
- Nombre produits commandés, numérique, C, entier,
 $\text{nombre} = \text{compte}(\text{Produits commandés})$

MCD : exercice



MCD : exercice

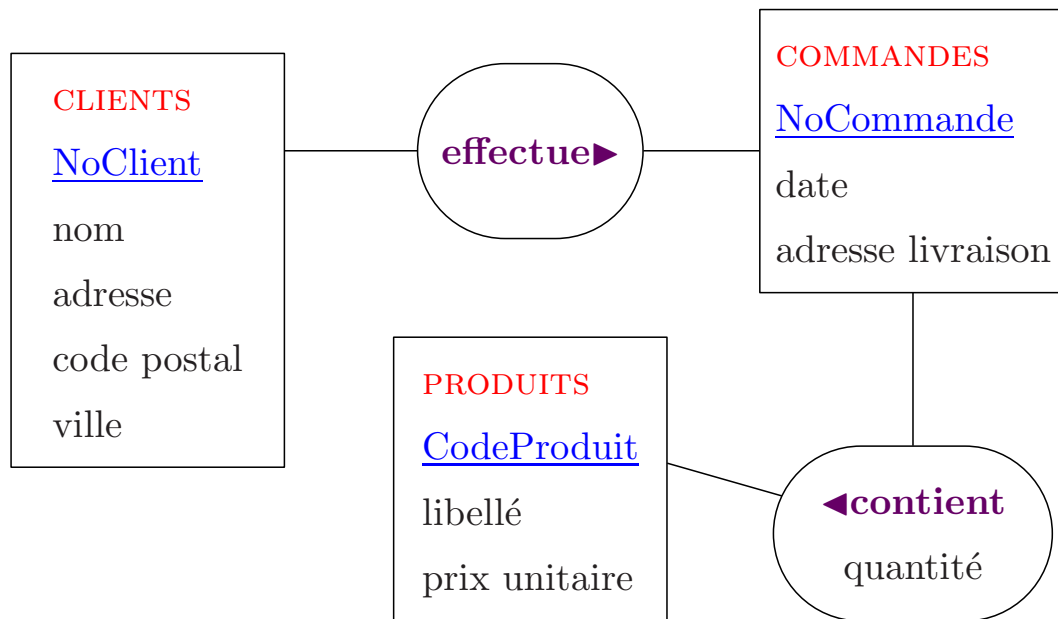
On aurait pu faire



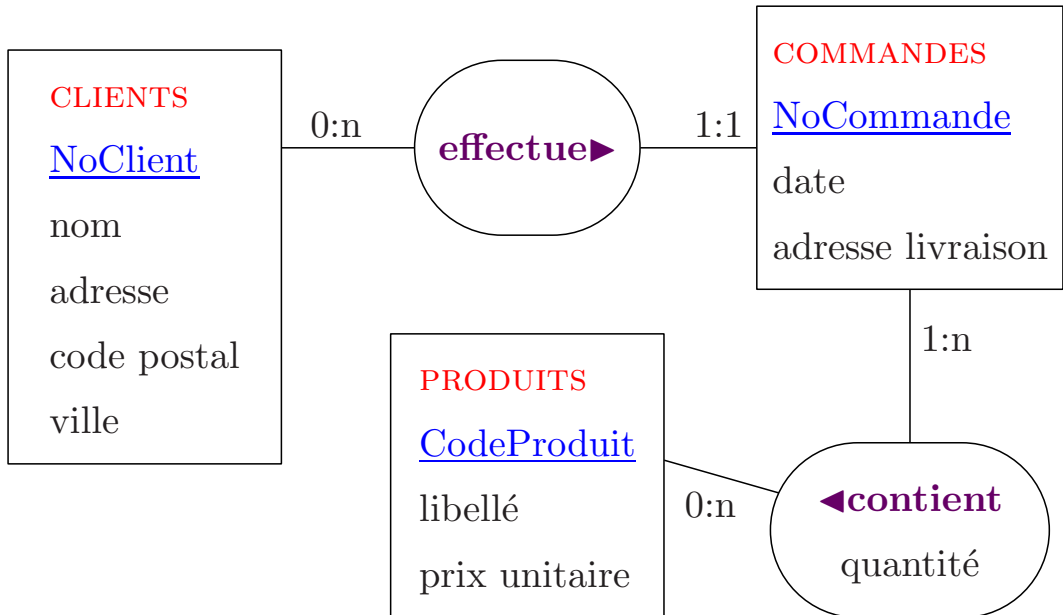
mais la règle *une propriété dépend de **tout** l'identifiant* n'est pas respectée.

Le NoCommande détermine seul la date.

MCD : exercice - cardinalités ?



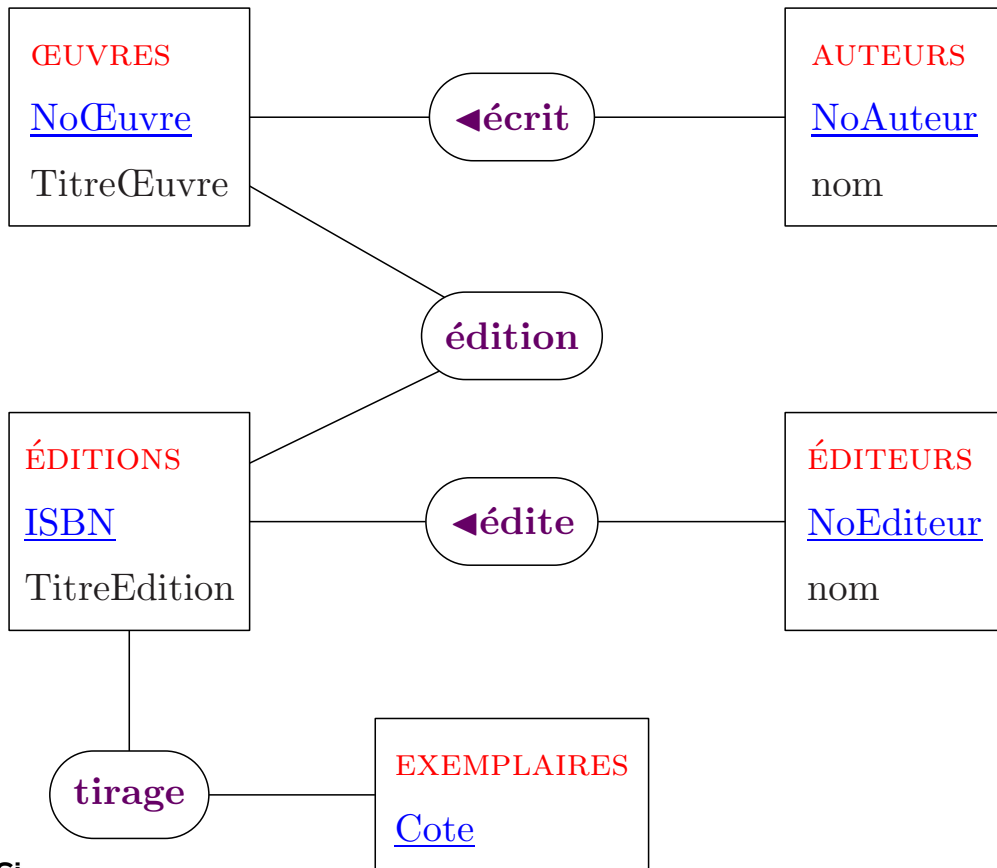
MCD : solution

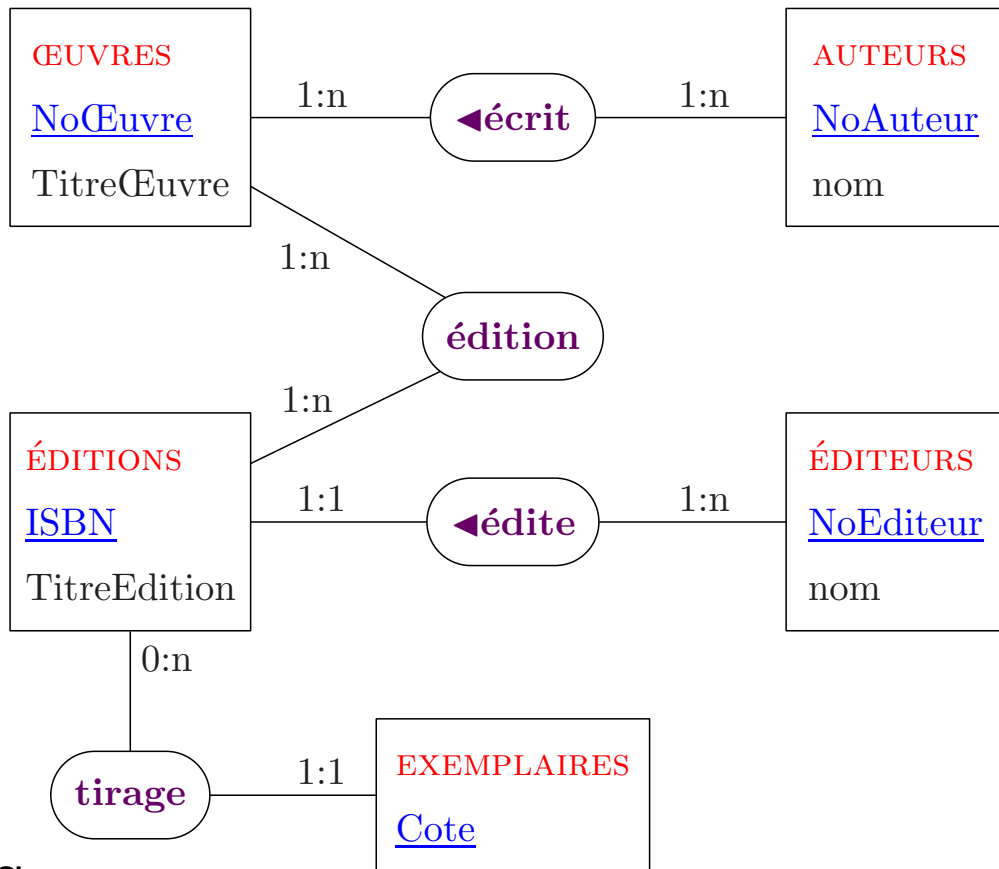


Modèle conceptuel des données - un exemple

Règles de gestion pour une bibliothèque

- On appellera œuvre une création littéraire.
- Une œuvre a un auteur au minimum. Elle peut faire l'objet de plusieurs éditions (première édition, réédition, livre de poche...).
- Une édition possède un ISBN unique. Elle peut contenir plusieurs œuvres (recueil, anthologie...). Elle est faite par un seul éditeur.
- Un livre peut exister en plusieurs exemplaires dans la bibliothèque. On ne mémorise pas d'informations sur des œuvres qu'on ne possède pas en bibliothèque ; mais, si un livre disparaît, on ne supprime pas ce qu'on avait déjà enregistré.





Modèle conceptuel des données - un exercice

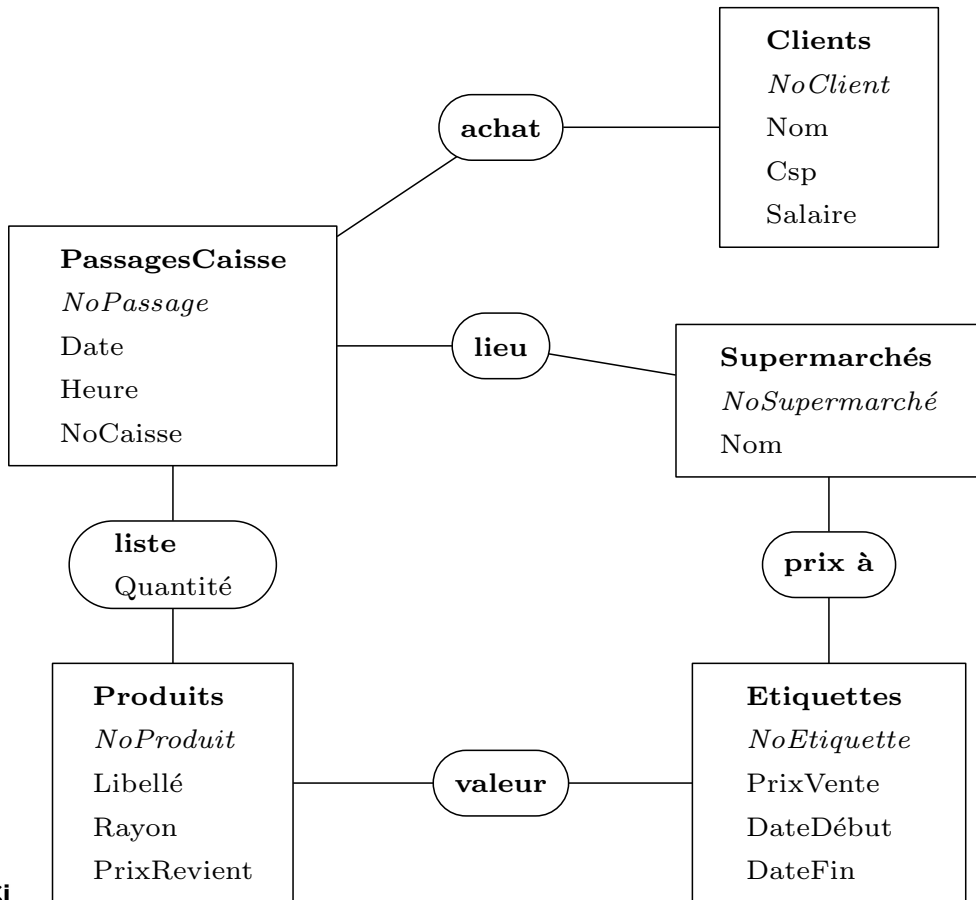
Gestion de réservations pour des spectacles de théâtre.

- Dans la région, il existe de nombreuses salles de spectacle qui ont un nom et se situe dans une ville.
- Pour gérer les réservations, on limitera les informations concernant un spectacle à son titre et au nom de compagnie.
- Un spectacle peut effectuer une tournée dans plusieurs salles. Mais à une date donnée, il ne peut avoir lieu que dans une seule salle. Pour nommer les entités, on appellera *soirée* un spectacle donné à une date donnée.
- Les spectateurs possèdent une identité et une adresse. On voudra savoir si les spectateurs vont principalement dans des salles de leur ville.
- Ils peuvent réserver plusieurs places pour une soirée donnée.

Modèle conceptuel des données - un exercice

Règles de gestion pour une chaîne de supermarchés (exam 03). On propose des cartes de fidélité et on mémorise toutes les ventes.

- Dans l'entité **Produits**, le prix de revient est identique pour tous les magasins de la chaîne ; le prix de vente dépend du magasin.
- Dans l'entité **Etiquettes**, on mémorise le prix de vente d'un produit dans un magasin, pour une période donnée ; le prix actuel se repère par une date de fin particulière : 31/12/2100.
- L'entité **PassagesCaisse** représente le passage d'un client à une caisse d'un supermarché.
- Certains achats ne seront pas associés à un client physique. On a prévu un client fictif (Monsieur Nobody) dont la fiche existe dans la table des clients et qui remplacera les clients dont on ne connaît pas l'identité. Toutes les ventes des magasins sont donc mémorisées.

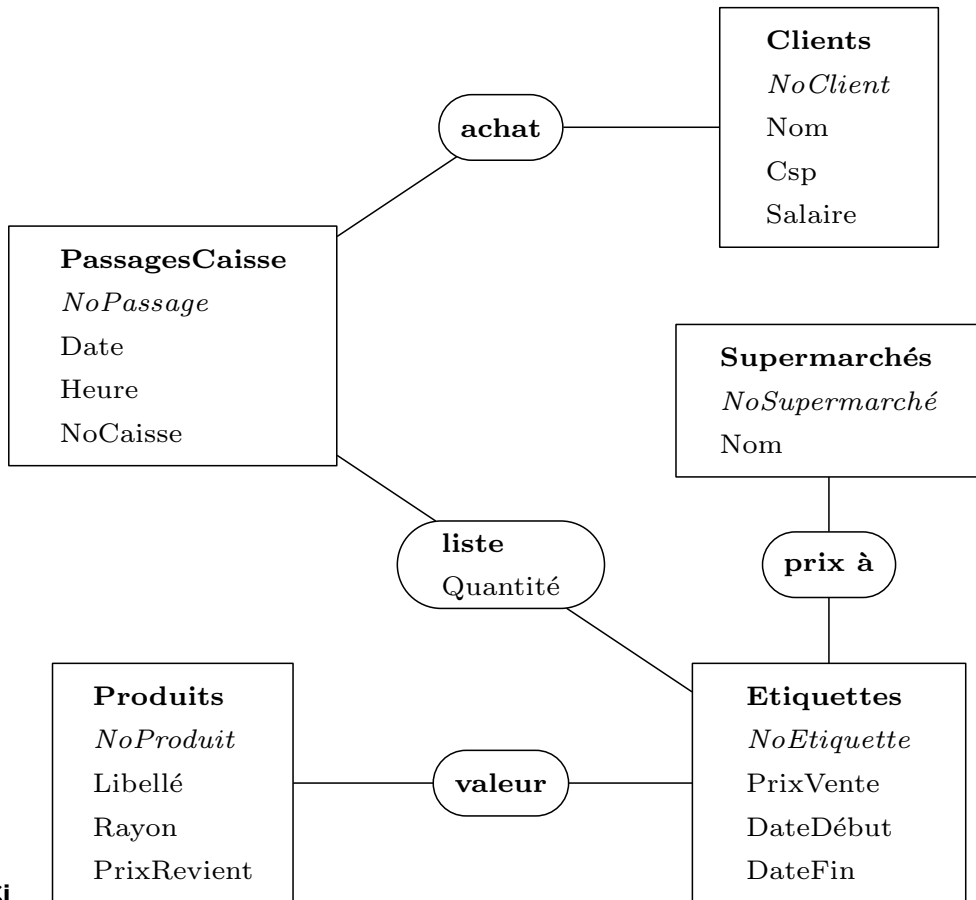


Modèle conceptuel des données - exercice

Moins naturelle, la solution suivante serait peut-être meilleure.

Au lieu d'associer, un ticket de caisse (**PassagesCaisse**) à une liste de produits, le ticket est directement associé aux étiquettes des produits achetés.

Une étiquette étant associée à un produit et à un magasin, on retrouve aisément la liste des produits achetés et le lieu.



Modèle conceptuel des données - un exercice

Règles de gestion supplémentaires.

- Les clients proviennent de toutes les villes avoisinantes ; le code postal pourra être pris comme identifiant. Le supermarché est situé dans une localité et on supposera qu'il est impossible qu'aucun client ne vienne de cette localité.
- Des hôtesses d'accueil sont employées aux caisses. Il y a forcément une hoteesse à la caisse lors du passage du client. Une hoteesse peut changer de caisse selon les besoins, selon l'affluence... Lorsqu'elle prend la responsabilité d'une caisse, elle entre son numéro d'identification et tous les passages à cette caisse lui sont attribués jusqu'à ce qu'elle quitte la caisse ; on ne crée pas d'entité pour les caisses. Une hoteesse travaille toujours dans le même supermarché.

Modèle conceptuel des données

Construction d'un MCD

FAQ : Frequently Asked Questions

- dois-je introduire un nouvel identifiant ?
- les propriétés étant élémentaires, quel niveau de décomposition atteindre ?
- faut-il multiplier les entités ?
- est-ce une entité ou une association ?
- cardinalité (0:n) ou (1:n) ?

FAQ : réponses

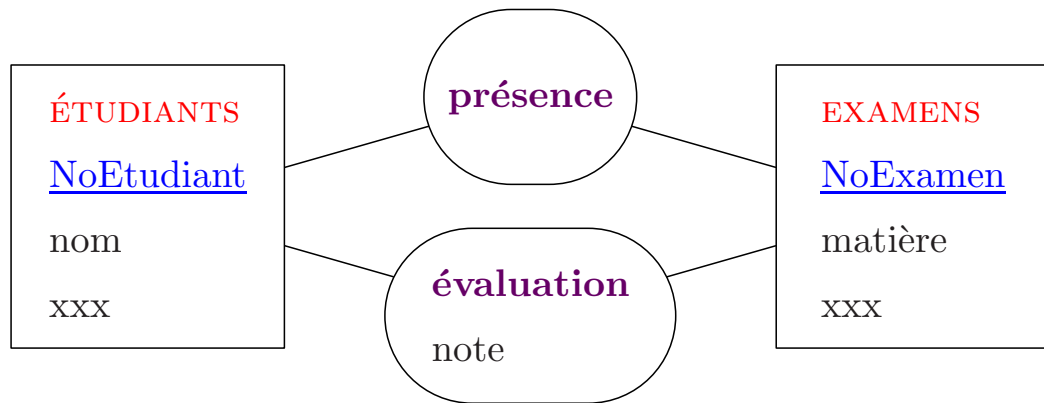
- Un identifiant est obligatoire.
- Le niveau de décomposition dépend des besoins en interrogation de la base.
- Même raisonnement ! Plus il y aura d'entités, plus le développement sera lourd... Ne créez pas d'entités inutiles. Mais vous devez vérifier les règles des entités.
- Si une association contient de nombreuses propriétés ou si elle relie trois entités (ou plus), envisagez une entité.
Si deux entités sont reliées par des cardinalités 1:1, regroupez-les en une seule.
- Répondre à la question *“à un élément de cette entité, combien puis-je associer, au minimum, d'éléments de l'autre entité ?”*. Sachez toutefois qu'elles seront traitées de la même façon lors du passage au MLD.

MCD : quelques exemples particuliers

- Une association peut être ternaire. Exemple : VENDEURS, LIEUX, ACHETEURS reliées par *vente*. Posez-vous alors la question “*et si je remplaçais l’association par une entité, ne serait-ce pas plus clair ?*” ... une entité ACTESDEVENTE.
- Une association peut être réflexive. Exemple : l’entité INDIVIDUS reliée avec elle-même par l’association *filiation*.
- Deux entités peuvent être reliées par plusieurs associations. Exemple : TRAJETSSNCF reliée à GARES par les associations *GareDépart*, *GareArrivée*, *Etapas*
- Les cardinalités, bien qu’étant très souvent 0:1, 1:1, 0:n ou 1:n, peuvent avoir d’autres valeurs. Exemple du tiercé :

PARIEURS $\overset{1:n}{\text{---}} \bigcirc \overset{1:1}{\text{---}} \text{TICKETS} \overset{3:3}{\text{---}} \bigcirc \overset{0:n}{\text{---}} \text{CHEVAUXPARTANTS}$

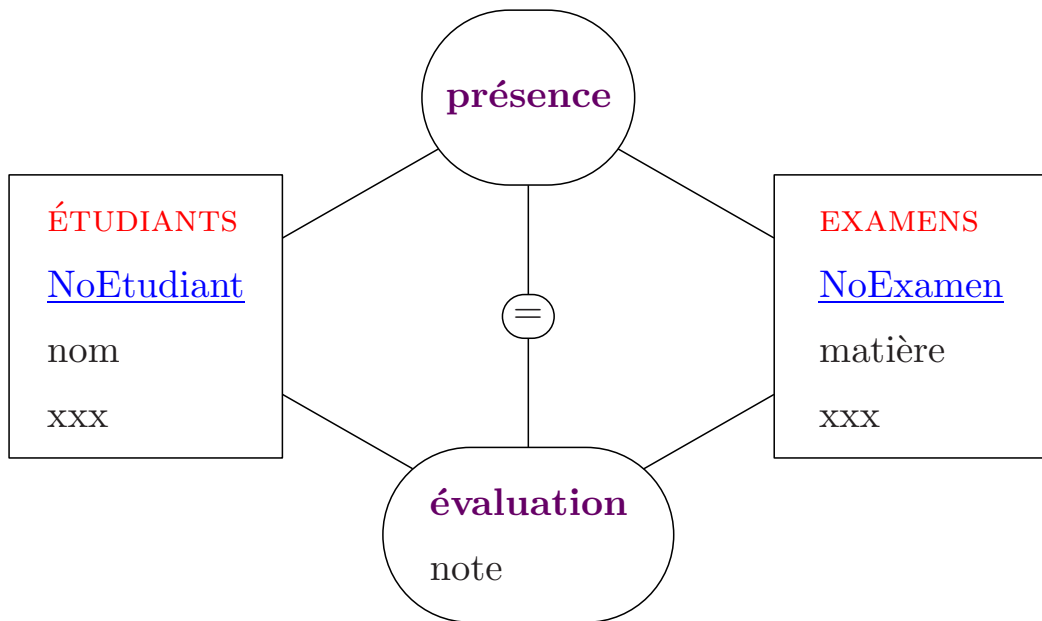
Extensions du modèle conceptuel des données



L'association *présence* représente la liste des présents au moment de l'examen et l'association *évaluation* représente le relevé des notes fourni par l'enseignant, après correction.

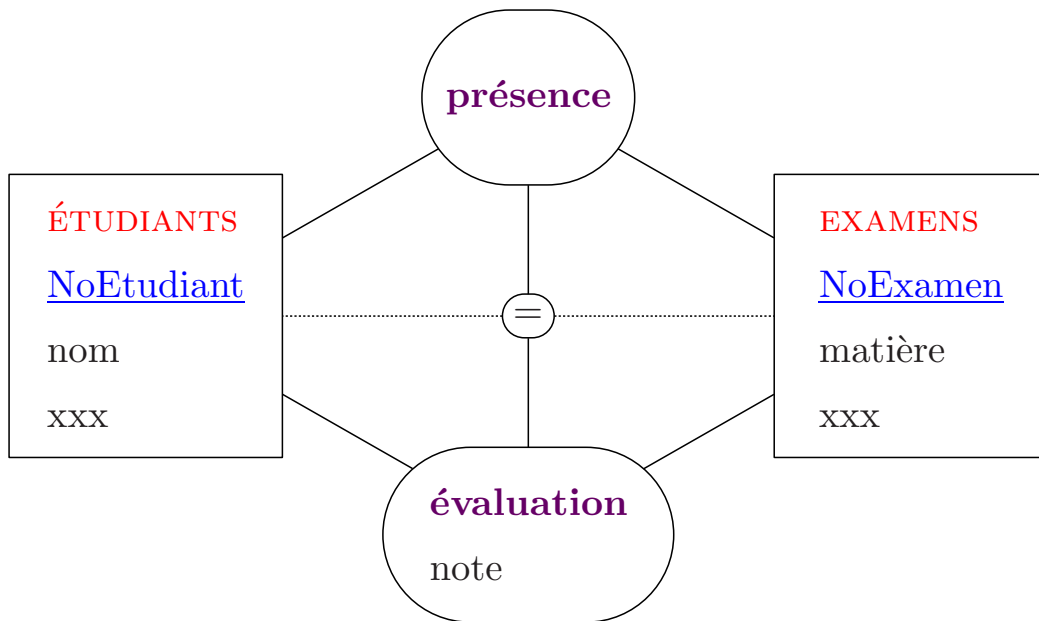
Pourrait-on indiquer qu'un étudiant présent doit avoir une note ?

Extensions du modèle conceptuel des données

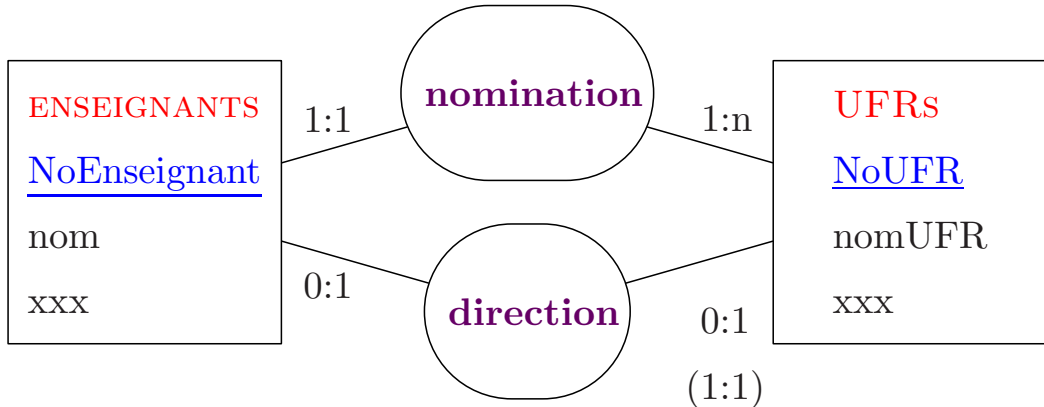


Extensions du modèle conceptuel des données

notation standard

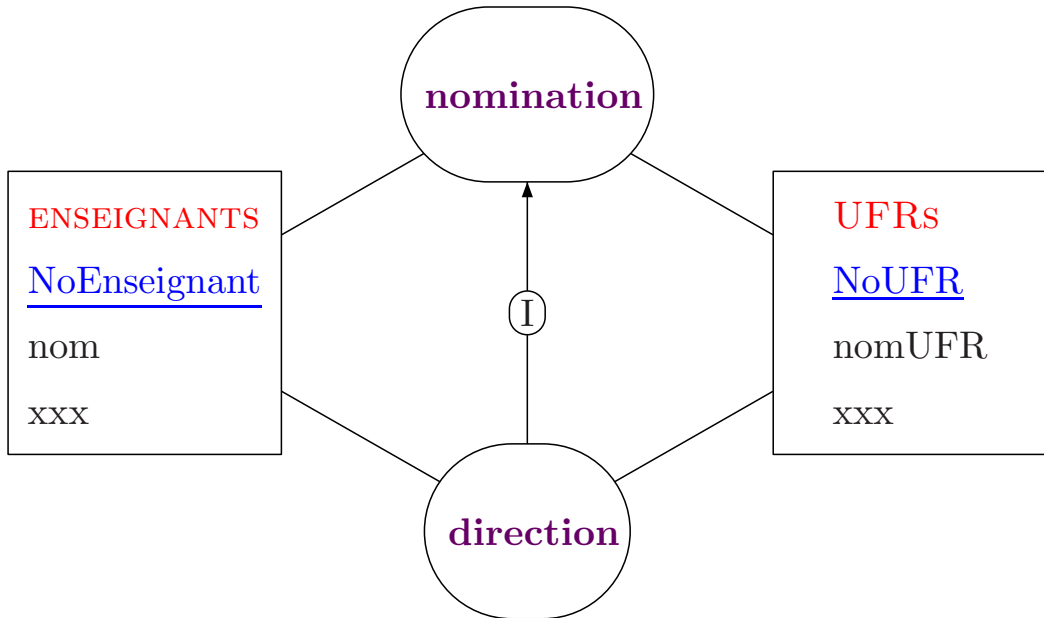


Extensions du modèle conceptuel des données



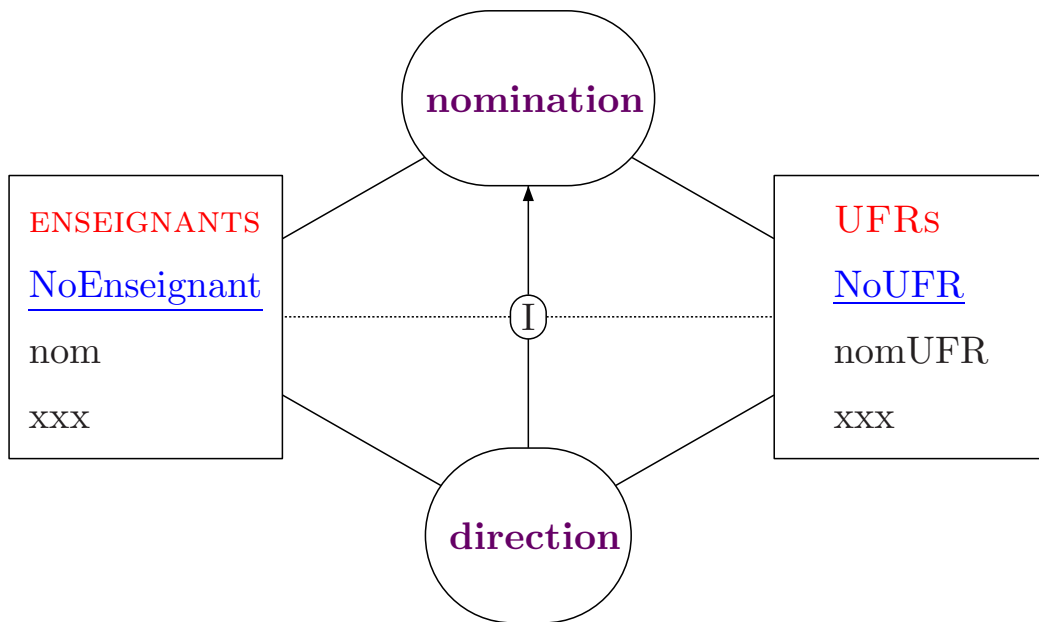
Pourrait-on indiquer que le directeur doit être un enseignant nommé dans l'UFR ?

Extensions du modèle conceptuel des données

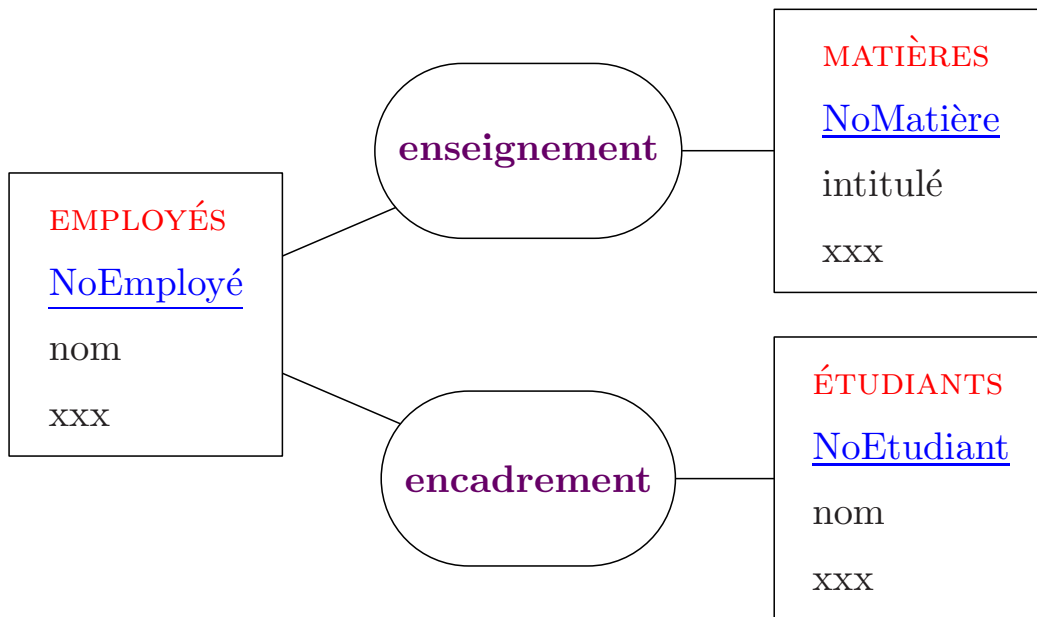


Extensions du modèle conceptuel des données

notation standard

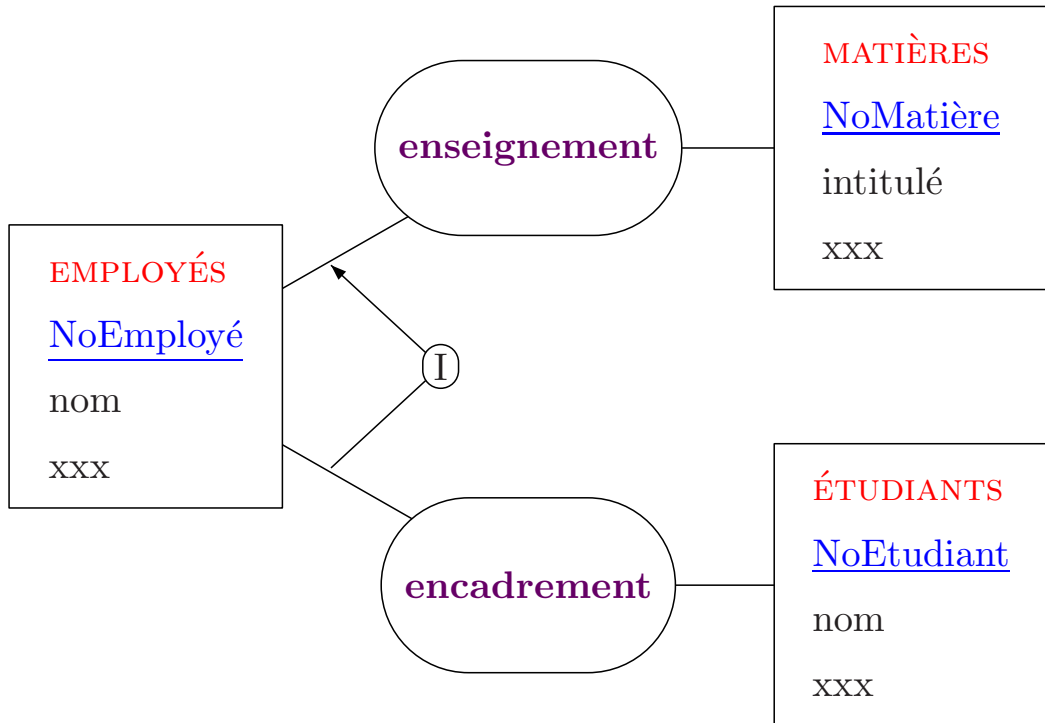


Extensions du modèle conceptuel des données



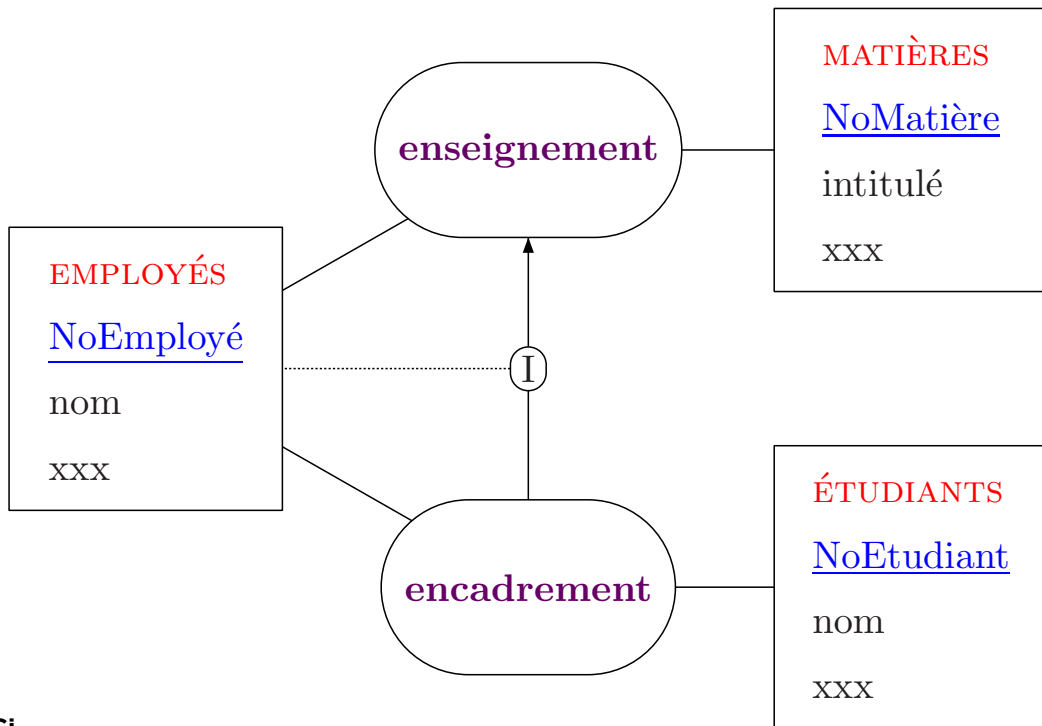
Pourrait-on indiquer qu'un encadrant doit être un enseignant ?

Extensions du modèle conceptuel des données

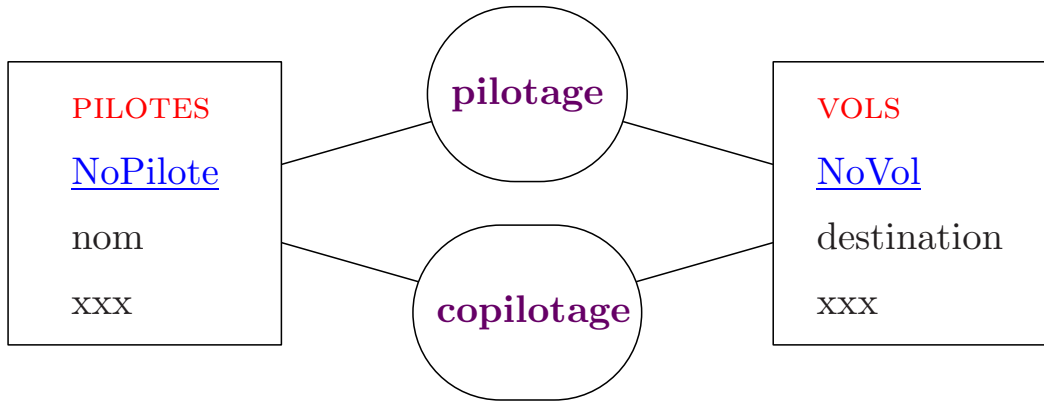


Extensions du modèle conceptuel des données

notation standard

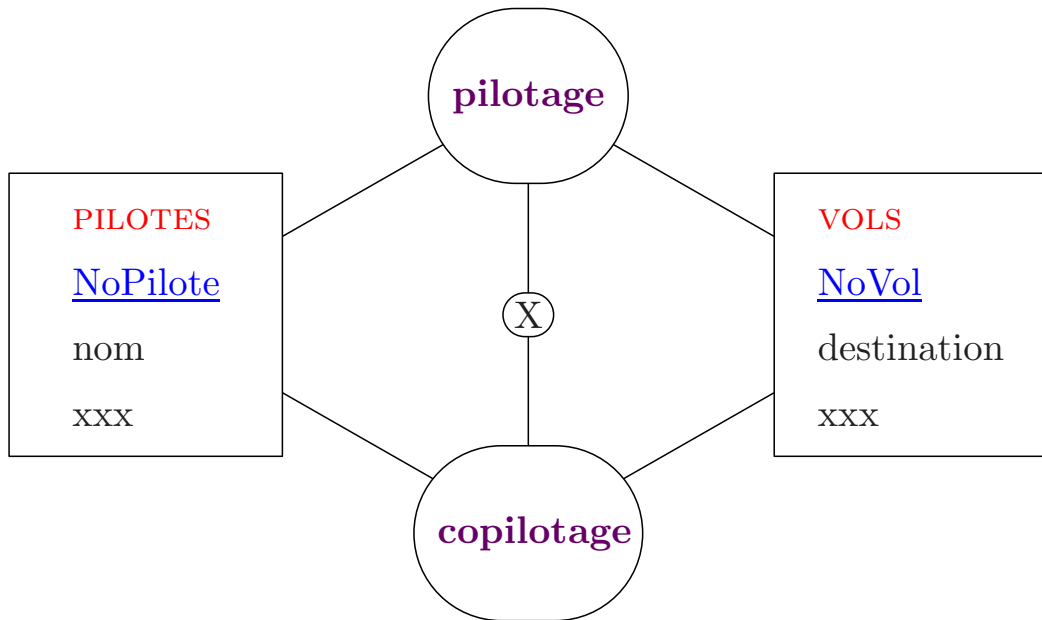


Extensions du modèle conceptuel des données



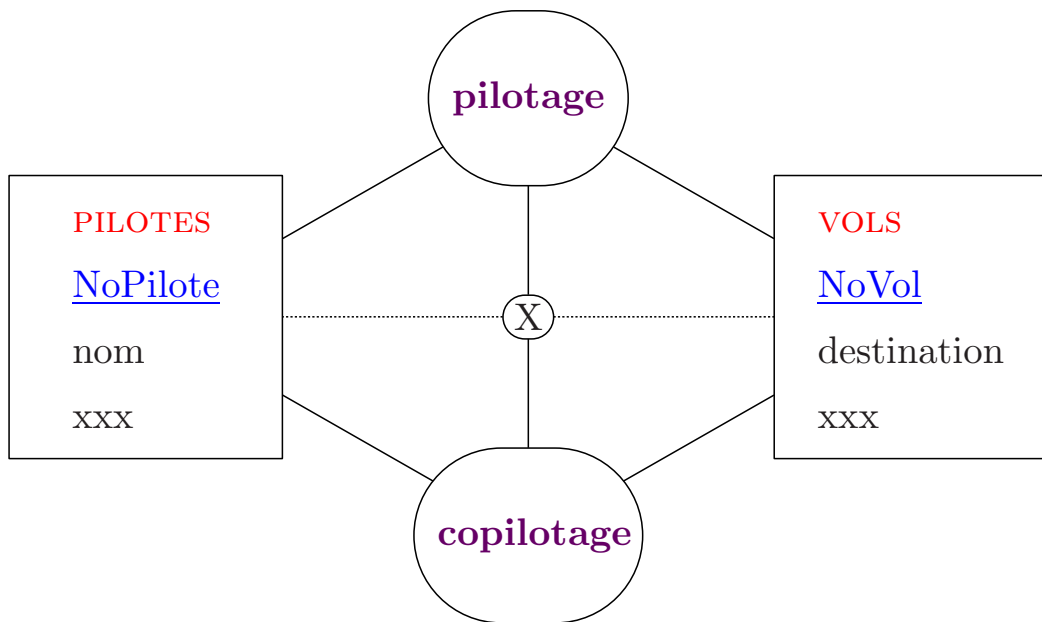
Pourrait-on indiquer que le pilote d'un vol est obligatoirement différent du copilote ?

Extensions du modèle conceptuel des données

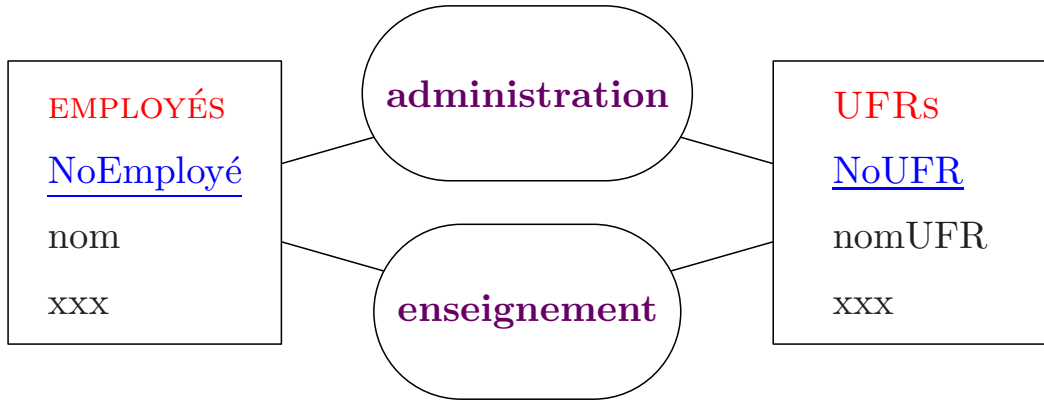


Extensions du modèle conceptuel des données

notation standard

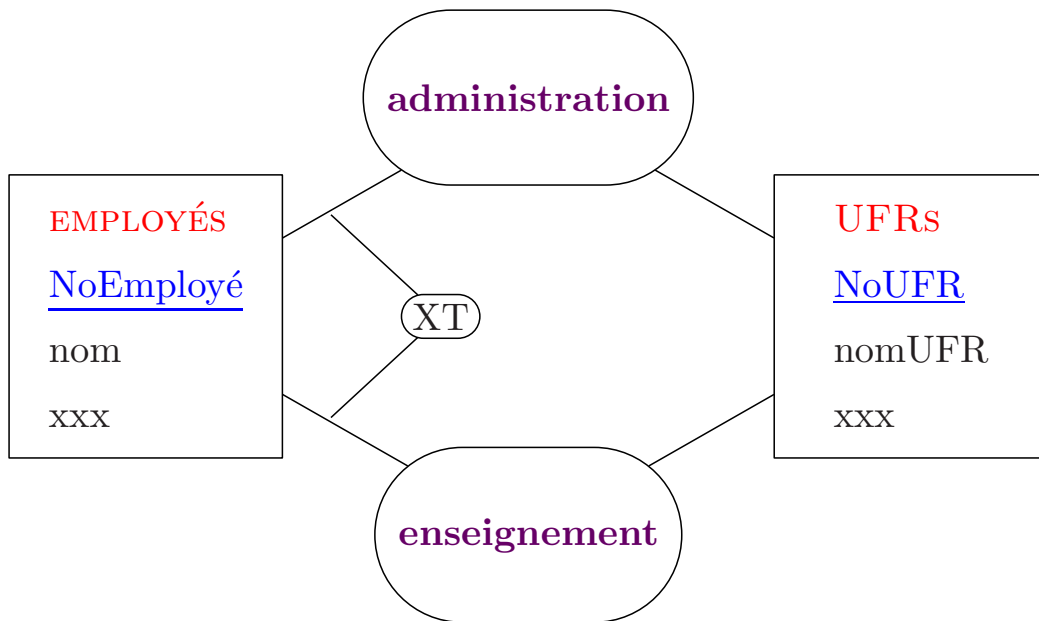


Extensions du modèle conceptuel des données



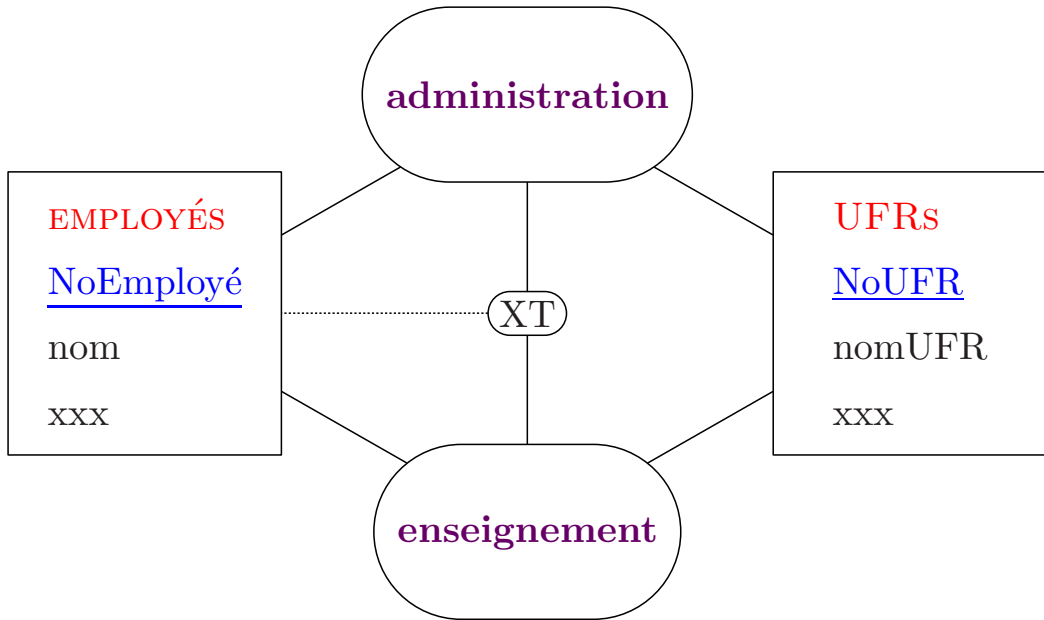
Pourrait-on indiquer qu'un employé de l'université est soit un enseignant, soit un administratif (et forcément l'un ou l'autre) ?

Extensions du modèle conceptuel des données



Extensions du modèle conceptuel des données

notation standard



Extensions du modèle conceptuel des données

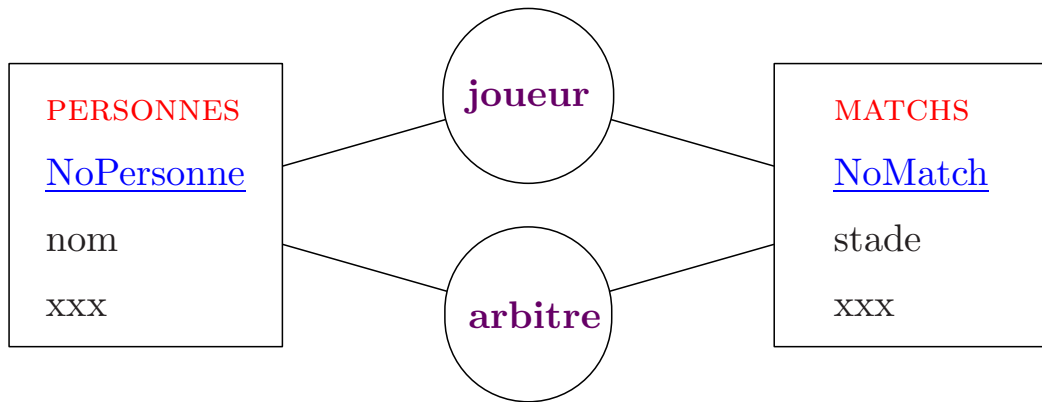
On peut exprimer

- l'égalité ($=$), $E_1 = E_2$
- l'inclusion (I), *orienté* $E_1 \subseteq E_2$
- l'exclusion (X), $E_1 \cap E_2 = \emptyset$
- la totalité (T), $E_1 \cup E_2 = E$
- l'exclusion et totalité. $E_1 \cap E_2 = \emptyset$ **et** $E_1 \cup E_2 = E$

sur les associations ou sur les arcs de l'association

Il est aussi possible de composer des associations pour indiquer une contrainte sur la composition.

Exercice



Comment indiquer que l'arbitre d'un match n'est pas un joueur de ce match, et inversement ?

Comment indiquer qu'aucun arbitre n'est un joueur, et inversement ?

Comment indiquer que tout arbitre a joué au moins un match ?

Comment indiquer que toute personne est soit arbitre, soit joueur ?

Contraintes d'intégrité fonctionnelle

Il y a contrainte d'intégrité fonctionnelle quand la connaissance d'une occurrence dans une (ou plusieurs) entité(s) détermine la connaissance de l'occurrence associée dans l'autre entité.

Cela ressemble à des cardinalités 1:1 sur une association qui serait intitulée *détermine...*

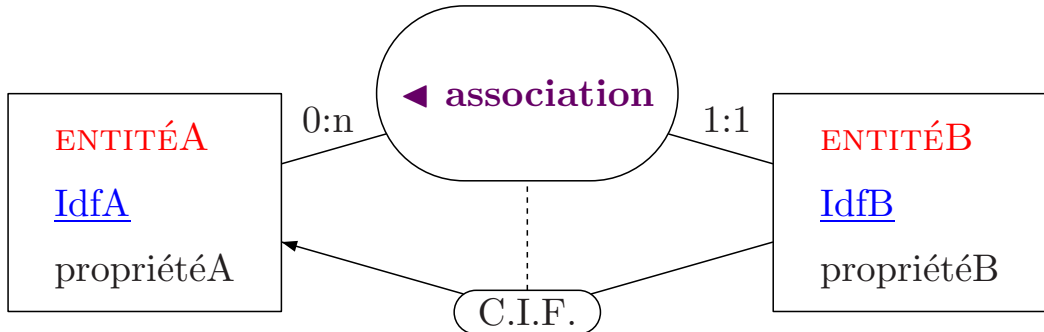
Inversement, si une association possède des cardinalités 1:1, il y a C.I.F.

Indiquer une C.I.F. apporte peu d'information dans le cas des associations binaires.

Contraintes d'intégrité fonctionnelle



pourrait être notée



Contraintes d'intégrité fonctionnelle

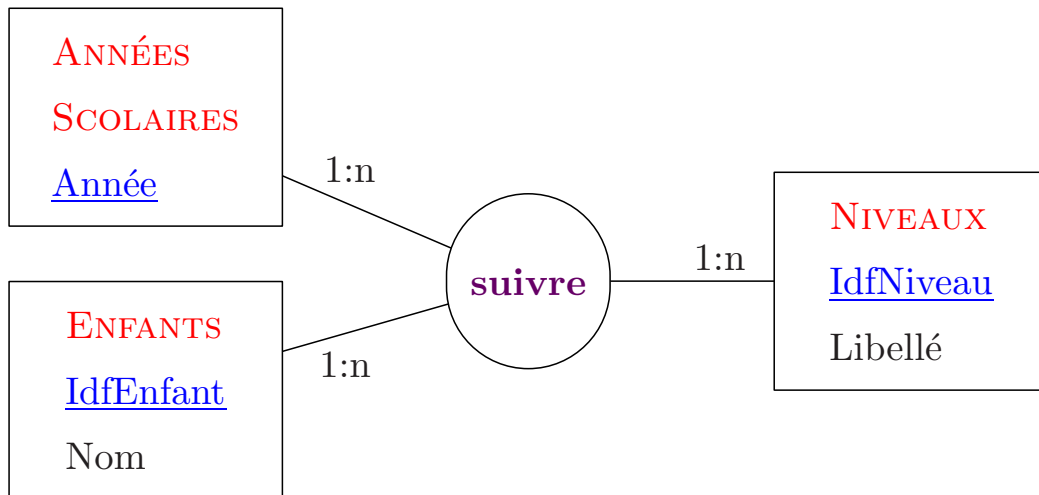
Dans le cas des associations ternaires, mettre en évidence une C.I.F. permet de faire apparaître des propriétés qui n'étaient pas indiquées.

Elle permet d'indiquer que la connaissance d'une occurrence dans deux entités détermine la connaissance d'une occurrence dans la troisième entité.

Vous remarquerez, dans l'exemple suivant, que la C.I.F. ne correspond pas forcément à des cardinalités 1:1.

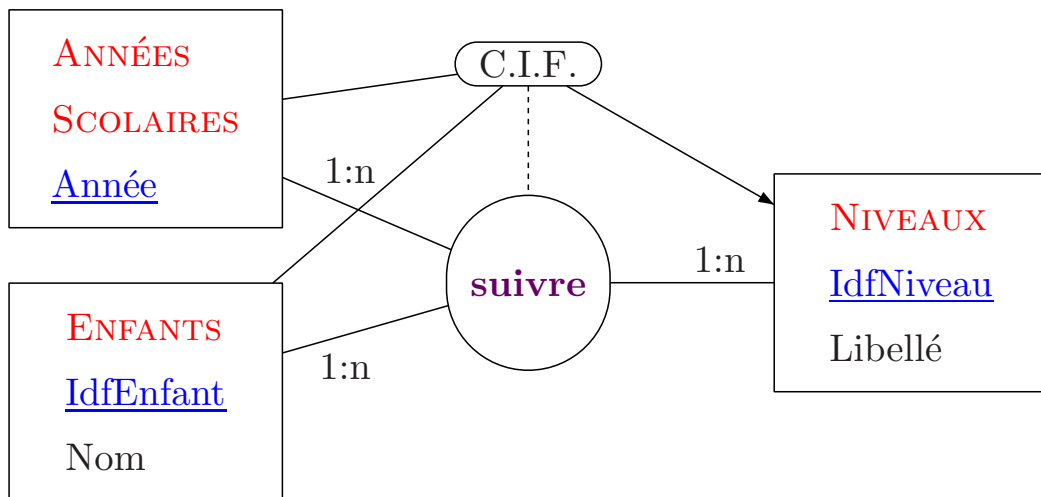
Contraintes d'intégrité fonctionnelle

Exemple : un élève suit un niveau scolaire chaque année.



Contraintes d'intégrité fonctionnelle

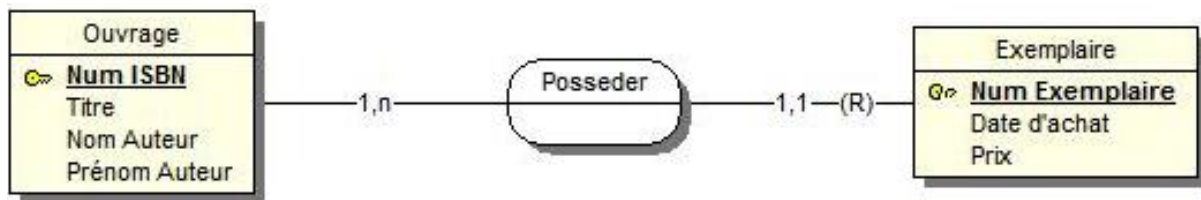
La C.I.F. permet d'indiquer qu'une année et un élève déterminent un niveau.



Notion d'identifiant relatif

Certaines entités ont une existence totalement dépendante d'autres entités. Dans ce cas nous avons recours à un identifiant relatif.

Exemple Nous devons gérer les exemplaires des livres commandés dans une bibliothèque :



Nous pouvons considérer qu'une bibliothèque achète plusieurs exemplaires du même livre.

Doit-elle numéroté les exemplaires :

▷ dans l'ordre séquentiel croissant c'est-à-dire de 1 à 6 000

- ▷ ou pour chaque ouvrage de 1 au nombre maximal de **cet ouvrage** présent dans la bibliothèque ?

Il paraît évident que la solution la plus judicieuse est la seconde proposition.

Ainsi, la clé d'identification de l'exemplaire sera la concaténation du Num ISBN et du Num Exemplaire. L'identification d'un identifiant relatif se traduit par le symbole **(R)** à côté de la cardinalité.

Pour reconnaître un identifiant relatif, nous pouvons aussi nous poser la question :

« si je supprime l'ouvrage X de ma bibliothèque, dois-je supprimer tous les exemplaires ? ».

La réponse est bien entendu oui. Nous pouvons donc dire que l'existence même de l'entité Exemplaire dépend de l'existence de l'entité Ouvrage.

Voici un autre exemple :

Vous êtes le responsable informatique d'un centre de formation possédant un campus sur lequel sont construits 3 bâtiments strictement identiques :

- ▷ Ils possèdent 3 étages
- ▷ chaque étage héberge 12 salles informatiques
- ▷ dans chaque salle peuvent être installés 10 ordinateurs.

Nous pouvons en conclure que dans chaque bâtiment, nous pouvons installer 360 ordinateurs et la capacité du campus est de 3×360 ordinateurs soit 1080.

Imaginons le scénario suivant, vous recevez une dotation de 1080 ordinateurs à répartir dans toutes les salles. En outre, vous devez les numéroté pour les identifier.

Quelle solution choisissez-vous ?

- Une solution serait de les numéroté de 1 à 1080.
- Une autre serait de rendre la numérotation relative à la salle.

Dans cette dernière solution, par exemple, dans la salle 1 du premier étage du bâtiment A la numérotation serait de 1-1 à 1-10, dans la salle 2 la numérotation irait de 2-1 à 2-10.

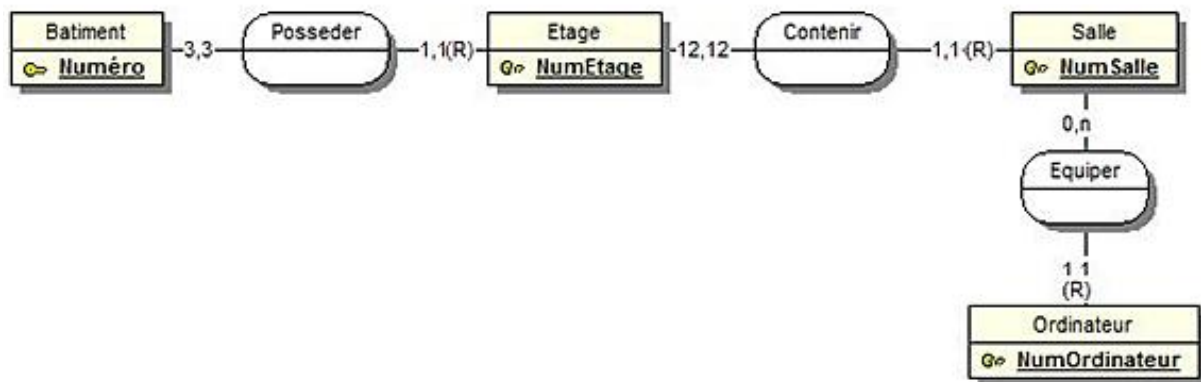
Maintenant, il serait judicieux de rendre la salle relative à l'étage. La numérotation deviendrait E1-1-1, pour le premier ordinateur de la salle 1 de l'étage 1, ou E3-12-10 pour le dixième ordinateur de la salle 12 de l'étage 3.

Nous pourrions aussi rendre les étages relatifs aux bâtiments et numéroté les ordinateurs de la façon suivante : BA-E3-12-10.

Ainsi, nous savons que nous avons affaire à l'ordinateur 10 de la salle 12 de l'étage 3 du bâtiment A.

Si les ordinateurs peuvent être changés de lieu rien qu'en regardant leur identifiant, vous pouvez les réinstaller dans leurs salles respectives.

Voici le modèle conceptuel découlant de cet exemple :



Si le centre de formation décide de vendre un bâtiment, tout disparaît en cascade !

4.3 Modèle Logique de Données (MLD)

Modèle logique des données

le **MCD** a pour but de modéliser les données (aspect statique) mémorisées dans le système d'information ;

le **MLD (Modèle Logique des Données)** tient compte des choix concernant le système de gestion des données utilisé dans l'entreprise. Le plus utilisé est le modèle relationnel associé aux bases de données relationnelles (Oracle, Informix, SQLserver, ..., Access, Foxpro, Paradox, ... HSQL, mySQL, postgresQL...)

Modèle logique des données

Modèle relationnel

- Le **domaine** est l'ensemble des valeurs que peut prendre une donnée.
- Une **table** porte un nom et est composée d'**attributs** prenant leurs valeurs dans les domaines correspondants.
Une table est un sous-ensemble du produit des domaines ; une table est donc un ensemble d'enregistrements (ou tuples, ou lignes).
- Une **clé** est constituée de 1 ou plusieurs attributs telle que une valeur de la clé détermine exactement l'enregistrement.
- Toute table possède une **clé primaire** et, éventuellement, des **clés candidates**.

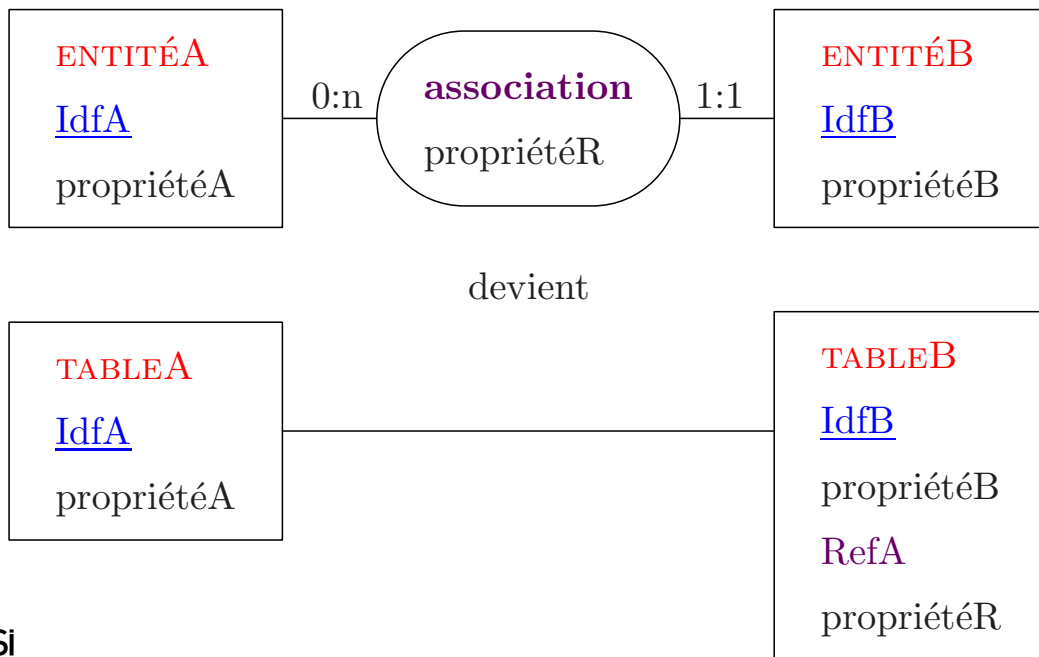
Modèle logique des données le passage du MCD au MLD

Pour les entités. Toute entité devient une table, les propriétés de l'entité sont les attributs de la table, l'identifiant de l'entité est la clé primaire de la table.

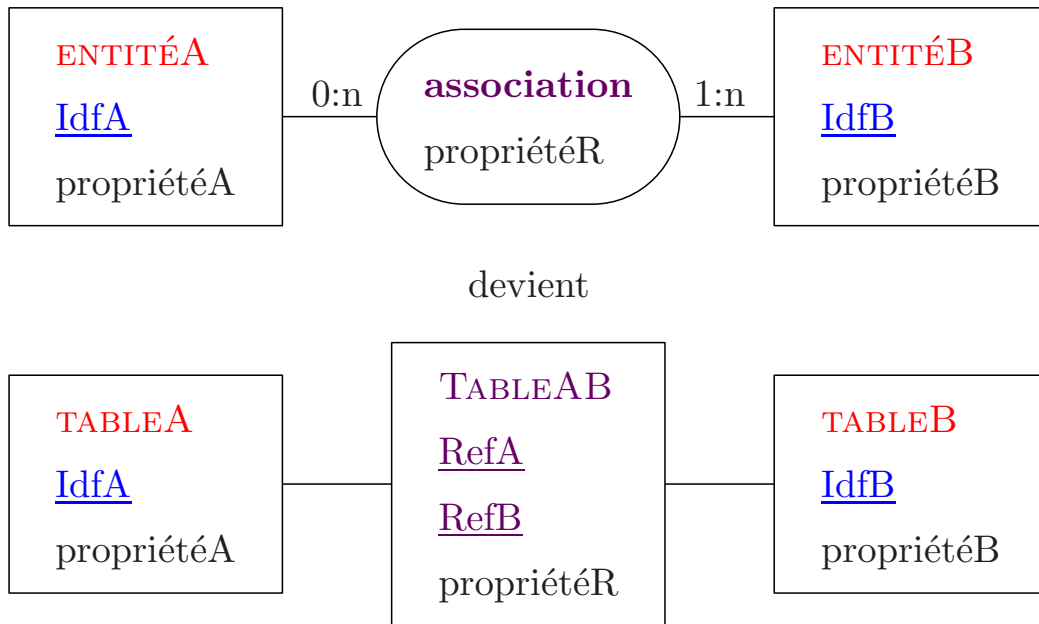
Pour les associations. Cela dépend des cardinalités. Deux cas sont possibles :

- association $\square - \overset{1:1}{-} - \bigcirc - \overset{1:n}{-} - \square$: la relation est matérialisée par l'ajout d'une clé étrangère.
- association $\square - \overset{1:n}{-} - \bigcirc - \overset{1:n}{-} - \square$: la relation donne lieu à la création d'une table.
- les cardinalités 0:1 (resp. 0:n) se traitent comme les cardinalités 1:1 (resp. 1:n).

Modèle logique des données le passage du MCD au MLD

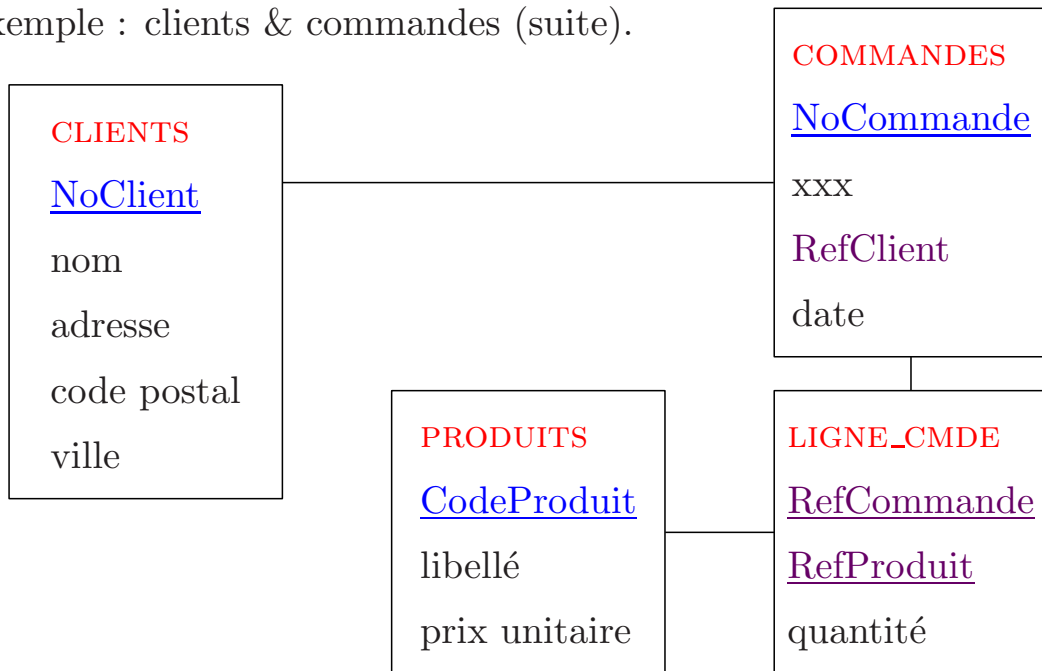


Modèle logique des données le passage du MCD au MLD

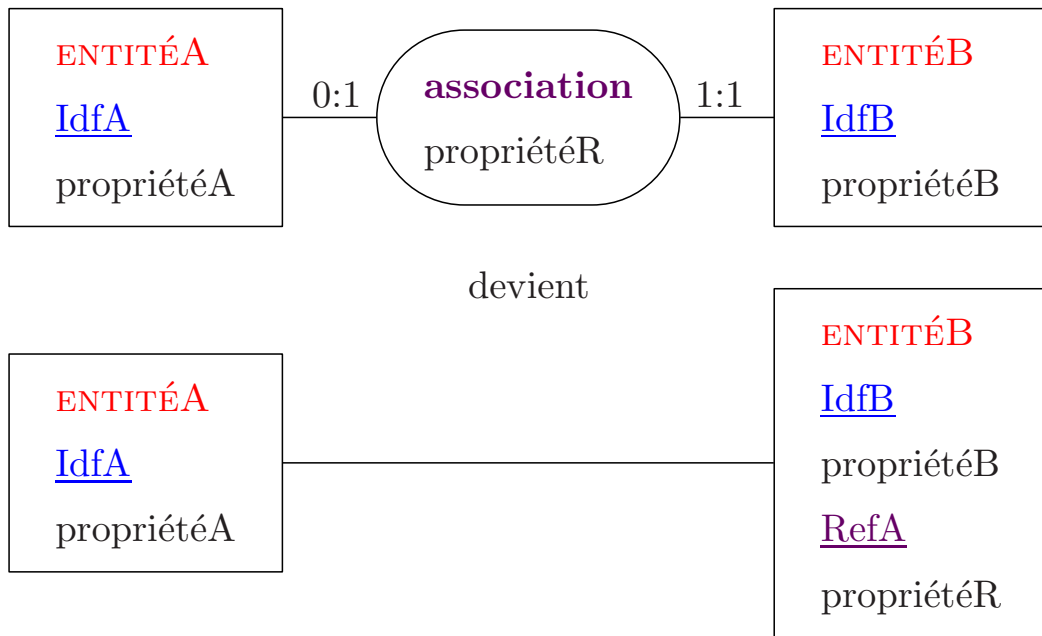


Modèle logique des données

Exemple : clients & commandes (suite).



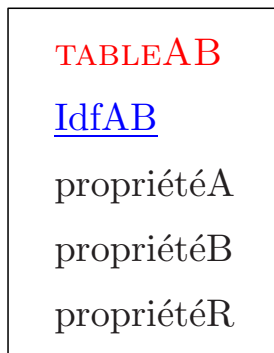
Modèle logique des données : cas particuliers



Modèle logique des données : cas particuliers

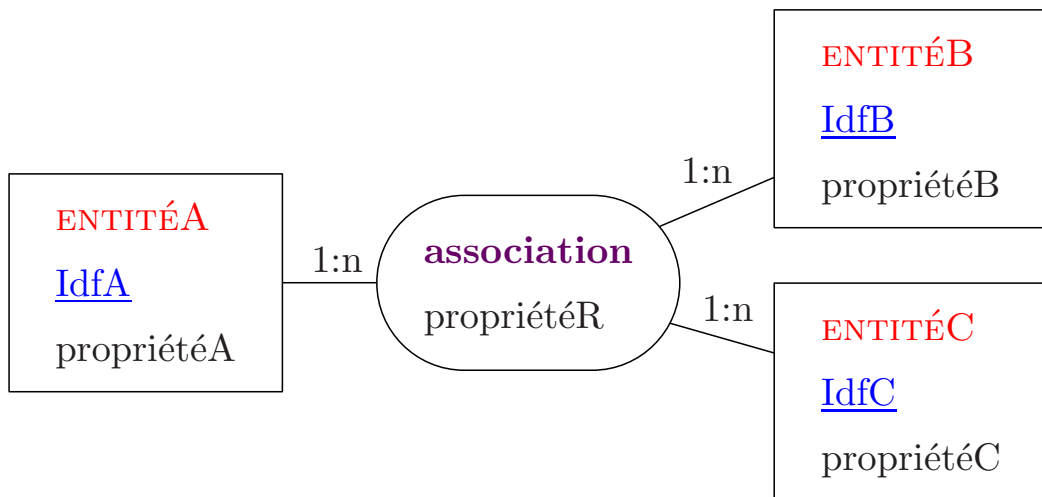


devient

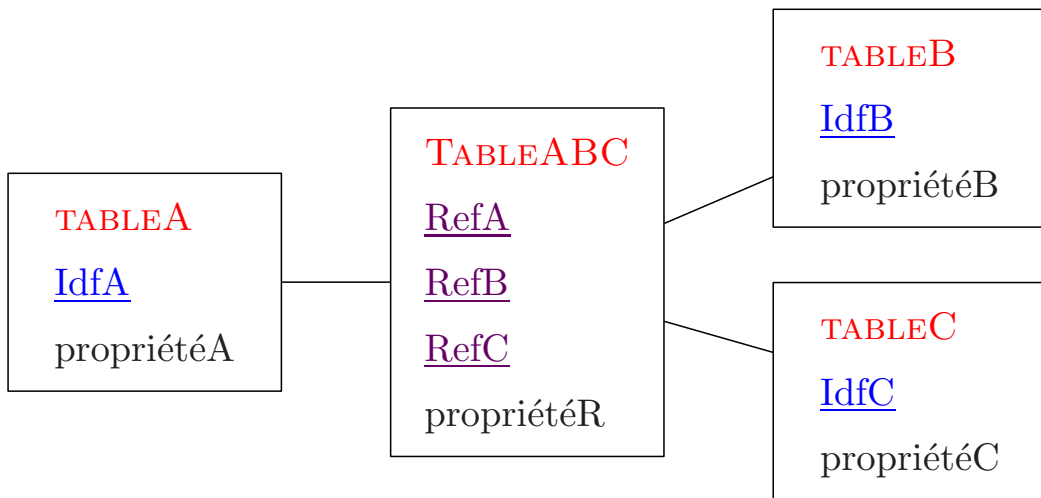


MLD : quelques cas particuliers

Une association ternaire devient une table si les cardinalités sont 1:n sur toutes les branches.

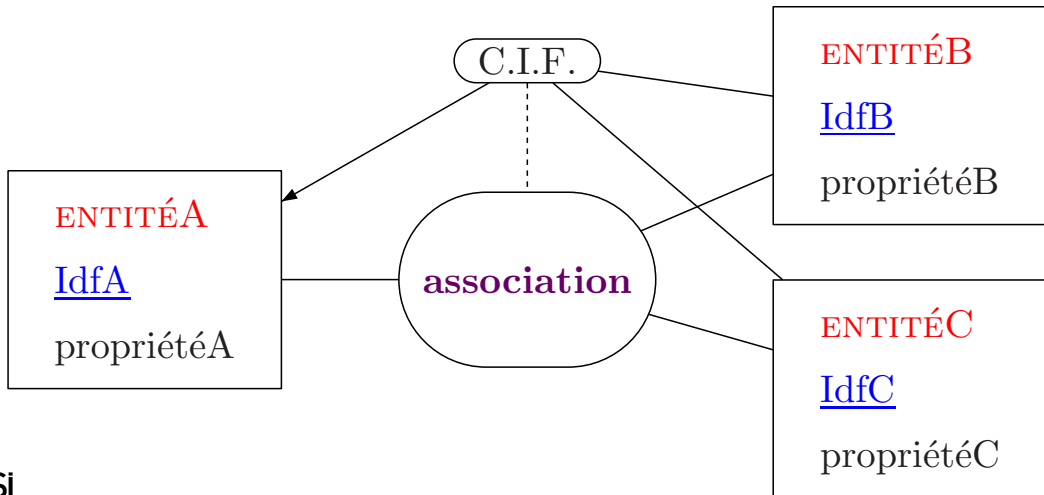


devient

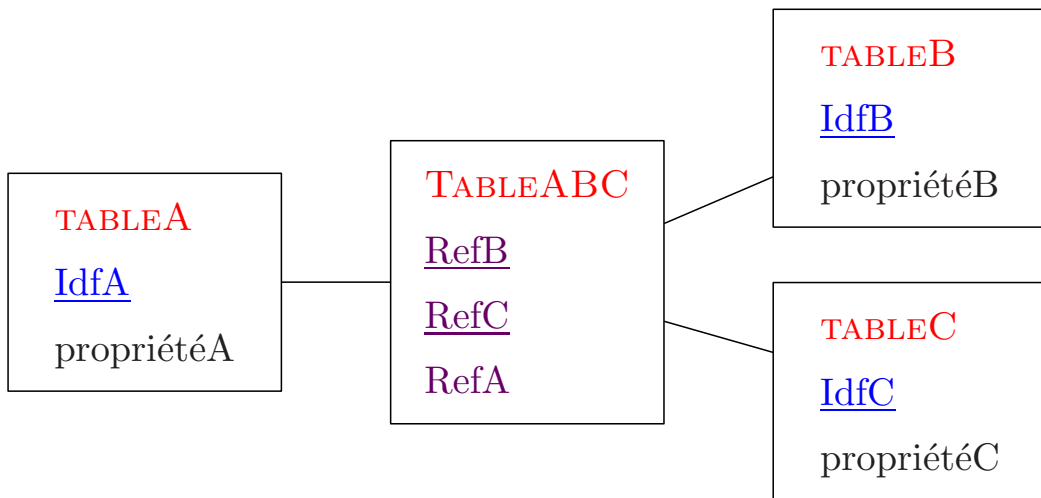


MLD : quelques cas particuliers

Si une C.I.F. est associée à une association ternaire (avec des cardinalités 1:n sur toutes les branches), l'identifiant de la table intermédiaire n'est pas la concaténation des trois identifiants.

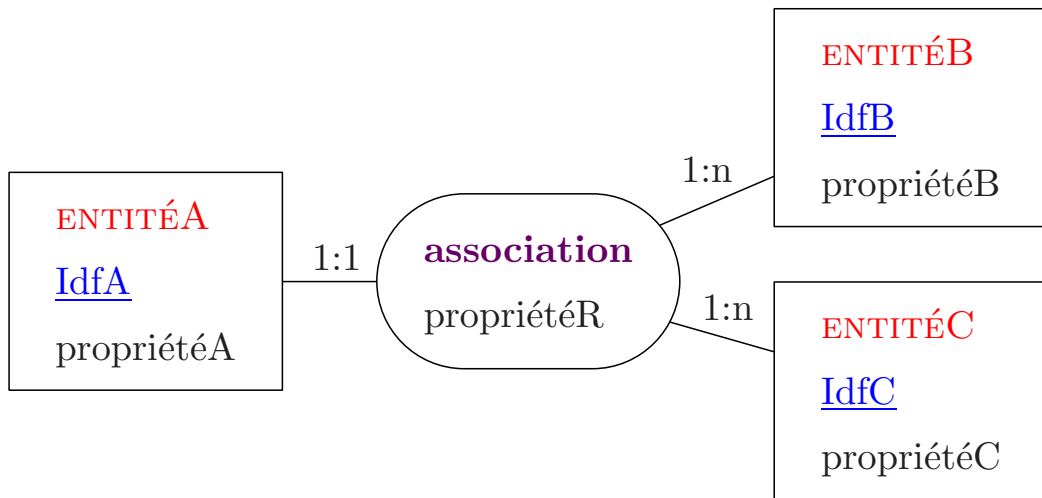


devient

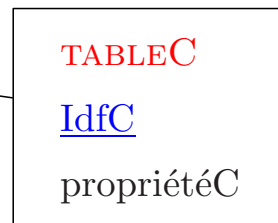
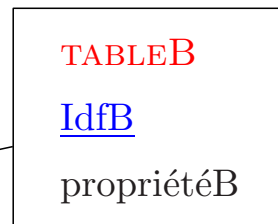
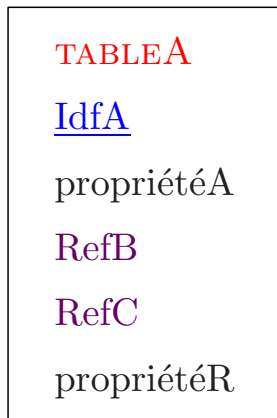


MLD : quelques cas particuliers

Si une association ternaire possède une branche avec une cardinalité 1:1, on place les références dans la table reliée par 1:1.



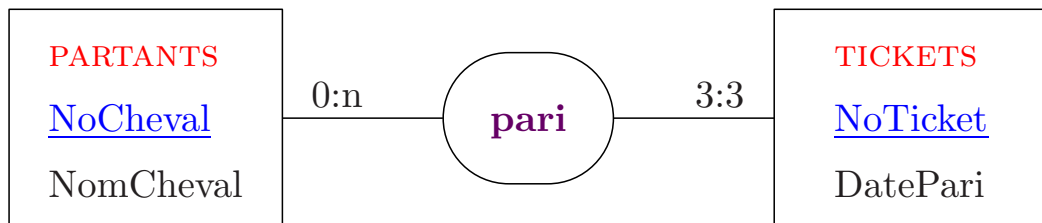
devient



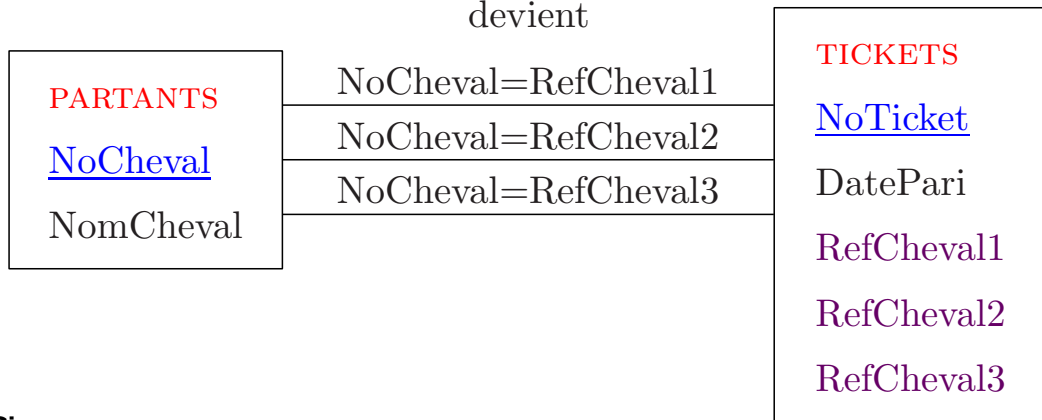
MLD : quelques cas particuliers

Si plusieurs associations relient deux entités, on traite chacune des associations indépendamment des autres, ce qui peut donner lieu à l'ajout de plusieurs références. Exemple : **TRAJETS****SNCF** (NoTrajet, HeureDépart, HeureArrivée, RefGareDepart, RefGareArrivée).

Les cardinalités k:k donnent aussi lieu à l'ajout de plusieurs références. Exemple du tiercé : **TICKETS** (NoTicket, DatePari, RefCheval1, RefCheval2, RefCheval3).



devient



Modèle logique des données

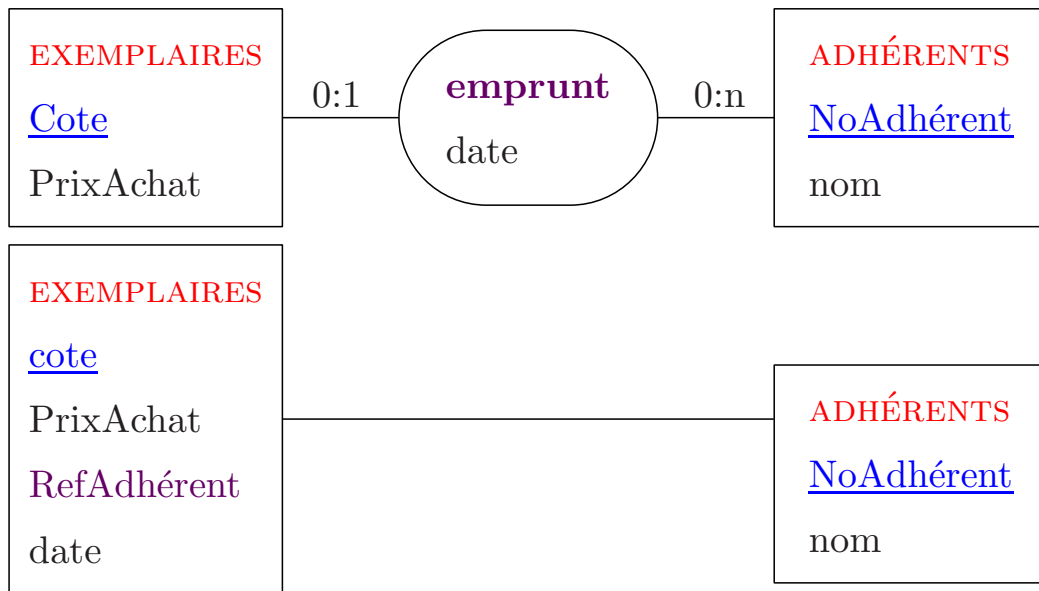
Optimisation du MLD

Le système de gestion des données doit répondre au souci d'une gestion sûre des données. Ceci est assuré par la construction des MCD et MLD. Mais cette gestion se doit aussi d'être efficace. Par conséquent, il faut **optimiser** le MLD, c'est-à-dire, par exemple :

- les jointures étant coûteuses, supprimer les tables inutiles ;
- parfois, il peut être utile d'ajouter des données calculées pour accélérer certains traitements ;
- il peut être intéressant de gagner de l'espace mémoire, en ne respectant pas les règles de transformation MCD \rightarrow MLD.

Modèle logique des données : optimisation

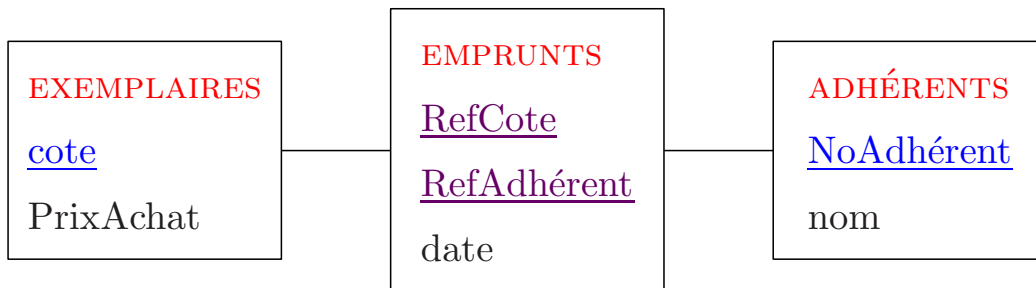
Les cardinalités 0:1 se traitent comme les cardinalités 1:1.



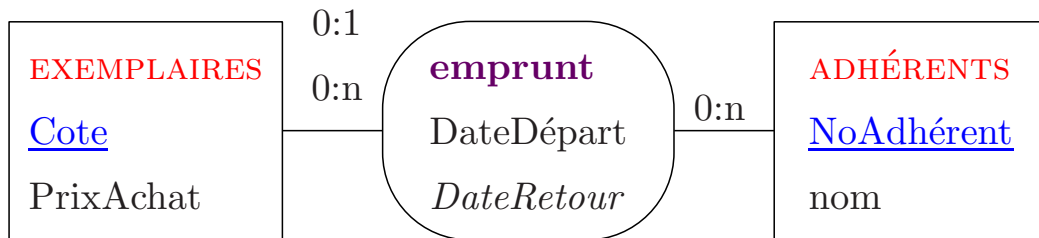
La valeur *Null* est autorisée.

Modèle logique des données : optimisation

Mais si peu de livres sont empruntés, de nombreuses fiches auront une référence *Null*. Ne serait-il pas mieux de faire ?



Modèle logique des données : optimisation

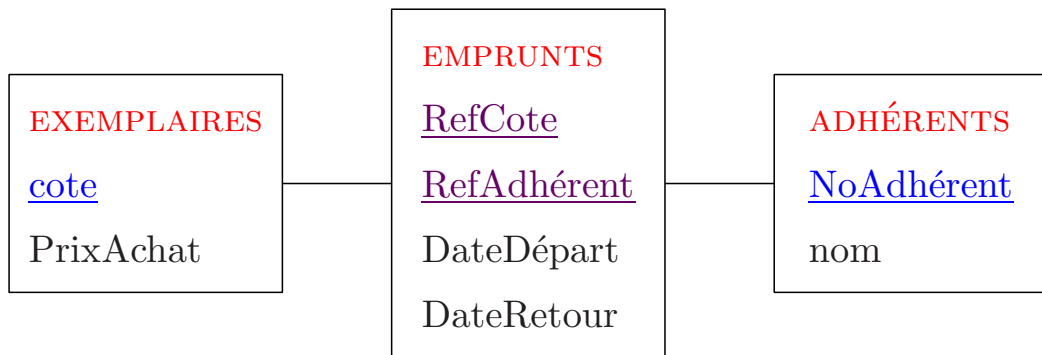


Les cardinalités 0:1 signifient qu'on gère seulement l'emprunt actuel. On ne mémoriserait pas la date de retour puisqu'au retour d'un exemplaire, on supprime la fiche d'emprunt.

Les cardinalités 0:n signifient qu'on gère l'historique des emprunts.

Modèle logique des données : optimisation

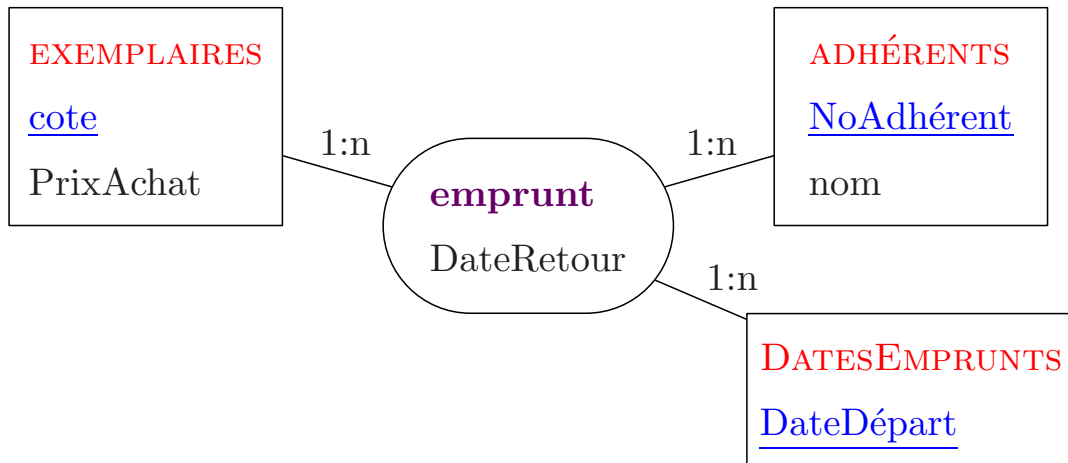
Dans le cas de la gestion de l'historique (cardinalités 0:n), on doit créer une table intermédiaire.



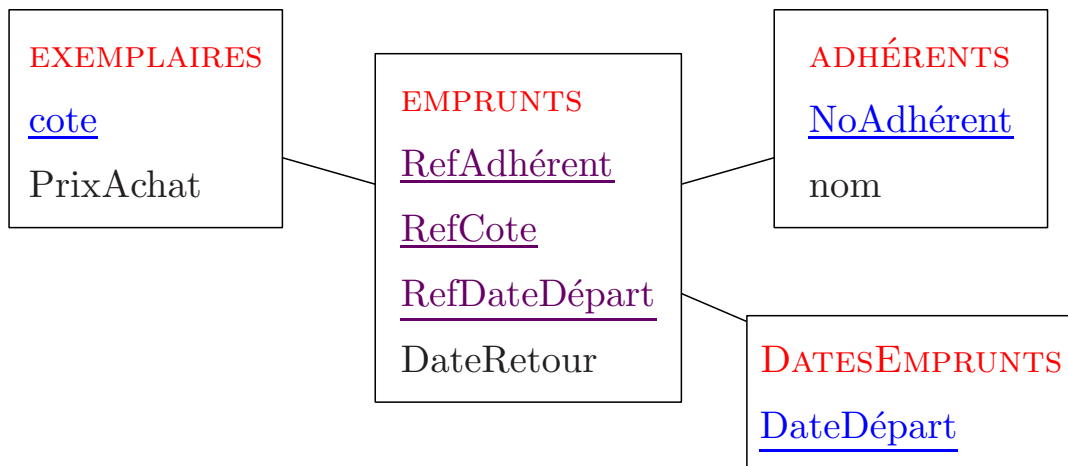
L'identifiant d'une association est la concaténation des identifiants des deux entités. Donc la clé primaire de la table EMPRUNTS serait le couple (RefCote,RefAdhérent). Est-ce le cas ?

Modèle logique des données : optimisation

Il est assez fréquent qu'une association ayant une date (une heure, une année...) comme propriété, cache une relation ternaire optimisée.



qui devient, dans le modèle logique des données



La table DATESEMPRUNTS est supprimée, par optimisation, car elle ne contient qu'une propriété dont les seules valeurs intéressantes sont dans la table EMPRUNTS.

Modèle logique des données

Contraintes d'intégrité

de structure : valeur de la clé primaire unique et toujours définie,

de domaine : les valeurs prises par un attribut doivent vérifier des contraintes,

de référence : les valeurs d'une clé étrangère doivent correspondre à des valeurs existantes dans la table d'origine.

La vérification des contraintes assurent que la base reste **intègre**. Cette vérification s'effectue, soit directement par le système de gestion de bases de données utilisé (quand c'est possible), soit par l'écriture de programmes.

4.3.1 Cas Pratique : Restaurant

Votre père est restaurateur et vous demande de lui réaliser un logiciel de gestion des commandes de repas. Voici les indications qu'il vous donne :

Il souhaite pouvoir gérer certaines informations concernant ses employés : nom, prénom, adresse complète, téléphone et diplômes.

Au niveau de la prise de commande, il souhaite savoir si elle porte sur le service de midi ou celui du soir et à quelle date elle a été passée.

Pour certains calculs statistiques, il souhaite aussi savoir quelle table a passé la commande et quel serveur l'a prise.

La carte du restaurant propose l'ensemble des plats d'entrées, principaux et desserts. Les menus proposés sont un assemblage des plats à la carte.

La carte des vins propose une sélection de vins qui sont stockés dans la cave du restaurant. Votre père désire connaître pour chaque bouteille son millésime, sa date d'achat, son prix d'achat et son prix de vente. Il voudrait saisir aussi pour chaque cru les informations concernant le viticulteur (nom, prénom, adresse complète, téléphone). À l'heure actuelle votre père,

amoureux du vin, met sur chaque goulot de chaque bouteille une étiquette contenant le prix d'achat ainsi que la date d'achat. Votre système doit pouvoir remplacer ce traitement manuel.

Ensuite, certaines boissons comme les apéritifs, les digestifs, les sodas ou les cafés sont gérés de façon simpliste juste par leur libellé et leur prix de vente.

Chaque serveur prenant une commande saisit l'ensemble des informations sur une mini-tablette qui transmet la commande via Wifi sur un ordinateur central.

À l'aide de ces quelques informations basiques, essayons de réaliser le modèle conceptuel. Nous allons commencer par un dictionnaire des données simplifié.

Travail demandé

Utilisez la démarche suivante pour définir le MCD :

- ▷ Établissez le dictionnaire des données
- ▷ Listez les dépendances fonctionnelles (élémentaires ou non)
- ▷ À partir des dépendances, établissez le MCD.
- ▷ À partir du MCD, établissez le MLD.

5 Modèle des traitements

Modèle Conceptuel des Traitements

Niveau Conceptuel : on ne tient pas compte de l'organisation ;

Les Traitements : partie dynamique du Système d'Information ; les traitements sont la traduction en actions des règles de gestion de l'entreprise ;

le MCT : est une représentation schématique de l'activité d'une entreprise indépendamment des choix d'organisation et des moyens d'exécution.

Modèle Conceptuel des Traitements

- Un **processus** est un sous-ensemble de l'activité de l'entreprise dont les points d'entrée et de sortie sont stables et indépendants de l'organisation ;
- Une **opération** est un ensemble d'actions exécutables sans interruption ;
- Un **événement** est une sollicitation (**externe** ou **interne**) du Système d'Information auquel celui-ci doit réagir ;
- Une **synchronisation d'événements** est une condition logique nécessaire au déclenchement d'une opération ;
- Un **résultat** est le produit d'une opération, événement interne qui peut être événement déclencheur d'autres opérations.

Modèle conceptuel des traitements

Construction d'un MCT

Quand : dans l'étude préalable : MCT de l'existant et ébauche du MCT de la nouvelle solution ; dans l'étude détaillée : MCT complet de la nouvelle solution.

Préalable : avoir explicité les règles de gestion ; avoir établi un diagramme des flux, c'est-à-dire une représentation graphique de la circulation des informations entre les différents acteurs de l'entreprise ;

Règles : Une opération est une suite ininterrompue d'actions ; aucun événement externe ne peut l'interrompre ; aucun résultat interne à une opération ne peut conditionner la suite de ses actions.

Remarque : Une application réelle est souvent constituée d'une juxtaposition de petits MCT.

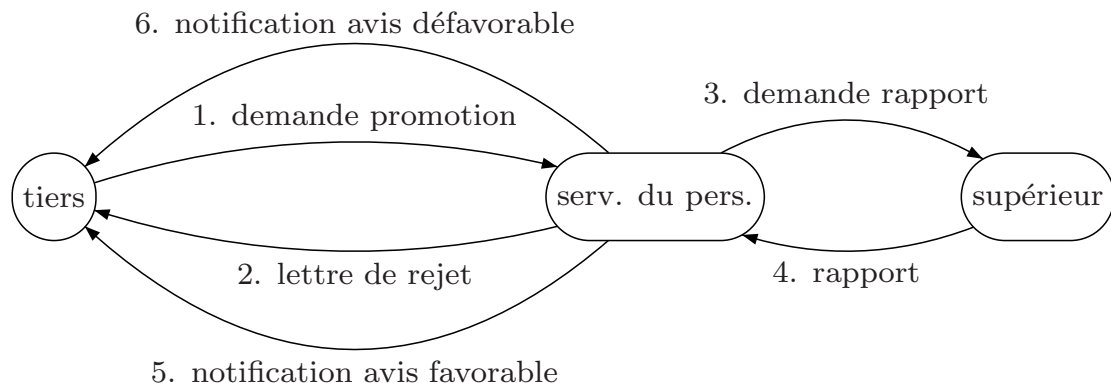
Modèle Conceptuel des Traitements : exemple

Dans une administration, les promotions sont traitées selon les règles de gestion suivante :

1. toute demande doit subir un examen préalable pour savoir si elle est recevable ;
2. l'examen d'une demande recevable ne peut se faire qu'après rapport du supérieur hiérarchique ;
3. après examen du dossier, la demande sera accordée ou refusée.

Modèle Conceptuel des Traitements : exemple

Première représentation de l'activité : le diagramme des flux, circulation chronologique des *informations* entre les *intervenants*.

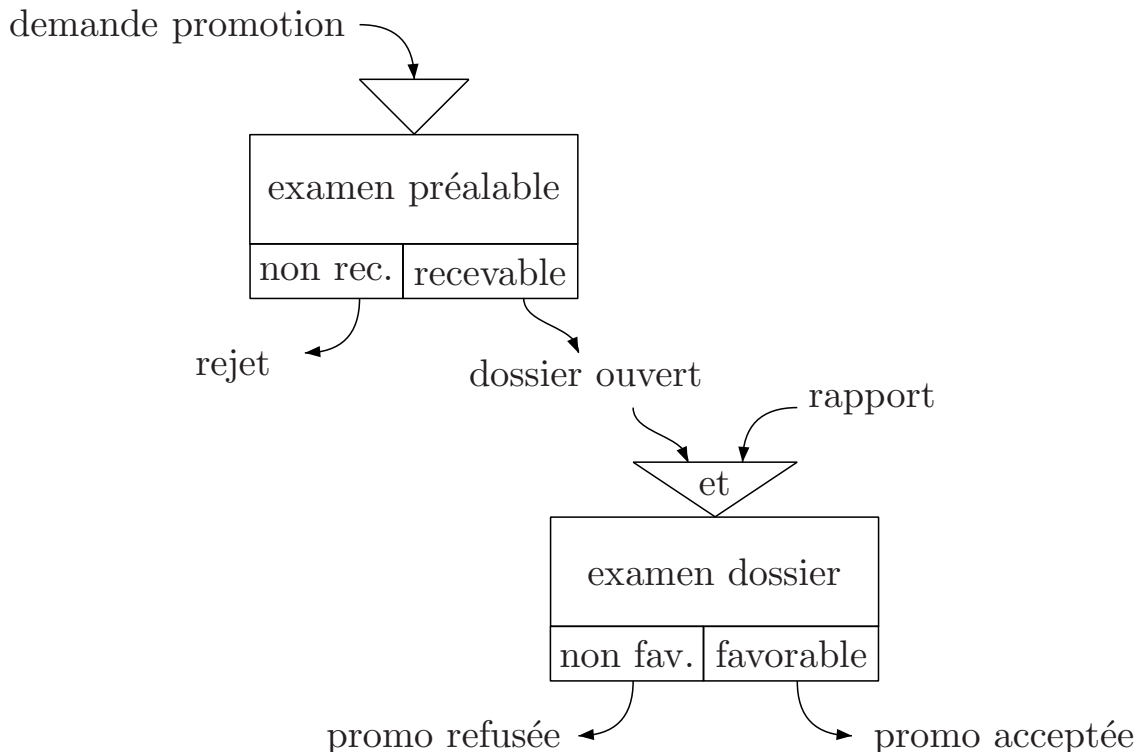


scénario “dossier rejeté” : 1,2

scénario “demande rejetée” : 1,3,4,6

scénario “demande acceptée” : 1,3,4,5

Modèle Conceptuel des Traitements : exemple

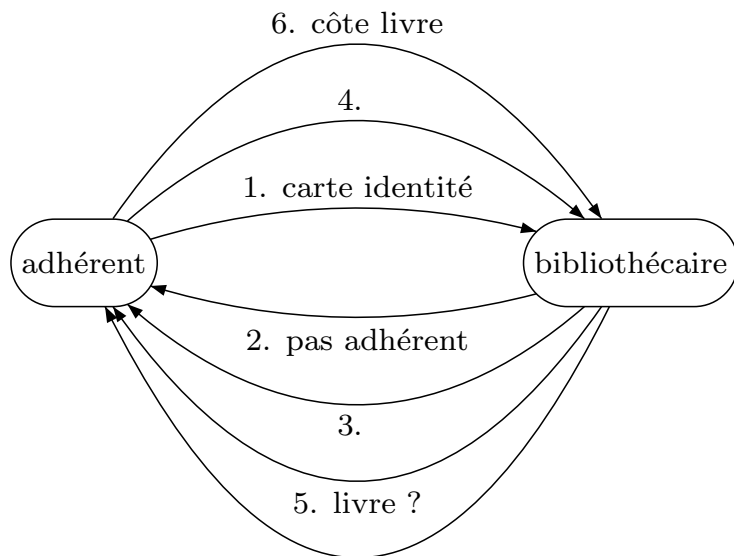


Modèle Conceptuel des Traitements : autre exemple

Règles de gestion pour l'emprunt de livres dans une bibliothèque :

1. L'emprunteur devrait présenter sa carte d'adhérent, mais, à défaut, on acceptera toute pièce d'identité attestant de l'adresse de la personne ;
2. l'emprunteur choisit ses livres dans les rayons et passent les enregistrer à l'accueil ;
3. les références du livre sont collées sur chaque exemplaire.

Le diagramme des flux.



7. au revoir et bonjour chez vous

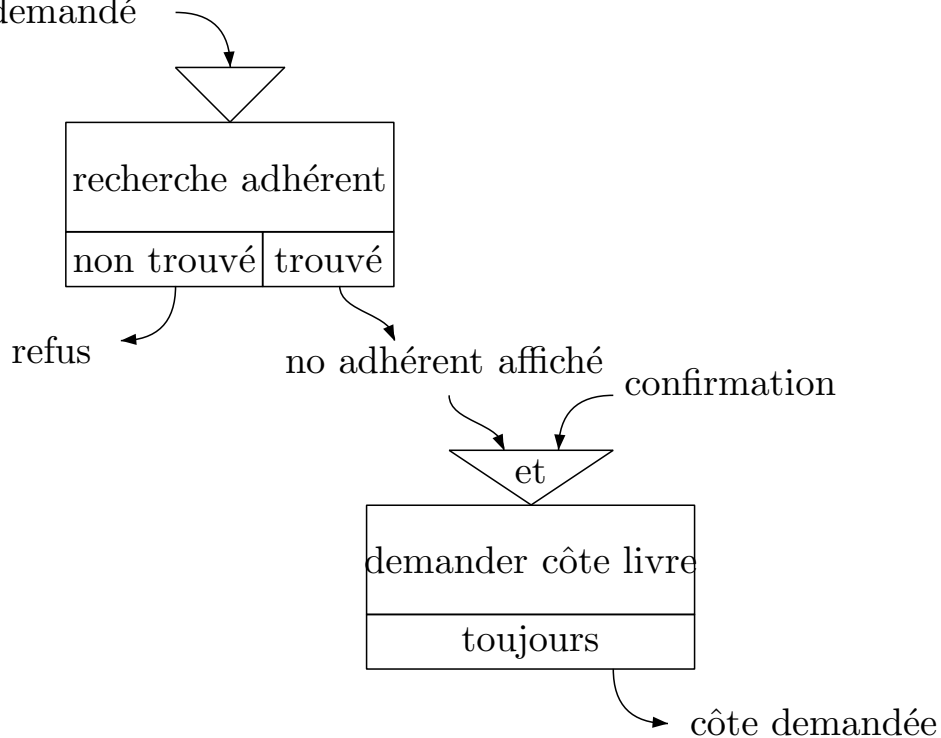
scénario “adhérent non enregistré” : 1,2

scénario “adhérent, carte d’identité” : 1,3,4,5,6,5,6...5,6,7

scénario “adhérent, carte d’adhérent” : 4,5,6,5,6...5,6,7

- **processus** : emprunt de livres
- **opérations** :
 - demander nom, prénom
 - les saisir
rechercher le numéro d'adhérent, l'afficher
demander confirmation
 - demander la référence du livre emprunté
 - saisir la référence
mémoriser l'emprunt et recommencer
- **événements** : nom donné, numéro adhérent affiché, numéro adhérent confirmé, côte livre demandé, côte livre
- **synchronisation** : no adh. affiché **ET** no adh. confirmé
- **résultat** : *numéro adhérent affiché* est le résultat de l'opération *rechercher le numéro d'adhérent*.

nom demandé



Modèle Organisationnel des Traitements : MERISE pur

On prend en compte l'organisation. On précise :

- affectation des traitements aux différents postes de travail ;
- enchaînement des traitements ;
- niveau et type d'automatisation des traitements :
 - manuel
 - automatisé :
 - * temps réel (interactif, conversationnel)
 - * temps différé (traitement “batch”)

Modèle Organisationnel des Traitements

“orienté traitement interactif”

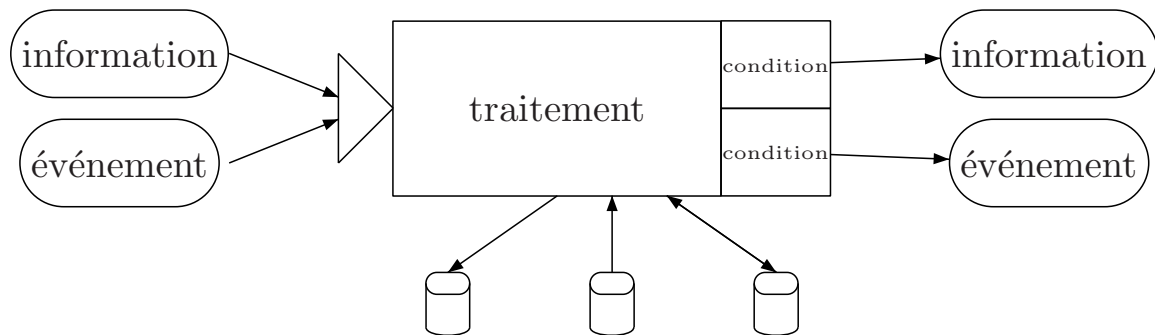
Le niveau de description des traitements dépend de la phase (étude préalable ou étude détaillée). Pour les traitements automatisés, on ajoute dans l'étude détaillée les informations suivantes dans le MOT :

- le traitement est découpé en unités de traitements (UT) ;
- les entrées nécessaires à chaque UT ;
- les tables concernées par chaque UT ;
- pour chacune des tables, le type d'accès (lecture, écriture, lecture-écriture) ;
- les résultats produits par UT ;
- les droits des utilisateurs.

Modèle Organisationnel des Traitements

Représentation pour les traitements automatisés

Le MOT reprend le même type de schéma que le MCT en plus détaillé. Les unités de traitement peuvent s'enchaîner. Une information fournie en résultat d'une unité de traitement sert alors d'entrée à une autre unité.



tables utilisées en écriture, lecture ou lecture/écriture

Étude Technique et Réalisation

Modèle physique des données : le niveau physique concerne l'administrateur de la base ; on définit les tables, les contraintes d'intégrité, les index, les requêtes, les vues (si différents niveaux d'utilisateurs), ... ; une partie importante est prise en charge par le logiciel

Modèle physique des traitements : On précise l'organisation générale de l'application, la nature des écrans et leurs enchaînements, les spécifications des programmes, ...

Étude Technique et Réalisation

- Il faut **structurer** l'application, c'est-à-dire regrouper les unités de traitement dans des ensembles cohérents. Dans la réalisation, on utilisera un type d'interface ou un autre selon les capacités de l'équipe de développement et les habitudes des futurs utilisateurs : menus déroulants ; menus pages ; barres d'outils ; ...
- Les choix doivent être cohérents au sein de l'application et permettre une navigation aisée dans les applications.
- Le développement se fait par modules, des tests devant être effectués à chaque étape de développement.

Récapitulatif

Étude préalable : MCD et MCT existant ; diagramme des flux ; ébauche MCD et MCT nouvelle solution ; évaluation et décision ;

Étude détaillée : MCD, MLD, MCT, MOT complets de la nouvelle solution ;

Étude technique : choix techniques et mise en place d'un cahier des charges de réalisation ;

Réalisation : programmation et tests ;

Mise en œuvre et maintenance : implantation ; saisie des données ; formation et cahier d'utilisation ; cahier de maintenance.