

La conception de bases de données relationnelles avec MERISE

Le Système d'information (SI)

Le système d'information est l'ensemble des moyens humains, matériels et immatériels mis en œuvre afin de gérer l'information au sein d'une unité, une entreprise par exemple.

Le SI possède quatre fonctions essentielles :

- la saisie ou collecte de l'information ;
- la mémorisation de l'information à l'aide de fichier ou de base de données ;
- le traitement de l'information afin de mieux l'exploiter (consultation, organisation, mise à jour, calculs pour obtenir de nouvelles données...) ;
- la diffusion de l'information.

MERISE

Merise est une méthode d'analyse et de conception de système d'information sur le principe de la séparation des données et des traitements.

Elle possède un certain nombre de modèles qui sont répartis sur trois niveaux :

- Le niveau **conceptuel**
- Le niveau **logique** ou **opérationnel**
- Le niveau **physique**

Le niveau CONCEPTUEL - le MCD

MCD : **Model Conceptuel de Données**

Représentation graphique et structurée des informations

Basé sur deux notions simples : **entités** et **associations**

L'élaboration passe par les étapes suivantes :

- la mise en place de **règles de gestion**
- l'élaboration du **dictionnaire des données**
- la recherche des **dépendances fonctionnelles** entre ces données ;
- l'élaboration du MCD (création des **entités** puis des **associations** puis ajout des **cardinalités**).

Le MCD - 1 - Les règles de gestion métier

Recueillir les besoins des futurs utilisateurs de votre application.

Établir les règles de gestion des données.

Le MCD - 1 - Les règles de gestion métier

Exemple

Demande initiale :

Nous voulons réaliser une application de gestion de notre bibliothèque. L'application recensera tous les livres que nous possédons, toutes les personnes inscrites à la bibliothèque et devra permettre de savoir qui a emprunter les livres.

Le MCD - 1 - Les règles de gestion métier

Tout ceci est très vague.

Quelles questions puis-je poser pour savoir quelles données je dois conserver ?

Demande initiale :

Nous voulons réaliser une application de gestion de notre bibliothèque.

L'application recensera tous les livres que nous possédons, toutes les personnes inscrites à la bibliothèque et devra permettre de savoir qui a emprunter les livres.

Le MCD - 1 - Les règles de gestion métier

- Quelles informations doit-on connaitre sur les livres ?
- Comment identifier les livres ?
- Comment emprunter un livre ?
- Comment s'inscrire pour emprunter un livre ?

Pour répondre à ces questions, un dialogue s'installe entre le développeur et le client.

Le MCD - 1 - Les règles de gestion métier

Les règles finales de notre exemple :

- pour chaque livre, on doit connaître le titre, l'année de parution, un résumé et le type (roman, poésie, science-fiction...) ;
- un livre peut être rédigé par aucun (dans le cas d'une œuvre anonyme), un ou plusieurs auteurs dont on connaît le nom, le prénom, la date de naissance et le pays d'origine ;
- chaque exemplaire d'un livre est identifié par une référence composée de lettres et de chiffres et ne peut être paru que dans une et une seule édition ;
- un inscrit est identifié par un numéro et on doit mémoriser son nom, prénom, adresse, téléphone et adresse e-mail ;
- un inscrit peut faire zéro, un ou plusieurs emprunts qui concernent chacun un et un seul exemplaire. Pour chaque emprunt, on connaît la date et le délai accordé (en nombre de jours).

Le MCD - 2 - Le dictionnaire de données

Le dictionnaire des données est un document qui regroupe toutes les données que vous aurez à conserver dans votre base (et qui figureront donc dans le MCD). Pour chaque donnée, il indique :

- Le code mnémonique (libellé unique désignant la donnée)
- La désignation
- Le type de données (Alphabétique, Numérique, Alphanumérique etc.)
- La taille de la donnée (nb de caractères)
- Remarques

Le MCD - 2 - Le dictionnaire de données

Code mnémonique	Désignation	Type	Taille	Remarque
id_i	Identifiant numérique d'un inscrit	N		
nom_i	Nom d'un inscrit	A	30	
prenom_i	Prénom d'un inscrit	A	30	
rue_i	Rue où habite un inscrit	AN	50	
ville_i	Ville où habite un inscrit	A	50	
cp_i	Code postal d'un inscrit	AN	5	
tel_i	Numéro de téléphone fixe d'un inscrit	AN	15	
tel_port_i	Numéro de téléphone portable d'un inscrit	AN	15	
email_i	Adresse e-mail d'un inscrit	AN	100	
date_naissance_i	Date de naissance d'un inscrit	Date	10	Au format AAAA-JJ-MM
id_l	Identifiant numérique d'un livre	N		
titre_l	Titre d'un livre	AN	50	
annee_l	Année de parution d'un livre	N	4	
resume_l	Résumé d'un livre	AN	1000	
ref_e	Code de référence d'un exemplaire d'un livre	AN	15	Cette référence servira également d'identifiant dans ce système
id_t	Identifiant numérique d'un type de livre	N		
libelle_t	Libellé d'un type de livre	AN	30	
id_ed	Identifiant numérique d'une édition de livre	N	6	
nom_ed	Nom d'une édition de livre	AN	30	
id_a	Identifiant numérique d'un auteur	N		
nom_a	Nom d'un auteur	A	30	
prenom_a	Prénom d'un auteur	A	30	
date_naissance_a	Date de naissance d'un auteur	Date		Au format AAAA-JJ-MM
id_p	Identifiant numérique d'un pays	N		
nom_p	Nom d'un pays	A	50	
id_em	Identifiant numérique d'un emprunt	N		
date_em	Date de l'emprunt	Date		Au format AAAA-JJ-MM
delais_em	Délai autorisé lors de l'emprunt du livre	N	3	S'exprime en nombre de jours

Les données doivent être élémentaires et non calculées.

On enregistrera un prix HT et un taux de TVA et pas un prix TTC par exemple.

Le MCD - 3 - Les dépendances fonctionnelles

Une dépendance fonctionnelle relie deux données si et seulement si on peut déduire une de ses données en fonction de l'autre.

Par exemple, un numéro d'emprunt me permet de déterminer une date d'emprunt, un délai, l'identifiant de l'inscrit ayant effectué l'emprunt, la référence de l'exemplaire emprunté.

On note ça comme ça : $id_em \rightarrow date_em, delais_em, id_i, ref_e$

Une DF doit être directe. Si $P1 \rightarrow P2$ et $P2 \rightarrow P3$ alors $P1 \rightarrow P3$ a été obtenue par transitivité et n'est donc pas directe. C'est pour ça qu'un numéro d'emprunt permet de déterminer la référence de l'exemplaire et pas l'intitulé de l'exemplaire.

Le MCD - 3 - Les dépendances fonctionnelles

En reprenant les données du dictionnaire précédent, on peut établir les DF suivantes :

id_em ? date_em, delais_em, id_i, ref_e

id_i ? nom_i, prenom_i, rue_i, ville_i, cp_i, tel_i, tel_port_i, email_i,
date_naissance_i

ref_e ? id_l

id_l ? titre_l, annee_l, resume_l, id_t, id_ed

id_t ? libelle_t

id_ed ? nom_ed

id_a ? nom_a, prenom_a, date_naissance_a, nom_p

Le MCD - 4 - L'élaboration du MCD

Les entités

Le formalisme d'une entité est le suivant :

Nom de l'entité
<u>identifiant</u>
propriété1
propriété2
...

Ainsi, si on reprend notre dictionnaire de données précédent, on schématise par exemple une entité «**Auteur**» comme ceci :

Auteur
<u>id_a</u>
nom_a
prenom_a
date_naissance_a

Le MCD - 4 - L'élaboration du MCD

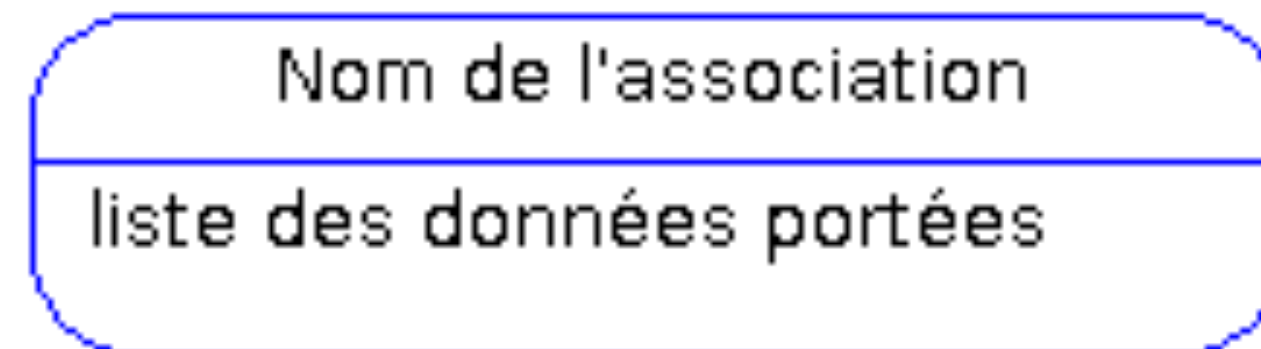
À partir de cette entité, on peut retrouver la règle de gestion suivante : un auteur est identifié par un numéro unique (id_a) et est caractérisé par un nom, un prénom et une date de naissance.

Une entité peut n'avoir aucune, une ou plusieurs occurrences. Pour illustrer ce terme d'« occurrence » qui a déjà été utilisé plusieurs fois, voici un exemple de table d'occurrences de l'entité Auteur :

id_a	nom_a	prenom_a	date_naissance_a
1	Hugo	Victor	1802-02-26
2	Rimbaud	Arthur	1854-10-20
3	de Maupassant	Guy	1850-08-05

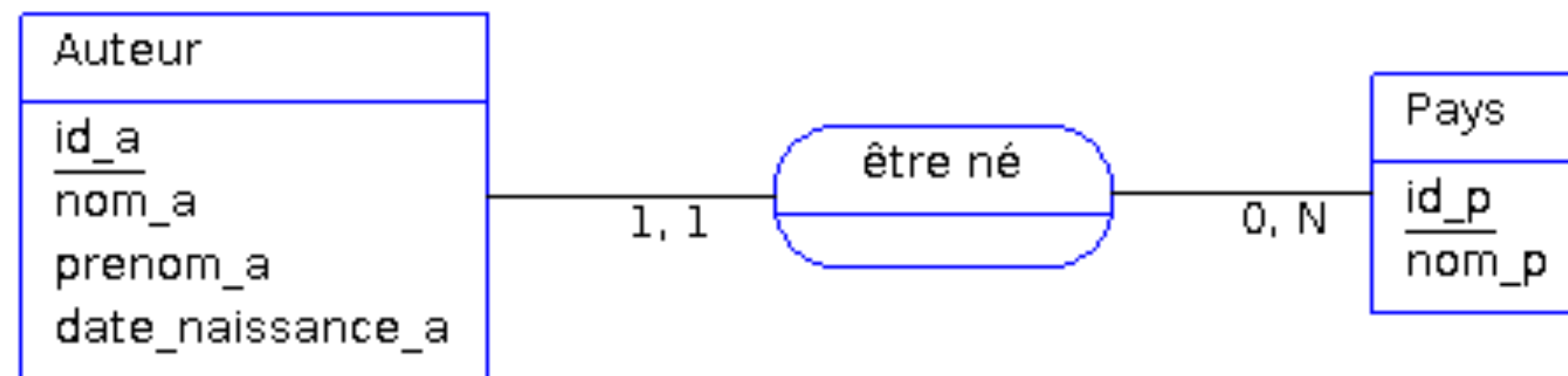
Le MCD - 4 - L'élaboration du MCD

Une **association** définit un lien sémantique entre une ou plusieurs entités. En effet, la définition de liens entre entités permet de traduire une partie des règles de gestion qui n'ont pas été satisfaites par la simple définition des entités.



Le MCD - 4 - L'élaboration du MCD

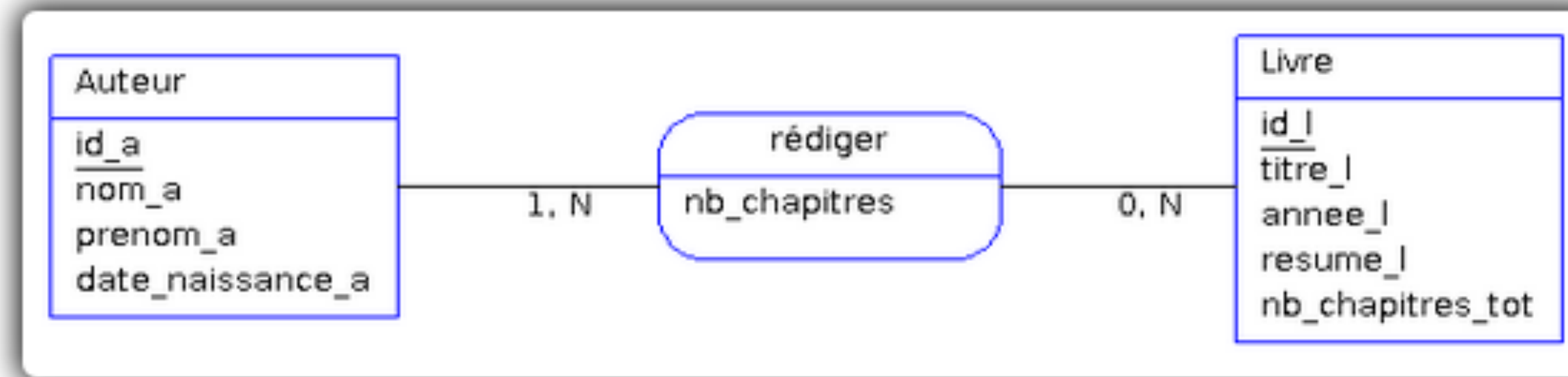
Généralement le nom de l'association est un verbe définissant le lien entre les entités qui sont reliées par cette dernière. Par exemple :



Ici l'association « être né » traduit les deux règles de gestion suivantes :

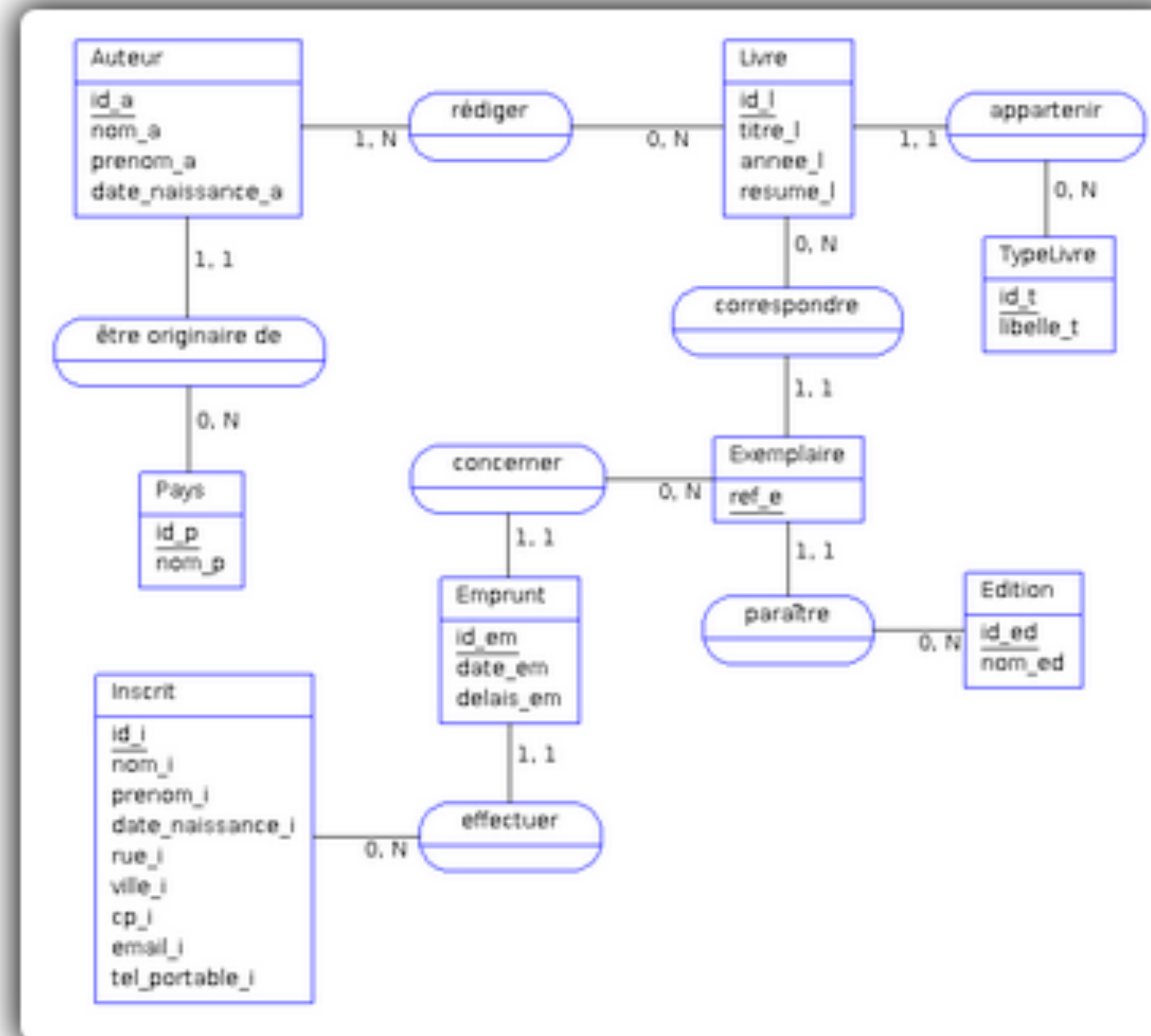
- un auteur est né dans un et un seul pays ;
- dans un pays, sont nés aucun, un ou plusieurs auteurs.

Le MCD - 4 - L'élaboration du MCD



Ici un auteur rédige au moins un ou plusieurs livres et pour chaque livre, on connaît le nombre de chapitres rédigés par l'auteur (on connaît aussi le nombre total de chapitres pour chaque livre).

Le MCD - 4 - L'élaboration du MCD



Du MCD vers le MLD

Le MLD : Modele Logique de Données

Le modèle logique des données consiste à décrire la structure de données utilisée sans faire référence à un langage de programmation. Il s'agit donc de préciser le type de données utilisées lors des traitements.

Ainsi, le modèle logique est dépendant du type de base de données utilisé.

Traduction d'une classe d'entité

Chaque classe d'entité du modèle conceptuel devient une table dans le modèle logique. Les identifiants de la classe d'entité sont appelé clés de la table, tandis que les attributs standards deviennent des attributs de la table, c'est-à-dire des colonnes.

Voici un premier exemple de relation (issue de l'entité « Edition » de notre précédent MCD) :

Edition (id_ed, nom_ed)

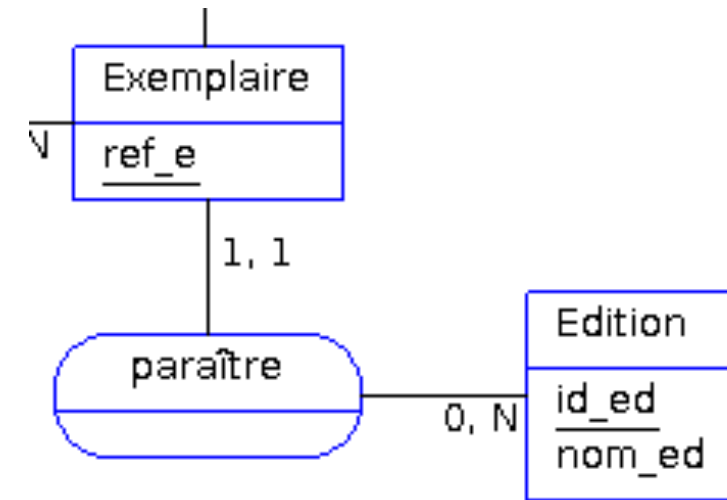
Légende :

x : relation

x : clef primaire

Traduction d'une classe de relation

Si une des classes d'entités possède une cardinalité (1,1) la table liée hérite d'un nouveau type de clés appelé **clef étrangère**. La clef étrangère est un attribut d'une relation qui fait référence à la clef primaire d'une autre relation (ces deux clés devront donc avoir le même type de données).



Complétons notre premier exemple avec une autre relation où apparaît une clef étrangère :

Edition (id_ed, nom_ed)

Exemple (ref_e, id_ed#)

Légende :

x : relation

x : clef primaire

x# : clef étrangère

Traduction d'une classe de relation

Dans les autres cas, on crée une table de relation qui comprend deux clés étrangères qui correspondent aux clés primaires des deux tables qui sont liées :



Categorie (id_cat, libelle_cat)

Livre (id_l, titre_l, annee_l, resume_l)

Appartenir (id_l#, id_cat#)

Légende :

x : relation

x : clef primaire

x# : clef étrangère

MLD final

Pays (id_p, nom_p)

Auteur (id_a, nom_a, prenom_a, date_naissance_a, id_p#)

TypeLivre (id_t, libelle_t)

Livre (id_l, titre_l, annee_l, resume_l, id_t#)

Rediger (id_a#, id_l#)

Edition (id_ed, nom_ed)

Exemplaire (ref_e, id_ed#, id_l#)

Inscrit (id_i, nom_i, prenom_i, date_naissance_i, rue_i, ville_i, cp_i, email_i, tel_i, tel_portable_i)

Emprunt (id_em, date_em, delais_em, id_i#, ref_e#)

Légende :

x : relation

x : clef primaire

x# : clef étrangère

Passage au SQL

Comme vous pouvez le constater, le schéma de la base est déjà fait.
Les règles de passage au SQL sont assez simples :

- chaque relation devient une table ;
- chaque attribut de la relation devient une colonne de la table correspondante ;
- chaque clef primaire devient une PRIMARY KEY ;
- chaque clef étrangère devient une FOREIGN KEY.

Exemple sur MYSQL