

Informatique documentaire et les bases de données

Wilfrid Niobet

▶ **To cite this version:**

Wilfrid Niobet. Informatique documentaire et les bases de données : Introduction UML/Merise/SQL. Master. Informatique documentaire et les bases de données, Université de Poitiers, département SHA, France. 2017, pp.64. cel-01540243

HAL Id: cel-01540243
https://hal.science/cel-01540243

Submitted on 16 Jun 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - ShareAlike 4.0 International License

Informatique documentaire et les bases de données
Introduction
UML / Merise / SQL

Master 2 esDOC
par Wilfrid NIOBET

13 mars 2017

BY-NC-SA

Contexte

Les bases de données sont au coeur du traitement de l'information. Dans l'environnement documentaire les normes permettent d'organiser l'information (ISBD, Marc21, Unimarc, FRBR, etc., et bientôt le RDA). Pourtant, les outils documentaires (SIGB) ne répondent pas toujours à tous les besoins des établissements documentaires. Quelles pourraient être les raisons de cette insatisfaction ?

- Les besoins documentaires sont en partie différents d'un établissement à un autre ;
- Les pratiques professionnelles sont hétérogènes ;
- Le traitement de l'information évolue en même temps que de nouveaux besoins apparaîssent ;
- Les objectifs et les résultats sont variables en fonction des politiques documentaires ;
- Les contraintes technologiques ne sont pas toujours faciles à suivre ;

BY-NC-SA

Objectifs

Dans un projet d'informatique documentaire (comme dans tout projet informatique), il est déterminant de définir :

- Les besoins (ce à quoi doit répondre l'outil informatique) ;
- Les objectifs (de pratiques et de résultats) ;
- Les contraintes (disponibilité, accessibilité, etc.) ;
- le cadre législatif (repect des données et de leur propriété, leur mise à disposition, les partenaires, etc.) ;
- Et ne pas oublier la sécurité de l'application pour elle même et l'environnement dans lequel elle se trouve (intégrité des données, sauvegarde des données et de l'application, failles de sécurité, etc.) ;

L'objectif de cette présentation est de souligner l'aspect déterminant et fondamentale de la conception d'une base de données, son fonctionnement et son rôle par rapport à son application dans un environnement documentaire.

BY-NC-SA

Dans l'environnement documentaire

D'une manière générale, le développement d'outils informatiques comme un système d'information s'inscrit dans un contexte spécifique à un moment donné pour répondre à des besoins précis. Toute la difficulté du dévelopement et de la mise en place d'un SIGB ou d'outils informatiques documentaires est de définir les usages communs aux établissements documentaires tout en laissant une grande souplesse dans la configuration pour permettre à chaque établissement de pouvoir associer le SIGB à une politique documentaire, une gestion et une communication qui lui est propre.

Pour bien choisir le SIGB d'un établissement il est nécessaire de bien connaître les besoins, les attentes, les évolutions politiques de l'établissement, l'origine du projet du SIGB, sa conception, la structure de la base de données, le choix technologique (langages de développement), environnement technique (besoin matériel et logiciel), sa durée de vie et sa capacité à évoluer.

Quelques exemples de besoins qui ne sont pas toujours pris en compte dans les outils documentaires :

- Il est souvent nécessaire d'associer au SIGB d'autres outils de diffusion pour répondre aux besoins des documents numériques : le SIGB de la BM de Limoges ne permet pas la gestion de documents patrimoniaux. Pour ce faire, ils ont choisi d'associer Omeka au SIGB pour permettre de mieux gérer les documents et les métadonnées associées (tout comme l'école des Chartes).
- Il faut parfois modifier l'interface des utilisateurs en l'intégrant dans un site Web pour la recherche de document. Au SCD d'Angers, ils ont réussi à intégrer dans leur CMS Drupal un moteur de recherche documentaire, Ex-Libris ne pouvant répondre à leur demande de développement.
- L'exploitation de statistiques d'utilisation servant à la valorisation. Au SCD de Poitiers, Absysnet ne permettrait pas de faire ressortir certains éléments statistiques. Ils ont été obligés d'utiliser une autre application "business object" pour faire ressortir ce dont ils avaient besoin.
- L'absence de considération pour certaines données métiers permettant la valorisation de l'établissement dans la conception même de l'outil informatique. Notamment les données utilisateurs qui restent succinctes et ne permettent pas d'élaborer des études sur les publics et l'usage qu'ils font du pôle documentaire.
- etc..

BY-NC-SA

Sommaire

Qu'est-ce qu'une base de données ?
- Représentation sur 4 niveaux

Pourquoi une base de données relationnelle ?
- Une donnée brute
- Caractèristique "relationnelle"

Organisation des données
- Unified Modeling Language (UML)
- Plusieurs manières de schématiser
- Que doit-on formaliser ?

Mise en oeuvre d'un projet
- Le traitement de l'information

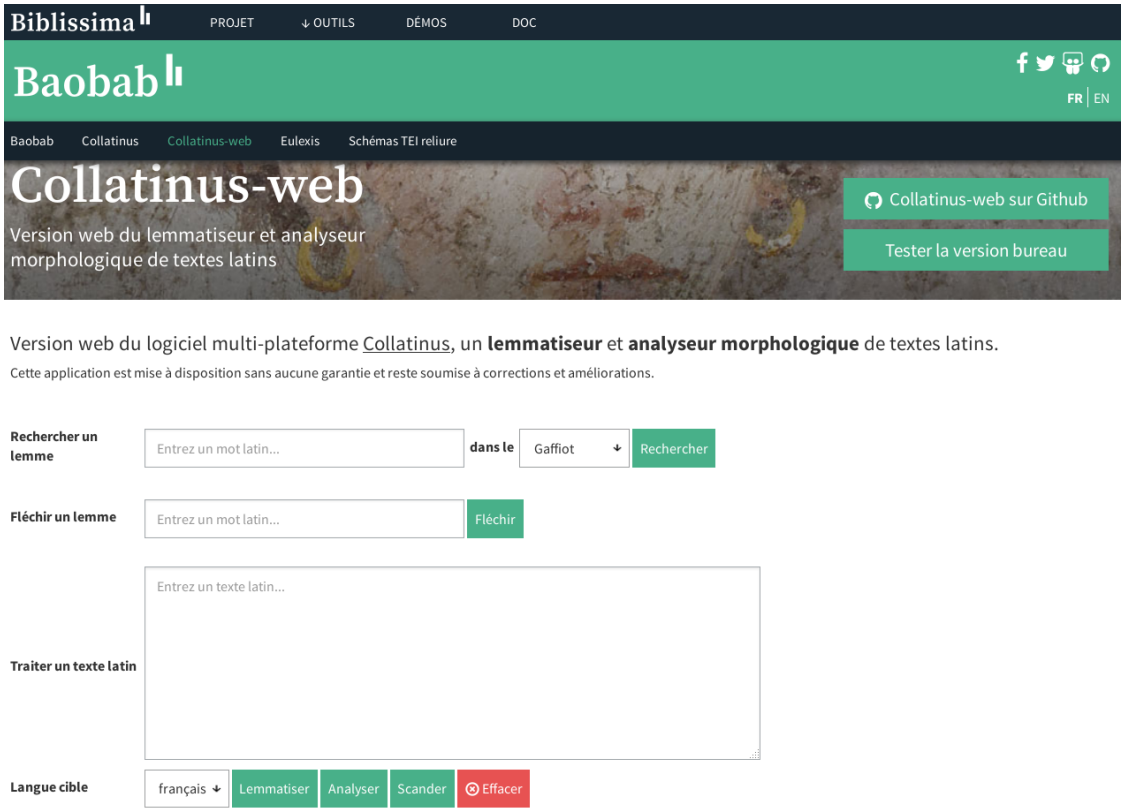
Méthode MERISE
- Modèle Conceptuel de Données
- Définition d'une cléf primaire
- Définition d'une cléf étrangère
- Association binaire
- Règles de transcodage
- Modèle Logique de Données
- Écriture logique

Base de données : le langage SQL
- SQL-Data Definition Language
- SQL-Data Manipulation Language
- SQL-Data Control Language

Ressources et références

Qu'est-ce qu'une base de données ?

Est-ce une base de données ?

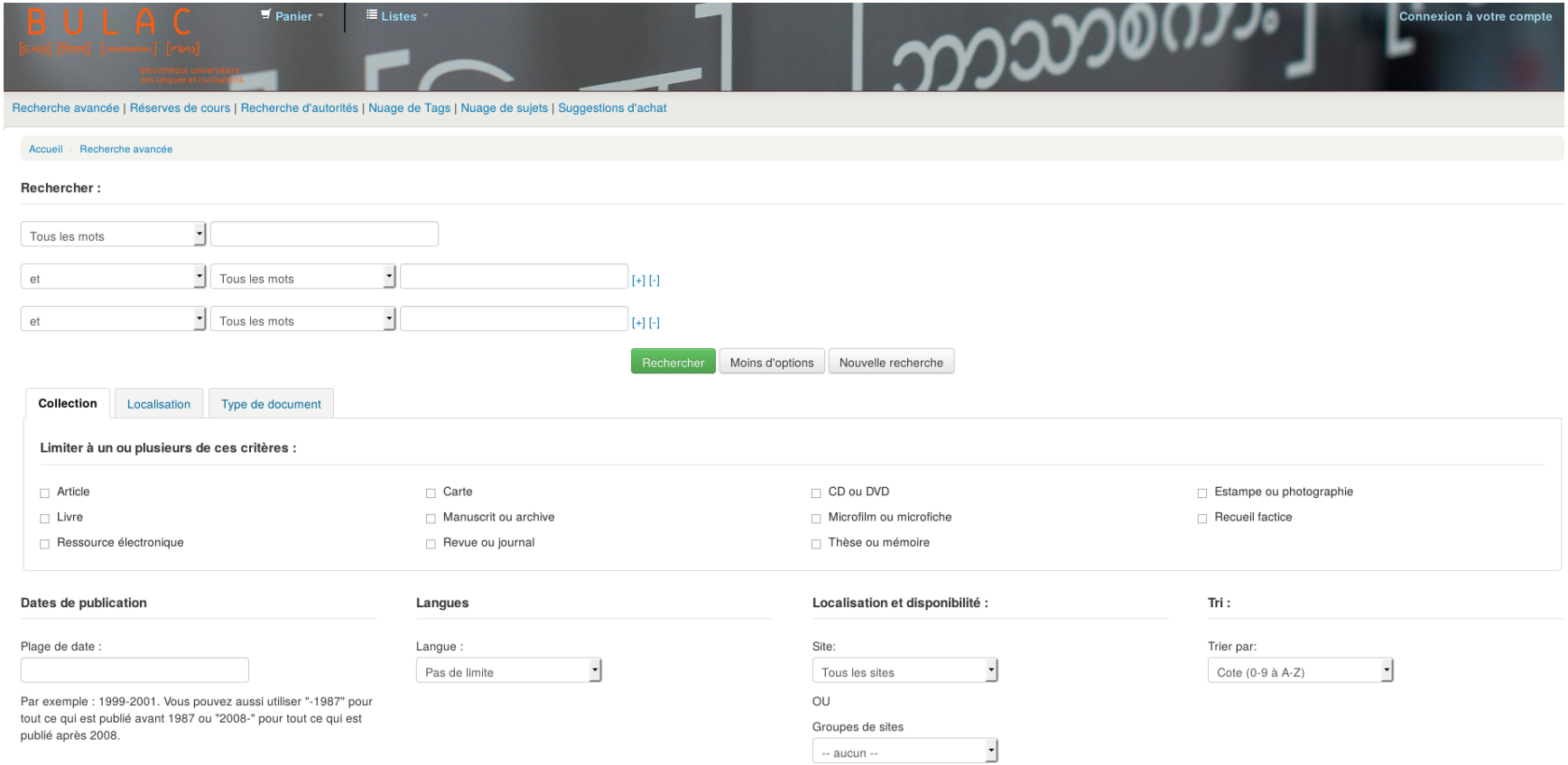


Biblissima

Une base de données **n'est pas une interface applicative, Web ou logiciel**.

Qu'est-ce qu'une base de données ?

Est-ce une base de données ?



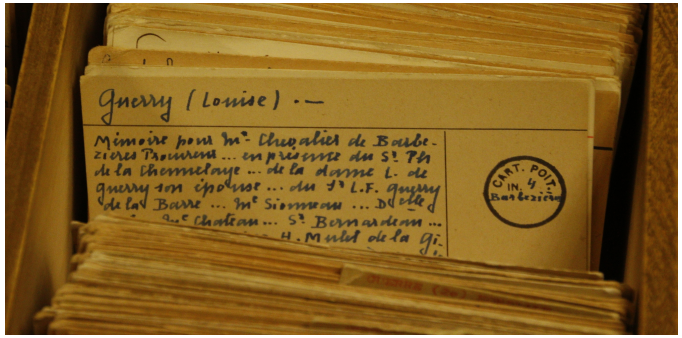
Catalogue de la Bulac

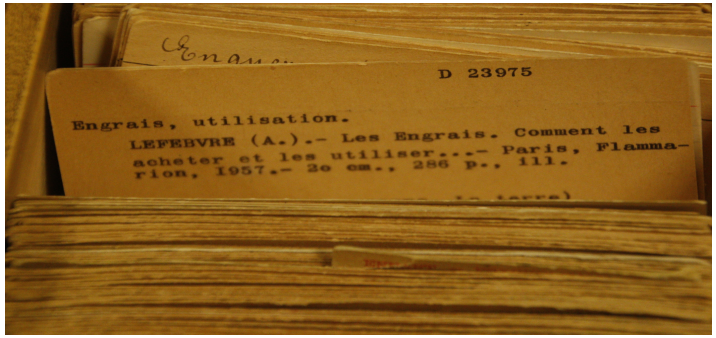
Une base de données n'est pas non plus **un système d'information** ou **système *intégré* de gestion de bibliothèque**

Qu'est-ce qu'une base de données ?

Est-ce une base de données ?









Meuble et fiches issus de la Médiathèque François Mittérand de Poitiers
Prise de vue par Wilfrid NIOBET

C'est une base de données ! Nous avons dans ce meuble des données brutes sur des fiches regroupées dans des tiroirs.

BY-NC-SA

Qu'est-ce qu'une base de données ?

Le TLFI, pas plus que le dictionnaire de l'Académie française ne définissent l'expression *base de données*. Cependant à l'entrée **DONNÉE** nous avons, entre autre, la définition suivante :

> *Spéc. „Ensemble des indications enregistrées en machine pour permettre l'analyse et ou la recherche automatique des informations (Cros-Gardin 1964). Banque de données ; données documentaires, données lexicales.*

Définition de référence pour cette présentation :

> ***Une base de données est une collection organisée de données.***

Sylvie Damy, 3e année de licence informatique, Université de Besançon (2012/2013)

Cette définition à la qualité d'être simple et ne se limite pas au domaine informatique. Aussi, le classement de notice bibliographique sous forme de fiches cartonnées peut aussi être considéré comme base de données.

Qu'est-ce qu'une base de données ?

En informatique, une base de données s'associe à des outils spécifiques qui permettent d'organiser et de structurer les données comme un **Système de gestion de base de données** (SGBD), logiciel qui gère et permet l'accès aux données de la base.

Généralement une base de données est associée à une **Application de base de données**, programme qui interagit avec la base de données à certains moments de son exécution en lui adressant une requête (en général dans le langage SQL). Ces programmes peuvent être développés dans un langage informatique : *php*, *Java*, *Delphi*, *Python*, etc. Actuellement ces programmes s'inscrivent de plus en plus comme des applications Web.

Nos environnements en informatique documentaire s'inscrivent dans un **Système d'information** qui prends le nom de **Système intégré de gestion de bibliothèques** (SIGB) définissant les applications métiers comme Primo, Alma, OCLC, Koha, PMB, etc. Il s'agit de l'ensemble formé par la base de données, le SGBD et les applications de base de données liées à cette base de données.

BY-NC-SA

Shématisation d'une base de données

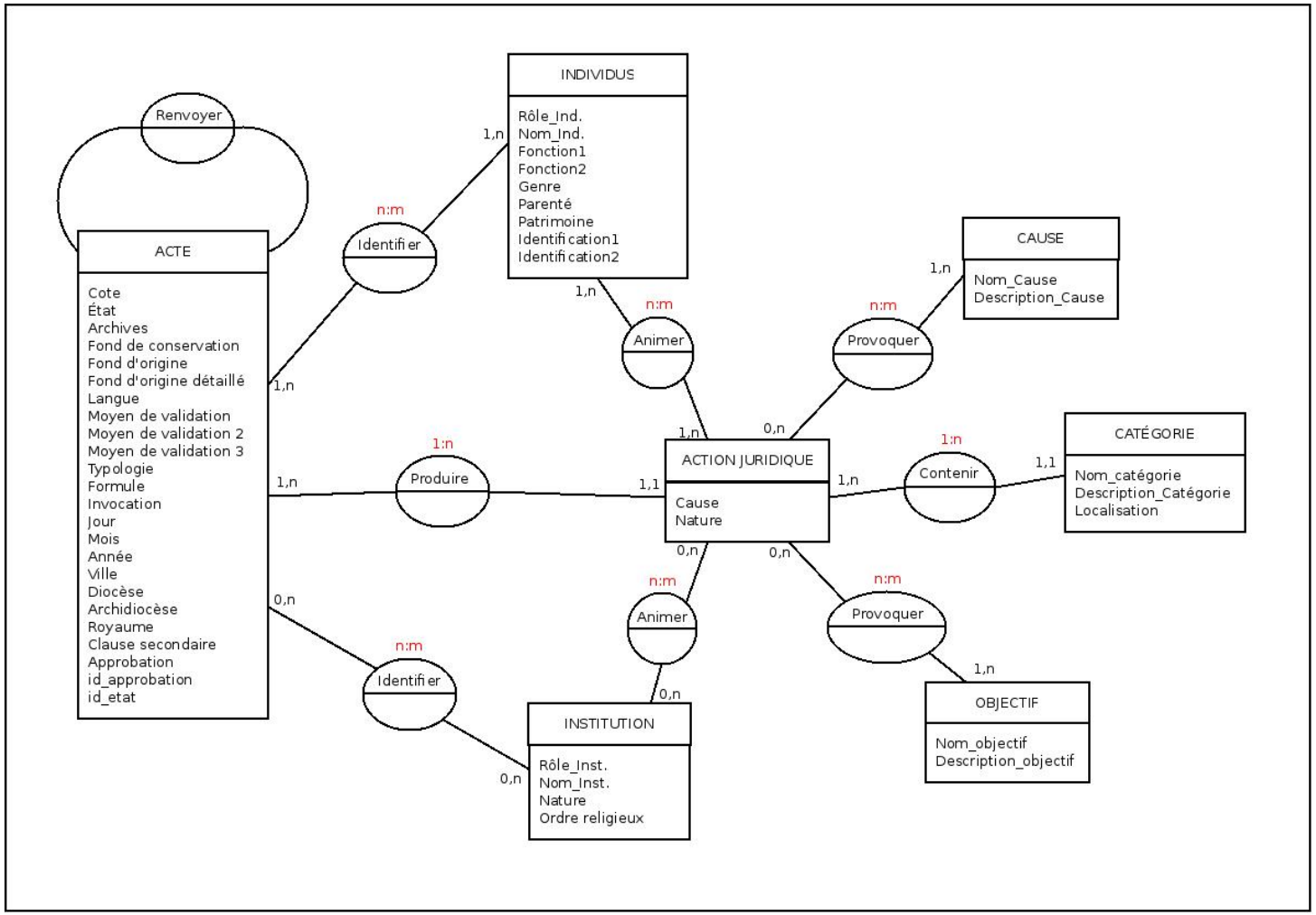


Schéma extrait d'un projet de thèse

SGBD - présentation 1

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| bibperso           |
| bibperso2          |
| charlotte          |
| charlotte_test     |
| eni                |
| koha_kohadb        |
| mysql              |
| omeka              |
| performance_schema |
| phpmyadmin         |
| pmb                |
| sarah1             |
| test_temp          |
| wordpress          |
+--------------------+
15 rows in set (0.01 sec)
```

```
mysql> use sarah1
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+------------------+
| Tables_in_sarah1 |
+------------------+
| acte             |
| acte_ind         |
| acte_inst        |
| action_juridique |
| aj_cause         |
| aj_ind           |
| aj_inst          |
| aj_obj           |
| archive01        |
| categorie        |
| cause            |
| dt_class_tables  |
| dt_requete       |
| dt_typo_table    |
| individu         |
| institution      |
| objectif         |
+------------------+
17 rows in set (0.00 sec)
```

Schéma extrait d'un projet de thèse

SGBD - présentation 2

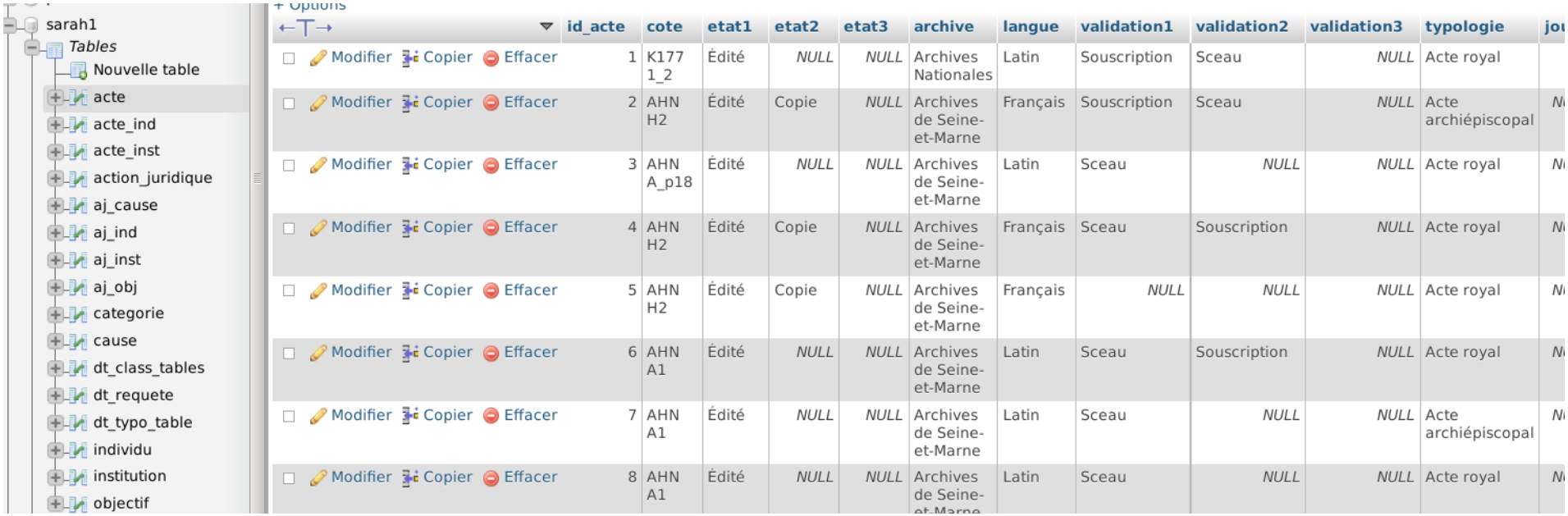
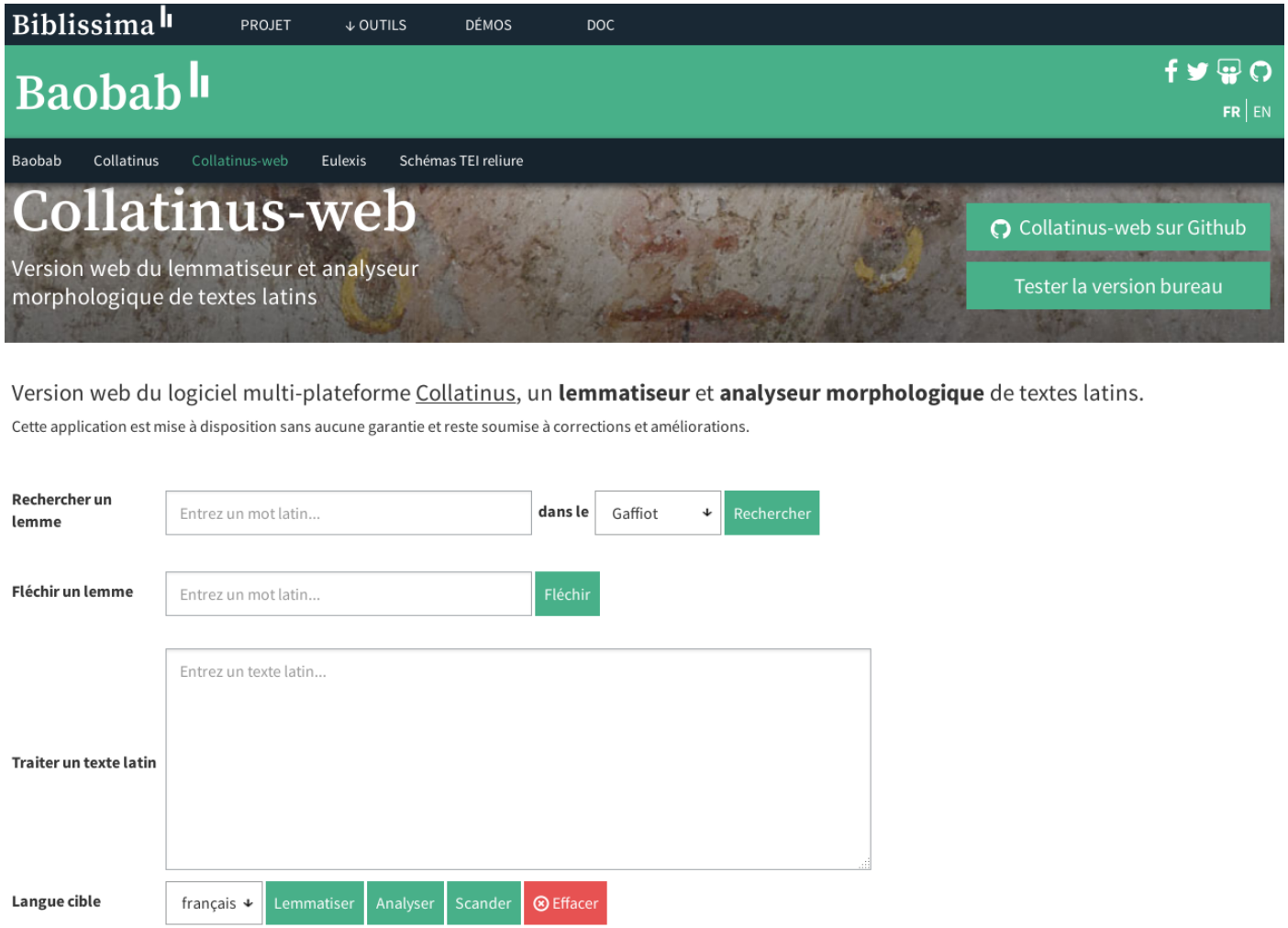


Schéma extrait d'un projet de thèse

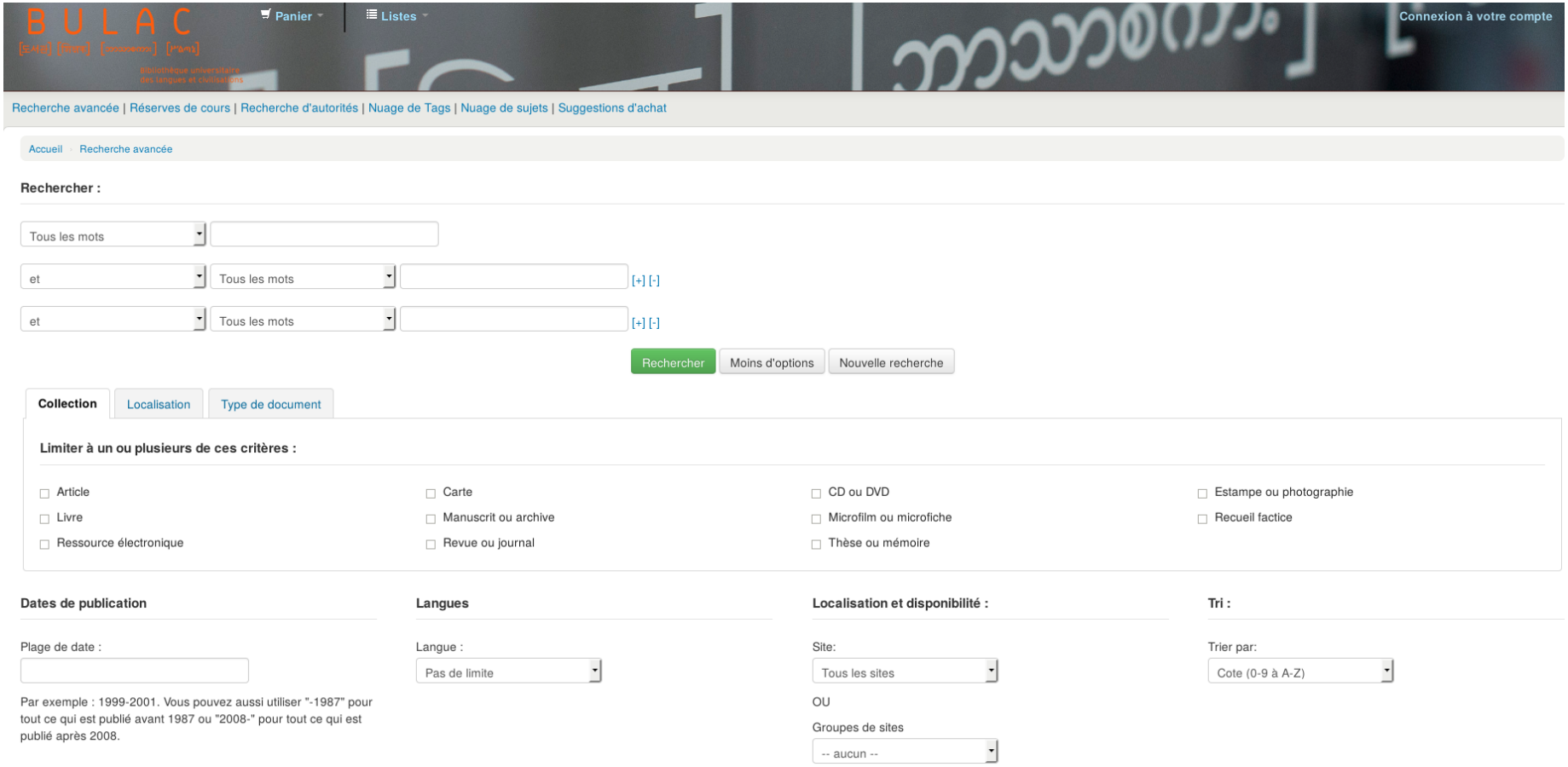
(cc) BY-NC-SA

Une application Web



Capture d'écran de l'application Web Baobab sur le site Web Biblissima

Système d'information ou SIGB



Capture d'écran du SIGB de la biliothèque Bulac

Les raisons d'une base de données

De manière générale une base de données répond aux exigences suivantes :

- Stocke l'intégralité des données brutes ;
- Centralise les données (élimine les redondances) ;
- Distingue les données et les traitements ;
- Structure des données complexes ;
- Partage des données ;
- Respecte l'intégrité et la cohérence des données ;
- Veille à la confidentialité des données ;
- Sécurise l'information ;

BY-NC-SA

Qu'est-ce qu'une données brutes ?

La dénomination de *données brutes* est particulièrement subjective et relative à nos besoins. Si idéalement nous pourrions considérer une donnée brute comme une donnée objective, c'est-à-dire exempte de jugement ou d'interprétation, elle ne peux être exclue de son environnement. De telles données s'inscriraient aussi dans un consensus, tel que peuvent l'être les données décritent dans un thésaurus universel.

Or nous savons que d'une base de données à une autre l'organisation des données ne se fonde pas sur un tel consensus. Les données brutes s'organisent en fonction d'un besoin applicatif.

La définition que nous pouvons proposer :

> ***une donnée brute*** *est une* ***donnée unitaire*** *qui répond à un besoin applicatif et structurel.*

Elle est unitaire car elle constitue une unité de traitement et se caractérise par sa globalité et sa totalité.

Qu'est-ce qu'une données brutes ?

Par exemple :

- Selon les besoins d'un système d'information, une adresse postale pourra être prise dans sa globalité et formerait une unité :
  {10 rue Victor Hugo 86000 POITIERS}
  ou de manière divisé où chaque partie de l'adresse correspond à une donnée propre :
  Le numéro de rue : {10}, le nom de la rue {rue Victor Hugo},
  le code postal {86000} et la ville {POITIERS}
- Le titre d'un document constitut une unité et il est traité dans sa globalité (zone 200 $a) ;
- Le nom et prénom d'un auteur sont non seulement distingués l'un de l'autre (zone 700 $a et $b), et la place des auteurs selon qu'il est l'auteur principale seul ou en collaboration avec d'autres auteurs, secondaire ou qu'il s'agisse d'une collectivité (zone 700, 701, 702, 710, etc...). Nous remarquons que la structuration des données est relativement fine quand au "acteur" dans la réalisation d'un document.

BY-NC-SA

Pourquoi parle-t-on d'une base de données "relationnelle" ?

Qu'est-ce que cette **relation** ? Qu'est-ce qui la caractérise ?

> *On appelle « relation » un ensemble d'attributs qui caractérisent une proposition ou une combinaison de propositions comme "un employé a un matricule, il a un nom, il a un employeur". Dans cet exemple, les attributs de l'employé sont : son matricule, son nom et son employeur. Chaque combinaison de propositions ainsi formée est appelée tuple ou collection ordonnée d'objets. Par exemple l'ensemble ("1245", "Jean Dupond", "Compagnie des belles lettres") constitue un tuple de relation "employé". Les relations sont d'ordinaire représentées sous la forme de tables. Dans l'exemple précédent, la table serait libellée "employé". Usuellement, les praticiens accordent la même signification aux concepts de "relation" et de "table". De même, ils assimilent d'une part la "ligne dans la table" et le tuple, et d'autre part le "libellé de colonne dans la table" et l'attribut. Par définition, chaque tuple d'une relation est unique. Il est identifié par un attribut (un identifiant unique appelé "clef primaire") ou une combinaison de plusieurs attributs qui forme la clef. L'ordre des tuples n'est pas significatif.*

Wikipedia, art. Modèle relationnel

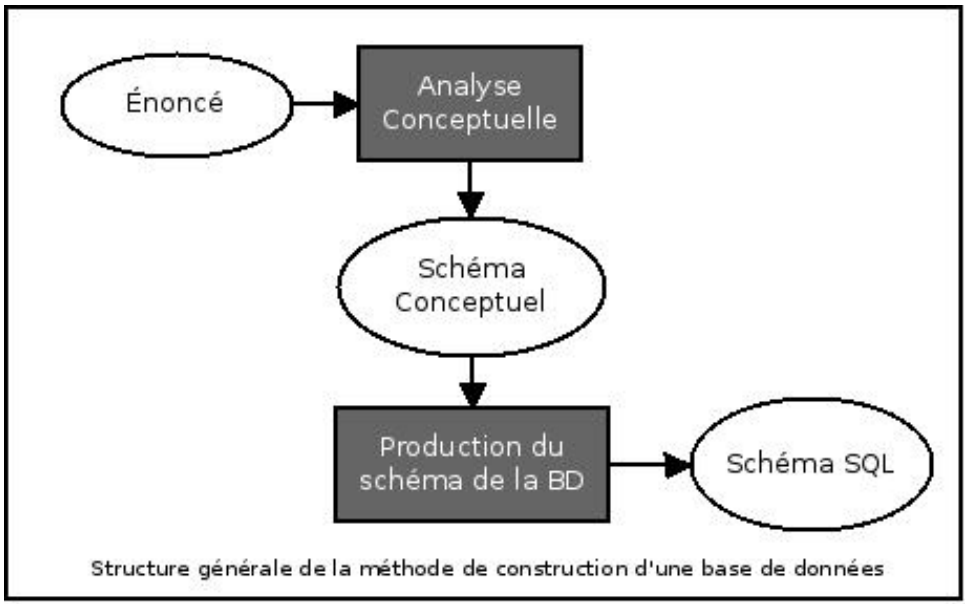
 BY-NC-SA

Organisation des données

Pour qu'une base de données puisse répondre à ses objectifs, par conséquent aux objectifs du projet dans lequel elle s'inscrit, il est nécessaire au préalable d'organiser et de structurer les données. Pour cela il existe différentes méthodes qui dans le fond reviennent toutes aux éléments de base.

Voici unes des méthodes utilisées pour la conception d'une base de données :

- le modèle Entité-association de la base ;
- la méthode MERISE dans laquelle s'inclus le modèle Entité-association ;



Structure générale de la méthode de construction d'une base de données

BY-NC-SA

UML, un langage de modélisation

La schématisation est un outil essentiel dans la conception d'une base de données. Elle permet de visualiser la structure de la base et la dynamique engagée entre les données.

Si dans un meuble bibliographique nous pouvons nous rendre compte facilement du rangement des données et de leur structure en consultant une fiche, il en est tout autrement lorsque nous consultons une notice bibliographique informatisée.

C'est pourquoi nous avons besoin d'outil pour comprendre comment sont organisées et structurées les données d'une base. Cette compréhension pourra nous être utile pour dépasser l'environnement applicatif défini par le SI/SIGB et adapter nos besoins aux besoins de l'établissement documentaire. Il pourra être important de préciser l'accès à ces éléments de conception dans un cahier des charges, même si dans l'établissement en question il n'y a pas encore de personne compétente pour manipuler les données brutes d'une base de données au moment de la rédaction.

BY-NC-SA

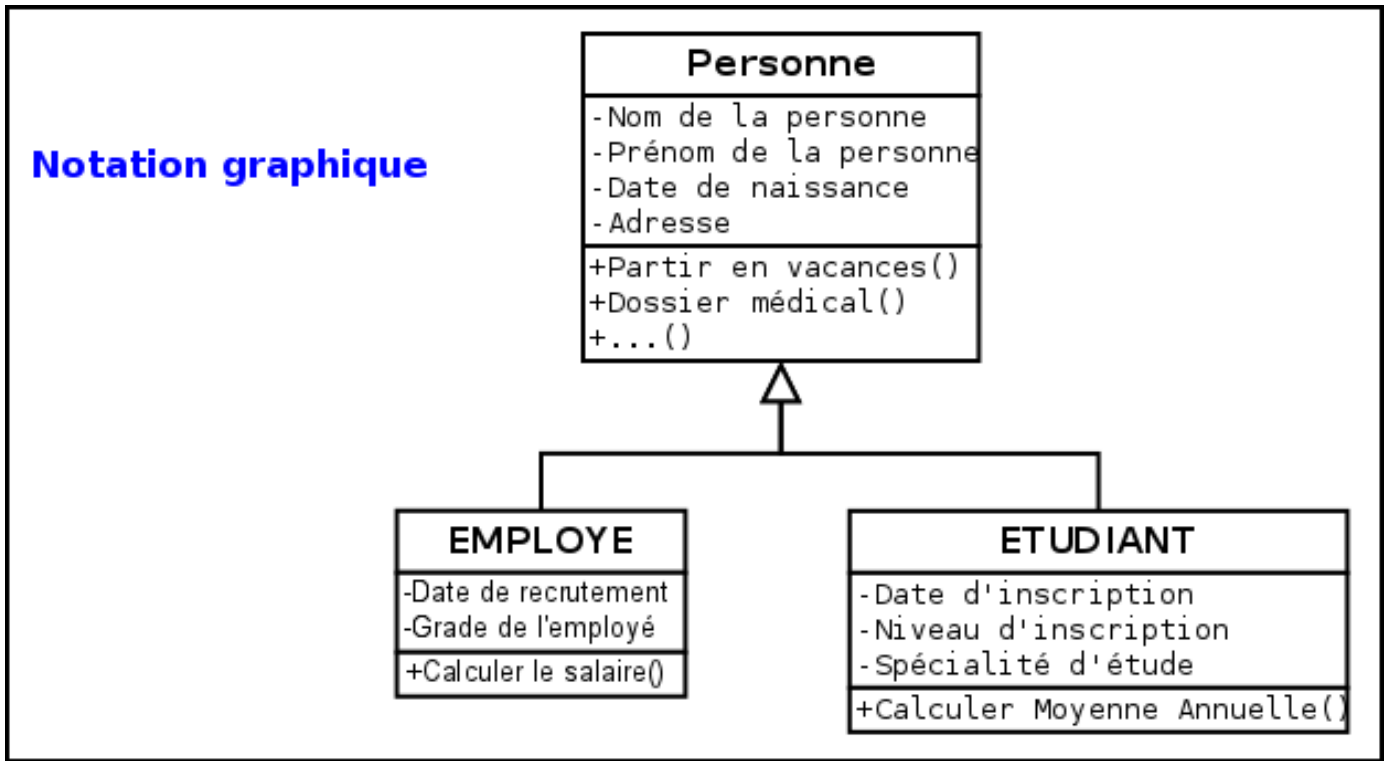
Les diagrammes d'UML
(Unified Modeling Language)

- *UML est un langage de modélisation orienté objet qui sert à représenter les composantes d'un SI par un ensemble de diagrammes ;*
- *Un modèle est une représentation simplifiée d'un problème ;*
- *Il y a 14 types de diagrammes qui s'organisent en deux grandes catégories : diagrammes de structure et statiques, diagrammes de comportement ;*
- *Chaque diagramme répond à un besoin spécifique de représentation.*

 BY-NC-SA

Les diagrammes d'UML
(Unified Modeling Language)

Voici un diagramme présenté dans le cours UML
où vous y avez vue une normalisation graphique commune.



Les diagrammes de structure

Ces types de diagramme décrivent l'aspect statique du système.

Vous en avez étudié 6 autours des objectifs suivant :

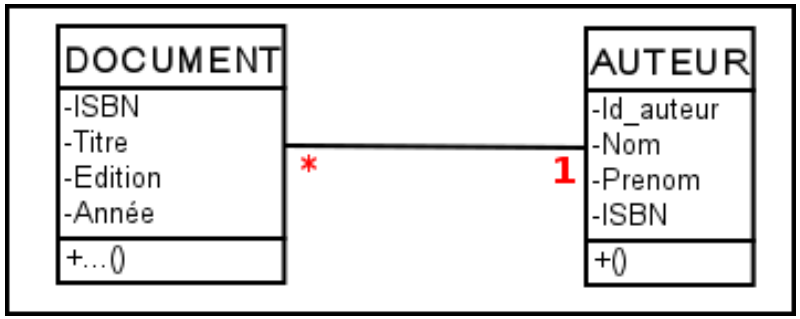
- Découvrir les différents diagrammes de structure.
- Comprendre les concepts relatifs à chaque type de diagramme.
- Apprendre à utiliser les différents diagrammes.

Ce qui caractérise ces diagrammes se sont les relations statiques qu'ils mettent en forme.

(cc) BY-NC-SA

Les diagrammes de structure

Pouvez vous expliquer la relation entre DOCUMENT est AUTEUR ?



Un document est écrit par un auteur.
Un auteur à écrit au moins un document.

Les diagrammes de comportement

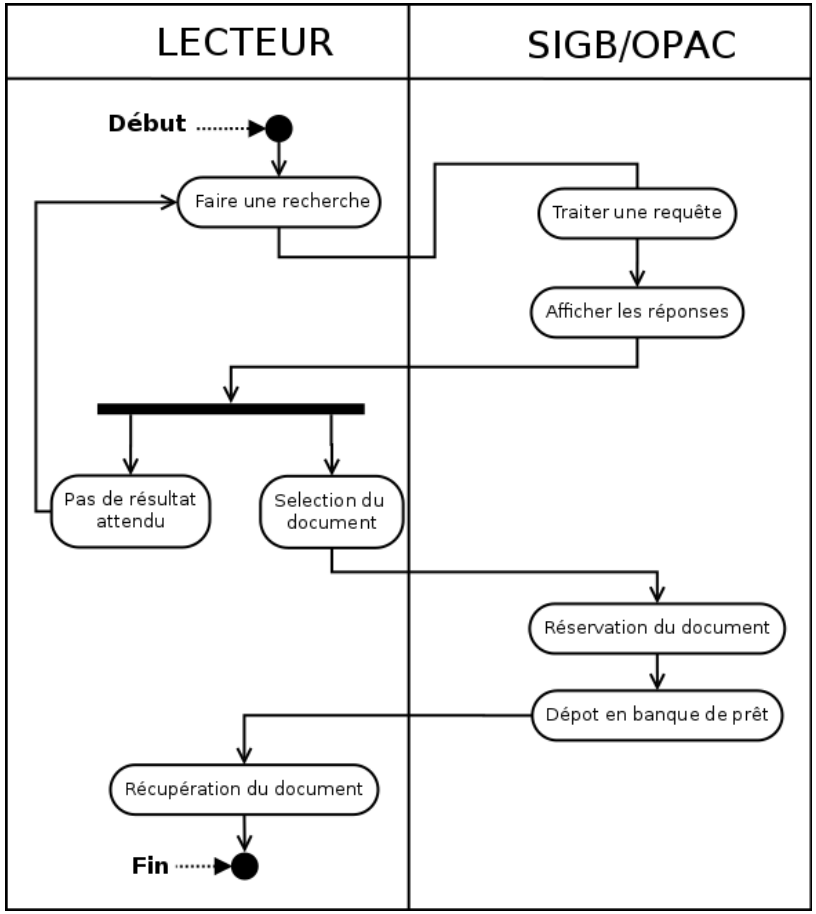
Ces types de diagramme présentent le fonctionnement et la dynamique d'un système. Les objectifs du cours d'UML est de :

- Découvrir les différents diagrammes décrivant le comportement.
- Découvrir et comprendre les concepts relatifs à chaque type de diagramme.
- Apprendre à utiliser les différents diagrammes et
- Connaître les avantages et les inconvénients de chaque type de diagramme.

Ils constituent une technique qui permet de déterminer les besoins des utilisateurs et de capturer les exigences fonctionnelles d'un système. En d'autres termes, ils décrivent le comportement d'un système du point de vue de ses utilisateurs. Ils décrivent les interactions entre les utilisateurs d'un système et le système lui-même.

Les diagrammes de comportement

Voici un exemple graphique qui permet de visualiser les différentes actions pour une demande de document.



Que doit-on formaliser ?

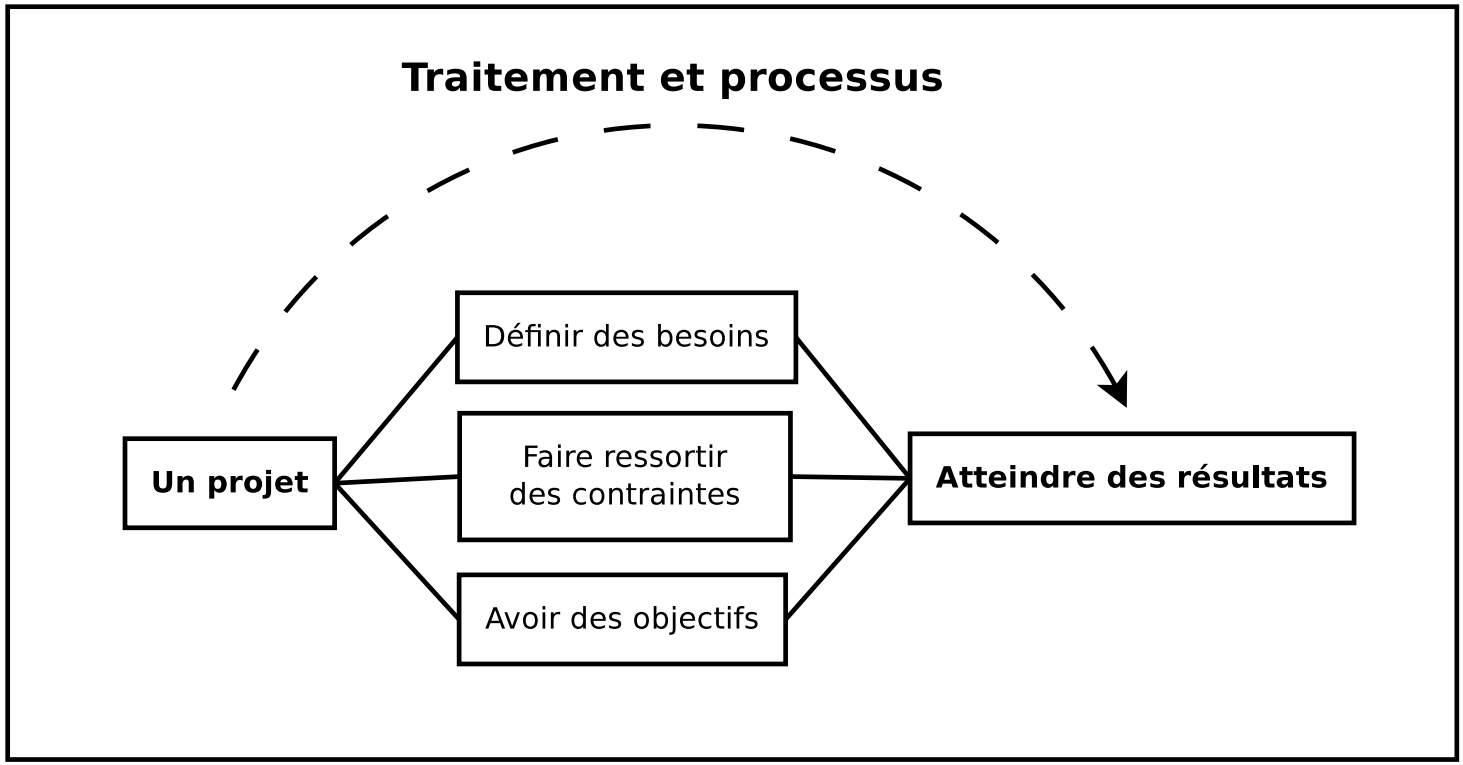
Dans le contexte de bases de données voici les questions que nous devons nous poser. Que doit-on formaliser ?

- De l'information, Quelle(s) information(s) ?
- Des procédures, Quelle(s) procédures ?
- Des relations, Quelle(s) relation(s) ?
- Des résultats, Quel(s) résultat(s) ?

BY-NC-SA

Un projet

Voici le parcours que nous allons faire...

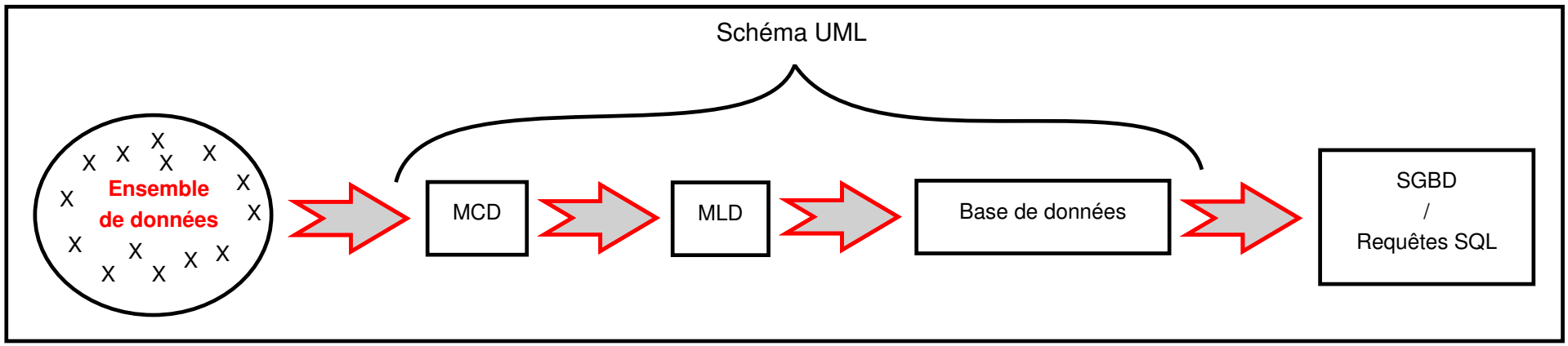


Un projet s'inscrit dans un processus (voire plusieurs) et dans le traitement d'informations (résultats).

Un projet

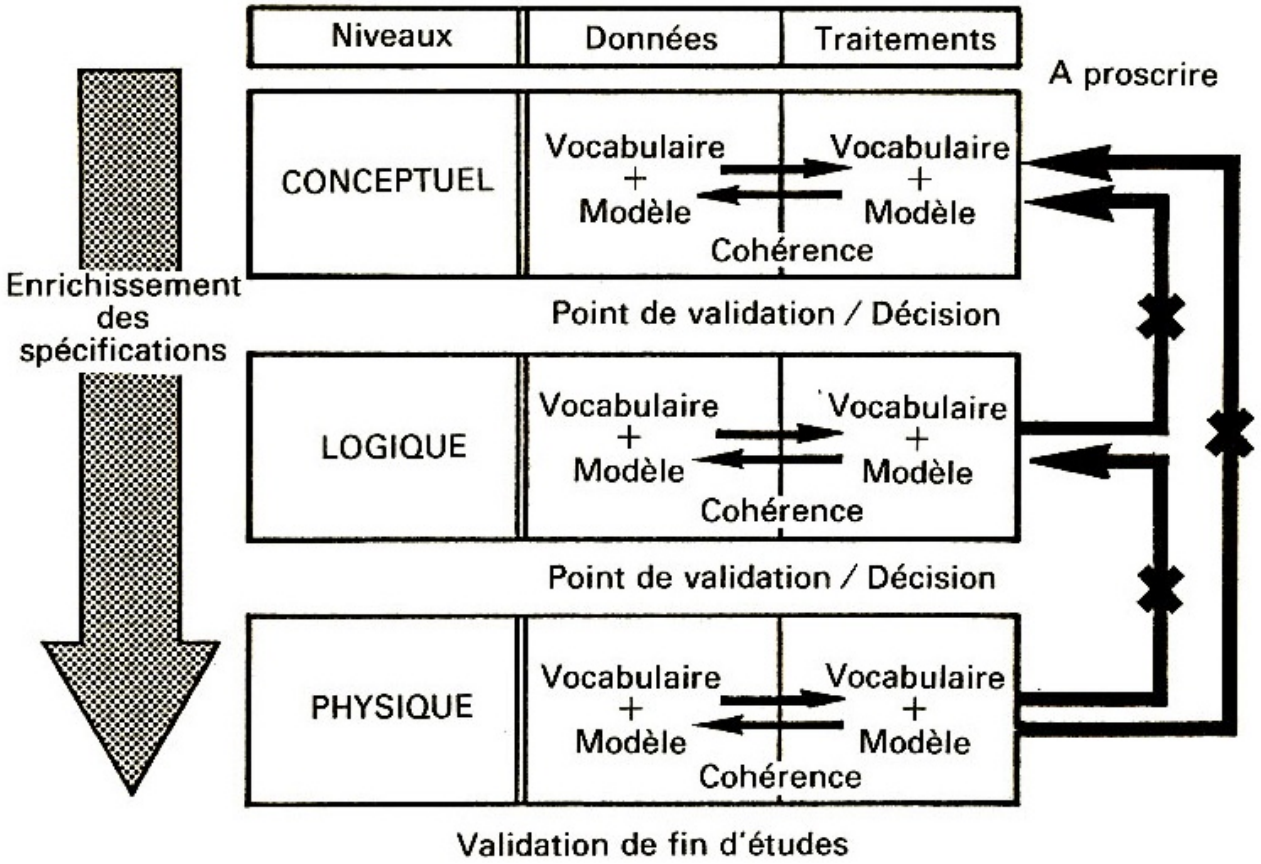
Afin de pouvoir définir les données pour le traitement d'informations, il nous faut suivre l'information depuis sa source jusqu'à son traitement. C'est ce que nous appellerons le déroulement du projet.

Ce déroulement est déterminant pour récolter les données nécessaires au projet. Les données sont définies comme à la base de l'information. Rappellons-nous qu'une base de données est le lieu où l'on stocke des données "brutes". Une information c'est déjà des données interprétées (résultat).



Méthode MERISE

La méthode MERISE s'organise sur trois niveaux : Conceptuel, Logique et Physique. Chacun d'eux analyse les données séparément de leur traitement comme le montre le schéma ci-dessous.



La méthode MERISE préconise d'analyser séparément données et traitements, à chaque niveau.

Méthode MERISE

Méthode d'Étude et de Réalisation Informatique pour les Systèmes d'Entreprise.

> *Issue de l'analyse systémique, la méthode MERISE est le résultat des travaux menés par René Coletti, Arnold Rochfeld et Hubert Tardieu dans les années 1970.*
>
> *MERISE sur Wikipédia*

Cette méthode prend un projet dans son ensemble, projet dans lequel il y a ou non une base de données en fonction des besoins. Par conséquent tous les points de cette méthode ne sont pas utiles pour la conception d'une base de données. Elle permet néanmoins d'avoir une vue d'ensemble profitable à sa réalisation.

Pour nos besoins nous n'utiliserons que le **MCD** (Modèle Conceptuel de Données) et le **MLD** (Modèle Logique de Données), afin de concevoir une base de données relationnel.

Nous laisserons le MPD et les traitements des différents niveaux, ils s'inscrivent dans un projet plus large, celui d'un système d'information.

MCD

Le Modèle Conceptuel de Données (MCD) permet d'organiser l'ensemble des données récoltées selon un schéma **entités-associations**.

Mettons nous en contexte et prenons l'exemple de notre bibliothèque personnelle. Considérons que nous voulions mettre en place un inventaire de nos livres (documents imprimés). Quelles sont les données dont nous avons besoins pour construire cet inventaire ?

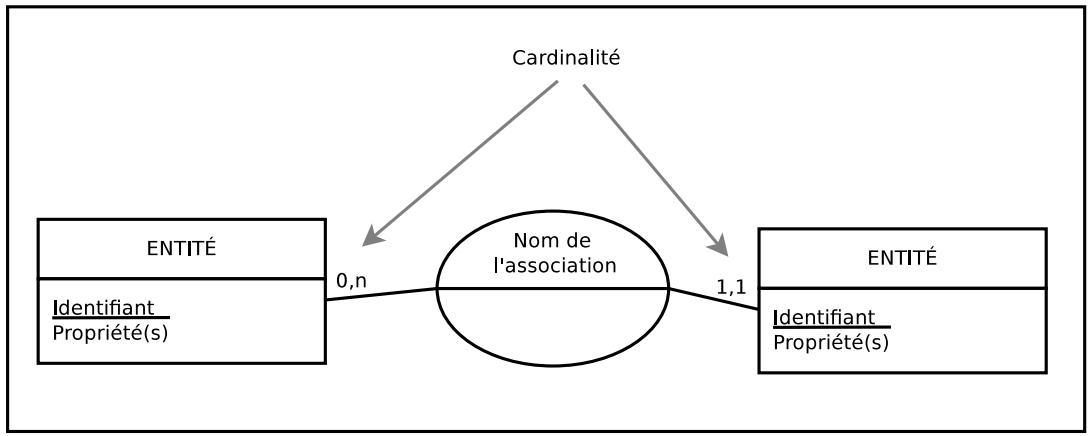
Voici par exemple un ensemble de données sous la forme d'un tableau :

Titre	Auteur	Editeur	Collection	Annee
La fabuleuse histoire du papyrus d'Artémidore	Canfora, Luciano	Anacharsis	Série Philologie	2014
La Morale d'Epicure, avec des reflexions	Épicure	Thomas Guillain		1685
The society of mind	Minsky, Marvin	Simon & Schuster		1988
Traité des animaux	Condillac, Étienne Bonnot de	Ch. Ant. Jombert		1766

Est-ce qu'il manque des données ? Comment vérifier qu'il ne nous en manque pas ?

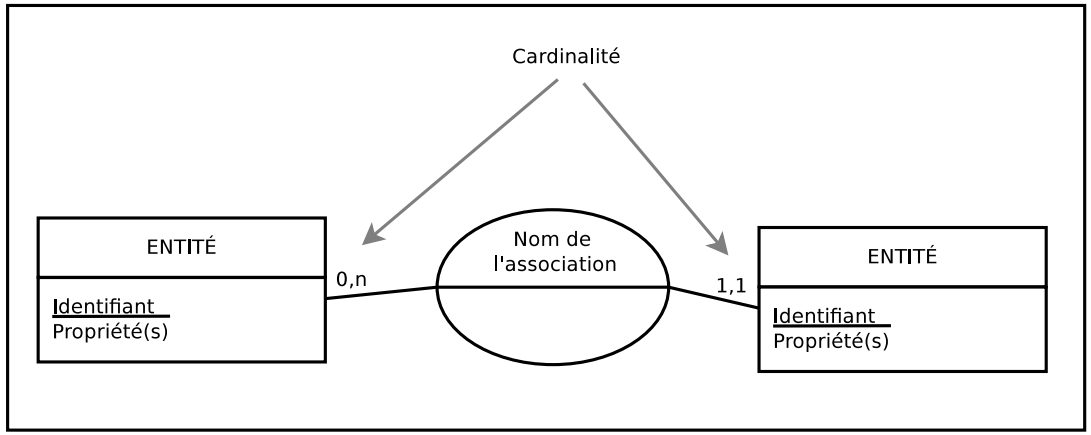
Une fois que l'on a une liste exhaustive des données dont on a besoin, il faut organiser ces données en fonction de :

- Entité : *Un "objet" pourvue d'une existence propre.*
- Association : *Relation entre **entités**, c'est un lien sémantique indépendant de tout traitement. Il est caractérisé par un verbe ou un substantif.*
- Propriété : *Plus petit élément d'information qui a un sens en lui-même et qui décrit une entité ou une association.*
- Identifiant : *Élément unique d'une entité qui permet de définir une seule occurence.*
- Occurence : *Ensemble des valeurs que peut prendre une propriété.*
- Cardinalité : *Indique le nombre minimum et le nombre maximum d'occurences de l'association pour une occurence de l'entité.*



MCD - Exercice

Vous avez tous les éléments pour pouvoir construire un schéma **Entité-Association**.

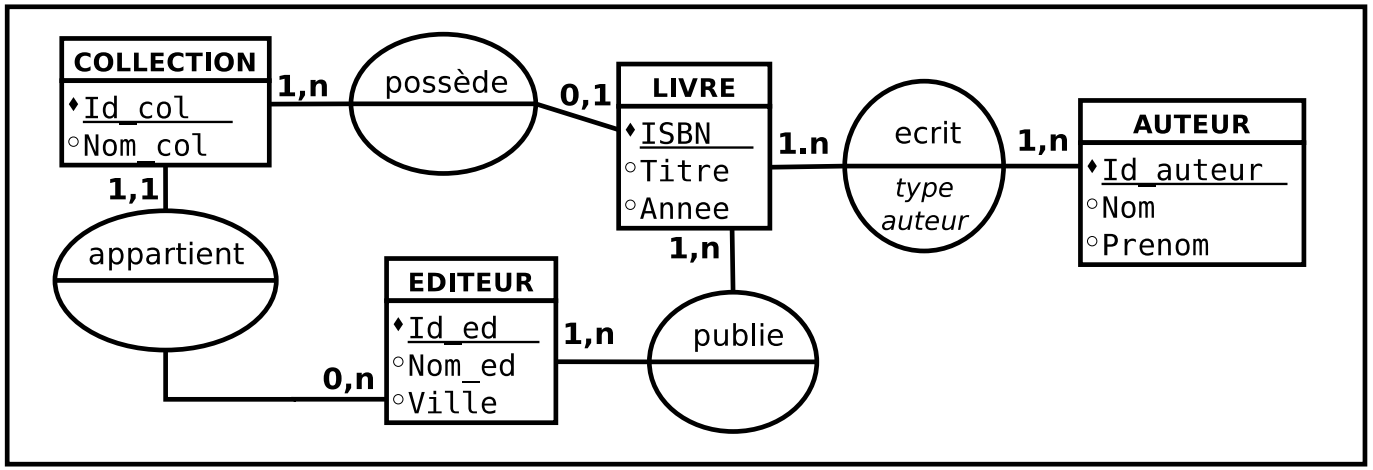


Une entité associe une et une seule entité. [cardinalité *1,1*]

Une entité est associée plusieurs fois ou n'est pas associée. [cardinatlité *0,n*]

30 minutes

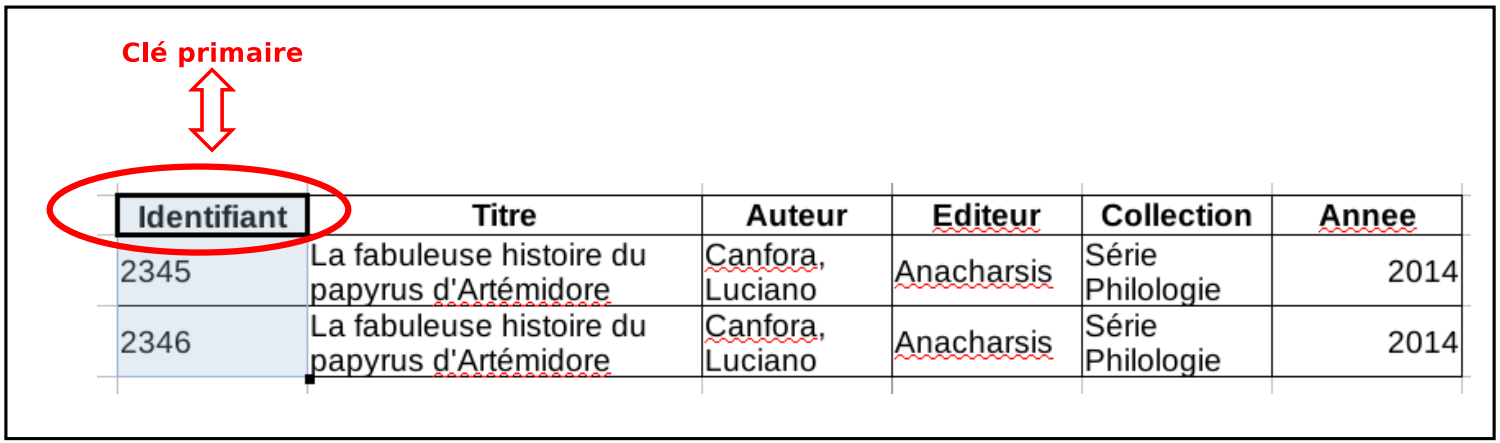
MCD - Exemple



- Un livre n'appartient pas nécessairement à une collection. Par contre, il ne peut appartenir qu'à une seule collection.
- Un livre est écrit au moins par un auteur ou un anonyme.
- Un livre est publié par au moins un éditeurs.
- Un auteur écrit au moins un livre.
- Un éditeur publie au moins un livre.
- Un éditeur ne crée pas nécessairement une collection et il peut en créer plusieurs.
- Une collection dépend d'un seul éditeur.
- Une collection contient au moins un livre.

Clé primaire

Maintenant que nous avons notre MCD, nous pouvons le transcrire en différents systèmes logiques comme celui du Modèle Logique de Données Relationnel que l'on trouve dans les SGBD. Nous avons vue ce qu'était un **identifiant** dans le schéma **entité-association**. Pour maintenir l'intégrité des données dans une organisation globale des données, il est nécessaire que chaque entité et les occurences qui lui sont attachées puissent être unique.

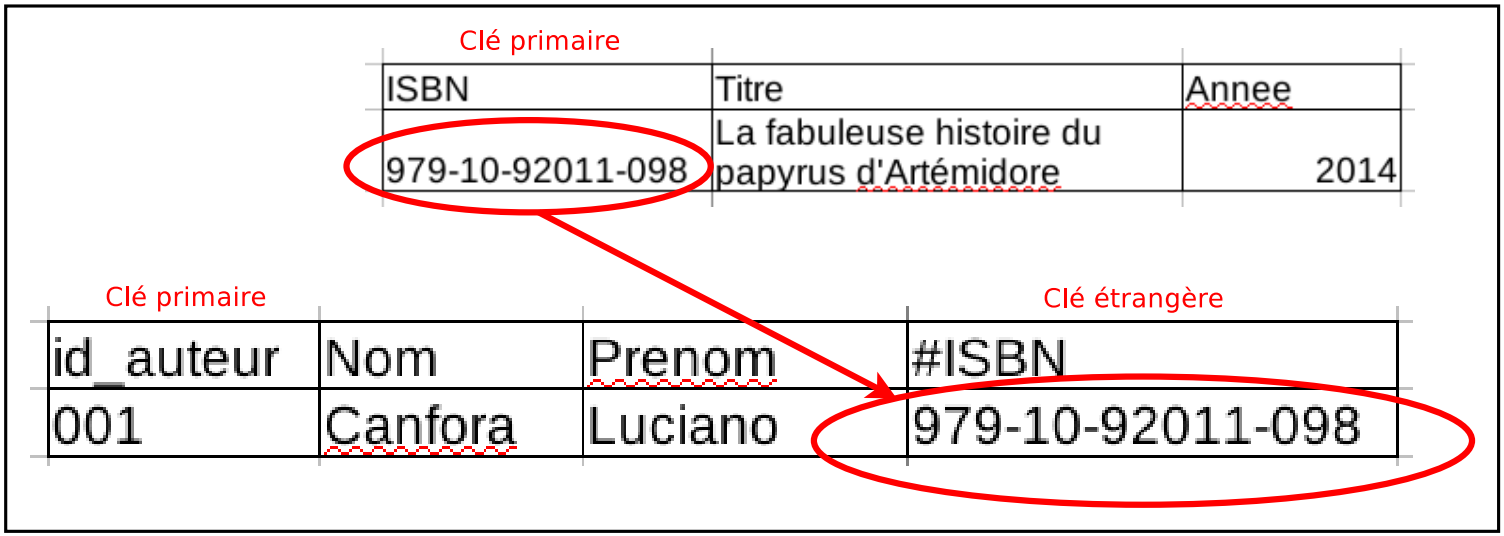


Identifiant	Titre	Auteur	Editeur	Collection	Annee
2345	La fabuleuse histoire du papyrus d'Artémidore	Canfora, Luciano	Anacharsis	Série Philologie	2014
2346	La fabuleuse histoire du papyrus d'Artémidore	Canfora, Luciano	Anacharsis	Série Philologie	2014

Une clé primaire est un **identifiant unique** définit pour chaque ligne d'**occurences**. Cela permet de maintenir l'intégrité des données.

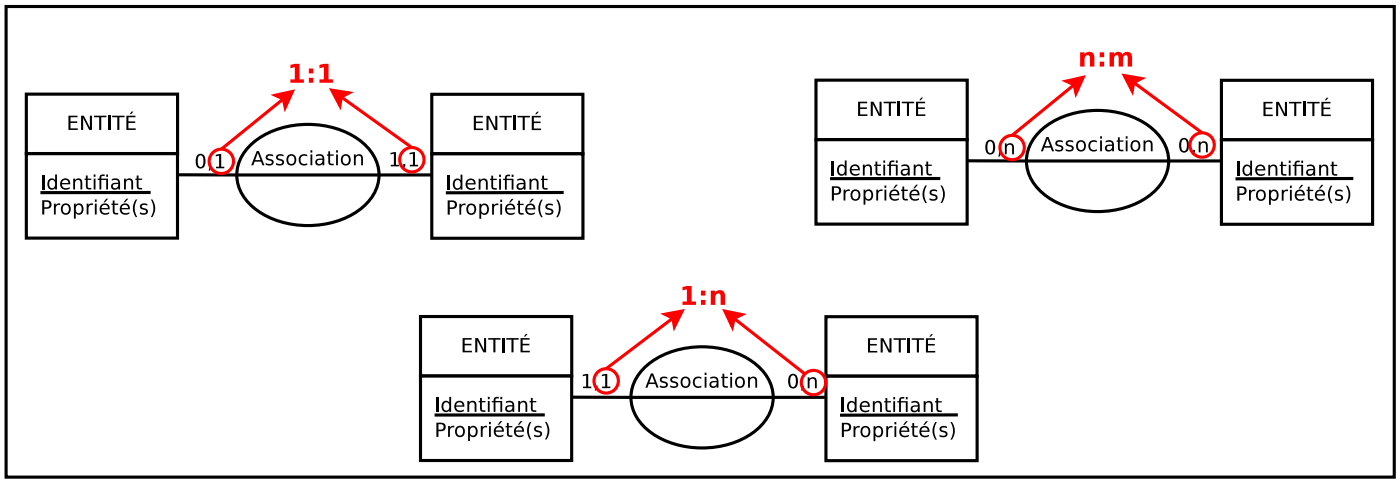
Clé étrangère

Une #clé étrangère reporte la clé primaire d'une autre entité afin de maintenir une correspondance entre les entités de manière à éviter les doublons : une clé étrangère a pour référence une clé primaire.



Notation d'une association binaire

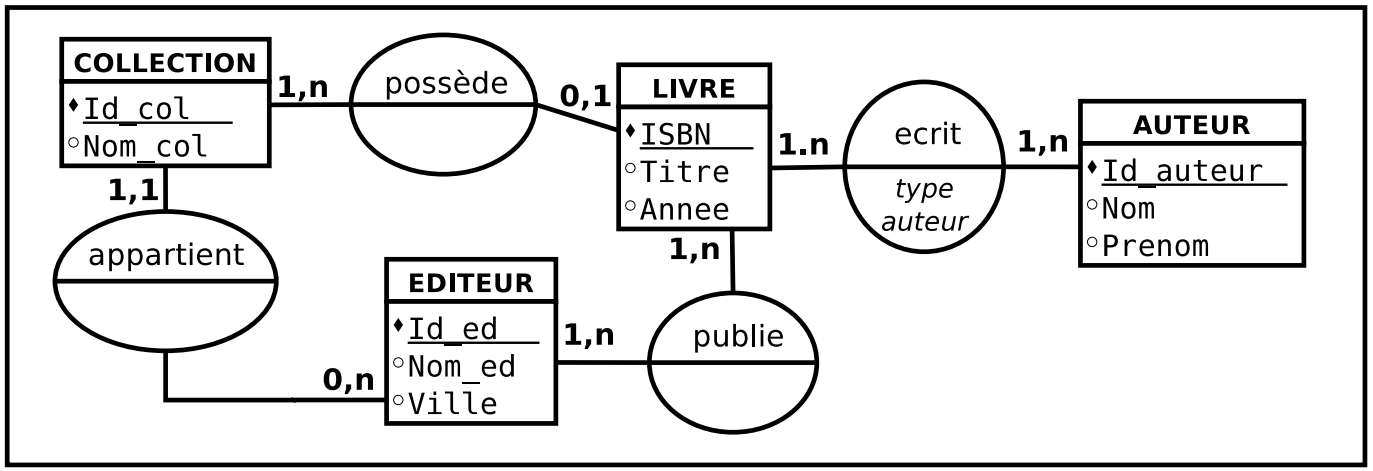
Une association binaire se contruit à partir de deux valeurs cardinales maximales.
Notations : on dit qu'une association binaire (entre deux entités ou réflexive) est de type :
– **1 :1** (un à un) si aucune des deux cardinalités maximales n'est *n* ;
– **1 :n** (un à plusieurs) si une des deux cardinalités maximales est *n* ;
– **n :m** (plusieurs à plusieurs) si les deux cardinalités maximales sont *n*.



En fait, un schéma relationnel ne peut faire la différence entre *0,n* et *1,n*. Par contre, il peut la faire entre *0,1* et *1,1*. Issue de cette notation, 5 règles vous nous permettre de passer du MCD au MLD.

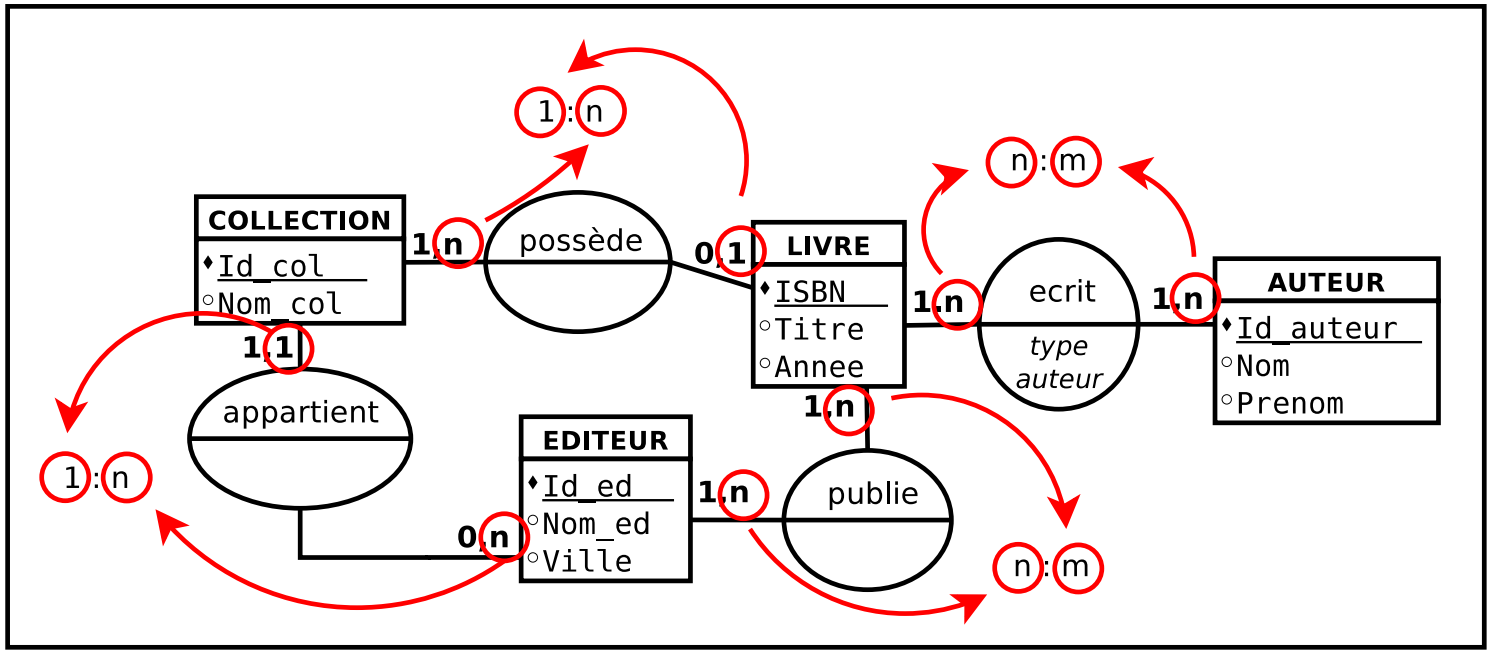
Association binaire - Exercice

Vous avez tous les éléments pour pouvoir construire les associations binaires de notre bibliographie à partir de notre schéma **Entité-Association**.



10 minutes

Présentation de l'association binaire de notre bibliographie



Passage d'un MCD vers un MLD

Maintenant que nous avons élaboré un Modèle Conceptuel de Données (MCD), nous pouvons transcrire ce modèle dans différents systèmes logiques. Nous allons élaborer le Modèle Logique de Données (MLD) et l'écriture logique qui s'y rapporte avec l'aide de cinq règles à notre disposition. Cela nous permettra ensuite la création de la base de données dans un SGBD.

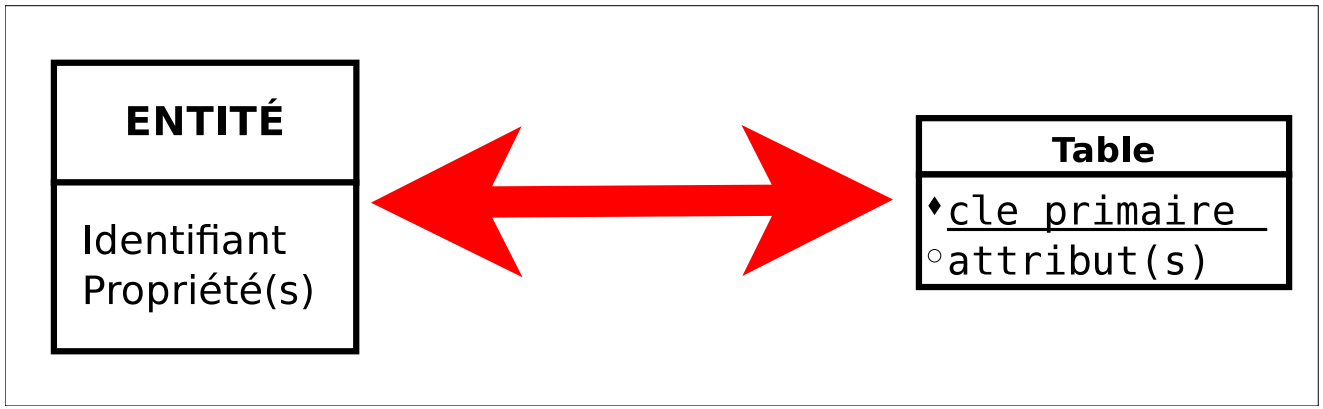
Le MCD doit être exhaustif, c'est-à-dire qu'il doit contenir toutes les informations nécessaires à la réalisation du projet. Cette étape est le fondement du projet, sa qualité et son évolution en dépendent. C'est aussi la partie la plus délicate car elle formalise tout le sens du projet.

Le MLD transforme :

- les **entités** en **table** ;
- les **identifiants** en **clefs primaires** ;
- les **propriétés** en **attributs**.

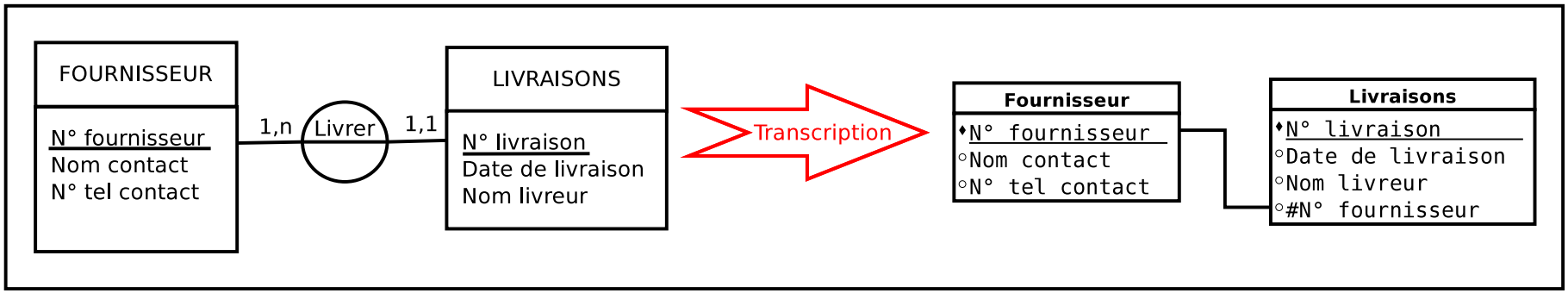
Règle 1

Règle 1 : Toute **entité** devient une **table** (*une relation*) dans laquelle les attributs deviennent les colonnes. L'identifiant de l'entité constitue alors la clé primaire de la table.



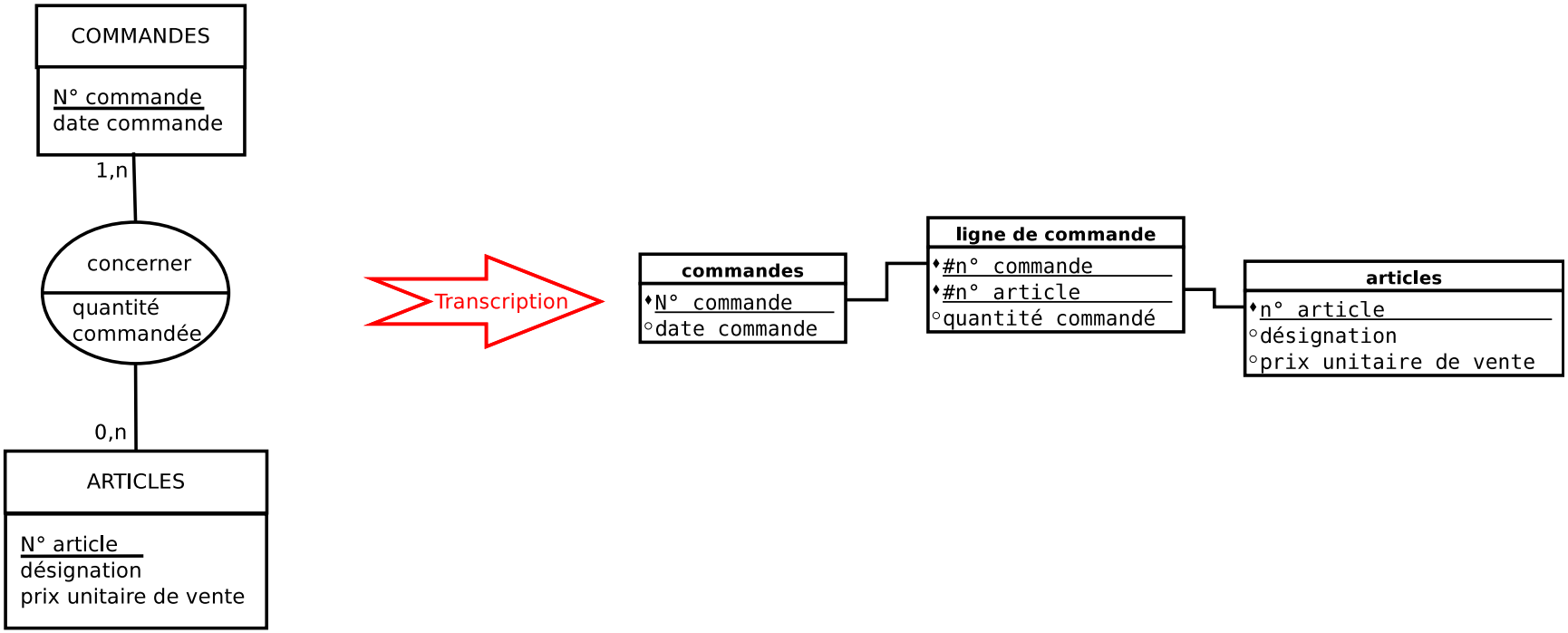
Règle 2

Règle 2 : une association binaire de type **1 :n** disparaît, au profit d'une clé étrangère dans la table côté *0,1* ou *1,1* qui référence la clé primaire de l'autre table. Cette clé étrangère ne peut pas recevoir la valeur vide si la cardinalité est *1,1*.



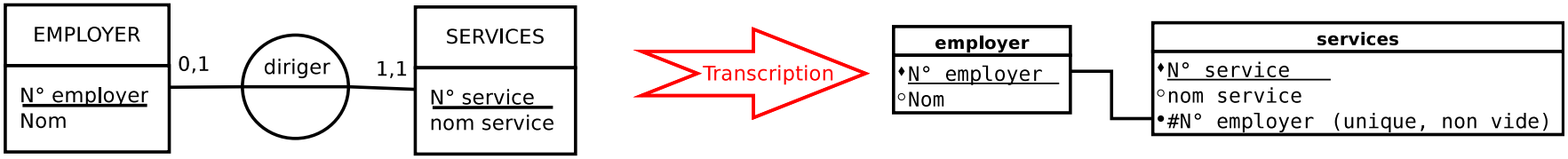
Règle 3

Règle 3 : une association binaire de type **n :m** devient une table supplémentaire (parfois appelée table de jonction, table de jointure ou table d'association) dont la clé primaire est composée de deux clés étrangères (qui référencent les deux clés primaires des deux tables en association). Les attributs de l'association deviennent des colonnes de cette nouvelle table.

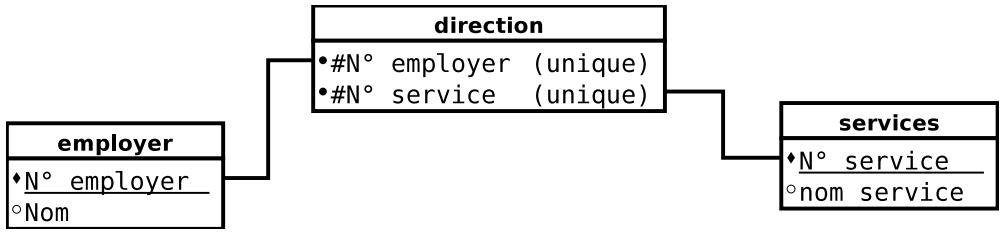


Règle 4

Régle 4 : une association binaire de type **1 :1** est traduite comme une association binaire de type **1 :n** sauf que la clé étrangère se voit imposer une contrainte d'unicité en plus d'une éventuelle contrainte de non vacuité (cette contrainte d'unicité impose à la colonne correspondante de ne prendre que des valeurs distinctes).

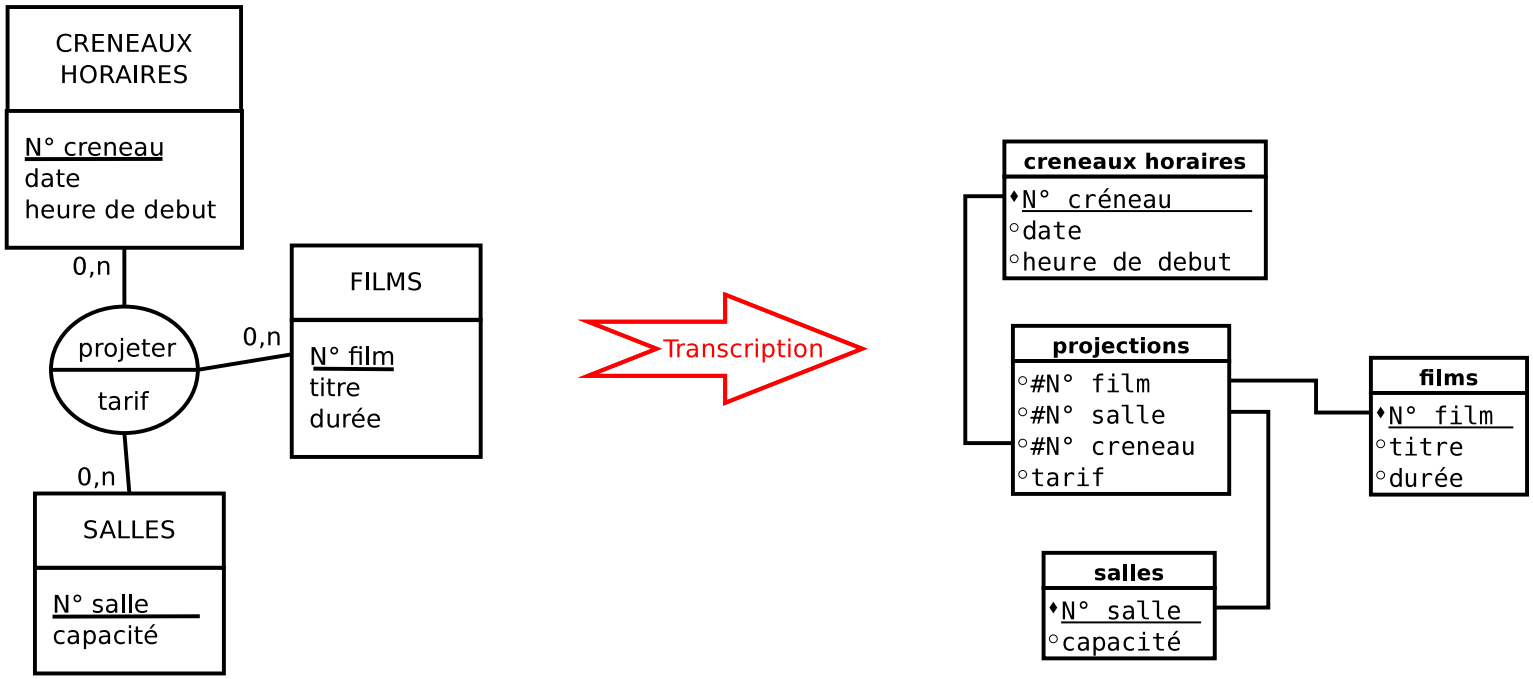


Une alternative consiste à voir une association binaire de type **1 :1** comme une association binaire de type **n :m** particulière. Il suffit pour cela d'ajouter une contrainte d'unicité sur chacune des clés étrangères de la table de jonction supplémentaire.



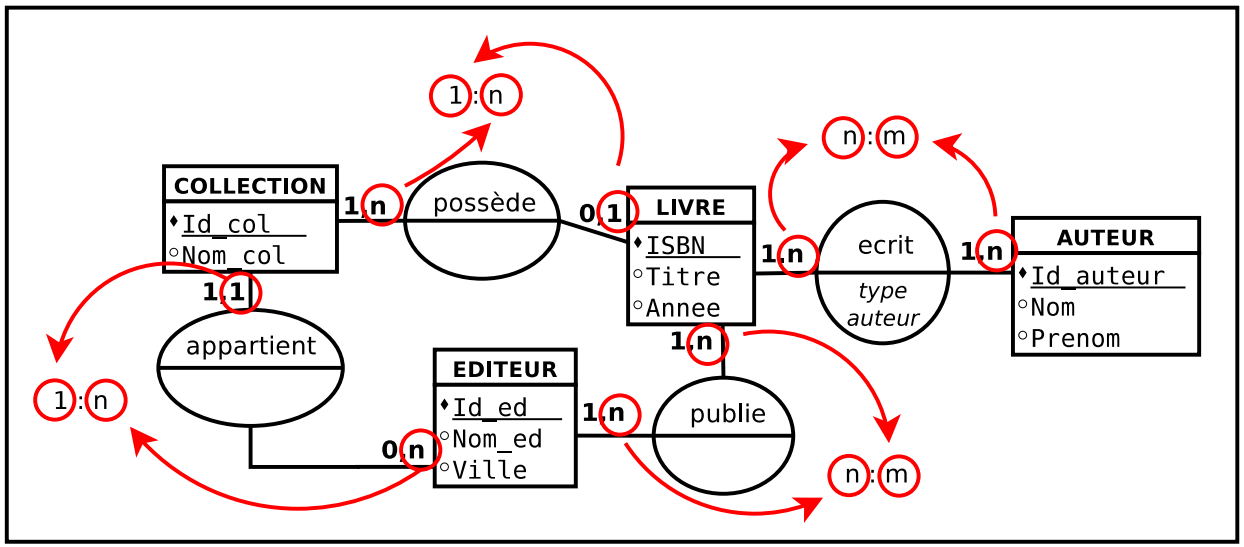
Règle 5

Règle 5 : une association non binaire est traduite par une table supplémentaire dont la clé primaire est composée d'autant de clés étrangères que d'entités en association. Les attributs de l'association deviennent des colonnes de cette nouvelle table.



MLD - Exercice

À partir des 5 règles que vous venez de découvrir,
essayez de réaliser un MLD à partir du MCD que vous avez réalisé.



20 minutes

MLD - Exemple bibliographique

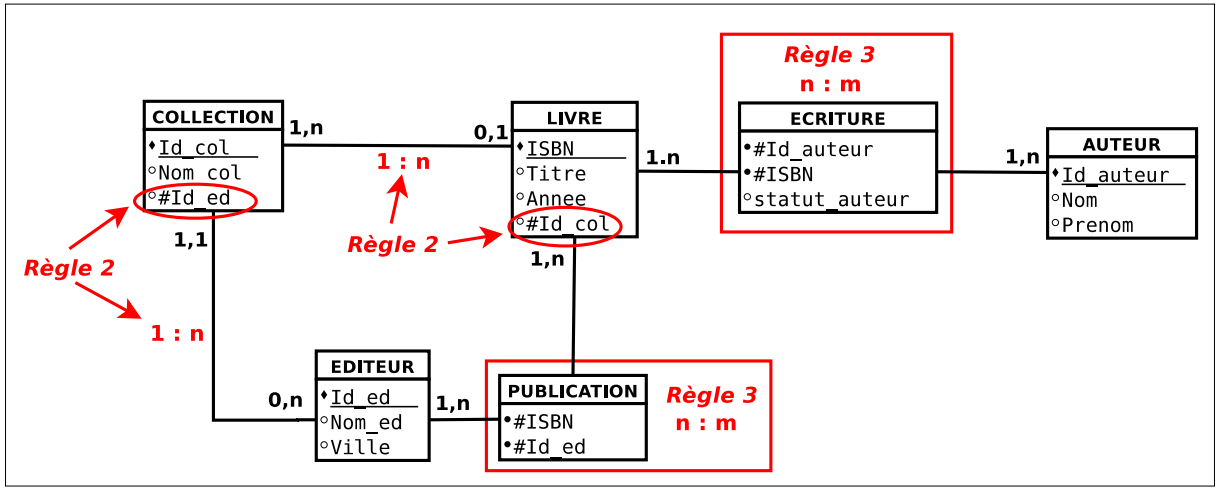


Schéma décomposant l'évolution de la transposition du MCD vers le MLD

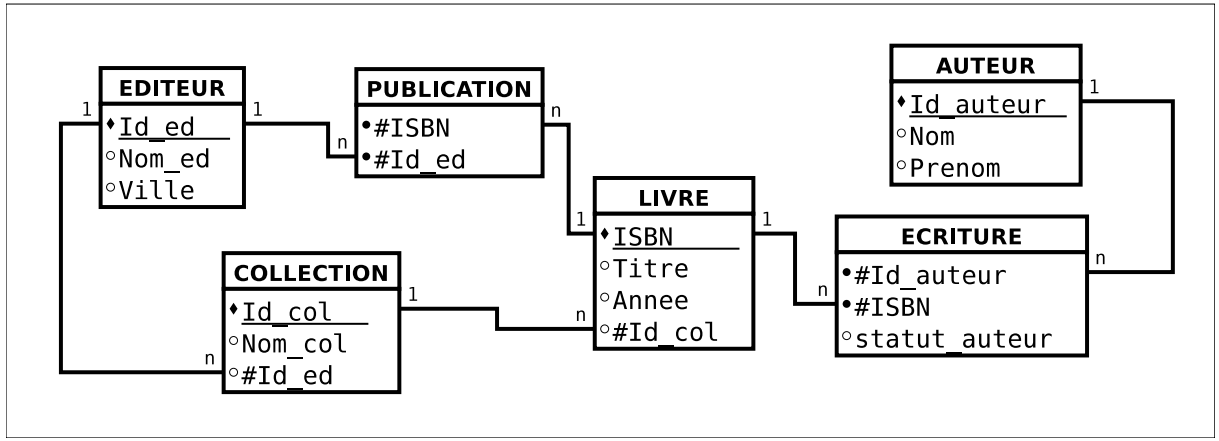


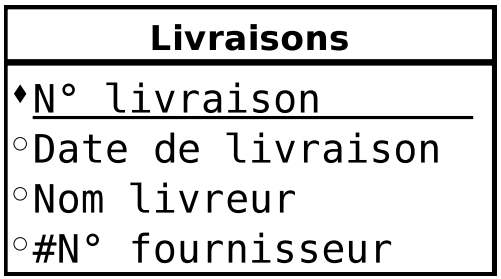
Schéma MLD

Écriture logique

L'écriture logique est le dernier point qu'il faut aborder avant la création concrète d'une base de données relationnelles avec le langage SQL. L'**écriture logique** permet de transposer le schéma en écriture formelle. Cette formalisation pourra en suite facilement être transposer en langage SQL.

- une clé primaire est soulignée,
- une clé étrangère est précédée d'un #,
- la table se trouve en tête de liste,
- les propriétés ou attributs se trouve entre parenthèses

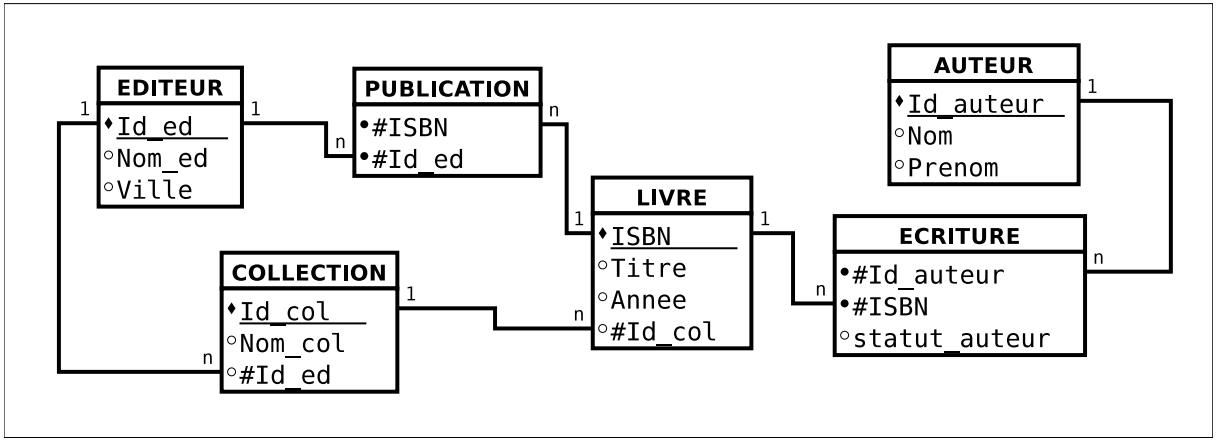
Pour le schéma suivant :



Nous devrions l'écrire logiquement de la manière suivante :

Livraisons(N°livraison,date de livraison, Nom livreur, #N°fournisseur)

Écriture logique - Exercice



À partir du schéma MLD que vous avez réalisé, transformez ce schéma en écriture logique.

15 minutes

Présentation de l'écriture logique de nos entités

LIVRE (ISBN, Titre, Annee, #Id_col)

ECRITURE (#Id_auteur [non vide], #Id_livre [non vide], statut_auteur)

AUTEUR (Id_auteur, Nom, Prenom)

EDITEUR (Id_ed, Nom_ed, Ville)

PUBLICATION (#ISBN [non vide], #Id_ed [non vide])

COLLECTION (Id_col, Nom_col, #Id_ed [non vide])

BY-NC-SA

Base de données

Tout ce travail préparatoire nous a permis de préciser notre projet et de le formaliser afin de pouvoir le transcrire informatiquement. Il nous reste à passer de l'écriture logique au langage SQL pour la création de la base de données.
Une fois intégré dans un SGBD, il nous faudra enregistrer les données dans la base de données et intéroger la base afin de répondre aux objectifs attendus du projet. Si le langage commun des SGBD est le langage SQL, chacun d'eux ont pris la liberté d'adopter quelques variantes dans le langage SQL.

SQL : Structured Query Language

Le langage SQL se divise en trois grandes catégories appellé aussi *sous-langage* : **DDL**, **DML** et **DCL**.

- **DDL**, *Data Definition Language* : permet de créer, supprimer et de modifier une table, un domaine, une colonne ou une contrainte. Il est constitué de commandes de définition et de modification de structures.
- **DML**, *Data Manipulation Language* : permet d'extraction et la modification des données.
- **DCL**, *Data Control Language* : permet de contrôler l'accès aux données d'une base de données.

BY-NC-SA

SQL-DDL

Voici les commandes SQL-DDL

CREATE : création de structure de données ;

ALTER : modification de structure de données ;

DROP : suppression de structure de données ;

RENAME : renommage de structure de données.

Quelques options à ces commandes répondants aux besoins structuraux de la base de données :

DATABASE : base de données (Uniquement avec CREATE et DROP) ;

TABLE : table ;

INDEX : indice ;

VIEW : table virtuelle ;

SEQUENCE : suite de nombres ;

SYNONYM : synonyme ;

USER : utilisateur.

Exemple de la base ’bibperso’

Avant de créer nos tables il est nécessaire de créer la base de données.

Dans notre exemple nous avons choisi d’utiliser MariaDB comme SGBD. MariaDB est une solution sous les licences GNU GPLv2 et GNU LGPL. Il s’agit d’un Fork de MySQL qui lui appartient à l’entreprise Oracle et qui n’est plus sous licence libre, bien que toujours gratuit.

Dans le langage SQL nous venons de voir les commandes et les options qui nous permettent de construire une base de données dans un SGBD.

Création de la base de données *bibperso* :

```
CREATE DATABASE IF NOT EXISTS ‘bibperso‘ DEFAULT CHARACTER SET utf8 COLLATE
utf8_unicode_ci ;
```

 BY-NC-SA

Reprennons l'écriture logique que nous avons réalisé depuis notre analyse des informations nécessaires à notre projet :

LIVRE (ISBN, Titre, Annee, #Id_col)

Nous allons maintenant transcrire cette écriture logique dans le langage SQL pour créer notre table 'livre'.

```
USE 'bibperso' ; (Entrer dans la base de donnée)
CREATE TABLE IF NOT EXISTS 'livre' (
      'ISBN' int(13) NOT NULL,
      'titre' varchar(50) COLLATE utf8_unicode_ci DEFAULT NULL,
      'annee' year(4) DEFAULT NULL,
      'id_col' int(11) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci ;
ALTER TABLE 'livre'
  ADD PRIMARY KEY ('ISBN'), ADD UNIQUE KEY 'id_col' ('id_col') ;
  ADD CONSTRAINT 'collection' FOREIGN KEY ('id_col') REFERENCES 'collection' ('id_col') ON
DELETE CASCADE ON UPDATE CASCADE ;
COMMIT ;
```

} Nous retrouvons les éléments de l'écriture logique

SQL-DML

Voici les principales commandes SQL-DML :

SELECT (extraction)
INSERT (ajoute)
UPDATE (modifie)
DELETE (supprime)

Clauses principales :

from *(indique la table ou les tables)*
where *(specifie les conditions)*
and *(ajoute une condition)*

Il existe d'autres clauses pour préciser les requêtes, comme le booleen, expression logique ou de calcul. Une requête peut être très généraliste ou au contraire très précise. On trouve une requête soit dans le SGBD choisi, soit dans le programme qui permet l'interfaçage avec le SGBD (PHP, Java, Python, C, etc.). En gros, pratiquement tous les langages de programmation peuvent interroger un SGBD.

 BY-NC-SA

Exemple SQL-DML : INSERT

```
INSERT INTO ‘livre‘(‘ISBN‘, ‘titre‘, ‘annee‘, ‘id_col‘) VALUES (”,"La fabuleuse histoire du papyrus d’Artémidore",’2014’,2) ;
INSERT INTO ‘livre‘(‘ISBN‘, ‘titre‘, ‘annee‘, ‘id_col‘) VALUES (”,"La Morale d’Epicure, avec des reflexions",’1685’,NULL) ;
INSERT INTO ‘livre‘(‘ISBN‘, ‘titre‘, ‘annee‘, ‘id_col‘) VALUES (”,"The society of mind",’1988’,NULL) ;
INSERT INTO ‘livre‘(‘ISBN‘, ‘titre‘, ‘annee‘, ‘id_col‘) VALUES (”,"Traité des animaux",’1766’,NULL) ;
```

```
INSERT INTO ‘auteur‘ (‘id_auteur‘, ‘nom‘, ‘prenom‘) VALUES
                    (1, ’Canfora’, ’Luciano’),
                    (2, ’Epicure’, NULL),
                    (3, ’Minsky’, ’Marvin Lee’),
                    (4, ’Condillac’, ’Etienne Bonnot de’) ;
```


(cc) BY-NC-SA

Exemple de requêtes de "recherche"

Liste l'ensemble des champs des tables *livre*, *auteur* et *ecriture*

```
SELECT ‘auteur‘.*, ‘ecriture‘.*, ‘livre‘.*
FROM ‘livre‘, ‘auteur‘, ‘ecriture‘
WHERE auteur.id_auteur = ecriture.id_auteur
AND livre.ISBN = ecriture.ISBN
```

Liste l'ensemble des livres avec leur auteur et éditeur respectif

```
SELECT livre.titre, auteur.nom, auteur.prenom, editeur.nom_ed
FROM livre, auteur, ecriture, editeur, publication
WHERE auteur.id_auteur = ecriture.id_auteur
AND livre.ISBN = ecriture.ISBN
AND livre.ISBN = publication.ISBN
AND editeur.id_ed = publication.id_ed
```

 (cc) BY-NC-SA

Exemple, suite ...

Liste l'ensemble des livres qui ont une collection

```
SELECT livre.titre, auteur.nom, auteur.prenom, editeur.nom_ed, collection.nom_col
FROM livre, auteur, ecriture, editeur, publication, collection
WHERE auteur.id_auteur = ecriture.id_auteur
AND livre.ISBN = ecriture.ISBN
AND livre.ISBN = publication.ISBN
AND editeur.id_ed = publication.id_ed
AND collection.id_ed = editeur.id_ed
```

Donne le nombre de livres enregistrés ou le nombre de livres enregistrés qui se trouve dans une collection :

```
select count(*) from livre
select count(*) from livre where id_col is not null
```


(cc) BY-NC-SA

SQL-DCL

Cette sous catégorie des commandes SQL définit le contrôle de données.

GRANT : autorisation d'un utilisateur à effectuer une action ;

DENY : interdiction à un utilisateur d'effectuer une action ;

REVOKE : annulation d'une commande de contrôle de données précédente ;

COMMIT : validation d'une transaction en cours ;

ROLLBACK : annulation d'une transaction en cours ;

LOCK : verrouillage sur une structure de données.

Par exemple, la commande GRANT permet de passer des droits d'accès à un utilisateur. Cette commande ce décompose comme suit :

GRANT privilège [liste de colonnes] ON table TO bénéficiaire [*WITH GRANT OPTION (facultatif)*] ;
GRANT UPDATE (nom, prenom) ON auteur TO catalogueur *WITH GRANT OPTION* ;

(cc) BY-NC-SA

Ressources et références

Méthodologie

- L'essentiel sur MERISE*, Dominique Dionisi, Éd.Eyrolles, 2e éd., 1998*
- De MERISE aux Base de données, S.G.B.D. : hiérarchique, réseau, relationnel*,Didier Banos et Michel Mouyssinat, Éd. Eyrolles, 1991*
- *Article Wikipedia sur la méthode MERISE*
- *Tutorial sur le site developpez.com sur la méthode MERISE*

Base de données

- Base de données, Concepts, utilisation et développement*, Jean-Luc Hainaut, Éd. Dunod, 3e éd., 2015*
- *Portail Wikipedia sur les bases de données*

Langage SQL

- *sql.sh*
- *W3C SQL Tutorial*
- *Wikibooks : Programmation SQL*

Les références et ressources présentés ici ne sont pas exhaustives, Internet et les publications sont nombreuses sur ces sujets. N'hésitez pas à les consuslter pour enrichir vos connaissances.

BY-NC-SA

Avez vous des Questions ?

 BY-NC-SA