

# **NSI Terminale - Base de données relationnelles**

Conception

---



## Menu

- Conception, cas général
  - Généralités
  - Analyse
  - MCD
  - Modèle Relationnel
- Conception à partir d'un csv

## Conception, cas général

---

## Intérêts d'un Système de Gestion de Base de Données (SGBD)

- Assure la persistance des données
- Structure l'information
- Permet de trouver rapidement une information
- Multi-utilisateurs
- Sécurise les données
  - Filtre les données qu'un groupe d'utilisateurs peut voir
  - Centralise la sauvegarde et la mise à jour des données
- Maintient la cohérence des données
  - Non redondance
  - Contrôle de l'intégrité des données (lors de la saisie, de la mise à jour, de la suppression)

## Type de SGBD

- Bases hiérarchiques (structure arborescente) ou réseau (structure de graphe)
  - navigation entre les données
- **Bases relationnelles**
  - **Données sous forme de table, langage SQL**
- Bases déductives
  - Intégration d'ensemble de règles, langage DATALOG
- Bases objet
  - Données sous forme d'objets
- Bases noSQL
  - Pas de structuration des données

## Exemples de SGBD Relationnels (SGDBR)

- Access (suite Microsoft Office)
- SQLite (libre, <https://www.sqlite.org/index.html>)
- PostgreSQL (libre, <https://www.postgresql.org/>)
- Oracle (<https://www.oracle.com/>)
- DB2 (<https://www.ibm.com/analytics/db2>)
- H2 (libre, [www.h2database.com](http://www.h2database.com))

## Exemple de la banque (BNP Paribas)

- Grande masse d'informations  $8.10^6$  clients
  - 4 comptes par client, donc  $32.10^6$  comptes
  - 20 écritures par mois par compte, donc  $6,4.10^8$  écritures par mois
- Plusieurs utilisateurs simultanément
  - 2140 agences
  - 31.460 collaborateurs
  - des milliers d'accès internet



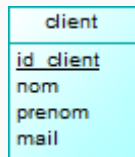
- Travail préalable à la création de la base de données
- Travail Complexe et Difficile
- Déterminer les informations qui sont nécessaires à l'application
  - gestion de la paie, des congés, du stock
  - application web

## **Dictionnaire de données**

- Parmi toutes les informations, on repère les données élémentaires ou propriétés
  - niveau de granularité dépend du contexte (adresse, client, . . . )
- Le dictionnaire de données représente l'ensemble des données élémentaires

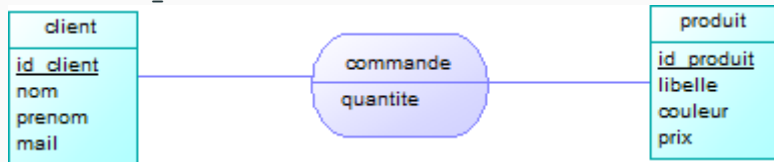
- Le MCD est une représentation du système d'informations à l'aide d'entités et d'associations
- C'est le résultat du travail des analystes, il sert de base à la création de la base de données
- Peut être lu et compris par des non informaticiens
- Un MCD est toujours contextuel

- Une entité regroupe les propriétés relatives à un même sujet, qui a du sens
  - Exemple: une voiture, un individu. . .
- Comporte un identifiant (*clé primaire*)
  - peut être composé par une seule ou plusieurs propriétés
  - est unique: ne peut être le même pour deux entités
- Se représente par un rectangle, l'identifiant est souligné

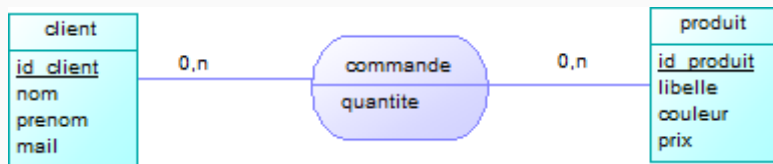


**Figure 1:** Entité client

- Les entités peuvent être liées par des associations
- Une association est une représentation abstraite de la mémorisation d'un lien entre entités
- Elle est représentée par un cercle entre entités
  - Exemples: commande, est\_inscrit, travaille\_pour, est\_marie, habite\_dans....



- Les cardinalités précisent le nombre de fois que l'entité peut intervenir dans une association.
- La valeur minimale est 0 ou 1, la valeur maximale est 1 ou n
- L'association peut-être hiérarchique (maximum 1 d'un côté, n de l'autre) ou maillée (maximum n des deux côtés), entre une ou plusieurs entités



**Figure 2:** Association commande



**Figure 3:** Association est\_inscrit



**Figure 4:** Association est\_inscrit

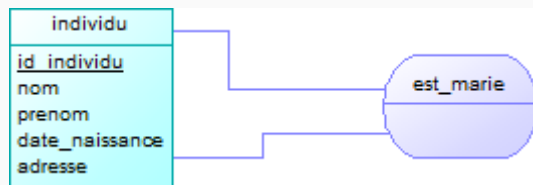


Figure 5: Association est\_marie



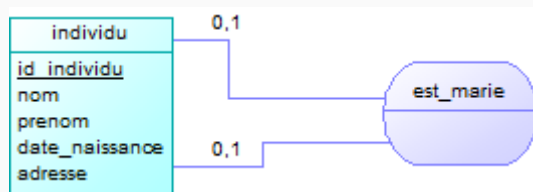


Figure 6: Association est\_marie

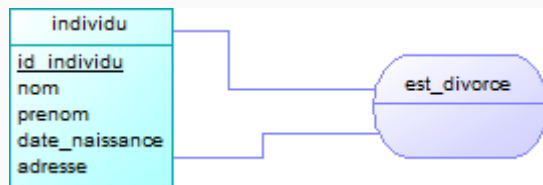


Figure 7: Association est\_divorce

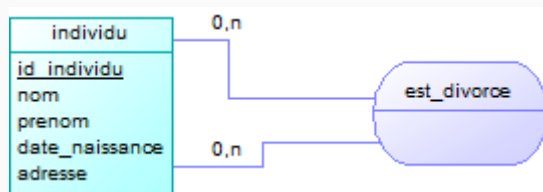


Figure 8: Association est\_divorce

## MCD: Autres exemples

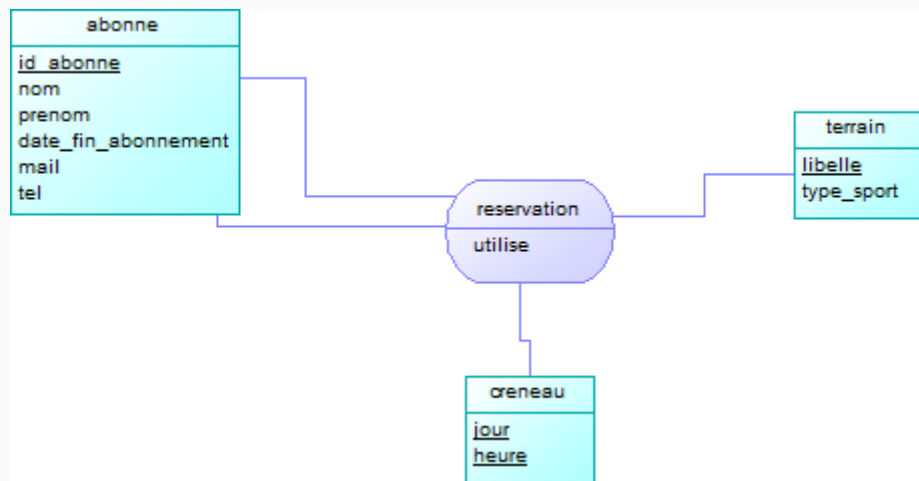


Figure 9: Association reservation

## MCD: Autres exemples

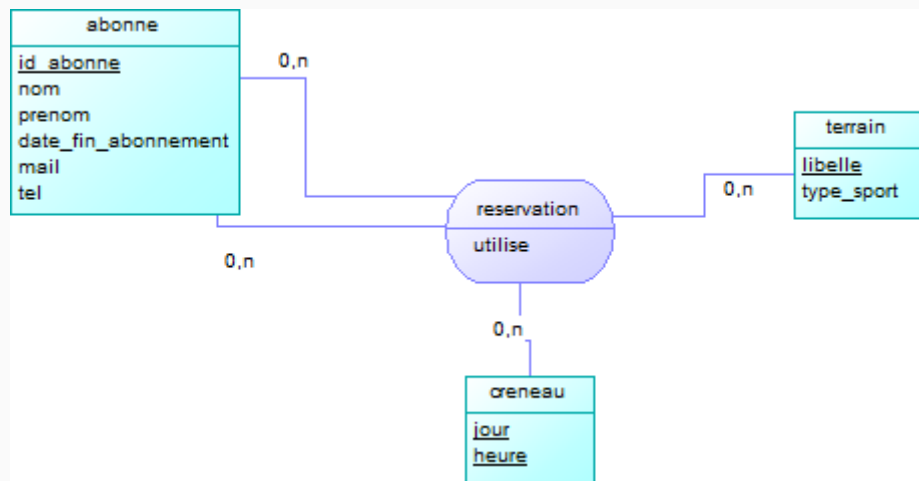


Figure 10: Association reservation

<http://tony3d3.free.fr/files/Les-Cardinalites.pdf>

- La traduction concrète du MCD dépend du modèle de la base
- Dans le cas d'une base de données relationnelle, on obtient un modèle relationnel

## Principe

- Données sous forme de tables
- Chaque table et chaque colonne (ou attribut) porte un nom
- Chaque attribut est typé
- Chaque ligne représente un enregistrement
- Pas de lien physique entre les tables

## Principe

- Chaque entité devient une table
- Chaque propriété d'une entité devient un attribut
- L'identifiant d'une entité devient la clé primaire de la table (Primary Key)

## Lien hiérarchique

- Une association (0-n)-(0-1) (lien hiérarchique) se traduit par la migration de la clé primaire côté n vers une clé étrangère (Foreign Key) côté 1





Figure 11: Association est\_inscrit

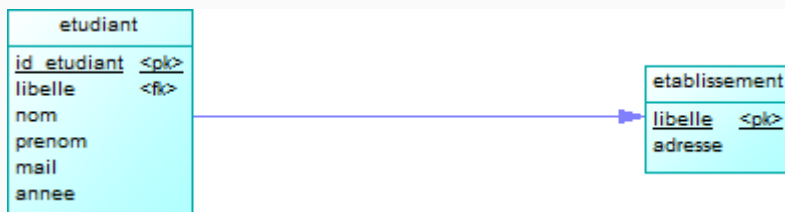


Figure 12: Modèle Relationnel-Lien Hiérarchique

## Lien Maillé

- Une association maillée (0-n)-(0-n) donne lieu à la création d'une nouvelle table dont la clé primaire est l'union des clés primaires des entités qu'elle relie. Ces attributs sont aussi des clés étrangères.

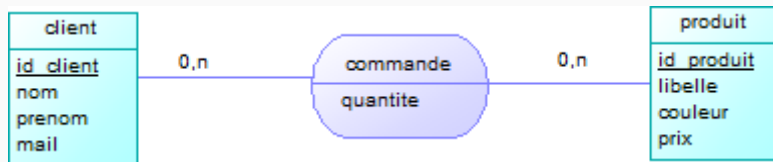


Figure 13: Association commande

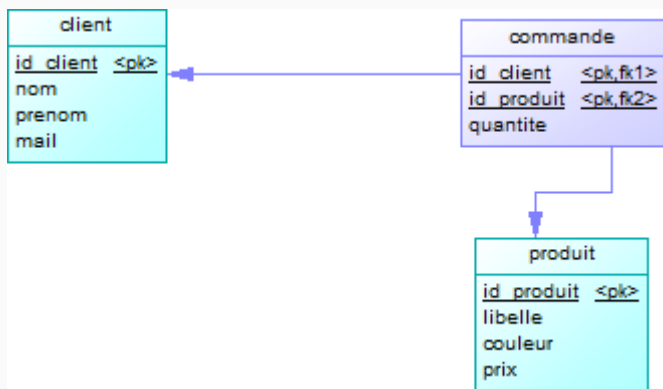


Figure 14: Modèle Relationnel-Lien Maillé

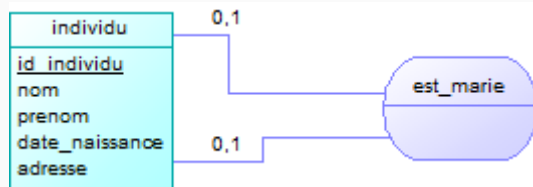


Figure 15: Association est\_marie

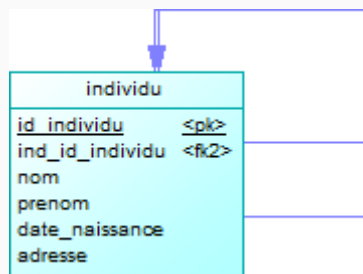


Figure 16: Modèle Relationnel mariage

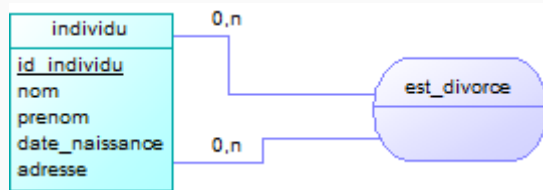


Figure 17: Association est\_divorce

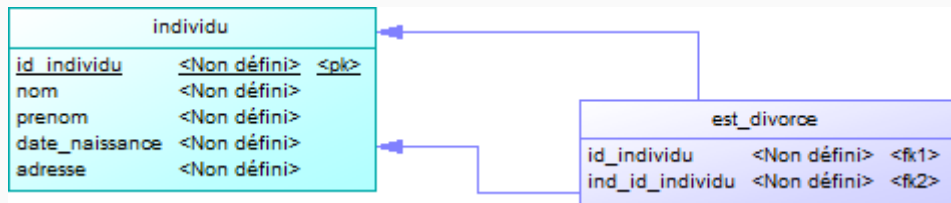


Figure 18: Modèle Relationnel divorce

## Autres exemples

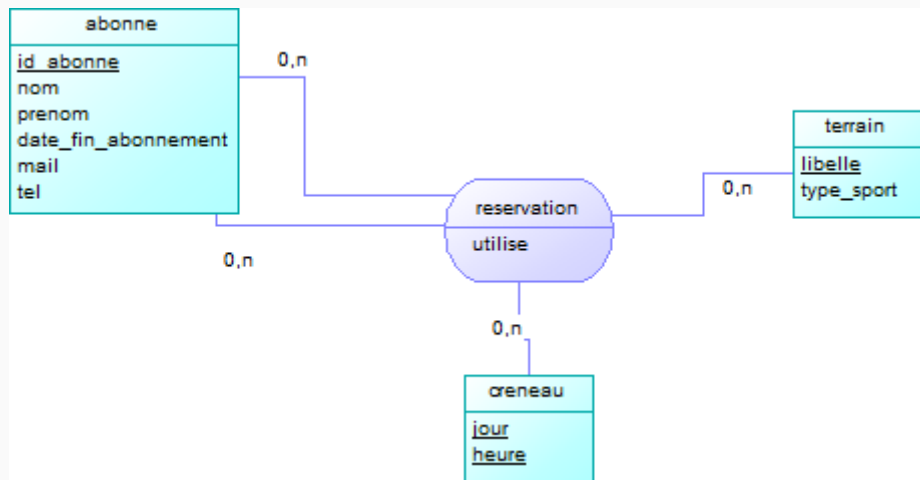


Figure 19: MCD Categorie

## Autres exemples

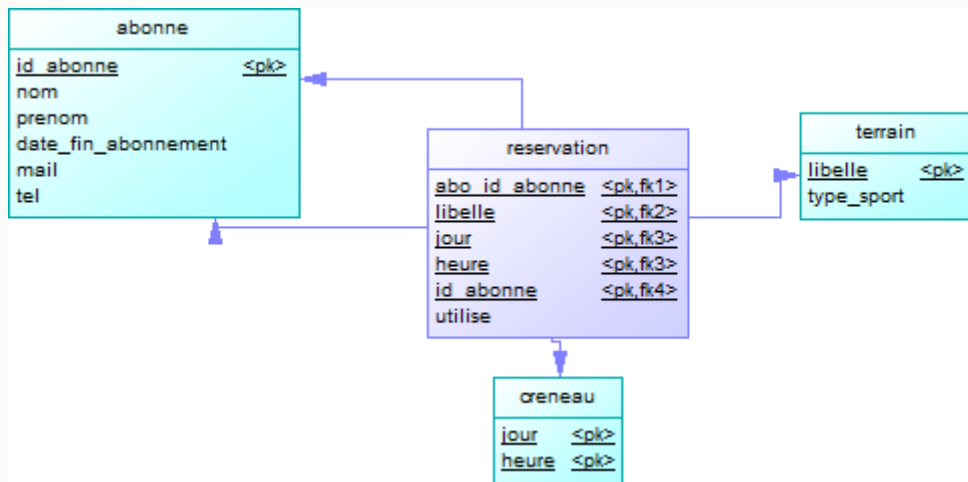


Figure 20: Modèle Relationnel Categorie

## Conception à partir d'un csv

---



- Dans le cas où on récupère des données réelles, le problème se pose différemment
- Le dictionnaire des données est déduit du csv récupéré
- En revanche, les données réelles peuvent poser d'autres problèmes: doublons, incohérences, données non complétées ou peu exploitables
- En général, il faut faire un travail de "nettoyage" pour pouvoir créer les tables

## Regrouper les données en tables

- Mettre dans une même table les données relatives à un même sujet
- Créer de nouvelles tables pour éviter la redondance des données
  - Limite les incohérences lors des mises à jour
  - Facilite la construction des requêtes et améliore la pertinence des résultats

## Établir les relations entre tables

- Définir les clés primaires (*permettant d'identifier les éléments*)
  - Uniques et non NULL
- Définir les clés étrangères
  - Référencent les clés primaires

## Définir des colonnes pertinentes

- Facilité d'interrogation des colonnes
- Données cohérentes au sein d'une colonne
- Ne pas conserver des données qui peuvent être calculées

## Cas des catégories socioprofessionnelles du Nord.

Code g	Rég	Dépa	Libellé géo	Coordonnées	Date	var	Populat	Sexe	Tranche p	Catégorie Socio-Professionnelle	catégorie
59001	31	59	Abancourt	50.2368696873.3.20	2012	15+ - Prof. intermédiaires	28	Femmes	15+	Professions Intermédiaires	Prof. Intermédiaires
59001	31	59	Abancourt	50.2368696873.3.20	2012	15+ - Employés	64	Total	15+	Employés	Employés
59001	31	59	Abancourt	50.2368696873.3.20	2012	55+ - Agriculteurs exploitants	0	Total	55+	Agriculteurs Exploitants	Agriculteurs
59001	31	59	Abancourt	50.2368696873.3.20	2012	15-24 - Employés	16	Total	15-24	Employés	Employés
59001	31	59	Abancourt	50.2368696873.3.20	2012	25-54 - Autres	12	Total	25-54	Autres	Autres
59001	31	59	Abancourt	50.2368696873.3.20	2012	25-54 -	176	Total	25-54	Total	Total
59001	31	59	Abancourt	50.2368696873.3.20	2012	15+ -	196	Hommes	15+	Total	Total
59001	31	59	Abancourt	50.2368696873.3.20	2012	15+ - Artisans, Comm., Chefs entr.	44	Total	15+	Artisans, Commerçants ,Chefs d'entreprises	Chefs d'entreprises
59001	31	59	Abancourt	50.2368696873.3.20	2012	15+ - Autres	32	Total	15+	Autres	Autres
59001	31	59	Abancourt	50.2368696873.3.20	2012	15+ - Ouvriers	8	Femmes	15+	Ouvriers	Ouvriers
59001	31	59	Abancourt	50.2368696873.3.20	2012	25-54 - Artisans, Comm., Chefs entr.	28	Total	25-54	Artisans, Commerçants ,Chefs d'entreprises	Chefs d'entreprises
59001	31	59	Abancourt	50.2368696873.3.20	2012	25-54 - Ouvriers	28	Total	25-54	Ouvriers	Ouvriers
59001	31	59	Abancourt	50.2368696873.3.20	2012	55+ - Autres	0	Total	55+	Autres	Autres
59001	31	59	Abancourt	50.2368696873.3.20	2012	15+ - Artisans, Comm., Chefs entr.	24	Hommes	15+	Artisans, Commerçants ,Chefs d'entreprises	Chefs d'entreprises
59001	31	59	Abancourt	50.2368696873.3.20	2012	Population en 2012 (princ)	445	Total	Population	Total	Total
59001	31	59	Abancourt	50.2368696873.3.20	2012	15+ - Cadres, Prof. intel. sup.	16	Total	15+	Cadres, Professions Intellectuelles Supérieures	Cadres, PIS
59001	31	59	Abancourt	50.2368696873.3.20	2012	15+ - Retraités	48	Femmes	15+	Retraités	Retraités
59001	31	59	Abancourt	50.2368696873.3.20	2012	55+ - Prof. intermédiaires	8	Total	55+	Professions Intermédiaires	Prof. Intermédiaires
59001	31	59	Abancourt	50.2368696873.3.20	2012	15+ - Employés	20	Hommes	15+	Employés	Employés

Figure 21: Fichier d'origine : CSV

## Regrouper les données en tables

- Tables = données relatives à un même sujet
  - Données sur les villes et sur les effectifs des catégories

## Éviter la redondance des colonnes

- var = Tranche + categorie
- categorie = abréviation Catégorie socioprofessionnelle

## Éviter la redondance des valeurs

- Les colonnes `departement` et `region` ne contiennent qu'une seule valeur
- À conserver uniquement si volonté d'étendre à d'autres données

## Données cohérentes

- Les colonnes `Sexe`, `Tranche`, ... contiennent aussi des totaux !
  - Les totaux peuvent être calculés à l'aide de fonctions et d'agrégats
- Décomposer `coordonnees` en deux `REAL`

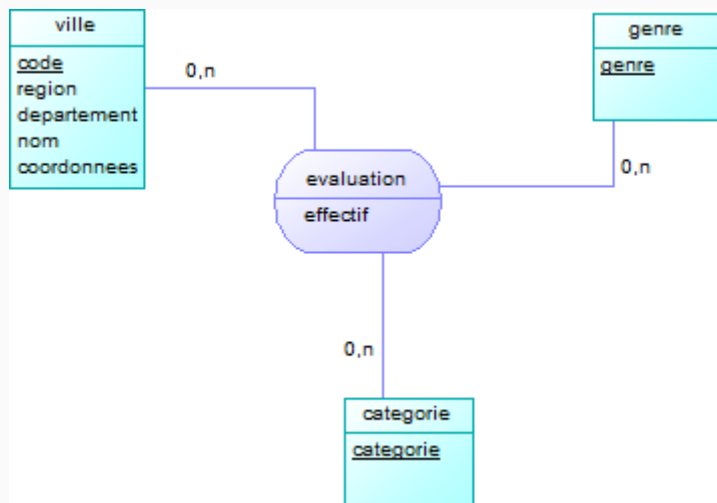


Figure 22: MCD Categorie

## Modèle Relationnel possible

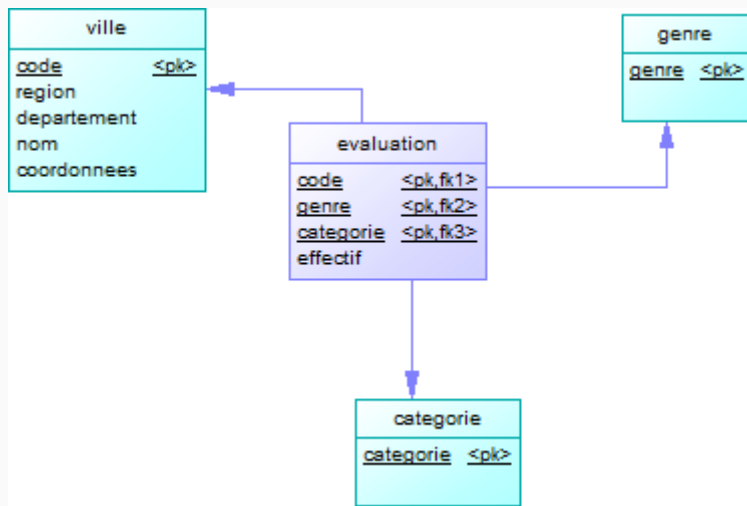


Figure 23: Modèle Relationnel Categorie



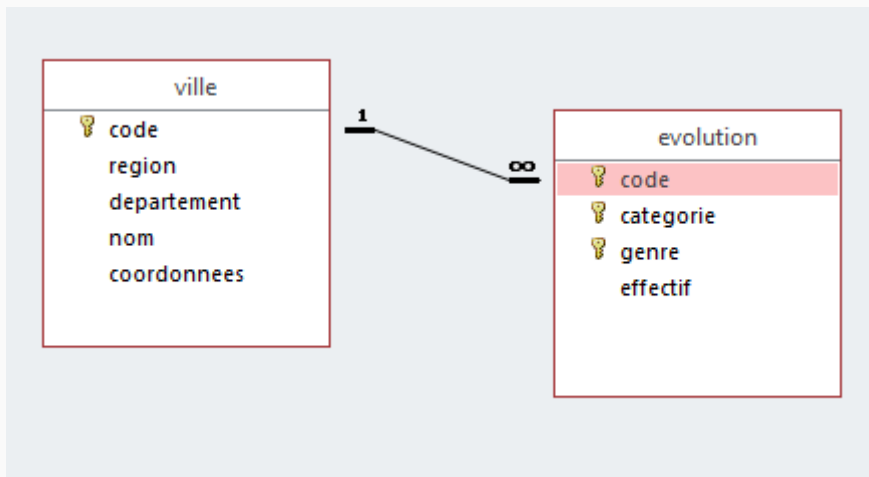


Figure 24: Modèle Relationnel utilisé

# Bilan

---

- Base de données relationnelle: ensemble de tables, aucun lien physique entre les tables
- Contraintes d'intégrité :
  - Clé primaire
  - Clé étrangère (contrainte référentielle)
  - Contrainte de domaine (CHECK prix > 0, CHECK reponse IN ('O','N')...)
- JOIN vs Produit Cartésien: JOIN plus efficace
- JOIN indépendant des contraintes référentielles
- Exécution d'une requête: `SELECT DISTINCT attribut_1, attribut_2 FROM table1 WHERE attribut_3 > 4 ORDER BY attribut_2;`