

NFE 113 – Conception et administration de bases de données

Présentation des TP

André Miralles

Sommaire

TP 7.	Évolution du schéma de la base de données « Bibliothèque »	2
TP 8.	Impact de l'évolution du modèle sur les requêtes	4
TP 9.	Mise en œuvre de View	5

Nota général :

1. Comme vous aurez à faire plusieurs versions de la même base de données, certains noms de table et de contraintes peuvent être en conflit. Afin d'éviter ces derniers, il est suggéré de préfixer les noms de tables et de contraintes par le numéro de TP (Ex. : **TP07_NomBase**).
2. Dans les diagrammes, remplacer le type **STRING** par des **VARCHAR** en mettant le nombre de caractères maximum entre parenthèses.

Travaux pratiques

TP 7. Évolution du schéma de la base de données « Bibliothèque »

Objectif :

Le but de ce TP est faire évoluer la base de données du TP 01 afin qu'elle soit plus générique.

Présentation de la 1^{re} évolution :

Le souhait de prendre en compte certains cas d'utilisation et d'améliorer les performances de la base de données conduit à entreprendre une évolution de cette base de données. Pour ce faire, une analyse approfondie est effectuée dont le résultat est le modèle de la Figure 2. Les différentes évolutions sont issues de l'analyse suivante :

- Lorsqu'on analyse le modèle du TP1, on constate que le nom de la **Ville** de résidence d'un **Abonné** apparaît sous forme d'attribut. Ce n'est pas cohérent parce que le concept de **Ville**, qui est un objet, est inclus dans le concept d'**Abonné**. Comme ce sont deux objets physiquement différents, il est préférable d'extraire le concept de **Ville** et d'en faire une entité en soi (cf. Figure 2) afin d'améliorer la sémantique et la stabilité de la base de données.
- Le concept d'**Abonné** introduit implicitement celui de **Bibliothèque** mais aussi celui d'**Abonnement**, qui est la relation (le contrat) liant un **Abonné** à une **Bibliothèque**. Ces deux concepts n'apparaissent pas dans le modèle du TP1. En poursuivant l'analyse, il est possible de constater que les attributs **Numéro Abonné**, **Tarif** et **Réduction** ne sont pas des propriétés propres à l'**Abonné** mais des propriétés du concept d'**Abonnement**, « contrat » pouvant lier un **Abonné** à une ou plusieurs **Bibliothèques**. La classe-association **Abonnement** est le résultat de cette analyse.
- Pour améliorer les performances, toutes les clés primaires de type *string* et *date* sont abandonnées et sont à remplacer par des clés de type *integer*.
- Par ailleurs, le choix est fait de ne pas utiliser le **Numéro Abonné** et **Numéro** de la classe **Exemplaire** comme clé primaire.

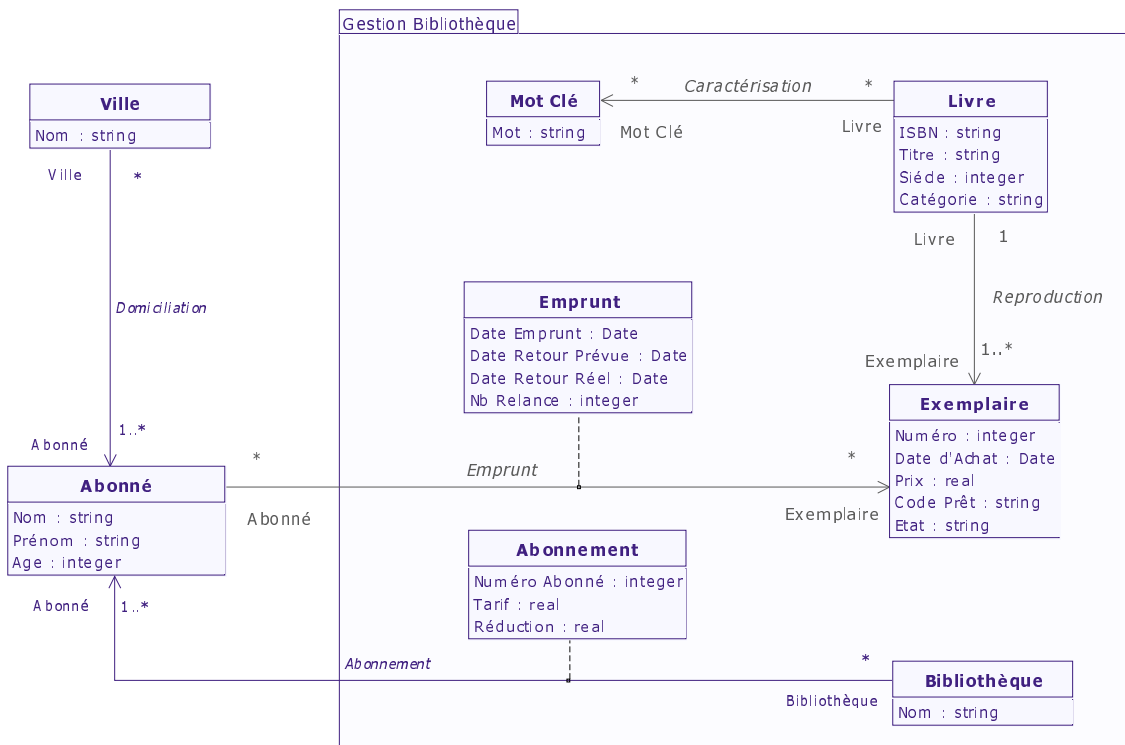


Figure 1. Modèle d'analyse UML de la nouvelle base de données

Présentation de la 2^e évolution :

Le souhait de prendre en compte certains cas d'utilisation et d'améliorer les performances de la base de données conduit à entreprendre une évolution de cette base de données. Pour ce faire, une analyse approfondie est effectuée dont le résultat est le modèle de la Figure 2. Les différentes évolutions sont issues de l'analyse suivante :

- La notion de **Ville** conduit à introduire celle de **Habitant** et plus généralement celle de **Personne Physique**.
- Souvent, les livres sont traduits dans différentes langues. Les utilisateurs de cette base souhaitent identifier l'original et les traductions associées ce qui induit aux deux évolutions :
 - Le numéro **ISBN** ne peut pas être utilisé comme identifiant du concept livre car chacune des éditions a son propre **ISBN**.
 - La langue du livre d'origine et des traductions étant différente, les utilisateurs souhaitent disposer de cette information. L'attribut **Langue** des classes **Livre Original** et **Livre Traduction** résulte de cette exigence (cf. Figure 2). Une contrainte de domaine est ajoutée à cet attribut pour imposer que la valeur appartienne à la liste 'FR', 'BR' ou 'SP'.

A faire :

Pour réaliser cette évolution de la base Bibliothèque, il faut :

- Créer la nouvelle base de données conforme au modèle de la Figure 2.
- Charger cette nouvelle base en utilisant les données de la base du TP08 et celles des fichiers *TP10 - Bibliothèque - Données Tables Hab Pers Phy.txt* et *TP10 - Bibliothèque - Données Table Livre Traduit.txt*.

Nota :

Il est préférable d'introduire les évolutions dans un script de création et un autre d'intégration de données plutôt que de le faire directement dans la base de données. Mettre en entête les instructions DROP et TRUNCATE permettent de relancer le script à la demande.

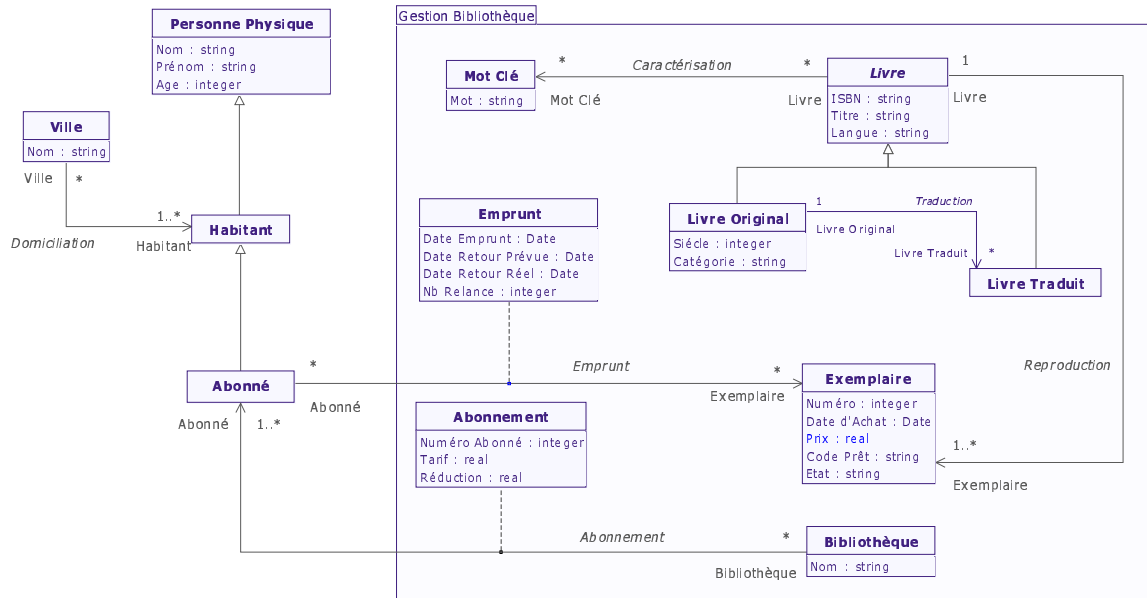


Figure 2. Troisième modèle d'analyse UML de la nouvelle base de données

TP 8. Impact de l'évolution du modèle sur les requêtes

Objectif :

Suite à l'évolution du modèle du TP08 vers celui de la Figure 2, l'objectif de ce TP est réécrire certaines requêtes Oracle pour la nouvelle base de données.

A faire :

Refaire les requêtes listées ci-dessous et comparer le nouveau code par rapport à celui des TP 3, 4 et 5 :

- Q1 Quels sont les nom et prénom des abonnées domiciliés à Montpellier ?
- Q7 Donnez la liste des exemplaires empruntés (leur numéro, code prêt et état) du livre de titre "LE MUR"
- Q8 Donnez la liste des exemplaires (leur numéro, code prêt et le titre du livre associé) d'un livre caractérisé par le mot clef "INFORMATIQUE"
- Q9 Quels sont les exemplaires (numéro) reproduisant le même livre que l'exempleire de numéro 4112 et dont l'état est "BON" ?
- Q10 Quels sont les abonnés (numéro et nom) ayant emprunté un exempleire du livre "LE MUR" ?

- Q11 Existe-t-il une catégorie pour laquelle aucun livre n'a été emprunté ?
- Q14 Par combien d'abonné le livre "LE MIRACLE DE LA ROSE" a-t-il été emprunté (tous exemplaires confondus) ?

TP 9. Mise en œuvre de View

Objectif :

L'objectif de ce TP est d'utiliser et de mettre en œuvre le concept de vue.

A faire :

- Afin d'assurer la continuité des applications utilisant la base de données du modèle de la figure 1 du TP01, créer dans la base TP07_Bibliotheque une vue OldAbonne identique à la table Abonne de cette figure 1.
- Créer dans cette même base de données une vue AllLivre regroupant des données des livres sans redondance d'information (ensemble de TOUTES les colonnes et de TOUS les tuplets).

Nota 1 :

La clé étrangère de la table LivreTraduit doit apparaître et être traitée comme une colonne d'information classique. Aussi, ne pas faire de jointure entre la clé étrangère de la table LivreTraduit et la clé primaire de la table LivreOriginal.

Nota 2 :

Une jointure de type Full Join par exemple entre les tables LivreOriginal et LivreTraduit va accoler les colonnes mais ne permettra pas de retrouver les données dans les bonnes colonnes sans redondance. Aussi, la table Livre doit être impérativement.