

# Rapport, Mini-Projet OR

## Bases de Données Avancées

HMIN106

Master 1 Informatique

Abdoulaye Diallo

Redoine El Ouasti

Marianne Borderes

Benoit Guibert

12 novembre 2015



## Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Description de la base . . . . .	3
1.2	Le modèle objet relationnel (OR) . . . . .	3
1.3	L'approche OR . . . . .	4
<b>2</b>	<b>Modèle conceptuel UML</b>	<b>5</b>
<b>3</b>	<b>Modèle logique</b>	<b>5</b>
<b>4</b>	<b>Exemples de traitement (requêtes)</b>	<b>10</b>
<b>5</b>	<b>Conclusion</b>	<b>11</b>

# 1 Introduction

Ce mini-projet s'inscrit dans le cadre de la formation du Master 1 Informatique (UE Bases de données avancées HMIN106) à l'Université de Montpellier. Son objectif consiste à concevoir et implémenter une base de données objet relationnel avec l'approche OR, RO ou vue objet. Nous avons choisi de modéliser une base de données pour un site référençant des films de cinéma avec l'approche OR.

## 1.1 Description de la base

A travers le site, un utilisateur peut à la fois obtenir des informations sur un film, ses acteurs et son réalisateur mais aussi faire des recherches afin de pouvoir exploiter les liens entre les films. Par exemple, il lui est possible de chercher tous les films d'un réalisateur/acteur ou les films les plus appréciés des utilisateurs du site. L'utilisateur a aussi la possibilité de devenir membre ce qui lui permet d'interagir directement ou indirectement avec les autres utilisateurs. En effet, en devenant membre, un utilisateur peut :

- ☐ Donner un avis sur un film en laissant un commentaire ou en le notant.
- ☐ Créer et/ou adhérer à un groupe de membres partageant des centres d'intérêts cinématographiques.

## 1.2 Le modèle objet relationnel (OR)

Tout en conservant les règles établies par le modèle relationnel, le modèle OR ouvre la porte à de nouvelles possibilités dans la conception et la manipulation des bases de données. Il intègre notamment certains avantages des langages objets qui lui confèrent plus de flexibilité pour représenter les données. La part de l'objet par rapport au relationnel peut se faire à différent niveau dans le modèle selon l'approche utilisée : relationnel-objet, objet-relationnel ou vue objet. Cependant, quelque soit l'approche, les bases de données relationnelles peuvent coexister avec les bases de données implémentant le modèle OR. C'est pourquoi, ce modèle s'est distingué des bases de données objet qui s'écartaient trop des concepts fondamentaux des bases de données existantes. Dans ce modèle, beaucoup des désavantages des bases

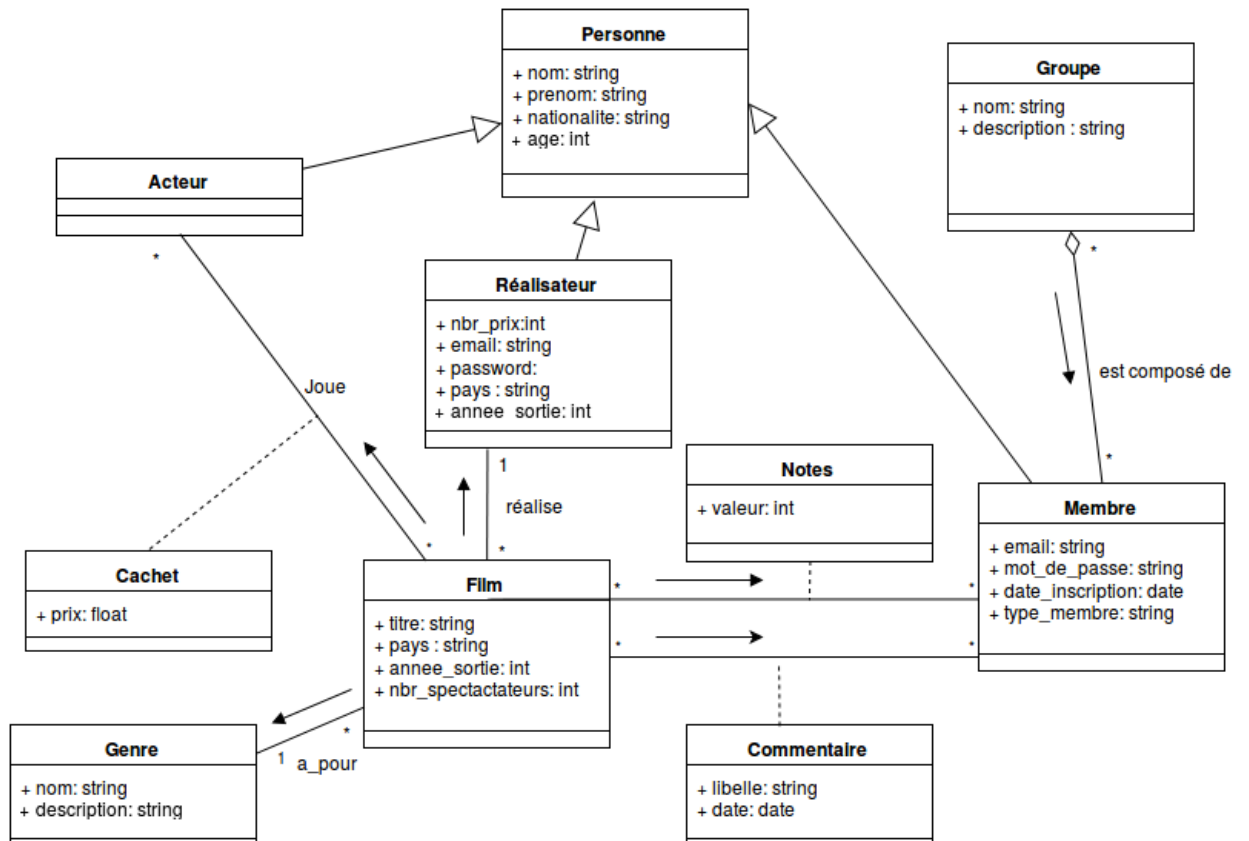
données relationnelles sont revisités tels que :

- Les jointures entre les tables qui étaient très coûteuses sont très peu utilisées voir plus du tout selon l'approche choisie.
- Les attributs peuvent être multi-valués et stockés dans les listes, tableaux ou ensembles.

### 1.3 L'approche OR

Pour réaliser notre base nous avons opté pour l'approche OR. Parmi les 3 approches, elle est celle qui intègre le plus les concepts objets. En effet, ce modèle n'établit pas de tables relationnelles. Toutes les tables du schéma conceptuel deviennent des types et seulement certaines vont être représentées sous forme de tables objet dans lesquelles chaque instance d'une table se voit attribuée un oid. Ces tables vont constituer des points d'entrées pour la manipulation des données. Nous avons choisi l'approche OR car elle permet grâce aux tables objets et aux oid de manipuler des références d'objet mais aussi de travailler avec des collections. Il nous intéresse surtout de faire des collections de référence d'objet. Cela permet de donner plus de cohérence entre les types et les tables.

## 2 Modèle conceptuel UML



## 3 Modèle logique

### Les types

```

1 Type personne_t (
2     nom ,
3     prenom ,
4     nationalite ,
5     age )
6 Type acteur_t specialise personne_t ()
7 Type liste_acteurs_t est une collection de ref acteur_t
8 Type genre_t (
9     nom ,
10    description )
  
```

```

11 Type realisateur_t specialise personne_t (
12     nb_prix )
13 Type membre_t specialise personne_t (
14     date_inscription ,
15     email ,
16     mot_de_passe ,
17     type_membre )
18 Type liste_membres_t collection de ref membre_t
19 Type groupe_t (
20     nom ,
21     description ,
22     liste_membres : liste_membre_t )
23 Type commentaire_t (
24     date_commentaire ,
25     libelle )
26 Type commentaire_nt (
27     commentaire : commentaire_t,
28     membre : reference membre_t )
29 Type liste_commentaires_nt collection de commentaires_nt
30 Type note_t (
31     valeur )
32 Type note_nt (
33     note : note_t ,
34     membre : reference membre_t )
35 Type liste_notes_nt collection notes_nt
36 Type cachet_t (
37     prix )
38 Type cachet_nt (
39     cachet : cachet_t
40     acteur : reference acteur_t
41 )
42 Type liste_cachet_nt est une collection de cachet_nt
43 Type film_t(
44     titre,
45     annee_sortie,
46     pays,
47     nb_spectateurs,
48     genre: genre_t,
49     realisateur: reference realisateur_t,
50     liste_acteurs: liste_acteurs_t,
51     liste_commentaires: liste_commentaires_nt,

```

```

52     liste_notes liste_notes_nt,
53     liste_cachets: liste_cachets_nt)

```

## Les tables objet

```

1  table objet de type film_t film (
2      liste_acteur : Tab_acteurs ,
3      liste_commentaires : Tab_commentaires ,
4      liste_notes : Tab_notes ,
5      liste_cachets : Tab_cachets ,)
6  table objet de type acteur_t acteur ()
7  table objet de type membre_t membre ()
8  table objet de type groupe_t groupe (
9      liste_membres : Tab_membres )
10 table type realisateur_t realisateur ()

```

Type personne\_t

personne_t			
Nom	Prenom	Nationalite	age

Type acteur\_t

acteur_t			
personne_t			
Nom	Prenom	Nationalite	age

Type genre\_t

genre_t	
nom	description

Type realisateur\_t

realisateur_t				
personne_t				nb_prix
Nom	Prenom	Nationalite	age	

Type membre\_t

membre_t							
personne_t				date_inscription	Email	mot_de_passe	type_membre
Nom	Prenom	Nationalite	age				

Type groupe\_t

groupe_t		
Nom	description	{ liste_membres }
		liste_membres_t

Type commentaire\_t

commentaire_t	
date_commentaire	libelle

Type commentaire\_nt

commentaire_nt		
commentaire_t		@membre
date_commentaire	libelle	membre_t

Type note\_t

note_t	
	valeur

Type note\_nt

note_nt	
note_t	@membre
valeur	membre_t



Type cachet\_t

cachet_t
prix

Type cachet\_nt

cachet_nt	
cachet_t	@acteur
prix	acteur_t

Type film\_t

film_t									
Titre	annee_sortie	Pays	nb_spectateurs	genre	@realisateur	{liste_acteurs}	{liste_commmentaires}	{liste_notes}	{liste_cachets}
				genre_t	realisateur_t	liste_acteurs_t	liste_commentaires_nt	liste_notes_nt	liste_cachets_nt

Table film

OID	Titre	annee_sortie	Pays	nb_spectateurs	genre	@realisateur	{Tab_acteurs}	{Tab_commentaires}	{Tab_notes}	{Tab_cachets}
					genre_t		liste_acteurs	liste_commentaires	liste_notes	liste_cachets

table acteur

OID	personne_t			
	nom	prenom	Nationalité	age

table membre

OID	personne_t				date_inscription	Email	mot_de_passe	type_membre
	Nom	Prenom	Nationalite	age				

table groupe

OID	Nom	description	{liste_membres}
			liste_membres_t

table realisateur

OID	personne_t				nb_prix
	Nom	Prenom	Nationalite	age	

## 4 Exemples de traitement (requêtes)

```

1  -- Noms des membre d'un groupe
2  select deref(value(t)).nom
3  from groupe g, Table(g.liste_membres) t
4  where g.nom = 'Fans du film Le livre d''eli ';

```

```

1  -- Noms des membres n'appartenant aucun groupe
2  select m.nom
3  from membre m
4  where m.nom not in (select deref(value(t)).nom from groupe g,
    Table(g.liste_membres) t );

```

```

1  -- Le top 3 des films
2  select f.titre, avg(value(t).note.valeur) as note_moyenne
3  from film f, Table(f.liste_notes) t
4  where rownum < 4
5  group by f.titre
6  order by note_moyenne desc;

```

```

1  -- Tous les films d'un realisateur
2  select f.titre
3  from film f
4  where deref(f.realisateur).nom = 'tarantino';

```

```

1
2  -- Films du realisateur ayant le plus de prix
3  select deref(f.realisateur).nom as nom, deref(f.realisateur).
    nb_prix as nb_prix, f.titre

```

```

4 from film f
5 where deref(f.realisateur).nb_prix >= all (select deref(f2.
    realisateur).nb_prix from film f2);

1 -- noms des acteurs et leur cachet dans un film
2 select value(c).acteur.nom, value(c).cachet.prix
3 from film f, Table(f.liste_cachets) c
4 where f.titre = 'Django Unchained';

1 -- tous les films d'un acteur
2 select deref(value(t)).nom as nom, f.titre
3 from film f, Table(f.liste_acteurs) t
4 where deref(value(t)).nom = 'DiCaprio';

1 -- Tous les commentaires d'un film
2 select deref(value(t).membre).nom as Nom, value(t).
    commentaire.libelle as Commentaire, value(t).commentaire.
    date_commentaire as date_comm
3 from film f, Table(f.liste_commentaires) t
4 where f.titre = 'Forest Gump';

1 -- Afficher des informations sur les tous les films
2 select f.titre, deref(f.realisateur).nom, f.genre.nom, f.
    annee_sortie, f.nb_spectateurs, f.pays
3 from film f;

1 -- Le plus gros cachet donne a un acteur dans un film
2 select value(t).cachet.prix as cachet, deref(value(t).acteur)
    .nom as nom, f.titre
3 from film f, Table(f.liste_cachets) t
4 where value(t).cachet.prix >= all (select value(c).cachet.
    prix from film f2, Table(f2.liste_cachets) c);

```

## 5 Conclusion

Nous avons dans ce présent document proposé une mise en pratique de nos acquis dans le cadre de l'enseignement des bases de données objets (HMIN106). La simplicité du code et le respect des concepts objets permettront aux curieux de modifier et d'améliorer notre projet avec aisance :

implémentation de nouvelles fonctionnalités, ajout de tables/types en fonction des besoins ... Une extension qui pourrait, selon nous, être intéressante, serait de permettre d'accéder depuis un acteur ou un réalisateur à la liste de ses films. Pour cela il faudrait ajouter dans ces deux tables une collection de références de films. De plus cela nécessiterait de gérer à la fois l'interdépendance lors de la création des types acteur, réalisateur et film mais aussi de s'assurer lors de l'insertion de la cohérence des collections. Par exemple lorsqu'on ajoute un acteur dans la collection d'acteurs d'un film, il faudrait vérifier que ce même film soit ajouté dans la collection des films d'un acteur. Enfin il est important de souligner les intérêts de ce projet qui sont avant tout un enrichissement personnel, une mise en évidence de l'approche Objet Relationnel mais aussi le travail de groupe.