

Equipe :

Anas Mejjari

**Hatim Ait
Ishaq**

Anas Benalla

**Soufiane
Taghy**

Grenoble
ENSIMAG



Projet Base de Données



**Responsable
projet :**

**Christophe
Bobineau**

Encadrant :

**Ugo
Comignani**

Plan du projet

1. Introduction
2. Analyse du problème
 - 2.1. Description de l'application
 - 2.2. Dépendances fonctionnelles et contraintes
 - 2.2.1. Dépendances fonctionnelles
 - 2.2.2. Contraintes de valeurs, multiplicité et contextuelles
3. Conception Entité/Associations
4. Schéma relationnel
5. Analyse des fonctionnalités
6. Bilan du projet

1. Introduction

Le but du présent projet de base de données est de traiter le sujet suivant :

- Proposer une gestion de base de données pour une société multimédia
- Mettre en oeuvre nos compétences en systèmes de gestion de base de données relationnelles

Le projet sera traité de la manière suivante :

- Analyse du problème
- Conception Entité Association
- Traduction en relationnel
- Analyse des fonctionnalités
- Bilan du projet

2. Analyse du problème

Dans ce projet, on nous a demandé de concevoir et d'implémenter la base de données décrivant les fichiers multimédias hébergés sur le serveur. Tout d'abord, nous avons analysé le problème à partir de la description de l'application. Nous avons par conséquent pu déterminer un ensemble de données et de contraintes.

Les données serviront à créer les dépendances fonctionnelles entre les attributs.

Cependant, nous aurons des contraintes sur ces relations que nous devons prendre en considération. Ensuite, nous allons modéliser ce travail d'analyse par un schéma Entité/Association, qu'on pourra traduire par la suite en un schéma relationnel.

Le schéma relationnel sera implémenté en SQL.

Description de l'application

Nous avons suivi une analyse avec la méthode du cours : analyse paragraphe par paragraphe pour en extraire les informations utiles.

Paragraphe 1 :

Les utilisateurs sont identifiés par une adresse email et décrits par leur nom, prénom, âge et langue de diffusion préférée. Un code d'accès à quatre chiffres (stocké en clair) est également associé au compte utilisateur. Ces informations seront utiles pour rendre l'interaction plus agréable aux utilisateurs, ainsi que pour respecter les recommandations d'âge minimal pour les contenus.

Les utilisateurs sont identifiés par une **adresse email** et décrits par leur **nom, prénom, âge et langue de diffusion préférée**. Un **code d'accès à quatre chiffres** (stocké en clair) est également associé au compte utilisateur. Ces informations seront utiles pour rendre l'interaction plus agréable aux utilisateurs, ainsi que pour respecter les **recommandations d'âge minimal pour les contenus**.

Les utilisateurs sont identifiés par une **adresse email** et décrits par leur **nom, prénom, âge et langue de diffusion préférée**. Un **code d'accès à quatre chiffres** (stocké en clair) est également associé au compte utilisateur. Ces informations seront utiles pour rendre l'interaction plus agréable aux utilisateurs, ainsi que pour respecter les **recommandations d'âge minimal pour les contenus**.

DF	Cont. Valeur	Cont. Multiplicité	Cont. Context.
Mail → Nom, Prenom, Age, langueUtilisateur, CodeAccess	Age > 0		Code d'accès à 4 chiffres. Age utilisateur > Age minimale

Paragraphe 2 :

Les fichiers déposés par les utilisateurs sur le serveur sont décrits par un identifiant unique généré automatiquement, leur taille en Ko, la date de leur dépôt, ainsi que la description du ou des flux multimédias qu'ils contiennent. Chaque flux d'un fichier multimédia est numéroté. Un flux peut être un flux vidéo (ne comprenant que l'image), un flux audio ou un flux texte (paroles d'une chanson ou sous-titre d'un film). La description des flux vidéo comprend le codec utilisé (par exemple : MPEG2, MPEG4, DivX, H264, ...), la résolution de l'image (largeur et hauteur) et le débit en Kbit/s. Les flux audio sont décrits par leur codec (comme MP3, MPEG4, ACC, AC3, ...), leur échantillonnage (16, 24 ou 32 bits), leur débit en Kbit/s et leur langue. Les flux texte sont simplement décrits par leur codec (tels que AQTitle, DKS, Kate, ...), leur débit en Kbit/s et leur langue.

Les fichiers déposés par les utilisateurs sur le serveur sont décrits par un **identifiant unique** généré automatiquement, leur **taille en Ko**, la **date** de leur dépôt, ainsi que **la description du ou des flux multimédias** qu'ils contiennent. Chaque **flux d'un fichier multimédia est numéroté**. Un flux peut être un **flux vidéo** (ne comprenant que l'image), **un flux audio ou un flux texte** (paroles d'une chanson ou sous-titre d'un film). La description des **flux vidéo comprend le codec** utilisé (par exemple : MPEG2, MPEG4, DivX, H264, ...), la résolution de l'image (largeur et hauteur) et **le débit en Kbit/s**. Les flux audio sont décrits **par leur codec** (comme MP3, MPEG4, ACC, AC3, ...), **leur échantillonnage (16, 24 ou 32 bits)**, **leur débit en Kbit/s et leur langue**. Les **flux texte** sont simplement décrits par leur **codec** (tels que AQTitle, DKS, Kate, ...), **leur débit en Kbit/s et leur langue**.

DF	Cont. Valeur	Cont. Multiplicité	Cont. Context.
idFichier → Taille,Date_Depot,Mail	Echantillonnage $\in \{8, 16, 32\}$	idFichier - - > > numFlux	
NumFlux → Codec,Debit,idFichier	{NumFluxVideo} U {NumFLuxAudio} U {NumFluxTexte} = {NumFlux}	idFichier - / -> titreFilm,AnneeSortie	
NumFluxVideo → Codec,Largeur,Hauter		titreFilm,AnneeSortie - -> idFichier	
NumFluxAudio → Codec,Echantillonnage, LangueA ,Codec	Debit > 0 ; Taille >0	idFichier - / -> numPiste numPiste - -> idFichier	
NumFluxTexte → Codec,LangueTxt			

Paragraphe 3 :

Les contenus multimédias sont soit des films, soit des pistes musicales. L'association entre fichiers et les contenus est normalement mise en place par un service web faisant intervenir un algorithme de « Deep Learning », mais dans le cadre de ce projet, l'association sera faite à la main. Plusieurs fichiers peuvent être associés à un même contenu. En effet, les sous-titre d'un film peuvent par exemple être séparés du fichier audio-vidéo, ou un même film peut être disponible dans des résolutions et ou débits différents ou encodé avec des codecs différents, de même pour les pistes musicales. L'utilisateur aura la possibilité de corriger l'association entre fichiers et contenus.

Ce paragraphe ne donne pas des informations sur la construction des tables, mais plutôt des informations sur la cardinalité et la manière dont seront reliés certaines tables.

Les contenus multimédias sont soit des films, soit des pistes musicales. L'association entre fichiers et les contenus est normalement mise en place par un service web faisant intervenir un algorithme de « Deep Learning », mais dans le cadre de ce projet, l'association sera faite à la main. **Plusieurs fichiers peuvent être associés à un même contenu** En effet, les sous-titre d'un film peuvent par exemple être séparés du fichier audio-vidéo, ou **un même film peut être disponible dans des résolutions et ou débits différents ou encodé avec des codecs différents, de même pour les pistes musicales.** L'utilisateur aura la possibilité de corriger l'association entre fichiers et contenus.

Paragraphe 4 :

Un film est identifié par son titre et son année de sortie. Il est décrit par une ou plusieurs catégories, un résumé, un âge minimum recommandé et l'URL de l'affiche du film. Une ou plusieurs URL de photo extraites du film peuvent également être incluses dans sa description.

Un film est identifié par son titre et son année de sortie. Il est décrit par une ou plusieurs catégories, un résumé, un âge minimum recommandé et l'URL de l'affiche du film. Une ou plusieurs URL de photo extraites du film peuvent également être incluses dans sa description.

DF	Cont. Valeur	Cont. Multiplicité	Cont. Context.
Titre, anneeSortie → resume, ageMin, URL	ageMin > 0	Titre, anneeSortie -->> categorieFilm Titre, anneeSortie -/->> UrlPhoto	age > ageMin

Paragraphe 5 :

Les pistes musicales sont organisées en albums. Un album a un numéro unique, un titre, un nom de groupe ou d'artiste, une date de sortie, l'URL d'une image de sa pochette et une ou plusieurs catégories. Les pistes d'un album, associées à des fichiers, sont identifiées par leur numéro de piste et décrits par leur titre et leur durée (en minutes). Une ou plusieurs catégories peuvent être également associées aux pistes si, par exemple, un même album contient des pistes de style très différents. Les catégories de films et de musique sont bien évidemment distinctes.

Les pistes musicales sont organisées en albums. Un album a un **numéro unique**, un **titre**, un **nom de groupe ou d'artiste**, une **date de sortie**, l'**URL** d'une image de sa pochette et **une ou plusieurs catégories**. Les pistes d'un album, associées à des fichiers, sont identifiées par leur **numéro de piste** et décrits par leur **titre** et leur **durée** (en minutes). **Une ou plusieurs catégories** peuvent être également associées aux pistes si, par exemple, un même album contient des pistes de style très différents. Les catégories de films et de musique sont bien évidemment distinctes.

DF	Cont. Valeur	Cont. Multiplicité	Cont. Context.
idAlbum -->> Titre, idArtiste, dateSortie, urlAlbum idAlbum, numPiste -- >> titre, durée	durée > 0	idAlbum -->> categorieMusique idAlbum, numPiste -->> categorieMusique	

Paragraphe 6 :

Différents artistes interviennent dans des contenus multimédias. Ils peuvent par exemple être acteurs dans un film (et l'on précisera le nom du personnage qu'il joue) ou avoir une autre fonction (par exemple: réalisateur). Ils peuvent être également musicien (en précisant son instrument) ou chanteur sur une piste audio. Les artistes sont identifiés par un numéro unique et décrits par leur nom, leur date de naissance (si elle est connue), l'URL d'une photo et leur spécialité principale. Une courte biographie peut également être donnée. Un même artiste peut être musicien dans des albums et également intervenir dans des films, ou l'inverse. Tout est possible.

Différents artistes interviennent dans des contenus multimédias. Ils peuvent par exemple être acteurs dans un film (et l'on précisera le nom du personnage qu'il joue) ou avoir une autre fonction (par exemple: réalisateur). Ils peuvent être également musicien (en précisant son instrument) ou chanteur sur une piste audio. Les **artistes** sont identifiés par un **numéro unique** et décrits par leur **nom**, leur **date de naissance** (si elle est connue), l'**URL d'une photo** et leur **spécialité principale**. Une **courte biographie** peut également être donnée. Un même artiste peut être musicien dans des albums et également intervenir dans des films, ou l'inverse. Tout est possible.

DF	Cont. Valeur	Cont. Multiplicité	Cont. Context.
idArtiste -->> Nom, dateNaissance, urlPhoto,specialite		idArtiste -/->>dateNaissance ,courteBiographie	

Paragraphe 7 :

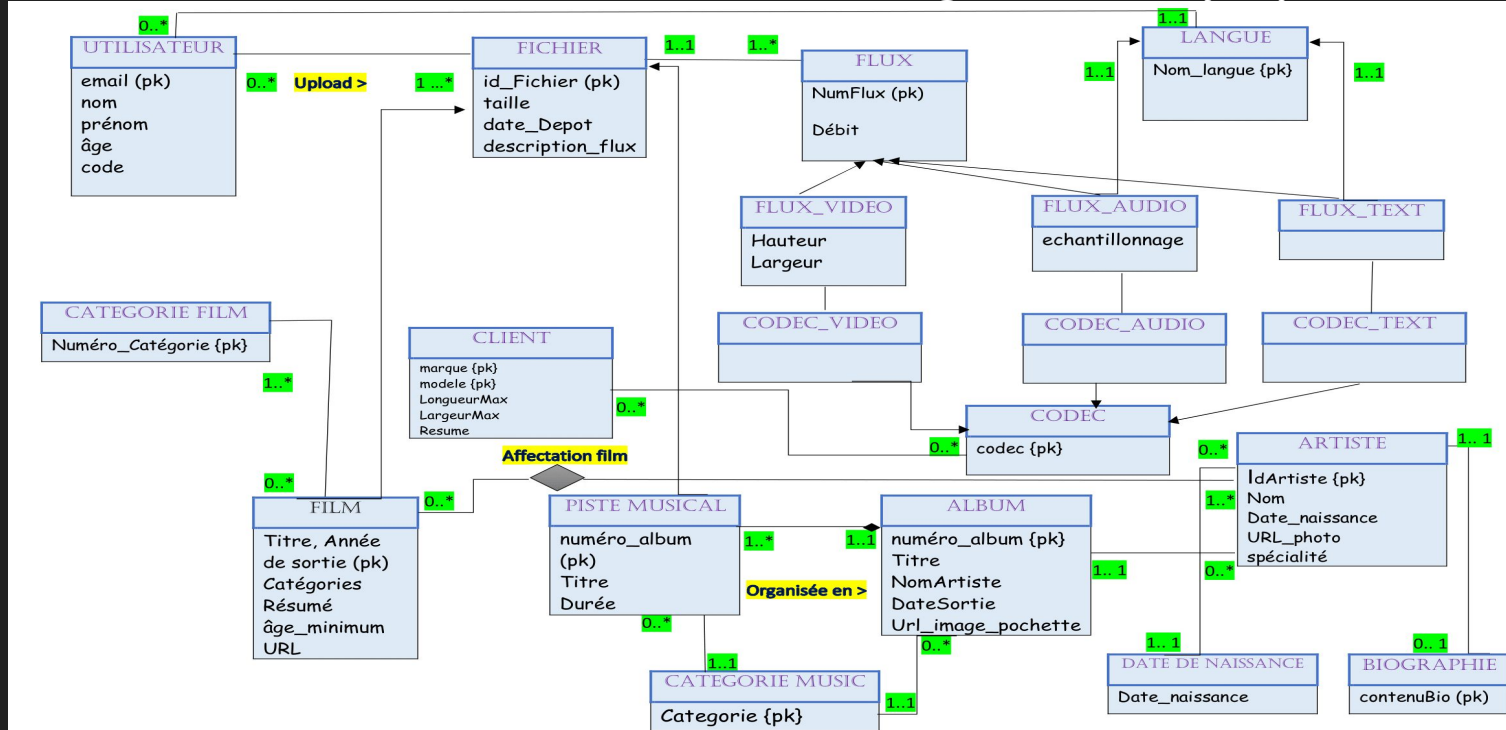
Les contenus multimédias seront joués sur des équipements ou des logiciels que nous appellerons des clients. Tous les clients n'ont pas les mêmes caractéristiques et ne peuvent donc pas jouer tous les contenus. La base de données doit donc également comprendre leur description. Celle-ci comprend la marque et le modèle (ou nom de logiciel) qui serviront d'identifiant. Elle comprend également la résolution maximale d'affichage (largeur et hauteur), ainsi que la liste des codecs audio, vidéo et texte qu'il peut interpréter.

Les contenus multimédias seront joués sur des équipements ou des logiciels que nous appellerons des clients. Tous les **clients** n'ont pas les mêmes caractéristiques et ne peuvent donc pas jouer tous les contenus. La base de données doit donc également comprendre leur **description**. Celle-ci comprend la **marque** et le **modèle** (ou nom de logiciel) qui serviront d'identifiant. Elle comprend également la résolution maximale d'affichage (**largeur et hauteur**), ainsi que la liste des codecs audio, vidéo et texte qu'il peut interpréter.

DF	Cont. Valeur	Cont. Multiplicité	Cont. Context.
Marque,Modele-->> resume,largeur,hauteur		Marque,Modele-->> CodecVideo,CodecAudio,CodecTexte	

3. Conception Entité / Associations

Nous avons construit le schéma Entité/Association grâce à l'analyse précédente :



4. Schéma Relationnel

Ensuite, notre travail consiste à traduire le modèle Entité/Association en schéma relationnel par la méthode vu en cours.

Par conséquent, certaines tables ont été ajoutées à partir des multiplicités indiquées dans le schéma entités-associations. Nous détaillons ce point dans la partie documentation qui se trouve dans le document pdf associé.

Remarque : Pour des raisons de coding style, certains noms de variables ont été légèrement modifiés par rapport à leur noms dans la partie analyse : comme par exemple `num_flux` qui devient `NumFlux`.

On obtient le schéma relationnel suivant

Utilisateur(Mail,Nom,Prenom,AGE,CodeAcces,LangueUtilisateur) : 3FNBCK

Fichier(idFichier,Taille,dateDepot,Mail): 3FNBCK

Flux(NumFlux,Debit ,idFichier): 3FNBCK

FluxVideo(NumFluxVideo,Codec,Largeur,Hauteur) : 3FNBCK

FluxAudio(NumFluxAudio,Echantillonnage,LangueA,Codec) : 3FNBCK

FluxTexte(NumFluxTexte,,LangueTxt,Codec) : 3FNBCK

Codec(Codec) : 3FNBCK

CodecAudio(Codec) : 3FNBCK

CodecVideo(Codec) : 3FNBCK

CodecTexte(Codec) : 3FNBCK

Artiste(IdArtiste,Nom,URL_photo,Specialite) : 3FNBCK

Biographie(IdArtiste,contenueBio) : 3FNBCK

DateNaissance(DateDeNaissance) : 3FNBCK

aPourDateDeNaissance(IdArtiste,DateDeNaissance) : 3FNBCK

* (IdArtiste) REFERENCES Artiste(IdArtiste)

*(DateDeNaissance) REFERENCES DateNaissance(DateDeNaissance)

*Vérifier qu'un artiste ait exactement une date de naissance

Album(IdAlbum,IdArtiste,dateSortie,urlAlbum) : 3FNBCK

PisteMusicale(IdAlbum,NumPiste,Titre,Duree) : 3FNBCK

*(IdAlbum) REFERENCES Album(IdAlbum)

Film(Titre,anneeSortie,Resume,ageMin) : 3FNBCK

FichierFilm(Titre,anneeSortie,idFichier) : 3FNBCK

*(idFichier) REFERENCES Fichier(idFichier)

*(Titre, anneeSortie) REFERENCES Film(Titre, anneeSortie)

FichierPiste(idAlbum,NumPiste,idFichier) : 3FNBCK

*(idFichier) REFERENCES Fichier(idFichier)

*(idAlbum, NumPiste) REFERENCES PisteMusicale(idAlbum, NumPiste)

Client(Marque,Modele,Resume,largeurMax,hauteurMax) : 3FNBCK

ClientFlux(Marque,Modele,NumFlux) : 3FNBCK

*(Marque, Modele) REFERENCES Client(Marque, Modele)

*(NumFlux) REFERENCES Flux(NumFlux)

CategorieFilm(Categorie) : 3FBNCK

CategorieMusique(Categorie) : 3FNBCK

AssocFilmCat(Titre,anneeSortie,Categorie) : 3FNBCK

***Categorie REFERENCES CategorieFilm**

***Titre, anneeSortie référence Film**

***Vérifier qu'un film ait au moins une catégorie**

AssocAlbumCat(idAlbum,Categorie) : 3FNBCK

***Categorie REFERENCES CategorieMusique**

***idAlbum REFERENCES Album**

***Vérifier qu'un album ait au moins une catégorie**

AssocPisteCat(idAlbum,NumPiste,Categorie) : 3FNBCK

*** Categorie REFERENCES CategorieMusique**

***idAlbum, NumPiste REFERENCES PisteMusicale**

***Vérifier qu'une piste ait au moins une catégorie**

aPourRole(idArtiste,Titre,anneeSortie,nomDuPersonnage,Categorie) : 3FNBCK

5. Analyse des fonctionnalités

Dans cette partie, on illustre les fonctionnalités par l'intermédiaire des transactions de manière similaire aux exemples du cours. On propose l'exemple suivant :

Création d'un nouvel album ou film

1. Begin;
2. Insert into **Album** values ("A", "titre", "idArtiste", "date", "URL");
3. Insert into **PisteMusicale** values("A", "numpiste", "titre", "duree");
4. Delete from **FichierPiste** where **NumPiste** not in (select **NumPiste** from **PisteMusicale**);
5. Delete from **Album** where **idAlbum** not in (select **idAlbum** from **PisteMusicale**);
6. Commit;
7. Insert into **Film** values ("titre", "annee", "resume", "age", "URL");
8. Commit;

Commentaire :

Pour réaliser les fonctionnalités nous avons défini les requêtes SQL2.

Nous avons proposé pour cette présentation un exemple proche de ce qui a été vu en cours.

Les requêtes et transactions (en particulier la transaction de la slide précédente) ont été testées sur Oracle pour en vérifier leur bon fonctionnement.

6. Conclusion et bilan du projet

Dans ce projet, nous avons appliqué une grande partie des connaissances que nous avons apprises, ce qui peut ne pas être pleinement démontré dans le rapport. Le plus de connaissances que nous utilisons est dans la partie conception, en particulier le passage du modèle entité association au schéma relationnel. En plus, nous avons manipulé de nouvelles choses comme par exemple la JDBC.

Nous avons beaucoup appris de ce projet et nous sommes satisfait du travail qui répond aux demandes du cahier des charges. En particulier, notre base de données fonctionne comme nous le montrons dans la vidéo de démonstration de notre produit.

Enfin, notons que les habitudes d'écriture du code peuvent avoir un impact important sur l'efficacité du programme. Par conséquent, nous devons développer de bonnes habitudes d'écriture de code, supprimer le code redondant à tout moment et souvent reconstruire le code, ce qui peut grandement aider notre