

Fabien GUIHARD

fabien.guihard@etudiant.univ-rennes1.fr

Axel HOMERY

axel.homery@etudiant.univ-rennes1.fr

1) Introduction

Nous avons choisi comme domaine d'application l'industrie de la musique et plus particulièrement du rap. La grande majorité de nos données proviennent de Wikipédia, comme par exemple les données de certaines maisons de disque, ici "92i" disponible via <https://fr.wikipedia.org/wiki/92i>.

Tallac Records	
Filiale	92i Records (créé en 2015) 7 Corp (créé en 2018) Piraterie Music (créé en 2020) 92i Africa (créé en 2021)
Fondation	20 janvier 2004 ¹
Fondateur	Booba
Maison de disques	Capitol , Universal
Genre	Rap français , Gangsta rap , RnB
Pays d'origine	 France
Siège	49 Avenue du Général de Gaulle , 92360 Meudon ¹
modifier 	

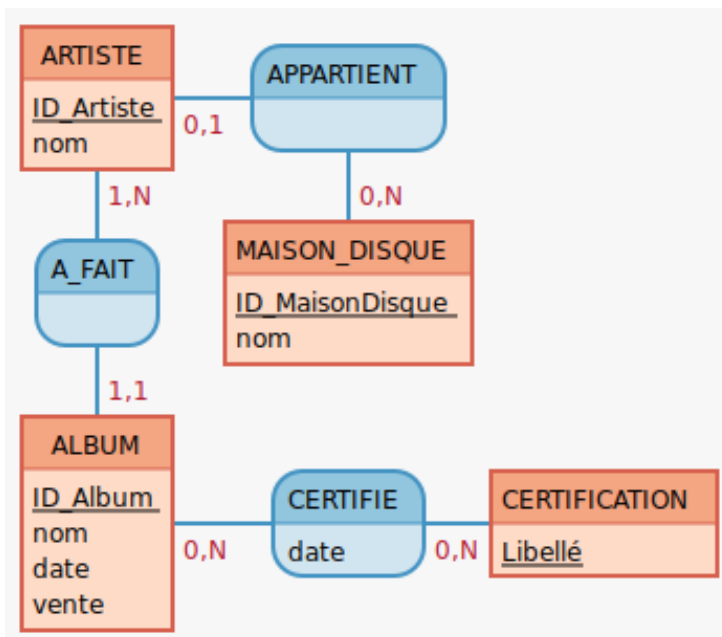
Capture d'écran du site de référence Wikipédia.

2) Modèle conceptuel (MCD)

MCD :

ALBUM : ID_Album, nom, date, vente, ID_Artiste
CERTIFICATION : Libellé
MAISON_DISQUE : ID_MaisonDisque, nom
ARTISTE : ID_Artiste, nom, ID_MaisonDisque

CERTIFIE, 0N ALBUM, 0N CERTIFICATION : date
APPARTIENT, 01 ARTISTE, 0N MAISON_DISQUE
A_FAIT, 1N ARTISTE, 11 ALBUM



Dessin représentant le MCD.

ENTITÉ

Entité ARTISTE :

- attribut id_Artiste* : INTEGER numéro de l'artiste, clé primaire.
- attribut nom : VARCHAR indiquant le nom ou le pseudonyme de l'artiste
- attribut id_MaisonDisque : clé étrangère (règle 2.2 merise car on a une cardinalité (0,1) vers l'entité MAISON_DISQUE.

Entité MAISON DISQUE :

- attribut id_MaisonDisque* : INTEGER numéro de la maison de disque, clé primaire.
- attribut nom : VARCHAR indiquant le nom de la maison de disque.

Entité CERTIFICATION :

- attribut libelle* : VARCHAR indiquant le type de certification, clé primaire

Entité ALBUM :

- attribut id_Album* : INTEGER numéro de de l'album, clé primaire.
- attribut nom : VARCHAR indiquant le titre de l'album.
- attribut date : YEAR indiquant l'année de sortie de l'album.
- attribut id_Artiste : clé étrangère (règle 2.1 merise car on a une cardinalité (1,1) vers l'entité ARTISTE).
- attribut vente : INTEGER comptabilisant le nombre de ventes de l'album.

ASSOCIATION

Association APPARTIENT :

- Cardinalité (0,1 \rightarrow 0,N) (ARTISTE \rightarrow MAISON_DISQUE) : Un artiste peut appartenir à 0 ou une maison de disque; Une maison de disques peut signer entre 0 et N artiste.
- La clé primaire de l'entité ARTISTE devient la clé primaire de l'association. La clé primaire de l'entité MAISON_DISQUE devient un attribut de l'association.

Association CERTIFIE :

- Cardinalité (0,N \rightarrow 0,N) (ALBUM \rightarrow CERTIFICATION) : Un album peut avoir entre 0 et N certification; Une certification peut certifier entre 0 et N albums.
- La clé primaire de l'entité ALBUM ainsi que la clé primaire de l'entité de CERTIFICATION viennent construire une clé *primaire composite* pour l'association.
- L'attribut date : YEAR indiquant la date pour laquelle la certification à été attribuée.

Association A FAIT :

- Cardinalité (1,N \rightarrow 1,1) (ARTISTE \rightarrow ALBUM) : Un artiste peut avoir fait 1 ou N albums; Un album est fait par un et un seul artiste.
- La clé primaire de l'entité ARTISTE vient migrer vers l'entité ALBUM en tant qu'attribut.

3) Schéma Relationnel (MLD)

ARTISTE(id_Artiste*, nom, id_MaisonDisque)

MAISON_DISQUE(id_MaisonDisque*, nom)

CERTIFICATION(libelle*)

ALBUM(id_Album*, nom, date, vente, id_Artiste)

CERTIFIE(id_Album*, libelle*, date)

APPARTIENT(id_Artiste*, id_MaisonDisque)

4) Schéma Physique (MPD)

Ce schéma ne contient pas la table APPARTIENT car la cardinalité (0,1) créant une table n'est pas prise en compte par le logiciel Mocodo.

```
CREATE TABLE `ARTISTE` (  
  `id_artiste` INTEGER,  
  `nom` VARCHAR(42),  
  `id_maisondisque` INTEGER,  
  PRIMARY KEY (`id_artiste`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `MAISON_DISQUE` (  
  `id_maisondisque` INTEGER,  
  `nom` VARCHAR(42),  
  PRIMARY KEY (`id_maisondisque`)
```

```

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `ALBUM` (
  `id_album` INTEGER,
  `nom` VARCHAR(42),
  `date` YEAR,
  `vente` INTEGER,
  `id_artiste` INTEGER,
  PRIMARY KEY (`id_album`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `CERTIFIE` (
  `id_album` INTEGER,
  `libellé` VARCHAR(42),
  `date` YEAR,
  PRIMARY KEY (`id_album`, `libellé`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

CREATE TABLE `CERTIFICATION` (
  `libellé` VARCHAR(42),
  PRIMARY KEY (`libellé`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

ALTER TABLE `ARTISTE` ADD FOREIGN KEY (`id_maisondisque`) REFERENCES
`MAISON_DISQUE` (`id_maisondisque`);
ALTER TABLE `ALBUM` ADD FOREIGN KEY (`id_artiste`) REFERENCES `ARTISTE`
(`id_artiste`);
ALTER TABLE `CERTIFIE` ADD FOREIGN KEY (`libellé`) REFERENCES `CERTIFICATION`
(`libellé`);
ALTER TABLE `CERTIFIE` ADD FOREIGN KEY (`id_album`) REFERENCES `ALBUM`
(`id_album`);

```

5) Peuplement des tables

ARTISTE : Nous avons un chanteur du nom de 'Orelsan' dont l'identifiant est 3 et qui appartient à la maison de disques 'Wagram Music' (identifiant 2).

ARTISTE(3, 'Orelsan', 2)

MAISON_DISQUE : La maison de disques nommée '92i' a un identifiant 1.

MAISON_DISQUE(1, '92i')

CERTIFICATION : Nous avons une certification nommée 'Or' est cela correspond à son identifiant.

CERTIFICATION('Or')

ALBUM : Nous avons un album dont le titre est "Futur" possédant l'identifiant 4; Cet album de Booba (ID=1) est sorti en 2012 et comptabilise 65300 ventes jusqu'à présent.

ALBUM(4, "Futur", 2012, 65300, 1)

6) Requêtes

A) une sélection avec projection

- Français : Tous les rappeurs qui sont dans la maison de disque '92i' (ID=1)
- Algèbre : $\pi \text{ ARTISTE}(\sigma(\text{id_MaisonDisque} = 1))$
- SQL : `SELECT * FROM ARTISTE WHERE id_MaisonDisque = 1`

B) une jointure

- Français : Liste des artistes accompagnés de leurs albums respectifs
- Algèbre : $\pi \text{ ARTISTE}(\sigma(\text{ALBUM.id_Artiste} = \text{ARTISTE.id_Artiste}(\text{ARTISTE} \bowtie \text{ALBUM})))$
- SQL :
`SELECT * FROM ALBUM INNER JOIN ARTISTE ON ALBUM.id_artiste = ARTISTE.id_artiste`

C) une moyenne sur l'intégralité d'un attribut

- Français : (ventes) TODO : ajouter des albums – moyenne des ventes d'albums de l'artiste 'Booba' (ID=1). On souhaite afficher l'id de cette artiste avec son nombre moyen de ventes d'album.
- Algèbre :
- SQL :

```
SELECT id_Artiste,AVG(vente) FROM ALBUM WHERE id_Artiste = 1
```

D) un regroupement avec calcul

- Français : Nombre total de ventes d'albums pour chaque artiste.
- Algèbre :
- SQL :
`SELECT id_Artiste, SUM(vente) FROM ALBUM GROUP BY id_Artiste`

E) une différence

- Français : L'ensemble des albums non certifié
- SQL :

```
SELECT id_Album FROM ALBUM  
WHERE id_Album NOT IN (SELECT id_Artiste FROM CERTIFIE);
```

Algèbre :

$$\text{ALBUM} - (\text{ALBUM} \times \text{CERTIFIE})$$

F) une division

- Français : Les albums qui ont **toutes** les certifications.
- Algèbre :

$$[\pi(\text{id_Album}, \text{libelle}) (\text{CERTIFIE})] \div [(\pi(\text{libelle}) (\text{CERTIFICATION}))]$$

- SQL :