

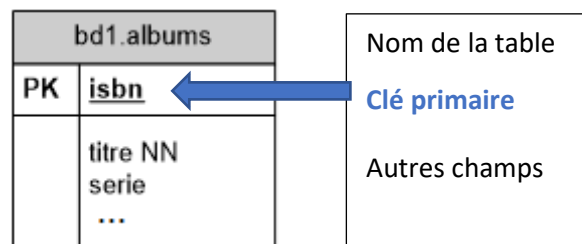
La représentation graphique d'un schéma de base de données

1. Conventions

Pour représenter un schéma, nous utilisons les conventions suivantes :

1. Utiliser des noms de table entièrement en minuscules.
2. Utiliser des noms de table au pluriel.
3. Utiliser des noms de champs entièrement en minuscules.
4. Utiliser « _ » pour séparer les mots.
5. Ne mettre aucun caractère accentué dans les noms de table et de champ.
6. Souligner la clé primaire et indiquer PK à gauche.
7. Indiquer le mot-clé FK entre () à côté d'une clé étrangère.
8. Indiquer NN pour les champs not null.
9. Respecter le vocabulaire métier (le vocabulaire de l'énoncé).
10. Vérifier la cohérence des clés étrangères (FK) et des relations.
11. Mentionner le mot-clé unique lorsque cela est nécessaire.
12. Mentionner les énumérés entre { }.

Graphiquement, une table est représentée de la sorte :



- Lorsqu'un seul champ compose la clé primaire, il est indiqué uniquement dans l'espace réservé à cet effet.

Cela correspond au CREATE TABLE :

```
CREATE TABLE bd1.albums
(
  isbn          char (14) PRIMARY KEY,
  titre        character varying (50) NOT NULL,
  serie        character varying (20),
  ...
)
```

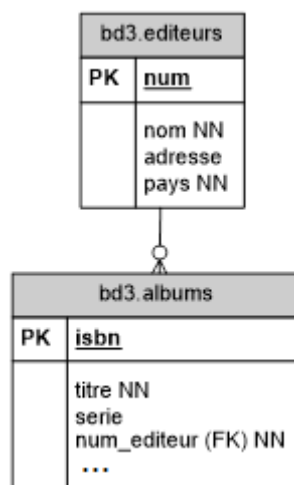
- Lorsque plusieurs champs composent la clé primaire, leur concaténation est indiquée dans l'espace réservé à cet effet et chaque champ est repris avec ses propriétés dans la liste des champs.

table1	
PK	<u>cle_partie1,</u> <u>cle_partie2</u>
	cle_partie1 NN cle_partie2 NN ...

Cela correspond au CREATE TABLE :

```
CREATE TABLE table1
(
  cle_partie1          char (4) NOT NULL,
  cle_partie2          char (3) NOT NULL,
  ...,
  PRIMARY KEY (cle_partie1, cle_partie2)
)
```

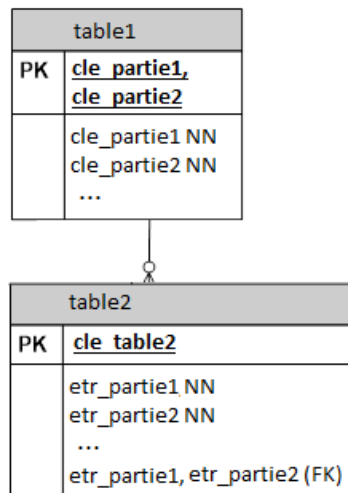
- Lorsqu'un seul champ compose la clé étrangère, le mot-clé FK est indiqué immédiatement derrière ce champ.



Cela correspond au CREATE TABLE :

```
CREATE TABLE bd3.albums
(
  isbn          char (14) PRIMARY KEY,
  titre         character varying (50) NOT NULL,
  serie         character varying (20),
  num_editeur integer NOT NULL REFERENCES bd3.editeurs (num),
  ...,
)
```

- Lorsque plusieurs champs composent la clé étrangère, chaque champ est repris avec ses propriétés dans la liste des champs et leur concaténation est ensuite indiquée avec le mot-clé FK.



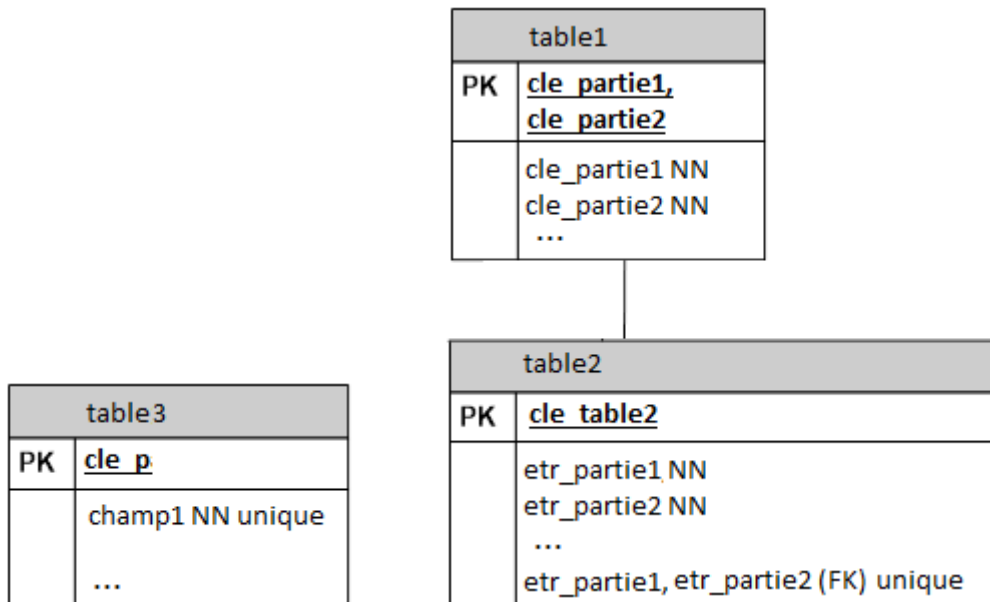
Cela correspond au CREATE TABLE :

```

CREATE TABLE table1
(
  cle_partie1          char(4) NOT NULL,
  cle_partie2          char(3) NOT NULL,
  ...,
  PRIMARY KEY (cle_partie1, cle_partie2)
)

CREATE TABLE table2
(
  cle_table2          integer PRIMARY KEY,
  etr_partie1         char(4) NOT NULL,
  etr_partie2         char(3) NOT NULL,
  ...,
  FOREIGN KEY (etr_partie1, etr_partie2)
  REFERENCES table1 (cle_partie1, cle_partie2)
)
  
```

- De la même manière, lorsqu'un champ est unique, le mot-clé unique est indiqué immédiatement après ce champ. Lorsque la concaténation de plusieurs champs est unique, chaque champ est repris avec ses propriétés dans la liste des champs et leur concaténation est ensuite indiquée avec le mot-clé unique. Dans l'exemple ci-dessous, on remarque que le mot-clé unique est combiné avec le mot-clé FK.



Cela correspond au CREATE TABLE :

```
CREATE TABLE table1

(cle_partie1          char(4) NOT NULL,
 cle_partie2          char(3) NOT NULL,
 ...,
 PRIMARY KEY (cle_partie1, cle_partie2)
)

CREATE TABLE table2

( cle_table2          integer PRIMARY KEY,
  etr_partie1         char(4) NOT NULL,
  etr_partie2         char(3) NOT NULL,
  ...,
  FOREIGN KEY (etr_partie1, etr_partie2)
  REFERENCES table1 (cle_partie1, cle_partie2)
  UNIQUE (etr_partie1, etr_partie2)
)
```

2. L'organisation graphique du DSD

Certaines règles simples permettent de structurer le dessin pour qu'il devienne un support fiable à la conception.

1. On placera en haut (à la périphérie) du dessin les entités les plus indépendantes et en bas (au centre) du dessin les entités les plus dépendantes. Généralement, les entités les plus dépendantes sont les plus volumineuses et les plus mouvementées. Souvent on *descend* vers les tables volumineuses.
2. On évitera de croiser les relations. Il faut éviter les schémas spaghetti ou jeu de piste.
3. Pour les schémas très importants, on peut utiliser des couleurs différentes pour mettre en valeur des tables de natures différentes ou utilisées dans différentes applications.
4. On peut également créer plusieurs schémas en reprenant les tables communes.

Indépendant



Dépendant

