



Yacine LAGHROUCHE  
&  
Tom GONTARD




COMPTE RENDU

**Conception et exploitation  
des bases de données**

---

24/11/2025





# Compte Rendu CEBD

## I-CONCEPTION UML ET NORMALISATION:

### Question 1 : identifions les dependances fonctionnelles des deux relations LesEpreuves et LesSportifs :

**Soit le schéma relationnel suivant :**

**LesEpreuves** (numEp, nomEp, formeEp, categorieEp, nbSportifsEp, dateEp, nomDi) /\* ∈ LesEpreuves ⇔ no est le numéro d'épreuve du nom n, forme f (individuelle, par équipe ou par couple), catégorie c (féminin, masculin ou mixte), un nombre de sportifs nb et une date d. L'épreuve fait partie de la discipline di \*/

**LesSportifsEQ** (numSp, nomSp, prenomSp, pays, categorieSp, dateNaisSp, numEq) /\* ∈ LesSportifs ⇔ no est le numéro de sportif, avec un nom n, un prénom p, un pays pa une date de naissance d et une catégorie c (feminin ou masculin). Il est inscrit dans l'équipe ne. \*/

On suppose :

- numEp identifie de manière unique une épreuve.
- numSp identifie de manière unique un sportif.
- Le couple (nomSp, prenomSp) identifie également de manière unique un sportif.

### **I-Dépendances fonctionnelles supposées :**

#### **Pour LesEpreuves:**

Dépendance essentielle :

numEp → nomEp, formeEp, categorieEp, nbSportifsEp, dateEp, nomDi (numEp clé)

Dépendance plus ambiguës :

(nomEp, dateEp, nomDi, formeEp, categorieEp) → numEp (ambiguë si plusieurs épreuves portant le même nom dans une même discipline à la même date)

formeEp, categorieEp → nbSportifsEp (potentiellement vrai si la forme détermine systématiquement le nombre de sportifs)

#### **Pour LesSportifsEQ:**

Dépendance essentielle :

numSp → nomSp, prenomSp, pays, categorieSp, dateNaisSp, numEq

nomSp, prenomSp → numSp, pays, categorieSp, dateNaisSp, numEq

numEq → pays

On constate notamment la transitivité : numSp → numEq → pays, donc numSp → pays



# Compte Rendu CEBD

## II-Analyse des formes normales

### LesEpreuves :

La clé candidate de la relation est numEp, On constate une atomicité des valeurs donc 1NF, puisque la clé est simple, il n'y a donc pas de dépendance partielle, la relation est donc en 2NF, On a aucun attribut non-primaire qui en détermine un autre donc la relation est en 3NF si numEp est la clé

BCNF : si la seule DF est ( numEp  $\rightarrow$  nomEp, formeEp, categorieEp, nbSportifsEp, dateEp, nomDi ) alors la relation est en BCNF.

### LesSportifsEQ:

La clé candidate de la relation est numSp et le couple nomSp, prenomSp. On constate une atomicité des valeurs donc 1NF, puisque la clé numSp est simple, il n'y a donc pas de dépendances fonctionnelles, la relation est en 2NF. On a numSp  $\rightarrow$  numEq et numEq  $\rightarrow$  pays  $\Rightarrow$  numSp  $\rightarrow$  pays. Cela indique une redondance : pays dépend transitivement de numSp via numEq, même si dans le schéma pays est directement sous numSp, la contrainte d'équipe implique cette transitivité et donc on a une violation possible de 3NF.

## III-Transformation en 3NF/BCNF

### LesEpreuves :

LesEpreuves est déjà en forme BCNF, aucun changement n'est à prévoir.

**LesEpreuves** (numEp, nomEp, formeEp, categorieEp, nbSportifsEp, dateEp, nomDi)

### LesSportifsEQ :

On décompose pour éliminer la violation de 3NF :

numSp  $\rightarrow$  nomSp, prenomSp, pays, categorieSp, dateNaisSp, numEq (2NF)

nomSp, prenomSp  $\rightarrow$  numSp, pays, categorieSp, dateNaisSp, numEq (2NF)

numEq  $\rightarrow$  pays

**LesSportifs** (numSp, nomSp, prenomSp, pays, categorieSp, dateNaisSp)

**Equipe** (numEq, pays)

avec numSp comme clé primaire de LesSportifs et (nomSp, prenomSp) clé alternative. Mais également numEq  $\rightarrow$  pays géré dans la relation Equipe.

## IV-Fermeture des ensembles d'attributs

{ numSp }<sup>+</sup> = { numSp, nomSp, prenomSp, pays, categorieSp, dateNaisSp }

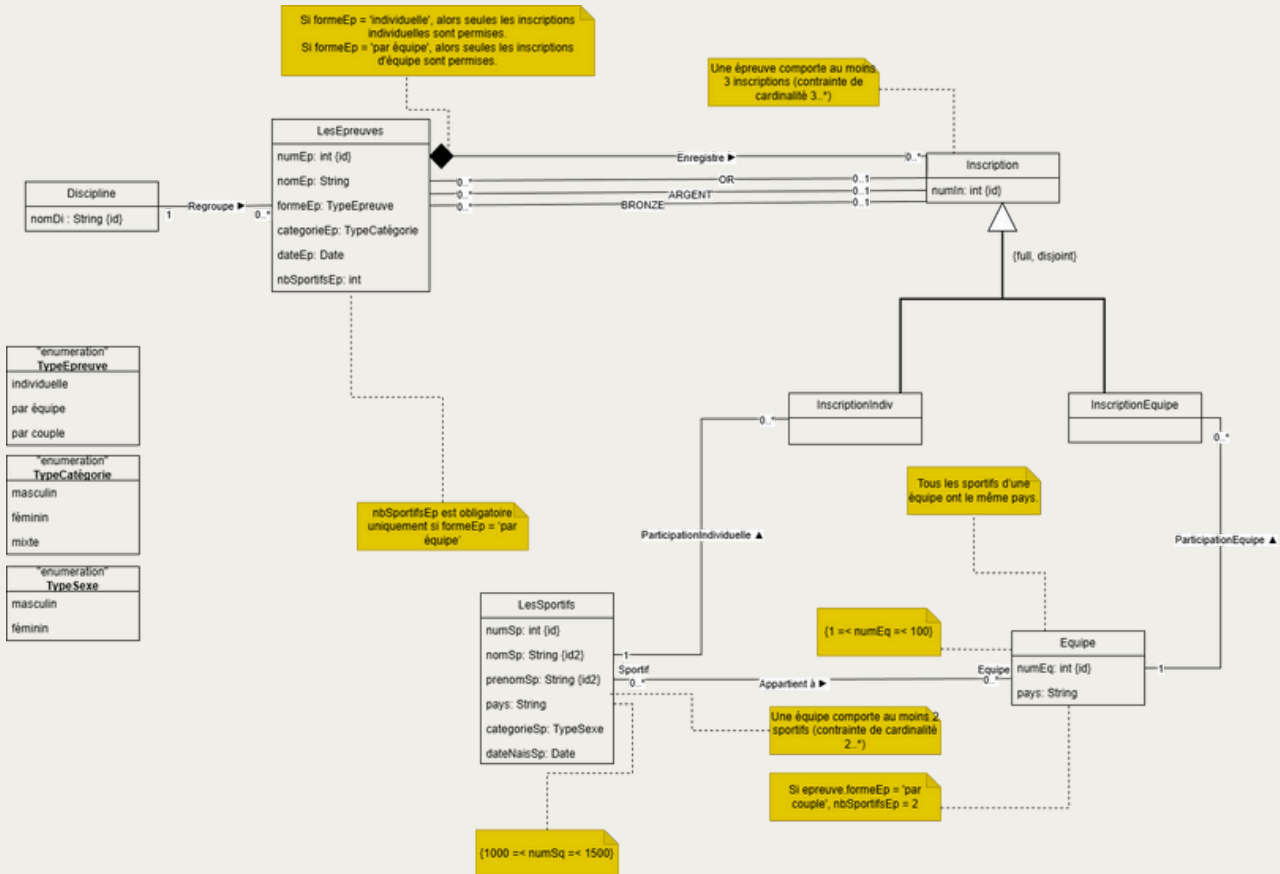
{ nomSp, prenomSp }<sup>+</sup> = { numSp, nomSp, prenomSp, pays, categorieSp, dateNaisSp }

La clé candidate de la relation est donc numSp, et le couple (nomSp, prenomSp) est une clé alternative



# Compte Rendu CEBD

## Question 2: Réalisons un diagramme de classe UML afin de modéliser les données de l'énoncé :



## II-IMPLEMENTATION:

### Question 3 : Dédisons un schéma relationnel du diagramme UML :

#### I- Schema relationnel proposé :

##### Discipline (nomDi)

/\* ∈ Discipline ⇔ nomDi est le nom de la discipline \*/

##### LesEpreuves (numEp, nomEp, formeEp, categorieEP, dateEp, nbSportifsEp, nomDi)

/\* ∈ LesEpreuves ⇔ no est le numéro d'épreuve du nom n, forme f (individuelle, par équipe ou par couple), catégorie c (féminin, masculin ou mixte), un nombre de sportifs nb et une date d. L'épreuve fait partie de la discipline di \*/

##### Or (numEp, numIn)

/\* ∈ Or ⇔ numEp est le numéro de l'épreuve, numIn est le numéro d'inscription du gagnant de la médaille \*/

##### Argent (numEp, numIn)

/\* ∈ Argent ⇔ numEp est le numéro de l'épreuve, numIn est le numéro d'inscription du gagnant de la médaille \*/



# Compte Rendu CEBD

**Bronze**(numEp, numIn)

/\* ∈ Bronze ⇔ numEp est le numéro de l'épreuve, numIn est le numéro d'inscription du gagnant de la médaille \*/

**Inscription**(numIn, numEp)

/\* ∈ Inscription ⇔ numEp est le numéro de l'épreuve, numIn est le numéro d'inscription de l'équipe ou du sportif \*/

**inscriptionIndiv**(numIn, numEp, numSp)

/\* ∈ inscriptionIndiv ⇔ numIn est le numéro d'inscription, numSp est le numéro du sportif inscrit \*/

**inscriptionEquipe**(numIn, numEp, numEq)

/\* ∈ inscriptionIndiv ⇔ numIn est le numéro d'inscription, numEq est le numéro de l'équipe inscrite \*/

**LesSportifs**(numSp, nomSp, prenomSp, pays, categorieSp, dateNaisSp)

/\* ∈ LesSportifs ⇔ no est le numéro de sportif, avec un nom n, un prénom p, un pays pa une date de naissance d et une catégorie c (feminin ou masculin). \*/

**Equipe**(numEq, pays)

/\* ∈ Equipe ⇔ numEq est le numéro de l'équipe, pays est le nom du pays de l'équipe \*/

**AppartientA**(numSp, numEq)

/\* ∈ AppartientA ⇔ numSp est le numéro du sportif, numEq est le numéro de l'équipe inscrite \*/

## II- Vues proposées :

(view) **Resultats**(numEp, Or, Argent, Bronze) → Donne les médailles pour chaque épreuve.

(view) **LesNbsEquipiers**(numEq, nbEquipiersEq) → Taille des équipes.

(view) **ClassementPays**(pays, nbOr, nbArgent, nbBronze) → Classement des nations.

(view) **AgeMoyEquipesOr**(numEq, ageMoyEq) → Âge moyen de l'équipe gagnante.

(view) **LesAgesSportifs**(numSp, nomSp, prenomSp, pays, categorieSp, dateNaisSp, ageSp)

→ Calcul de l'âge des sportifs.

## III- Contraintes de domaine :

domaine (numEq) = [1, 100]

domaine (numSp) = [1000, 1500]

domaine (dateNaisSp) = date(dateEp) = Date (exemple 17/11/2025 )

domaine (formeEp) = {"individuelle", "par equipe", "par couple"}

domaine (categorieEp) = {"feminin", "masculin", "mixte"}

domaine (categorieSp) = {"feminin", "masculin"}

domaine (nomDi) = domaine (nomEp) = domaine (nomSp) = domaine (prenomSp) = domaine (pays) = chaines de caractères

domaine (numEp) = domaine (numSp) = domaine (nbSportifsEp) = domaine (numEq) = domaine (numIn) = domaine (or) = domaine (argent) = domaine (bronze) = entier > 0



# Compte Rendu CEBD

## IV- Contraintes d'intégrité référentielle

$\text{LesEpreuves}[\text{nomDi}] \subseteq \text{Discipline}[\text{nomDi}]$   
 $\text{Inscription}[\text{numEp}] \subseteq \text{LesEpreuves}[\text{numEp}]$   
 $\text{InscriptionIndiv}[\text{numSp}] \subseteq \text{LesSportifs}[\text{numSp}]$   
 $\text{InscriptionEquipe}[\text{numEq}] \subseteq \text{Equipe}[\text{numEq}]$   
 $\text{Equipe}[\text{pays}] \subseteq \text{LesSportifs}[\text{pays}]$   
 $\text{AppartientA}[\text{numSp}] \subseteq \text{LesSportifs}[\text{numSp}]$   
 $\text{AppartientA}[\text{numEq}] \subseteq \text{Equipe}[\text{numEq}]$   
 $\text{Or}[\text{numEp}, \text{numIn}] \subseteq \text{Inscription}[\text{numEp}, \text{numIn}]$   
 $\text{Argent}[\text{numEp}, \text{numIn}] \subseteq \text{Inscription}[\text{numEp}, \text{numIn}]$   
 $\text{Bronze}[\text{numEp}, \text{numIn}] \subseteq \text{Inscription}[\text{numEp}, \text{numIn}]$

## V- Contraintes disjonction / couverture :

$\text{InscriptionIndiv}[\text{numIn}] \cap \text{InscriptionEquipe}[\text{numIn}] = \emptyset$  (disjoint)  
 $\text{InscriptionIndiv}[\text{numIn}] \cup \text{InscriptionEquipe}[\text{numIn}] = \text{Inscription}[\text{numIn}]$  (full)

## VI-Contraintes d'intégrité supplémentaires :

numSp est compris entre 1000 et 1500. ( $1000 \leq \text{numSp} \leq 1500$ )  
numEq est compris entre 1 et 100. ( $1 \leq \text{numEq} \leq 100$ )  
Une équipe comporte au minimum 2 sportifs. (contrainte de cardinalité 2..\*)  
Tous les sportifs membre d'une même équipe proviennent du même pays.  
Une inscription ne peut être reliée qu'à une seule médaille par épreuve.  
Une épreuve comporte au moins 3 inscriptions. (contrainte de cardinalité 3..\*)  
Il n'y a pas d'ex-aequo, toutes les médailles sont distribuées.  
Si une médaille est distribuée, alors nécessairement, la médaille du rang suivant est distribuée.  
nomDi ne peut pas être NULL dans la relation LesEpreuves.  
Si la forme de l'épreuve (formeEp) est "individuelle", alors seules les inscriptions individuelles sont permises.  
Si la forme de l'épreuve (formeEp) est "par équipe", alors seules les inscriptions d'équipe sont permises.  
Le nombre de sportifs par épreuve (nbSportifsEp) est obligatoire uniquement si la forme de l'épreuve (formeEp) est "par équipe".  
Si la forme de l'épreuve (epreuve[formeEp]) est par couple alors le nombre de sportifs par épreuve (nbSportifsEp) est égal à 2.

## Question 4 : implementation de la base de donnée & script SQL :

Consulter la page git :

<https://github.com/Laghrouy/CEBD-JO>