

Projet: A propos de la Gestion des Jeux Olympiques

Duy Minh LE, 12315392

1. Conception UML et Normalisation

Question 1

On rappelle le schéma relationnel fourni:

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

/ <no, n, f, c, nb, da, di> ∈ LesEpreuves ⇔ no est le numéro d'épreuve du nom n, forme (individuelle, par equipe 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)

/ <no, n, p, pa, d, c, ne> ∈ 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 peut en suite en deduire les dependances fonctionnelles

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

numEp → nomEp, formeEP, categorieEP, nbSportifsEp, dateEp, nomDi

On voit bien que **nomEp** determine tous les autres attributs, donc une clé. C'est en fait, la seule clé, on détermine bien que cette relation est de forme **BCNF**.

- **LesSportifsEQ** (numSp, nomSp, prenomSp, pays, categorieSp, dateNaisSp, numEq)

numSp → nomSp, prenomSp, pays, categorieSp, dateNaisSp

numEq → pays

Ici on a **numEq** qui ne determine que pays, donc via la regle d'augmentation, (**numSp, numEq**) est une clé. Vu que tous les attributs sauf numEq sont dependants a numSp, cette relation est de forme **1NF**.

Ici le numero d'équipe numEq est dans la clé, bien qu'il y aie des sportifs sans équipe. On va ultimativement attaquer ce problème en transformant le schéma

On transforme donc la relation **LesSportifsEQ** en la coupant par 2:

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

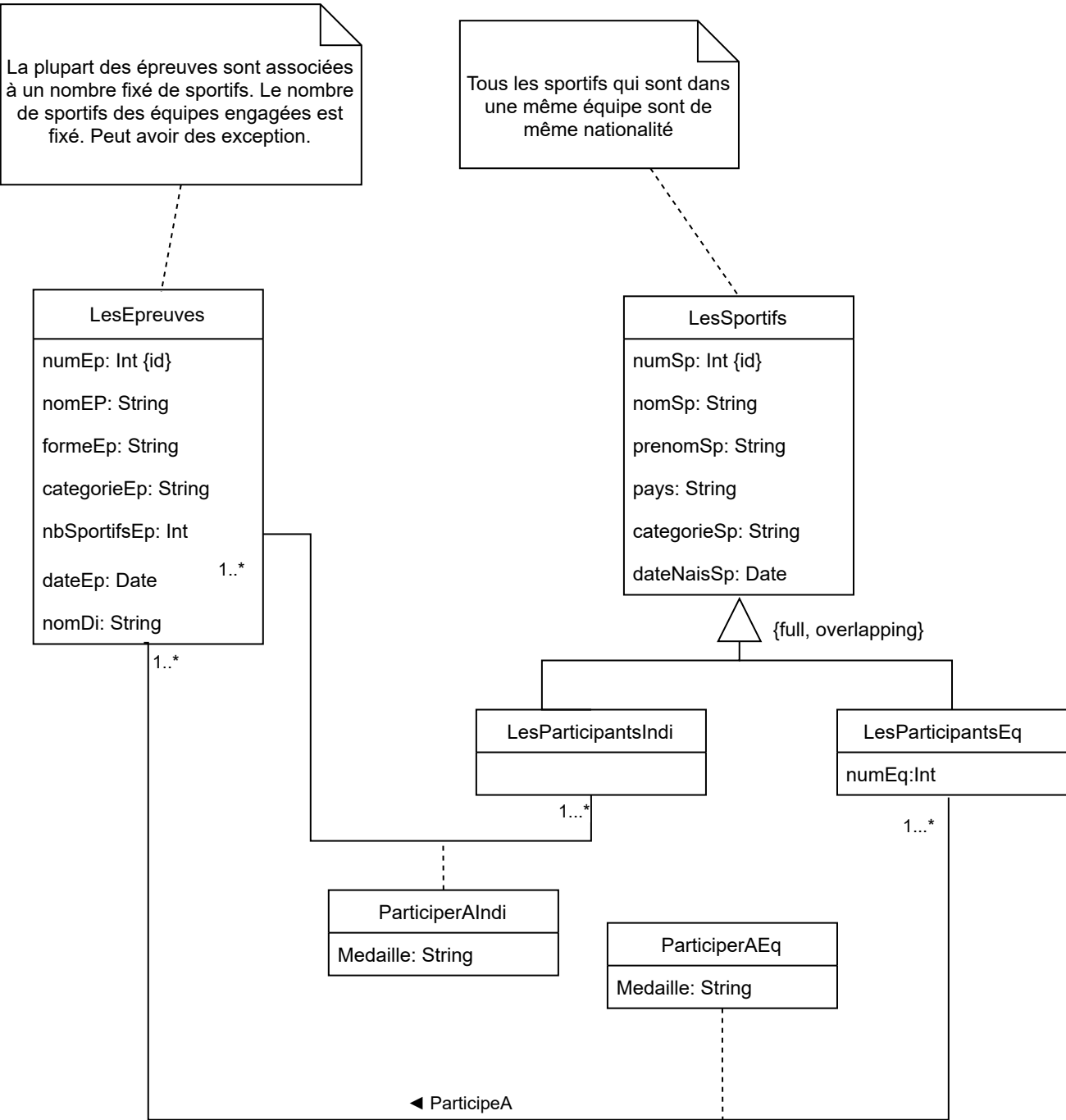
numSp → nomSp, prenomSp, pays, categorieSp, dateNaisSp

numSp est facilement la clé dans ce cas, et cette relation est de forme **BCNF**.

- **LesParticipants** (numSp, numEQ)
Pas de DF donc **BCNF** .

Question 2

On va maintenant modifier le schema obtenu a la question precedente pour completer la BD.



2. Implementation

Question 3

Schema propose

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

- **LesSportifs** (numSP, nomSp, prenomSp, pays, categorieSp, dateNaisSp)
- **LesParticipantsIndi**(numSp)
- **LesParticipantsEq**(numSp, numEq)
- **ParticiperAIndi**(numSp, numEp, medaille)
- **ParticiperAEq**(numEq, numEp, medaille)

Domaines

- domaine (dateNaisSp) = date(dateEp) = Date
- domaine (formeEp) = {'individuelle', 'par equipe', 'par couple'}
- domaine (categorieEp) = {'feminin', 'masculin', 'mixte'}
- domaine (categorieSp) = {'feminin', 'masculin'}
- domaine (medaille) = {'Gold', 'Silver', 'Bronze'}
- domaine (nomDi) = domaine (nomEp) = domaine (nomSp) = domaine (prenomSp) = domaine (pays) = chaines de caracteres
- domaine (numSp) = domaine (numEp) = domaine (nbSportifsEp) = entier > 0

Contraintes d'integralite

- LesparticipantsIndi[numSp] \subseteq LesSportifs[numSp]
- LesparticipantsEq[numSp] \subseteq LesSportifs[numSp]
- LesParticipantsIndi[numSp] \cup LesParticipantsEq[numSp] = LesSportifs[numSp]
- ParticiperAIndi[numSp] = Lesparticipantsindi[numSP]
- ParticiperAIndi[numEp] \subseteq LesEpreuves[numEp]
- ParticiperAEq[numEq] = LesParticipantsEq[numEq]
- ParticiperAEq[numEp] \subseteq LesEpreuves[numEp]
- ParticiperAEq[numEp] \cap ParticiperAIndi[numEp] = \emptyset