IFT2935 Hiver 2020

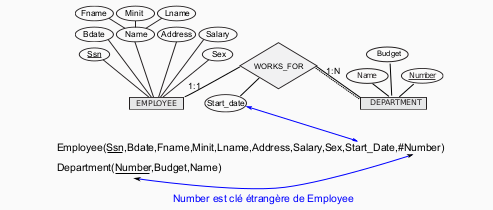
Modèle/Schéma relationnel

Traduction des entités

* Toute entité devient une relation
* Ses attributs correspondent aux propriétés simples de l’entité
* La clé primaire traduit un identifiant de l’entité
* Pour les attributs composites, chaque composante simple devient un attribut

Association de dimension 2 avec lien 1:1

* Une clé étrangère correspondant à la clé primaire de l’entité en association
* Les Attributs de l’association



Coupe\_Du\_Monde (**coupe\_id**, nom\_coupe, pays, dateDebut, dateFin)

* Chaque coupe du monde va augmenter de 1 à chaque évènement, alors nous allons utiliser cela comme clé primaire. Son nom (ex. FIFA), quel pays les accueille cette année-là, la date début et de fin.

Equipe\_Foot (**equipe\_id**, #coupe\_id**,** #entraineur\_id, pays\_representant, position\_final)

* Une équipe va avoir un ID. Cet ID va être unique pour chaque équipe de chaque année différente. Chaque équipe a fait partie d’une coupe, a un entraineur chef, elle représente un pays et a une position finale dans le tournoi.
* Puisque la relation est 1..1 pour equipe\_foot et participe. Nous pouvons la mettre comme clé étrangère dans cette relation. Association de dimension 2 avec lien 1:1

Personne (**personne\_id**, nom, prenom, date\_de\_naissance, pays\_natal, sexe)

* Nous avons choisi de faire de l’héritage pour les personnes. Chaque profession a leurs propres attributs importants à eux. Les personnes on des attribut assez générales, nom, prénom, date de naissance, pays natal, genre.

Joueur (#**personne\_id,** premier\_match\_prof)

* Unique a eu, les joueurs ont un numéro de joueurs (équipe professionnel) particulier a eu. Leur premier match dans une League professionnel.

Entraineur (#**personne\_id***, entraineur\_depuis*)

* La date de leur premier match en tant qu’entraineur

Collaborateur (#**personne\_id**, expertise)

* Expertise d’un collaborateur. Ex. Medecin, physio, diététiste, etc.

Arbitre (#**personne\_id***, arbitre\_depuis*)

* Date début de carrière arbitre

Staff\_technique ***(#equipe\_id***, ***#collaborateur\_id***)

* Cette relation représente qui sont les collaborateurs qui assiste l’équipe

Joueur\_Equipe (***#joueur\_id***, ***#equipe\_id***, position, numero\_dossard, equipe\_prof)

* Cette relation représente laquelle joueur a joué pour quelle équipe. Lorsque nous avons l’équipe plus le jouer, alors nous pouvons accéder a d’autre information comme sa position au sein de l’équipe, numéro de joueur assigner (peu changer de celui de son équipe car seulement une personne peut avoir un numéro à la fois dans cette équipe et plusieurs joueurs viennent de différentes équipes pour jouer leurs pays.), equipe\_professionelle lors de cette coupe.

Arbitre\_Match (***#arbitre\_id***, ***#matchFoot\_id,*** type\_arbitre)

* Représente qui a arbitré quel match, si nous avons l’arbitre plus le match, alors nous pouvons savoir s’il était arbitre principal ou assistant.

Sanction (**sanction\_id**, #joueur\_id, #arbitre\_id, #match\_id, couleur, description)

* Une sanction est donnée par un arbitre a un joueur dans un match en particulier. La clé primaire est un incrementateur car il n’y a pas d’autre façon unique de définir une sanction car un même joueur peut recevoir plusieurs sanctions dans le même match par le même arbitre de même couleur

Match\_Foot (**match\_id**, *#coupe\_id*, *#equipe1\_ID*, *#equipe2\_ID*, *#stade\_id*, date, rang, score\_1, score\_2)

* La même approche que pour les équipes ici. Chaque Match joué aurait un identificateur unique. Un match a une date, un rang dans le tournoi (ex. 1e round, demi final, final, etc), équipe1 plus son score, équipe 2 plus son score. Nous avons pu absorber les association accueil, joue, organise car elles sont des associations de dimension 2 avec lien 1:1

Stade (**stade\_id**, nom, pays, adresse, capacite, année\_construction)

* Un stade a son identificateur, un nom (ex. Centre Bell), un pays, une adresse et une capacité maximale. Nous allons considérer adresse comme 1 string qui contiendra tout l’adresse au lieu de plusieurs string différent contenant rue. Ville, etc

Pour les associations avec des liens 1:N, on a créé une entité pour l’association même et les clés des entités associées sont incluses dans cette nouvelle entité comme clé étrangère. Pour les entités ayant une association de type 1:1, on n’a qu’à ajouter la clé d’une des entités comme clé étrangère dans l’autre entité.

***Schema relationnel final***

\*\*Entité\*\*

Coupe\_Du\_Monde(**coupe\_id**, nom\_coupe, pays, dateDebut, dateFin)

Equipe\_Foot (**equipe\_id**, #coupe\_id**,** *#entraineur\_id*, pays\_representant, position\_final)

Personne (**personne\_id**, nom, prenom, date\_de\_naissance, pays\_natal, sexe)

Joueur (#**personne\_id,** premier\_match\_prof)

Entraineur (#**personne\_id***, entraineur\_depuis*)

Collaborateur (#**personne\_id**, expertise)

*Arbitre (#***personne\_id***, arbitre\_depuis)*

*Stade (****stade\_id****, nom, pays, adresse, capacite, année\_construction)*

*Match\_Foot (****match\_id****, #coupe\_id, #equipe1\_ID, #equipe2\_ID, #stade\_id,* date, rang, score\_1, score\_2)

*Sanction (***#joueur\_id**, **#arbitre\_id**, **#match\_id**, couleur, description*)*

*\*\*Association entre entité\*\**

*Staff\_technique* ***(#equipe\_id****, #****collaborateur \_id****)*

*Joueur\_Equipe (****#joueur\_id****,* ***#equipe\_id****,* position, numero\_dossard, equipe\_prof)

*Arbitre\_Match (****#arbitre\_id****,* ***#match\_id****,* type\_arbitre)

***#3 La normalisation***

Coupe\_du\_monde ( **coupe\_id**, nom\_coupe, pays, date\_debut, date\_fin)

Df = { coupe\_id -> nom\_coupe, pays, date\_debut, date\_fin }

Les coupes comment a 1 et augmente progressivement, alors cela est un candidat parfait pour une clé de notre relation. Elle définit le nom de cette coupe dans quel pays elle a eu lieu et les dates de debut et fin.

Equipe\_foot ( **equipe\_id,** *#coupe\_id, #entraineur\_id*, pays\_representant, position\_final)

Df = { equipe\_id -> #entraineur\_id, pays\_representant, position\_final)

Puisque que chaque equipe a un identificateur unique. Ceci devient le candidat parfait pour être le déterminant. Donc l’équipe\_id determine sont entraineur, quel pays elle représente, sa position finale. Entraineur ID est une foreign key qui pointeur vers l’identificateur d'Entraineur qui est personne\_id.

Match\_Foot ( **match\_id**, *#coupe\_id, #stade\_id, #equipe1\_id, #equipe2\_id*, score\_1, score\_2, date, rang)

Df= { match\_id -> coupe\_id, stade\_id, equipe1\_id, equipe2\_id, score\_1, score\_2, date, rang)

Avec la clé du match (match\_id) nous avons accès à toutes les informations que le catégorise. Quelle coupe faisait-il parti, les équipe qui s'affrontait, leur score, la date, le rang de ce match.

Stade ( **stade\_id**, nom, adress, capacité, pays)

Df = { stade\_id -> nom, adresse, capacite, pays)

Chaque entre d’un stade va avoir un numéro identificateur unique. Donc déterminant de la relation qui détermine son nom, adresse, capacite et pays.

Personne ( **personne\_id,** nom, prenom, date\_de\_naissance, pays\_natal, sex)

Df ={ personne\_id -> nom, prenom, date\_de\_naissance, pays\_natal, sexe)

Comme un SSN aux yeux de la fédération/ notre BDSM une personne a un identificateur unique. Chaque personne a un nom, prénom, adresse, date de naissance, pays natal et un sexe.

Joueur ( personne\_id, numero\_dossard\_personnel, premier\_match\_prof)

Df = { personne\_id -> numero\_dossard\_personnel, premier\_match\_prof)

Un joueur hérite de Personne alors dans la relation Joueur, avec son identificateur de personne nous pouvons accéder à son numéro dossard dans sa carrière, quand il a commencé à jouer au foot professionnellement.

Entraineur ( personne\_id, entraineur\_depuis)

Df = { personne\_id -> entraineur\_depuis }

Comme joueur sauf que l’attribut est la date de son premier match en tant que coach. De cette façon même un ancien joueur peux être entraineur et être dans la même BDD

Collaborateur ( personne\_id, expertise)

Df = { personne\_id -> expertise)

Suit la même logique Joueur et entraineur.

Arbitre ( personne\_id, arbitre\_depuis )

Df = { personne\_id -> arbitre\_depuis)

Même logique que les 3 précédents.

*Sanction (sanction\_id, j*oueur\_id, arbitre\_id, match\_id, couleur, description*)*

Df = {sanction\_id -> joueur\_id, arbitre\_id, match\_id, couleur, description}

Nous avons besoin d’un id unique car un même joueur peut recevoir plusieurs sanctions dans le même match par le même arbitre de même couleur. Alors, c’est la seul facon d’identifier les sanctions

*Staff\_technique* ***(#equipe\_id****, #****collaborateur \_id****)*

*Df = { }*

*Aucune DF car il n’y a pas d’attribut particulier à cette association. Cependant nous avons besoin de cette relation pour joindre les colaborateur au reste de la BDD par jointure.*

*Joueur\_Equipe (****#joueur\_id****,* ***#equipe\_id****,* position, numero\_dossard, equipe\_prof)

Df = { joueur\_id, equipe\_id -> position, numero\_dossard, equipe\_prof }

La raison nous avons besoin des 2 clés pour définir position, dossard et équipe prof est parce qu’un même joueur peut jouer dans plusieurs équipes et une équipe a plusieurs joueurs. La raison pour laquelle nous avons besoin de joueur\_id + equipe\_id pour définir position, dossard, et équipe professionnelle est que la position, le dossard et l’équipe professionnel actuelle au moment de la coupe peuvent changer entre chaque coupe.

*Arbitre\_Match (****#arbitre\_id****,* ***#match\_id****,* type\_arbitre)

Df = { arbitre\_id, match\_id -> type\_arbitre }

Un peu dans la même optique que joueur\_equipe. Il y a plusieurs arbitres dans un match et un arbitre peu arbitrer plusieurs matchs. Alors nous avons besoin des 2 clés étrangères pour définir la relation. A partir de ça, nous pouvons savoir si l’arbitre était principal ou assistant.

Nous avons fait attention de créer notre modèle entité-association le plus normal possible pour rendre la table le plus maintenable possible. Il n’y a pas de normalisation particulière à faire à partir des DF défini plus haut.

1NF :

Nous pouvons remarquer que nos relations sont toutes 1NF car il n’y a aucun attributs multivalués.

2NF :

La majorité de nos relations ont une clé simple. Ceux qui ont une clé composite, tous les attributs ont besoin de la clé complète pour les définir. Ce qui rend nos relations 2NF

3NF:

Il n’y a aucune transitivité dans nos relations. Nous avons pris soin de mettre les clés étrangères pour nos permettre d’accéder à l'information pertinente à nos entités.

BCNF:

Tous les relations n’ont qu’une clé, alors BCNF.