

# MODELISER VOS BASES DE DONNEES RELATIONNELLES

## P4 AJOUTEZ DES ATTRIBUTS AUX RELATIONS PLUSIEURS A PLUSIEURS

### SOMMAIRE

Sommaire .....	1
<b>P4 - Ajoutez des attributs aux relations plusieurs à plusieurs.....</b>	<b>3</b>
<b>C1 - Ajoutez un attribut dans une table de liaisons.....</b>	<b>3</b>
S1: Élargissez les possibilités d'une table de liaisons .....	3
S2: Découvrez la clé primaire dans une table de liaisons avec attributs .....	5
S3 : En résumé .....	6
<b>C2 - Découvrez les modèles adaptés aux tables de liaisons avec attributs .....</b>	<b>7</b>
S1: Découvrez le MLD issu d'une table de liaisons avec des attributs.....	7
S2: Découvrez le MCD à avec une relation porteuse d'attributs .....	11
3: Découvrez l'utilité finale du MCD.....	14
<b>C3 - Appropriiez-vous la méthode complète de modélisation .....</b>	<b>15</b>
S1: Maîtriser les étapes de la modélisation des BD relationnelles .....	15
S2: Découvrez la méthode complète de réalisation d'un MCD .....	16
S3: Complétez les règles de passage du MCD au MLD.....	17
S4: Entraînez-vous à trouver des relations entre les tables.....	18
<b>C4 - Codez le MPD de liaisons avec des attributs .....</b>	<b>19</b>
S1 : Constatez que le MPD graphique est équivalent au MLD .....	19
S2: Découvrez une clé primaire concaténée dans le code SQL.....	21
<b>C5 - Exercez-vous – 1 .....</b>	<b>23</b>
S1: MCD de DoctoNow .....	23
S2: MCD de Paris Librairie .....	23
<b>C5 - Exercez-vous – 2 – Critiques et corrections .....</b>	<b>24</b>
1 Contrat et client.....	24
2 Employé et fonction.....	24
3 Stock.....	25
4 Etudiant et Groupe de TD .....	25
<b>C5 - Exercez-vous – 3 – Conception MCD et MLD .....</b>	<b>26</b>
1 Donateurs .....	26
2 Billets de concert .....	27
3 La maison de disques .....	28
4 Les commandes .....	29
5 La cinémathèque.....	30
6 Les enchères .....	31
7 ASSURAUTO .....	32
<b>C5 - Exercez-vous – 4 – A partir d'un code SQL .....</b>	<b>33</b>
1 La bibliothèque – Reprise MLD-MR.....	33
2 SQL-DDL Chantiers.....	35

<b>C5 - Exercez-vous – 5 – Modélisation spéciale .....</b>	<b>36</b>
1 Les chantiers d'été – Héritage !!! .....	36
<b>C5 - Exercez-vous – 6 – Modélisation spéciale .....</b>	<b>37</b>
1 Les chantiers d'été complet.....	37
2 Les employés et leurs carrières .....	38
3 Gestion de projet .....	39
4 Le cyclisme .....	40
5 Rolland Garros .....	41
6 Le chenil .....	42
<b>C5 - Exercez-vous – 7 – Départ UML .....</b>	<b>43</b>
1. UML-utc .....	43
2. UML-174 .....	44
3. UML-147 .....	45
4. UML-172 .....	46
5. UML-147 .....	47
6. UML-148 .....	48
7. UML-149 .....	49
<b>C5 - Exercez-vous – 8 – départ Merise.....</b>	<b>50</b>
1 MCD-171 .....	50
2 MCD-173 .....	51
3 MCD-175 .....	52

Ce cours a été écrit par Bertrand Liaudet en collaboration avec [Open Class Room](#) en juillet-septembre 2020.  
**Edition Bertrand Liaudet : décembre 2020.**

## P4 - AJOUTEZ DES ATTRIBUTS AUX RELATIONS PLUSIEURS A PLUSIEURS

### C1 - Ajoutez un attribut dans une table de liaisons

#### S1: Élargissez les possibilités d'une table de liaisons

- On reprend notre projet DoctoNow et l'organisation des conférences.
- Les tables de données étaient les suivantes :

Tables des médecins			Table conf-med		Table des conférences		
id	Nom	...	id_med	id_conf	id	Libellé	...
1	Mongéné		1	1	1	conf 1	
2	Maderma		2	1	2	conf 2	
3	Dureigne		7	1	3	conf 3	
4	Maderma		1	2			
5	Hamou		4	2			
6	Mongéné		1	3			
7	Bohbot		2	3			
8	Powel		8	3			
9	El Jabri		10	3			
10	Sipani						
11	Servajean						

- La table « Conf-Med » décrit les médecins qui « font » une conférence.
- On veut enregistrer les informations supplémentaires suivantes :
  - Les médecins qui « font » la conférence peuvent être :
    - ⇒ uniquement organisateur
    - ⇒ ou uniquement conférencier, c'est-à-dire prendre la parole pour présenter son travail
    - ⇒ ou les deux.

- On va faire un exercice avec les données suivantes :
  - Pour la conf 1 : le médecin 1 est organisateur et conférencier. Les autres sont uniquement conférenciers.
  - Pour la conf 2 : le médecin 1 est organisateur et le médecin 4 est conférencier.
  - Pour la conf 3 : le médecin 10 est organisateur, le médecin 1 organisateur et conférencier et les autres sont uniquement conférenciers.
  
- On vous demande de trouver une solution d'organisation pour stocker ces statuts des participants aux conférences dans des tables de type Excel qui restent en 1<sup>ère</sup> forme normale.

## S2: Découvrez la clé primaire dans une table de liaisons avec attributs

### Présentation de la solution :

- La solution ne notre problème consiste en 2 points :
  - ajouter un attribut statut dans la table de liaison
  - si un médecin a deux statuts (organisateur et conférencier) pour une conférence, mettre 2 fois ce médecin dans la table de liaison, avec un statut différent à chaque fois.
- Voici la table de liaison mise à jour et la solution de notre problème :

Tables des médecins			Table conf-med			Table des conférences		
id	Nom	...	id_med	id_conf	statut	id	Libellé	...
1	Mongéné		1	1	organisateur	1	conf 1	
2	Maderma		1	1	conférencier	2	conf 2	
3	Dureigne		2	1	conférencier	3	conf 3	
4	Maderma		7	1	conférencier			
5	Hamou		1	2	organisateur			
6	Mongéné		4	2	conférencier			
7	Bohbot		1	3	organisateur			
8	Powel		1	3	conférencier			
9	El Jabri		2	3	conférencier			
10	Sipani		8	3	conférencier			
11	Servajean		10	3	organisateur			

- On voit que le médecin 1 se retrouve 2 fois sur la conférence 1 dans la table de liaison : 1 fois comme organisateur et une fois comme conférencier.
- Le couple (id\_med, id\_conf) n'est plus la clé primaire puisqu'on peut trouver deux individus avec la même valeur pour ce couple. La clé primaire est désormais constituée des 3 attributs.

### **S3 : En résumé**

- Nous venons de voir comment intégrer un attribut dans une table de liaisons.
- Nous venons de voir comment trouver la clé primaire d'une table de liaison.

## C2 - Découvrez les modèles adaptés aux tables de liaisons avec attributs

### S1: Découvrez le MLD issu d'une table de liaisons avec des attributs

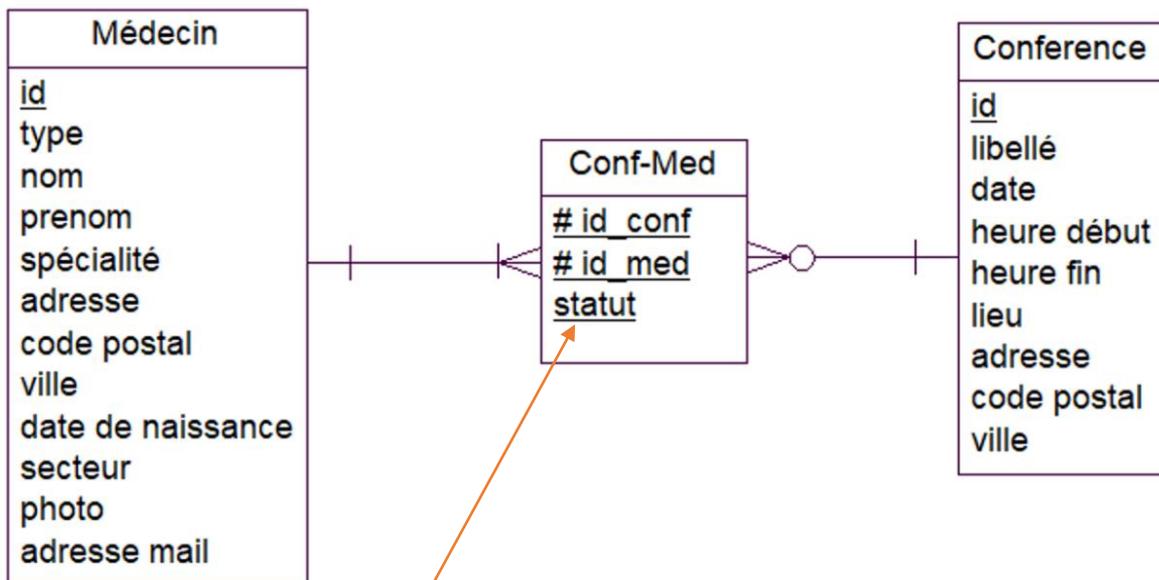
#### MLD-ERD graphique

On reprend nos trois tables de médecins, conférences et de liaisons, cette dernière avec son attribut « statut ».

Voici les tables qu'on commence à bien connaître 😊 :

Tables des médecins			Table conf-med			Table des conférences		
id	Nom	...	id_med	id_conf	statut	id	Libellé	...
1	Mongéné		1	1	organisateur	1	conf 1	
2	Maderma		1	1	conférencier	2	conf 2	
3	Dureigne		2	1	conférencier	3	conf 3	
4	Maderma		7	1	conférencier			
5	Hamou		1	2	organisateur			
6	Mongéné		4	2	conférencier			
7	Bohbot		1	3	organisateur			
8	Powel		1	3	conférencier			
9	El Jabri		2	3	conférencier			
10	Sipani		8	3	conférencier			
11	Servajean		10	3	organisateur			

Voici le MLD :



- On retrouve le même modèle que lorsqu'il n'y avait pas d'attribut dans la table de liaisons, mais avec l'attribut « **statut** » en plus dans la table de liaison.
- De plus, la clé primaire de la table Conf-Med est désormais constituée de 3 attributs !

## Identifiez la clé primaire

Comment trouver la clé primaire de la table de liaisons maintenant qu'on a ajouté cet attribut ?

- Dans ce cas, il n'y a pas de règle pour trouver la clé primaire : il faut réfléchir à une concaténation d'attributs qui puissent faire office d'identifiant.
- id\_med et id\_conf ne forment plus la clé primaire : en effet, on trouve deux individus, les 2 premiers, qui ont les mêmes valeurs pour le couple id\_med et id\_conf, en l'occurrence (1, 1).
- Il faut donc ajouter l'attribut statut pour avoir la clé primaire qui est donc : id\_med, id\_conf, statut.
- Chaque individu de la table est bien un médecin sur un conférence avec un certain statut.

**Question tag :** Ca fait quand même beaucoup d'attributs pour une clé primaire. Est-ce qu'on ne pourrait pas ajouter un attribut « id » qu'on numéroterait comme pour les médecins ou les conférences ?

On peut. Mais ce n'est pas une bonne idée !

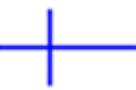
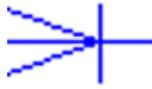
- D'abord parce que cet attribut, on ne s'en servira finalement jamais. C'est une information qui ne sera pas utilisée dans notre application.
- Mais aussi parce que si on fait ça, on n'aura pas identifié que le triplet (id\_conf, id\_med, statut) est la clé primaire et on prendra le risque d'avoir des doublons sur ce triplet et donc des données incohérentes.

## Rappels de syntaxe

On retrouve la syntaxe déjà vue dans les parties 2 et 3

- On retrouve les premières lignes comme dans le modèle présenté dans la première partie.
  - Les attributs qui forment la clé primaire sont placés en premier et soulignés.
  - Les clés étrangères sont placées de préférence en dernier. On peut les faire précéder d'un symbole qui les identifie comme clé étrangère (par exemple un hashtag : #).
  - Dans une clé primaire concaténée, les clés étrangères qui participent à la clé primaire sont placées en premier.
- 
- Pour la relation entre les tables, on utilise le formalisme des Entity relationship Diagrams :



	<p>Cette extrémité veut dire : 1 et 1 seul. Elle traduit les deux justifications :</p> <ul style="list-style-type: none"><li>• Un participant à une conférence (une ligne de conf-med) concerne une conférence et une seule.</li><li>• Un participant à une conférence concerne un médecin et un seul.</li></ul>
 ou 	<p>Ces extrémités veulent dire : plusieurs. Elles traduisent les deux justifications :</p> <ul style="list-style-type: none"><li>• Un médecin peut être plusieurs fois participants.</li><li>• Une conférence a plusieurs participants.</li></ul> <p><i>Remarque : le O veut dire « zéro » : un médecin n'est pas forcément un participant. On peut aussi mettre un I à la place du O : une conférence a forcément au moins un participant : son organisateur. Sachez que cette distinction n'est très importante. Certains modèles ne la présentent pas.</i></p>

## MLD textuel

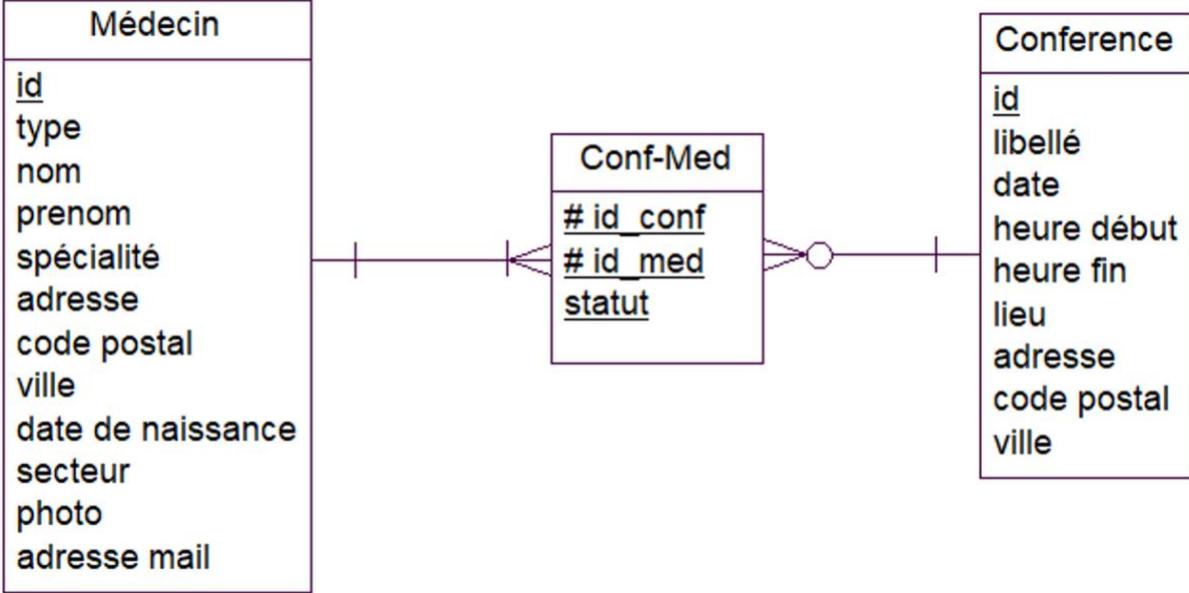
On peut représenter le MLD sans schéma, juste avec du texte :

- Medecin(id, type, nom, ...)
- Conference(id, libelle, ...)
- Conf-Med( #id\_conference, #id\_medecin, statut)

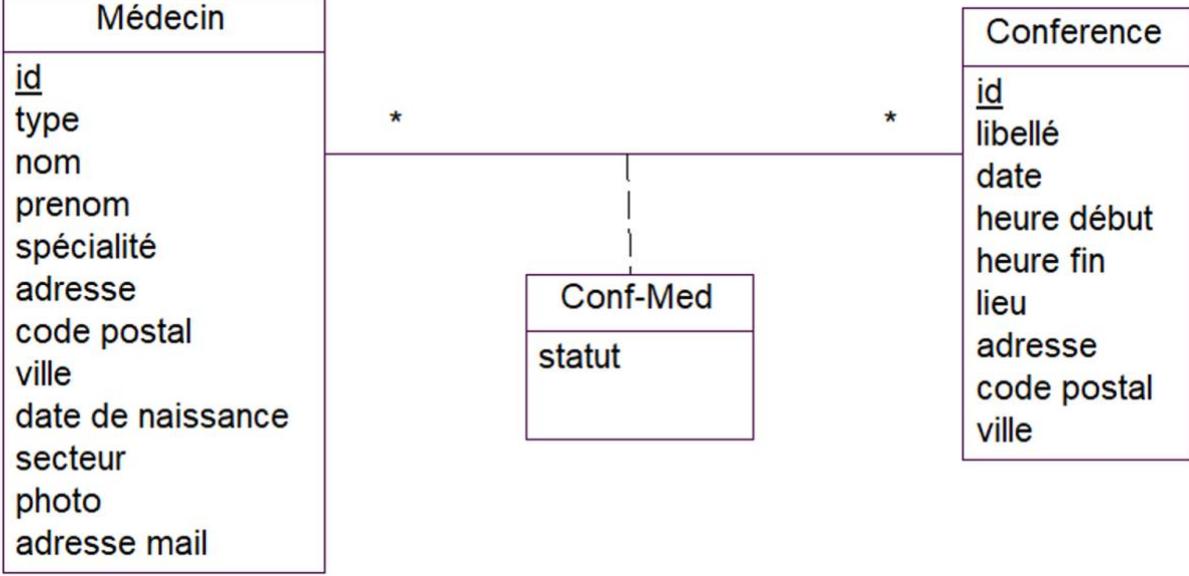
C'est exactement la même chose que le MLD graphique.

**S2: Découvrez le MCD à avec une relation porteuse d'attributs**

On reprend le MLD précédent :



Regardez comment il se traduit en MCD :



## Rappels de syntaxe

On retrouve la syntaxe déjà vue dans les partie 2 et 3

- Les clés primaires sont placées en premier.
- **Les clés étrangères ont disparu** : il ne doit pas y avoir de clés étrangères dans un MCD. C'est la relation et la cardinalité qui remplacent la clé étrangère.
- Une relations 1 à plusieurs se représente ainsi :

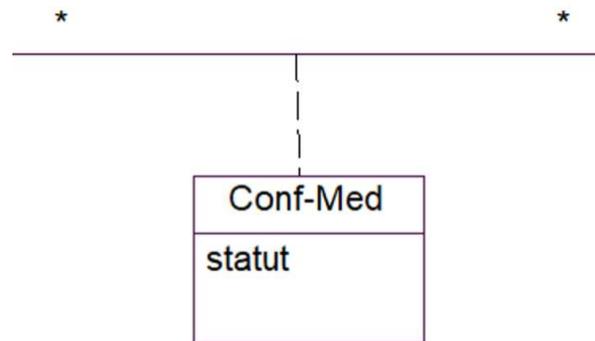


- Une relation plusieurs à plusieurs se représente ainsi :



### Nouveauté de syntaxe : la relation porteuse d'attributs.

- La seule nouveauté syntaxique dans ce MCD se trouve dans la relation : c'est une **relation plusieurs à plusieurs « porteuse d'attributs »**. Pour pouvoir placer ces attributs, on va créer une « classe-association » dans le vocabulaire UML (c'est notre table de liaisons) qu'on va raccrocher à la relation plusieurs à plusieurs par un trait discontinu :



- Le « \* » se retrouvent aux deux extrémités. Il veut dire : plusieurs.
- A l'extrémité du côté des conférences, Il traduit le fait qu'un médecin peut participer à plusieurs conférences.
- A l'extrémité du côté des médecins, Il traduit le fait qu'une conférence peut avoir plusieurs médecins participants.
- La « classe-association » veut dire que la relation porte des attributs. Ici un seul : le statut.
- La « classe-association » a un nom : ce sera celui de la table de liaison qui résultera de la relation plusieurs à plusieurs. On avait déjà choisi « Conf-Med » comme abréviation du nom des 2 tables reliées.
- **Tag information - astuce**
- Les relations porteuses d'attributs sont obligatoirement des relations plusieurs à plusieurs.

### 3: Découvrez l'utilité finale du MCD

#### **Le MCD permet d'organiser les données plus facilement**

Trouver directement des tables de liaisons avec des attributs dans un modèle logique est difficile.

Par contre :

- Il est assez facile d'identifier une relation plusieurs à plusieurs entre 2 tables.
- Une fois cette relation identifiée, il est assez facile aussi de trouver des attributs à lui faire porter, s'il y en a.

Le MCD trouve donc son intérêt majeur avec les relations plusieurs à plusieurs porteuses d'attributs.

#### **Rappel : Le MCD permet de communiquer avec des non informaticiens**

Le MCD offre aussi un autre avantage sur le MLD : il permet de parler de l'organisation des données sans faire intervenir du vocabulaire technique comme « clé étrangère ». Et ça, c'est pratique pour discuter avec le client et être sûr qu'on se comprend bien.

## C3 - Appropriiez-vous la méthode complète de modélisation

### S1: Maîtriser les étapes de la modélisation des BD relationnelles

On a déjà vu les 5 étapes de la méthode pour modéliser des bases de données relationnelles : MCD, MLD, MPD et peuplement.

Voici le détail final de ces étapes :

1. L'étape du MCD est la principale étape d'analyse. C'est le cœur de la modélisation.
2. Une fois le MCD réalisé, on le traduit en MLD en appliquant les règles de passage du MCD au MLD.
3. Avec un MLD, on a l'équivalent de tables de type Excel. On peut valider son modèle en peuplant les tables « sur le papier » ou dans un tableur. Mettre des données concrètes dans nos tables est utile pour découvrir d'éventuelles erreurs. 😊 S'il n'y a pas d'erreurs, le MLD et le MCD sont alors validés.
4. Une fois le MLD réalisé et validé, on code le MPD correspondant. On peut adapter le modèle à des contraintes techniques spécifiques.
5. Une fois le MPD réalisé et mis en œuvre dans un logiciel de gestion de bases de données comme MySQL ou Oracle, on pourra peupler la BD dans le logiciel. Si les étapes 1 à 3 ont bien été réalisées, ça ne doit pas poser de problème ! 😊

Les sections suivantes vont finir de détailler l'étape du MLD et les règles de passage du MCD au MLD.

Le cours suivant finira de détailler l'étape du MPD.

## S2: Découvrez la méthode complète de réalisation d'un MCD

On va présenter maintenant la méthode complète pour réaliser un MCD.

1. Pour faire le MCD, on commence par trouver les tables les plus évidentes. Ce sont celles qui contiennent des informations sur les réalités les plus concrètes. Par exemple : les médecins et les conférences.
2. Dès qu'on a une table, il faut chercher des attributs. Une clé primaire, une clé s'usage, d'autres attributs. Et il faut bien vérifier qu'on a une relation « a 1 et 1 seul » entre la table et chaque attribut.
3. Dès qu'on a 2 tables, il faut se demander si on n'a pas une relation entre ces deux tables. Cette relation se traduit par un verbe qui peut relier les 2 tables comme par exemple : les médecins « participent » à des conférences.
4. Quand on a identifié une relation, on va analyser les cardinalités pour savoir si c'est une relation 1 à plusieurs, et dans ce cas, dans quelle sens elle se place, ou si c'est une relation plusieurs à plusieurs.
5. Quand on a une relation plusieurs à plusieurs, on se demande si elle ne peut pas porter des attributs. Par exemple, les médecins participent à des conférences. C'est une relation plusieurs à plusieurs. Ils peuvent avoir un statut de participation.

La méthode de réalisation d'un MCD se résume essentiellement à ces 5 points.

### S3: Complétez les règles de passage du MCD au MLD

Quand le MCD est fini, pour faire le MLD, on applique 3 règles de transformation qui permettent de transformer facilement le MCD en MLD.

Ces trois règles sont les suivantes :

**Règle 1 :** Les tables du MCD se transforment en tables dans le MLD avec tous leurs attributs, à l'identique. Les clés primaires des tables du MCD deviennent clés primaires des tables du MLD.

**Règle 2 :** Les relations 1 à plusieurs se transforment en un attribut clé étrangère. Cet attribut rentre dans la table du côté plusieurs de la relation. Cette clé étrangère fait référence à la clé primaire de la table du côté 1 de la relation. Cette clé étrangère s'appelle en général : id\_nom-de-la-table-du-côté-1.

**Règle 3 :**

- Les relations plusieurs à plusieurs se transforment en une table de liaisons qui contient 2 attributs clés étrangères. Ces clés étrangères font référence aux clés primaires des 2 tables reliées. Chaque clé étrangère s'appelle en général :  
id\_nom-de-la-table-reliée.
- Si la relation plusieurs à plusieurs est porteuse d'attributs, ceux-ci entrent dans la table de liaisons.
- La clé primaire de la table de liaison est constituée de la concaténation des clés étrangères si la relation n'est pas porteuse d'attributs. Sinon, il faut rechercher « à la main » les attributs qui forment la clé primaire.

## S4: Entraînez-vous à trouver des relations entre les tables

### Exercice 1 :

On envoie des courriers à des clients (le même courrier pour tout un groupe de clients).

Un client à un nom et une adresse.

Un courrier un titre, un texte et une date.

Parfois, le courrier nous revient : l'adresse n'est plus valide.

On veut pouvoir savoir à qui on a écrit. Quelles sont les courriers qui sont revenus parce que l'adresse n'était pas bonne et quand on a récupéré cette information. On veut éviter de réécrire aux clients dont l'adresse n'est plus valide mais conserver leur dernière adresse valide.

Faire MCD, MLD, graphe des tables et simulation.

### Exercice 2 :

Les membres d'une association peuvent participer au conseil d'administration.

Un conseil d'administration fonctionne en général pour une année. Les conseils d'administration sont appelés par le nom de l'année pour lequel il fonctionne (par exemple : CA-2022). Si un deuxième CA doit exister pendant l'année 2022, on l'appellera CA-2022-2. Un CA a une date de début de fonctionnement et une date de fin de fonctionnement.

Un membre a un nom et une adresse.

Quand un membre participe à un CA, il peut avoir une fonction. Cette fonction peut changer en cours d'année. On veut pouvoir connaître les dates de responsabilités pour chaque administrateur. Par exemple : Toto a été secrétaire du début du CA au 1 juin, puis trésorier du 2 juin à la fin du CA.

Faire MCD, MLD, graphe des tables et simulation.

## C4 - Codez le MPD de liaisons avec des attributs

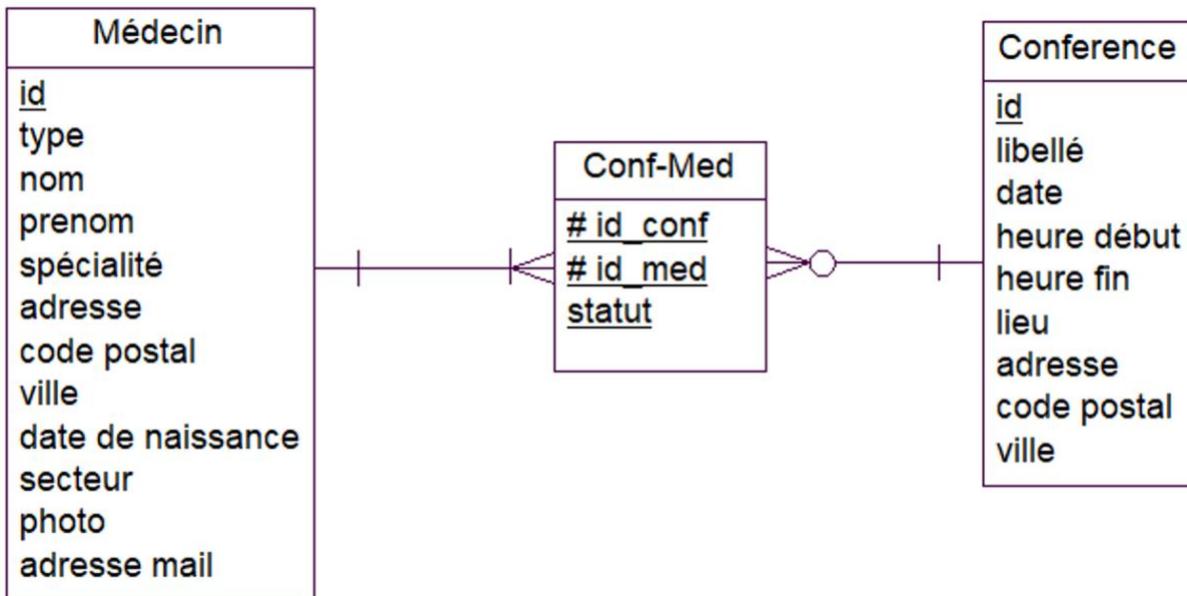
### S1 : Constatez que le MPD graphique est équivalent au MLD

On va regarder maintenant comment produire le MPD à partir d'un MLD avec une relation porteuse d'attributs, c'est-à-dire finalement, comment coder notre modèle.

Le passage au MPD va être identique en terme de méthode à ce qu'on a fait dans le cas d'une relation plusieurs à plusieurs sans attribut.

#### MLD

On reprend le MLD graphique et textuelles de notre gestion de conférences :

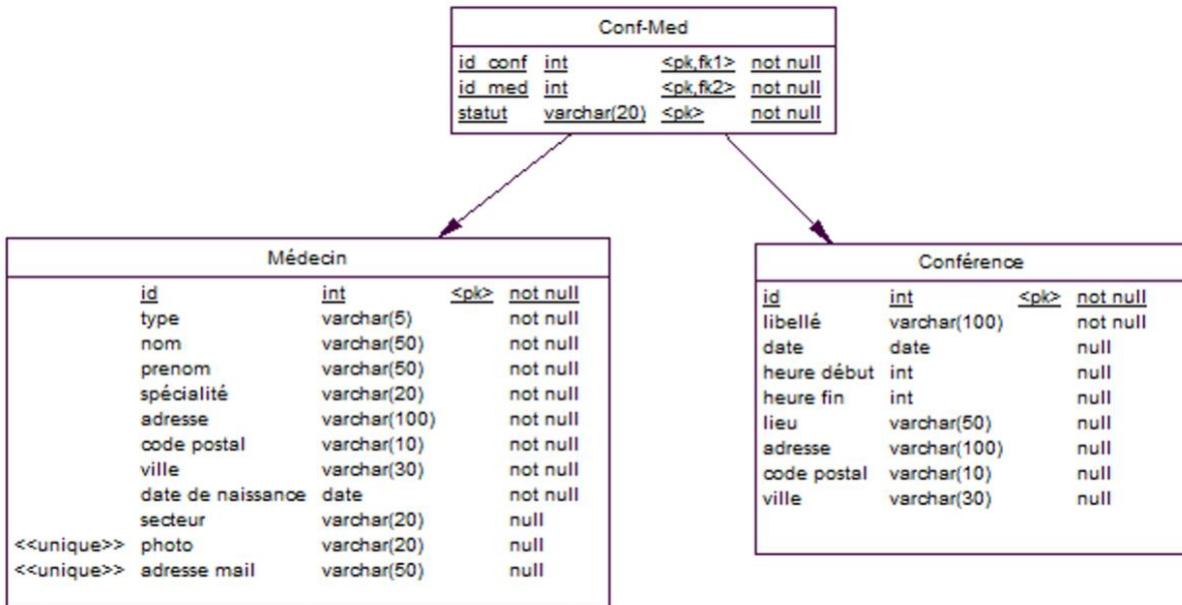


- Medecin(id, type, nom, ...)
- Conference(id, libelle, ...)
- Conf-Med( #id conference, #id medecin, statut)

## MPD graphique

Le MPD graphique est quasiment équivalent au MLD. En général, on y précise les caractéristiques techniques.

Il correspond à ce schéma :



## La syntaxe du MPD graphique est inchangée

Les flèches vont de la clé étrangère vers la clé primaire.

Les clés étrangères sont identifiées avec le symbole <fk>

Les clés primaires sont soulignées.

Les attributs sont typés : int, date, text, varchar(50)

Tous les attributs sont obligatoires : not null.

## S2: Découvrez une clé primaire concaténée dans le code SQL

### MPD-SQL :

On va regarder maintenant le code SQL de notre MPD.

Il n'y a aucune nouveauté syntaxique par rapport au code pour une relation plusieurs à plusieurs sans attribut : simplement, la clé primaire concaténée aura 3 attributs et plus seulement 2.

Voici le code :

```
create table MEDECIN (
  ID          int          not null,
  TYPE       varchar(5)   not null,
  NOM        varchar(50)  not null,
  prenom     varchar(50)  not null,
  specialite varchar(20)  not null,
  adresse    varchar(100) not null,
  code_postal varchar(10) not null,
  ville      varchar(30)  not null,
  date_de_naissance date   not null,
  secteur    varchar(20),
  photo      varchar(20),
  adresse_mail varchar(50),
  primary key (id)
);

create table CONFERENCE (
  id          int          not null,
  libelle     varchar(100) not null,
  date        date,
  heure_debut int,
  heure_fin   int,
  lieu        varchar(50),
  adresse     varchar(100),
  code_postal varchar(10),
  ville       varchar(30),
  primary key (id)
);

create table CONF_MED (
  id_medecin  int          not null,
  id_conference int       not null,
  statut      varchar(20)  not null,
  primary key (id_medecin, id_conference, statut)
);

alter table CONF_MED
  add foreign key (ID_ MEDECIN)
  references MECECIN (id);

alter table CONF_MED
  add foreign key (id_conference)
  references CONFERENCE (id);
```

### **La syntaxe du MPD-SQL est inchangée**

On retrouve la même syntaxe pour la création des tables.

On retrouve la même syntaxe pour la création des clés étrangères : on ajoute un « alter table ». On modifie la table « CONF\_MED ». On ajoute à chaque fois une clé étrangère en précisant l'attribut concerné et à quelle clé primaire il fait référence.

On retrouve la même syntaxe pour la création de la clé primaire concaténée : il suffit de lister les attributs qui forment la clé primaire.

## C5 - Exercez-vous – 1

### S1: MCD de DoctoNow

- Reprenez tout ce qu'on a vu jusqu'à présent dans DoctoNow et faites le MCD correspondant. Puis le MLD, le graphe des tables et le MPD.
- On ajoute que les docteurs s'inscrivent à des conférences.
- On rappelle que les médecins peuvent parler plusieurs langues.
- Réfléchissez à d'autres relations plusieurs à plusieurs.

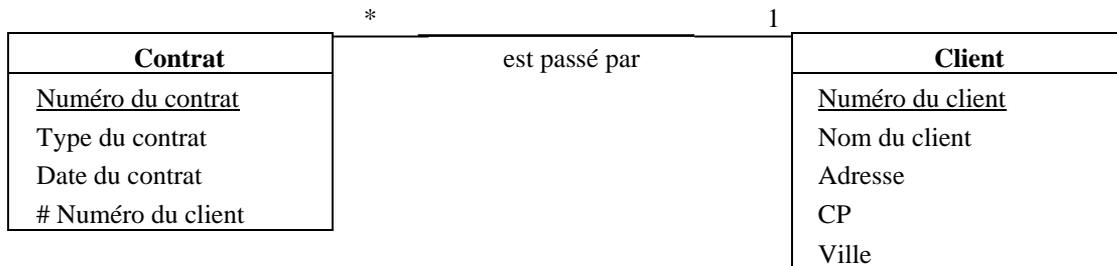
### S2: MCD de Paris Librairie

- Reprenez tout ce qu'on a vu jusqu'à présent dans l'application Paris Librairies et faites le MCD correspondant. Puis le MLD, le graphe des tables et le MPD.
- Les livres peuvent avoir plusieurs sous catégories définies par l'éditeur.
- Réfléchissez à d'autres relations plusieurs à plusieurs.

## C5 - Exercez-vous – 2 – Critiques et corrections

### 1 Contrat et client

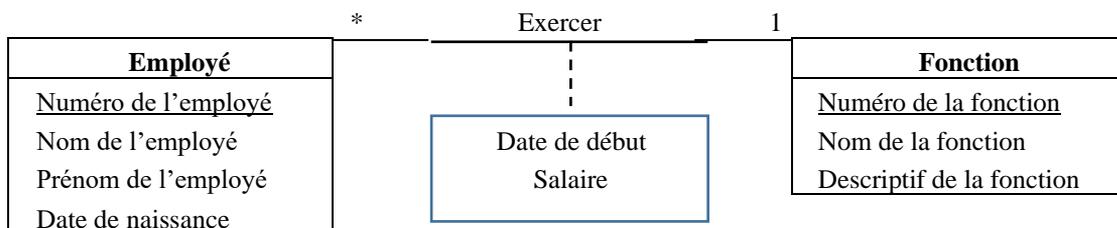
Soit le modèle entité-association suivant :



Un contrat est passé par un client et un seul. Un client peut passer plusieurs contrats.  
Quelle critique formelle pouvez-vous faire au MCD présenté ci-dessous ?  
Proposez un modèle corrigé.

### 2 Employé et fonction

Soit le modèle entité-association suivant :



Un employé peut exercer plusieurs fonctions successivement avec différents salaires, mais pas en même temps.

- Quelle critique formelle (= syntaxique, = non sémantique = est-ce que ça « compile » ?) pouvez-vous faire au MCD présenté ci-dessous ?
- Proposez un modèle corrigé.
- Proposez le MLD correspondant et le graphe des tables.

### 3 Stock

Soit le modèle entité-association suivant :

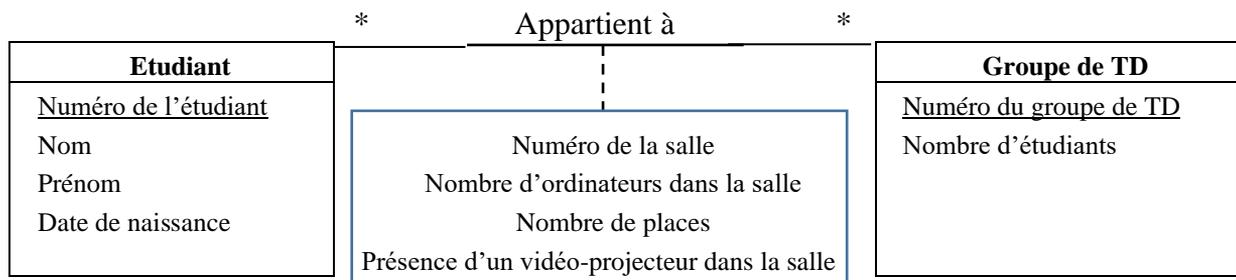
Stock
<u>NuméroProduit, NuméroDépôt</u>
NomProduit
PrixUnitaireProduit
Quantité
AdresseDuDépôt
CodePostalDépôt
VilleDépôt
TéléphoneDépôt

La quantité correspond à la quantité de produit stocké pour un numéro de produit et un numéro de dépôt. Dans un dépôt, il peut y avoir plusieurs produits. Un dépôt peut être vide. Un produit peut être stocké dans plusieurs dépôts. Un produit peut être en rupture de stock.

- Quelle critique formelle (=syntaxique, = non sémantique = est-ce que ça « compile » ?) pouvez-vous faire au MCD présenté ci-dessous ?
- Proposez un modèle corrigé.
- Proposez le MLD correspondant et le graphe des tables.

### 4 Etudiant et Groupe de TD

Soit le modèle entité-association suivant :



Un étudiant appartient à un groupe de TD et un seul. Un groupe de TD est associé à une salle et une seule.

- Quelle critique formelle (=syntaxique, = non sémantique = est-ce que ça « compile » ?) pouvez-vous faire au MCD présenté ci-dessous ?
- Proposez un modèle corrigé.
- Proposez le MLD correspondant et le graphe des tables.

## **1 Donateurs**

Une association reçoit des dons de donateurs privés.

Pour chaque don, l'association enregistre la date du don, son montant ainsi que le nom et l'adresse du donateur, le téléphone et l'adresse mail si possible. Les dons ont un numéro identifiant qui les distingue les uns des autres.

Vers le mois de février, l'association envoie des reçus fiscaux aux donateurs de l'année. Sur le reçu, on affiche l'année fiscale, le nom et l'adresse du donateur, la liste des dons concernés par le reçu avec leur montant et leur date, le total des dons pour ce reçu. Le reçu est caractérisé par un numéro identifiant. On garde aussi la date d'envoi du reçu. Si un donateur se plaint de ne pas avoir récupéré son reçu, on lui renvoie en mettant à jour la date d'envoi.

L'association veut pouvoir afficher en permanence le total des dons de l'année en cours ainsi que le nombre de donateurs et le montant moyen des dons par donateurs (total des dons divisé par le nombre de donateurs). Elle veut aussi pouvoir obtenir ces informations pour toutes les années de fonctionnement de l'association.

## 2 Billets de concert

Sur un billet de concert, on trouve le nom de l'artiste ou du groupe, la date du concert, le nom de la salle de concert, une catégorie de place (un chiffre ou un intitulé). Chaque billet à un code barre qui permet de l'identifier. Une fois scanné à l'entrée du concert, le billet ne peut plus être utilisé.

Les billets sont vendus à l'unité.

Le vendeur de billets enregistre pour chaque billet vendu la date de vente, le prix de vente et la catégorie de la place. Il enregistre aussi des informations sur le client : un nom, un code postal, une ville et une adresse mail si possible.

Une salle de concert est caractérisée par son nom, son adresse (adresse, code postale et ville) et le nombre de places maximum de la salle.

Un concert est caractérisé par le nom de l'artiste ou du groupe qui se produit, la date du concert et la salle du concert.

### **3 La maison de disques**

Une maison de disques gère des disques. Un disque contient des chansons. On connaît le nombre des chansons du disque. On connaît aussi sa durée. Les chansons ont un numéro sur le disque. Les chansons ont un titre et une durée. Pour chaque chanson, on connaît la liste des musiciens qui y participent ainsi que le nom du ou des chanteurs, le nom du ou des auteurs et le nom du ou des compositeurs. Pour chaque personne, on connaît son nom et sa nationalité.

## 4 Les commandes

Les tables suivantes permettent de gérer des commandes.

Commandes(NC, dateCommande, numClient, nomClient, adClient, réductionCom)

ArticlesCommandes(NC, numArticle, nomArticle, descriptionArticle, quantitéCommandée, prixUnitaireArticle, prixTotal, réduction)

Toutefois, cette organisation n'est pas la bonne : en effet, par exemple, si un client passe 2 commandes, on retrouve 2 fois toutes ses informations dans la table commande.

Sachant que :

Une commande est constituée de plusieurs articles. Un article commandé l'est dans une certaine quantité et pour un certain prix unitaire. Chaque article commandé peut bénéficier d'une réduction particulière. Chaque commande peut bénéficier d'une réduction particulière. Pour calculer le prix réel de l'article, on prend en compte une seule réduction : la plus favorable au client.

Vous devez faire un MLD-MR qui évitera toute duplication d'information.

## 5 La cinémathèque

Une cinémathèque veut mettre en place un système d'information concernant des films et leurs diffusions. La cinémathèque gère plusieurs sites dans différentes villes (un site par ville). Pour chaque film on connaît son titre, sa date de création, ses acteurs avec leur nom, leur prénom et leur nationalité, les réalisateurs avec les mêmes informations, la langue du film. On connaît aussi les salles dans lesquelles il a été projeté, avec l'heure et le jour de la projection, la version projetée (version originale ou version française). On connaît le nombre de places des salles, le type d'écran et le type de son, ainsi que le site correspondant, avec son nom, son adresse, son téléphone et le nombre de salles du cinéma. On veut pouvoir faire un classement hebdomadaire, mensuel et annuel du nombre d'entrées par film, par réalisateur et par acteur.

Pour le jeu de données, on se donne au moins 2 films projetés sur 2 journées. Un film avec 1 réalisateur qui n'est pas acteur. Un film avec un réalisateur qui est aussi acteur dans son film.

On souhaite développer un système d'enchères électroniques. Les premières fonctionnalités décrites sont les suivantes :

Tout utilisateur du système (vendeur ou acheteur) doit préalablement s'inscrire en fournissant une adresse email qui lui servira d'identificateur unique pour ses futures interventions, un mot de passe choisi par lui qui lui servira d'authentification pour ses futures interventions et des renseignements d'état-civil : nom, prénom, adresse.

Les utilisateurs peuvent mettre en vente des objets. Pour cela, ils doivent fournir leur identificateur, leur mot de passe, la catégorie de l'objet pour une recherche par thème, un intitulé court de l'objet, une description détaillée de l'objet avec ses défauts et ses qualités, une mise à prix en euros, la date de début de l'enchère, la durée de l'annonce en jours, une ou plusieurs photos, s'il le souhaite. Le système fournit en retour un numéro de lot unique pour l'objet à vendre.

Un utilisateur ayant proposé un objet à la vente peut réviser la mise à prix à la baisse tant qu'il n'y a aucunes enchères sur l'objet.

Les utilisateurs peuvent effectuer des recherches parmi les objets mis en vente selon les critères suivants : catégorie de l'objet, mots de la description de l'objet, fourchette de prix.

**Un utilisateur peut enchérir sur un objet. Pour être valable, l'enchère doit être supérieure de 0,5 euro à la plus haute enchère déjà effectuée ou à la mise à prix si c'est la première enchère ; l'annonce ne doit pas être expirée ; l'utilisateur ne doit pas être le vendeur de l'objet. Pour suivre l'évolution de l'enchère, le système garde aussi la date de chaque enchère.**

Quand l'échéance de fin d'enchère tombe, le système informe le vendeur dans tous les cas, et l'acheteur s'il y a lieu, du résultat de l'enchère.

Si un vendeur veut remettre en vente un objet non vendu, il doit créer un nouvel objet dans le système avec ses caractéristiques propres.

Par ailleurs, les utilisateurs peuvent aussi sélectionner certains objets dont ils veulent suivre l'enchère. Le système enverra un mail d'alerte à l'utilisateur 24 heures avant la fin de l'enchère. L'utilisateur pourra supprimer les éléments de sa sélection.

## 7 ASSURAUTO

On souhaite construire un MR destiné à la gestion de sinistres pour la compagnie d'assurance-automobile. ASSURAUTO à partir de la lettre de demande d'expertise envoyée par un de ses agents d'assurance.

Voici la lettre :

A Paris, le 10 juin 2009

De la part de M. Pierre Durand, Agent d'assurance  
ASSURAUTO  
10, rue Saint-Antoine, 75001 Paris  
tél : 01 42 42 24 24

A M. Henri Dupond  
Expert en assurance

Monsieur,

Je vous saisis par le présent courrier pour vous demander de suivre le dossier du sinistre SIS20020934 constitué à ce jour suite à l'accident survenu le 02 juin 2005 au croisement de la rue de Rivoli et de la rue du Renard à Paris 1<sup>er</sup> arrondissement, à 11 heures du soir, par temps de pluie, d'où l'état glissant de la chaussée à ce moment-là. M. André Danlemur est un des protagonistes de l'accident. Il a souscrit chez nous, depuis le 17 décembre 2002, un contrat enregistré sous le n° CTR2002120 et de type tout risque sur le véhicule impliqué dans le sinistre et immatriculé 123 ABC 75. Ce véhicule est une Renault Espace de 100 chevaux. M. Oliveau habite à Paris, 25 rue de Rivoli, dans le 1<sup>er</sup> arrondissement. Il est joignable au 01 42 23 12 20 chez lui, ou au 06 12 12 12 12 sur son portable, ou au 01 23 43 23 34 à son travail. Vous voudrez bien prendre contact avec lui pour convenir d'un rendez-vous en vue de l'expertise que vous voudrez bien effectuer au garage « Saroul », 45, rue Rousseau, 75002 Paris, dans un délai de 15 jours.

Dans l'attente de votre rapport d'expertise, nous vous prions de croire, Monsieur l'expert, à nos sentiments distingués.

M. Durant, Agent d'assurance

**On ajoute les spécifications suivantes :** un accident est expertisé une ou plusieurs fois. En cas de ré-expertise, on change forcément d'expert. Un accident est suivi par l'agent responsable du contrat. Pour un accident, on enregistre aussi, si nécessaire, la liste de tous les véhicules impliqués un accident concerne donc des véhicules assurés ou pas par ASSURAUTO. Le contrat d'assurance est passé par un client et un seul et concerne une voiture et une seule.

## C5 - Exercez-vous – 4 – A partir d'un code SQL

### 1 La bibliothèque – Reprise MLD-MR

Le code de création de la BD est le suivant :

```
drop database if exists biblio;
```

```
create database biblio;
```

```
use biblio;
```

```
CREATE TABLE oeuvres(  
  NO      integer primary key auto_increment,  
  titre  varchar(150),  
  auteur varchar(100)  
) ENGINE InnoDB;
```

```
CREATE TABLE adherents (  
  NA      int PRIMARY KEY AUTO_INCREMENT,  
  nom     varchar(30) not null,  
  prenom  varchar(30),  
  adr     varchar(100) not null,  
  tel     char(10)  
) ENGINE InnoDB;
```

```
CREATE TABLE livres (  
  NL      integer primary key auto_increment,  
  editeur varchar(50),  
  NO      integer not null,  
  foreign key(NO) references oeuvres(NO)  
) ENGINE InnoDB;
```

```
CREATE TABLE categories(  
  NC      integer primary key auto_increment,  
  categorie varchar(150)  
) ENGINE InnoDB;
```

```
CREATE TABLE thematique (  
  NO      integer not null,  
  NC      integer not null,  
  foreign key(NO) references oeuvres(NO),  
  foreign key(NC) references categories(NC),  
  primary key (NO, NC)  
) ENGINE InnoDB;
```

```
CREATE TABLE emprunter (  
  NL      integer not null,  
  datEmp  date not null,  
  dureeMax integer not null,  
  dateRet date,  
  NA      integer not null,
```

```
foreign key(NA) references adherents(NA),  
foreign key(NL) references livres(NL),  
primary key (NL, datEmp)  
) ENGINE InnoDB;
```

### **Ce code répond à la situation suivante :**

Une bibliothèque gère les emprunts des livres de ses adhérents. Les livres ont un titre, un auteur, un éditeur, une année d'édition. On peut avoir plusieurs exemplaire du même livre mais aussi plusieurs éditions différentes de la même œuvre. Les adhérents ont un nom, une adresse, une adresse mail et éventuellement un téléphone. On souhaite archiver tous les emprunts. Un livre ne peut pas être rendu le jour même de son emprunt. La durée maximum d'emprunt est de 21 jours. Un adhérent ne peut pas avoir plus de 15 emprunts en cours. Tout retard dans les rendus bloque la possibilité de nouveaux emprunts.

Les livres sont rattachés à des genres (jeunesse, SF, fantastique, roman, bande dessinée, philosophie, politique, histoire, etc.). Une œuvre peut être rattachée à plusieurs genres.

A tout moment, on connaît le nombre de livres actuellement empruntés par chaque adhérent ainsi que le nombre de livres actuellement en retard de rendu.

La bibliothèque souhaite pouvoir faire des statistiques sur la pratique des abonnés (nombre de livres empruntés par an, répartition des emprunts par genre, nombre d'emprunts par livre, etc.

De plus, les abonnés peuvent commander des livres. Ils peuvent en commander 5 au maximum. Une commande peut être annulée ou honorée si le livre commandé a finalement été emprunté. La bibliothèque souhaite garder la trace de toutes les commandes effectuées, qu'elles aient été annulées ou honorée.

### **Exercice**

Faire le graphe des tables

Faire le MCD à partir du MLD.

## 2 SQL-DDL Chantiers

Le DDL ci-dessous est associé aux règles de gestion suivantes :

- La BD gère des visites de chantier
- Une visite est faite dans une voiture. Un employé est conducteur.
- D'autres employés sont visiteurs.
- Une visite peut faire intervenir plusieurs voitures.
- Il ne peut y avoir qu'une visite par jour pour un chantier.
- Un véhicule a une capacité, qui exclut le conducteur

```
CREATE TABLE employe (  
    n_emp          VARCHAR(4) PRIMARY KEY,  
    nom_emp        VARCHAR(20) not null,  
    qualif_emp     VARCHAR(12) not null  
) type innodb;  
CREATE TABLE chantier (  
    n_chantier     VARCHAR(10) PRIMARY KEY,  
    nom_ch         VARCHAR(10) not null,  
    adresse_ch     VARCHAR(15) not null,  
    kilometres     INTEGER not null  
) type innodb;  
CREATE TABLE vehicule (  
    n_vehicule     VARCHAR(10) PRIMARY KEY,  
    type_vehicule  VARCHAR(1) not null,  
    capacite       INTEGER not null check(capacite >0)  
) type innodb;  
CREATE TABLE visite(  
    n_chantier     VARCHAR(10),  
    n_vehicule     VARCHAR(10),  
    date_heure     DATETIME not null,  
    n_conducteur   VARCHAR(4) not null,  
    FOREIGN KEY(n_chantier) REFERENCES chantier(n_chantier),  
    FOREIGN KEY(n_vehicule) REFERENCES vehicule(n_vehicule),  
    FOREIGN KEY(n_conducteur) REFERENCES employe(n_emp),  
    PRIMARY KEY(n_chantier,n_vehicule, date_heure)  
) type innodb;  
CREATE TABLE transporter (  
    n_chantier     VARCHAR(10),  
    n_vehicule     VARCHAR(10),  
    date_heure     DATETIME,  
    n_transporte   VARCHAR(4) not null,  
    PRIMARY KEY(n_chantier,n_vehicule,date_heure,n_transporte),  
    FOREIGN KEY(n_chantier,n_vehicule,date_heure)  
        REFERENCES visite(n_chantier,n_vehicule,date_heure),  
    FOREIGN KEY(n_transporte) REFERENCES employe(n_emp)  
) type innodb;
```

### Exercice

Faire le graphe des tables

Faire le MCD à partir du MLD.

**1 Les chantiers d'été – Héritage !!!**

**Modélisez cette première partie : MCD, MLD, GT, Simulation, Valorisation**

Une association organise des chantiers d'été. Il y en a de 3 sortes : chantiers de réhabilitation, chantiers de construction ou chantiers archéologiques. Pour les chantiers de construction et de réhabilitation, on connaît le type de bâtiment. Pour les chantiers de réhabilitation, on connaît la date de création de l'ancien bâtiment et des informations sur l'ancien bâtiment. Pour les chantiers archéologiques, on connaît la période archéologique et la surface du chantier (qui peut être très grande). Les chantiers ont une date de début et une date de fin (qui peut être plusieurs années après la date de fin).

**Modélisez cette deuxième partie : MCD, MLD, GT, Simulation, Valorisation**

Des équipes viennent sur le chantier. Une équipe est constituée pour une durée limitée (environ un mois pendant l'été). Elle est affectée à un chantier et un seul.

Ceux qui participent à un chantier sont membres de l'association. On connaît leur nom, prénom, adresse, email et téléphone.

L'association envoie des courriers à des personnes. Ce sont des envois collectifs fait à une certaine date. Les personnes ont un nom, un prénom et une adresse. Parfois un email et un téléphone. Un courrier a un titre et un libellé.

**Modélisez cette trois ! partie : MCD, MLD, GT, Simulation, Valorisation**

L'association peut recevoir des dons de personnes. Les dons ont un montant et une date de don. Ils sont faits par une personne dont on connaît au moins le nom, le prénom, l'adresse et l'email. Chaque don donne lieu à un reçu fiscal envoyé à une certaine date. La date d'envoi du reçu est conservée.

Les membres payent des cotisations annuelles. Une cotisation à un montant et un don. Pour chaque membre, on enregistre sa date d'entrée dans l'association ainsi que la date de sa dernière cotisation. Les cotisations ne donnent pas lieu à un reçu fiscal.

### 1 Les chantiers d'été complet

Pour cette exercice, on peut partir directement du sujet ci-dessous, ou commencer par l'exercice précédent qui décompose la modélisation.

Une association organise des chantiers d'été. Il y en a de 3 sortes : chantiers de réhabilitation, chantiers de construction ou chantiers archéologiques. Pour les chantiers de construction et de réhabilitation, on connaît le type de bâtiment. Pour les chantiers de réhabilitation, on connaît la date de création de l'ancien bâtiment et des informations sur l'ancien bâtiment. Pour les chantiers archéologiques, on connaît la période archéologique et la surface du chantier (qui peut être très grande). Les chantiers ont une date de début et une date de fin (qui peut être plusieurs années après la date de fin). Des équipes viennent sur le chantier. Une équipe est constituée pour une durée limitée (environ un mois pendant l'été). Elle est affectée à un chantier et un seul. Les équipes sont constituées de participants. Les participants aux chantiers sont des personnes. Leur participation à l'équipe peut être inférieure à la durée de vie de l'équipe elle-même. Les personnes ont un nom, un prénom, une adresse, un email et un téléphone.

L'association peut recevoir des dons de personnes. Les dons ont un montant et une date de don. Ils sont faits par une personne. Ils donnent lieu à un reçu fiscal envoyé à une certaine date.

L'association envoie des courriers aux personnes. Les courriers ont un titre, une date et un libellé.

L'association gère aussi des membres. Les membres payent des cotisations annuelles. Les cotisations ne donnent pas lieu à un reçu fiscal. On distingue donc deux types de dons : ceux avec reçu et les cotisations.

Les membres de l'association peuvent participer au conseil d'administration (CA). Le CA est renouvelé en général tous les ans. Les participants au CA ont une fonction particulière. Cette fonction peut changer en cours de CA. On conserve les dates de début et de fin de chaque fonction occupée. Chaque CA a un certain nombre de participants.

#### **Exercice**

Faire le MCD.

En déduire le MLD en appliquant les règle de passage du MCD au MLD.

Faire le GT.

Faire le MLD valorisé (au moins CP, unique, obligatoire, modifiable, énuméré, automatique, référence).

Vérification : mettre des données dans les tables.

## 2 Les employés et leurs carrières

On souhaite gérer le personnel d'une société. Chaque membre du personnel a un nom, une fonction, un salaire, une commission (part de salaire variable), un salaire totale (salaire + commission), une date d'entrée dans la société. Chaque membre du personnel travaille dans un département caractérisé par son nom (commercial, production, personnel, comptable et recherche) et la ville dans laquelle il se trouve.

Chaque membre du personnel a un supérieur hiérarchique et un seul lui-même membre du personnel, sauf le président qui n'a pas de supérieur hiérarchique.

A cette situation, qui donne lieu au schéma de la BD abordé en cours, on ajoute les éléments suivants :

Le service des Ressources Humaines veut gérer la carrière des employés. Pour chaque employé, on considère chaque modification de salaire, chaque changement de poste, chaque changement de supérieur hiérarchique et chaque changement de département comme une étape dans sa carrière. L'historique des commissions n'est, par contre, pas gardé.

### **Exercice**

Faire le MCD.

En déduire le MLD en appliquant les règles de passage du MCD au MLD.

Faire le GT.

Faire le MLD valorisé (au moins CP, unique, obligatoire, modifiable, énuméré, automatique, référence).

Vérification : mettre des données dans les tables.

### 3 Gestion de projet

On modélise l'organisation des projets en tâches dans une entreprise. Un projet est caractérisé par un nom, une date début, une date de fin et un identifiant unique. Un projet est constitué de plusieurs tâches numérotées de 1 à N. Chaque tâche a un intitulé, une date de début et une date de fin. Elle est précédée par 0 ou plusieurs tâches. Elle est suivie par 0 ou plusieurs tâches. Deux tâches d'un même projet peuvent démarrer à la même date. Deux tâches d'un même projet peuvent se terminer à la même date. Une tâche démarre au plus tôt le lendemain de la fin de la tâche qu'elle suit. Une tâche ne peut pas commencer avant le début du projet, ni finir après la fin du projet.

Dans l'entreprise, les membres du personnel ont un nom, une fonction et un identifiant. A chaque projet, on affecte des membres du personnel avec une fonction spécifique pour chaque projet.

Ensuite, on affecte les membres du projet à des tâches du projet.

Chaque tâche peut requérir des ressources spécifiques. L'allocation d'une ressource par une tâche se fait à partir d'un certain jour et pour une certaine durée. Une ressource peut être allouée plusieurs fois par une même tâche à des périodes différentes. Une ressource a un intitulé et coût journalier.

#### **Exercice**

Faire le MCD.

En déduire le MLD en appliquant les règles de passage du MCD au MLD.

Faire le GT.

Faire le MLD valorisé (au moins CP, unique, obligatoire, modifiable, énuméré, automatique, référence).

Vérification : mettre des données dans les tables.

## 4 Le cyclisme

*Cet exercice suit un vieux principe de logique appelé : « Le rasoir d'Occam » (1287-1349) et qui dit : Entia non sunt multiplicanda praeter necessitatem, c'est-à-dire : il ne faut pas multiplier les entités plus que nécessaire.*

Une entreprise de fabrication de produits nécessaires pour le cyclisme veut automatiser la gestion de son stock et de ses coûts de fabrication. Elle a besoin de connaître à tout moment la quantité en stock de chacun de ses produits, la composition détaillée de chacun des produits composés, etc.

La liste de ses produits se présente comme suit :

1. Un vélo est composé de deux roues, d'un guidon, de deux gardes boue, d'une chaîne, d'une selle, etc.
2. Un tricycle est composé de trois roues, d'un guidon, de trois gardes boue, d'une chaîne, d'une selle, etc.
3. Une roue est composée de cinquante rayons, d'une jante, d'une chambre à air, etc.
4. Etc.

### Exercice

Faire le MCD.

En déduire le MLD en appliquant les règles de passage du MCD au MLD.

Faire le GT.

Faire le MLD valorisé (au moins CP, unique, obligatoire, modifiable, énuméré, automatique, référence).

Vérification : mettre des données dans les tables.

## 5 Rolland Garros

On veut enregistrer et archiver les données d'une compétition de tennis. Les matchs ont lieu à une certaine date, sur un certain court et correspondent à un certain niveau dans la compétition (finale, demi, quart, etc.).

On veut pouvoir connaître les résultats de matchs et pouvoir compter le nombre de jeux et de sets pour chaque match et pour chaque joueur.

### Exercice

Faire le MCD.

En déduire le MLD en appliquant les règles de passage du MCD au MLD.

Faire le GT.

Faire le MLD valorisé (au moins CP, unique, obligatoire, modifiable, énuméré, automatique, référence).

Vérification : mettre des données dans les tables.

## 6 Le chenil

Un réseau de chenils souhaite informatiser son activité et vous demande de créer un modèle conceptuel de sa future base de données. Un chenil est un établissement destiné à l'élevage ou à la pension des chiens. Dans ce réseau, les chenils s'occupent des chiens de leur naissance jusqu'à la fin de leur apprentissage. L'identité du propriétaire étant privée, aucun chien ne sera relié à un propriétaire dans cette base de données qui sert aussi de vitrine sur le savoir-faire du réseau.

Un chenil du réseau possède un nom unique permettant de l'identifier, un nom de contact ainsi qu'un numéro de téléphone, et une description de ses activités. Chaque chien est identifié par son nom et celui du chenil qui gère son apprentissage ; il possède une date de naissance et est associé à une race. Certains des chiens suivront un apprentissage plus spécifique : ils deviendront guides d'aveugle, chiens de garde ou chiens de course. Chaque chien de garde possède une spécialité : attaque, défense, pistage ou détection. Un chien de course quand à lui possède une vitesse maximum mesurée.

Le chenil gère également l'historique des poids de chaque animal, afin d'enregistrer régulièrement la courbe de croissance. Pour cela chaque relevé de poids est identifié par le nom du chien et la date du relevé.

Les chiens aux parcours plus spécifiques (guides, chiens de garde ou chiens de course) sont liés à des entités qui les géreront après l'apprentissage. Ces entités sont identifiées par un nom, et possèdent une description ainsi qu'un nom de contact avec un numéro de téléphone. Les chiens guides d'aveugle sont ainsi gérés par une association qui propose ensuite les chiens aux personnes malvoyantes. Les chiens de gardes sont utilisés par des entreprises (le chenil souhaite pouvoir ajouter une description de l'activité principale de chaque entreprise). Et un chien de course est possédé par une écurie (qui a une date de création). Bien sûr, chacune de ces entités peut gérer plusieurs animaux.

2) On suit les mêmes étapes que pour la série 1 pour le sujet suivant. On distinguera bien les nouveautés par rapport à l'exercice précédent.

Le réseau souhaiterait compléter cette première modélisation en ajoutant des informations sur les chiens de course, pour améliorer sa publicité. L'objectif est de garder en mémoire pour chaque épreuve sportive auquel participe un chien de course son numéro de dossard (de 1 à 6), sa position à l'issue de l'épreuve (de 1 à 6), son temps de course, et s'il a abandonné ou non.

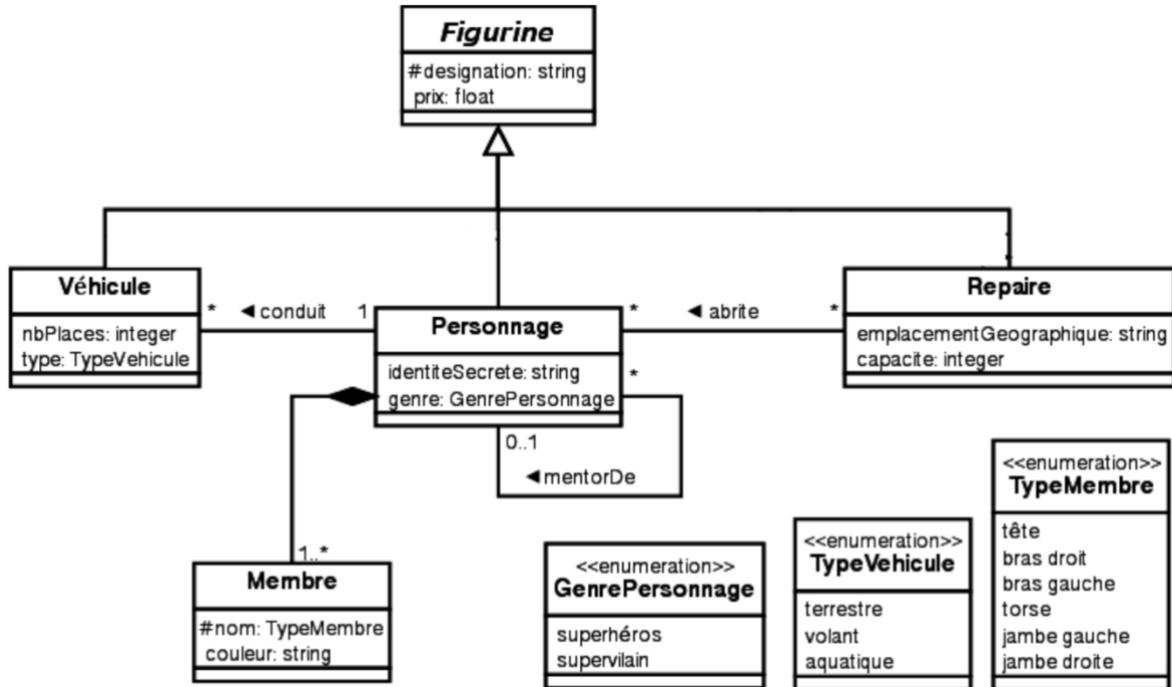
Une épreuve est identifiée par la date et le nom du tournoi à laquelle elle appartient (un tournoi est identifié uniquement par son nom). Il y a toujours 6 chiens qui courent dans une épreuve, mais ils ne proviennent pas tous d'un des chenils du réseau (nous ne gérons pas les

chiens en dehors du réseau). Chaque épreuve a lieu dans un cynodrome identifié par son nom, et chaque cynodrome se situe dans un département, identifié par son numéro et possédant un nom.

## C5 - Exercez-vous – 7 – Départ UML

Pour cette série, on suit les mêmes étapes que pour la série 1. L'objectif n°1 est de produire un MLD textuel, puis un MCD. Les modèles proviennent pour la plupart de « UML 2 pour les bases de données – Christian Sautou ».

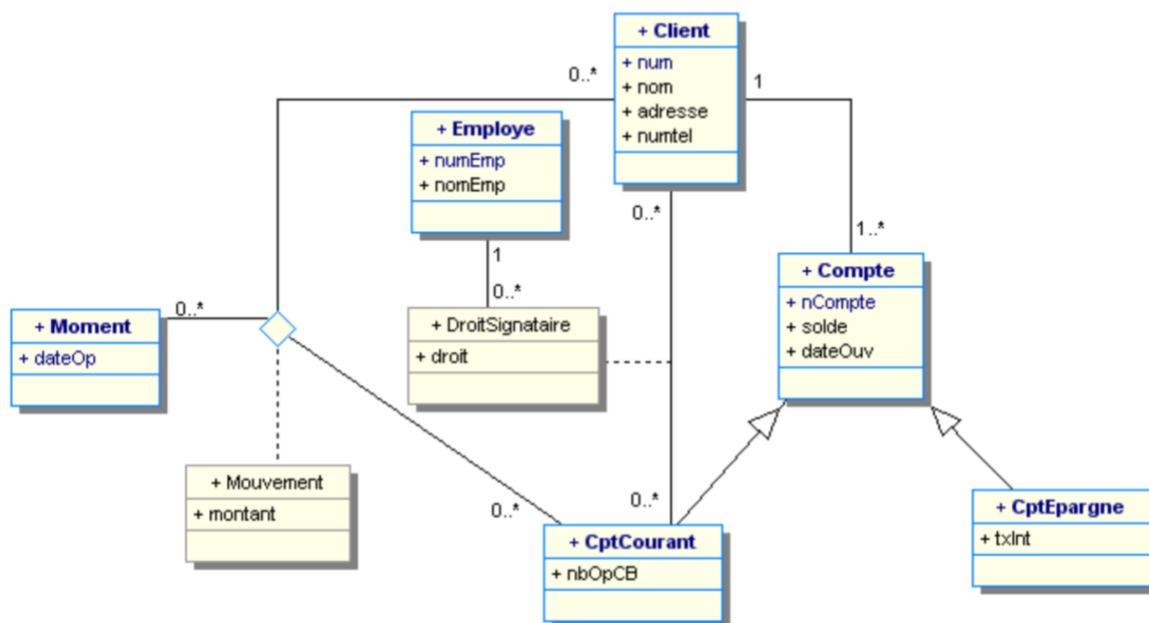
### 1. UML-utc



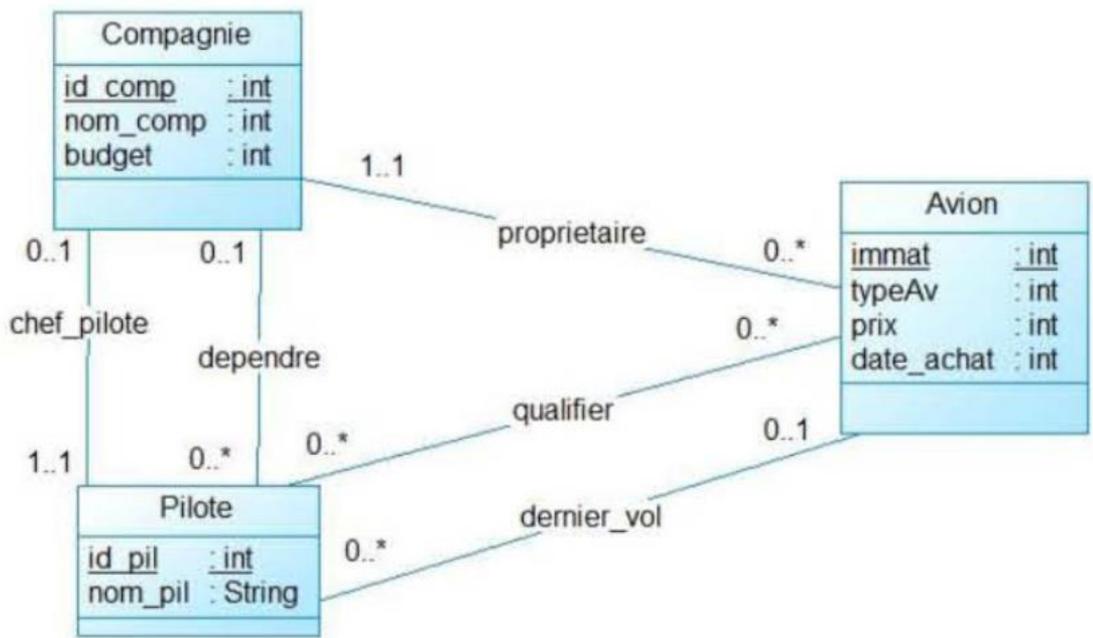
## 2. UML-174

Décrire un schéma relationnel à partir de la modélisation UML suivante. On suppose qu'un compte courant ne peut pas être rémunéré.

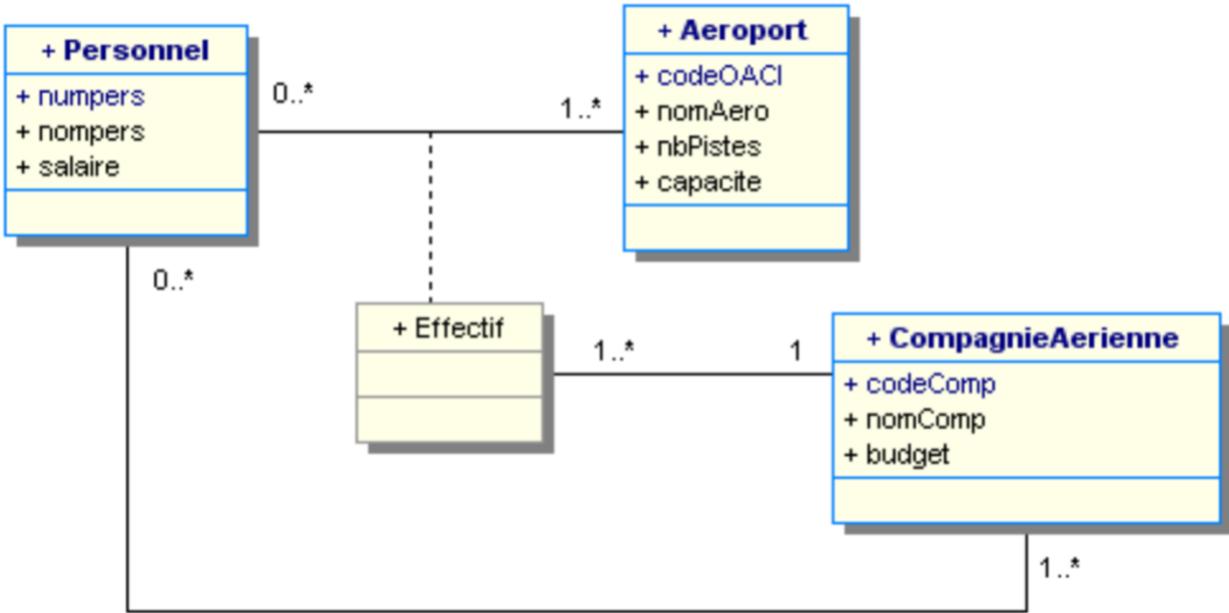
Figure 2-90 Association n-aire UML



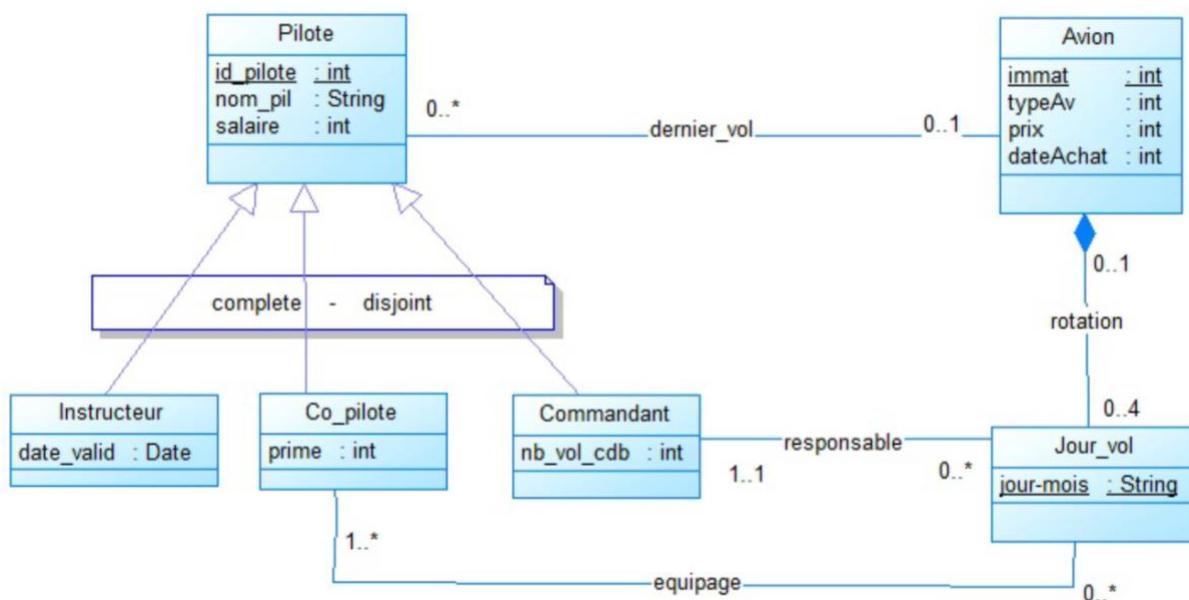
3. UML-147



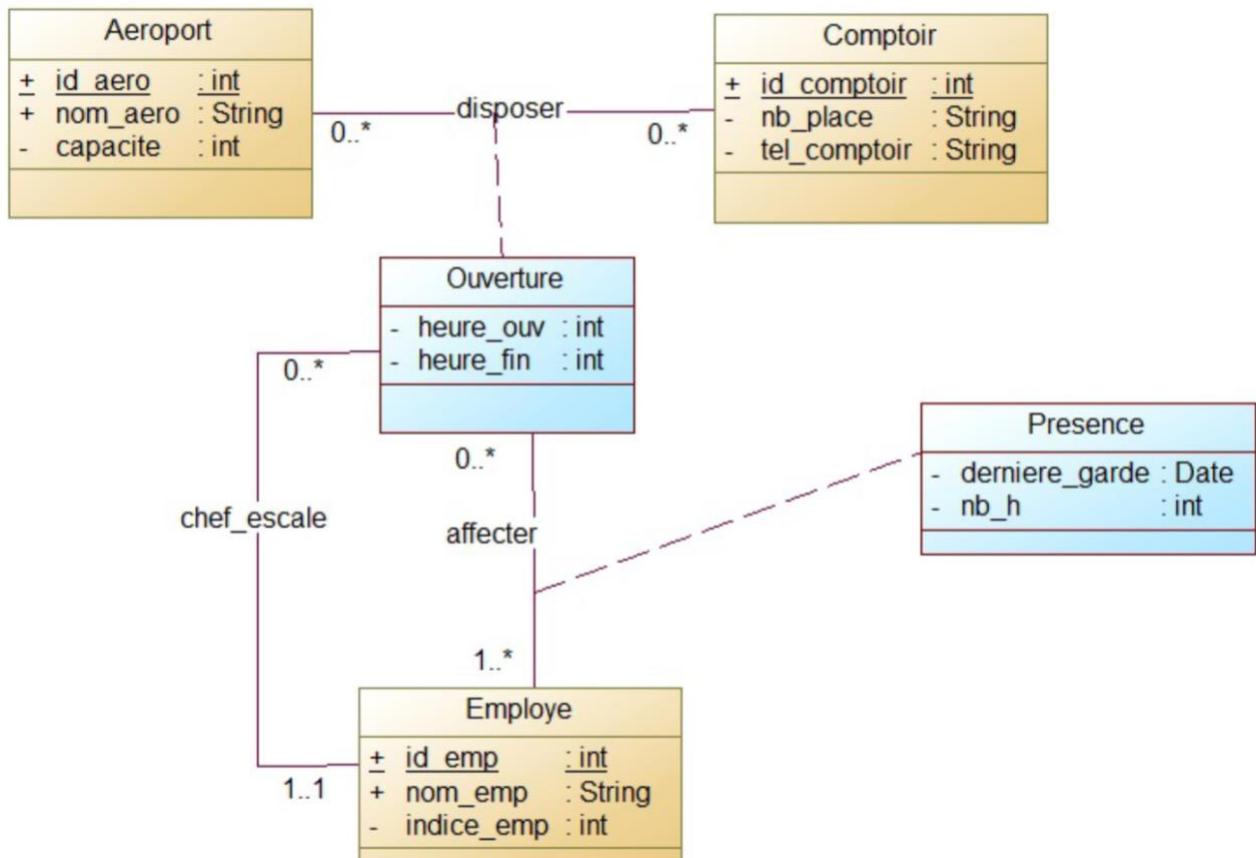
4. UML-172



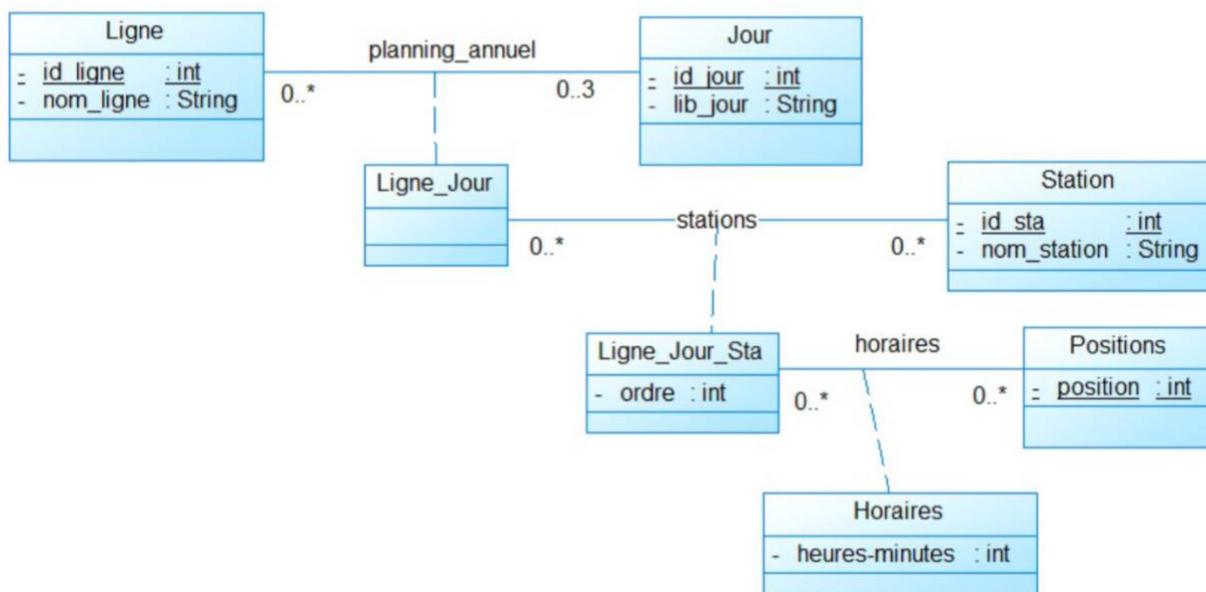
5. UML-147



## 6. UML-148



## 7. UML-149



## C5 - Exercez-vous – 8 – départ Merise

Pour cette série, on suit les mêmes étapes que pour la série 1.

Les modèles proviennent pour la plupart de « UML 2 pour les bases de données – Christian Sautou ».

### 1 MCD-171

