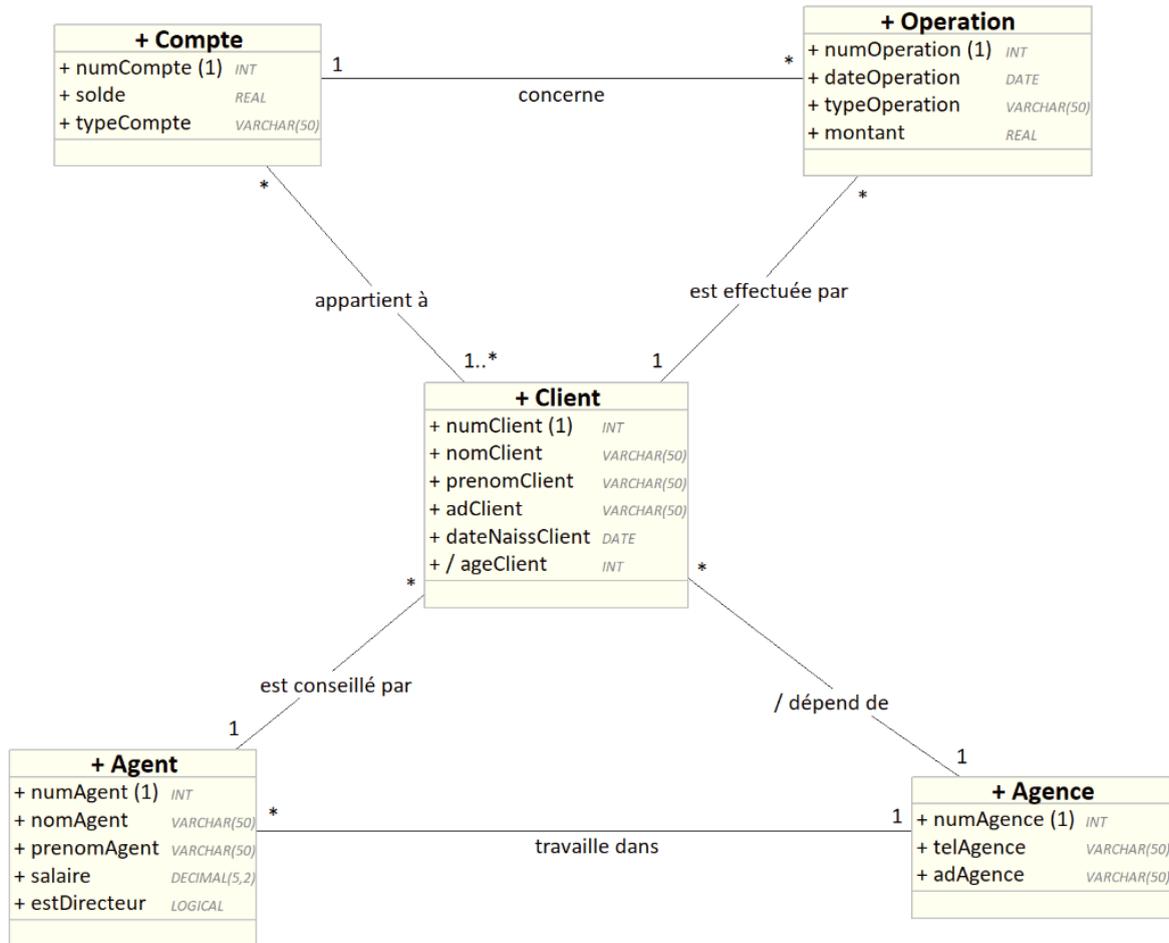


Considérons le diagramme de classes UML suivant complété par des contraintes textuelles :



#### Contraintes textuelles :

- Le montant d'une opération est toujours positif
- Le type d'opération est soit RETRAIT, soit DEPOT
- Le type de compte considéré est soit COURANT, soit EPARGNE
- Par défaut, la date d'opération est la date du jour courant
- Aucun salaire ne doit être inférieur au Smic (1554 euros brut)
- Une augmentation de salaire ne doit pas dépasser 10% et une baisse 8%
- Un client ne peut pas être conseillé par un agent portant le même nom que lui
- Un client ne doit pas pouvoir effectuer un retrait dont le montant est supérieur au solde
- Un client ne doit pas pouvoir retirer de l'argent que sur un compte qui lui appartient
- Une agence a forcément un et un seul directeur
- Le directeur d'une agence est mieux payé que les agents de son agence

## 1. Tables et vues

1. Transformer le diagramme de classes UML en schéma relationnel
2. Écrire le script *tables.sql* permettant de créer les tables en gérant le maximum de contraintes
3. Écrire le script *vues.sql* permettant de créer :
  - la vue fournissant tous les éléments dérivables (attribut et association)
  - la vue détectant les éventuels défauts de surjectivité

## 2. Déclencheurs de lignes

4. Écrire le script *triggers\_ligne.sql* permettant de créer les quatre triggers de ligne
5. Utiliser le script *tests\_triggers\_ligne.sql*, disponible sur Moodle, pour les tester

## 3. Déclencheurs d'état (sans curseur)

6. Écrire le script *triggers\_etat.sql* permettant de créer les deux triggers d'état qui doivent afficher le nombre d'erreurs<sup>[1]</sup>
7. Utiliser le script *tests\_triggers\_etat.sql*, disponible sur Moodle, pour les tester

## 4. Requêtes

8. Écrire le script *requetes.sql* permettant de répondre aux questions suivantes :
  - (a) Quel est le directeur le mieux payé?
  - (b) Pour chaque directeur, indiquer (ordre décroissant) le nombre d'agents.
  - (c) Quels sont les agents qui gèrent le plus grand nombre de comptes épargnes?
  - (d) Lister les 5 derniers retraits du plus récent au plus ancien.
  - (e) Pour chaque agence, indiquer le nombre de clients ne possédant pas encore de compte épargne.
9. Utiliser le script *remplissage.sql*, disponible sur Moodle, pour remplir vos tables et tester vos requêtes

## 5. Bonus

10. Ajouter une contrainte nécessitant un trigger de ligne (respectivement d'état)
11. Écrire le script *triggers\_bonus.sql* permettant de créer les deux triggers et de les tester



| Une archive *Nom.zip* contenant tous les fichiers est à déposer sur Moodle

---

[1]. L'instruction *INSERT ALL* peut provoquer plusieurs erreurs!