

## Thème : Gestion du trafic aérien

### 1. Objectifs du Projet

- Trouver les routes aériennes entre deux aéroports donnés et identifier les plans de vols réalisés par une compagnie aérienne.
- Identifier toutes les compagnies aériennes qui desservent un pays donné.
- Rechercher un aéroport dans un pays donné (plusieurs critères de recherche seront prévus)
- Gestion de l'authentification et administration du trafic (Create Update Delete) avec Django admin
- Point de statistique générale

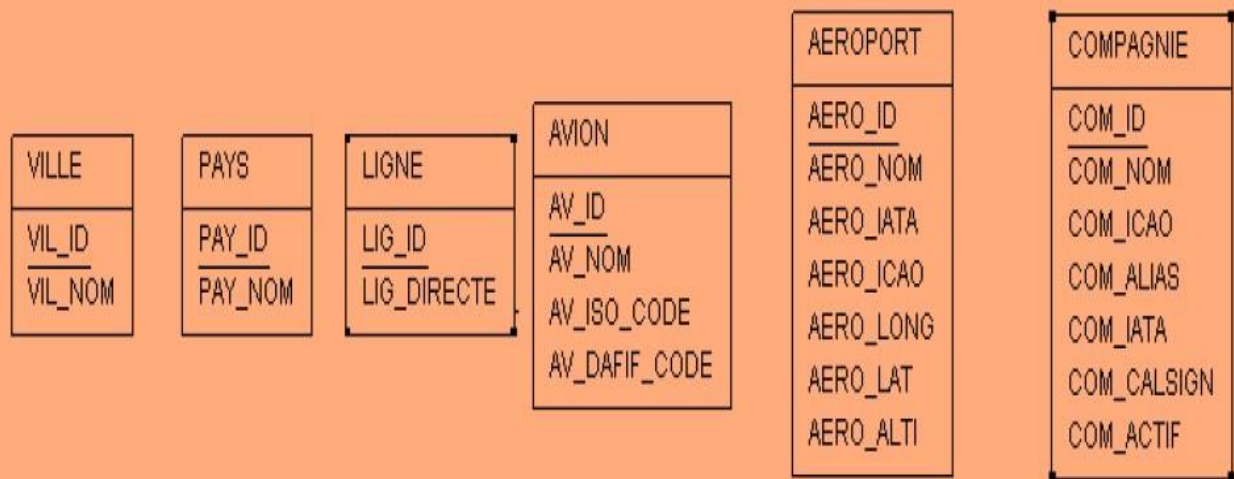
### 2. Remarques

La tâche qui nous incombe, ne prendra pas en compte toutes les dimensions réelles d'un projet de gestion du trafic aérien telles que la fluidité, la sécurité du trafic, la sûreté et autres. Nous nous limiterons tout simplement à une application pouvant servir à observer la densité du trafic entre deux pays ou villes données, à établir une prévisibilité de congestion pour un aéroport donné par l'indicateur du nombre de compagnies qui y desservent, préparer, modifier ou chercher un plan de vol réalisé ou à réaliser par un aéronef donné.

### 3. Préparation des associations

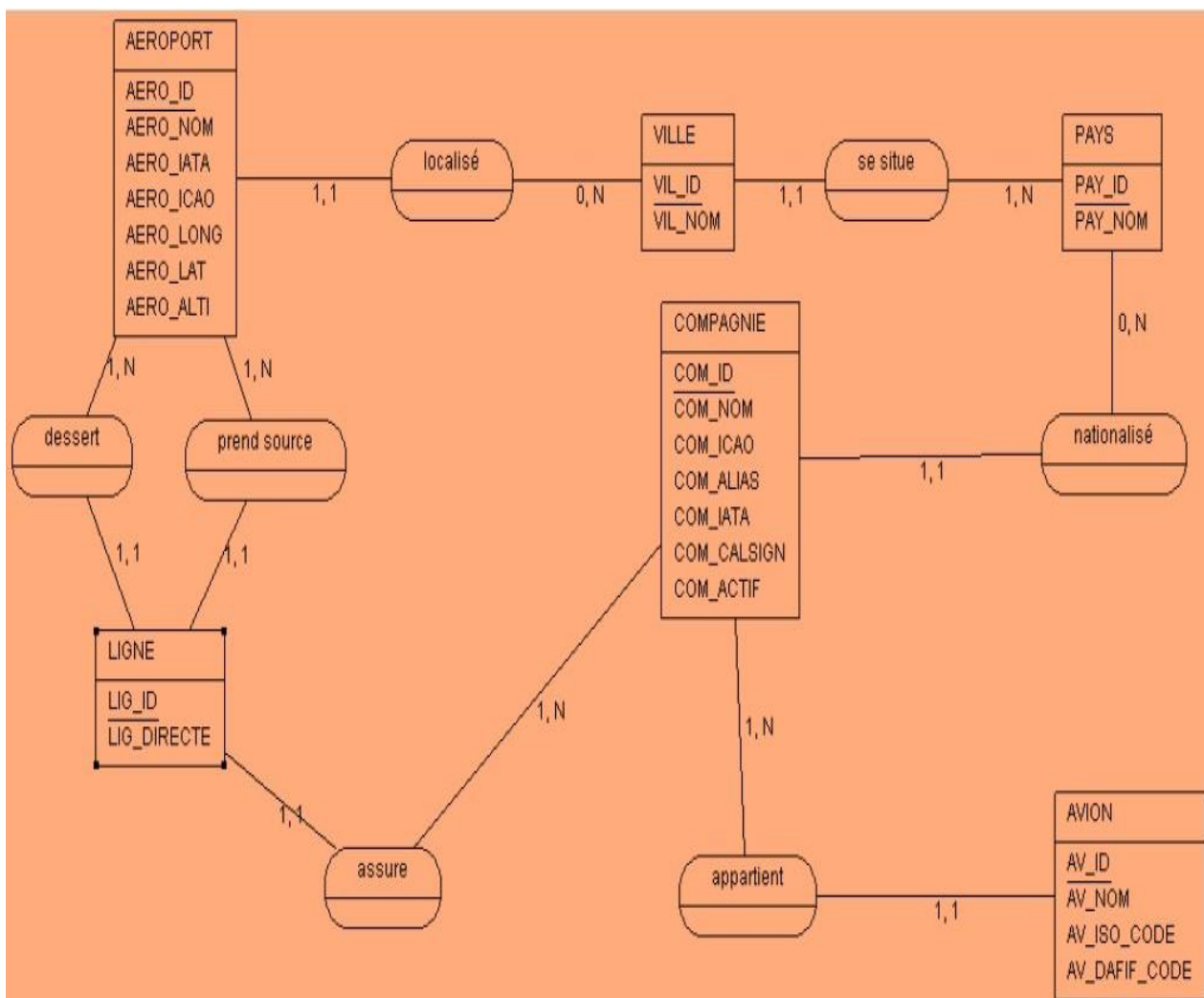
- Un aéroport est localisé dans une ville précise
  - Une ville est située dans un pays donné
  - Un avion appartient à une compagnie
  - Une compagnie est originaire d'un pays
  - Une ligne aérienne est assurée entre deux aéroports donnés par une compagnie
- Notons qu'il n'y aura pas de route ici comme table ni comme entité, bien qu'elle soit définie dans le jeu des données. En effet la dépendance fonctionnelle N°4 ne permet pas de l'avoir ! C'est notre application qui devra le fournir. Donc la table ligne ici aura un attribut booléen DIRECT (oui /non) qui permettra de retrouver le nombre d'escales qui est appelé dans le fichier des données number of stops. Il sera recalculé par l'application.

### 4. Récolte des entités



Notons qu'en schéma relationnel, il faudrait que nous codions l'identifiant de la table ligne par un triplet de clés primaires car en réalité c'est une association ternaire que j'ai décomposé en binaire.

## 5. Modèle Conceptuel de données



## 6. Modèle logique de données

```
COMPAGNIE (COM_ID, COM_NOM, COM_ICAO, COM_ALIAS,  
COM_IATA, COM_CALSIGN, COM_ACTIF, #PAY_ID)  
PAYS (PAY_ID, PAY_NOM)  
VILLE (VIL_ID, VIL_NOM, #PAY_ID)  
AEROPORT (AERO_ID, AERO_NOM, AERO_IATA, AERO_ICAO,  
AERO_LONG, AERO_LAT, AERO_ALTI, #VIL_ID)  
AVION (AV_ID, AV_NOM, AV_ISO_CODE, AV_DAFIF_CODE,  
#COM_ID)  
LIGNE (#AERO_ID_SOURCE, #AERO_ID_DESTI, #COM_ID,  
LIG_DIRECTE,)
```