

## Administration et LDD

### Objectifs

- Découverte de `psql`
- Administration d'une base de données et manipulation du LDD.

### Introduction

Nous vous proposons d'implanter une solution de la base de données « cabinet médical » conçue en travaux dirigés. La solution choisie est celle du diagramme entité-association ci-dessous.

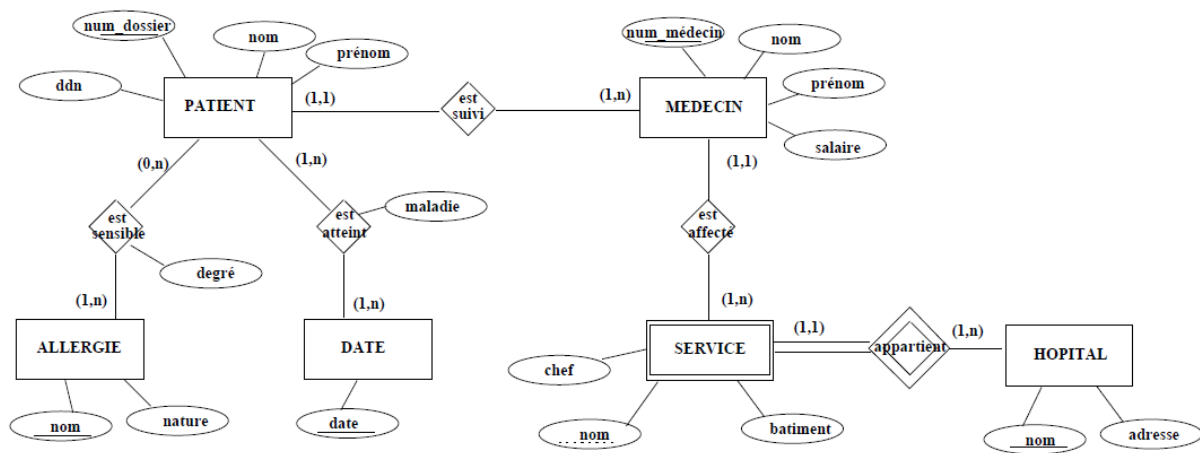


Fig 1. Diagramme entité-association

### Les tables à définir et les contraintes associées

Une implantation relationnelle possible est la suivante (les types des attributs sont suffisamment évidents pour que l'on ne les spécifie pas, on rappellera simplement l'existence du type `serial`):

```

PATIENT (num_dossier, nom, prenom, ddn, num_medecin_traitant)
MEDECIN (num_medecin, nom, prenom, salaire, nom_h, nom_s)
HOPITAL (nom_hopital, adresse)
ALLERGIE (allergie, nature)
ATTEINT (date, num_patient, maladie)
SENSIBLE (allergie, num_patient, degre)
SERVICE (nom_hopital, nom_service, chef, bâtiment)
    
```

Les contraintes à définir sont de plusieurs types :

1. *clés primaires* : attributs soulignés du diagramme et pour les tables traduisant une association, la clé est définie par les attributs des clés primaires des tables associées (le cas de l'entité faible est évidemment à particulariser)
2. *clés étrangères* ou *contraintes d'intégrité référentielle* : les attributs ci-dessous sont des clés étrangères pour les tables indiquées (l'origine de ces attributs est indiquée dans les deux colonnes suivantes, quel attribut, quelle table) :

```
num_medecin_traitant PATIENT num_medecin MEDECIN  
hopital MEDECIN nom_hopital SERVICE  
service MEDECIN nom_service SERVICE  
num_patient ATTEINT num_dossier PATIENT  
num_patient SENSIBLE num_dossier PATIENT  
allergie SENSIBLE allergie ALLERGIE  
nom_hopital SERVICE nom_hopital HOPITAL
```

L'action CASCADE sera choisie en cas de destruction.

3. *contraintes d'intégrité* : on exige que tout salaire soit supérieur à 2000 euros ;

4. *autres contraintes statiques* : les numéros de dossiers et de médecins sont *des séquences* ; les attributs qualifiant un nom au sein de toutes les entités ne doivent pas être NULL.

### **Déroulement du TP**

Pour chaque étudiant il y a une BD identifiée par son login, c'est dans cette base qu'il faut faire les manips.

#### ***Connexion à la base de données***

On utilisera l'outil psql, dans un terminal vous tapez la commande suivante :

```
psql -h gailuron -U <login> -d <nom_base>
```

gailuron est le nom de la machine sur laquelle s'exécute le serveur, <login> est le login de l'utilisateur, <nom\_base> est la base vers laquelle on se connecte.

#### ***Mise à jour du contenu de la base : création des tables***

L'ordre de création des tables n'est pas anodin : en considérant les contraintes spécifiées, vous déterminez un ordre cohérent de création puis vous créez l'ensemble des tables et leurs contraintes.

#### ***Insertion dans les tables et tests***

Pour tester la bonne implantation de la base et la vérification automatique des contraintes, vous insérez des n-uplets corrects puis erronés, c'est-à-dire violant l'une des contraintes spécifiées.

En particulier, vous tentez l'insertion d'un nouveau médecin travaillant dans un hôpital existant mais affecté dans un service non référencé dans cet hôpital (mais existant dans d'autres hôpitaux).

Cette insertion doit être évidemment impossible pour conserver une base cohérente.

#### ***Les clés étrangères***

Afin de montrer le fonctionnement automatique lié à la prise en compte des clés étrangères définies, vous détruisez un hôpital dans lequel au moins un service est défini et dans ce service travaille au moins un médecin ayant sous sa responsabilité des patients.

Que constatez-vous pour les autres tables (SERVICE, MEDECIN, PATIENT...).