

# CHOIX DE CONCEPTION

Nous avons décidé de découper l'application en trois parties :

- Analyseur syntaxique : tout ce qui est relatif à l'analyse des requêtes grâce à JavaCC
- BDD : tout ce qui est relatif au noyau de la base de données
  - + Table : décliné en 2 versions :
    - > Tables stockés sur le disque dur, avec chaque ligne indexées dans un fichier d'index
    - > Table stockées en mémoire vive : elles permettent de pouvoir réaliser plusieurs opérations à la suite dessus (ex : Selection sur un table stockés sur le disque dur, cette selection renvoie une table temporaire stockée en mémoire, on va ensuite faire une projections sur celle-ci)
  - + Schema : déclinés en 2 versions :
    - > Schema stockée sur le disque dur dans un fichier
    - > Schema temporaire avec seulement une liste d'attributs (pour les tables temporaires)
  - + Attribut : représente un attribut du schéma
  - + Tuple : un ligne de la table, représenté par un index
  - + Donnee : represente une case de la table
- Gestion : toutes les opérations relatives à la récupération de données précises dans la base (difference, jointure, union, produit cartésien, selection, max, min, count, avg)

Ainsi avec cette organisation on sépare les différentes fonctions de la base de données, ce qui permet une réutilisation du code. En effet on pourra réutiliser l'analyseur syntaxique seul, le noyau seul et les operations seules. Ainsi l'ajout de fonctionnalité sera facilité.