

# TP: Plans d'exécution

Pierre Senellart ([pierre.senellart@ens.fr](mailto:pierre.senellart@ens.fr))

3 mai 2017

Le but de ce TP est d'examiner les plans d'exécution utilisés par PostgreSQL pour les sélections et jointures, et de comprendre dans quelles circonstances les différents opérateurs physiques sont utilisés.

## Rendu

Le rendu de ce TP doit être fait par mél, avant 23:59 le 3 mai, à [pierre.senellart@ens.fr](mailto:pierre.senellart@ens.fr) en fournissant, en attachement du mél, un fichier texte ou PDF contenant un compte-rendu de vos réponses aux questions (y compris les ordres SQL utilisés). Des pénalités seront comptées en cas de rendu tardif (0,5 point par heure de retard).

## EXPLAIN en PostgreSQL

Dans PostgreSQL, l'opérateur **EXPLAIN** préfixant une requête indique à PostgreSQL d'afficher le plan d'exécution de cette requête plutôt que de l'évaluer.

On utilisera la base de données Unicode, téléchargeable depuis <http://ftp.unicode.org/Public/UNIDATA/UnicodeData.txt>, en utilisant les ordres fournis sur la page du cours ([tp9-schema.sql](#)) pour la création des tables.

Dans les questions qui suivent, le but du jeu est de construire une requête dont le plan d'exécution fait apparaître chacun des opérateurs demandés, et d'expliquer en quelques lignes pourquoi l'optimiseur de PostgreSQL fait le choix de cet opérateur plutôt qu'un autre.

1. **SeqScan**. Parcours linéaire d'une table.
2. **Index Scan**. Accès à un index, suivi d'un accès à la table correspondante.
3. **Index Only Scan**. Accès à un index sans avoir besoin d'accéder à la table correspondante.
4. **Bitmap Index Scan** et **Bitmap Heap Scan**. Accès à des entrées multiples d'un index, et accès à la table correspondante, avec un mécanisme empêchant d'accéder deux fois à la même page disque de la base.
5. **BitmapOr** Sélection disjonctive par union d'identifiants de tuples.
6. **BitmapAnd** Sélection conjonctive par intersection d'identifiants de tuples.
7. **Filter** Filtre manuel sur des résultats d'un **Scan**.
8. **Nested Loop**. Jointure par boucles imbriquées.
9. **Hash Join**. Jointure par hachage.
10. **Merge Join**. Jointure par fusion.