

## TRAVAUX PRATIQUES

Cette série de question utilise la base **magasin**. La base **magasin** peut être téléchargée depuis [https://dali.bo/tp\\_magasin](https://dali.bo/tp_magasin) (dump de 96 Mo, pour 667 Mo sur le disque au final). Elle s'importe de manière très classique (une erreur sur le schéma **public** déjà présent est normale), ici dans une base nommée aussi **magasin** :

```
curl -L https://dali.bo/tp_magasin -o magasin.dump  
pg_restore -d magasin magasin.dump
```

Toutes les données sont dans deux schémas nommés **magasin** et **facturation**.

### Index « simples »



**But :** Mettre en avant un cas d'usage d'un index « simple »

Considérons le cas d'usage d'une recherche de commandes par date. Le besoin fonctionnel est le suivant : renvoyer l'intégralité des commandes passées au mois de janvier 2014.

Créer la requête affichant l'intégralité des commandes passées au mois de janvier 2014.

Afficher le plan de la requête, en utilisant EXPLAIN (ANALYZE, BUFFERS). Que constate-t-on ?

Nous souhaitons désormais afficher les résultats à l'utilisateur par ordre de date croissante.

Réécrire la requête par ordre de date croissante. Afficher de nouveau son plan. Que constate-t-on ?

Maintenant, nous allons essayer d'optimiser ces deux requêtes.

Créer un index permettant de répondre à ces requêtes.

Afficher de nouveau le plan des deux requêtes. Que constate-t-on ?

Maintenant, étudions l'impact des index pour une opération de jointure. Le besoin fonctionnel est désormais de lister toutes les commandes associées à un client (admettons, dont le `client_id` vaut 3), avec les informations du client lui-même.

Écrire la requête affichant `commandes.nummero_commande` et `clients.type_client` pour `client_id = 3`. Afficher son plan. Que constate-t-on ?

Créer un index pour accélérer cette requête.

Afficher de nouveau son plan. Que constate-t-on ?

## Sélectivité



**But :** Comprendre la sélectivité des index.

Écrire une requête renvoyant l'intégralité des clients qui sont du type entreprise ('E'), une autre pour l'intégralité des clients qui sont du type particulier ('P').

Ajouter un index sur la colonne `type_client`, et rejouer les requêtes précédentes.

Afficher leurs plans d'exécution. Que se passe-t-il ? Pourquoi ?

## Index partiels



**But :** Mettre en avant un cas d'usage d'un index partiel

Sur la base fournie pour les TPs, les lots non livrés sont constamment requêtés. Notamment, un système d'alerte est mis en place afin d'assurer un suivi qualité sur les lots expédié depuis plus de 3 jours (selon la date d'expédition), mais non réceptionné (date de réception à NULL).

Écrire la requête correspondant à ce besoin fonctionnel (il est normal qu'elle ne retourne rien).

Afficher le plan d'exécution.

Quel index partiel peut-on créer pour optimiser ?

Afficher le nouveau plan d'exécution et vérifier l'utilisation du nouvel index.

## Index fonctionnels



**But :** Cas d'usage d'un index fonctionnel

Ce TP utilise la base **magasin**. La base **magasin** peut être téléchargée depuis [https://dali.bo/tp\\_magasin](https://dali.bo/tp_magasin) (dump de 96 Mo, pour 667 Mo sur le disque au final). Elle s'importe de manière très classique (une erreur sur le schéma **public** déjà présent est normale), ici dans une base nommée aussi **magasin** :

```
curl -L https://dali.bo/tp_magasin -o magasin.dump  
pg_restore -d magasin magasin.dump
```

Toutes les données sont dans deux schémas nommés **magasin** et **facturation**.

Pour répondre aux exigences de stockage, l'application a besoin de pouvoir trouver rapidement les produits dont le volume est compris entre certaines bornes (nous négligeons ici le facteur de forme, qui est problématique dans le cadre d'un véritable stockage en entrepôt !).

Écrire une requête permettant de renvoyer l'ensemble des produits (table `magasin.produits`) dont le volume ne dépasse pas 1 litre (les unités de longueur sont en mm, 1 litre = 1 000 000 mm<sup>3</sup>).

Quel index permet d'optimiser cette requête ? (Utiliser une fonction est possible, mais pas obligatoire.)

## Cas d'index non utilisés



**But :** Mettre en avant des cas d'index inutilisés

Un développeur cherche à récupérer les commandes dont le numéro d'expédition est 190774 avec cette requête :

```
SELECT * FROM lignes_commandes WHERE numero_lot_expedition = '190774'::numeric ;
```

Afficher le plan de la requête.

Créer un index pour améliorer son exécution.

L'index est-il utilisé ? Quel est le problème ?

Écrire une requête pour obtenir les commandes dont la quantité est comprise entre 1 et 8 produits.

Créer un index pour améliorer l'exécution de cette requête.

Pourquoi celui-ci n'est-il pas utilisé ? (Conseil : regarder la vue `pg_stats`)

Faire le test avec les commandes dont la quantité est comprise entre 1 et 4 produits.