

Exploitation d'une Base de Données

Cours 7 - Vues

Anaïs Durand

4 Mai 2023

Vue = table virtuelle

- ▶ Résultat d'une requête **SELECT**
- ▶ Nommée

⇒ utilisable comme une table

Vue = table virtuelle

- ▶ Résultat d'une requête **SELECT**
- ▶ Nommée

⇒ utilisable comme une table

- ▶ Pas de données stockées

⇒ À l'exécution, le SGBD remplace la vue par le code de la requête **SELECT** correspondant

Création d'une vue

```
CREATE VIEW nom_de_la_vue AS  
SELECT ...;
```

Exemple :

```
CREATE VIEW PetitColis AS  
SELECT c.num, c.dateExped, e.num, e.gps, cl.*  
FROM Colis c, Entrepot e, Client cl  
WHERE c.entrepot = e.id AND c.client = cl.code  
      AND c.poids <= 0.25  
      AND c.largeur*c.longueur*c.hauteur <= 1;
```

Utilisation d'une vue

- S'utilise comme une table :

```
SELECT *  
FROM PetitColis  
WHERE codePostal = '63000';
```



En lecture seule ! (Sauf cas particuliers)

- Supression :

```
DROP VIEW PetitColis;
```

À quoi ça sert ?

- ▶ Simplification : Simplifie l'écriture des requêtes.

À quoi ça sert ?

- ▶ **Simplification** : Simplifie l'écriture des requêtes.
- ▶ **Encapsulation** : Permet de rendre visible seulement certaines données (via les vues) et pas l'intégralité des tables.

À quoi ça sert ?

- ▶ **Simplification** : Simplifie l'écriture des requêtes.
- ▶ **Encapsulation** : Permet de rendre visible seulement certaines données (via les vues) et pas l'intégralité des tables.
- ▶ **Flexibilité** : Permet de modifier la structure des tables sous-jacentes sans modifier les requêtes utilisant les vues.

Vues Matérialisées

Vue dont les données sont *stockées*.

```
CREATE MATERIALIZED VIEW PetitColis AS
SELECT c.num, c.dateExped, e.num, e.localisation,
       cl.*
FROM Colis c, Entrepot e, Client cl
WHERE c.entrepot = e.num AND c.client = cl.num
      AND c.poids <= 0.25
      AND c.largeur*c.longueur*c.hauteur <= 1;
```

Vues Matérialisées

Vue dont les données sont *stockées*.

```
CREATE MATERIALIZED VIEW PetitColis AS
SELECT c.num, c.dateExped, e.num, e.localisation,
       cl.*
FROM Colis c, Entrepot e, Client cl
WHERE c.entrepot = e.num AND c.client = cl.num
      AND c.poids <= 0.25
      AND c.largeur*c.longueur*c.hauteur <= 1;
```

Les modifications des tables sous-jacentes ne seront pas prises en compte sauf si la vue est manuellement *rafraîchie* :

```
REFRESH MATERIALIZED VIEW PetitColis;
```

Quand utiliser une vue matérialisée ?

- Besoin d'un accès rapide à des données issues d'une requête complexe.

Les données de la vue sont stockées \Rightarrow l'exécution de la requête a été fait au préalable.

Quand utiliser une vue matérialisée ?

- Besoin d'un accès rapide à des données issues d'une requête complexe.

Les données de la vue sont stockées \Rightarrow l'exécution de la requête a été fait au préalable.

ET

- Pas besoin de données systématiquement à jour

Les modifications depuis le dernier rafraîchissement de la vue ne sont pas prises en compte.

Quand utiliser une vue matérialisée ?

Exemple :

```
CREATE MATERIALIZED VIEW FacturesImpayees AS
SELECT f.noFacture facture,
       f.montant montantTotal,
       sum(p.montant) montantPaye
FROM Facture f, Paiement p
WHERE f.noFacture = p.Facture
GROUP BY f.noFacture, f.montant
HAVING sum(p.montant) < f.montant;
```

Solde impayé :

```
SELECT sum(montantTotal - montantPaye)
FROM FacturesImpayees;
```

Pas besoin de la valeur exacte \Rightarrow rafraîchissement une fois par jour,
par semaine, ...