

Examen Administration des SGBD

M1 : IL-SII-MIV

Enoncé :

Une entreprise utilise une base de données relationnelle décrite par le schéma relationnel ci-dessous pour gérer les drones et missions, avec différents types d'utilisateurs : administrateurs, opérateurs de drones, superviseurs et contrôleur de qualité.

Modele (modeld, modelnom)

Drone (numd, modeld, dateaqcui, statut);

Mission (codem, numd, typemid, zid, datedeb, datefin);

Utilisateurs (idp, nom, prenom, email, role) ;

TypeMission (typemid, type) ;

Zone (zid, region, coordonnees) ;

En gras et souligné les clés primaires, en italique les clés étrangères.

- Statut : cette colonne indique si le drone est : Disponible, En Maintenance ou En Mission.
- La colonne Type de mission prend les valeurs : Agriculture, Sécurité ou Environnement ;
- Rôle peut être : administrateurs, opérateurs de drones, superviseurs ou contrôleur de qualité
- Région : Est, Ouest, Sud ou Nord.

Nous disposons des informations suivantes :

Modele	Card(Modele)=1000
Drone	Card(Drone)=100000
Mission	Card(Mission)=10000000
Utilisateur	Card(Utilisateur)=100
TypeMission	Card(TypeMission)= 3
Zone	Card(Zone)= 4000
On suppose que chaque attribut est codé sur 50 octets. Que la base de données contient les missions des drones datées entre le 01-01-2020 et 31-12-2024.	

Exercice 1 : Fonctions générales de SGBD et Optimisation (12 points)

1. Le DBA exécute les requêtes suivantes :

<pre>CREATE ROLE ROLE_ADMIN; CREATE ROLE ROLE_OPERATEUR; CREATE ROLE ROLE_SUPERVISEUR; CREATE ROLE ROLE_CONTROLEUR_QUALITE; GRANT ALL PRIVILEGES ON Drone TO ROLE_ADMIN; GRANT ALL PRIVILEGES ON Mission TO ROLE_ADMIN; GRANT ALL PRIVILEGES ON Utilisateur TO ROLE_ADMIN; GRANT SELECT ON Drone TO ROLE_OPERATEUR; GRANT INSERT, UPDATE ON Mission TO ROLE_OPERATEUR; GRANT SELECT ON Drone TO ROLE_SUPERVISEUR; GRANT SELECT ON Mission TO ROLE_SUPERVISEUR; GRANT SELECT ON Drone TO ROLE_CONTROLEUR_QUALITE; GRANT UPDATE (status) ON Drone TO ROLE_CONTROLEUR_QUALITE;</pre>	<pre>CREATE PROFILE PROFIL_OPERATEUR LIMIT SESSIONS_PER_USER 2 CONNECT_TIME 120 PRIVATE_SGA 50M; CREATE USER admin IDENTIFIED BY admin_pass; GRANT ROLE_ADMIN TO admin; CREATE USER operateur IDENTIFIED BY psw; GRANT ROLE_OPERATEUR TO operateur; ALTER USER operateur PROFILE PROFIL_OPERATEUR; CREATE USER superviseur IDENTIFIED BY psw; GRANT ROLE_SUPERVISEUR TO superviseur; CREATE USER controleur IDENTIFIED BY psw; GRANT ROLE_CONTROLEUR_QUALITE TO controleur;</pre>
---	---

- a. Quels privilèges l'utilisateur opérateur possède-t-il ? Quelles sont les limitations imposées par son profil ?
- b. Un superviseur essaie de modifier une mission ? Quelle sera la réponse du SGBD. Justifiez.

- c. L'utilisateur contrôleur exécute la requête suivante grant UPDATE (status) ON Drone TO superviseur ; Quelle sera la réponse du SGBD. Justifiez.
2. Donnez le script SQL pour créer la table **mission** avec toutes les contraintes d'intégrité sachant que les autres tables sont déjà créées.
3. L'administrateur souhaite suivre le nombre total de missions affectées à chaque drone.
 - a. Ecrivez une fonction qui calcule le nombre de missions affectées à un drone.
 - b. Comment garantir que le nombre de missions affectées à un drone ne dépasse pas 100 missions ?
4. L'administrateur a constaté que les requêtes les plus fréquentes sur la table **mission** utilisent les colonnes **codem** et **datedeb**.
 - a. Écrivez la commande SQL pour créer un index B-Arbre sur la colonne **datedeb**.
 - b. Expliquez pourquoi un index B-Arbre est mieux adapté qu'un index bitmap pour une colonne comme **datedeb**.
 - c. Si la colonne **datedeb** subit des mises à jour fréquentes, comment cela peut-il affecter les performances de l'index B-Arbre ?
 - d. Pourquoi l'index B-Arbre est particulièrement utile pour les requêtes utilisant **ORDER BY** ?
5. En exécutant la requête ci-dessous ; L'administrateur a constaté qu'elle est lente.

```
SELECT distinct d.numd, d. modeld
FROM Drone d, Mission m, TypeMission t
WHERE d.numd = m.numd
AND m. typem=t. typemid
AND m.datedeb > '01-01-2024'
AND t.type = 'Agriculture'
ORDER BY d.numd, d. modeld DESC;
```

- a. Que calcule cette requête ?
- b. Quels sont les facteurs qui peuvent ralentir cette requête ?
- c. Indiquez les index à créer pour améliorer le temps de réponse de cette requête.
- d. Donnez un plan logique de cette requête et calculer la taille de résultat.

NB : - Pour $A > c$, le Facteur de Sélectivité est défini comme suit $(\text{Max}(A) - c) / (\text{Max}(A) - \text{Min}(A))$.

- Pour éliminer les doublons d'une relation $R(a_1, \dots, a_n)$ l'estimation de taille du résultat est donnée par la formule suivante $\text{MIN} [\text{Val} (R, a_1) * \dots * \text{Val}(R, a_n), 1/2 * \text{Card}(R)]$.

Exercice 2 : Gestions des accès concurrents (08 points)

1. Considérons l'ordonnancement suivant $R1(X) W4(Z) W2(X) R2(Y) W3(Y) W3(X) R1(Y) R4(Y) W1(Z)$
 - a. Vérifiez si cet ordonnancement est sérialisable en identifiant les conflits et en construisant le graphe de précedence.
 - b. L'algorithme d'estampillage avec deux estampilles accepte cet ordonnancement sans rejet ?
2. Considérons les transactions suivantes :

$T1 : R1(A) W1(C) ;$	$T2 : R2(B) W2(A) ;$	$T3 : R3(C) W3(D) ;$	$T4 : W4(D) R4(B) ;$
----------------------	----------------------	----------------------	----------------------

Répondez aux questions suivantes, en fournissant une justification détaillée pour chaque réponse.

- a. Combien d'ordonnements non sérialisables peut-on obtenir avec $T1, T2, T3$ et $T4$?
 - b. Montrez qu'en utilisant le mécanisme d'accès par verrouillage à deux phases, aucun ordonnancement de $T1, T2, T3$ et $T4$ ne peut engendrer un blocage mutuel ?
3. Considérons un contexte avec des verrous partagés et exclusifs. Une transaction T est dite **prudente** si toutes ses demandes de verrouillage apparaissent avant toute autre action de T (par exemple $T1 : SLOCK1(X) XLOCK1(Y) XLOCK1(Z) R1(X) W1(Y) W1(Z)$). Un ordonnancement est dit **prudent** si toutes ses transactions sont prudentes. Un ordonnanceur est dit **conservateur** s'il se comporte comme un ordonnanceur V2P avec la seule restriction supplémentaire qu'il n'accepte pas les ordonnancements qui ne sont pas prudents.

Prouvez ou réfutez les affirmations suivantes.

- a. Tout ordonnancement prudent est sérialisable.
- b. Tout ordonnanceur conservateur peut éviter de gérer les interblocages.
- c. Comparez les deux types d'ordonnanceurs V2P et conservateur ? En termes Sérialisabilité, interblocages et concurrence.

Bon courage