

Technologie du serveur de base de données

J. Akoka - I. Wattiau

1

Objectif

- Comprendre comment fonctionne un SGBD pour :
 - faire les bons choix de paramétrage
 - configurer correctement les services offerts par un SGBD
- C'est une des tâches de l'administrateur de base de données (DBA)

2

1. Composants d 'un serveur de BD

- Optimiseur :
 - décide de la meilleure stratégie d 'accès aux données, traite les requêtes et les transforme en interne pour en faciliter l 'exécution
- Gestionnaire de méthodes d 'accès :
 - effectue les accès physiques aux données, selon la stratégie définie par l'optimiseur
- Gestionnaire de tampon :
 - responsable des transferts de page de mémoire secondaire en mémoire centrale

3

1. Composants d 'un serveur de BD (suite)

- Système de contrôle de la fiabilité :
 - préserve le contenu de la BD en cas de panne
- Système de contrôle de concurrence
 - régule les accès simultanés
 - empêche les incohérences en cas d 'interférences entre applications

4

2. Les transactions

- Définition : unité élémentaire de l'activité d'une application, à laquelle on alloue des caractéristiques de fiabilité et de cohérence
- on peut considérer une transaction comme une suite d'opérations de lecture et d'écriture dans la base
- NB : une modification = une lecture + une écriture

5

2. Les transactions (suite)

- Syntaxiquement :
 - délimitée par les deux instructions
 - begin transaction (bot) - implicite sous Oracle
 - end transaction (eot)
 - à l'intérieur, deux instructions particulières
 - commit : validation
 - abort ou rollback : annulation
 - L'instruction d'annulation peut être décidée par l'utilisateur ou par le système

6

2. Les transactions (suite)

- Exemple : transfert bancaire

```
begin transaction
x := x - 10;
y := y + 10;
commit
end transaction
```
- Transaction bien formée :
 - dans toute exécution possible, il y a une seule des deux commandes commit ou rollback
 - pas de mise à jour après la commande commit ou rollback
- Transaction système \diamond transaction utilisateur

7

Propriétés ACID des transactions

- Atomicité : une transaction est une unité d'exécution indivisible
- Cohérence :
 - respecte les contraintes d'intégrité
 - les contraintes de type différé sont vérifiées à la validation
- Isolation : l'exécution est indépendante de l'exécution des autres transactions
- Durabilité : le résultat d'une transaction validée ne doit pas être perdue

8

Atomicité

- Une transaction doit rendre la base dans un état cohérent
- Elle s'exécute en tout-ou-rien : si on ne peut pas la terminer, on doit défaire ce qui a été fait

9

Cohérence

- L'ordre des opérations d'une transaction peut être important
- Exécuter les opérations dans un autre ordre peut conduire à une situation ne vérifiant plus les contraintes d'intégrité (cas des opérations non permutable)

10

Isolation

- Soient les deux transactions ci-dessous :
- T1 :a) $x \leftarrow \text{lire}(X)$; b) $x \leftarrow x - N$; c) $X \leftarrow \text{écrire}(x)$
- T2 :a) $y \leftarrow \text{lire}(X)$; b) $y \leftarrow y + M$; c) $X \leftarrow \text{écrire}(y)$
- Si l'on exécute :
T1a),T1b),T2a),T2b),T1c),T2c), l'effet de l'opération T1b) est perdue
- T1 doit travailler en isolation jusqu'à sa terminaison.

11

Durabilité des modifications

- Non-répétabilité des lectures : y a changé
- Tuples fantômes :
une transaction visualise des tuples avant et après qu'une autre les supprime

T1	T2
$x \leftarrow \text{lire}(X)$	
	$y \leftarrow \text{lire}(X)$
	afficher(y)
$x \leftarrow x + 100$	
$X \leftarrow \text{écrire}(x)$	
	$y \leftarrow \text{lire}(X)$
	afficher(y)

12

Propriétés ACID des transactions

Atomicité	Système de contrôle de la fiabilité
Cohérence	Compilateur du DDL
Isolation	Système de contrôle de concurrence
Durabilité	Système de contrôle de la fiabilité