

TP Bases de données avancées – Transactions et concurrence – SGBD Postgresql

Remarques préliminaires :

La base de données à utiliser est celle du "transports dans les établissements de soins" (cf. plateforme pédagogique pour plus de détails)

Le SGBD Postgresql est souvent en mode « AUTOCOMMIT » mais il propose la possibilité d'utiliser la commande BEGIN TRANSACTION pour marquer explicitement le début d'une transaction.

Si nécessaire, entre 2 questions, vous pouvez réinitialiser le jeu de données de la BD en utilisant le script adéquat (cf. plateforme pédagogique).

Transactions

- a)** Exécuter une transaction qui modifie le type du Container 98 en 'Repas', qui vérifie la modification et qui se termine par une validation de la modification. Après la transaction, afficher les caractéristiques du Container 98.

(0)	Commencer une transaction	
(1)	Afficher le type du container 98	
(2)	Modifier le type du container 98 en 'Repas'	
(3)	Afficher le type du container 98	
(4)	Valider la modification	
(5)	Afficher le type du container 98	

- b)** Exécuter une transaction qui modifie le type du Container 98 en 'Analyse', qui vérifie la modification et qui se termine par une annulation de la modification. Après la transaction, afficher les caractéristiques du Container 98.

(0)	Commencer une transaction	
(1)	Afficher le type du container 98	
(2)	Modifier le type du container 98 en 'Analyse'	
(3)	Afficher le type du container 98	
(4)	Annuler la modification	
(5)	Afficher le type du container 98	

- c)** Point de sauvegardes intermédiaires : exécuter une transaction qui :

(0)	Commence une transaction	
(1)	Affiche les caractéristiques des containers	
(2)	Modifie le type du Container 98 en 'Analyse'	
(3)	définit un point intermédiaire de sauvegarde 'un'	
(4)	modifie le type du Container 106 en 'Courrier'	
(5)	définit un point intermédiaire de sauvegarde 'deux'	
(6)	modifie le type du Container 116 en 'Déchets'	
(7)	définit le point de sauvegarde 'trois'	
(8)	affiche tout les containers	

(9)	annule les modifications jusqu'au point 'un'	
(10)	enregistre les modifications	
(11)	affiche tout les containers	

conclusion ?

Accès concurrents

Afin de tester, l'exécution de 2 transactions en accès concurrentiel, créer une deuxième connexion avec le même nom d'utilisateur. Pour la suite, on nommera ces 2 connexions (1) et (2).

d) Mise en évidence du pb de *lecture non reproductible*.

Session 1		Session 2	
	débuter une transaction		débuter une transaction
	se mettre en mode d'isolation 'read committed'		se mettre en mode d'isolation 'read committed'
	visualiser le type du Container 98		visualiser le type du Container 98
			modifier le type du Container 98 en 'Repas'
	visualiser le type du Container 98		visualiser le type du Container 98
			valider la modification
	visualiser le type du Container 98		visualiser le type du Container 98
	terminer la transaction		
			visualiser le type du Container 98

conclusion ?

e) Mettre en évidence un pb de *lecture fantôme*

Session 1		Session 2	
	débuter une transaction		débuter une transaction
	se mettre en mode d'isolation 'read committed'		se mettre en mode d'isolation 'read committed'
	liste des containers de type 'Linge'		liste des containers de type 'Linge'
			ajouter le Container (645, 'Linge', 'Hors service', 'Vide')
	liste des containers de type 'Linge'		liste des containrs de type 'Linge'
			valider l'ajout
			débuter une transaction
	liste des containers de type 'Linge'		liste des containers de type 'Linge'
			Supprimer le container 645
	liste des containers de type 'Linge'		liste des containers de type 'Linge'
			Valider la suppression
	liste des containers de type 'Linge'		liste des containers de type 'Linge'

conclusion ?

f) Reprendre la séquence du *d)* mais en mode isolation 'serializable' pour la transaction de la session (1)

Session 1		Session 2	
	débuter une transaction		débuter une transaction
	se mettre en mode d'isolation 'serializable'		se mettre en mode d'isolation 'read committed'
	visualiser le type du Container 98		visualiser le type du Container 98
			modifier le type du Container 98 en 'Repas'
	visualiser le type du Container 98		visualiser le type du Container 98
			valider la modification
	visualiser le type du Container 98		visualiser le type du Container 98
	terminer la transaction		
	visualiser le type du Container 98		visualiser le type du Container 98

conclusion ?

g) reprendre la séquence du e) mais en mode isolation 'serializable' pour la transaction de la session (1)

Session 1		Session 2	
	débuter une transaction		débuter une transaction
	se mettre en mode d'isolation 'serializable'		se mettre en mode d'isolation 'read committed'
	liste des containers de type 'Linge'		liste des containers de type 'Linge'
			ajouter le Container (645, 'Linge', 'Hors service', 'Vide')
	liste des containers de type 'Linge'		liste des containers de type 'Linge'
			valider l'ajout
	liste des containers de type 'Linge'		liste des containers de type 'Linge'
			Supprimer le container 645
	liste des containers de type 'Linge'		liste des containers de type 'Linge'
			Valider la suppression
	liste des containers de type 'Linge'		liste des containers de type 'Linge'

Conclusion ?

Verrous explicites

h) verrou exclusif

Session 1		Session 2	
	débuter une transaction		débuter une transaction
	poser un verrou en mode 'accès exclusif' sur la relation Container		
			modifier le container 98 ; remarque ?
	terminer la transaction		Remarque ?
			terminer la transaction

Verrous implicites

i)

Session 1		Session 2	
	débuter une transaction		débuter une transaction
	visualiser le type du Container 98		visualiser le type du Container 98
	modifier le type du Container 98 en 'Repas'		
	visualiser le type du Container 98		modifier le type du Container 98 en 'Analyse' ; remarque ?
	terminer la transaction		
	visualiser le type du Container 98		visualiser le type du Container 98
			terminer la transaction
	visualiser le type du Container 98		

j)

Session 1		Session 2	
	débuter une transaction		débuter une transaction
	visualiser le type des Container 98 et 106		visualiser le type des Container 98 et 106
	modifier le type du Container 98 en 'Repas'		
	visualiser le type des Container 98 et 106		modifier le type du Container 106 en 'Dechets' ;
	terminer la transaction		visualiser le type des Container 98 et 106
			terminer la transaction
	visualiser le type des Container 98 et 106		visualiser le type des Container 98 et 106

k)

Session 1		Session 2	
	débuter une transaction		débuter une transaction
	visualiser le type des Container 98 et 106		visualiser le type des Container 98 et 106
	modifier le type du Container 98 en 'Repas'		modifier le type du Container 106 en 'Courrier'
	visualiser le type des Container 98 et 106		visualiser le type des Container 98 et 106
	modifier le type du Container 106 en 'Déchets'		modifier le type du Container 98 en 'Linge'
	Remarque ?		Remarque ?
	visualiser le type des Container 98 et 106		visualiser le type des Container 98 et 106
			visualiser le type des Container 98 et 106
	Valider les modifications		
	visualiser le type des Container 98 et 106		visualiser le type des Container 98 et 106

Verrouillage pour modification

la transaction (1) veut rajouter un transport qui utilise le container du type 'Repas'

la transaction (2) veut supprimer le container du type 'Repas' car il n'y a aucun transport qui l'utilise

l)

Session 1		Session 2	
	débuter une transaction		débuter une transaction
	Sélectionner le code du container du type 'Dechets' (le 106)		
			suppression du container du type 'Repas'
	rajouter le transport matériel (9345, 18, codec, 1, 12, null, null, null) – codec est la valeur du code récupéré par la requête précédente		valider la suppression

Remarque ?

m)

Session 1		Session 2	
	débuter une transaction		débuter une transaction
	Sélectionner le code du container du type 'Repas' pour modification		
			suppression du container du type 'Repas'
			remarque ?
	rajouter le transport matériel (9345, 18, codec, 1, 12, null, null, null) – codec est la valeur du code récupéré par la requête précédente remarque ?		
	Valider l'ajout		.

Remarque ?

==> exercice :

Définir une transaction ayant pour but de rajouter un nouveau container ; le code de ce nouveau container sera déterminé en utilisant la valeur maximum des codes container existants. Cette transaction doit pouvoir s'exécuter en concurrence.