

TP IBD (R1.05)

Introduction aux Bases de Données

TP#04 Accès transactionnels à la base de données (BD)

Ce TP vise à comprendre la notion de transaction. Lorsque l'on réalise une séquence de manipulations (SQL-LMD), le SGBD gère ces accès dans le cadre d'une transaction. Cette dernière respecte 4 propriétés désignées ACID, c'est-à-dire

- Atomicité,
- Cohérence,
- Isolation,
- Durabilité.

Nous allons manipuler les données afin d'illustrer et comprendre la signification de ces propriétés.

1 Atomicité

Une transaction est réalisée entièrement ou pas du tout (principe du « tout ou rien »).

(1) Effectuez une transaction (séquence d'ordres SQL-LMD) consistant à

- Ajouter le continent suivant.

```
INSERT INTO Continent (name, area)  
VALUES ('IUT', NULL);
```

- Puis, ajouter le pays suivant.

```
INSERT INTO Country (iso3, iso2, name, capital, rcode, cname)  
VALUES ('UT2', 'U2', 'IUT de Blagnac', 'Blagnac', '155',  
'IUT');
```

(2) Affichez le contenu des deux tables afin de vérifier la présence des deux tuples.

```
SELECT *  
FROM Continent  
WHERE name = 'IUT';  
  
SELECT *  
FROM Country  
WHERE iso3 = 'UT2';
```

(3) Annulez cette transaction (ROLLBACK).

(4) Affichez le contenu des deux tables.

Que constatez-vous ? Expliquez-en quoi la transaction respecte bien le principe d'atomicité.

2 Durabilité

Une transaction validée est définitivement conservée dans le SGBD.

(1) Effectuez la même transaction, mais cette fois en validant la transaction (COMMIT) à l'étape (3).

(2) Déconnectez-vous, puis reconnectez-vous à Oracle.

(3) Affichez le contenu des deux tables.

Que constatez-vous ? Expliquez-en quoi la transaction respecte bien le principe de durabilité.

3 Cohérence

Une transaction transforme la BD d'un état cohérent vers un autre état cohérent. La BD est considérée cohérente lorsque l'ensemble des contraintes d'intégrités (PRIMARY KEY, UNIQUE, FOREIGN KEY, NOT NULL, CHECK) sont vérifiées.

(1) Nous allons autoriser des manipulations plaçant la BD dans des états incohérents. Pour cela, nous allons désactiver la contrainte de clé étrangère cname# de la relation Country.

```
ALTER TABLE Country  
DISABLE CONSTRAINT fk_country_cname;
```

(2) Ajoutez un nouveau pays comme suit :

```
INSERT INTO Country (iso3, iso2, name, capital, rcode, cname)  
VALUES ('UT3', 'U3', 'IUT de Rangueil', 'Toulouse', '155',  
'XXX');
```

Expliquez-en quoi la BD est placée dans un état incohérent.

(3) Validez cette transaction (COMMIT).

On constate que la transaction est validée puisque la contrainte est désactivée. Elle n'est donc pas prise en compte pour l'instant.

- (4) Tentez de réactiver la contrainte de clé étrangère `cname#` de la relation `Country` afin qu'elle soit vérifiée.

```
ALTER TABLE Country  
ENABLE CONSTRAINT fk_country_cname;
```

Que constatez-vous ? Expliquez-en quoi la transaction respecte bien le principe de cohérence.

- (5) Supprimez donc l'enregistrement invalide qui place de fait la BD dans un état incohérent par rapport à la contrainte de clé étrangère `cname#` de la relation `Country`.

```
DELETE FROM Country  
WHERE iso3 = 'UT3';
```

- (6) Validez cette transaction (COMMIT), puis réactivez la contrainte de clé étrangère `cname#` de la relation `Country` afin qu'elle soit vérifiée.

Que constatez-vous ? Expliquez-en quoi la transaction respecte bien le principe de cohérence.

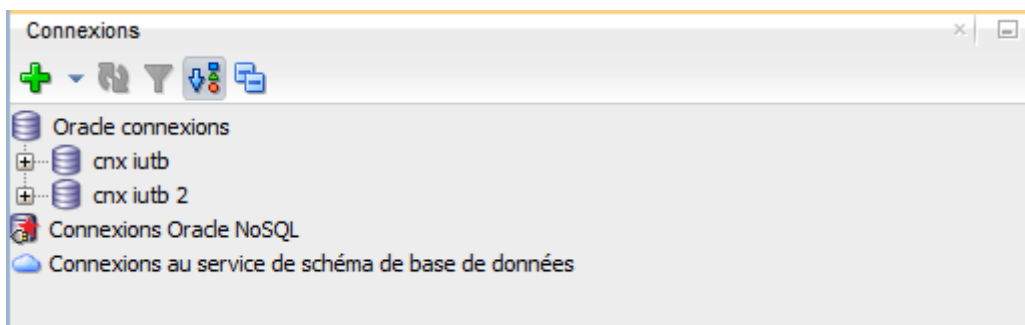
4 Isolation

Une transaction ne peut pas interférer (perturber son exécution) avec une autre

- (1) Créez deux connexions avec SQL Developer pour accéder à votre BD.



Vous devez définir dans SQL Developer deux connexions différentes, en utilisant les mêmes paramètres de connexion. Le but est de simuler deux utilisateurs qui accèdent de manière concurrente à la même BD.



4.1 Accès en lecture/écriture

Connexion #1	Connexion #2
1) Ajoutez deux pays	
<pre>INSERT INTO Country (iso3, iso2, name, capital, rcode, cname) VALUES ('UT3', 'U3', 'IUT de Rangueil', 'Toulouse', '155', 'IUT'); INSERT INTO Country (iso3, iso2, name, capital, rcode, cname) VALUES ('UT1', 'U1', 'IUT de Rodez', 'Rodez', '155', 'IUT');</pre>	
2) Affichez le contenu de la table.	
<pre>SELECT * FROM Country WHERE cname = 'IUT';</pre>	
	3) Affichez le contenu de la table.
	<pre>SELECT * FROM Country WHERE cname = 'IUT';</pre>
	Expliquez-en quoi la transaction respecte bien le principe d'isolation.
4) Validez la transaction (COMMIT).	
	5) Affichez le contenu de la table.
	Expliquez-en quoi la transaction respecte bien le principe d'isolation.

4.2 Accès en écriture/écriture

Connexion #1	Connexion #2
<p>1) Supprimez un pays.</p> <pre>DELETE FROM Country WHERE iso3 = 'UT1';</pre> <p>2) Modifiez un pays.</p> <pre>UPDATE Country SET capital = 'Rangueil' WHERE iso3 = 'UT3';</pre>	<p>3) Supprimez l'un des deux pays en cours de manipulation par la transaction concurrente Connexion #1.</p> <pre>DELETE FROM Country WHERE iso3 = 'UT3';</pre> <p>Expliquez pourquoi la transaction est placée en attente. En quoi la transaction respecte bien le principe d'isolation.</p> <p>Représentez la table avec les verrous apposés par la transaction concurrente Connexion #1.</p>
<p>4) Validez la transaction (COMMIT).</p>	<p>Expliquez ce qui se passe. En quoi la transaction respecte bien le principe d'isolation.</p> <p>Représentez la table avec les verrous apposés par la transaction Connexion #2.</p>
	<p>5) Validez la transaction (COMMIT).</p> <p>6) Affichez le contenu de la table.</p>

4.3 Gestion des inter-blocages

Connexion #1	Connexion #2
<p>1) Ajoutez un pays.</p> <pre>INSERT INTO Country (iso3, iso2, name, capital, rcode, cname) VALUES ('UT3', 'U3', 'IUT de Rangueil', 'Toulouse', '155', 'IUT');</pre> <p>2) Validez la transaction (COMMIT).</p>	<p>3) Ajoutez un autre pays.</p> <pre>INSERT INTO Country (iso3, iso2, name, capital, rcode, cname) VALUES ('UT1', 'U1', 'IUT de Rodez', 'Rodez', '155', 'IUT');</pre> <p>4) Validez la transaction (COMMIT).</p> <p>5) Modifiez le pays.</p> <pre>UPDATE Country SET capital = 'Capitole' WHERE iso3 = 'UT1';</pre>
<p>6) Modifiez le pays.</p> <pre>UPDATE Country SET capital = 'Rangueil' WHERE iso3 = 'UT3';</pre> <p>Représentez la table avec les verrous apposés par les deux transactions concurrentes Connexion #1 et Connexion #2.</p> <p>7) Modifiez le pays de la transaction concurrente Connexion #2.</p> <pre>UPDATE Country SET capital = 'Rodez'</pre>	

WHERE iso3 = 'UT1';

Expliquez pourquoi la transaction est placée en attente.

8) Modifiez le pays de la transaction concurrente Connexion #1.

UPDATE Country
SET capital = 'Rangueil'
WHERE iso3 = 'UT3';

Expliquez ce qui se passe.

9) Validez la transaction (**COMMIT**).

10) Validez la transaction (**COMMIT**).

5 Compte rendu

Vous devrez remettre un compte rendu qui sera noté. Le compte rendu devra être constitué suivant un plan comme suit.

- Page de garde
- Sommaire
- Introduction (objectif/thème du TP)
- Scripts, résultats et commentaires éventuels
- Conclusion (notions importantes à retenir)

Le format du fichier devra strictement être du PDF.

Tout autre format sera refusé, et entraînera la non prise en compte de ce dernier.
