Polytech'Lille 2ème année - T.P. SGBD

Gestion des transactions

1 Les transactions sous Postgres Version 7.1.1

On peut paramétrer la gestion de la concurrence au niveau système et au niveau des transactions. Par défaut, les transactions au niveau système sont non sériables (pour des raisons de performances). Il est donc nécessaire de verrouiller explicitement les données dans certains cas.

- Certaines commandes SQL provoquent un verrouillage implicite. Ce TP vous permettra d'en voir quelques exemples.
- Si le comportement par défaut ne convient pas, l'utilisateur peut explicitement poser des verrous. Ce verrouillage dure le temps de la transaction : il prend fin au premier commit ou rollback (explicite ou implicite). C'est la commande `LOCK TABLE`. Cette instruction doit être la première de la transaction. Il existe plusieurs variantes de la commande `LOCK TABLE` :

```
LOCK [ TABLE ] name
LOCK [ TABLE ] name IN [ ROW | ACCESS ] { SHARE | EXCLUSIVE } MODE
LOCK [ TABLE ] name IN SHARE ROW EXCLUSIVE MODE
```

2 Expérimentations

Il est nécessaire de travailler par deux connectés simultanément sur deux machines voisines. Les manipulations qui suivent utiliseront une table T ayant 2 colonnes numériques A et B. Vous devez donc commencer par créer cette table (1 par binôme) et la rendre accessible aux deux expérimentateurs (instruction `GRANT`). Insérez les lignes `(0,0)`, `(2,3)` et `(0,1)`.
Par la suite, j'utiliserai les abbréviations :

- `select T` pour `select * from T ;`
- `select T(A = 1)` pour `select * from T where A=1 ;`
- `insert (3,6)` pour `insert into T values (3,6);`
- `update(A <- A+1)` pour `update T set A = A+1 ;`
- `update(A <- 3)|(A = 4)` pour `update T set A=3 where A=4 ;`
- `delete(A = 4)` pour `delete from T where A=4 ;`

2.1 Transactions standards

Expérience 1

```
      session 1                 session 2
-----------------------------------------------------------
   begin transaction         begin transaction
   select T
                             update(A <- A+2)
   select T
                             commit
   select T
   commit
-----------------------------------------------------------
```

Session 1 : quel problème est mis en évidence ?

Expérience 2

Sur les deux machines, nous allons appliquez le mode de transaction suivant :

```
set transaction isolation level serializable
```

Ce mode sera valide dans la transaction ou il est défini, le mode de transaction par défaut peut être lancé par la commande suivante :

```
set transaction isolation level read committed
```

```
      session 1                 session 2
-----------------------------------------------------------
   begin transaction         begin transaction
   set transaction           set transaction
     isolation level          isolation level
     serializable             serializable
   select T
                             update(A <- A-1)
   select T
                             commit
   select T
   commit
-----------------------------------------------------------
```

Notez la différence par rapport à la première expérience.

Expérience 3

```
      session 1
----------------------------
    begin transaction
    select T
    update(A <- A+1)
    select T
    rollback
    select T
----------------------------
```

Quel est le but de `rollback` ?

Expérience 4

```
      session 1                 session 2
-----------------------------------------------------------
   begin transaction         begin transaction
   update(A <- A+1)
   select T
                             insert(3,7)
                             select T
                             delete(A=2)
                             delete(A=1)
   commit
                             commit
-----------------------------------------------------------
```

Ces deux transactions sont-elles sériables ?

Expérience 5

```
      session 1                session 2
-------------------------------------------------------
   begin transaction        begin transaction
   update(A<-3)|(A=2)
                            update(B<-2)|(B=3)
                            select T
   select T
                            update(A<-3)|(A=2)
   update(B<-2)|(B=3)
   commit                   commit
-------------------------------------------------------
```

Quels sont les verrous posés par `session 1` et `session 2` avant le deuxième `update` de `session 2` ?

Expérience 6

Avec Postgres (également implémenté sous Oracle), il existe une clause `for update` à l'instruction select. L'expérience suivante va vous permettre de comprendre son utilité.

```
      session 1                    session 2
-------------------------------------------------------
   begin transaction           begin transaction
   select T(B=7) for update
                               update(B<-10)|(B=0)
                               select T ;
                               update(A<-5)|(A=3)
   update(B<-6)|(B=7) ;
   commit

                               commit
                               select T
-------------------------------------------------------
```

Quel type de verrouillage est effectué ?

2.2 Verrouillages explicites

Expérience 7

```
      session 1                             session 2
-------------------------------------------------------------
   begin transaction                    begin transaction
   lock table T in share mode ;
                                        select T
                                        insert(5,1)
   commit ;
                                        commit
   begin ;                              begin ;
   lock table T in row share mode ;
   update(B<-B+1) ;                     insert(5,2)
                                        update(B<-B+1) ;
   commit ;
                                        commit ;
-------------------------------------------------------------
```

Comment expliquez-vous ces résultats ?

Expérience 8

```
      session 1                             session 2
-------------------------------------------------------------
   begin transaction                    begin transaction
   lock table T in exclusive mode ;
   select T
                                        select T
                                        select T for update
   commit
                                        commit

-------------------------------------------------------------
```

Comment interprétez-vous ces réponses ?

2.3 Les instructions du DDL

Expérience 9

```
      session 1
-------------------------------------------------------
   begin transaction
   create table t2 ...
   donner les droits ...
   rollback
   begin transaction
   create table t2 ...
   commit
   begin transaction
   drop table t2
   rollback
-------------------------------------------------------
```

Comment se comportent les instructions du DDL ?