

Одесский национальный политехнический университет  
Институт компьютерных систем  
Кафедра информационных систем

Лабораторная работа № 10  
По дисциплине: «Операционные системы»  
**Тема: «Управление процессами-транзакциями в базах данных. Часть 2 »**

Выполнила:  
Студентка группы АИ-205  
Колдунова А. А.  
Проверил:  
Блажко О.А.

Одесса 2020

**Цель работы:** исследовать поведение процессов-транзакций в базах данных и средства управления ими через механизм блокировки с использованием современных систем управления базами данных.

## План работы.

### 1 Теоретические сведения

#### 1.1 Основы управления транзакциями-процессами

#### 1.2 Примеры возникающих проблем при одновременном выполнении транзакций

#### 1.3 Противодействие возникновению проблем квазипараллельного выполнения транзакций

#### 1.4 Борьба с тупиковым блокировкой транзакций

#### 1.5 Основы управления базами данных с использованием языка SQL

#### 1.6 Основы работы с СУБД PostgreSQL

#### 1.7 Работа с транзакциями в СУБД

## Решение:

### Задание 1:

#### T1:

```
koldunova_anastasiya=> SELECT * FROM person
koldunova_anastasiya-> ;
 p_id |      name      |      bd
-----+-----+-----
    2 | Petrov         | 2001-02-05
    1 | Ivanov         | 2002-03-05
    3 | Koldunova      | 2003-04-28
(3 rows)

koldunova_anastasiya=> START TRANSACTION
koldunova_anastasiya-> ;
START TRANSACTION
koldunova_anastasiya=> SELECT txid_current();
 txid_current
-----
        3693
(1 row)

koldunova_anastasiya=> INSERT INTO person VALUES(4, 'Smith', '2000/07/11');
INSERT 0 1
koldunova_anastasiya=> SELECT xmin,xmax,p_id,name,bd FROM person;
 xmin | xmax | p_id |      name      |      bd
-----+-----+-----+-----+-----
 2622 |    0 |    2 | Petrov         | 2001-02-05
 2624 | 2626 |    1 | Ivanov         | 2002-03-05
 3683 | 3690 |    3 | Koldunova      | 2003-04-28
 3693 |    0 |    4 | Smith          | 2000-07-11
(4 rows)

koldunova_anastasiya=> commit;
COMMIT
```

#### T2:

```
[koldunova_anastasiya=> START TRANSACTION;
START TRANSACTION
[koldunova_anastasiya=> SELECT xmin,xmax,p_id,name,bd FROM person;
 xmin | xmax | p_id |      name      |      bd
-----+-----+-----+-----+-----
 2622 |    0 |    2 | Petrov         | 2001-02-05
 2624 | 2626 |    1 | Ivanov         | 2002-03-05
 3683 | 3690 |    3 | Koldunova      | 2003-04-28
(3 rows)

[koldunova_anastasiya=> SELECT xmin,xmax,p_id,name,bd FROM person;
 xmin | xmax | p_id |      name      |      bd
-----+-----+-----+-----+-----
 2622 |    0 |    2 | Petrov         | 2001-02-05
 2624 | 2626 |    1 | Ivanov         | 2002-03-05
 3683 | 3690 |    3 | Koldunova      | 2003-04-28
 3693 |    0 |    4 | Smith          | 2000-07-11
(4 rows)
```

**Вывод:** T2 не видит изменения T1, которые исполнила команда insert. После завершения T1, T2 видит все изменения.

**T3:**

```
[koldunova_anastasiya=> commit;
COMMIT
koldunova_anastasiya=> START TRANSACTION
;
START TRANSACTION
[koldunova_anastasiya=> DELETE FROM person WHERE p_id=3;
DELETE 1
[koldunova_anastasiya=> rollback;
ROLLBACK
```

**T2:**

```
koldunova_anastasiya=> SELECT xmin,xmax,p_id,name,bd FROM person;
xmin | xmax | p_id | name | bd
-----+-----+-----+-----+-----
2622 | 0 | 2 | Petrov | 2001-02-05
2624 | 2626 | 1 | Ivanov | 2002-03-05
3683 | 3690 | 3 | Koldunova | 2003-04-28
3693 | 0 | 4 | Smith | 2000-07-11
(4 rows)
```

```
koldunova_anastasiya=> SELECT xmin,xmax,p_id,name,bd FROM person;
xmin | xmax | p_id | name | bd
-----+-----+-----+-----+-----
2622 | 0 | 2 | Petrov | 2001-02-05
2624 | 2626 | 1 | Ivanov | 2002-03-05
3683 | 3695 | 3 | Koldunova | 2003-04-28
3693 | 0 | 4 | Smith | 2000-07-11
(4 rows)
```

**Вывод:** После удаления ряда с p\_id = 3 в T1, о отменения операции, в T2 показывается что с рядом где p\_id = 3 взаимодействовали, с помощью изменения xmax = 3690 на xmax = 3695

**T4:**

```
koldunova_anastasiya=> START TRANSACTION
;
START TRANSACTION
[koldunova_anastasiya=> UPDATE person SET name='Kalina' WHERE p_id=3;
UPDATE 1
[koldunova_anastasiya=> commit;
COMMIT
koldunova_anastasiya=> []
```

**T2:**

```
[koldunova_anastasiya=> SELECT xmin,xmax,p_id,name,bd FROM person;
xmin | xmax | p_id | name | bd
-----+-----+-----+-----+-----
2622 | 0 | 2 | Petrov | 2001-02-05
2624 | 2626 | 1 | Ivanov | 2002-03-05
3683 | 3695 | 3 | Koldunova | 2003-04-28
3693 | 0 | 4 | Smith | 2000-07-11
(4 rows)

[koldunova_anastasiya=> SELECT xmin,xmax,p_id,name,bd FROM person;
xmin | xmax | p_id | name | bd
-----+-----+-----+-----+-----
2622 | 0 | 2 | Petrov | 2001-02-05
2624 | 2626 | 1 | Ivanov | 2002-03-05
3683 | 3696 | 3 | Koldunova | 2003-04-28
3693 | 0 | 4 | Smith | 2000-07-11
(4 rows)

[koldunova_anastasiya=> SELECT xmin,xmax,p_id,name,bd FROM person;
xmin | xmax | p_id | name | bd
-----+-----+-----+-----+-----
2622 | 0 | 2 | Petrov | 2001-02-05
2624 | 2626 | 1 | Ivanov | 2002-03-05
3693 | 0 | 4 | Smith | 2000-07-11
3696 | 0 | 3 | Kalina | 2003-04-28
(4 rows)
```

**Вывод:** В ходе выполнения операция: xmax поменял значение с 3695 на 3696 в рядку 3, а это означает, что выполнялись действия. После фиксации xmin = 3696, xmax = 0, это обозначает что текущий результат был создан транзакцией 3693, и нет новых версий созданными другими транзакциями

## Задание 2

IS:

```
[koldunova_anastasiya=> START TRANSACTION
[koldunova_anastasiya-> ;
START TRANSACTION
[koldunova_anastasiya=> lock table in row share mode;
ERROR:  syntax error at or near "in"
LINE 1: lock table in row share mode;
                ^
[koldunova_anastasiya=> commit;
ROLLBACK
[koldunova_anastasiya=> START TRANSACTION
;
START TRANSACTION
[koldunova_anastasiya=> lock table person in row share mode;
LOCK TABLE
[koldunova_anastasiya=> commit;
COMMIT
```

IX:

```
16885 | relation | 32/8402 | 8788 | ShareRowExclusiveLock | t
[koldunova_anastasiya=> select relation,locktype,virtualtransaction,pid,mode,granted from pg_lo
cks where locktype='relation';
 relation | locktype | virtualtransaction | pid | mode | gran
ted
-----+-----+-----+-----+-----+-----
16816 | relation | 13/8864 | 17111 | RowShareLock | t
16816 | relation | 8/48551 | 13391 | AccessShareLock | t
11673 | relation | 9/41611 | 16380 | AccessShareLock | t
16816 | relation | 9/41611 | 16380 | RowExclusiveLock | t
11673 | relation | 2/1598315 | 6131 | AccessShareLock | t
16885 | relation | 25/5917 | 2488 | AccessShareLock | t
16885 | relation | 7/38110 | 11442 | RowExclusiveLock | f
16885 | relation | 6/110455 | 11300 | ExclusiveLock | f
16885 | relation | 2/1598315 | 6131 | ExclusiveLock | t
16885 | relation | 3/107101 | 7518 | ShareRowExclusiveLock | f
16885 | relation | 31/3451 | 8730 | ShareRowExclusiveLock | f
16885 | relation | 4/213776 | 7715 | ShareRowExclusiveLock | f
16885 | relation | 33/3041 | 9375 | RowExclusiveLock | f
16885 | relation | 29/2299 | 8393 | ExclusiveLock | f
16885 | relation | 28/5089 | 8055 | ShareRowExclusiveLock | f
(15 rows)
```

**Вывод:** IX и IS совместны

## SIX:

```
koldunova_anastasiya=> START TRANSACTION;
START TRANSACTION
koldunova_anastasiya=> lock table person in share row exclusive mode;
LOCK TABLE
koldunova_anastasiya=> select relation,locktype,virtualtransaction,pid,mode,granted from pg_lo
cks where locktype='relation';
```

relation	locktype	virtualtransaction	pid	mode	granted
11673	relation	9/41612	16380	AccessShareLock	t
11673	relation	2/1598315	6131	AccessShareLock	t
16858	relation	11/13266	16860	AccessShareLock	t
16858	relation	11/13266	16860	RowExclusiveLock	t
3455	relation	11/13266	16860	AccessShareLock	t
2663	relation	11/13266	16860	AccessShareLock	t
2662	relation	11/13266	16860	AccessShareLock	t
2685	relation	11/13266	16860	AccessShareLock	t
2684	relation	11/13266	16860	AccessShareLock	t
2615	relation	11/13266	16860	AccessShareLock	t
1259	relation	11/13266	16860	AccessShareLock	t
11673	relation	14/9035	18404	AccessShareLock	t
16816	relation	13/8865	17111	RowExclusiveLock	f
16885	relation	25/5917	2488	AccessShareLock	t
16885	relation	7/38110	11442	RowExclusiveLock	f
16885	relation	6/110455	11300	ExclusiveLock	f
16885	relation	2/1598315	6131	ExclusiveLock	t
16885	relation	3/107101	7518	ShareRowExclusiveLock	f
16885	relation	31/3451	8730	ShareRowExclusiveLock	f
16885	relation	4/213776	7715	ShareRowExclusiveLock	f
16816	relation	8/48551	13391	AccessShareLock	t
16885	relation	33/3041	9375	RowExclusiveLock	f
16885	relation	29/2299	8393	ExclusiveLock	f
16885	relation	28/5089	8055	ShareRowExclusiveLock	f
16816	relation	9/41612	16380	ShareRowExclusiveLock	t

(25 rows)

## IX:

```
koldunova_anastasiya=> START TRANSACTION
;
START TRANSACTION
koldunova_anastasiya=> lock table person in row exclusive mode;
commit;
LOCK TABLE
```

**Вывод:** SIX и IX не совместимы

## SIX:

```
koldunova_anastasiya=> START TRANSACTION;
START TRANSACTION
koldunova_anastasiya=> lock table person in share row exclusive mode;
LOCK TABLE
koldunova_anastasiya=> select relation,locktype,virtualtransaction,pid,mode,granted from pg_lo
cks where locktype='relation';
```

relation	locktype	virtualtransaction	pid	mode	granted
11673	relation	9/41613	16380	AccessShareLock	t
11673	relation	2/1598315	6131	AccessShareLock	t
16858	relation	11/13266	16860	AccessShareLock	t
16858	relation	11/13266	16860	RowExclusiveLock	t
3455	relation	11/13266	16860	AccessShareLock	t
2663	relation	11/13266	16860	AccessShareLock	t
2662	relation	11/13266	16860	AccessShareLock	t
2685	relation	11/13266	16860	AccessShareLock	t
2684	relation	11/13266	16860	AccessShareLock	t
2615	relation	11/13266	16860	AccessShareLock	t
1259	relation	11/13266	16860	AccessShareLock	t
11673	relation	14/9035	18404	AccessShareLock	t
16816	relation	13/8868	17111	RowShareLock	t
16885	relation	25/5917	2488	AccessShareLock	t
16885	relation	7/38110	11442	RowExclusiveLock	f
16885	relation	6/110455	11300	ExclusiveLock	f
16885	relation	2/1598315	6131	ExclusiveLock	t
16885	relation	3/107101	7518	ShareRowExclusiveLock	f
16885	relation	31/3451	8730	ShareRowExclusiveLock	f
16885	relation	4/213776	7715	ShareRowExclusiveLock	f
16816	relation	8/48551	13391	AccessShareLock	t
16885	relation	33/3041	9375	RowExclusiveLock	f
16885	relation	29/2299	8393	ExclusiveLock	f
16885	relation	28/5089	8055	ShareRowExclusiveLock	f
16816	relation	9/41613	16380	ShareRowExclusiveLock	t

(25 rows)

```
koldunova_anastasiya=> commit;
COMMIT
```

## IS:

```
koldunova_anastasiya=> START TRANSACTION
;
START TRANSACTION
koldunova_anastasiya=> lock table person in row share mode;
LOCK TABLE
koldunova_anastasiya=> commit;
COMMIT
```

**Вывод:** SIX и IS совместны

### Задание 3:

#### T1:

```
koldunova_anastasiya=> START TRANSACTION
;
START TRANSACTION
koldunova_anastasiya=> set transaction isolation level read committed;
SET
koldunova_anastasiya=> SELECT * FROM person WHERE p_id=1;
  p_id |          name          |          bd
-----+-----+-----
      1 | Ivanov                 | 2002-03-05
(1 row)

koldunova_anastasiya=> UPDATE person SET bd='2000/01/01' WHERE p_id=1;
UPDATE 1
koldunova_anastasiya=> SELECT * FROM person WHERE p_id=1;
  p_id |          name          |          bd
-----+-----+-----
      1 | Ivanov                 | 2000-01-01
(1 row)

koldunova_anastasiya=> commit;
COMMIT
```

#### T2:

```
koldunova_anastasiya=> START TRANSACTION;
START TRANSACTION
koldunova_anastasiya=> set transaction isolation level read committed;
SET
koldunova_anastasiya=> SELECT * FROM person WHERE p_id=1;
  p_id |          name          |          bd
-----+-----+-----
      1 | Ivanov                 | 2002-03-05
(1 row)

koldunova_anastasiya=> SELECT * FROM person WHERE p_id=1;
  p_id |          name          |          bd
-----+-----+-----
      1 | Ivanov                 | 2002-03-05
(1 row)

koldunova_anastasiya=> UPDATE person SET name='Aliev' WHERE p_id=1;
UPDATE 1
koldunova_anastasiya=> SELECT * FROM person WHERE p_id=1;
  p_id |          name          |          bd
-----+-----+-----
      1 | Aliev                  | 2000-01-01
(1 row)

koldunova_anastasiya=> commit;
COMMIT
```

**Вывод:** При выполнении операции на уровне изоляции READ COMMITTED, операция происходит успешно. Операция update в T2 переходит в режим ожидания и после завершения T1 успешно изменяет данные

## T1:

```
[koldunova_anastasiya=> set transaction isolation level repeatable read;
SET
[koldunova_anastasiya=> SELECT * FROM person WHERE p_id=1;
 p_id |          name          |          bd
-----+-----+-----
      1 | Aliev                  | 2000-01-01
(1 row)

[koldunova_anastasiya=> UPDATE person SET bd='2003/07/12' WHERE p_id=1;
UPDATE 1
[koldunova_anastasiya=> SELECT * FROM person WHERE p_id=1;
 p_id |          name          |          bd
-----+-----+-----
      1 | Aliev                  | 2003-07-12
(1 row)

[koldunova_anastasiya=> commit;
COMMIT
```

## T2

```
[koldunova_anastasiya=> START TRANSACTION;
START TRANSACTION
[koldunova_anastasiya=> set transaction isolation level repeatable read;
SET
[koldunova_anastasiya=> SELECT * FROM person WHERE p_id=1;
 p_id |          name          |          bd
-----+-----+-----
      1 | Aliev                  | 2000-01-01
(1 row)

[koldunova_anastasiya=> UPDATE person SET name='Block' WHERE p_id=1;
ERROR:  could not serialize access due to concurrent update
```

**Вывод:** При выполнении операции на уровне изоляции REPEATABLE READ, операция происходит не успешно. Операция update в T2 переходит в режим ожидания и после завершения T1 уведомляет об ошибке.

## T1:

```
koldunova_anastasiya=> START TRANSACTION
;
START TRANSACTION
[koldunova_anastasiya=> set transaction isolation level serializable;
SET
[koldunova_anastasiya=> SELECT * FROM person WHERE p_id=1;
p_id |          name          |          bd
-----+-----+-----
    1 | Aliev                  | 2003-07-12
(1 row)

[koldunova_anastasiya=> UPDATE person SET bd='2003/01/01' WHERE p_id=1;
UPDATE 1
[koldunova_anastasiya=> SELECT * FROM person WHERE p_id=1;
p_id |          name          |          bd
-----+-----+-----
    1 | Aliev                  | 2003-01-01
(1 row)

[koldunova_anastasiya=> commit;
COMMIT
koldunova_anastasiya=> █
```

## T2:

```
koldunova_anastasiya=> START TRANSACTION;
START TRANSACTION
[koldunova_anastasiya=> set transaction isolation level serializable;
SET
[koldunova_anastasiya=> SELECT * FROM person WHERE p_id=1;
p_id |          name          |          bd
-----+-----+-----
    1 | Aliev                  | 2003-07-12
(1 row)

[koldunova_anastasiya=> UPDATE person SET name='Block' WHERE p_id=1;
ERROR:  could not serialize access due to concurrent update
koldunova_anastasiya=> █
```

**Вывод:** При выполнении операции на уровне изоляции SERIALIZABLE, операция происходит не успешно. Операция update в T2 переходит в режим ожидания и после завершения T1 уведомляет об ошибке.

## Задание 4 :

```
[koldunova_anastasiya=> START TRANSACTION;
START TRANSACTION
[koldunova_anastasiya=> UPDATE person SET name='Petrov' WHERE p_id=2;
UPDATE 1
[koldunova_anastasiya=> UPDATE person SET name='Sidorov' WHERE p_id=1;
UPDATE 1
koldunova_anastasiya=> █
```

```
[koldunova_anastasiya=> START TRANSACTION;
START TRANSACTION
[koldunova_anastasiya=> UPDATE person SET name='Lviv' WHERE p_id=1;
UPDATE 1
[koldunova_anastasiya=> UPDATE person SET name='Koldunova' WHERE p_id=2;
ERROR:  deadlock detected
DETAIL:  Process 4291 waits for ShareLock on transaction 3866; blocked by process 4280.
Process 4280 waits for ShareLock on transaction 3867; blocked by process 4291.
HINT:  See server log for query details.
CONTEXT:  while updating tuple (0,21) in relation "person"
koldunova_anastasiya=> █
```



При выполнении операции update в T2 она получила сообщение об ошибке, так как ожидала завершения T1, а T1 ожидала завершения T2 — это привело к тупика, поэтому необходимо было принудительно завершить транзакцию, которая привела к нему, т.е. T2.

**Вывод:** В данной лабораторной научились управлять процессами-транзакциями в базах данных.