

Lab IV

Transaction Management

Database Laboratory

Objectives

- To work with transactions in ORACLE
- To study the properties of transactions in ORACLE

Transactions

- Database integrity must be controlled when access operations take place, generally coming from the applications.
- The access operations to a database are organised in transactions.

TRANSACTION 

A sequence of access operations to the database that constitutes a logical execution unit.

Transactions

(ACID) Properties which all transaction must comply with:

- **atomicity**: a transaction is an atomic execution unit (either all the operations are performed or none)
- **consistence**: the transaction must yield a consistent state for the database (all the integrity constraints must be met)
- **isolation**: the modifications introduced by a non-confirmed transaction are not visible by other transactions
- **durability**: the confirmation involves the recording of the changes performed on the database, so that they cannot be lost due to a system failure or other transactions

Transactions

Operations:

- **Start:** there is no specific instruction.
- **End:** the transaction is partially confirmed through
COMMIT [WORK]
- **Cancellation**
ROLLBACK [WORK]

Transactions

Strategies to check consistency:

- **Immediate mode:** It checks the constraint after each execution of a relevant update operation (IMMEDIATE)
- **Deferred mode:** after the end of a transaction that includes an SQL instruction which is relevant for the constraint (DEFERRED)

Constraint:

- NOT DEFERRABLE: it is always in immediate mode
- DEFERRABLE: it can be either immediate or deferred

Transactions

Semantics for each option:

- If this clause is not used, i.e. it is NOT DEFERRABLE (and hence immediate)
- DEFERRABLE INITIALLY IMMEDIATE
- DEFERRABLE INITIALLY DEFERRED
- NOT DEFERRABLE INITIALLY DEFERRED -> Forbidden
- DEFERRABLE: initially immediate
- NOT DEFERRABLE: it must be immediate.
- INITIALLY IMMEDIATE: not deferred
- INITIALLY DEFERRED: logically it is deferrable

Transactions

The mode of a constraint can be changed through:

```
SET CONSTRAINT {constraint_name | ALL}  
               {IMMEDIATE | DEFERRED}
```

- Every constraint which is specified in the list must be deferrable and the option ALL refers to all the deferrable constraints in the database schema.
- The scope of the change of the SET CONSTRAINT instruction is the transaction in which it appears or until another SET CONSTRAINT instruction changes the constraint mode.
- If the instruction is executed in the middle of a transaction with the IMMEDIATE, the constraints included in the instruction are checked; if one fails, the instruction SET fails and the mode of all constraints remains unchanged.

Exercises

- a) In ORACLE, there is no "update in cascade" directive for the restoration of the referential integrity.

How can you modify the code of a user that had loans, without violating the referential integrity of the loan table?

(need of the concept of transaction).

Exercises

b) Design a transaction which performs tuple insertions over the table *User2* according the following transaction schema:

```
INSERT INTO user2 VALUES (s1, ..., ..., 0);  
INSERT INTO user2 VALUES (s2, ..., ..., 0);  
INSERT INTO user2 VALUES (s1, ..., ..., 0);  
INSERT INTO user2VALUES (s3, ..., ..., 0);
```

Execute two instances of the previous transaction schema, one with a primary key constraint for the table user in IMMEDIATE mode and the other with DEFERRED MODE.

Which differences do you see? (atomicity and consistency).

Exercises

c) Start in your PC two different sessions over the databases. In each session perform the following transactions (t_i indicates the order in which the operations must be performed):

Session 1

t0 “query the total number of users”

t2 “insert a new user”

t3 “query the total number of users”

t5 “end the transaction”

t7 “insert a new user”

t8 “query the total number of users”

t10 “query the total number of users”

Session 2

t1 “query the total number of users”

t4 “query the total number of users”

t6 “query the total number of users”

t9 “cancel the transaction”

t11 “query the total number of users”

Once both transactions have been finished, query the total number of users in both sessions. How do you interpret the results? (isolation and persistence).

According to this, can you affirm that ORACLE maintains the persistence property?