

EAPLI Formative assessment

ORM, JPA

Concurrent Data Access

Correct answers are highlighted in bold font.

1 – Who has the responsibility to update the @Version attribute in the database?

- a) The database management system (DBMS) since the attribute @Version respects to the record version in the database which might be different from the object state in memory.
- b) JPA, through EntityManager, to assure that the locking protocol is not hampered.**
- c) The responsibility is shared between DBMS and JPA since the attribute @Version is declared in an @Entity class and persisted in the database table.

2 – Wikipedia articles are downloaded for review by several experts with interest in the specific field. After a long reviewing process, the reviewed articles are submitted for update. Under such a setting I would recommend Wikipedia to use:

- a) Optimistic locking**
- b) Pessimistic locking

3 – I have to design a multi-user solution for a new domain I do not master. I have no details about the frequency and volume of data updates. Under such circumstances:

- a) I should base my design on Optimistic Locking to start with and review the concurrent data access strategy later on, once I have more details about data updates**
- b) I should not recommend any strategy to manage concurrent data access to avoid erroneous decisions
- c) I should recommend Pessimist Locking to be on the safe side

4 – I am working at the IT department of a bank and reviewing the concurrent data access strategies of the commercial banking application. This application is used in every branch to perform bank operations with clients at the desk (cash deposits and withdrawals, update user

data, create new accounts, etc.). We have approximately 200 daily transactions between 9 am and 3 pm in each branch. The chief of the IT department claims that our database is a critical asset and has to be permanently in a consistent state so we should use Pessimistic Locking.

a) Comment this decision.

The consistency of the data is assured by Optimistic Locking as much as by Pessimist Locking, The fact that the data has to be permanently in a consistent state is not a reason to opt by one or another policy.

b) Make a recommendation regarding the best strategy to use.

This is a case of a transactional processing application with frequent accesses to the database each updating few records. Pessimistic Locking is appropriate.