INFS3200 Advanced Database Systems

Tutorial 3: Distributed TM

Semester 1, 2021

Question 1: Consider the following database designed for an Olympic Game, where an athlete (identified by `CompID`) participates an event (identified by `EventID`).

```
Results(EventID, CompID, Position)
Medals(EventID, Medal, CompID) //Medal: Gold/Silver/Bronze
MedalTally(Country, Medal, Number)
Competitors(CompID, Country)
Medalists(CompID, NMedals) //NMedals: number of medals won
```

Assume that when an event is completed, a file is created at the venue giving `CompID` and `Position` associated with that `EventID`, then a series of transactions are executed to update other tables.

(a) Write a program as a single transaction to perform the updates.
(b) For each table, discuss whether or not any interference with another transaction is possible. Use an example to illustrate.
(c) Revise your program with read-lock, write-lock and unlock statements. The notation should make clear the granularity level of locking. In each case, indicate whether the granularity of locking is more than strictly necessary. Justify your decision based on the characteristics of the application.
(d) Show that your lock/unlock annotations constitute two-phase locking (2PL).
(e) Is there any circumstance in which the program would have to abort the transaction?

Question 2: Each table in the above question can be stored at a different site, and some tables, such as **Medals, Medalists** and **MedalTally**, may need to be replicated at several sites. Clearly, the transaction in the previous question will comprise of a set of sub-transactions executing at different sites to update data there. The transaction can only finish when all its sub-transactions successfully finish. This process is enforced by using the two-phase commit (2PC) protocol.

(a) Suppose there are a large number of replicas for **MedalTally**. Should we adopt the synchronous replication strategy for this table?
(b) Would you recommend using asynchronous replication for this application?
(c) Suppose one of the **MedalTally** replica sites fails during the transaction. Show the exchange of messages among sub-transactions resulting in the two-phase commit protocol to issue an abort instruction.

Question 3: When 2PC is used as the commit protocol, explain how the system recovers from failure and deals with particular transaction *T* in each of the following cases:

(a) A subordinate site for *T* fails before receiving a *prepare* message.
(b) A subordinate site for *T* fails after receiving a *prepare* message but before making a decision.
(c) The coordinator site for *T* fails before sending a *prepare* message.
(d) The coordinator site for *T* fails after writing an *abort* log record but before sending any further messages to its subordinates.

Question 4: Distributed transaction management can use primary-copy-based approach or voting-based approach to maintain data consistency among multiple copies. Please explain how these two approaches work and compare their advantages and disadvantages.