

Sean Overton

SN: 6421490

Solution 3:

Explanation as to why solution3.sql can cause database corruption at **read-committed isolation level**.

T1	T2
EXECUTE UPDATE_AVERAGE_GOALS(); //ie. The statements below	
SELECT SUM(HTSCORE) INTO totalHomeGoals FROM GAME t WHERE t.HTNAME = teamRow.NAME;	
	DELETE * FROM GAME WHERE HTSCORE > 1;
	COMMIT;
SELECT COUNT(*) INTO totalHomeGames FROM GAME t WHERE t.HTNAME = teamRow.NAME;	
UPDATE TEAM SET AVERAGE_GOALS_AT_HOME = totalHomeGoals / totalHomeGames WHERE TEAM.NAME = teamRow.NAME;	

-this procedure could definitely result in a corrupt database upon execution when interleaved with other transactions as observed above in an example.

-if any modifications are **committed** between the two select statements (as the isolation level is read committed) of the procedure, modifying the GAME relational table's 'HTSCORE' or the count of 'HTNAME' (eg. by deleting rows like in the example above) then the calculated AVERAGE_GOALS_AT_HOME values would be incorrect, corrupting the database.

-An example with actual numbers can be observed below:

T1	T2
EXECUTE UPDATE_AVERAGE_GOALS(); //ie. The statements below	
SELECT SUM(HTSCORE) INTO totalHomeGoals FROM GAME t WHERE t.HTNAME = teamRow.NAME; eg. 10 goals for that team (from the 7 home games they played in GAME)	
	DELETE * FROM GAME WHERE HTSCORE > 1;
	COMMIT; -now there is only 2 home games in

	GAME for that same team
SELECT COUNT(*) INTO totalHomeGames FROM GAME t WHERE t.HTNAME = teamRow.NAME; -this would now return only <u>2</u> home games in GAME for that same team	
UPDATE TEAM SET AVERAGE_GOALS_AT_HOME = totalHomeGoals / totalHomeGames WHERE TEAM.NAME = teamRow.NAME; -this calculation is now incorrect. le. $7/2 = \text{average of } 3.5$ -when for example if the two (2) games remaining have 1 goal in each, resulting in a total of 2 goals in 2 games; the average should be then $2/2 = 1$. Not the 3.5 that would actually be stored in the table.	