Introduction to Database Systems

Problem Set 3

Due: March 10, 2015 at 11:59 PM

1. Describe an example application for each of the databases below. Show how the application's workload will take advantage of each's design.

   (a) A main memory database.

   (b) DynamoDB

2. Analyze the following transaction schedules. Draw a precedence graph for each, labeling the edges with the two steps that created it. Determine whether each schedule is view serializable, conflict serializable, or both and state why.

   (a) Schedule:

Transaction 1	Transaction 2
1. R1	
2. R2	
3. W1	
	4. R2
	5. R1
	6. W3
	commit
7. W2	
commit	

   (b) Schedule:

Transaction 1	Transaction 2
1. W2	
2. R1	
3. R3	
	4. W2
	5. W3
	commit
6. W1	
commit	

   (c) Schedule:

Transaction 1	Transaction 2
1. R1	
2. R2	
	3. W2
4. W1	
commit	
	5. W1
	commit

3. Say that we use two-phase locking for the following query. Show any locks that are acquired or released at each step. Annotate the lock point. Label each lock as 'S', 'X', or 'U' for shared, exclusive or upgrade respectively.

Transaction 1	Transaction 2
1. R3	
2. R2	
3. W2	
	4. R2
	5. W2
	6. R1
	7. W4
8. R3	
commit	
	9. W1
	commit

4. For each of the following sets of transactions, state whether they are vulnerable to the phantom problem. Briefly explain your answer.

   (a) Queries:

Transaction 1	Transaction 2

`avg_rate = SELECT avg(rate)`
`FROM health_insurance`
`WHERE family_size = 2;`	

	`UPDATE health_insurance`
	`SET rate=rate * 1.03`
	`WHERE state='IL';`

`SELECT name, rate`
`FROM health_insurance`
`WHERE family_size=2 and rate > avg_rate;`	

   (b) Queries:

Transaction 1	Transaction 2

`SELECT name`
`FROM schools`
`WHERE name like 'S%';`	

	`UPDATE schools`
	`FROM SET projected_enrollment=enrollment * 1.1`
	`WHERE district='Evanston';`

`SELECT count(*)`
`FROM schools`
`WHERE name like 'S%';`	

   (c) Queries:

Transaction 1	Transaction 2

`SELECT name`
`FROM residents`
`WHERE NOW() - year_of_birth > 18;`	

	`INSERT INTO residents` `VALUES ('John', 1979);`

`SELECT count(*)`
`FROM residents`
`WHERE NOW() - year_of_birth > 18;`	

5. We are using optimistic concurrency control with serial validation to manage two interleaving transactions, T_1 and T_2. Each has been assigned their final transaction id. For the following read and write sets, determine whether T_2 will commit, abort, or the result is uncertain. Why is this outcome expected?

   (a) T_1: Read(A), Write (A, B)
   T_2: Read(B), Write(A)

   (b) T_1: Read(A,B), Write (A, C)
   T_2: Read(B), Write(A)

   (c) T_1: Read(A), Write (B)
   T_2: Read(A), Write(C)

6. Suppose that we are partitioning a Dynamo database over 8 nodes, and that we have no virtual nodes. We have a hash function of $v\, mod\, 256$ and nodes $A, B, ..., H$, where node A covers the hash values ranging from 0...31, B's partition has the next 32 slots and so on.

   (a) If we have a sloppy quorum where $N = 5, R = 1, W = 3$, what nodes would be valid write targets for a key of 755?

   (b) To what nodes may we write if our quorum configuration is $N = 4, R = 1, W = 2$?

   (c) Continuing with the $N = 4$ configuration, describe the sets of valid participants in this write. Use at least one real example.

   (d) If the first replica in this $N = 4$ set becomes unavailable, what sets nodes may accept this write? Use at least one valid set of hosts in your description.