

Viet Tran
Database 1
Prof. Cindy Chen
Dec 4th, 2017

212

Homework 8

25.1 What are the two types of time, and how are they different? Why does it make sense to have both types of time associated with a tuple?

There are two types of time in database systems: time as measured by the system and time as observed in the real world.

- Transaction time is the time interval during the fact is current within the database system.
- Valid time is the set of time intervals during which the fact is true in the real world.

It is necessary to have both so that data can be understood by human using the valid time system and easier data crunching programmatically using calculation based on transaction time.

25.2 Suppose you have a relation containing the x, y coordinate and names of restaurants. Suppose also that the only queries that will be asked are of the following form: The query specifies a point, and asks if there is a restaurant exactly at that point. Which type of index would be preferable, R-tree or B-tree? Why?

Since there are only two dimensions, x and y , R-trees are not preferable since it is useful for indexing set of rectangles and other polygons.

Since k -d B tree partitions space into two, it is very convenient for this scenario. One partition store the restaurant's location coordinates and the other stores the restaurant's names. B-tree will only need 2 dimensions for this specific case.

25.3 Suppose you have a spatial database that supports region queries (with circular regions) but not nearest-neighbor queries. Describe an algorithm to find the nearest neighbor by making use of multiple region queries.

Without nearest-neighbor queries, the algorithm would have to keep querying while increasing the searching area until a result is returned.

- Pick a point P in database of points.
- Sweep the area in a circular manner with P as center, increasing the radius with each sweep.
- Once a sweep contains points within in is found, calculate the shortest distance between P and these points so as to find the nearest neighbor.

25.9 Will functional dependencies be preserved if a relation is converted to a temporal relation by adding a time attribute? How is the problem handled in a temporal database?

No, functional dependencies will not be preserved if a relation is converted to a temporal relation by adding a time attribute. This is because when adding a time attribute to the relation, the dependency would be violated (e.g. in a store database, there exists a

functional dependency transaction number ~~by~~ amount, adding time to this relation would make the transaction number mismatched with the amount)

This problem is handled by temporal database with a different functional dependency called **temporal functional dependency**, a snapshot is created with each *time* instance, ensuring that all functional dependencies must be preserved at all time.