Databases 2000

P5q8
JKMM
P12q9

Question 1 Solution cont'd) within a particular view of the enterprise model, which is maintained by the DBA. Provided that evolution of the enterprise does not affect that view there should be no need to modify application code.

Note that relational DMLs do not specify the access path. Logical access paths — indexes, etc. — may be modified by the DBA. Query optimizers can alter the tactics of evaluation in response to such changes.

A *relational database schema* consists of a collection of attributes that describe members of entity sets associated with the enterprise being modelled. A set of attributes X *functionally determines* an attribute B if, whenever

Question 1 Solution (ctd) a set of values is known for each attribute $A \in X$, the value of B is determined uniquely.

A set of attributes Y is functionally dependent on a set X if each attribute $B \in Y$ is functionally determined by X.

A relation R is ~~Boyce-Codd~~ a set of tuples defined over attribute set Y.

$X \subset Y$ is a *determinant* if X functionally determines some attribute $B \in (Y - X)$.

R is in Boyce-Codd Normal Form if for every determinant X, Y is functionally dependent on X.

Question 1 Solution ctd)

We're not asked to track the vehicles, and we assume that each sensor has a clock and a unique identifier, and that there is a cartographic database which supports journey reconstruction.

It's not made clear whether the funds must be pre-allocated to individual vehicles, but the words AS MODIFIED! suggest NOT — that's what we assume. So we need to account on the basis of car ownership, but manage infringements per vehicle as they occur; this makes good sense for corporate fleets.

Detecting congestion is NOT our concern. Once amber lights show, timed positions of every vehicle within the controlled area must be

Question 1 Solution ctd) sent periodically for entry in the database.

All vehicles have a unique registration no. We assume that each vehicle has a registered owner, and that owners are also uniquely identified. A relation keyed on veh# determines owner#; an index on owner# supports the inversion.

The enforcement is managed through a relation keyed on owner#. This gives name and address information, also a balance of account with an indication of its currency.

Two relations handle accounting of vehicle movement. The first is all key, and contains tuples (veh#, sensor#, time). The second is established by sorting this position data by time within veh#, and calculating differences in order to establish progress. The relation records/steps on chains of

Question 1 Solution ctd) which fines are payable, together with the fine ~~and~~ due. After each ~~step~~ period of infringement by a vehicle has been processed the owner's balance is updated; if funds are insufficient, the owner is informed at once and any subsequent fine is trebled. Accounts are prepared (e.g.) monthly, itemised chronologically within veh #, and the corresponding lines in the progress relation are marked as accounted for.

I think that's OK, though harder than I'd hoped. It's vital that a note be added to the Q. to stress that the detail of the fining process is outside the scope of the question.

Traffic movement analysis is a real-time problem. Everything required for the answer above can be calculated retrospectively, however, and there is no need to account in real time.