Advanced Database Management Systems

Friday 23rd January 2009

Time: 14:00 – 16:00

Please answer any THREE questions from the FIVE questions provided

The use of electronic calculators is permitted provided they are not programmable and do not store text
1. a) This course unit discussed a number of variations on what was referred to as classical DBMSs and motivated some of those variations in terms of the evolving nature of modern business organizations. Briefly cite:

i) One assumption made in classical DBMSs. (1 mark)

ii) One capability that aims to advance DBMS technology towards not having to make the assumption you cited in (Question 1.a.i) above. (1 mark)

iii) One reason why, having the capability cited in (Question 1.a.ii) above, matters to modern business organizations. (1 mark)

b) Let a database keep a record of flights in terms of flight number, where the flight starts from, where it flies to, the date on which it departs and the date on which it arrives. Let it also keep a record of which individual plane is used for which flight on which weekday, as well as which plane type is usable in which flight. Finally, let it keep a record of which pilot is certified to fly which plane type. Using abbreviations in the obvious way, the database has been modelled as the following set of relation schemas (the attributes that make up the primary key are underlined; an attribute in italic font is a foreign key into another relation, as indicated by similarity of attribute name and type):

```
Flights (fltno: str, from: str, to: str, dep: date, arr: date)
UsedFor (planeid: str, fltid: str, weekday: str)
Usable (flid:str, pltype: str)
Certified (pilid:str, planetype: str)
```

i) Write the following query in SQL: “Retrieve the id of pilots who are certified to fly planes of type 767 used in Friday flights.” (2 marks)

ii) Write the direct, clause-by-clause translation of the SQL query in your answer to (Question 1.b.i) above into relational algebra. (2 marks)

iii) Recall the rewrite rules used in heuristics-based logical optimization discussed in the course unit. Cite two rewrite rules that you can apply to the relational algebraic expression that constitutes your answer to (Question 1.b.ii) above and explain the heuristic basis for applying each of them. (5 marks)
iv) Translate into SQL the following relational-algebraic expression over the relation schemas above:

\[ \pi_{\text{planeid,from}}(\sigma_{\text{weekday='Monday'}}(\pi_{\text{weekday,planeid,from}}(\text{Flights} \bowtie \text{UsedFor}))) \] (4 marks)

v) Translate the following semi-join expression into English:

Usable \bowtie \text{UsedFor} (4 marks)

2. a) Assume that a query execution plan contains a join R \bowtie S over intermediate results R and S. Further assume that the DBMS has been configured to use B bytes of buffer space.

i) Briefly explain how large B must be in terms of the cardinalities of R and S for it to make sense for the optimizer to select a nested-loop join as the physical algorithm with which to evaluate R \bowtie S. (2 marks)

ii) Suppose that B is smaller than the value of the expression you gave as an answer to (Question 2.a.i) above and that as a consequence the optimizer has chosen a hash join. Further suppose that the evaluation strategy is pipelined. Briefly explain the consequences of this choice for the activity of the operators that are ascendants of the join (i.e., lie downstream from it in the query execution plan). (2 marks)

iii) Briefly comment on why the optimizer will not contemplate using an indexed join in this case. (2 marks)
b) Assume the same database described in (Question 1.b) above. Now let |R| denote the estimated cardinality of the relation R and V(R, A) the estimated number of distinct values in the domain of attribute A in R. Given the statistics in the following table:

| R            | |R|   | V(R, A)               |
|--------------|------|-----------------------|
| Flights      | 700  | V(Flights, to) = 12   |
|              |      | V(Flights, from) = 8  |
|              |      | V(Flights, fltno) = 700|
| UsedFor      | 800  | V(UsedFor, fltid) = 700|
|              |      | V(UsedFor, weekday) = 6|

draw the operator tree for the following query

\[ \pi_{\text{planeid}} \left( (\sigma_{\text{to} = \text{MAN} \land \text{from} = \text{LHR}}(\text{Flights})) \bowtie (\sigma_{\text{weekday} = \text{Monday}}(\text{UsedFor})) \right) \]

and annotate each node with an estimate of the cardinality of its output.

(10 marks)

c) Assume that, in a database application, the cost of most queries is dominated by the cost of scans. The organization wants to use parallelization and, as a result, is considering which partitioning strategy to use. State one partitioning strategy you would **not recommend** and briefly explain why. (2 marks)

d) Assume now that, instead of what is stated in (Question 2.c) above, most queries involve equality predicates. State one partitioning strategy you would **recommend** and briefly explain why. (2 marks)
3. a) Assume the same database described in (Question 1.b) above. Further assume that range predicates (such as greater than, smaller than, etc.) are valid over string-valued attributes.

i) Let the Flights relation be fragmented as follows. Flights with a from value less than (in lexicographic order) JFK are held in a fragment F1 in site S1, those with a from value greater than (in lexicographic order) LHR are held in a fragment F3 in site S3 and those in between are held in a fragment F2 in site S2. Write the three algebraic expressions that correctly define each of the fragments.

   (3 marks)

ii) Assume the fragmentation of the Flights relation described in (Question 3.a.i) above. Given the following SQL query

   ```sql
   SELECT F.from, F.to
   FROM Flights F
   WHERE F.from='MAN'
   ```

   show how to translate, localize and reduce it and briefly justify each step of the process.

   (10 marks)

iii) Briefly explain one performance gain made possible by the fragmentation in (Question 3.a.i) above by making reference to your answer to (Question 3.a.ii) above.

   (2 marks)

iv) Now, instead of what is stated in (Question 3.a.i) above, let the Flights relation be fragmented in such a way that fragment F4 holds data about the origin and the destination of flights and fragment F5 holds data about the departure and arrival date of flights. Write the two algebraic expressions that correctly define each of the fragments.

   (2 marks)

b) Fragmentation decisions such as described in (Question 3.a.i) above are said to characterize a top-down design for a distributed database. Briefly explain what is meant by that.

   (1 mark)

c) Briefly explain when a bottom-up design for a distributed database is more appropriate.

   (1 mark)

d) In the context of your answers to (Question 3.b) and (Question 3.c) above, briefly explain what is a wrapper and what is a mediator.

   (1 mark)
4. a) A start-up company, called FASTA, is offering a data appliance, called F1, that achieves great performance by using specially-designed RAM modules in the presence of which query plans suffer from an order of magnitude less cache misses than the competition. On the basis of this information, state which type of data appliance is FASTA F1? (2 marks)

b) Assume now that as time passes and data volumes grow, FASTA F1 begins to lose market for not being as fast as the competition, who successfully caught up with them in the meantime. The FASTA executive board is discussing how to react. The CEO says that they should scale up the F1 design, while the CTO says they should go for a scale out design.

i) Briefly suggest what might it mean to scale up the F1 design. (2 marks)

ii) The CTO proposes to call his scaled out design the FASTA F4. Given the name of the new product, briefly suggest what is the relationship between it and the previous FASTA F1 design. [ (2 marks)

iii) Recall that total cost of ownership (TCO) has three main (additive) components: acquisition costs (ACQ), administration costs (ADM) and operational costs (OPE). Assume that a classical DBMS offers the following TCO proposition: ACQ = 10, ADM = 70 and OPE= 20. The CEO proposes to market their data appliance on the following TCO proposition: ACQ = 5, ADM = 70 and OPE= 5. The CTO says this makes little sense. She argues that “After all, FASTA is in the data appliance market.”. Briefly explain whether you agree with the CTO or not. (2 marks)

c) Assume the same database described in (Question 1.b) above. Further assume the fragmentation described in (Question 3.a.i) above. Consider now the following SQL query over the distributed database that results from such fragmentation:

```
SELECT MAX(F.arr)
FROM Flights F
WHERE F.dep > 20081026
GROUP BY F.from
```

Irrespective of other considerations that may bear on the decision, briefly show that this query is amenable to being efficiently evaluated using a map-reduce engine (such as discussed in the course unit) by briefly explaining how the computation could be carried out in terms of a map phase and a reduce phase. (6 marks)

d) State two optimization strategies that can be used in a map-reduce engine and briefly explain the rationale behind them. (6 marks)
5. a) Briefly explain why one cannot enforce a requirement that the output type of a query on pure streams is a set. (1 mark)

b) Briefly explain why query evaluation over pure streams has to contend with primary memory management issues in a different way from classical DBMSs. (1 mark)

c) Assume that a sliding window equijoin is running over two windows X and Y (under bag semantics). Further assume that X is defined to hold the last five tuples to arrive in it and Y the last three. Finally, assume that both tuples are unary with a single attribute of type string and are time-stamped (with integer values). Let the current state of the windows be as follows:

\[
X = [(b, 1), (b, 2), (c, 2), (a, 3), (c, 4)]
\]

\[
Y = [(c, 1), (a, 3), (c, 3)]
\]

Using the core semantics of a sliding window join, describe the steps that are carried out when:

i) (c, 5) arrives in X (when the state is as above). (3 marks)

ii) (b, 6) arrives in Y (when the state is as resulted from the previous item). (3 marks)

c) Briefly explain why the issues and challenges in your answer to (Question 5.b) above do not normally apply to query evaluation over sensor networks. (2 marks)

d) Briefly explain what is meant by the problem of clock drift in the context of sensor network query processing on mote-like devices. (2 marks)

e) Briefly describe the strategy used in TinyDB to contend with the problem of clock drift. (4 marks)

f) Consider the following query in TinyDB:

```
SELECT acceleration, magneticForce
FROM Sensors
WHERE acceleration > 5
AND magneticForce > 10
SAMPLE INTERVAL 1s
```

i) Briefly explain why is there an issue regarding the evaluation order of the predicates in the WHERE clause. (2 marks)

ii) Briefly explain whether selectivity plays a role in the choice of order. (2 marks)

END OF EXAMINATION