THE BCS PROFESSIONAL EXAMINATION
Professional Graduate Diploma

April 2005

EXAMINERS' REPORT

Advanced Database Management Systems

Question 1

1. In early papers on the relational model of data, Dr. E. F. Codd proposed two mechanisms for manipulating data stored in a relational database. These were *relational algebra* and *relational calculus*. Compare and contrast these approaches and discuss their relevance to modern relational database products. **(25 marks)**

Answer Pointers

This question assesses part 1 of the syllabus: Theoretical concepts

Relational algebra is the principal component of the manipulative part of Codd's model. It is essentially a set of operators that take relations as their operands and return a relation as their result. The original operators were eight in number and consisted of four set operators: union, intersection, difference, Cartesian product and four special operators: restrict, project, join and divide. These were not a minimal set, some of the operations could be defined in terms of others. Further operators have subsequently been defined. The only constraint on such operators is that they only ever yield relations. The importance of this closure property is that the result of one operation can always be the input to another.

The calculus provides a notation for stating the definition of a desired relation in terms of a set of given relations. Whereas algebra is prescriptive in defining a relation calculus is descriptive. Algebra is procedural whereas calculus is non-procedural. A system that executes calculus is told about the characteristics of the desired result and must decide how to achieve this result. Actually there is equivalence between calculus and algebra which was set out in a paper by Codd. Every calculus expression has an equivalent algebraic representation. Calculus is based on the mathematical concept of predicate logic.

Both algebra and calculus are central to any relational database. Algebraic operations can be implemented directly by a DBMS. A good DBMS will be able to apply algebraic optimisation rules to improve query retrieval times. Since algebra is relationally complete database implementers can be assured that their product will be capable of dealing with all possible relational queries. Calculus is the basis of SQL i.e. SQL is essentially an English-like calculus language. Since any calculus expression can be converted into algebra a DBMS can operate by mapping SQL expressions to the set of relational algebra operations it supports.

Examiner's Guidance Notes

Most candidates answered this question poorly in terms of content and structure. In some cases candidates merely recalled examples of relational algebra (RA) tors from text books and provided little in the way of comparison with relational calculus (RC) apart from the obvious procedural vs declarative styles of processing. An example of good examination practice would have been to compare the same query expressed in SQL as RA and RC equivalents. Some knowledge of predicate logic quantifiers and expressions is expected but very few candidates could express RC at all.

Some candidates spent a lot of time discussing SQL-99 and went down a blind alley as far as modern interpretation of RC. Any dialect of SQL is relevant but not as significant as natural logic based query languages such as datalog and prolog.

Question 2

2. Explain why a query optimiser within a relational database management system is capable of generating query plans which execute faster than those generated by a human programmer. **(12 marks)**

a) Given two tables:

OrderHeader(OrderNo, CustomerNo, OrderDate)
OrderLine(OrderNo,PartNo,Qty)

Consider the query "Get the CustomerNo of customers who order part P1".

Assume that there are 100 OrderHeader rows and 1000 OrderLine rows and that P1 appears in 50 OrderLine rows.

b) Write down two different relational algebra sequences that satisfy this query and demonstrate that one is more efficient than the other. State one general optimisation principle that is evident from this example. **(13 marks)**

Answer Pointers

a) DBMS optimisers are able to access statistics unavailable to the programmer

i) Number of values in each domain
ii) The current cardinality of a relation
iii) The current numbers of different values in a column
iv) The number of times each value is repeated
v) Etc.

These statistics will change over time. A DBMS optimiser will be flexible enough to plan new schemes as they change, a hand coded program is normally inflexible.

The DBMS optimiser can exhaustively evaluate different schemes; a programmer is likely to select one.

The skills of the best programmer can only be used on limited applications; the optimiser can be used by all users of the DBMS.

b) JOIN OrderHeader, OrderLine over OrderNo

Read each of the OrderLine rows (10,000), read each of the 100 OrderHeader rows 10,000 times. Temporary result will have 10,000 rows.

RESTRICT RESULT WHERE PartNo=P1

Read 10,000 tuples and output 50.

PROJECT RESULT OVER CustomerNO

Output 50 smaller tuples

Alternative:

RESTRICT OrderLine where PartNo=P1

Read 10,000 tuples and select 50 into temporary result

JOIN RESULT, OrderHeader over OrderNO

Read 100 order headers once, output 50 rows

Project RESULT over CustomerNO

As before

Second solution requires significantly less data access and produces smaller temporary results.

General principle is do RESTRICT before JOIN.

Examiner's Guidance Notes

A popular question. Most candidates were able to answer part a), however, in many cases the explanations for the better performance of query optimisers lacked detail and therefore did not attract the full mark. Fewer candidates attempted part b) although answers were generally good and the majority of attempts correctly identified the general principle of applying RESTRICT before JOIN. The answer given in the answer pointers is only a general outline and sensible solutions that differed from this but were justified by appropriate assumptions were also awarded credit.

Question 3

3. *a)* "Now we can state what might be regarded as the fundamental principle of distributed database: to the user, a distributed system should look exactly like a nondistributed system". [C. J. Date]

Discuss the subsidiary objectives that follow from this principle. **(18 marks)**

b) Outline three problems that are associated with distributed databases and give a brief description of the solution to each of them. **(7 marks)**

Answer Pointers

a)
Candidates are expected to raise the following issues (though not necessarily in the same terms):

Local autonomy: sites in the system should be autonomous
No reliance on central site: failure of on site should not cause rest of system to fail
Continuous operation: greater availability and reliability than single remote database
Location independence: identical queries run anywhere with same results
Fragmentation independence: ability to split tables across sites without user's knowledge
Replication independence: can duplicate data but system handles duplication
Distributed query processing: queries executes concurrently at different sites
Distributed transaction management: transaction control is not centralised
Hardware independence: system accommodates non-homogenous hardware
Operating System independence: not necessary that all nodes run the same OS
Network Independence: system operates over different network protocols
DBMS independence: Local databases talk to global databases from different vendors

b)
Answers may include (but are not limited to)

Query processing: standard optimisation problem now affected by the need to avoid shipping large numbers of rows across the network. Needs specialised query optimisation algorithm.

Catalogue management: not only the database but the catalogue should be distributed. This implies a global naming scheme and other innovations.

Update propagation: if copies of data are kept to improve system performance then these copies must be updated. A number of schemes to achieve this have been proposed.

Recovery Control: if a transaction fails then rollback must happen on all the sites affected, however the mechanism must account for sites that are temporarily unreachable during the time the transaction was active. Two phase commit is the solution.

Concurrency control: locks are no longer held centrally and there is a possibility of global deadlock. Sites must interchange wait-for graphs.

Examiner's Guidance Notes

This was an extremely popular question and was answered by the majority of candidates. Unfortunately, many candidates regarded the question as an opportunity to write everything they knew about distributed databases. Part a) was clearly directed towards those aspects of distributed databases which make the user unaware that the database is distributed. Answers that ignored this aspect of the question received little credit. A large number of scripts were marked down for not addressing the question set. Part b) allowed more scope and was in general answered well.

Question 4

4. There is increasing interest in using databases to store and process 'geo-spatial' data (also known as 'location-aware' data) that contains data about the geographic location of people and things. An application is given below that uses geo-spatial data.

a) Describe two different ways of representing geo-spatial data such as that implied in the application described below. Include in your answer the associated relationships and constraints that would be needed to express geo-spatial data. Give examples of how you might model the data and associated relationships and constraints using:

ii) A relational database
ii) An object-oriented database **(16 marks)**

b) Discuss the extended features of a relational query language (such as SQL) that will be needed to support the querying of geo-spatial data. Your answer should refer to the queries in the application described below.

An application that processes geo-spatial data.
A University has adopted a personal identity card (PID) system to improve security and to restrict access to different groups of people at certain times and dates. To enter a building or room a person swipes their PID card through a card reader outside the door of the building or room. The data collected from the card reader is stored in a database for later analysis. For example it is possible to calculate the route a person has taken, the time they enter and leave each area.
The database can support queries such as:
Display all the rooms that person X has entered within the geographic area of buildings Y and Z.
Display the nearest lecture room to lab Y. **(9 marks)**

Answer Pointers

This question covered section 4 of the syllabus specifically relating to new Database applications and partly concerned with constraint databases.

Part a)
Two main representation models exist; these being raster and vector oriented representations. With raster the geometric shapes of buildings are made up of pixels. The underlying topology can be expressed using quad tree regions (or equivalent) so that individual objects can deduced as being operated on as being - within, overlayed, adjacent to for example.
A better representation (and largely used in practice) is vector representations.

A relational model represents composite objects by simple relationships between nodes as in a directed graph. The complexity is realising the geometry lies in the spatial operations!

Primitive objects would include points, linear features, polygons (closed!). and topological rules (if point X is within polygon Y then X is contained within Y) If X crosses line (part of polygon). Data integrity constraints need building in to express semantics (these are just being built in with ITN2 data) so that routes have meaning e.g. cannot drive a car through here because the river signals a route across a river but this is not a bridge.

Relational model of routes is possible expressed as connected nodes as a directed graph, for examples as follows:-

NodeID	**x**	**y**	**seq**	**polygonid**
1	112	190	1	2
2	167	356	2	2
3	134	522	3	2
4	234	231	1	3

All the polygons include nodes ordered by a nodeID and sequence (in case the polygon is updated). Other tables are relevant such as a table of coordinates for all the routes and for a tracking table of objects (point data). This forms the basis of the data model for personnel tracking and sufficient for this question.

Candidates should appreciate the need to overlay objects over a base map, say, of the campus. The base objects being geometric shape primitives (lines, polygons and points). Each object constructs a geometry and would contain geo-spatial coordinates (e.g. lat/long) according to a specific Euclidian coordinate system (flat world, round world, azimuth/centipodal) with a geometric scale and bounded (tick points).

A OO view of this would involve creating classes and methods that encapsulate behaviour. Candidates should give simple examples that represent data and their relationships in terms of objects / classes and aggregation / association / generalisation / rules respectively. Complex data can be expressed by means of object / class hierarchies expressing implicit constraints expressed by means of generalisation and rules expressed as methods.

Part b)
Spatial query operators and object-relational features for defining constructs similar to ADTs are available in SQL3 but candidates are not expected to know these in detail. Oracle has advanced spatial operators again a specific nuance that candidates are not expected to know. Only the generic requirements are expected, therefore:

1. Geometries may interact with a given geometry.
(Oracle SDO_FILTER)
2. Determines the nearest neighbour geometries to a geometry.
(Oracle SDO_NN)
3 Determines whether or not two geometries interact in a specified way.
(Oracle SDO_RELATE)

4. Determines if two geometries are within a specified Euclidean distance from one another.
(Oracle SDO_WITHIN_DISTANCE)

The use of general spatial operators to act like special aggregate functions that work on collections or groups but seamlessly intersected with aspatial data. The return values (or result set) can contain say xy coordinates and attributes (label objects) for subsequent rendering on a map by client/middleware services. Also it is possible to produce a result stream in XML support the earlier premise of seamless interaction with other data formats and message formats.

For example a spatial join for computing intersections of regions for example in the second query and similarly for differing themes (lab use or lecture use)

Nearness queries request objects that lie near a specified location and require a nearest neighbour algorithm (note distance doesn't matter).

Examiner's Comments

Part a)
A difficult question for most candidates and for this reason it was the least popular question on the paper. Answers varied to a large extent. Many candidates emphasised the data model far too much spending a lot of time on designing a very detailed ER model of the scenario. Candidates should have balanced their answers better and thus give equal weight to the distinctive nature of geo-spatial data, that includes the ability to distinguish constraints (or business rules) that apply to the enterprise and those that have to represent spatial data constraints. For example,
Person X could have authentication and roles for admission to building X and room Y at certain times of day so this should be themeatically displayed. Candidates should appreciate the difference between a-spatial and spatial data and how these are blended in seamlessly

Part b)
Very few candidates produced any sensible answers to this part, mainly guessing (poorly) desirable elements of SQL rather than concrete examples.

Question 5

5. The protagonists of XML and supporting technologies claim that:

XML separates data structure from data representation.
XML will revolutionise data interchange and presentation on the web.
XML will revolutionise search and authoring on the web.

Discuss. **(25 marks)**

Answer Pointers

Expect some opening statement such as:
XML is known as the 'lingua franca' of the web it carries with it great expectation and hype but its usefulness is limited by that fact that the result of data access can only be of any use if it is directly 'useable/readable' by end users and the needs to be used by an application. In many cases results of data access need a combination of different types of data (formats/media etc) from different (possibly distributed) data sources. All of this needs to be packaged and presented to the end user.

Candidates should then deal with each claim specifically.

1- XML defines a standard first and foremost for describing and exchanging data on the web. With adoption by W3C in particular and supported by ODMG. Being ext based it makes it human readable and has set standards in new areas which have become contorted with bizarre formats and standards (e.g. GIS –GML Geographic mark up with XLST parsers and SVG grammars)
2- Therefore candidates should discuss the significance of the same data represented in XML can be displayed, queried (Xpath) and stored in exactly the same way. W3C are expected to apply standard stylesheet language for XML
3- Currently a lot of web searching is largely based on keywords and meta tags in HTML documents. Even though HTML pages have some structure, the structure is display oriented. By separating structure from presentation XML structure has meaning and can be used to provide more meaningful search. Richer forms of XML markup are being defined all the time (already mentioned GML) that specifically tune into an application domain and thus search on web sites that use these enriched markups will allow even more semantically based search patterns.

Candidates should also reserve some criticism mostly along the lines that XML cannot be used to solve every problem. The biggest issue is the provision of XML-type databases given that most XML files are massive and need the provision of DM techniques just like any persistent data store. The solution at the moment lies in middleware between RDB and Applications. A final point might come from CORBA enthusiasts in that distributed transaction models see XML based interoperability (transaction support) is very limited (e.g. web services/SOAP).

Examiner's Comments

Again most candidates could not structure their answers for this type of open-ended discussion – a little plan thought out at the beginning should be done.

Most candidates could express XML and are increasingly becoming familiar with the link with database technology. The notion of schemas and translation (XLST) was slightly less familiar to candidates. Some candidates went a lot further and thought XML/XLST) formatted text files would replace the relational model as persistent storage mechanisms without any qualification of the problems concerned with this approach. Real examples of XML enabled database applications were rarely found in answers. This suggests candidates are not exposed to development techniques and software tools such as XMLspy and embedded XML classes in languages such as Java and C#.NET.