THE BCS PROFESSIONAL EXAMINATION
Professional Graduate Diploma

October 2007

EXAMINERS' REPORT

Advanced Database Management Systems

General

Table 1 shows the statistics based on all questions answered. Averages are fairly consistent across questions, though, Q2 is a little on the low side. For standard deviation results, Q2 is relatively high as about one third of candidates failed to produce any meaningful answer, so that it has the lowest pass rate and the lowest average mark amongst questions; Q4 is the lowest amongst questions that means that students who have done this question made a relatively equal attempt. Questions 1, 2 and 4 were the most popular amongst candidates, whereas the more specific questions (3 and 5) were attempted by only a minority of candidates; thus it is concluded that candidates appear to prefer general questions on ADBMS issues and technologies rather than specific technique type questions on particular issues. However, for the specific technique type question Q3, it has the highest pass rate amongst questions, which means that most candidates who have answered this question are relatively competent in this particular knowledge.

Overall, from the evidence of marking results, the concepts of RDBMS and OODBMS/OORDBMS are described fairly adequate by most candidates, but OO and Web services programming are relatively challenging by the examination. It follows that candidates are expected to improve programming for ADBMS in their ability as the appreciation of these advanced techniques is essential to future ADBMS.

TABLE 1: stats for all questions answered

Examiner (initials)	Q1	Q2	Q3	Q4	Q5	Total
Number Attempted	99	70	26	95	43	
% Attempted	86.84%	61.40%	22.81%	83.33%	37.72%	
Number Accepted	99	70	26	95	43	
% Accepted	86.84%	61.40%	22.81%	83.33%	37.72%	
Number Passed	61	22	17	38	23	75
% Passed	61.62%	31.43%	65.38%	40.00%	53.49%	65.79%
Max Mark	24	25	20	25	22	70.66667
Min Mark	1	0	0	3	0	1.333333
Average Mark	11.62	6.79	10.15	9.24	10.19	37.49
Standard Deviation	4.82	6.59	5.59	4.05	5.15	13.19

Question 1

Relational database management systems are based on the relational model of data. It is often argued that there is no similar model underpinning object-oriented database management systems. Explain what is meant by the term *data model* and discuss whether the lack of an underpinning data model is a hindrance to the acceptance of object-oriented database systems.

(25 marks)

Indicative Solutions

Question 1: 25 marks. 5 marks for each key point and 5 marks for critical discussion indicated below.

A data model encompasses four things:

Data Structure: defines how data are organized: (hierarchical, network, relational, object-oriented).

Data Integrity: provides a language or implicit rules for maintaining data integrity in the data model *instance*.

Data Manipulation: provides a language to create, update and delete data.

Data Querying: provides a language to query data.

In relational databases everything is stored in relations. There are two integrity rules entity integrity and referential integrity. Data manipulation and querying is achieved through a set of relational algebra operators. Every relational database conforms to the same rules.

In object-oriented databases there are no hard and set rules for the structures that may be used. In fact it is generally expected that an OODBMS will be able to make any arbitrary complex data structure persist in permanent storage.

Since there are no rules as to the structures that can be created the data model is not able to specify rules which govern the validity of the structures.

OO databases generally specify a programming language which may be used to manipulate them. However, these programming languages are usually much more complex than relational languages. (Relational languages are relationally complete but not computationally complete).

There are at least two consequences to the fact that is not possible to specify a model for OO data. Firstly, it is not possible to specify a universal query language. The language OQL which is the nearest equivalent to SQL in the OO database world relies on users limiting themselves to structures which can be described by ODL. This is only capable of describing a subset of the structures which can be created in an ODBMS. The second consequence is that it makes it difficult for ODBMS products to interoperate as they have no common interpretation of the underlying data.

Examiner's comments

The majority of candidates (86.84%) answered question 1 and made a reasonable attempt (average 11.62/25; standard deviation 4.82). For the Data structure, it was answered well, but data manipulation and data querying caused a problem in that students failed to describe the basic concepts in providing a language to create, update and delete data; providing a language to query data. Most students assumed that data model is only related to data structure. They described the data structural difference between RDBMS and OODBMS reasonably well. About three candidates made an attempt to answer all the points mentioned in the indicative solutions.

Question 2

Almost all database management systems permit users to store and retrieve information about the date and times at which events occur. Given this, discuss the additional features that are expected of temporal database management systems.

(25 marks)

Indicative Solutions

Question 2: 25 marks. 10 marks for two important concepts to data, 5 marks for description of bitemporal concept, 5 marks for description of their relationship and 5 marks for critical discussion.

Temporal databases add two important concepts to data:

- Valid-time
- Transaction-time

Databases that support both of these concepts are said to be bitemporal.

Valid-time is the time that a fact is true in the real world. In a database that supports valid-time we do not just say that a part is red, we say when the part became red and when it ceased to be red.

Transaction-time records the time a transaction was made. Using transaction time a complete history of a fact as stored in the database can be recorded. We may for example that a part was coloured red between 2000 and 2006. We may subsequently discover that in fact the part was only painted red in 2001. Using transaction-time we can record the fact that for a given period we thought the part had been coloured red in 2000 and the date that this was corrected.

Both transaction-time and valid-time are time periods. Temporal databases will also deal with time instances. They will include special temporal functions to support instances and periods. For example functions will exist to determine whether an instance lies within a time period, whether one period encloses another and whether periods precede or antecede each other.

Examiner's comments

Less than two third of candidates (61.40%) answered question 2 and one third of candidates (31.43%) are passed and achieve an average low mark of 6.79/25; standard deviation is relatively high (6.59). One student answered all the points mentioned in the indicative solutions. About one third of candidates failed to answer most points mentioned in the indicative solutions. The common mistake is the lack of clear understanding in and confusion of the concepts between temporal databases and temporary databases.

Question 3 (This question uses Appendix A : EA database application)

a) Refer to Table A1 in Appendix A and the two Program batches below.

----- PROGRAM batch 1----------

```
CREATE VIEW View1
(when_taken, precipitation, rating)
AS
SELECT when_taken, precipitation,
(SELECT COUNT(DISTINCT when_taken)
FROM measurements2005 AS T1
WHERE T1.when_taken <= T0.when_taken) AS rating
FROM measurements2005 AS T0
```

----- PROGRAM batch 2----------

```
SELECT P1_WHEN=V1.when_taken, P2_WHEN=V2.when_taken,
P1=V1.precipitation, P2=V2.precipitation,
DIFF=(V2.precipitation - V1.precipitation)
FROM View1 AS V1 LEFT OUTER JOIN View1 AS V2
ON (V2.rating=V1.rating + 1)
ORDER BY V1.Rating
```

i) Work out the output produced from PROGRAM batch 2

(6 marks)

ii) Explain why the above code is inefficient given the context in which it runs.

(5 marks)

iii) Rewrite the above Program Batches so that they run more efficiently.

(8 marks)

b) Explain how you would satisfy the following requirement:

SQL code is required that will *efficiently* compare data selected from 2 different measurement tables, one that contains data from 5 years ago, the other that contains data from 9 years ago. The selection criteria might include a range of dates or a set of observatory data for a particular date and river.

(6 marks)

Indicative Solutions

Question 3.a: i) 8 marks listed in the indicative solutions.

Candidates have to realise the output depends on the View. – **1 mark**
V1 and V2 are virtual tables that accesses the measurements table with a computed column (the inner select). – **1 mark**
The query is complicated because of the use of a correlated query construct (i.e. build from two instantiations of itself.
Typical output will be (1 mark for getting each)
Date(WhenTaken) – first interval – **1 mark**
DataWhen taken) – second interval – **1 mark**
Precipitation at the first interval (from V1) – **1 mark**
second interval (from V2) – **1 mark**
The difference between the 2 precipitation measurements – **1 mark**
And the OUTER JOIN includes non-matching ratings (ranked) values which is the derived column created in the View. – **1 mark**

Providing sample data (Measurements) will help in understanding the expected output, - hence the term ‘work out’ is used.

Question 3.a: ii) 5 marks. General distribution of marks according to the candidates' descriptions.
The code is very inefficient because of the run-time performance of maintaining a view – this is computed at run time and unlike a table has to be constructed every time to support the underlying query /SQL statement. It is also very inefficient to use in-built selects within selects as this means the inner select has to compute values before the outer select can work. Similarly the use of correlated queries is poor for performance because the derived table has been generated on the fly as virtual tables. Virtual tables cannot be optimised as the optimiser has no prior knowledge of memory of their execution

Question 3.a: iii) 8 marks. General distribution of marks according to the code created for 1) create table commands and 2) sub-query command are used.
There are many solutions to this depending on how much programming functionality is known about. It is expected students will substitute the View V1 with a table variable so that the definition of V1 becomes CREATE TABLE @View1 (when_taken datetime ……..) The View is then not necessary but the query (batch 2) can be merged with the select part of batch1 to form 1 query. The second improvement would be to persist the derived Rating column i.e. compute the COUNT and store it. The downside is the overhead of updating it. The batch 2 code might look like the same but using different constructs and adding the where clause from the program in Batch1. For the really bright students they might use a sub-query instead of an outer Join. This is valid but would not make the stark improvement compared to the other things mentioned. Other marks awarded include use of stored procedures. SELECT ……, Rating FROM v1.rating IN (SELECT …..)

Question 3.b: 6 marks. General distribution of marks according to 1) the general description of selection functions of SQL for 3 marks, and 2) advanced techniques for 3 marks.
This again might raise a few different answers depending on vendor support for efficient gains made by constructs such as partitioning (scaling out) using a partitioned view so that many servers can operate on the large data set in parallel also candidates might mention the use of an indexed view whereby the index leaves of a B-tree persist the access paths to a large data set. In Oracle this is called a materialised view. Most of these solutions are valid but the students will need to appreciate these advanced techniques in practice to fully apply these to the problem stated.

Examiner's comments

Less than a quarter of candidates (22.33%) answered question 3 and achieve an average mark of 9.24/25; standard deviation 5.59. More than two third of candidates used appendix well so that in part 3.a, answers are reasonably matched to some of indicative solutions. Part b is relatively challenging to the candidates. Although most candidates answered the question, they have not discussed it adequately as expected in the indicative solutions. For the whole question, about 4 candidates missed most points mentioned in the indicative solutions.

Question 4 (All parts of this question refers to Appendix A)

a) Explain with the aid of examples how you would differentiate a Data Warehouse from a Database System. Use examples from the scenario given in Appendix A in your answer.

(6 marks)

b) With reference to Appendix A give an example of an application that would require the support of a Data Warehouse and associated analysis tools. Include in your answer the type and nature of the data represented in the Data Warehouse and the software tools you would use. State any assumptions you have made.

(10 marks)

c) With reference to Appendix A explain the stages required updating a Data Warehouse with the raw data that has been recently collected from observatories and weather feeds.

(9 marks)

Indicative Solutions

Question 4.a: 6 marks. 4 marks for explanation, 2 marks for providing examples
Data Warehouse: Typically in the nature of use (OLAP) and the nature of its storage organisation being geared for analysis and data mining of non-relational data (Cubes). Data Warehouse has to support decision support type of queries and ad-hoc queries are non SQL standard for example MDX. Some people think a data warehouse is closer to a large excel spreadsheet containing pivot tables than it is to a database system. Database system – mainly OLTP nature of use – transaction based with performance and data integrity of high importance. Support relational model and normalisation. More standardised from vendors than a data warehouse with SQL standardisation fairly prominent. Have the tools to support conversion and population of a DW rather than the other way round.

Question 4.b: 10 marks. Marks are listed in the following indicative solutions.
The answer needs concrete examples of a Cube data structure from the Case Study this means a specification of the dimensions Define dimensions - River level; Time Period; Rainfall; Location **1 mark** Using these dimensions measures are defined to define discrete boundaries of the Cube i.e. a cell might define a rainfall value at a particular time on a particular river from different observatories and this could be used to compare the same results for a different date therefore showing the speed at which a flooding event occurred. This would allow the emergency services to assess the minimum and the maximum time necessary to evacuate people located next to a river. **3 marks** Candidates must provide similar realistic examples and show ***imagination*** in how the data is visualised (maybe using maps or hierarchical pivot tables) **3 marks** Candidates will gain extra marks for referencing DW tools they have used or are familiar with and how these tools are used **3 marks**

Question 4.c: 9 marks. 3 marks for key stages, 6 marks for explanation that must be credible and applied to EA.

Answering this question will benefit candidates with real knowledge of a DW system, this might have been studied via on-line tutorials such as those supplied by msdn.com or Microsoft learning.com. Students can also benefit from applying their own experience of hands-on use of DW tools.

In particular candidates have to identify a process or method and the key stages of the process of getting raw data into a DW.

These are the key stages:

Extraction
Transformation
Load
3 marks

The strategy for EA might follow these lines

Extraction (data feeds to local files (comma separated or CSV files includes cleaning the data of missing/erroneous values and reporting back these errors)

Transformation (Integration services or bulk load utility first to a local DB

For consistency there must be further checking and filtering (apply integrity constraints on relational data before transforming to starfish schema)

Load (regulate or schedule feeds into existing data stores in DW structure may need to reconcile/compute aggregate data on the fly prior to population and load.

2 marks each explanation must be credible and applied to EA = 6 marks in total

Examiner's comments

Most candidates (83.33%) answered question 4 and achieved an average mark of 9.24/25; it has the lowest standard deviation 4.05 amongst questions. In general, candidates made a reasonably equal attempt to answer the question. One candidate answered all points mentioned in the indicative solutions. Majority of candidates described the differences between data warehouse and databases. In part c, most candidates described partially about the identification of a process or method and the key stages of the process of getting raw data into a DW, but failed to discuss them adequately as expected in the indicative solutions.

Question 5

a) Explain the function of the following components of *Web Services* technology - WSDL; UDDI ; SOAP. Use a diagram to assist your answer.

(6 marks)

b) Compare *Web Services* with CORBA and explain why Web Services/SOA has become more widely used than CORBA.

(9 marks)

c) Explain, with reference to stated software tools you are familiar and supported by example programs, how a Web Service would be deployed to enable access to data running on a distant server.

(10 marks)

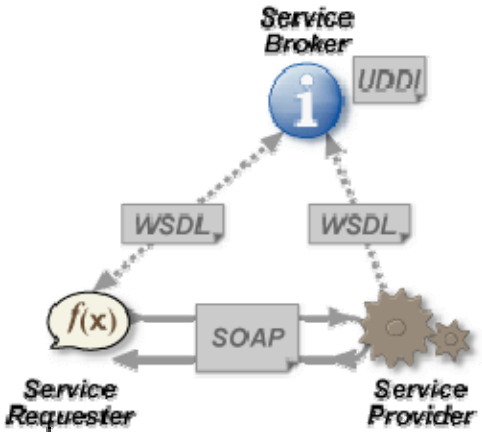
Indicative Solutions

Question 5: Part a): 6 marks. 2 marks for each explanation but drawing useful as well.

SOAP; WSDL and XML – web services

Web services use SOAP – (Simple Object Access Protocol)-formatted XML envelopes and have their interfaces described by WSDL (web services description language) written in XML. A WSDL document describes a Web service in detail, the messages the Web service will send, the data types used in those messages, the style of the messages, and also the location of the service. Web services can are used to implement a Service-oriented architecture (SOA) over HTTP, where the basic unit of communication is a message, rather than an operation. This is often referred to as "message-oriented" services – diagram to describe.

2 marks each explanation but drawing useful as well



Question 5: Part b), 9 marks. Each point gains 3 marks with an option for 4 points.

What is CORBA?

CORBA - Common Object Request Broker Architecture is a standard defined by the Object Management Group (OMG) that enables software components written in multiple computer languages and running on multiple computers to interoperate.

It is a standard that was adopted to support distributed object architectures Using the standard protocol IIOP, a CORBA-based program from any vendor, on almost any computer, operating system, programming language, and network, can interoperate with a CORBA

In answering this part candidates have to equate the common techniques used to gain interoperability these are highlighted below – each point gains **3 marks with an option for 4 points (i.e. 3 points required in answer scheme)**

Point1

SOAP is an XML-based protocol intended to exchange information between senders and receivers, usually over HTTP, in decentralized distributed environments. Its protocol consists of 3 parts. Like IIOP, SOAP is a protocol for conveying messages between applications. Unlike IIOP, which represents message data in binary format, SOAP represents its message content using XML.

Point 2
CORBA communicates not via HTTP but via RMI (remote method invocation)
In SOA/web services XML provides the structured flexible description and definition of data where as CORBA primarily focuses on system functionality rather than system data. CORBA is vendor neutral largely supported an independent body – the OMG where as SOA/web services is mainly vendor specific with Microsoft one of the big players.

Point 3
IDL is used to exchange data use structs and sequences. For example, you could define a tree node in IDL like this:

```
struct XMLElement;
typedef sequence<XMLElement> XMLElementSeq;

struct XMLAttr {
   wstring name;
   wstring value;
};
typedef sequence<XMLAttr> XMLAttrSeq;

struct XMLElement {
   wstring name;
   wstring value;
   XMLAttrSeq attributes;
   XMLElementSeq children;
   // ...
};
```

CORBA applications can also represent XML data as IDL strings but this has caused problems such as
It's inefficient because it requires transformation to or from string form whenever an application wants to send or receive XML data from another CORBA application.
It's error-prone because it relies heavily on programming conventions

Point 4 optional about message based protocols etc

Question 5: Part c): 5 marks for development frameworks, 5 marks for example programs.

5 marks for development frameworks candidates have experience or done tutorials on thus:
Developers use toolkits like Axis and Visual Studio to generate WSDL dynamically EG Axis. Axis is an open source toolkit that is developed as part of Apache. Axis allows developers to write Java code and deploy that code as a Web service. There are several ways of deploying as a Web service one way is to place a file with a JWS extension into an Axis Web application. This file will contain Java source code. Any public methods in a JWS file are exposed as methods on the Web service. Instead, most Axis users will write standard Java code and then make that code available as a Web service. A simple example is required using code.

5 marks for example programs – simple as possible thus:
For example a Java class that looks like this:

```
public class MathService
{
        public double add(
                double d1, double d2)
        {
        return d1+d2;
        }
}
```

You can deploy this class as a Web service in Axis by providing a server-config.wsdl file that

```
contains this section:
<service name="MathService"
        provider="java:RPC"
        style="wrapped" use=
        "literal">

        <parameter name="className"
                value=" MathService"/>
        <parameter name=
                "allowedMethods" value=
                "*"/>
</service>
```

Examiner's comments

More than one third of candidates (37.72%) answered question 5 and made a reasonable attempt (average 10.19/25; standard deviation 5.15). More than half candidates passed. One candidate hardly produced any meaningful answer mentioned in the indicative solutions. For part a, majority of candidates answered very well. For part b, less than half candidates described the definitions of CORBA and Web services; most candidates described message based protocols. For point 1 of part b, when the discussion related to the SOAP, Microsoft .Net is mentioned. For point 2 of part b, it is rarely mentioned CORBA with RIM. For point 3 of part b, very few candidates gave the example code of IDL. For part c, common technology like Microsoft .NET Visual Studio has been addressed but points are not adequately described as expected in the indicative solutions. Not many candidates provided the example of program.