BCS THE CHARTERED INSTITUTE FOR IT
BCS HIGHER EDUCATION QUALIFICATIONS
BCS Level 6 Professional Graduate Diploma in IT
MARCH 2014

EXAMINERS' REPORT
Advanced Database Systems

Part A

QA1 [Attempted by 72 students]

(a) Using a simple example of your own choosing, explain and discuss the following data warehouse design issues (and how they relate/differ to the traditional relational data modelling process for an Online Transaction Processing (OLTP) database):
Star schemas
Snowflake schemas
Dimensional Modelling

You should particularly address the roles of primary and foreign keys, normalised & de-normalised data. Good diagrams are essential. **(10 Marks)**
This will be marked holistically with bonus marks for clear diagrams of the various models but the key points to be covered are:
Dimension modelling as a specialized example of an ER Model – both based on entities and relationships, the use of a *fact table* (with composite primary key) and a set of *dimension tables* (each with an atomic primary key) related to the fact table via foreign keys – thus producing a *star schema* (star join) model. The better students should then go on to discuss issues such as the fact table is much larger than the dimension tables, that the fact table works best when the 'facts' recorded are numeric (grades, prices, ages etc) thus allowing aggregated computations to be run leading to summarized data, that dimension tables tend to hold more descriptive data (names, addresses, identifiers etc), the use of de-normalized data to replicate attributes across multiple dimension tables (for example, storing address or contact data in several different dimension tables) thus avoiding additional joins and enhancing query performance. Finally, a few words on what a *snowflake schema* is (where dimensions can have their own dimensions) – caused by normalizing the original dimension table data down into two or more child dimension tables, all linked to the 'parent' dimension table via the familiar PK/FK technique. So *star schemas* use de-normalized (repeated) data and *snowflake schemas* use normalized (minimized duplication) data.
The star and snowflake aspects were very well covered with just about all students providing a good diagram and a clear explanation of fact and dimension tables plus, in many cases, a further explanation of the role of normalization. A smaller number discussed the OLTP/ERD connection. The dimensional modelling responses were either very good or very bad. The good ones provided more

quality (data cube) diagrams while the bad ones simply ignored it or made short superficial comments. Overall, this was a good question for most candidates.

(b) Discuss and explain the use of the following data warehouse techniques:
Summary Management
Analytical Functions
Indexing & Optimization Techniques **(10 Marks)**
Marks spread evenly over the following topics…
Summary management – DW queries are often seeking aggregated data, not the fine detail of individual rows, particularly aggregation via specific dimensions (month, product, region etc.) so the DBMS must support pre-computed summaries and aggregates to avoid run-time computation.
The concept of aggregated data and the need for summarization within a DW was clearly grasped by most students but there was a tendency here to generic and high-level discourse without the finer details of pre-computed values being stored or the type of aggregation that may be employed across specific data dimensions. So correct, but often lightweight, responses.
Analytical functions – many BI and DW applications want to use SQL ranking and aggregate functions, cumulative aggregates or maybe the CUBE and ROLLUP operators for OLAP analysis.
Many students touched upon the roll-up and drill-down concepts but very few got into the specifics of SQL statements used in BI operations – so the same issue as the summary management – correct, but generic.
Indexing & Optimization -
Bitmap indexes – via compression, the support for low cardinality data and more 'open' type of queries rather than the usual B-tree indexes used to search for individual identifiers.
Advanced join methods - such as in-memory hash joins and partition-wise joins – very useful for the large data volumes involved in a DW.
Advanced optimizers - that can use statistics and histograms - especially on skewed data distributions – such as Oracle's cost-based optimizer when working on star schema queries.
Function-based indexes - that can pre-index complex calculations and other expressions often used in DW applications. Sampling functions that can gather mathematical data like averages without having to read every row.

Bitmap indexes and the associated issues highlighted in the above marking scheme (cardinality etc.) were well addressed by many students (often supported by good case studies and bitmap diagrams) but the remainder of these marking scheme points (function-based indexes, cost-based optimization and advanced joins) were never seriously discussed.

(c) Briefly describe the different *characteristics* (not specific products) of types of *data analysis and extraction* tools that a data analyst may use to interact with a data warehouse, briefly highlighting the primary purpose of each. **(5 Marks)**
Major categories are:
Reporting tools (end-user desktop)
Query tools (SQL development and QBE interfaces)
Application development tools (for more advanced and regular tasks)
EIS tools (decision-support)
OLAP tools (multi-dimensional queries)
Data mining tools (identification of new, unpredicted patterns and trends).

By far the worst aspect of question 1 - with a very small minority of students addressing the issues raised in the marking scheme. Sadly, the bulk of responses to part (c) were short and/or superficial.

QA2 [Attempted by 87 students]

(a) Using your own simple examples and/or time-line diagrams, describe how data may be damaged, lost or misread in a *multi-user* database if *concurrency control* techniques are not fully implemented.
(10 Marks)
Marked holistically, but to cover…
The lost update problem
Dirty (uncommitted) read problem
Non-repeatable reads
Phantom reads

For each, a description of the problem and a simple time-lapse example involving two transactions T1 and T2 is needed for full marks. Referral to the different types of schedule – serial and interleaved – and the impact on these problems would get bonus marks.
Generally very well answered – with the vast bulk of students addressing the lost update problem, dirty reads issue and the better ones moving onto the other potential concurrency problems. Answers tended to be extensive and detailed with many good case studies and time-line diagrams also supplied.

(b) For each of the following transaction control terms, write a *single sentence* (no need for extended responses, examples or diagrams) explaining the key concept.
Schedule
Cascaded rollback
Optimistic locking
Pessimistic locking
Checkpoint

(8 Marks)

Schedule – a sequence of operations drawn from two or more concurrent transactions that preserves the order of the operations in each of the individual transactions. A schedule can be serial or interleaved.
Cascaded rollback - when transaction T1 fails and induces a rollback which in turn causes other transactions - which were dependent on the success of T1 – to likewise fail and be rolled back.
Optimistic locking – based on the assumption that inter-transaction conflict is rare so individual transactions are allowed to proceed unsynchronized and are only checked for conflicts at the end – just before commit. Useful in low data contention environments because it avoids the overhead of locking/waiting for unlocking but inefficient if data conflicts are common as transactions will have to be repeatedly restarted.
Pessimistic locking – assumes a high degree of inter-transaction conflict and locks up all data resources ahead of access immediately – even if other transactions never try and access them. Saves re-running transactions in highly contentious environments but this 'belt and braces' approach can induce overhead in locking/releasing data resources that were never in conflict in the first place.
Checkpoint – A point of synchronization between the database and the transaction log file (journal) that allows easy identification of which transactions need to be re-processed (redo/undo) in the event of database recovery being needed.
Despite some interpretation and modest flexibility being called upon by the marker, most students clearly had a sound grasp of the above five concepts and were soundly rewarded where appropriate. This was a good question for most students.

(c) In your own words, describe what is meant by the following *transaction-processing* terms:
Two-phase locking (and the function of each stage)
Serializability (and the role of serial and interleaved schedules)

You should supply any suitable examples and/or diagrams that you deem appropriate to support your answer. **(7 Marks)**
Marked holistically but the following points should be addressed…
Two-phase locking should mention the concepts of a transaction having two distinct stages: a 'growing' phase where it acquires all locks necessary for it to complete its tasks (and cannot release any locks) and a 'shrinking' phase in which it systematically releases those locks and returns the data resources as it runs down (and is not allowed to acquire any new locks). In other words, 2PL ensures that a transaction must acquire a lock on a data item before doing any work on that data and once a transaction has finished with a lock and cannot grab more locks.

This was almost universally well answered. Most did not supply diagrams but the text made it quite clear that they knew what the main issues were.

Serializability is the idea that parallel transactions can execute concurrently - via interleaving (using a non-serial schedule) - yet without interfering with one another – so as give a final outcome on the database the same as if those transactions had been executed in a sequential (serial) manner. 2PL thus stops two competing transactions from colliding over the same data item(s) and thus violating the ACID principles.

Not as strongly answered as the two-phase locking responses, but most students grasped the core idea (often supported by clear time-line diagrams) so, as a whole, part (c) was a good question for most students.

Q3

Consider the following database that contains information about directors and the films they have directed:

Film (filmNbr, title, year)

Director (directID, name)

Directs (directID*, filmNbr*)

(a) Consider the following query:

```
SELECT Film.title
FROMFilm, Director, Directs
WHERE Film.filmNbr = Directs.filmNbr
AND   Director.directID = Directs.directID
AND   Director.name = 'Lucas';
```

Suppose this query is run by executing the following sequence of steps:

1. R1 = Join of Director and Directs
2. R2 = Join of Film and R1
3. R3 = Selection (name = 'Lucas') from R2
4. R4 = Projection (title) from R3

(i) What is the problem caused if the query is executed based on the sequence above.

(ii) Suggest a new sequence that will make the query more efficient. Hint: You may need to introduce extra steps and not just re-arrange the existing steps.

(4, 12)

Answer Pointers and Marking Scheme:

(i) By joining the tables together first, large intermediate tables may be created which may be too large to process and too expensive to read from secondary memory. This would also be an inefficient use of computing resources (CPU, I/O, memory) and would increase query processing time. (4 marks)

(ii) A more efficient way of processing this query is to reduce the amount of records that need to be read from the tables as early as possible by performing selections and projections. A better sequence could be as follows: (2 marks for each step)

R1 = Selection (name = 'Lucas') Director
R2 = Projection (directID) R1
R3 = Join of R2 and Directs
R4 = Projection (filmNbr, title) Film
R5 = Join of R3 and R4
R6 = Projection (title) from R5

Note that the answer could also be given in the form of a query tree.

Examiner's Comments: Most students had a good attempt at this question.

(b) Suppose there is an index on the column "title" of the "Film" table above. Explain how this index could be used when executing each of the following queries:

(i)

```
SELECT *
FROM Film
WHERE title = 'The God Father';
```

(ii)

```
SELECT *
FROM Film
ORDER BY title;
```

(iii)

```
SELECT COUNT(title)
FROM Film;
```

(3, 3, 3)

Answer Pointers and Marking Scheme:

(i) Given that the search is based on the title (WHERE clause), the index will serve to locate the specific title (move from the root down to a leaf node in a B tree index) and, then, point to the specific record, in the data file, associated with this title. **(3 marks)**

(ii) An index on title would already be sorted and could therefore be used in the first instance to read all titles (leaf nodes in a B tree index) and then point to the associated records in the data file. This will avoid having a sort operation done separately after a table scan. **(3 marks)**

(iii) The index is all what is needed to run this query (no need to access the data file). All what needs to be done is a full scan of the index (leaf nodes in a B tree index) in order to count the number of titles. **(3 marks)**

Examiner's Comments: Most students managed to recognise the way an index could be used in the first query. However, not many understood how an index could be used in the two other queries.

QUESTION B4

Examiners Comments

Quite a popular question with over 60% attempts. But the performance was worse than expected with only a third getting more than 10 marks. The poor performance could be attributed to the level of answers that candidates expect to produce at this level. Many candidates failed to exploit the rich and diverse case study to produce any kind of meaningful examples particularly to part b). Answers were generally woolly and insignificant. It was disappointing because GIS as a database application has appeared in previous papers and covered very much the same ground. Furthermore knowledge of Object Oriented data modelling formed the most important part of Part b) (First part) and maybe because this was not explicitly stated, actual modelling examples from the discourse were missing. The answer pointer below shows that OO modelling examples were necessary in candidates answers. An OO model also revealed the relational model limitations. The second part of the EITHER OR question also produced disappointing answers that again were woolly and below the level expected usually focusing on candidates experience of on line mapping services such as google and bing maps

QUESTION B4

A GIS is a database application that processes geo-spatial data, in other words information about objects that exhibit geographical features on a map. There are three basic type of geo-spatial objects the simplest is based on a single point or location having a geographical reference such as latitude and longitude. Consider the following scenario

A University has adopted a personal identity card (PID) system to improve security and to restrict access to different groups of people to certain locations such as buildings and rooms within buildings. Permission to enter a building does not necessarily allow access to every room in that building. To enter a building or room a person swipes their PID card through a card reader outside the door of the building or room. Staff and students can only enter designated buildings or rooms once their access rights are confirmed. When a card is swiped the PID reader records who has entered a building or room. Using the PID system it is possible to calculate the routes taken, when an individual enters or leaves a particular zone building or room.

With specific reference to the scenario above:-

a) Give an example of **two further** basic types of geo-spatial data made up of points that would be used to model the PID system.

(5 marks)

EITHER

b) Drawing upon examples from the scenario above, discuss the limitations of the relational model in modelling the relationship between geo-spatial objects

(20 marks)

OR

Describe the characteristics of modern geo-spatial database applications that have allowed them to become very popular and easily accessible over the WWW and describe how these could apply to the PID system.

(20 marks)

ANSWER POINTERS Part a)

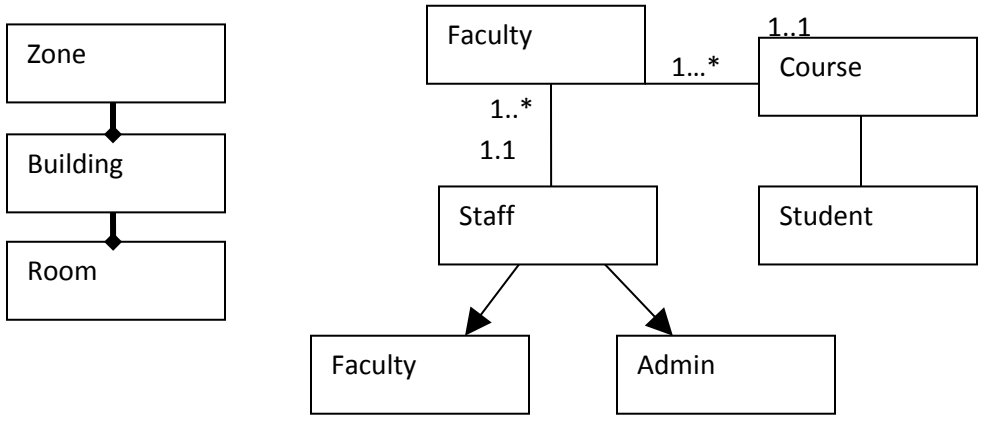
The three types include with an example of each for example :

polygon – zone that restricts access

Line – a route that a person is tracked from PID to PID access places

(3 marks each example 1 for listing)

ANSWER POINTERS Part b)
First EITHER OR question
Candidates should cover the limitations of representing modelling constructs that would be need to be covered such as
Whole part ie aggregation (or composition}
Generalisation
Spatial objects referenced in the Scenario are classes PID. Rooms.Buildings, zone, staff, students, faculty, group
Examples should be derived from case study … this was stated in the question. We have 2 examples of containment (whole-part)
Polygons. A PID may be located outside a room or outside a building. In turn rooms are contained within buildings so to access a room we must first access a building and in turn we could collect all buildings within a zone so we model something this like…



For instance relationship we have Faculty manages many courses and a course is based in one Faculty. Also staff belong to one faculty and a faculty has many staff.
Generalisation staff are specialised into Faculty Staff and AdminStaff and Security Staff. For Faculty staff a group is created that restricts access to staff contained in this group

ANSWER POINTER Part b) second EITHER OR QUESTION
Eg Google maps Bing maps open data standards in which GIS data is consumed and portable across systems. Typical example of cloud computing with user access through a browser but with rich client content to enhance interactivity. Bandwidth limits the amount that can be downloaded.
PID application this was expected to provide an open ended answer. Candidates were expected to mention access to remote database on tablet/phone for example room location on map. Access to timetable to show what building room x is located. Examples from user experience with an explanation of the role of databases in the technology involved.

QUESTION B4

Examiners Comments

Quite a popular question with over 60% attempts. But the performance was worse than expected with only a third getting more than 10 marks. The poor performance could be attributed to the level of answers that candidates expect to produce at this level. Many candidates failed to exploit the rich and diverse case study to produce any kind of meaningful examples particularly to part b). Answers were generally woolly and insignificant. It was disappointing because GIS as a database application has appeared in previous papers and covered very much the same ground. Furthermore knowledge of Object Oriented data modelling formed the most important part of Part b) (First part) and maybe because this was not explicitly stated, actual modelling examples from the discourse were missing. The answer pointer below shows that OO modelling examples were necessary in candidates answers. An OO model also revealed the relational model limitations. The second part of the EITHER OR question also produced disappointing answers that again were woolly and below the level expected usually focusing on candidates experience of on line mapping services such as google and bing maps

QUESTION B4

A GIS is a database application that processes geo-spatial data, in other words information about objects that exhibit geographical features on a map. There are three basic type of geo-spatial objects the simplest is based on a single point or location having a geographical reference such as latitude and longitude. Consider the following scenario

QUESTION B5

Examiners Comments

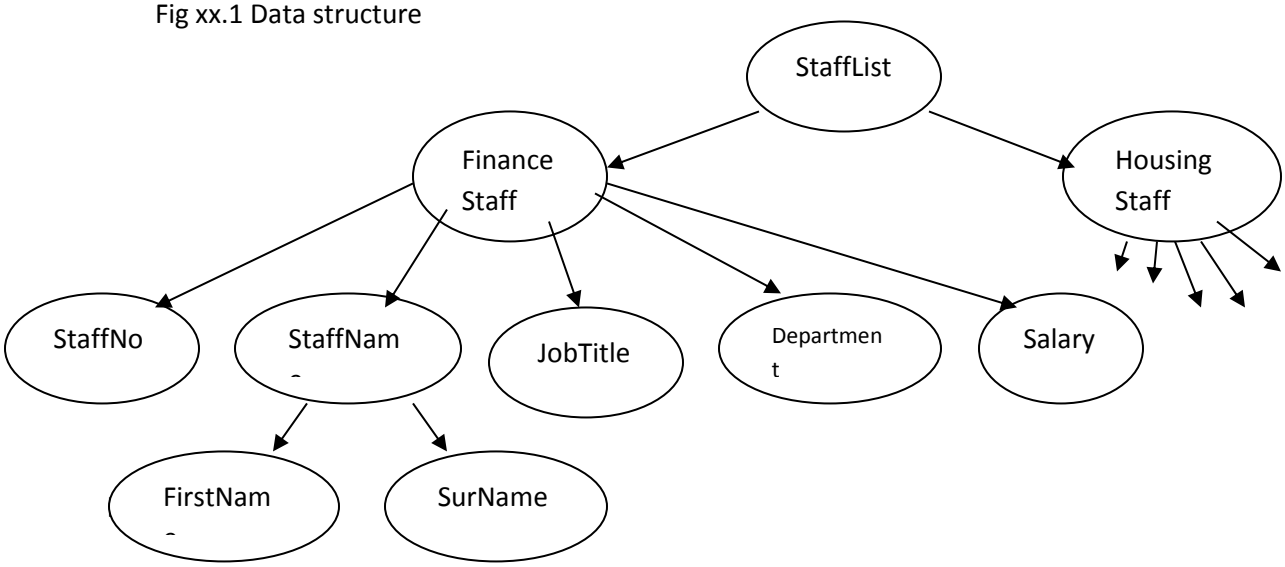
The examiner was generally pleased with candidate performance on this question generally on XML databases. Just fewer than half attempted it with an average mark close to 13/25. There were no major issues that concerned the examiner except that it was quite common for candidates to miss off the namespace in answers to part a).

a) Represent the contents of Relation StaffList as an XML document using the data structure given in Fig A2[1] using a namespace called *StaffList* with a url = http://www.Idealhomestaff.co.uk/branch. [8 marks]

Relation StaffList

StaffNo	**FirstName**	**SurName**	**Department**	**Salary**	**JobTitle**
123	Robert	Rogers	Housing	34300	Manager
126	Duncan	Boult	Finance	22000	Accountant

Fig xx.1 Data structure



ANSWER part a) a rudimentary knowledge of XML was required if present this was an easy question…. Document would look like this with namespace included (important)

<STAFFLIST xmlns=http://www.idealhome.co.uk/branch>
<STAFF deptno=”FinanceStaff”>
<NAME>
<FNAME>John</FNAME>
<NAME>
<JOBTITLE>Manager</JOBTITLE>
<SALARY>23212</SALARY>
---- etc
<STAFF deptno=<”personal”>
<NAME> etc
Must have accurate syntax ie no trailing ‘<’ or ‘>’

b) XQuery is a language used to query XML data.

Describe with the aid of examples using the XML document you produced in fig A2[1] how each of the following types of XQuery expressions are formulated

- path expressions,
- conditional expressions, **(10 marks)**

ANSWER POINTER part b)
Path expressions are used to get an ordered list of nodes including decendent nodes
Eg find the staff number of the first member of staff
Doc(“stafflist.xml”)/STAFFLIST/STAFF[1]//STAFFING
Conditional expressions Find if there are any staff listed (very simple example expected).

<Staff>
{ $b/Staffid }
{ $b/Staffname }
{
IF (empty($b))
THEN <status>inactive</status>
ELSE <status>active</status>
}
</user>

c) Explain the type of expression and the function of the following XQuery expression

```
{
LET $doc := document("stafflist.xml")
FOR $t IN distinct($doc/staff/salary)
LET $p := $doc/staff[jobtitle = $t]/salary
RETURN
<avgesalary jobtitle={ $t/text() }>
{
avge($p)
}
</avgesalary>
}
```

(7 marks)

ANSWER POINTER part c)
This Xquery is a FLWOR expression and uses variables to assign values in the document it resturns theb average salary for each job title listed in the xml file. The output is XML and a document that can be subsequently queried. Expected an explanation of each line of code to get full 7 marks.

A University has adopted a personal identity card (PID) system to improve security and to restrict access to different groups of people to certain locations such as buildings and rooms within buildings. Permission to enter a building does not necessarily allow access to every room in that building. To enter a building or room a person swipes their PID card through a card reader outside the door of the building or room. Staff and students can only enter designated buildings or rooms once their access rights are confirmed. When a card is swiped the PID reader records who has entered a building or room. Using the PID system it is possible to calculate the routes taken, when an individual enters or leaves a particular zone building or room.

With specific reference to the scenario above:-

c) Give an example of **two further** basic types of geo-spatial data made up of points that would be used to model the PID system.

(5 marks)

EITHER

d) Drawing upon examples from the scenario above, discuss the limitations of the relational model in modelling the relationship between geo-spatial objects

(20 marks)

OR

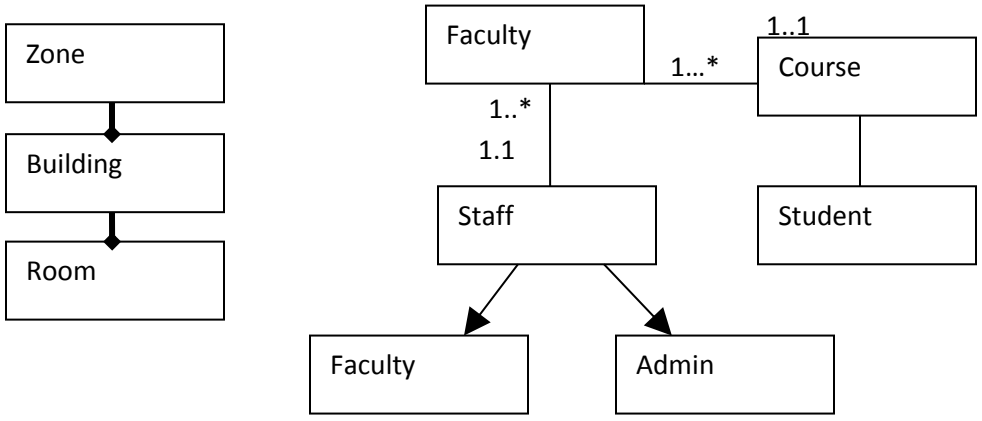
Describe the characteristics of modern geo-spatial database applications that have allowed them to become very popular and easily accessible over the WWW and describe how these could apply to the PID system.

(20 marks)

ANSWER POINTERS Part a)
The three types include with an example of each for example :
polygon – zone that restricts access
Line – a route that a person is tracked from PID to PID access places

(3 marks each example 1 for listing)

ANSWER POINTERS Part b) first EITHER OR question
Candidates should cover the limitations of representing modelling constructs that would be need to be covered such as
Whole part ie aggregation (or composition}
Generalisation
Spatial objects referenced in the Scenario are classes PID. Rooms.Buildings, zone, staff, students, faculty, group
Examples should be derived from case study … this was stated in the question.
We have 2 examples of containment (whole-part)
Polygons. A PID may be located outside a room or outside a building. In turn rooms are contained within buildings so to access a room we must first access a building and in turn we could collect all buildings within a zone so we model something this like…



For instance relationship we have Faculty manages many courses and a course is based in one Faculty. Also staff belong to one faculty and a faculty has many staff.

Generalisation staff are specialised into Faculty Staff and AdminStaff and Security Staff. For Faculty staff a group is created that restricts access to staff contained in this group

ANSWER POINTER Part b) second EITHER OR QUESTION
Eg Google maps Bing maps open data standards in which GIS data is consumed and portable across systems. Typical example of cloud computing with user access through a browser but with rich client content to enhance interactivity. Bandwidth limits the amount that can be downloaded.

PID application this was expected to provide an open ended answer. Candidates were expected to mention access to remote database on tablet/phone for example room location on map. Access to timetable to show what building room x is located. Examples from user experience with an explanation of the role of databases in the technology involved.

QUESTION B5

Examiners Comments

The examiner was generally pleased with candidate performance on this question generally on XML databases. Just fewer than half attempted it with an average mark close to 13/25. There were no major issues that concerned the examiner except that it was quite common for candidates to miss off the namespace in answers to part a).

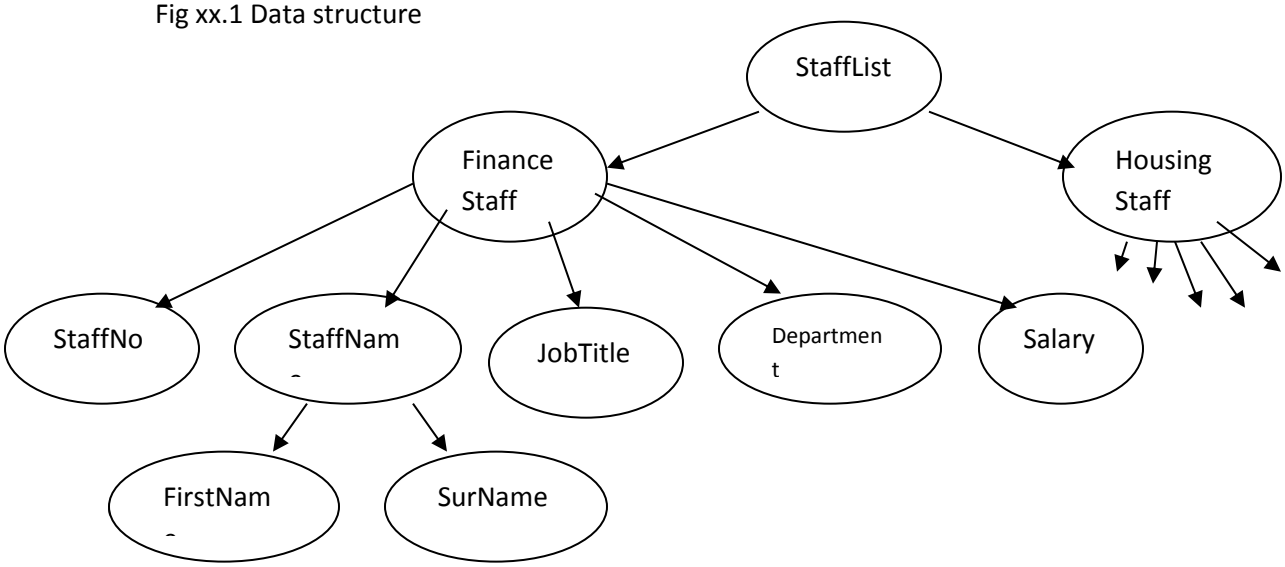
c) Represent the contents of Relation StaffList as an XML document using the data structure given in Fig A2[1] using a namespace called *StaffList* with a url = http://www.Idealhomestaff.co.uk/branch.

(8 marks)

Relation StaffList

StaffNo	FirstName	SurName	Department	Salary	JobTitle
123	Robert	Rogers	Housing	34300	Manager
126	Duncan	Boult	Finance	22000	Accountant

Fig xx.1 Data structure



ANSWER part a) a rudimentary knowledge of XML was required if present this was an easy question…. Document would look like this with namespace included (important)

<STAFFLIST xmlns=http://www.idealhome.co.uk/branch>
<STAFF deptno="FinanceStaff">
<NAME>
<FNAME>John</FNAME>
<NAME>
<JOBTITLE>Manager</JOBTITLE>
<SALARY>23212</SALARY>
---- etc
<STAFF deptno=<"personal">
<NAME> etc
Must have accurate syntax ie no trailing '<' or '>'

c) XQuery is a language used to query XML data.

Describe with the aid of examples using the XML document you produced in fig A2[1] how each of the following types of XQuery expressions are formulated

- path expressions,
- conditional expressions, **(10 marks)**

ANSWER POINTER part b)
Path expressions are used to get an ordered list of nodes including decendent nodes
Eg find the staff number of the first member of staff
Doc("stafflist.xml")/STAFFLIST/STAFF[1]//STAFFING
Conditional expressions Find if there are any staff listed (very simple example expected).

<Staff>
{ $b/Staffid }
{ $b/Staffname }
{
IF (empty($b))
THEN <status>inactive</status>
ELSE <status>active</status>
}
</user>

c) Explain the type of expression and the function of the following XQuery expression

```
{
LET $doc := document("stafflist.xml")
FOR $t IN distinct($doc/staff/salary)
LET $p := $doc/staff[jobtitle = $t]/salary
RETURN
<avgesalary jobtitle={ $t/text() }>
{
avge($p)
}
</avgesalary>
}
```

(7 marks)

ANSWER POINTER part c)
This Xquery is a FLWOR expression and uses variables to assign values in the document it resturns theb average salary for each job title listed in the xml file. The output is XML and a document that can be subsequently queried. Expected an explanation of each line of code to get full 7 marks.