THE BCS PROFESSIONAL EXAMINATION
Diploma

April 2000

EXAMINERS' REPORT

Systems Analysis

General Comments

Solutions are indicative; credit will be given for reasonable alternatives.
Exam technique is a very important aspect and as might be expected the weaker candidates showed poor exam technique as well as lack of knowledge. For example in question 3 writing as much or more for 6 marks as for 19. Also, writing all they know about a topic instead of answering the question; e.g. writing about CASE tools in general in answer to question 5 instead of discussing more specifically their contribution to quality.

Answer Pointers – Question 1

This question was in three parts dealing with, (a) data flow diagrams, (b) entity relationship modelling, and (c) entity life histories.

a. ***Produce a 'Top Level Logical Required Systems DFD' model to support the processing requirements outlined above.***

(20 Marks)

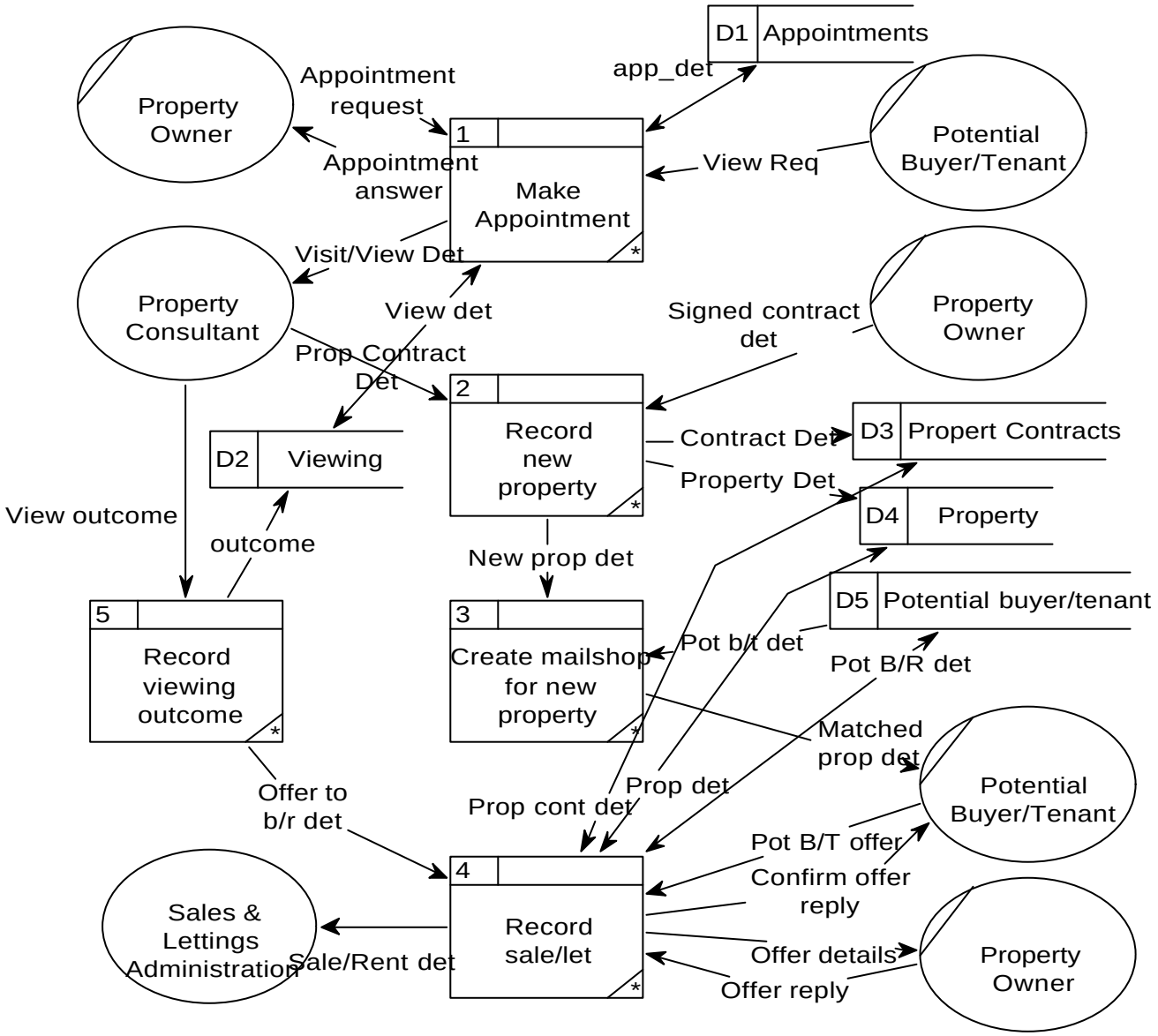
The diagrams presented represent only one possible interpretation of this scenario and as such other well constructed and comprehensive answers will be equally rewarded. Given the time available answers are likely to be less detailed than the solution provided.

Marks will be awarded for:

consistent and correct use of symbols (ie. Yourdon, SSADM, etc)	2
correct identification of external entities	3
completeness of processing	7
appropriate labelling of processes, stores and flows	2
logical not physical representation	1
clarity of diagrams	1
numbering of processes	1
conformity to DFD rules,	3
i.e.. all flows labelled	
no read/write only stores	
no direct relationship between sources/sinks	
flows to and from stores need not be labelled	
etc.	

To a maximum of 20 marks

Sample Answer



For the most part this question was answered at a level commensurate with the Diploma level and expectations. However, some key observations were made.

i. Many answers were too physical for a proposed logical model. For example, candidates frequently presented ‘view property’, ‘visit property to discuss requirements’, ‘sign agreement’, etc, as proposed logical processes, clearly this cannot be the case.
ii. Many candidates failed to label data flows and when they did they were sometimes far to physical. Other labelling errors noted consistently included, flows into and out of a process with the same name, labelling data flows with ‘active verb’ names such as ‘Search for buyer’, ‘Create new account’, etc.
iii. Some models were too complex with as many 9 or 10 processes for this relatively small scenario. Candidates that developed models such as this invariably included all the processes that mirrored the human activity systems almost without regard for ‘logicalising’ it.
iv. Processes were presented with no ‘trigger’ and as such they could never be started. Many processes were included (without explanatory notes) that had either no input, or no output.
v. Some candidates included a ‘Context Diagram’ instead of a ‘Top Level DFD’. Those that did this were awarded marks and where possible whilst remaining broadly true to the original marking scheme.

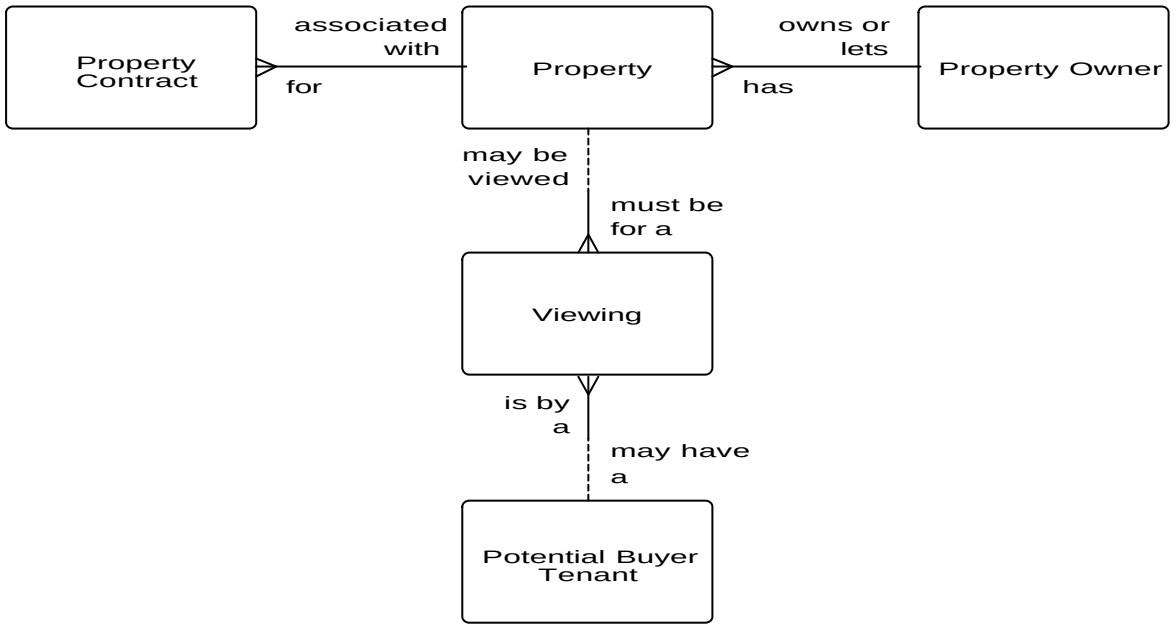
Candidates that presented models including some (or in some cases, most) of the above errors lost marks but were still able to attain a pass mark in this component if there answers included, consistent notation, numbered processing, identification of the overall processing requirements, etc. As Diploma candidates it was not expected that all the rules and technical DFD elements would be adhered to throughout and the scripts were marked accordingly.

b. ***Produce an Entity-Relationship diagram and skeleton tables (i.e., tables that include primary and foreign keys only) for the required system.***

(20 Marks)

1. It is not required that candidates identify and model 'specialised types' though answers that do will be welcome.
2. The ERD provided represent a core set of entities - candidates may identify others or use different names. All comprehensive and well 'engineered' answers will be appropriately rewarded.

Sample answer ERD.



Skeleton Tables are:

Property (Prop#,Owner#,.....),
Property Contract (PropCon#,Prop#.....),
Property Owner (Owner#,.....),
Viewing (Prop#,PotBT#,.....),
Potential Buyer Tenant (PotBT#,.....).

It is expected that candidates will be able to identify all the main entities and model them appropriately.

Marks will be awarded for:

consistent and correct use of notation	2
correct identification of entities	4
appropriate membership classes identified and documented.	3
appropriate degree of relationships identified and documented	3
Primary keys appropriate and assigned	4
Foreign keys appropriately posted	4

To a maximum of 20 marks

Most candidates presented reasonable solutions to this task, but some technical errors were consistently noted.

i. Some models were much to complex both in terms of the number of entities and the relationships between them. Some candidates included all aspects of the processing and presented entities such as, 'Property Consultant' (which is quite reasonable), to 'Letting Department of ME' (which is far less likely or necessary). Other frequently presented entities that were regarded as doubtful included 'Millennium Estates', 'Sales Division', 'Signed Agreement', etc. Where these were present they were often accompanied by very complex relationships in that far too many entities had direct relationships with many other entities. This in effect gave rise, in some cases, to a complex web of 1:1 relationships that would be difficult to justify had candidates tried.

ii. Most candidates presented the primary keys of tables effectively and appropriately, thereby achieving the full marks for this element. Dealing with foreign keys, however, was far more problematic. Few candidates were able to identify and allocate the full set of PK/FK relationships which was disappointing as, in effect, this can be achieved quite simply by learning and applying a few simple 'ERM engineering rules'.

iii. Some candidates included all table attributes even though only skeleton tables were asked for.

iv. Very few candidates included any stated assumptions that underpinned their ER models. This was seen as regrettable. Some stated assumptions may have given some insight into the reasoning process of the candidate and thereby enable marks to be awarded for modelling decisions taken that appeared questionable.

c. What are Entity Life Histories and how do they assist in the checking of the validity of the process (DFD) and data (ERD) models. Illustrate your answer with reference to the models created in answer to parts (a) and (b) of this question.

(10 Marks)

ELH's show how entities are affected by events over time. It's a third view to complement the process (DFD) and data (ERD) views. As they are essentially a third view of the same thing the reality being reflected in the ELH should match that found in the other two models. However, as ELHs provide an insight into processing from a time perspective, we can sometime discover data or functionality that is not present to support all the states an entity can be in over its life in a system. Once an ELH is produced it can be checked against the other two models to ensure that all the processing needed to change an entity from one state to another is present and that all the required data is also represented in the data model. ELH's give us a way of focusing on the 'exception' to normal processing and as such they are a valuable tools in the 'toolkit'.

Example from the given scenario may include:

ELH for a 'Visit' may show that there is no functionality or data to support or record a repeat visit if the first one was cancelled for some reason or other.

The ELH for a 'purchase' may highlight the problems of re-activating a sale property if the purchase does not proceed to completion.

An ELH for mailshots may show that the company 'flag' those for whom the property most closely matched their needs and calls them a week later to try to follow-up the sale. Given the discovery of this additional functionality extra data (ie. an new entity) and processing is needed.

However, most ELH's will provide additional confidence that our proposed systems does meet all requirements both in terms of processes and data.

Marks will be awarded for:

Definition of what an ELH is.	2
Explanation of its role in checking DFD with examples	4
Explanation of its role in checking DFDs with examples	4

To a maximum of 10 marks

This element of this question was, in most cases poorly answered. Very few candidates were able to present a clear definition and illustration from the models developed in Parts A and B of this question (as requested). Indeed, it was quite apparent that some candidates did not know what an ELH actually is and their answers were quickly seen to be based on (quite intelligent) guesswork.

Candidates that did know what ELH's are, frequently illustrated them by covering the life history of physical systems component/actor rather than its logical processing element. The life history of a client for example was frequently presented and included such 'events' as, 'rings in for an appointment', 'meet with property consultant', 'shows someone round the property', etc. I would rather have seen logical events such as, 'create client', 'assign property consultant to client', 'change client details', delete client', etc.

Disappointingly, only a very few candidates actually recognised the important contribution ELH's can make in identifying exceptional events not captured by data and or process modelling.

Finally, some candidates presented Entity Life Cycles and or State Transition Diagrams in place of ELH's. Whilst it is acknowledged that considerable similarities exist between these models, they are not the same. However, marks were awarded to candidates making this error when possible.

Answer Pointers – Question 2

(a) Identify the people other than development personnel who may be involved in an Information Systems Development project.

(8 marks)

People might include: Various user groups such as: top management, departmental heads, administrative staff; customers/clients; public at large.

2 marks each for each reasonable group of people identified to a maximum of 8.

(b) Explain the importance of involving user groups in an Information Systems Development project. Discuss, with reference to any appropriate methodologies, some of the ways in which their involvement can be achieved.

(17 marks)

Important issues include:

- Need to ensure that user requirements are fully understood; cannot be done without reference to user groups;
- Involvement of users in design (e.g. evaluation of prototypes) will ensure more usable system;
- Involvement will help users take ownership of, and responsibility for the success of, the new system

2 marks for each relevant issue identified, to a maximum of 6.

User groups might be involved via an Information Systems strategy group, or by membership of a Steering Committee for a particular project or set of projects.
Candidates are expected to refer to at least one participative methodology (e.g ETHICS or a RAD approach) and note that these methodologies actively involved users in design groups (JAD sessions, in the case of RAD).

Up to 2 marks each for mentioning/describing IS strategy group and Steering committee.
Up to 7 marks for discussion of user involvement in a participative methodology.

Total: 25 marks

The weaker candidates showed little evidence of knowledge of participative approaches to systems development. Quite a few talked about fact finding techniques in general (interviews, questionnaires etc.) and credit was given for this.

Answer Pointers – Question 3

(a) Explain briefly what is meant by a 'hard' and by a 'soft' approach to systems development, providing an example of a methodology of each type of approach.

(6 marks)

'Hard': e.g. SSADM. Sees systems development as technical problem; scientific, rational, reductionist approach.

'Soft': e.g. Soft Systems Methodology. Cannot divorce computer system from the people who will be using it and the organisation within which it is embedded.

1 mark for appropriate methodology identified, 2 for accurate explanation of each approach.

(b) Discuss the disadvantages of using only a 'hard' or soft' methodology and examine any proposals that have been made to overcome these drawbacks.

(19 marks)

Disadvantages of 'hard' only include:

- As business context not considered, may develop the 'wrong' system
- System may not be acceptable to the end users
- Assumes that requirements are clear and unambiguous
- Conflicts among user groups may not be addressed
- May lead to expensive 'maintenance' to fulfill unmet requirements

Disadvantages of 'soft' only include:

- (SSM) does not provide the tools to build working system; only helps clarify 'the problem situation'
- Does not cover the entire lifecycle; does not address technical issues of analysis, design and implementation
- Requires management support/cooperation

1-2 marks for each disadvantage identified/discussed, to a maximum of 14 (no more than 7 if disadvantages of only one approach are mentioned)

To overcome:
Multiview claims to incorporate elements of both hard and soft approaches, to provide a methodology that combines the strengths of each.
SSADM v4+ takes a more business-focussed view, suggesting the incorporation of the SSM conceptual model.

For these or any other reasonable suggestions as to how to incorporate aspects of both approaches: up to 5 marks.

Total: 25 marks

Often a confusion between the concept of a 'methodology' and a 'technique' (this applies to question 4, too). So quite a number of candidates gave Rich Pictures as an example of a 'soft' approach/methodology to systems development.

Those candidates who did not do well on this question mostly misunderstood the concept of a 'soft' approach to systems development.

Answer Pointers – Question 4

Write brief notes on each of the following:

(a) Traditional (waterfall) lifecycle (6 marks)

- These or similar stages may be identified: project initiation/selection, feasibility, analysis, design, implementation, maintenance and review.
- Stages follow in a linear fashion.
- Gave better control over the development process than previously.
- But: inflexible, over large project, there may be considerable requirements drift between analysis and implementation.

(b) Rapid Application Development (6 marks)

- Aims to deliver small pieces of functionality within short timescale.
- Incremental systems development.
- Control achieved by timeboxing.
- Iterative approach, with use of prototypes and high level of user involvement.
- Suitable when there are clearly identified user groups, visible user interface, system can be divided into suitably small subsystems for incremental delivery.

(c) Rich Pictures (6 marks)

- A technique used within Soft Systems Methodology.
- Notation may be described/illustrated: stick figures for groups of people, 'bubbles' for worries/concerns, crossed swords for conflicts, 'eye' for outside scrutiny etc.
- Useful for helping to identify key features of a problem situation including organisational structures, roles played, potential boundary of proposed system, primary tasks.
- Highlights worries, conflicts, political issues.

(d) Class Diagrams (as used in object-oriented approaches) (7 marks)

- Illustrate the object classes, with their attributes and operations, and the associations between them. Notation (e.g. UML notation) may be illustrated.
- Associations represent the potential for message-passing between object classes; multiplicity is shown; special types of association are aggregation and inheritance.
- First drawn during analysis stage, to identify key object classes from the application domain.
- Enhanced during design, when additional detail (in particular, operations) is added.

Total: 25 marks

This was, as might have been expected, the most popular question.
Rich Pictures were not understood by a sizeable number of candidates.
They were also weak on Class Diagrams; clearly most candidates have studied/use structured methods and so do not appreciate the very different paradigm of OO approaches to development.

Answer Pointers – Question 5

Explain the ways in which the quality of a software system can be assured during the systems development process. Describe the contribution that Computer Aided Software Engineering (CASE) can make towards software quality.

(25 marks)

'Quality': often defined as fitness for purpose; a quality software system is one that will meet the user requirements, with the required levels of reliability, usability etc. This can be assured by means that include:

- Use of a methodology
  To ensure a standard process; to define the phases through which a systems development project should pass, etc.
- Communication with users
  Essential to ensure that requirements are correctly determined and modelled; that feedback is received and incorporated on proposed designs, etc.
- Quality check points
  SSADM, for example, provided quality criteria for each product and also formal quality checkpoints when products should be signed off with the customer.
- Walkthroughs
  Peer review of products, to ensure their quality
  Procedure for conduct of walkthrough may be described.

Up to 15 marks for the identification of these or other appropriate means by which to assure quality.

CASE tools:

- Help enforce methodology
- Standard notation
- Standard documentation
- Consistency/completeness checks
- Amending/updating achieved more easily
- Can produce/evaluate alternative designs
- Capture of data for re-use
- Clearer communication with user
- Can provide data to support the metrics function, thus providing a quantitative measure of quality.

Up to 10 marks for discussion of contribution of CASE tools.

Total: 25 marks

The weaker candidates tended to ignore the first part of the question (worth 15 marks) and only talk about CASE tools and their contribution to quality.

Statistics on Questions 2-5

No. of candidates choosing each question:

2	3	4	5
91	71	130	87

No. (%) of candidates achieving a pass mark (10/25) for each question:

2	3	4	5
77 (85)	36 (51)	96 (74)	48 (55)