The Derivation, Implementation and Evaluation

of a Model for

CBL Specification and Design.

Geneen Elizabeth Stubbs

A submission presented in partial fulfilment of the requirements of the

University of Glamorgan/Prifysgol Morgannwg

for the degree of Doctor of Philosophy

January 2001

Abstract

This thesis details the derivation and implementation of a Computer Based Learning (CBL) design and development model UDRIPS, UDRIPS stands for Universal picture; Definitions; Rules; Illustrative examples; Problem solving; Summary. The need for the model was highlighted as a result of involvement in the Teaching and Learning Technology Programme, Phase Two (TLTP-2) project W.I.S.D.E.N. (Wide-ranging Integrated Software Design Education Network). Courseware produced within the consortium covered several topics within the area of software design and development. The model was constructed by combining principles from the software engineering and pedagogic areas. This allowed courseware to be built which adhered to basic software engineering principles but which was also pedagogically valid.

The model was designed to be an addendum to existing CBL development methods and is intended primarily to enhance the instructional design phases of those methods. This can be seen to be a mirror of the types of techniques enjoyed by software engineers and which are inherent in the software design and development methods in that field.

To test the model, CBL courseware was developed in a topic of Structured Methods, Entity-Relationship Modelling. This courseware formed the basis for a number of usability and learning effectiveness evaluations. Based on the results of the evaluations and further research both the material and the model were refined.

UDRIPS, the refined CBL design and development model, was distributed to the members of the consortium involved in the W.I.S.D.E.N. project and their reactions and use of the model observed and noted. The courseware produced by the members was also evaluated with respect to its usability and effectiveness.

Acknowledgements

Writing this thesis has been a long journey. I'd like to thank Dr. Paul Beynon-Davies for acting as my guide and enabling me to complete my journey.

I'd particularly like to thank Mike Watkins for allowing me to undertake that journey and discover myself.

I would also like to acknowledge the part played by the W.I.S.D.E.N. project, the consortium members and the Teaching and Learning Technology Programme (TLTP) initiative in providing the opportunity to engage in this research.

Last, and by no means least, I'd like to thank Terry for his unwavering support and belief in me.

Contents

10   Further Evaluations

11   Conclusions and Future Work

Figures

9   Refinement of the Development Method

10   Further Evaluations

11   Conclusions and Future Work

Certificate of Research

This is to certify that, except where specific reference is made, the work described in this thesis is the result of the candidate. Neither this thesis, nor any part of it, has been presented, or is currently submitted, in candidature for any degree at any other University.

Signed   *G. E. Stubbs*

                   Candidate

Signed   *M Watkins*

                   Director of Studies

Date   01/07/2001

Chapter 1

Introduction

1.1 Introduction

This chapter will give an overview of the contents and structure of the thesis together with the research undertaken and methodology employed in this project. It will outline briefly the topics covered in each of the chapters. More detail will be found in the body of the report.

The report will explain the background to the research which resulted in the creation of a structured Computer Based Learning (CBL) design and development model UDRIPS, UDRIPS stands for Universal picture; Definitions; Rules; Illustrative examples; Problem solving; Summary. The research was carried out as a result of involvement in the W.I.S.D.E.N. project. W.I.S.D.E.N. was a project which aimed to produce CBL material and from which a need for a structured CBL development model emerged. The CBL development model UDRIPS was devised to meet a need for a systematic, structured method for producing CBL material.

1.2 Objectives of the Research

UDRIPS aims to emulate structured methods which are available for conventional software development, in that: it is structured; it is implementation independent; it enables the development of a "quality" product in that it meets user requirements.

The objectives of the research are to:

- Adopt a software engineering approach to CBL development;
  - o Structured;
  - o Reproducible;
  - o Able to be taught;
  - o Implementation independent;
  - o Facilitates user-developer, developer-developer communication;

- Combine software engineering principles with pedagogic principles to produce a CBL model, UDRIPS, that provides a structured approach to CBL development that is pedagogically valid;
- Investigate the amalgamation of the CBL design and development model with existing CBL development methods;
- Use the CBL design and development model to underpin the development of a tutorial type CBL prototype which teaches E-R Modelling;
- Formatively evaluate the prototype;
- Refine the UDRIPS model based on the results of the evaluation and further research to form an enhanced CBL system to teach E-R Modelling;
- Evaluate the usability and effectiveness of that CBL system with student users.

Each of the following sections summarises the content in the chapters of this thesis, it is structured so that the sequence of the sections mirrors the sequence of the chapters.

1.3 Research Methodology Adopted

This project has adopted the Action Research Methodology in its approach to the research activities (McNiff et al, 1996, Cornford & Smithson, 1996). This methodology has, at its heart, the paradigm of the researcher becoming an active participant in the work undertaken during the course of the project. The research is derived from two key points, "firstly the researcher uses their theoretical knowledge to shape the activity they participate in. Secondly, through their reflection on this experience, they can relate events to prior theoretical knowledge." (ibid.). In this approach, the research is undertaken as a consequence of participation in a "real world" activity. This gives the added benefit of observing, directly, the participants of the project as they contribute to the activities of that project. The research came about as a result of involvement in the W.I.S.D.E.N. project.

1.4 W.I.S.D.E.N.

The research which this thesis details resulted from the involvement in the W.I.S.D.E.N. project. W.I.S.D.E.N. (Wide-ranging Integrated Software Design Education Network) was a three year project funded under the TLTP-2 (Teaching and Learning Technology Programme, phase Two) initiative. TLTP was launched in 1992 by the Universities Funding Council (UFC) and continued until 1996. Over the two phases of the TLTP initiative 76 projects were supported. Subjects covered over the two phases were very

diverse ranging from Mathematics, Physics and Chemistry to Language Learning, Study Skills and Behavioural Sciences.

The main aim of the W.I.S.D.E.N. project was the development of interactive computer-based learning materials in the main branches of software systems analysis and design, i.e., in Structured Methods, Object Orientation, Formal Methods and Real Time Systems.

The original consortium was formed under the leadership of Professor Allan Norcliffe from Sheffield Hallam University and comprised the computing departments (or similar) from each of the following universities:- Brighton, Glamorgan, Heriot-Watt, Loughborough, Sheffield Hallam, South Bank, and Teesside. Each department provided one subject matter expert, usually an academic, and one developer, a research assistant. The role of the subject matter experts was to structure and detail each of the topics and the role of the research assistant was to produce the Computer Based Learning systems for each of the topics identified. Within the W.I.S.D.E.N. consortium, the various subject experts had backgrounds in the computing fields that covered the topics of the project, this was also true of some, but not all, of the developers.

The University of Glamorgan worked in partnership with South Bank University in the area of structured methods. Glamorgan was assigned the topics: entity relationship diagrams; relational data analysis; and entity life histories / event analysis. The remaining consortium members were allocated the rest of the topics within the analysis and design field, covering object oriented methods, formal methods and real time systems.

Experience in the software design area shows that there are many methods and techniques available to assist developers in producing many different types of software systems. Within the CBL area, whilst there are several high level development methods, they have few, if any, associated techniques to assist the process. Thus, CBL developers often produce systems using ad-hoc approaches which show little evidence of consistency. This is, perhaps, not surprising as many CBL developers are not familiar with the software engineering ethos. Their area of expertise lies outside the field of Computing which appears to explain the lack of standardised CBL development techniques available. Examination of the TLTP projects in both phase I and II shows that those involved with the projects rarely have computing backgrounds (http://www.ncteam.ac.uk/tltp/catalogue/), further details of the W.I.S.D.E.N. project may be found in chapter 2.

1.5 Software Engineering and CBL Development

Software Engineering methods are geared towards building software systems in an orderly, systematic manner (Wood et al, 1988). The various methods adopt a variety of approaches to the task of software development. Some take a top down approach such as data flow diagramming or entity-relationship modelling whilst others take a bottom up approach such as normalisation. Yet others decompose systems into objects and classes. Whilst, existing software engineering methods provide tools and techniques to assist the developer to produce quality products, those products do not have a requirement that they teach, nor facilitate learning, for a user. It is this didactic feature of CBL systems that differentiates them from conventional software systems.

The aim of a final CBL system is to aid the user in meeting the objectives for a topic set by a tutor or tutors and as such part of its quality is judged by the success of learners at meeting those objectives. This success may be judged by various assessment means, such as quizzes, tests or exercises within the CBL or, after using the CBL, in more traditional assessments such as exams or assignments.

However, despite the didactic element of the CBL, it cannot be overlooked that the final system is to be delivered via a computer and, as such, is still a software system. Thus, just as in conventional software systems, the CBL developer must address the computer environment to ensure that the final product will run efficiently and effectively and meet its aim As such, the same rigour must be applied to the design and development of a CBL system as its conventional software counterpart. Elements of a software system that are addressed during the development of conventional systems include the interface and the environment. These aspects can impact on the use of CBL systems as much as they do in any software system since the user must be able to access and view the CBL material to be able to learn from it. A learner must be able to utilise the CBL software to learn and some of the factors that may affect this include: its structure; the navigation facilities available; the elements that are included in the CBL such as problem solving opportunities or real-life examples. It is necessary, therefore, that any CBL method should address pedagogic principles to facilitate the didactic element of this particular type of software system but must also adhere to software engineering principles which apply to any other software system.

1.5.1 CBL Development

Initially within the project, the research focussed on CBL development and the methods available to facilitate this. Several methods were examined and found to be lacking from the low level point of view of constructing an individual learning activity such as a CBL "lesson". Within W.I.S.D.E.N., CBL lessons were designed to cover sub-topics which had been produced by decomposing the main topic into its low level objectives. These low level objectives could then be chunked by grouping together related objectives to form lessons. The decomposition was carried out by the subject experts and the chunking or grouping done by the developer in conjunction with the subject expert. This decomposition was instigated by the subject experts within the consortium and was deemed a necessary activity for any teaching/learning activity from their experience of scheme development and their academic background in teaching/learning.

The decomposition of topics into objectives was a feature in many of the CBL methods found and is consistently a major activity in any CBL project. However, whilst grouping of the objectives can determine the contents of a lesson, just as in a conventional teaching/learning situation, providing information alone does not constitute a complete learning experience. Several CBL methods have additional phases that refer to the inclusion of media and assessment items, these phases within the models are advisory rather than explicit in that they do not advise which media to use in particular situations but merely which media could be included e.g. still pictures, audio etc. (Gerlach and Ely, 1980).

What is missing, from the point of view of the developer, is the essence or core of a CBL lesson i.e. what elements make a good CBL lesson to satisfy both the tutor and the learner. In a CBL development project, the tutor often knows little about what a computerised CBL lesson can provide and, thus, can offer little advice on its structure. At the same time, the developer may know very little of the activities that can facilitate learning. This mismatch can be alleviated by providing a template for a CBL lesson which incorporates elements from the pedagogic theories combined with software engineering principles so that (s)he may provide lessons which are valid from the teaching/learning point of view.

The CBL method devised as a result of this project is intended to provide a developer with a template for CBL lessons which take into account pedagogic principles whilst at the same time providing a sound software engineering approach. The aim is for this explicit

approach to underpin the phases of existing CBL development methods without detracting from what they, too, can offer.

Despite, the apparently prescriptive nature of the template, it is sufficiently flexible to allow for the inclusion of additional features, such as media elements to enhance the CBL lessons without destroying the integrity of the template. The template, also provides a communication tool between the developer and the tutor to ensure that those elements that are deemed beneficial for learning are provided and included within the CBL lesson. The developer is able to acquire content for the lesson by using the sections of the template as a guide to what to request from the tutor.

1.5.2 Software Engineering

The aim of a software developer is to analyse, decompose and understand system characteristics, to ensure that software systems are built to meet end-user needs and requirements. Similarly, with CBL development, the developer may have little experience in the educational field and, therefore, needs to analyse and decompose the topic to ensure that the final system meets end user needs and requirements. This is both from the point of view of a tutor or lecturer wishing to teach the subject and from the point of view of a learner attempting to learn from the system. To do this the developer needs the assistance of the tutor who is the subject expert, (s)he needs to provide the developer with the objectives of the topic that are to form the basis of the CBL "lesson(s)". The developer must take the list of objectives and from these build a useful teaching/learning device, to satisfy both tutor and learner needs.

The main software engineering methods were investigated to determine what they provided for the software developer. The areas of software analysis and design provide a number of methods to cover the Process (Function) Oriented (Yourdon & Constantine, 1979), the Data Oriented (Jackson, 1983) and the Object Oriented (Booch, 1991, Coad & Yourdon, 1991) Paradigms, together with an assortment of techniques such as E-R Modelling, Data Flow Diagramming etc. Their aims are to facilitate the analysis of existing systems to uncover the components of the system under investigation and from these "building blocks" to devise computerised solutions. The items uncovered are represented in a number of ways e.g. entities, objects or functions and these in turn are defined in terms of their attributes and relationships, methods and processes. The key for any method is to provide the developer with an analysis mechanism for the particular system under development and, for each of the systems, the means to represent those core

elements in a particular way. Details of how the core elements are represented also include what is and is not allowed. In summary, the generic features that are provided by the software engineering methods are the ability to determine the legitimate and illegitimate states of the core elements of the system under investigation together with their particular characteristics.

In the CBL area, the key features for the teaching/learning of a particular topic are represented by the objectives set by the tutor. Grouping those objectives gives the sub-topics and hence the CBL lessons for those sub-topics. Each lesson will address one or more of the objectives identified for its particular sub-topic. The decomposition process reveals the structure of the topic which in turn clarifies the dependencies in the lesson hierarchy i.e. if a lesson is dependent on knowledge from any others. This dependency can determine the most suitable path through a topic area by showing which lesson should be undertaken prior to any others. This reflects the determinacy and dependence found in normalisation where "if data-item A is the determinant data-item and B the dependent data-item then the direction of the association is from A to B and not vice versa" (Beynon-Davies, 1998).

Hence, there is a direct correlation between the analysis carried out by the software developer and that carried out by the CBL developer to ensure, just as in conventional software, that the final CBL system represents the "things of interest" identified from that analysis. The "things of interest" need to be defined and their legitimate and illegitimate uses identified. This aspect of analysis forms part of the CBL method devised and correlates closely with the generic features of software engineering principles identified above.

To complete the CBL method, it needs to be combined with the principles derived from the pedagogic theories to provide a comprehensive approach to CBL development. This combination provides a tool for the developer which addresses both key areas which is especially useful when (s)he is inexperienced either in producing educational software or software in general. This means that the method may be used by both developers and by tutors who wish to develop material but who have little experience of developing any form of software. In recent years the emphasis in CBL development has moved away from small individual projects to larger often distributed consortium based projects, initiating an additional difficulty of the communication between project members. A systematic CBL method provides a mechanism for a common approach for all members whilst at the same

time ensuring that individual creativity is not stifled. The examination of software engineering and CBL methods may be found in chapter 4.

1.6 Pedagogy, the Science of Teaching

The primary aim of CBL material is to facilitate learning. Learning may be defined in a number of ways, such as: "the knowledge acquired by study" (Oxford Encyclopaedic English Dictionary, 1991) and "changes in the behaviour of human beings and in their capabilities for particular behaviours following their experience within certain identifiable situations" (Gagné et al, 1992). In producing CBL material, the developer is seeking to bring about a recognisable change in the knowledge and/or behaviour of a learner which may be demonstrated to satisfy some external criteria such as passing an examination or completing an assessment exercise or successfully completing a task. Learning, however, is a complex process and is reliant on many variables such as motivation, learning style and prior experience or knowledge. The pedagogic theories attempt to address these issues in order to influence learning in a positive manner. Pedagogy is defined as "the science of teaching" and didactic as "meant to instruct", (Oxford Encyclopaedic English Dictionary, 1991).

Areas such as motivation and learning style have been widely researched, however, researchers have come to no conclusion. Learners are individual and the factors influencing them diverse. This means that learning theories invariably provide guidelines rather than rules; these guidelines often appearing tenuous to the practitioners attempting to provide a practical solution.

In this research project, an investigation into the pedagogic theories was carried out in order to ascertain the principles enshrined in those theories. The aim was to combine these principles with those from the software engineering field to provide a hybrid CBL development model which, whilst prescriptive in its nature, was meant to be used to aid in the production of CBL lessons as a foundation rather than as a complete structure. The benefit of this type of model is that it may be adjusted or amended in a specific topic area and learning situation at the discretion of the developer and/or the tutor but which forms a basis for the CBL development. This corresponds well to the software engineering models which, whilst prescriptive, are sufficiently flexible to ensure the creativity of the developer(s) is not stifled. Many recent software development methods are far more contingent than they once were, they offer advice on procedures rather than rules of

application e.g. DSDM (Dynamic Systems Development Method), (DSDM Consortium, 1995).

As mentioned previously, no one pedagogic theory purports to provide the answer to learning. What they provide is clues to the factors that affect learning and in some cases advice on how to positively influence learning. Early work in the area of learning concentrated on the behaviourist approach which investigated how learning was influenced by stimuli, responses to those stimuli and the reinforcement of behaviour, the focus of this work was on how to change or reinforce the behaviour of the learner, (Pavlov, 1927, Thorndike, 1913, Skinner, 1938). This early work was influential for many years and was the basis for the positive reinforcement epitomised by rewards for good behaviour such as increased responsibility and the negative reinforcement characterised by punishments such as detention or caning found in many classrooms in years gone by. An important contribution to the learning experience that emerged from this theory was the emphasis on objectives which provide a detailed account of what is to be taught to the learner and a derivation of learning outcomes based on objectives to demonstrate competence by the learner. This provision of objectives not only allows the learner to have a clear outline of the topic under tuition but it also allows the developer to delineate lessons and judge the size of the project (s)he has undertaken.

Another important feature elicited from behaviourism is the concept of the provision of feedback to reinforce correct responses or alter or explain incorrect ones. This is especially important in the assessment activities of the learner to assist in the learning process. Feedback allows the learner to see quickly which parts of the topic they have mastered and which parts require more work. For example, it can provide clues to where the learner went wrong or recommendations on which part of the topic needs further work. Cognitive psychology arose as a result of the dissatisfaction of some psychologists with the behaviourist approach (Ausubel, 1978, Bruner, 1966). Many felt that Behaviourism did not fully explain the learning process as it did not address the full range of human behaviour e.g. elements such as memory and thought. Emphasis in this field moved towards the study of memory, attention, perception, language, reasoning, problem solving and creativity. Learning was seen as an active process where the learner actively tries to understand the environment and increase knowledge. A key facet of learning in this theory is the link between existing and acquired knowledge and how that is stored in memory, a student may only acquire new knowledge when they are ready to do so, i.e., when they have the correct prior knowledge to be able to proceed.

9

Another characteristic identified as important in this theory is that of problem solving and being able to actively apply any new knowledge to acquire experience in diverse situations. This problem solving activity can, also, be linked to assessment to judge ability and achievement levels within a topic. Problem solving allows learners to generalise their knowledge to a wide area rather than assuming the knowledge is specific to a single area. More recently work in the area of learning has moved to encompass the unique nature of learners and how they individually represent knowledge. Constructivism is an approach which advocates the "belief that reason is the primary source of knowledge and that reality is constructed rather than discovered". The key assumptions of this theory are that knowledge is acquired through experience, learning is a personal interpretation of that knowledge and that learning must be an active process in which meaning is derived from experience (Smith & Ragan, 1999). A key contribution of this theory is the idea of placing learning in context by providing problem solving opportunities that are relevant to real life, and also in providing meaningful examples to illustrate the relevance of the knowledge to the learner. Chapter 5 describes the Investigation into the Pedagogic Theories more fully.

1.7 UDRIP

The principles identified from the software engineering field were combined with those principles advocated by the pedagogic theories to form a hybrid CBL development model, UDRIP. The pedagogic principles included determination of pre-requisite knowledge; the setting of objectives; the inclusion of problem solving opportunities and the provision of illustrative examples to link the learning with relevant experience.
Identification of pre-requisite knowledge allows the learner to judge if they are prepared for the forthcoming learning experience. This is useful for both the tutor and the learner to ensure that the learning outcome is a successful one. That is not to say that a learner should be prohibited from browsing a lesson even if they were ill-prepared at that time. It is often useful to have an overview of what the topic is about to encourage further interest. The setting of objectives provides a goal for the learner and through related learning outcomes a means to judge success, this is, again, useful for both learner and tutor. From the tutor's point of view it can provide clues to the areas where remedial work is needed and from the learner's point of view, it can identify areas of weakness where more work is required. Learning outcomes can form the basis of an assessment exercise which can be provided within the CBL or away from the computer altogether.

Problem solving opportunities allow the learner to generalise the knowledge gained from the learning experience by providing varied exercises or questions to extend understanding. Feedback at this time is essential to allow the learner to consolidate their understanding by acknowledging correct responses whilst clarifying any incorrect response through an explanation of the rationale.

Finally, illustrative examples allow the learner to establish a thorough understanding of a concept through its relationship to their "real world" experience. Metaphors and similes, together with examples, case studies and relevant scenarios can extend this to provide connections from the abstract concept to the concrete reality.

Software engineering principles included the need to define and characterise the key elements in any lesson. The key elements relate to the objectives set for that particular topic and their characteristics include the details of how they may be applied or used. These principles were combined to form the UDRIP CBL development model which forms a framework, backbone or skeleton for CBL lessons. We hypothesise that UDRIP provides a prescriptive, structured model for CBL development which enables the developer to build CBL lessons that are flexible enough to be tailored to a variety of topics and situations and which does not stifle the developer's creativity with respect to screen design, media elements and environment.

Evidence from the development of CBL material within W.I.S.D.E.N. also suggests that the benefits of using UDRIP are that the developer does not need to have a deep understanding of the pedagogic theories to be able to build valid CBL lessons (Stubbs et al, 1995a, 1995b). Also, developers may use the model to request content from the tutor to populate the lessons with relevant material, exercises, examples etc. The prescriptive nature of UDRIP is epitomised by the provision of specific sections within the model which may be taken directly to form the structure of the CBL lessons or which may be customised to suit a particular learning situation. Whilst it is anticipated that the model is used as is, it should also be noted that all sections of the model should be considered even in situations where the developer decides to omit certain sections. UDRIP acts as a flexible skeleton for individual CBL lessons which has a contingent approach to the structure of the lessons. UDRIP is the topic of chapter 6. The application of UDRIP to the topic of E-R Modelling is detailed in chapter 7, a usability evaluation of the material developed using UDRIP in chapter 8 and the model's refinement to UDRIPS outlined in chapter 9.

1.8 Application of UDRIP to the CBL Development

As mentioned in section 1.2, Glamorgan was part of the Structured Methods group within W.I.S.D.E.N. and were assigned the area of E-R Modelling as the topic for the CBL material. Work began to develop CBL lessons in this area, using UDRIP as the framework for the content and structure of those lessons. Eventually, after conducting a task analysis and grouping the low level objectives five lessons were produced to teach E-R Modelling. These lessons formed the basis of the evaluation exercises conducted both formatively, during development, and summatively, once the CBL material was complete and ready for use in its intended manner.

UDRIP formed the sections of the lessons produced so that each lesson consisted of at least four separate sections each of which had a number of screens containing the content. These sections were:

1. Objectives and Pre-requisites (U – Universal Picture);
2. Definitions of keywords and/or concepts (D - Definitions);
3. Usage of the keywords and/or concepts (R - Rules);
4. Exercises, Quizzes, Questions (P – Problem Solving).

The illustrative examples were used in sections 2 and 3 to try to put the learning into context and relate abstract concepts to real life situations. The problem solving section also provided feedback to the user to try and identify why some answers were incorrect and, where relevant, how to correct them.

Learners could move sequentially between sections 1-4 or move directly to an individual section such as problem solving. This dual navigation facility allowed users to utilise the CBL material in a variety of situations. For example, users could move screen by screen through the material learning the topic in a systematic manner. They could also use the material for revision by moving directly to the relevant section to re-enforce or refresh their knowledge. They could also test themselves by moving directly to the problem-solving section to identify the strengths and weaknesses of their knowledge.

Using UDRIP to construct CBL lessons does not mean that two lessons developed by two separate individuals will be identical. This corresponds closely to the situation in conventional software development where no two software engineers will produce exactly the same data flow diagrams for the same system. There is still an element of individuality in such design depending on the experience and expertise of the software developer. Thus, it cannot be assumed that all lessons developed using UDRIP will be identical. Some may

include additional multimedia elements such as audio and video whilst others may have different interfaces or navigation structures depending on the topic to be taught, the needs of the audience and the conditions under which the learning takes place.

1.9 Preliminary Usability Evaluation

Just as in any learning situation, computer or classroom based, where the efficacy of the experience is tested by assessing the learner with respect to some pre-determined objectives and learning outcomes, the application of UDRIP to CBL development needed to be tested to ensure its application produces products which are valid and useful. The aim of producing CBL material is to promote learning for an individual. Therefore, application of UDRIP to CBL development needs to conform to this goal of facilitating learning for those using the CBL lessons to learn a topic. To this end, the CBL material needs to be evaluated with respect to both its usability and also with respect to its effectiveness as a learning aid. The usability of the material is important as it can have an effect on its effectiveness since if the material is of the very best quality but the learner cannot access it correctly then they will be unable to have a successful learning experience. Initially, evaluation focused on the usability of the software and was conducted as a formative exercise to improve any difficulties encountered by the learners (Stubbs et al, 1996a, 1996b). Formative evaluation is conducted to help form the system, sometimes to test out new ideas such as a new design, sometimes to check that existing features work effectively e.g. that the navigation buttons move the learner through the CBL material in the predicted manner. This early evaluation helped to pave the way for further evaluations to test the learning effectiveness of the lessons. The evaluation took place in a computer laboratory with a small group of learners, their comments, use of the system and completed questionnaires formed the basis of the evaluation and the justification for subsequent modifications to the CBL material. Details of the evaluation methodology adopted together with the results from this evaluation exercise may be found in chapter 8.

1.10 Refinement of the Development Method

During the course of the design and application of UDRIP, research continued into the pedagogic theories. Research into learning styles was, initially, undertaken to ensure that as many types of learners as possible were considered during the construction of the CBL material from the point of navigation and interface design. An early finding indicated that learners might fall into two main categories, sequential learners who would need a

sequential, linear type navigation strategy and holistic learners who would need a direct or browsing type navigation facility (Pask, 1976, Clarke, 1989). These findings influenced the structure of the template used within the W.I.S.D.E.N. consortium.

As research continued, development progressed and formative evaluation began, deeper investigation into the area of learning styles uncovered a plethora of theories dealing with the way people prefer to learn. A particularly influential theory came from David Kolb and could be seen as underpinning several of the other theories. This theory related to the various stages of learning from "having an experience" through "reviewing the experience" to "concluding from the experience" finally to "planning the next steps" and subsequently back to having another experience. The stages of this theory were depicted as cyclic but it was emphasised that some learners preferred some stages to others. Which stage the learner preferred characterised their learning style.

Based on this theory, Honey and Mumford extended this work to identify four types of learner: the activist; the reflector; the theorist and the pragmatist. The individual learning styles did not mean learners never went through the other stages of the learning cycle, merely, that they preferred some stages over others. It became clear that whilst the UDRIP model produced CBL material that catered for the activist, the theorist and the pragmatist there was little provision for the reflector (Honey & Mumford, 1992) though the sequential navigation facility did allow learners to revisit material within the lesson at any time. This was borne out by research into the work of Donald Schön, (Schön, 1987), who investigated and propounded the theory of reflection in action, reflection on action etc. and more recently the work of Phil Race with his "wanting; doing; digesting and feedback" model (Race, 1994) where he advocates time to "digest" what has been learned.

To accommodate this type of learner within any CBL material developed within W.I.S.D.E.N., a further section was added to the UDRIP model to ensure the reflective learner had a chance to review what they had just learned within a CBL lesson. This reflective section took the form of a summary which highlighted the objectives that had just been addressed within the package. It became the last section of the model and was actually the last section of the CBL material to encourage an element of reflection before the learner moved on to new material. Thus UDRIP evolved into UDRIPS and the evaluation exercises continued with the material being adapted to include this additional section in all of the lessons on E-R Modelling (Stubbs et al, 1996c, 1997). UDRIPS, and its evolution from UDRIP, is the focus of chapter 9.

1.11 Usability and Learning Effectiveness Evaluations

Further evaluations were conducted on the revised CBL material to ensure both the usability and learning effectiveness for the learners. Usability was again examined using learners' comments, use of the system and completed questionnaires, whilst, learning effectiveness was assessed using pre and post-tests. The pre and post-tests were multiple choice question tests where the questions were designed to correspond to the objectives set by the tutor for the subject.

The aim in this evaluation was not only to test the knowledge of the learner but also to test the improvement made by the learner from the pre-test to the post-test; (another feature of the tests was the inclusion of "confidence testers"). A criticism of multiple choice questions is that learners may guess correct answers rather than choosing them based on their knowledge of the subject. It was felt, however, that someone who guessed an answer would be less confident that it was correct than someone who believed they knew the answer. To test this hypothesis, confidence testers were included in the tests so that even if the results from the pre-tests were the same as those for the post-tests, provided the confidence of the learner had increased the conclusion that could be drawn was that the result was now based on knowledge rather than guessing and could be seen as a positive improvement in their learning. A description of the evaluation methodology and results from both the usability evaluations and the learning effectiveness evaluations can be seen in chapter 10.

1.12 Conclusion

The UDRIPS model was used at Glamorgan to produce CBL material to teach all three areas of Structured Methods assigned to the group i.e. E-R Modelling; Normalisation and Entity Life Histories. The most complete of these and the material that underpinned the evaluation exercises was E-R Modelling. Preliminary evaluations showed favourable attitudes to the CBL material by developers, students and subject experts. To this end, work began to produce CBL material throughout the consortium that conformed to the UDRIPS model. It is important to stress that UDRIPS was not imposed on the developers but offered as a useful tool. UDRIPS was adopted as a standard, by consensus, and used to aid in the development of the W.I.S.D.E.N. material (Norcliffe, 1996).

It is evident from this research that developing CBL material in this particular subject domain, using the UDRIPS development model, works well. Evidence also suggests that

many learners find CBL material very useful during the learning process. Comments on the use of the CBL suggest that it is particularly welcomed by students who lack confidence in the topic. This may be explained by the fact that CBL lessons allow the learners to progress at their own pace and to study at a time that suits them; the CBL was also perceived as non-judgemental allowing learners to go over material as often as they wished which they felt contrasted markedly with some other learning situations such as lectures or tutorials. Whilst feedback from learners was positive, the overwhelming agreement from the findings indicate that learners would prefer CBL systems to be used in conjunction with existing teaching/learning methods rather than as a replacement for them. Evidence from the evaluations indicates that students increased, not only their knowledge of the subject during the use of the CBL, but also their confidence in their answers from the pre-test evaluation results to the post-test results.

This research has revealed that within the W.I.S.D.E.N. project the application of UDRIPS to the design and development of CBL material has had a positive effect on the quality of the lessons produced. Quality is defined here as the ability to facilitate learning for the user, this is demonstrated by the evaluation results. UDRIPS was also perceived as beneficial by the developers in the structuring of CBL lessons, this affected time taken to develop the material. However, these findings must be verified and extended by further work.

To consolidate the findings from this project, it is necessary to evaluate two main areas, the process using UDRIPS rather than product produced and the domain of application for which UDRIPS is suited. Evaluation of the process will require that UDRIPS is adopted by a variety of developers to test its usefulness in the design and development of various CBL material. For novice developers, UDRIPS has proven invaluable in reducing the time taken to produce CBL material since it obviates the need to research learning theories for those with little experience in the area, whilst at the same time providing a clear structure to start the development process. It also enables novice developers to elicit the required content from the tutor since it shows what is needed in individual CBL lessons. This involvement of the tutor helps to form a rapport between the two parties, the developer and the end user, which has been shown to be advantageous in areas such as Rapid Application Development (Martin, 1991) and prototyping in conventional software development.

More experienced developers may well find UDRIPS an excellent starting point in any CBL development project since it does not restrict their creativity in developing the material but does give a logical structure with which they can work. It may also prove

beneficial in large group-based projects which require developer-developer and developer-tutor communication since it provides a framework for the content.

Secondly, UDRIPS will need to be used to produce material in various topic domains and its applicability determined within those domains. Work within W.I.S.D.E.N. has focussed on an area of computing, software systems design. This area appears to have benefited from the application of a CBL design and development model but these results need to be more formally scrutinised to ensure they are correct. Additional domains also need to be explored, firstly within the computing area, and subsequently outside the computing area to determine the overall applicability of UDRIPS.

Obviously, extensive evaluations would need to be conducted to determine the overall effectiveness of UDRIPS, the areas where it would be most suited and the developers for whom it was useful. Early indications are favourable, however, and UDRIPS appears to have had a beneficial effect on both developers and the CBL users. The conclusions from this research and future work are explored more fully in chapter 11.

Chapter 2

W.I.S.D.E.N.

2.1 Introduction

In order to put the research, described in this report, into context, it is necessary to describe the work undertaken in the W.I.S.D.E.N. project. It is as a result of involvement in that project that the structured Computer Based Learning (CBL) development model UDRIPS was developed. The UDRIPS model, however, was devised independently of the project and subsequently used to develop the CBL material produced at the University of Glamorgan. This chapter will detail the aims and objectives of the W.I.S.D.E.N. project. It will also detail the breakdown of the topics for the CBL material, the standards devised during the project and the development strategy adopted.

W.I.S.D.E.N. (Wide-ranging Integrated Software Design Education Network) was a three year project funded under TLTP-2 (the Teaching and Learning Technology Programme, phase Two). Phase one of TLTP started in 1992 and phase two projects started in 1993, each phase was to be funded for a period of three years. W.I.S.D.E.N. started in late 1993 and covered a period to 1996. Subsequently, funding was provided for a commercialisation phase starting in 1998 to cover a further three years.

The main aim of the project was the development of a wide range of interactive computer-based learning materials in the area of software design, with the additional aim of making teaching and learning more productive and efficient in the mainstream areas of undergraduate computing courses. The objectives of the project were:

1. to develop and disseminate a wide range of interactive CBL material in the area of software specification and design;

2. to develop materials with a commonality of approach;

3. to critically evaluate materials produced for the purposes of establishing best practice for developing interactive CBL materials.

The development of the W.I.S.D.E.N. CBL material was distributed amongst seven consortium universities, Sheffield Hallam; Loughborough; Teesside; Glamorgan; South Bank; Brighton and Heriott-Watt, and a commercial partner CBL Technology Ltd; project management was centred at the lead university site, Sheffield Hallam. Each consortium member took responsibility for development of CBL material in a specific topic area of software development methods, this included: Structured Methods; Formal Methods; Object Orientation and Real-Time Systems.

A consortium member comprised a developer and subject expert for each university. The subject expert was an academic, experienced in teaching that subject and the developer a research assistant, with responsibility for building the CBL material.

Glamorgan and South Bank universities had responsibility for the Structured Methods topics. Glamorgan was charged with developing material to cover:

Data Analysis / Logical Data Modelling;

- Entity Relationship Diagrams;
- Relational Data Analysis;
- Entity Life Histories / Event Analysis.

This distributed development required a need for close collaboration between the consortium members and a regular review of produced work to seek to ensure some consistency of level and style.

2.2 Consortium Members

Within the consortium, the expertise and experience of the members varied greatly, the subject experts were experienced academics. However the developers had a diverse set of backgrounds ranging from Agricultural Economics to Computer Studies. One or two of the subject experts had already developed CBL material in the field of Computing and these people were instrumental in organising a standards group together with some of the newer members to address areas such as interface design and navigation strategies.

2.3 Interface Standards and Navigation Issues

To concur with the objective of the project to produce material with a commonality of approach, both the interface standards and the navigation issues were researched by a working party and their recommendations adopted universally by all group members.

These aspects of CBL authoring have been widely addressed elsewhere, (Shneiderman, 1987, Eberts, 1994, Rettig, 1992, Laurel, 1990, Boyle, 1997) and (Siviter & Brown, 1992, Ross, 1993).

2.3.1 Interface Standards

The interface standards addressed issues such as text colour, fonts & font sizes and background colours. Standards were agreed in the following areas

- Screen resolution - 800x600 with 256 colours;

- Body text - sans serif Arial, black or white, and the minimum size 12 point;

- Hot words - green, to be indicated by turning the cursor into a hand over the hot word;

- Backgrounds - to contrast with the black or white text but colour not prescribed;

- User instructions - italicised, same size as body text but coloured dark blue.

2.3.2 Navigation Issues

Navigation involved adopting a hybrid approach, offering both a sequential path and a quick access or browse facility for each of the lessons within a subject area. This enabled the CBL material to deliver the content in a manner which would suit both serialist and holistic learners (Pask, 1976). Clarke (1989) classifies learning styles as lying on a "continuum ranging from a heavy dependence on structure and guidance, through to a strong preference for minimal structure and guidance". Serialist learners are those who prefer to learn in a step by step manner, they tend to have a focused approach to learning the topic(s). Holistic learners prefer a more unstructured approach allowing them to delve into a topic or topics forming a broad understanding of the subject and refining that understanding as they explore. Pask affirms that "if the teaching strategy is matched to the same type of learning style, the student will learn more quickly and retain the information for longer".

To facilitate this composite navigation feature, it was necessary for the material to be structured in such a way as to correspond to the idea of a lesson within a topic within a course. The standards group of the consortium defined a course as follows:

- A course is a collection of topics

- A topic contains zero or more sub-topics

- A topic or sub-topic contains one or more "lessons"

- A lesson contains one or more sections

- A section contains one or more screens

This structure is illustrated in Figure 2.1.
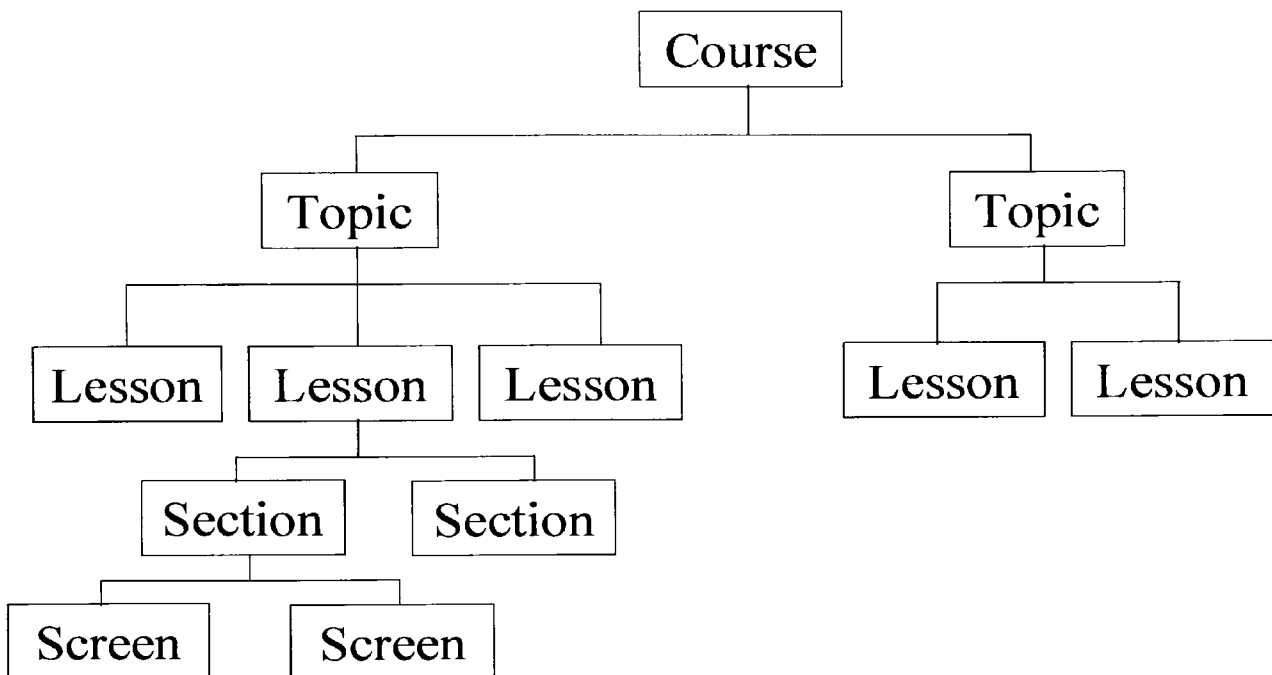
Figure 2.1

Figure 2-1. The Decomposition of a Course.

To achieve this structure it was necessary to refine each subject area from its original high level aims, through broad objectives, to very low level learning outcomes to enable the "chunking" necessary to delimit individual lessons. Within each lesson the learner was provided with a number of sections and within each section one or more pages or screens of instructional material. This concurs with the process advocated by Diana Laurillard for "the design of learning materials for any medium" (Laurillard, 1995). Development of CBL material within W.I.S.D.E.N. was characterised by the close collaboration of the developers and the subject experts throughout the project's lifetime.

From the definitions above, it can be seen that the intention is for the student to be able to choose a topic from a course and from that topic be able to choose each/all of the lessons that comprise that topic. Once the student is in a lesson they may choose each/all of the

sections in the lesson and within those sections there may be one or more screens. It is, therefore, crucial that they are provided with orientation information which gives them a relative position at all times. The orientation information was provided in the form of a course title, topic title and lesson title, so that they could identify where they were within the overall structure and also section name & number and screen number so that they could distinguish where they were within the lesson itself.

2.3.3 The Template

A template was developed in Authorware Professional at the lead site, Sheffield Hallam, and distributed to each of the consortium members. The template was structured to accommodate the structure outlined in Figure 2.1 with respect to lessons and topics. This template incorporated the agreed standards in terms of both interface and navigation and obviated the possibility of duplication of effort.
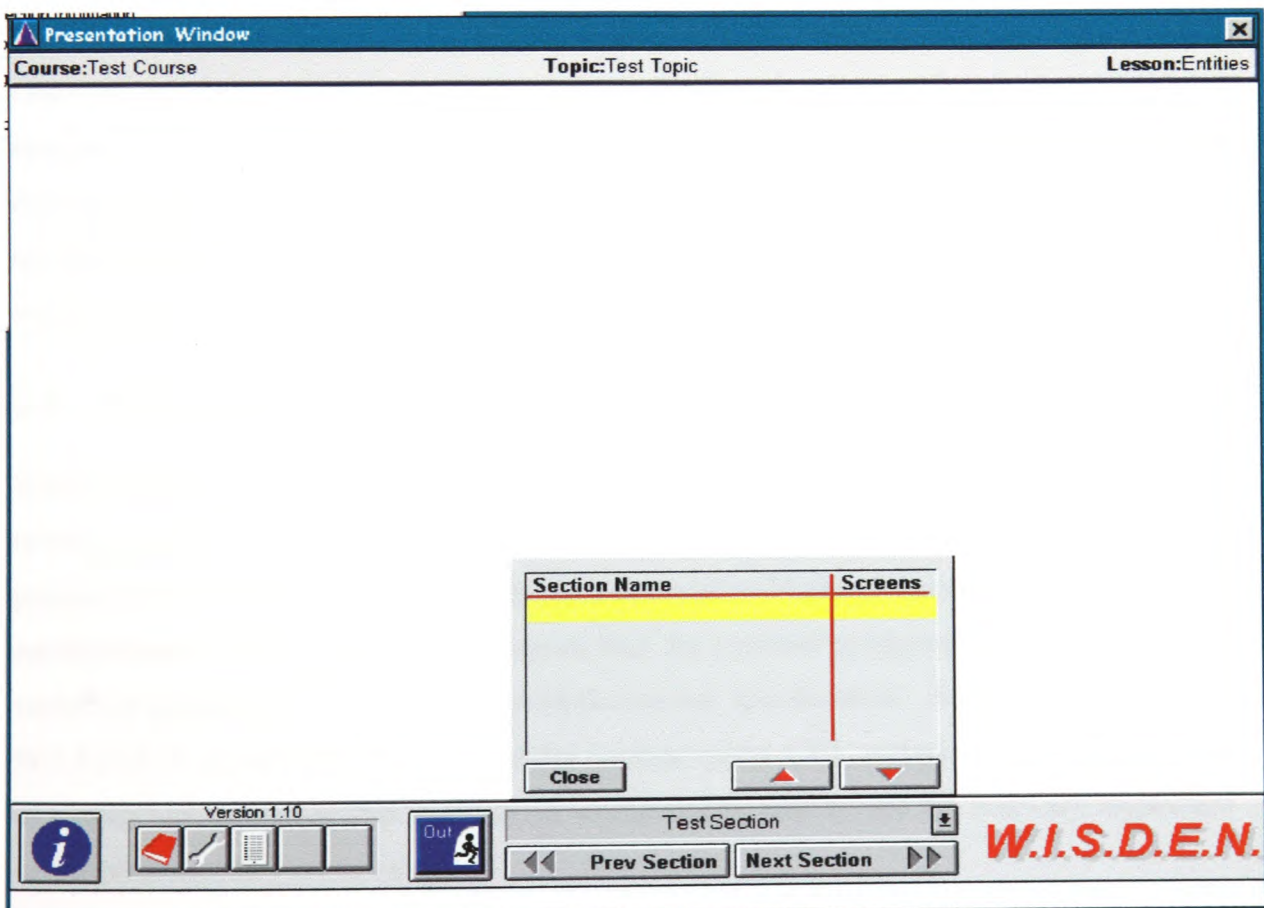
Figure 2-2. The W.I.S.D.E.N. Template.

The template offered:

- A help button, this provided help for on screen elements such as buttons or titles and details of the lesson's author, date created and a short description;

- An orientation bar which contained: Course Title, Topic Title & Lesson Title;

  Both these sections were customised for each consortium member's CBL material.

- Paging buttons for forward and backward navigation, both forward and backward section buttons and where the section contained more than one page, forward and backward screen buttons. If the section contained only one page the W.I.S.D.E.N. logo was displayed instead of the screen paging buttons;

- A menu option offering a direct access facility to any individual section within a lesson;

- An exit button, so the user could leave the lesson at any time.

Once the standards were agreed this left the individual sites free to decide on screen layout, images, animation and subject content and structure. The template built on the structure of screens within sections within lessons and the navigation employed allowed for direct navigation between sections, as we shall see in chapter 7. This concurred with the structure of UDRIPS which focussed on the section details for each lesson.

2.4 Multimedia Elements

Within the W.I.S.D.E.N. consortium, members agreed to limit themselves to the inclusion of text, graphics and animation only. One of the primary considerations for all TLTP projects was the distribution of the finished material to Higher Education (HE) establishments free of charge. This meant that the material produced needed to run on a variety of platforms including those with the lowest specification. As a result, within the W.I.S.D.E.N. consortium the multimedia content of the CBL material produced was kept to a minimum to ensure that institutions would still be able to use the software regardless of the platform they were using.

2.5 Development Strategy

Despite the prescriptive nature of the standards adopted by the consortium members, there were several decisions which still needed to be made. The content of an individual lesson

was decided upon by the individual member as was the number of sections in a lesson and the number of screens in a section. There was a clear need for guidance on how individual lessons should be developed to ensure a beneficial learning opportunity for the student. This problem became clear early in the project when each site was asked to produce an initial prototype related to their subject area and these were then compared with respect to their interface design and general usability. These initial prototypes showed many diverse approaches to these areas and the lack of commonality was clear. It was apparent that no-one had adopted a standard technique to developing CBL material as might have been the case in a more traditional software development project. These early prototypes were evaluated by an educational psychologist and were heavily criticised for their lack of consistency with respect to structure and ease of use.

A primary aim of the W.I.S.D.E.N. project was to produce material with a commonality of approach. It was intended that the individual CBL packages could be combined by tutors to produce a coherent course in any of the topics covered by the project. This included, for example, allowing a tutor to incorporate a lesson on entities within a course on Object Orientation to highlight the similarity between objects and entities. To this end it was necessary that the CBL material produced had a consistent look and feel so that all users could interact with any lesson in a way that did not entail re-learning new navigation techniques and did not require re-learning the structure of lessons each time they encountered one.

In a more conventional software project, many methods and techniques exist to guide the developer and facilitate the production of software. These methods and techniques are often system specific e.g. database design and development, so it would seem reasonable to assume that similar methods and techniques exist to facilitate CBL design and development. To this end, initially, the field of CBL development was researched to identify a method for the production of CBL material which could be distributed throughout the consortium for use by the developers. It quickly became clear that while there were indeed CBL development methods similar to those for conventional software development, techniques to accompany them were not available. This omission provided the rationale for the research project described in this thesis.

This report will follow the development of the CBL material from the initial involvement in the W.I.S.D.E.N. project through the prototyping phases and formative evaluation exercises to the "finished" product and the summative evaluation exercises.

Research into CBL development uncovered a plethora of Instructional Design Methodologies (see chapter 4), however, these design methodologies focused on the high level development process and offered little to individual lesson development. This research project has evolved due to the need to provide a structured approach to CBL development for a number of developers with a number of topics to address, each topic with a number of sub-topics, the lessons, and subsequently to disseminate best practice techniques for future developers.

2.6 Research Areas

The W.I.S.D.E.N. project involved the production of CBL material. The research associated with the project resulted in the development of the UDRIPS model which facilitated that production by assisting in the design and structure of the CBL lessons. In order to form the model, research was conducted in areas associated with software and CBL development and also the pedagogical area to elicit features that might be combined and incorporated to ensure a final model that was efficient, effective and appropriate. A comparison between the fields of traditional software development and CBL development was intended to uncover any differences that exist between the two, and also any common features. In addition, since this project was concerned not only with the development of a CBL package but also with the teaching and learning of student users, an analysis of the educational learning theories was also undertaken. Results from this exercise can be found in chapters 4, 5 and 6.

Subsequent chapters deal with the construction of a structured CBL development model, its application to a CBL prototype, evaluation of the prototype with student users, assessment by other consortium members and finally, its adoption by the consortium as the standard development model for all CBL material produced.

2.7 Summary

W.I.S.D.E.N. was a TLTP-2 project funded for three years to produce CBL material in the area of software systems analysis and design. The project was a consortium based undertaking involving members from geographically disparate locations from a diverse set of backgrounds. One of the aims of the project was to produce material with a commonality of approach, this led to the adoption of a number of agreed standards. Standards were adopted with respect to fonts, colours etc. and a template was formed which was used to underpin lessons, section 2.3.3. Involvement in the project highlighted

the lack of techniques which could be used in conjunction with existing CBL development methods to produce well structured CBL material.

The research described in this thesis details the need for, and development of a CBL design and development model UDRIPS. The aim was to produce a model that could assist in the development of structured CBL material and one that could also be combined with the existing CBL methods, see chapter 4. UDRIPS was devised to address both the pedagogical and software aspects of CBL material by combining principles from both fields. UDRIPS, too, was adopted as a standard by the W.I.S.D.E.N. consortium.

Chapter 3

Computer Based Learning Systems

3.1 Introduction

The aim of this chapter is to define the terms used in the Instructional Systems field and to explain the common acronyms associated with the topic. The various types of systems are identified and explained with reference to their most suitable mode of use. An instructional system is an "arrangement of resources and procedures to promote learning. Instructional design is the systematic process of developing instructional systems and instructional development is the process of implementing the system or plan." (ARL Collaboratory, 2000). As such, instructional systems could cover the whole facet of learning and could include, for example, a book or lecture, though "system" is more closely aligned with a computer based solution. In this research, the instructional systems under investigation are computer based systems. These, therefore, are a sub-set of instructional systems in general. This chapter will provide details of the definitions and use of Computer Based instructional systems, there will also be a short history of instructional design to provide a background to the research.

Instructional material delivered via the computer is generally referred to as courseware. The term is derived from *courses* and soft*ware* but conveys no information about the structure of the subject matter, navigation strategy employed, assessment method adopted nor the suitability of the course for the computer environment.

Courseware is the material delivered via instructional systems such as Computer Aided / Assisted Learning (CAL), Computer Based Learning (CBL), Computer Aided / Assisted Instruction (CAI) and Computer Based Training (CBT) systems, though CBT is usually associated more with the development of work related skills than learning. On the other hand Computer Managed Instruction (CMI) is concerned with the management of instructional material and activities and not directly with the teaching process (Alessi & Trollip, 1991). More recently terms such as learning technology, instructional technology,

educational technology and even academic technology have emerged to cover the use of IT in educational settings. Learning Technology, for example, is defined as: "The application of technology for the enhancement of teaching, learning and assessment. Learning Technology includes computer-based learning and multimedia materials and the use of networks and communications systems to support learning." (Rist & Hewer, 1996).

3.2 Definitions

3.2.1 CAL

CAL is described as: "all those applications that teach with computers" (Hooper, 1977), "the computer contributing to a student's learning" (Institute of Computer Based Learning, 1995). This does not necessarily mean the computer delivers the learning material but aids the learning process, for example providing a web page with links to resources such as a computer based tutorial system, additional reading material, interesting relevant web sites could be classed as a CAL system.

3.2.2 CAI

CAI is described as: "instruction mediated by computer in which the systems allow for remedial action based on answers but not for a change in the underlying program structure" (Patton-Bennington, 1997); "an instructional aid that can help you to attain previously formulated objectives"(Travers, Elliott & Kratochwill, 1993). These are very similar to CAL systems described above.

3.2.3 CBL

CBL is often used interchangeably with CAL, CAI and CBT. "Some will describe it as the use of the computer to present instructional material to students. Others will describe it in broader terms to include all the various teaching, training and learning activities which might involve a computer" (Institute of Computer Based Learning, 1995).
"Computer-based learning (CBL) describes teaching and learning with computers. In such settings, the learner interacts with a computer and computer program that controls and directs the instructional sequence" (Oliver & Grant, 1996). This definition most closely matches the tutorial type instructional system built to illustrate the use of the design model developed in this research project.

The boundaries between all these terms are not clearly defined and the meanings are often subjective. In many cases the term CBL is used to cover traditional tutorial type systems which many people are familiar with, through experiences in work or education. Several other CBL systems may be employed to enhance or extend the learning experience

3.3 Types of CBL System

The are several types of CBL systems that may be developed, these include:

- Drill and Practice

- Tutorials

- Information Retrieval Systems

- Simulations

- Microworlds

- Cognitive Tools

- Communications Tools

- Performance Support Systems

- Intelligent Tutoring Systems (ibid., 1996)

Each of these types of systems will be briefly reviewed.

3.3.1 Drill and Practice

These systems are intended to provide practice for students in a subject which they have already been taught, the taught material may have been provided via a lecture or a classroom type lesson. The drill and practice system provides the student with structured reinforcement that supplements the learning of the previously delivered concepts, it commonly consists of question and answer interactions, with appropriate feedback (Rist, 1996), they may often be found embedded in other systems such as the tutorial type systems.

Drill and practice systems are not intended to teach, however, the practice element of the learning process is an extremely important one. Drill and practice systems can assist the learning of basic Maths skills; foreign languages; spelling and English usage (Alessi & Trollip, 1978).

In summary, drill and practice systems are designed to underpin the teaching / learning experience by providing supportive environments in which the student can "master" new skills or knowledge.

3.3.2 Tutorials

Tutorial systems are used to teach new concepts and processes, the material is given to the student in a structured form and usually includes worked examples with questions, answers and feedback (Briggs, 1991). Tutorials are a primary instructional event as opposed to drill and practice systems which are effectively a supplementary instructional event. A significant characteristic of tutorial systems is that they have clear learning outcomes, these can be used to evaluate the teaching / learning effectiveness of the final product.

Examples of Tutorial systems are CLEM, a system to teach Modula 2 (Boyle et al, 1994) and material derived from the W.I.S.D.E.N. project, in particular, a tutorial system to teach Entity Relationship Modelling (Stubbs & Watkins, 1997).

3.3.3 Information Retrieval Systems

In these systems information is stored and the learner browses or searches for any required material. They include on-line databases e.g. University of Glamorgan's Computerised Library Catalogue known as OPAC (Online Public Access Catalogue) and BIDS (Bath Information & Data Services) Education Web Service: BEI (British Education Index) and ERIC (Educational Resources Information Center); CD-ROM dictionaries and encyclopaedias e.g. Microsoft Encarta and Microsoft Musical Instruments; hypertext and hypermedia reference systems e.g. help systems and electronic books, journals or magazines. In these systems there are no expected learning outcomes and as such they also represent a supportive environment for an existing teaching / learning strategy. The World Wide Web, on the other hand, has the facility to provide a browsing / searching environment and also to provide access to CBL material (URL, Altavista, Macromedia,).

3.3.4 Simulations

Simulations model the world by simulating real, or even, imaginary situations. Simulations, generally, have a high graphic content to ensure the environment is as realistic as possible, however, not all simulations are graphical. The system may be closed, in which case the student observes a pre-set condition, or open, the student may modify the simulation and observe the results. Examples of simulations are: a business plan; a laboratory experiment; an animation of the working of a hazardous environment. Simulations are used to illustrate and investigate relatively complex processes or events. The reasons for using simulations may be because the task itself is:

too dangerous (e.g. a car crash); too impractical (e.g. manage the finances for an entire company) or impossible (e.g. a theoretically impossible chemical reaction). "In a simulation the student learns by actually performing the activities to be learned in a context that is similar to the real world" (Alessi & Trollip, 1978).

3.3.5 Microworlds

Microworlds are a similar concept to simulations but the learner is immersed in an imaginary world rather than a simulated situation. The learners interact with this world and make relevant decisions or solve problems based on their experiences as they encounter new situations. "A Microworld is a small, but complete subset of reality to which one can go to learn about a specific domain."(Rieber, 1994). An example of a Microworld can be found in the METRIC Project (Kent & Ramsden, 1996), ODEWorld: a microworld for Ordinary Differential Equations. "We use 'microworld' to mean a computational environment that represents a particular knowledge domain and that is constructed for the purpose of learning about that domain. It contains:

- computational objects that embody key mathematical ideas;
- activities designed so that by operating on these objects, and constructing other objects out of them, the students can encounter, recognise and explore the mathematical ideas" (Kent, Ramsden & James, 1997).

3.3.6 Cognitive Tools

"A cognitive tool for learning is simply a device, or technique, for focusing the learner's analytical processes" (Mayes, 1992). The intention is for the learner to use a cognitive tool such as an expert system shell or an authoring system to structure their knowledge so that they or another person may view and understand it, this enhances the learner's understanding of the subject area. Similarly, concept mapping tools "encourage the author to divide the knowledge domain into discrete and optionally linked concepts, the act of generating a map provides a useful opportunity for reflection." (Miller, 1995). The concept map itself will not engender learning but the act of arranging and reflecting on the knowledge contained within the map can assist with the learning process. An example of a cognitive tool is the mind-mapping technique devised by Tony Buzan (Buzan, 1993) where a topic is decomposed into its sub-topics each of which is linked both to other sub-topics and to the main topic
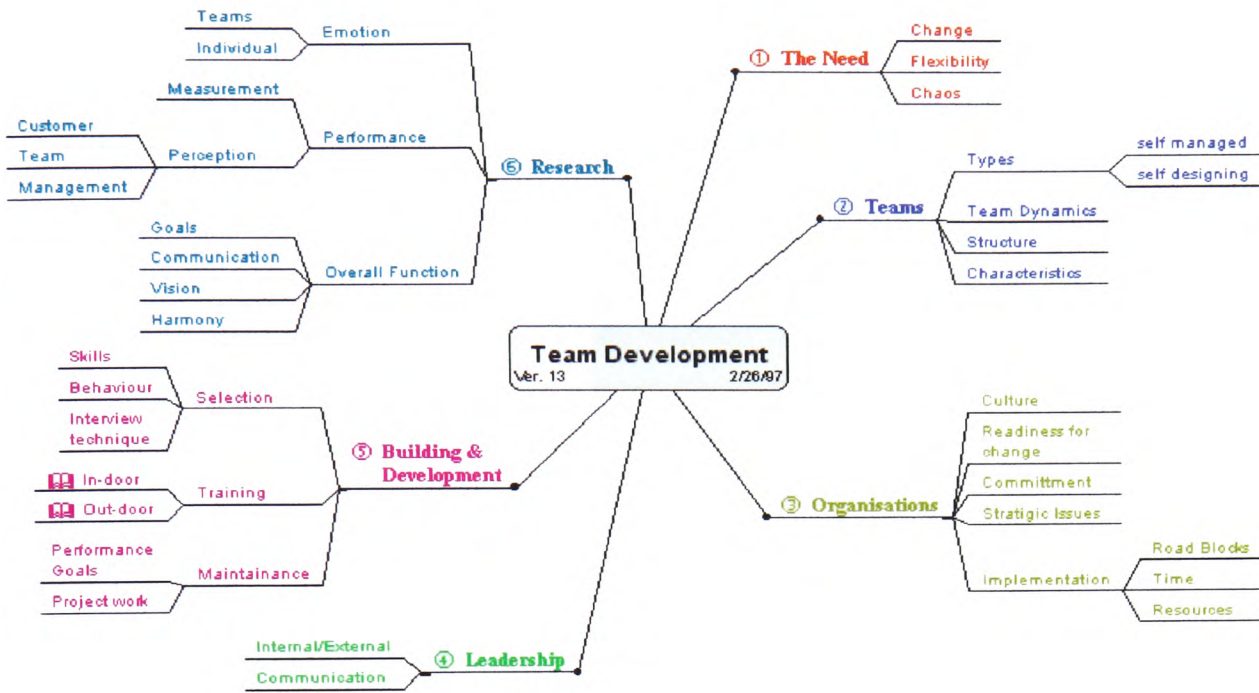
Figure 3-1. Example of a Mind Map.

3.3.7 Communications Tools

Communications Tools are used to enable computer supported collaborative / co-operative work. Communication tools can consist of electronic mail (email), conferencing systems (Video, Audio, Computer) and Internet resources such as File Transfer, Usenet groups, Bulletin Boards and the World Wide Web. These tools give students access to joint work; submission and publication of individual assignments and the Tutor for the course. They also provide an opportunity for students to share ideas and information with one another. This is a supportive activity rather than a subject delivery mechanism and ensures the student has the additional benefit of social contact and extensive supplementary information. "Over the last decade there has been increasing interest in the application of computer mediated communication systems to support distance learning." (Birchall & Smith, 1996).

3.3.8 Performance Support Systems

"A Performance Support System is essentially a 'job aid' that enables its users to improve the efficiency and effectiveness with which they are able to complete particular tasks" (Barker et al, 1994). In other words the learning activity is tied in explicitly with a particular task that is to be performed e.g. help systems as found in word processing

packages, spreadsheet applications or Authoring Systems such as Authorware Professional. They are also provided in the form of Wizards that guide users through common tasks e.g. setting up a Powerpoint presentation.

3.3.9 Intelligent Tutoring Systems

The aim of Intelligent Tutoring Systems, is to provide each student with an individual instructional experience. In these systems the students' mistakes and interests drive the tutorial interactions, each system adapts individually to the needs and actions of the user. "The main components of an intelligent CAI (ICAI) system are problem-solving expertise, the student model and tutoring strategies." (Barr & Feigenbaum, 1982).

3.4 Comparison of CBL Types

CBL systems fall into two main categories those that provide a direct learning experience for the learner and those that provide an indirect learning experience. The direct learning experience is described in section 3.4.2 as a primary instructional system and the indirect as a supplementary instructional system, section 3.4.1. Which CBL system to adopt is dependent on a number of factors. These include: the subject to be taught; the learner; time and cost available; expertise of the developer and requirements of the lecturer / teacher for the subject, the developer and the lecturer may or may not be the same person. It is possible that more than one system may be combined to deliver the instructional material e.g. a tutorial system might contain a case study as an illustrative example which is delivered in the form of a simulation. Many of the CBL systems outlined above do not deliver instruction as such and are used as a supplementary guidance or support mechanisms. However, the primary consideration is the type of learning experience required for a particular group of learners. It is dependent on whether or not the learning is to be a primary instructional event or a supplementary instructional event.

3.4.1 Supplementary Instructional Systems

It is necessary to consider if the CBL system is to provide a primary instructional event i.e. is the subject matter to be taught, to be delivered via the computer and is there a requirement for the instruction to be well structured? If this is the case, then drill and practice systems, information retrieval systems, cognitive tools, communication tools and performance support systems should not be considered, as these are supplementary resources within a learning situation Here, the subject matter to be learned has been

delivered via another method e.g. lecture or seminar and the supplementary instructional systems provide additional problem solving opportunities or resources.

3.4.2 Primary Instructional Systems

Simulations and Microworlds provide a "real world" environment for the student and as such are more suited to certain areas than others e.g. experiments, hazardous environments, Mathematics environments etc., it is also possible, however, to combine a modest version of either the simulation or the Microworld within another CBL system such as a tutorial, as mentioned above, this combination could then provide a primary instructional event. However, Simulations and Microworlds are often used as additional resources in the learning process providing exploratory and often problem solving opportunities rather than delivering the topic under instruction.

As far as primary instructional events are concerned the two principal methods for providing this learning experience are tutoring systems; both conventional and intelligent tutoring (IT) systems. In the comparison between Tutoring Systems and IT Systems, the time, cost and expertise necessary to develop IT Systems is considerably greater than that taken to develop conventional Tutoring Systems. IT systems, not only deliver the subject matter to be taught but also seek to provide a personalised learning experience, this takes much time and effort to analyse and develop. Problems that can arise are:

- considering every individual who might use the system;
- assessing the students' difficulties and adapting the system to rectifying them;
- providing a suitable remedial strategy.

The conventional tutorial system has the benefit of delivering the subject matter to be taught, at the same time providing worked examples, questions, answers and feedback and also, in most cases a problem solving opportunity. They do not, however, attempt to provide the individualised learning experience mentioned above, nevertheless, that does not imply that they will provide a static learning environment, many tutorial systems offer composite navigation structures in order to facilitate the learning styles of a diverse set of learners. They can also provide an interactive environment which can stimulate motivation and aid learning. These features are all designed to combat a common criticism of CBL systems i.e. that they are merely page turning activities with little regard to learning in terms of student input or motivation. In the past, such systems were said to provide a passive learning experience and "in spite of some well-intentioned programming, were not

very educational." (Schank, 1994). These criticisms must be borne in mind by the diligent developer and can be addressed and overcome as described above.

3.5 Summary

Several factors impact on the decision of which type of CBL to employ in a particular teaching/learning activity, they include the subject matter to be delivered e.g. if the subject is an experiment, it would seem logical to use a simulation, and then to decide if it is to be embedded in a structured "lesson" such as a tutorial system. If the CBL is not intended to deliver new material but rather to re-enforce or give practice in some aspect of already delivered material then it would seem pointless in developing a tutorial system.

In summary, it is necessary to determine what each system provides and decide on the relevant type based on the individual instructional activity for which there is a need to employ a CBL approach.

Questions which can determine the most suitable CBL approach are focused on areas such as:

- the audience needs;
- the lecturer's needs;
- time allocated for development;
- cost;
- subject matter;
- type of instructional system required e.g.
  - ➤ primary instructional system;
  - ➤ supplementary instructional system.

This research focuses on the design and development of tutorial type CBL systems. These are, as described in section 3.4.2, primary instructional systems. As such, the design requires the developer to address:

- the objectives of the subject matter;
- the structure of the instructional system;
- the provision of examples;
- the provision of questions;
- feedback for the learner.

This research will examine the approach to CBL design and development with the intention of providing a model/method to assist CBL developers. Instructional systems

have been investigated and produced for many years. Work has, more recently, focussed on the provision of computer based solutions.

3.6 History of Instructional Systems

CBL research and development has been in existence for a considerable amount of time and stems from work in the field of Psychology and interest in promoting and facilitating learning. The definitions and types of systems outlined above reflect the extensive work that has taken place and continues to take place in this field.

Early work in this field was conducted by Sidney L. Pressey who designed "a simple apparatus which gives tests and scores - and teaches," (Pressey, 1926). Obviously, at this early stage, the machine in question was mechanical rather than electronic but nevertheless the idea of providing automated instruction has existed for over 70 years. Subsequently, after a number of years when little progress had been made in the area, Skinner (1958) experimented with "teaching machines". These machines were based on the behaviourist principles propounded by Skinner and attempted to "teach" the user by inducing a change in behaviour. These machines had a number of key features:

- The user was provided with the opportunity to "compose" responses rather than choose from multiple choice type questions i.e. active responding;

- The material was provided in a carefully constructed sequence, logical sequence;

- Within the sequenced material the steps taken had to be sufficiently small so that the user could achieve "fully competent behaviour";

- Users were allowed to progress at their own pace;

- On-screen textual stimuli were gradually "vanished";

- The machine provided re-enforcement for each correct response with immediate feedback;

- The material incorporated into the teaching machine was derived from an analysis of the knowledge to be imparted to the user.

This work underpinned much of the ensuing research activity and is particularly interesting from the point of view that Skinner promoted mechanised instruction as an additional resource rather than a replacement for the teacher/lecturer, (Skinner, 1958).

In 1963, Patrick Suppes began to investigate the use of Computer instruction to teach basic skills to disadvantaged students, in particular, elementary mathematical skills and his colleague Richard Atkinson focussed on basic reading skills. By 1975, Suppes had also

gained experience in teaching French, German, Russian and Chinese by computer (Suppes, 1980).

By the 1960's the University of Illinois, Urbana-Champaign had embarked on the PLATO (Programmed Logic for Automatic Teaching Operation) project. The aim of this project was to deliver CBL material via a mainframe system to provide a more powerful environment for the provision of teaching material. The PLATO system used a proprietary programming language called TUTOR to write instructional software, however, as the system progressed, additional features such as Talkomatic and term-talk were incorporated. Talkomatic was a multi-user chat facility, term talk was similar but restricted the number of participants to two: term talk, however, also provided a paging facility for users, (O'Shea and Self, 1983, Wooley, 1994).

By 1971, the NSF (National Science Foundation of America) had decided to fund both the PLATO project and a collaborative project, the TICCIT (Time-shared, Interactive, Computer Controlled, Information Television) system; the collaborative partners were from Brigham Young University and a group from the MITRE corporation. The aim of this system was to demonstrate that CBL could provide better instruction in English and Maths at a lower cost than traditional instructional methods at community colleges (Pagliaro, 1983). Within the TICCIT project, one of the major contributions to emerge was the development of an instructional theory, the Component Display (Design) Theory developed by M. David Merrill, (Merrill, 1980). The Component Display Theory presents a classification of learning material which consists of four categories: Facts; Concepts; Procedures; Principles.

A fact is defined as an association between objects or events and the symbols used to represent them. A concept is a group of objects, events or symbols with common characteristics and a particular label or name. A procedure is a sequential series of steps which when followed produce a desired outcome. A principle provides an explanation for a particular event or incident. Since then Merrill has been involved in extending the Component Display Theory; with Charles Reigeluth he has developed the Elaboration Theory, which works at course level and sequences instruction in order of complexity from the least to the most complex (Reigeluth, 1999), and as part of the ID2[1] group at Utah State University the Instructional Transaction Theory which deals with the grouping of "knowledge objects" and the representation of those knowledge objects to facilitate student

---

[1] ID2 Research is the study of second generation instructional design theory

learning (Merrill, 1980, Merrill, 1991, Department of Instructional Technology, Utah State University).

By the 1970's the first microcomputers began to emerge, these machines were considerably smaller and more manageable than their mainframe predecessors but still possessed comparable processor power and cost a great deal less. By 1975, the first of the microcomputers, the Altair 8800, the Intellec-8 and the Motorola Microcomputer were released in kit form, however, these machines did require the user to have some expertise in order to be able to construct them. The first microcomputer to appeal to the public at large was the Pet 2001 which was produced ready assembled in 1977 and required no specialist computer or electronics knowledge to use (Pagliaro, 1983).

In the UK, between 1967 and 1969, NCET (The National Council for Educational Technology) undertook a feasibility study into the use of CBL systems in Education. As a result of this study, the NDPCAL (National Development Programme in Computer Assisted Learning) was approved and began in 1973, the project lasted until 1977. Within this programme around thirty five projects were funded, they were in five categories, Tertiary Education, Schools, Armed Services, Industrial Training and Transferability. Of these, 17 were development projects, 6 were transferability projects and 12 were feasibility studies. The final report of this programme defined the terms Computer Assisted Learning (CAL) and Computer Managed Learning (CML); CAL was defined with respect to two distinct uses, in the first the computer acts as a sort of machine tutor and in the second the computer acts as a learning resource. CML was defined as the computer helping the tutor to manage the learning experience rather than providing a learning opportunity. Subsequently, the microcomputer enabled both individuals and groups, such as the Computer Based Learning Unit at Leeds, to develop CBL material either as a research exercise or to underpin the teaching/learning activity in individual subject areas, (URL, CBL Unit, Leeds).

3.7 Specialist Groups

Interest in this area spawned various specialist groups, many funded through the Higher Education Funding Councils (HEFC), such as CTI (The Computers in Teaching Initiative) which originated in 1984 but which, in its present incarnation, began in 1989; ITTI (The Information Technology Training Initiative) started in 1991 and the three phases of TLTP (Teaching and Learning Technology Programme) the first of which began in 1992 through

to the third phase which started in 1997 and which finished in 2000, (URL, TLTP
Projects);

ALT (the Association of Learning Technology) which started in 1993 at the CAL '93
conference in York was funded by a BT grant but in 1997 became a charity, (URL, ALT).

3.7.1 Objectives

The CTI is a group of 24 "discipline-specific support centres", e.g. Biology, Chemistry
etc., each of which each has the aim of promoting the use of communication and
information technologies, within that specialist area, in UK higher education (HE). The
ITTI strives to make the use of technology in HE more effective and provides IT material
to HE institutions; it has also funded 29 projects at universities in the United Kingdom.
TLTP has funded 43 projects in its first phase; 33 projects in its second phase and is
currently funding 32 projects in a third phase. The objectives of Phase 3 are to:

1. "Encourage the take up and integration of TLTP materials and other technology-based
   materials into mainstream teaching and learning.

2. Explore, adapt and disseminate experiences from integrating such materials, to identify
   successful approaches that can be applied generically, rather than just to specific
   subjects.

3. Develop effective networks to deliver materials to end-users.

4. Encourage continuing collaboration between higher education providers to develop and
   implement materials using standard delivery environments (that is, the technical and
   physical means to deliver the materials directly to the student)." (URL, TLTP Phase 3)

As can be seen from this, there is a considerable amount of activity in this area both in the
USA and UK and indeed throughout the education community, a major conference
exploring these issues is the ED-MEDIA conference organised by AACE (the Association
for the Advancement of Computing in Education). This conference alternates its venue
between North America and Europe and its proceedings reflect the global activity of
researchers and practitioners working in this field (URL, EdMedia Conference).

3.8 IEEE Learning Technology Standards Committee

Currently the IEEE Learning Technology Standards Committee has been working to define
and develop "technical Standards, Recommended Practices, and Guides for software
components, tools, technologies and design methods that facilitate the development,
deployment, maintenance and interoperation of computer implementations of education

and training components and systems." (IEEE Learning Technology Standards Committee). This committee is made up of members from the Institute of Electrical and Electronics Engineers (URL, IEEE), the Computer Society and the Standards Activity Board. An interesting area under consideration refers to a Learning Technology Systems Architecture which describes "a high level systems architecture and layering for learning technology systems" (Farance & Tonkel, 1998). The architecture covers the main components of a learning system:
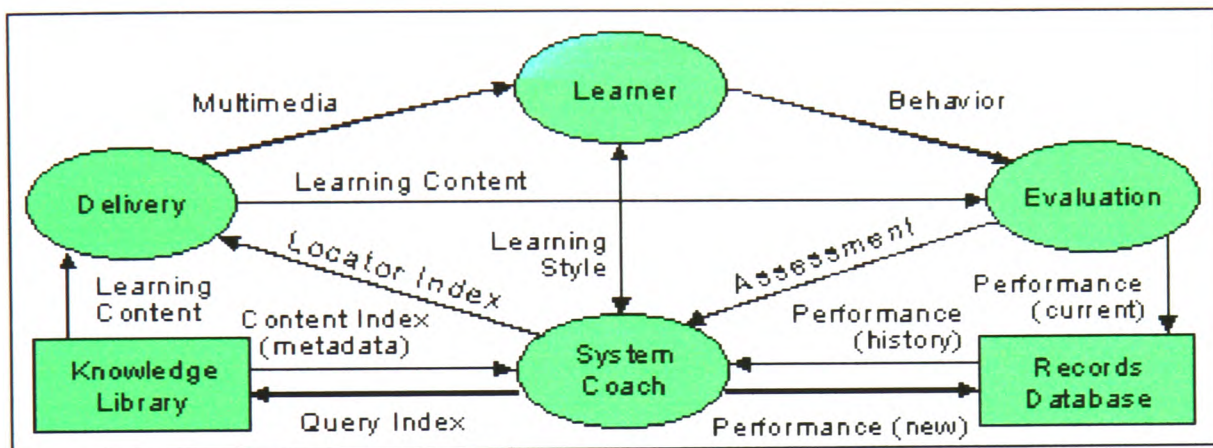
Figure 3-2. Learning Technology Systems Architecture.

(ibid., 1998).

The architecture depicts four processes Learner, Evaluation, System Coach and Delivery and two stores, the Records Database and the Knowledge Library, the arrows show information flows between the elements. The Records Database holds information about the learner, the Knowledge Library stores resources for the learner such as tutorials, CBL courseware, simulations etc. The Learner is a student who comes to the system to learn a topic and the System Coach may be the Learner in a "learner-centred" system or a tutor in an "institution-centred" system (ibid, 1998). Evaluation assesses the learner and determines the most appropriate resources and delivery depending on the individual outcome.

With respect to the CBL courseware developed in this research project the area of interest is the knowledge library which stores "knowledge, presentations, tutorials, etc. as resources for the learning experience". In this definition of an LTSA, the CBL would be one of the resources provided for learners.

3.9 Conclusion

Over the years, as CBL development has progressed there has been a move to make the process more rigorous and structured. This mirrors very well the history and progress of the software engineering field where the same characteristics have been sought for the same reasons, a more detailed account of this comparison may be found in chapter 4. However, despite this thrust there is still a wide gap in the provision of methods and models available for software engineers over those available for CBL developers. Within the W.I.S.D.E.N. project, the CBL material produced took the form of tutorial systems which were anticipated to teach the various subjects. These tutorial systems were also designed to provide, exercises and problem solving opportunities to assist in the learning process. In some of the tutorial systems, simulations were also included as case studies or illustrative examples. The tutorials were intended as primary instructional systems to provide as complete a learning experience as possible. Initially, the approach taken was that the CBL tutorial systems would be used as additional resources alongside more conventional approaches such as lectures, seminars and tutorials. The aim of the research reported in this thesis was to provide a structured CBL design and development model to assist in the production of the CBL material. The overall aim was to bring some structure and order to the process of CBL development which mirrored those advocated in traditional software projects.

Chapter 4

Comparison of Software and CBL Development

4.1 Introduction

This chapter seeks to identify the common elements of software development and CBL development to determine if a more structured or methodical approach to CBL development is a) warranted, b) feasible and c) likely to be beneficial. The CBL system that was developed during this project was a tutorial type system. These systems offer a primary instructional experience in the sense that the material to be learned is delivered via the computer together with opportunities for practice and problem solving. As such, the CBL development methods discussed will be examined to ascertain their relevance to this particular domain.

4.2 Background to Software Development

Software projects cover areas from Business to Engineering and Science to Artificial Intelligence and the need to address the development process to alleviate problems with time delays, cost escalations and quality has been, and continues to be, a high priority. Initially, the process of software development was an ad-hoc activity with much reliance on the experience of the developer and little or poor interaction between the developer and the intended user, this resulted in the production of software with many problems. Often, the software failed to meet the user requirements and was abandoned or required major alterations resulting in escalating costs and massive time delays. As hardware costs dropped and the speed and power of computers grew, the potential for more complex systems became a reality, the size of software projects increased and the problems, outlined above, intensified until the industry found itself approaching a "software crisis". See Sommerville (1989), Ince (1990) for a more in-depth exploration of this area.

It was, clearly, necessary to address the problems to improve standards and quality. To this end techniques from the engineering disciplines were examined and subsequently adapted and adopted to meet this need. Software projects over the past twenty five to thirty years have expended much effort in seeking to impose some structure on to the process of software design and development. An early definition of software engineering was given by Fritz Bauer in 1969 which was: "Software Engineering is the establishment and use of sound engineering principles in order to obtain economically software that is reliable and works efficiently on real machines." (Bauer, 1969). Another definition is: "The use of methodologies, tools, and techniques to resolve the practical problems that arise in the construction, deployment, support and evolution of software." (Gentleman, 1990). Thus, as long ago as 1969, it was determined that there was a problem with software development and that the aim of software engineering was, and is, to emulate the traditional engineering disciplines with respect to their rigour and usefulness.

4.3 Background to CBL Development

CBL development is another area of software development but one which has been, primarily, though not exclusively, an activity associated with the academic world, (see chapter 3). Academic CBL projects have tended to fall into three categories:

- student projects (group or individual);

- CBL schemes by staff to cover their own specialist area (group or individual), these may be research type projects where staff wish to test out hypotheses or projects to provide supplementary or replacement resources. Staff may be in a single institution or part of a disparate group, here the staff are both subject experts and developers;

- provision by members of staff of content material for projects undertaken to produce CBL material either locally or remotely. Here the staff are acting as content providers only i.e. subject experts but the implementation is carried out by specialist developers. Both these latter projects may be large, consortia based projects as per the TLTP initiatives, or might be a single site initiative such as a departmental scheme where a single developer or a team of developers produce material for an entire department.

Many CBL projects have grown in size and moved from a small-scale undertaking to a larger more complex operation requiring just as much consideration of project

management in terms of costs and time as conventional software projects. This growth in project size has been, in part, as a response to evaluation reports from funded initiatives such as the Teaching and Learning Technology Programme (TLTP) (http://www.ncteam.ac.uk/tltp/) and the Fund for the Development of Teaching and Learning (FDTL) (http://www.ncteam.ac.uk/fdtl/).

An additional consideration has been the "not invented here" syndrome where CBL material has not been utilised as anticipated due to the fact that it did not exactly match the needs of academics who might have incorporated it into their teaching/learning strategy (Davies & Crowther, 1995). In order to counteract this perception, consortia based projects have sought to involve many academics either as developers or subject experts, in order to consider as many viewpoints as possible and to make the final CBL acceptable to a wider community. An evaluation report on the TLTP initiative by Coopers and Lybrand confirmed the hypothesis that involvement in a consortium based project made the uptake of CBL material much more likely, and thus the majority of the funding provided for projects was, and continues to be, based on large, often geographically disparate groups (Coopers & Lybrand, 1996). Thus it can be seen that CBL projects have mirrored software projects with respect to their growth in size and complexity.

4.4 CBL & Software

The rise of software engineering with its "structured" approaches was a direct result of the ills that have plagued, and continue to plague, the process of software design, development and implementation such as:

- software production and maintenance is costly in terms of time and money;
- software production is labour intensive and takes a large share of project budget;
- software maintenance frequently costs more than the original production;
- delivered software is often unusable in terms of its reliability and robustness;
- testing of software is often inadequate and can be expensive when software is poorly designed.

The structured approach to software development has sought to improve the design, implementation and testing to produce software that is clear and understandable, reliable and robust, easy to maintain, easy to test and efficient. The main activities of structured design methods are: top-down decomposition which involves the functional decomposition of a complex process or procedure into a set of components which make up the system;

44

and abstraction which deals with the important aspects of a system without the consideration of any inessential details (Britton & Doake, 1993).

Examples of these 'structured' design methods include the Process(Function) Oriented Paradigm (Yourdon & Constantine, 1979), the Data Oriented Paradigm (Jackson, 1983) and the Object Oriented Paradigm (Booch, 1991, Coad & Yourdon, 1991), coupled to these have been the structured systems methodologies such as Structured Systems Analysis and Design Methodology (SSADM), Jackson System Development (JSD), an extension of Jackson Structured Programming (JSP), Object-Oriented Analysis (OOA) and Object Oriented Design (OOD) and project management methods such as Prince (Bentley, 1992). The advent of Princess is an attempt to add the educational perspective to conventional project management issues and address the educational software development process (Hobbs, 1995).

The benefits of this approach to software development has been recognised and in an effort to provide a more rigorous approach to the process of CBL development, many instructional design models have been adapted from the field of Software Design or created specifically for the CBL area. Some methods mirror very closely those in the software development area, e.g. the life-cycle model (Sommerville, 1989, Braxton, 1995) and rapid prototyping, (Boehm, 1976, Tripp & Bichelmeyer, 1990, Wilson et al., 1993).
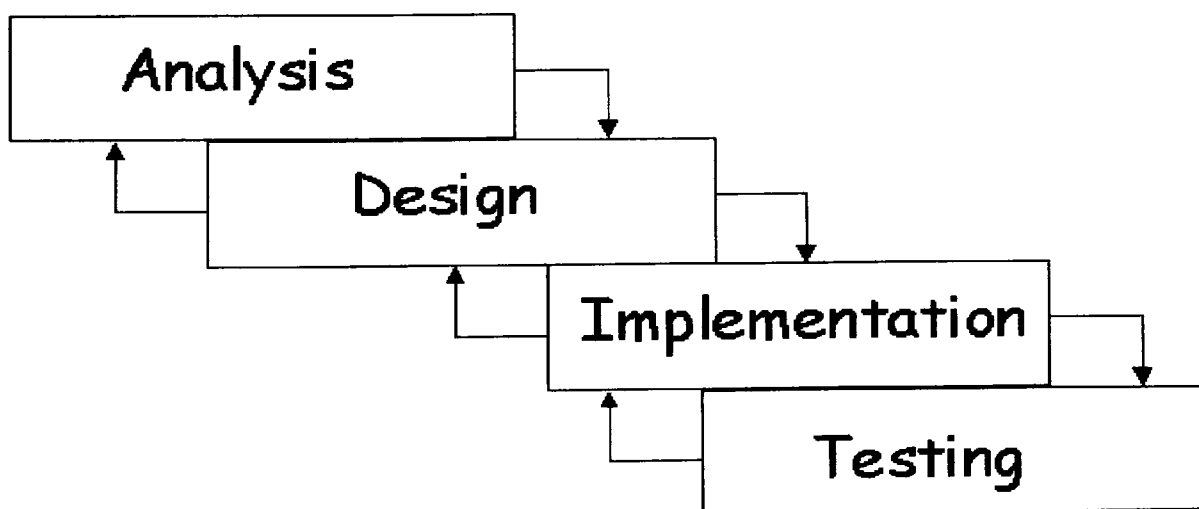
Figure 4-1. Iterative Waterfall Model (Sommerville, 1989).

This diagram depicts the phases in a software project. During the analysis phase, the functions, constraints and performance of the software system are determined and agreed between the software engineer and the user. During the design phase the outline of the software system is established. During the implementation phase the actual system is built.

Finally, during the testing phase the system is assessed in use with the users. This model depicts an iterative process where feedback is used to refine the output from each of the phases.
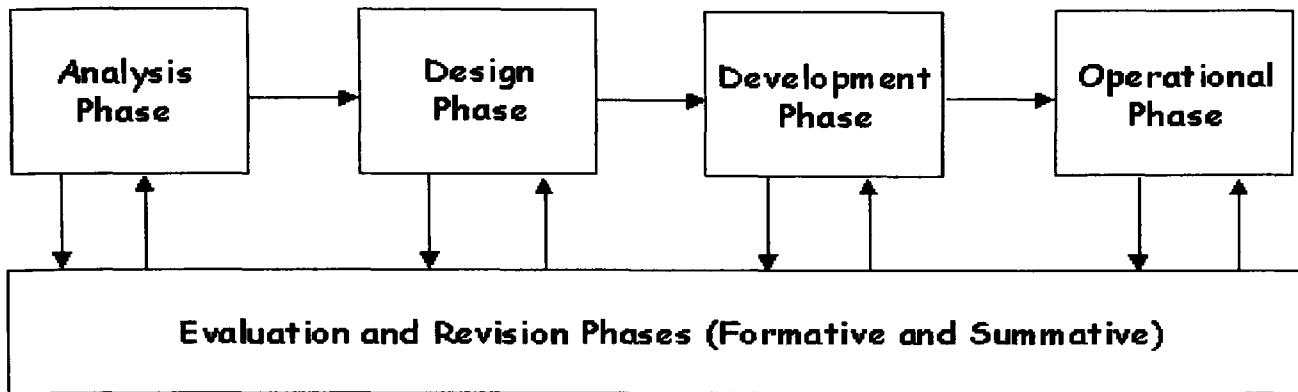
Figure 4-2. The CBL Development Life-cycle (Braxton, 1995).

This model is very similar to the iterative waterfall model detailed above but is specific to CBL development. In this case the phases are very similar but the idea of testing is refined to formative and summative evaluation to enhance the quality of the CBL product. Formative evaluation is performed throughout development to help form the system. Summative evaluation is performed "in situ" with the target users.

Just as in software development where this phased approach underpins many software development methods such as SSADM, Object Orientation etc., models for CBL development are often underpinned by the life-cycle phases outlined above.

A typical tutorial type CBL system aims to teach a subject to the user. The system also provides scope for practising with the acquired knowledge through completing exercises or problem solving opportunities. To this end, the major difference between the CBL system and the conventional software system is that the learners' needs and skills need to be addressed as does the didactic nature of the system. Methods which specifically apply to CBL development address these areas and involve phases devoted to an analysis of the learner with respect to their needs and current knowledge (Dick and Carey, 1996, Hannafin and Peck, 1988, Knirk and Gustafson, 1986, Jerrold Kemp, 1994, Gerlach and Ely, 1980). Of these, perhaps the most well known is the Dick and Carey Systems Approach Model for Designing Instruction.
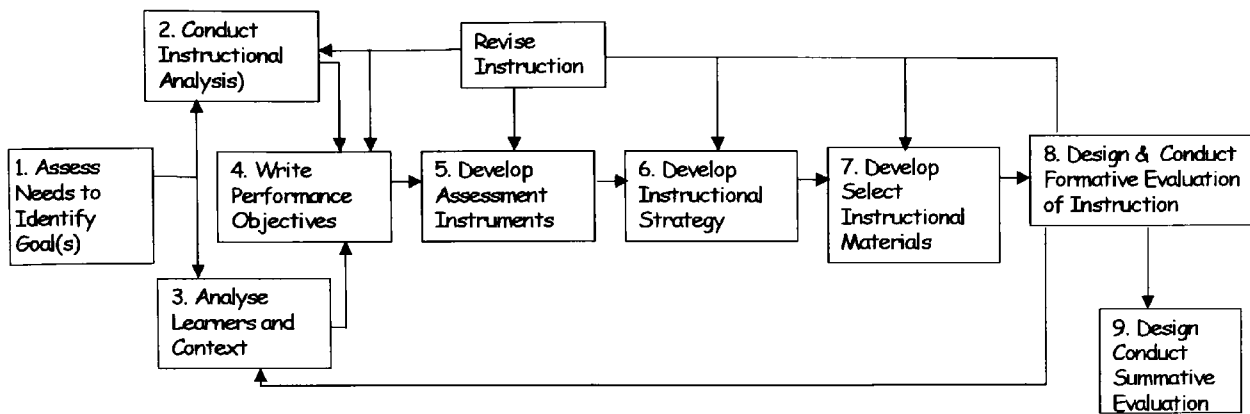
Figure 4-3. Dick and Carey Systems Approach Model for Designing Instruction

(Dick and Carey, 1996).

This model shows an iterative process, which begins with the identification of the instructional goals and ends with summative evaluation. The model moves through the stages to develop and refine the CBL material. Initially, the analysis focuses on the needs of the learners. Subsequently the topic area is decomposed into low level objectives. These objectives form the basis of the CBL system together with assessment activities and additional media elements such as graphics, sound or video. Throughout this process the material is formatively evaluated to aid in its refinement. Finally, the CBL material is summatively evaluated to determine its effectiveness with the target users. Many of the phases have techniques which facilitate the activity undertaken during that phase. For example, the decomposition of a topic into low level objectives may be aided by the use of Bloom's Taxonomy (Bloom, 1956) which identifies the level of learning in any instructional situation.

Thus, it can be seen that whilst the development of a CBL system is a software activity which involves the production of a computerised product, the development is extended to address the pedagogic needs of the learner and the didactic needs of the tutor associated with the teaching/learning of a topic. To this end, the models associated with CBL development attempt to combine techniques from the pedagogic theories, e.g. needs analysis, together with practices found in conventional software development, e.g. the phased approach.

4.5 Characteristics of Methods/Models

Models used in software development are characterised by three elements:

Methods; tools and procedures (Pressman, 1992). Here, the methods outline the tasks within the life-cycle e.g. requirements analysis, design, implementation etc., tools provide support for the methods and procedures provide the sequence of the tasks, the deliverables and relevant milestones throughout the project. Tools that currently may be applied in a software project include, but are not limited to, Data Flow Diagramming, Entity Relationship Modelling and Object Oriented Modelling.

The characteristic elements may also be identified as:

Methods; techniques and tools (Beynon-Davies,1998). Here the methods, again, outline the tasks within the project, but the techniques determine how the tasks may be accomplished and the tools relate to any hardware, software etc. that may be needed within the project e.g. email for communications between team members. In this definition, techniques applicable in a project include Data Flow Diagramming, Entity Relationship Modelling, Object Oriented Modelling etc.

Whichever categorisation is used it is clear that there is a comprehensive approach to software development which provides a methodical mechanism for producing software that is often, though not always, prescriptive. All these techniques allow the developer to analyse existing systems with respect to the "items of importance" and their relationship to, or impact on, one another. They also provide a mechanism for the developer to elicit the required information from the user by explicitly stating where the developer needs to look to derive the information from the existing system. This enables the developer to communicate his/her needs to the user in an often unfamiliar environment.

A particular method which typifies this approach is SSADM (Structured Systems Analysis and Design Methodology), introduced in 1982 by the UK government to aid the development of information systems in government departments (see Figure 4-4).
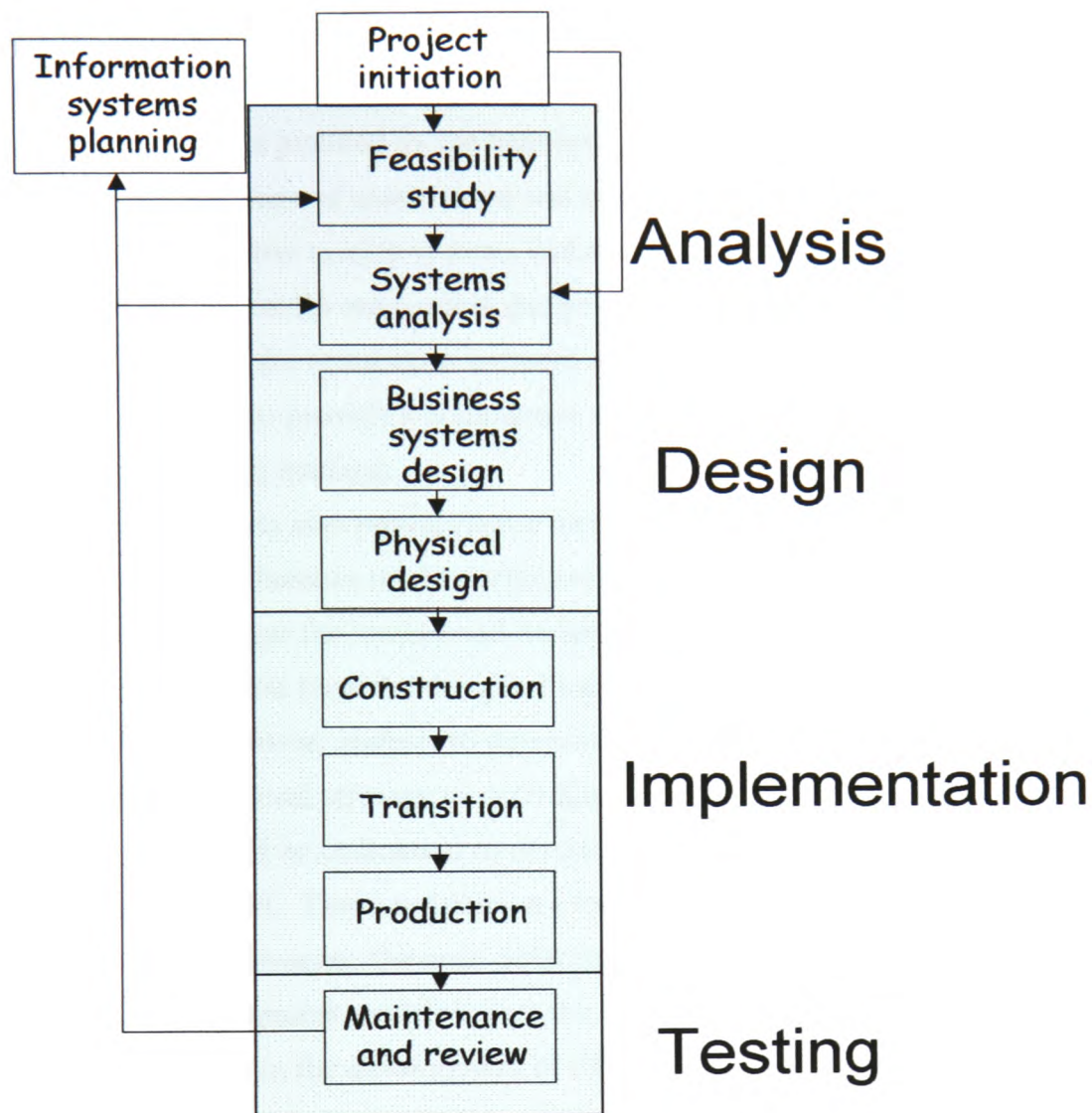
Figure 4-4. SSADM (Goodland & Slater, 1995).

The methodology does not cover the implementation phases of a project but is intended for the analysis and design phases. SSADM utilises three techniques for analysing the data in a system, these are Logical Data Models, which show how data is stored and inter-related, Data Flow Models, which show how data flows through the system, and Entity Life Histories, which show how data changes during its existence. These three techniques are related and aim to provide three different views of the same data so that each one can be used to check the "consistency and completeness of the others" (Goodland & Slater, 1995). The aim in this method is to move through an analysis of the existing system to form a high level "conceptual" design of the system with respect to its functions and requirements, to a logical system design which shows how the new computerised system will work which is hardware and software independent, to a physical design which

49

combines hardware and software details with the logical design. This method admits to a prescriptive approach but insists it allows for a great deal of flexibility for the developer. The prescriptive nature is justified by the benefits associated with its application which include: "more effective use of experienced and inexperienced staff; improved project planning and control; better quality systems and a clear requirements statement that provides a firm foundation for subsequent design and implementation" (ibid.). This method is used to show the connection between a method and its techniques. In the same way, this research aims to provide a comparable technique which may be used in conjunction with existing methods.

CBL development models also provide some methods and techniques for CBL development but the techniques tend to originate in the field of Psychology and they focus on various ways to analyse the learner and derive objectives which cover the topic under instruction. In its simplest form the design of instructional systems is defined as:

1. "Perform an instructional analysis to determine 'where we're going'.

2. Develop an instructional strategy to determine 'how we'll get there'.

3. Develop and conduct an evaluation to determine 'how we'll know when we're there'".

(Smith and Ragan, 1999). These activities are found to a lesser or greater extent in all the CBL design models. However, the main difference between the CBL development models and the software development models is that the CBL development models do not provide a prescriptive approach to the development of CBL. They also do not provide a clear path from the existing instructional approach through an abstract logical view of the process to the physical implementation of computerised instruction or "lessons". This can prove problematic for a CBL developer who is:

    a) an experienced software developer but an inexperienced CBL developer or

    b) an experienced tutor but an inexperienced software or CBL developer or

    c) an experienced CBL developer but one who is inexperienced in the particular subject area.

Techniques available during the analysis phase of CBL development include but, again, are not limited to:

"Needs analysis" which is defined as the determination of the "characteristics, current competencies and needs" of prospective learners, (Allessi and Trollip, 1991) and also as "Needs Assessment" which focuses more on whether there is a need for intervention in the instructional process (Kemp at al, 1998). Both these techniques focus on the learner and are used in the Dick & Carey model, above, to determine the instructional goals for the

system during the analysis phase of the project. Subsequent phases refine the goals into objectives and, hence, learning outcomes which determine the skills the learner needs to acquire in order to satisfy a particular objective or set of objectives. An example of an objective might be that a learner should be able to perform file processing activities within the Microsoft Word package, the learning outcomes would, then, be those skills involved in file processing that the tutor identifies as necessary for the user to learn i.e. the learner will be able to:

create a file;

save a file;

open an existing file; etc.

Associated with this refinement of the objectives, is an examination of the characteristics of the learner to determine what knowledge or skills they already possess, this allows the CBL developer to ascertain where the instruction needs to start. Other learner characteristics that need to be identified include attitude, motivation and learning styles etc. It is this phase of the development process that focuses on the pedagogical theories for guidance on how to determine appropriate learner characteristics. This area is vast and offers many methods to analyse learning and learners (see chapter 5). Identifying the pertinent information can prove time-consuming and confusing, especially in the case of inexperienced CBL developers. In many cases, the information can be provided by the tutor who has a clear understanding of his current students and also prospective students, the key lies in the fact that a CBL developer must know that (s)he requires this information and its role in the development process. Information about the learner can ensure the development of "effective, efficient and interesting instructional materials" (Smith and Ragan, 1999).

As the developer moves into the design phase emphasis moves from the learner to the topic and its structure within the CBL environment, the sequence of instruction is determined through an analysis and refinement of the high level objectives through to the very low level learning outcomes. Related outcomes are grouped to form "lessons". Assessment questions may be established from the learning outcomes to judge if the learner has met the original objectives. These three elements: the lesson outcomes; the learner characteristics and the assessment questions form the basis of the implementation. During the implementation phase, the developer devises an instructional strategy and selects and produces the instructional material. These form the lessons, outlined previously, together with any additional media elements that might be deemed beneficial such as sound, video,

animation etc. Care must be taken that any media elements required are either produced specifically for the project or that the copyright holder's permission is sought and obtained for their use.

A technique that is aimed at the implementation phase of a CBL project is that proposed by Gagné & Briggs, (1992). Their nine instructional events are "designed to support the internal process of learning".

The nine instructional events are:

1. Gaining attention;
2. Informing the learner of the objective;
3. Stimulating recall of previous learning;
4. Presenting the stimulus material;
5. Providing learning guidance;
6. Eliciting the performance;
7. Providing feedback about performance correctness;
8. Assessing the performance;
9. Enhancing retention and transfer.

This technique is designed to advise the developer on the events that should be addressed rather than explicitly detailing how the content should be structured. Much work is required to provide a mechanism to address each of the events e.g. developers are exhorted to enhance retention and transfer by providing the opportunity for "*reviews* spaced at intervals throughout weeks and months" or by providing "some *variety* of new tasks for the learner" or providing "variety and novelty in problem solving tasks". Much of this advice is contingent rather than explicit and it is this facet of CBL development models that is most strikingly different to the majority of the conventional software development models.

Throughout the phases of the project, evaluation is a key facet that must be addressed in order to ensure the quality and validity of the final product. Evaluation may be formative or summative. Formative evaluation helps to form the system through feedback from users and summative evaluation is conducted once the product is complete to ensure the system works effectively and efficiently with the planned target users. Evaluation techniques exist which may be used for both the formative and summative evaluation phases (see chapters 8 and 10).

4.6 Conclusion

The comparison between conventional software development and CBL development highlights the areas that are similar for both fields such as the increase in size and complexity of projects undertaken and the phased models that underpin the project life cycle. Evidence from the CBL development methods suggests that some effort has been made to introduce a number of software development activities such as, for example, rapid prototyping to the process but that, in general, the types of techniques associated with software development are lacking in the CBL field. Many of the CBL techniques available for teaching/learning analysis, for example, originate from the pedagogic area or from the field of Psychology in general. Techniques such as needs analysis or needs assessment, learner analysis or determination of learner characteristics rely heavily on psychological principles and approaches such as learning style questionnaires and tests. This aspect of the analysis phase highlights the differences between CBL projects and software projects i.e. the contingent nature of the CBL techniques as opposed to the more prescriptive, or guided, software development techniques. The primary problem with the CBL techniques is the time taken to implement them and the experience needed in the pedagogical theories to analyse learners and didactic requirements, to produce effective and efficient CBL material.

The techniques of software development result in models or diagrams of the system to be developed which be used as a basis for discussions between developers or between developers and end-users. They may be used to hone a developer's understanding of the system to be produced. The models are software and hardware independent and give a conceptual view of the system. The CBL techniques do not provide a corresponding facility for the CBL developer. What is missing from the CBL development life cycle is an analogous technique which provides an explicit, prescriptive mechanism for producing CBL lesson models. These lesson models also need to be software and hardware independent and may also act as a communication tool between developers or between developers and tutors.

The analysis of the software engineering area shows that the central element that is embodied into the techniques that are employed by developers, is the mechanism for analysing existing systems, leading to a design which outlines the key items and their relationship to, or impact on, one another. To this end, one of the key components of a

CBL development model must be a mechanism for addressing the key elements in a topic area, how they are defined and how they related or applied.

Within a CBL project the primary aim is to teach a learner about a particular subject or topic within a subject. In a conventional teaching situation, the tutor will identify the key elements of a topic by specifying the objectives for that topic. To this end, there has been much work in assisting tutors to specify objectives (Bloom, 1956, Mager, 1984, Beard et al, 1974). This practice is also found in the CBL development methods, (Dick and Carey, 1996, Hannafin and Peck, 1988, Knirk and Gustafson, 1986, Jerrold Kemp, 1994, Gerlach and Ely, 1980) and is a fundamental activity in any instructional situation.

Decomposition of topics into their component objectives also facilitates the "chunking" of these objectives into individual lessons which then comprise the relevant sub-topics. Consequently, in any one CBL lesson there may be one or more objectives depending on the individual sub-topic. Thus, just as in the conventional analysis exercise where the "things of interest" are identified and defined, so, in the CBL analysis, the objectives addressed in the individual lessons must also be identified and defined.

So, as in Entity-Relationship Modelling, for example, where for each identified entity, its attributes and relationships are defined, within each CBL lesson the definitions for each concept or keyword, related to the specific objectives, are defined together with their "rules" i.e. details of their application or use. This mirrors the software engineering approach and is one facet of the CBL development model devised in this research. However, not only must the CBL development model fulfil the developer's need for identifying and defining the "things of interest" within the system, it must also address the pedagogic domain of identifying and meeting learner requirements. The combination of these two approaches, one from the software field and one from the pedagogic field is an advance in the area of CBL development. The provision of a structured technique for CBL development analogous to the software engineering techniques and which may be used in conjunction with the existing CBL development methods is also an advance. The aim of this was to devise such a model so that both experienced and inexperienced CBL developers will produce CBL material in a structured, systematic way which adheres to pedagogical principles but which is sufficiently flexible to allow for creativity and individuality. The model also produces CBL lessons which are hardware and software independent and which may form the basis of discussions between developers, and developers and tutors to elicit CBL material, media items or to hone and perfect lesson content and structure.

To this end, the pedagogic area was researched to derive those pedagogic principles which are identified as fundamental to learning and to incorporate them with the software engineering principles to provide a comprehensive model. The model therefore comprises "definitions and rules" plus pedagogic principles. This amalgamation is intended to produce CBL material which is recognisable as excellent as defined by the Coopers & Lybrand evaluation of the TLTP projects (Coopers & Lybrand (b), 1996) where inspirational material was found to be developed through a "synthesis of computing, subject discipline and educational expertise".

Chapter 5

Pedagogy

5.1 Introduction

This chapter reports on the examination of the pedagogic areas which was undertaken to elicit guidelines that purport to facilitate learning. These principles have been amalgamated with the significant software engineering principles to form a comprehensive CBL development model that supports the production of CBL material. The model provides an approach that allows the developer to produce well structured lessons that facilitate the learning process through the adherence to principles which are beneficial to learners.

5.2 Background

The Coopers and Lybrand evaluation of the Teaching and Learning Technology Programme (TLTP) stated that some of the projects appeared "naive" in how they regarded the "complexity of the educational task" with only a minority of projects addressing pedagogic issues. Findings from the report show that it is only through the application of sound pedagogic principles that the quality of the final product could be ensured. It can be seen, therefore, that if a method is to be produced which provides a mechanism for producing CBL material, it, too, must examine the pedagogic area.

In order to produce a thorough CBL development method, it is necessary to investigate the pedagogical theories which prevail to elicit the most effective method(s) of promoting learning. The didactic nature of CBL material is what sets it apart from traditional software. However, since there is no one universal learning theory, the CBL development method will never ensure a particular student will learn a particular topic only that as many learning styles and strategies as possible are addressed to secure as favourable a result as possible.

Despite the fact that no one learning theory purports to offer the complete answer to how learning may be assured, the perusal of the seemingly disparate theories elicits some interesting common features. These include: learning objectives; pre-requisites; problem solving and illustrative examples.

The primary aim of CBL material is to facilitate learning. Learning may be defined in a number of ways, such as: "the knowledge acquired by study" (Oxford Encyclopaedic English Dictionary, 1991) and "changes in the behaviour of human beings and in their capabilities for particular behaviours following their experience within certain identifiable situations" (Gagné et al, 1992). In producing CBL material, the developer is seeking to bring about a recognisable change in the knowledge and/or behaviour of a learner which may be demonstrated to satisfy some external criteria such as passing an examination or completing an assessment exercise or successfully completing a task. Learning, however, is a complex process and is reliant on many variables such as motivation, learning style and prior experience or knowledge, the pedagogic theories attempt to address these issues in order to influence learning in a positive manner. Areas such as motivation and learning style have been widely researched, however, researchers have come to no conclusion. Learners are individual and the factors influencing them diverse. This means that learning theories invariably provide guidelines rather than rules; these guidelines often appearing tenuous to the practitioners attempting to provide a practical solution.

In this research project, an investigation into the pedagogic theories was carried out in order to ascertain the principles enshrined in these theories. The aim was to combine these principles with those from the software engineering field to provide a comprehensive CBL development model which, whilst prescriptive in its nature, was meant to be used to aid in the production of CBL lessons as a foundation rather than as a complete structure. The benefit of this type of model is that it may be adjusted or amended in a specific topic area and learning situation at the discretion of the developer and/or the tutor but which forms a basis for the CBL development. This corresponds closely to software engineering models which, whilst prescriptive, are sufficiently flexible to ensure the creativity of the developer(s) is not stifled. Many recent software development methods are far more contingent than they once were, they offer advice on procedures rather than rules of application (e.g. DSDM, 1995).

The key principles distilled from the pedagogic areas were found to be: learning objectives; prior knowledge; problem solving opportunities and, finally, examples that

relate abstract concepts to real life experiences of the learner. These principles were found in several of the learning theories and appeared, by consensus, to be essential to facilitate learning. The following sections will illustrate the uses and benefits of these principles.

5.3 Learning Objectives and Prior Knowledge

Early work in the area of learning concentrated on the behaviourist approach which investigated how learning was influenced by stimuli, responses to those stimuli and the reinforcement of behaviour. The focus of this work was on how to change or reinforce the behaviour of the learner, (Pavlov, 1927, Thorndike, 1913, Skinner, 1938). This early work was influential for many years and was the basis for the positive reinforcement epitomised by rewards for good behaviour such as increased responsibility and the negative reinforcement characterised by punishments such as detention or caning found in many classrooms in years gone by. An important contribution to the learning experience that emerged from this theory was the emphasis on objectives which provide a detailed account of what is to be taught to the learner and a derivation of learning outcomes based on objectives to demonstrate competence by the leaner. This provision of objectives not only allows the learner to have a clear outline of the topic under tuition but it also allows the developer to delineate lessons and judge the size of the project (s)he has undertaken. Another important feature elicited from behaviourism is the concept of the provision of feedback to reinforce correct responses or alter or explain incorrect ones (Atkinson et al, 1993). This is especially important in the assessment activities of the learner to assist in the learning process. Feedback allows the learner to see quickly which parts of the topic they have mastered and which parts require more work, for example, it can provide clues to where the learner went wrong or recommendations on which part of the topic needs further work.

Aims and objectives are inextricably linked, "aims represent the vision that orientates and motivates the project; objectives identify the deliverables by which the success of the project will be judged." (Davies & Brailsford, 1994). An aim is a "broad statement of intent" whilst an objective is "a sharper, more precise statement of intent" (Race, 1994). Learning objectives serve two purposes, they outline the skills or knowledge the student will achieve, they can also motivate students by fulfilling learning needs. The objectives outline the tutor/teacher's perspective of what they wish the student to learn within a topic, however, the learning outcomes are the activities undertaken by the student which demonstrate the achievement of the objectives. Thus objectives must be:

- Precise, sufficient detail to enable learning outcomes to be derived and judged;
- Necessary, such that without them the aim cannot be achieved;
- Complete, cover all the necessary material (Laurillard, 1995).

In this way, an assessment strategy can be adopted which demonstrates the level to which the learning outcomes match the original objectives.

Several theories make reference to both learning objectives and prior knowledge, it seems obvious that a learner without the requisite prior knowledge will struggle or even fail to learn a new topic e.g. if a student is learning about Entity-Relationship Modelling, they cannot resolve many to many relationships if they do not know what an entity is. To ensure success in the learning experience it is, therefore, necessary for the subject expert (tutor/teacher) to have a clear understanding of what knowledge is needed from the prospective student and to specify this so that both student and tutor have a common starting point.

Ausubel's (1978) "advance organisers" are intended to "bridge the gap between what the learner *already knows* and what he *needs to know* before he can meaningfully learn the task at hand." They are intended to illustrate both the commonality and differences between existing material i.e. that which is already learned, and new material. They are a bridge between prior knowledge and the knowledge to be learned, they show the learner, in advance of the learning, the details of the topic at hand i.e. the objectives of that topic. Obviously, to achieve this bridge, elements of prior knowledge must be present; as Ausubel states "the most important single factor influencing learning is what the learner already knows." Thus, it is crucial that the learner is aware of what prior knowledge is expected as (s)he approaches a learning exercise in order to reach a successful conclusion, that is, to learn the topic. It can, therefore, be seen that the tutor must specify what prior knowledge (s)he expects in order to assure only those learners, who are prepared, attempt to learn the topic areas. Ausubel's "anchoring ideas" are intended to show that components of existing knowledge must be present in order for the objectives (advance organisers) to be relevant.

Similarly, Skinner (1968) sets out the first step in designing instruction as "define the terminal behaviour" i.e. "What is the student to do as a result of having been taught?" To do this it is necessary to break the topic under instruction into its constituent parts so that the individual outcomes which combine to achieve the overall aim can be identified.

Gagné proposes a set of Principles of Instructional Design to optimise learning, these principles consist of nine instructional events, two of which are "inform learners of the objective" and "stimulate recall of pre-requisites" (see section 5.6). He proposes that the learner is informed of the objective(s) and stipulates that an example of this would be to "tell learners what they will be able to do after learning", this, he states, gives the student an expectancy which helps to sustain motivation. He also reiterates the need to disassemble the material to enable any intermediate objectives to be determined so that the logical sequence can be deduced enabling the learner to progress through the material until the overall aim has be accomplished. At each stage, the learner should be presented with the appropriate material to enable them to move on to the next stage. The importance of pre-requisites is seen from two perspectives, firstly, the learner must be at the level where the new knowledge can be assimilated e.g. Arithmetic is generally needed for a learner to be able to master Algebra. Secondly, it is necessary for any pre-requisite knowledge to be recalled "so that it can be recoded as part of the new skill" (ibid) the learner needs to incorporate the new knowledge into their existing knowledge base and the pre-requisite knowledge can act as a bridge to facilitate this.

Anderson (1995) has devised eight principles for the design of tutoring software the first of these informs the designer/developer that "the tutoring enterprise should be informed by an accurate model of the target skill" and this then "allows us to set appropriate curriculum objectives and to properly interpret the actions of the student".

The idea of encouraging learners to organise the information to be learned is reiterated by Travers, Elliot & Kratochwill (1993) and stated to be aided by relating the concept to the student's prior knowledge.

Herbart's (1982) five steps in learning incorporates a preparation stage where prior knowledge is related to the new ideas.

The consensus of providing clear learning objectives and stating any prior knowledge as aiding in the process of learning is impossible to ignore. It is evident that no learning model, whether it be for CBL development or not, would be complete without a section devoted to the pre-requisites and learning objectives to facilitate learning. The learning objectives and prior knowledge can provide the foundation so that the learning experience can be the bridge from "what is" to "what will be".

5.4 Problem Solving

Cognitive psychology arose as a result of the dissatisfaction of some psychologists with the behaviourist approach (Ausubel, 1978, Bruner, 1966). Many felt that it did not fully explain the learning process as it did not address the full range of human behaviour e.g. elements such as memory and thought. Emphasis in this field moved towards the study of memory, attention, perception, language, reasoning, problem solving and creativity. Learning was seen as a dynamic process where the learner actively tries to understand the environment and increase knowledge. A key facet of learning in this theory is the link between existing and acquired knowledge and how that is stored in memory, a student may only acquire new knowledge when they are ready to do so, i.e., when they have the correct prior knowledge to be able to proceed. An example would be that a student cannot learn multiplication and division until they can perform addition and subtraction. This concurs with elements of behaviourism.

Another characteristic identified as important in this theory is that of problem solving and being able to actively apply any new knowledge to acquire experience in diverse situations. This problem solving activity can, also, be linked to assessment to judge ability and achievement levels within a topic. Problem solving allows learners to generalise their knowledge to a wide area rather than assuming the knowledge is specific to a single area. For example, a learner may learn how to add two digits but it is important to be able to realise that skill is also required in situations like adding amounts of money or adding distances on a map to determine the overall length of a journey.

Many theories allude to the beneficial aspect of providing problem solving activities for the learner (Skinner, 1958, Gagné et al, 1992, Travers, Elliot & Kratochwill, 1993, Herbart, 1982, Boyle & Margetts, 1992), these activities can provide the opportunity to apply new found skills and knowledge which can both increase confidence and consolidate expertise, it can also highlight any possible deficiencies in knowledge of which the learner may be unaware. Gagné describes problem solving as "a process by which the learner discovers a combination of previously learned rules and plans their application so as to achieve a solution for a novel problem situation". The learner may need practice at problem solving to refine their hypotheses within each topic area.

In CBL, in particular, an analysis of user performance during these problem solving exercises can provide the lecturer/tutor or content provider with details of any possible deficiencies in the CBL material which may then be rectified. Deficiencies can manifest

themselves as a group of students unable to answer particular problems or questions correctly, from this evidence it is possible to identify which topic the problems cover since the problems are derived to cover the original learning outcomes. The relevant "teaching" area within the body of the CBL may then be altered, deepened or corrected to solve this.

5.5 Illustrative Examples

More recently, work in the area of learning has moved to encompass the unique nature of learners and how they individually represent knowledge. Constructivism (Duffy & Jonassen, 1992, Jonassen et al., 1994, Black and McClintock, 1995) is an approach which advocates the "belief that reason is the primary source of knowledge and that reality is constructed rather than discovered". The key assumptions of this theory are that knowledge is acquired through experience, learning is a personal interpretation of that knowledge and that learning must be an active process in which meaning is derived from experience (Smith & Ragan, 1999). A key contribution of this theory is the idea of placing learning in context by providing problem solving opportunities that are relevant to real life (Tessmer and Richey, 1997), and also in providing meaningful examples to illustrate the relevance of the knowledge to the learner.

Illustrative examples can be used to move from the abstract concept to the generalisation of the principle e.g. if a student is told the definition of an entity, with the help of many examples such as a library case study or a business case study it is possible to illustrate the method of identifying entities in general rather than in particular. Illustrative examples also allow the transformation from abstract concept to "objective reality" without which the usefulness of the abstract concept would be debatable in terms of "the structure of knowledge and for purposes of learning, problem solving and communication" (Ausubel et al, 1978). Gagné asserts that "stimulus presentation for the learning of concepts and rules requires the use of a variety of examples" (see section 5.6). This facility is taken very much for granted in the normal use of everyday language where we are very familiar with the idea of metaphors and analogies to clarify understanding, e.g. as green as grass. The provision of examples is only useful to the student if those examples fall within the realm of experience of that student, e.g. it feels just like weightlessness is a useless analogy if the student has never experienced weightlessness, similarly, it is necessary to try to provide examples within any CBL lesson with which the student is familiar. In the case of undergraduate students, for instance, examples of Information Retrieval Systems can be equated with computerised library catalogues. This theory is also well represented in

many scientific disciplines e.g. in Mathematics "simplicity through analogy" is a recommended strategy, in this approach "one attempts to remove complexities by comparing a new concept or process with some familiar or commonplace situation or activity" (Macnab and Cummine, 1986), the comparison may have only a superficial connection but this may be sufficient to clarify the idea e.g. relating the multiplication of two negative integers to a double negative in English such as "I don't know nothing" meaning "I know something".

5.6 Gagné's Events of Instruction

One of the most commonly adopted strategies for CBL development is taken from Gagné's nine events of instruction (Gagné et al, 1992). These are specified as:

1. Gaining attention

In a computer based lesson this is advised as "presenting initial operating instructions on screen" (ibid.), the problem with this strategy is that if the operating instructions are only accessible via the first screen a student may require assistance as they progress into the lesson and have to page back to the first screen to access it. To this end several CBL systems present the operating instructions when a help button is pressed, this button is then accessible at any point within the CBL lesson. Additional recommendations include "call attention to screen presentation, using words such as 'Look!', 'Watch!' Etc." These need to be handled very carefully so that the user does not feel patronised.

2. Informing the learner of the objective

The importance of this has been outlined in the investigation of the pedagogic theories has been shown to be an essential ingredient in CBL lessons. It is achieved through "stating what the student will have accomplished once he or she has learned", i.e. the learning outcomes. Objectives and learning outcomes reflect the principles of behaviourism outlined in section 5.1.

3. Stimulating recall of pre-requisite learned capabilities

Again, this concurs with the findings above and is linked with the readiness of the student to learn by ensuring that their prior knowledge is outlined and appropriate for them to acquire the new knowledge to be taught.

4. Presenting the stimulus material

This is achieved by presenting stimuli to the student that relate to the topic under instruction, for example, if the student is learning French, they may be given pronunciation examples or asked to read aloud French words from the screen. "Stimulus presentation

63

often emphasises *features* that determine selective selection" (ibid.). Thus students may be presented with material where the key points are highlighted e.g. bold or italicised text or coloured text. In undertaking this activity it is recommended that the student is given many illustrative examples of the concept under tuition. This concurs very closely with the precepts of constructivism outlined in section 5.3.

5. Providing learning guidance

Show examples of the concept to be learned, again described above as a key facet for learning. Learning guidance is not meant to give the answer to the student, more to suggest ways to think of how concepts are related and combined to find the answer. The key is to encourage the learner to think in their own way about the problem and its solution, again a key element of constructivism.

6. Eliciting the performance

This is assessed by asking the student questions or providing problems and getting the student to respond to those questions/problems. The primary aim here is, not only to show the tutor that the student has learned a concept or concepts, but also to convince the students themselves that they really do know the material. This helps to reinforce confidence.

7. Providing feedback

The provision of feedback is related to the elicitation of performance and allows the student to assess their learning by providing information on correct and incorrect responses to the problems given. Through the elicitation of performance and the provision of feedback the student interacts with the system and from this interaction is able to gauge their understanding, this is one of the key differences between a CBL system and a book.

8. Assessing the performance

This is achieved by presenting the student with many opportunities to problem solve and from this to identify and inform the student if they have mastered the topic or not, if they have not mastered the topic they will need guidance on what they must do to achieve mastery. Mastery relates directly to the objectives and subsequent learning outcomes derived for the topic by the tutor. Assessment allows both the student and the tutor to judge whether the objectives have been met.

9. Enhancing retention and transfer

This is achieved by asking questions or presenting problems spaced throughout the CBL material, the student may "practice" their new found knowledge or skills to enhance and strengthen retention. Transfer of knowledge or skills involves widening the experience of

the student e.g. showing that addition relates to numbers, coins, objects etc. The main point here is to provide a variety of problem solving opportunities or tasks.

5.6.1 Summary

These principles encompass the key points from the behaviourist, cognitivist and constructivist theories and are widely used to underpin CBL development. Their strength lies in the fact that they relate so closely to the pedagogic theories outlined in the previous sections but their weakness lies in the fact that they do not address the software element of the CBL material. They do, however, correspond to the findings from the research into pedagogic theories and can act as the basis for the pedagogic component of the CBL development model.

5.7 Conclusion

Gagné's Events of Instruction provide a good initial guide to preparing a CBL lesson and concur with the research into the most important points highlighted from the investigation into the pedagogic theories. It must be emphasised, however, that there is no guarantee that a student will learn from a lesson no matter what principles are adhered to in its preparation. There are many factors that affect learning such as environment, attitude and motivation which are outside the control of the CBL developer. Some aspects of the lesson may positively influence motivation such as providing the goals of the lesson to allow the student to gauge its usefulness. If there were one unified pedagogic theory that assured learning it would have been employed in every classroom all over the world with everyone learning all that was needed. The fact that each student is unique and that each student reacts differently to each subject taught, and prefers to learn in their own way, prevents this from happening. What may be achieved, however, is that every student gets the best opportunity to learn, based on the consensus of research and informed opinion in the field.

To conclude, a CBL development model must address the principles outlined above, namely:

- Define and provide learning objectives;
- Identify pre-requisite or prior knowledge;
- Present the learner with problem solving opportunities;
- Use illustrative examples to relate the concept to real world experience.

These key points will be combined with those from the field of software engineering to provide the comprehensive CBL development model sought.

Chapter 6

The CBL Development Method

6.1 Introduction

This chapter will detail the CBL development model UDRIP which was derived from an amalgamation of two fields, software engineering and pedagogic theory. Fundamental principles from both fields were combined to provide a structured CBL model to assist developers. Chapter 4 outlines the comparison of CBL and software development which highlights the need and rationale for such a model.

CBL development requires that material is produced to teach a particular topic or topics to a diverse group of learners. The development may be undertaken in small single member projects or large consortium based projects. These projects may cover single topics or multiple topics and regardless of size or composition require careful planning (Coopers & Lybrand, 1996).

In a conventional software project, the type of system under development dictates the methods, tools and techniques adopted by the team members. For example, a project to build a large information system might use the SSADM methodology together with E-R Modelling, Data Flow Diagramming, Entity Life Histories and Normalisation as associated techniques.

Similarly, in CBL development, there are several methods (see chapter 4) which may be used to organise the process of producing CBL material. Whilst many of these methods are well documented and researched, there is a lack of associated techniques in this area. The benefits of software engineering methods and the reasons for their introduction have been well documented and include producing software in the most effective way (Sommerville, 1989). The aim is to produce software which is of a high quality within a set time period at a finite cost.

The derivation of a structured CBL development model seeks to reproduce these benefits by providing a model that combines the generic features of software engineering methods

with the key principles from the pedagogic theories. However, software engineering methods alone fail to address one of the primary features of CBL material i.e. its requirement to facilitate learning. To counteract this deficiency, pedagogic principles were used to extend the software engineering concepts to produce the UDRIP design and development model. UDRIP aims to emulate methods which are available for conventional software development, in that: it is structured; it is implementation independent; it enables the development of a "quality" product which meets user requirements.

6.2 UDRIP

UDRIP can be considered both a CBL development model and method. It can be described as a method [1] as it provides a "defined or systematic way" of developing CBL lessons. It may also be considered as a model [2] since it provides a clear structure for CBL lessons for all developers to follow. However, since UDRIP is intended to be used as an adjunct to existing CBL methods, we choose to refer to it here as a model.

UDRIP has the following features:

- It is based on learning theories and incorporates sound pedagogical principles. In this way it addresses the needs of learners by incorporating those principles seen as beneficial in the learning process;

- It offers a structured approach to CBL development. This is analogous to the models and methods available to software engineers (see chapter 4);

- It offers consistency and reliability. This is invaluable to developers who may be producing material in many topic areas and/or within a distributed group;

- It is generic, i.e. it can be applied to many subject areas;

- It can be taught and it is independent of implementation. This enables all developers on a project to learn and adopt the model and also to create designs for discussion prior to commencing implementation.

The need for such a CBL design and development model stems from the fact that developers often need to produce CBL material in a number of topics and often in a

---

[1] A method is defined as "a special form of procedure, the orderly arrangement of ideas" (Concise Oxford Dictionary, 1995). A method is "a mode of procedure, a defined or systematic way of doing a thing especially in accordance with a particular theory" (The New Shorter Oxford English Dictionary, 1993).

[2] A model is defined as "a person or thing used, or for use, as an example to copy or imitate" (ibid). A model is "a representation of structure" (ibid).

distributed project. In order to ensure consistency between the components of a CBL project and also to facilitate communication between developers and between a developer and a subject expert, there needs to be a reliable, effective mechanism understood by all participants. UDRIP fulfils these requirements. It enables CBL material to be produced with a commonality of approach which is particularly important in distributed projects or diverse topics.

UDRIP addresses key questions from both the students' and developers' perspectives. UDRIP stands for:

6.2.1 Universal picture:

Student - where should I be coming from, where am I going, what will I be able to do?

Developer – what are the pre-requisites & expected learning outcomes?

6.2.2 Definitions:

Student - what don't I know?

Developer – what are the keywords & concepts that need to be defined? This is the declarative knowledge "the facts that we know" (Anderson, 1980).

6.2.3 Rules:

Student - how are things applied or used?

Developer – how are the keywords and concepts applied and used? This is the procedural knowledge "the skills we know how to perform" (ibid).

6.2.4 Illustrative examples:

Student - are practical examples available?

Developer - provide embedded scenarios and solutions.

6.2.5 Problem solving:

Student - do I really understand?

Developer – is there an opportunity for self assessment?

6.3 Implementing UDRIP

6.3.1 Universal Picture

The **universal picture** is presented by informing the user of the pre-requisites and expected learning outcomes for a particular lesson within a CBL topic. The pre-requisites show the student the link between any previous knowledge and what is to be presented. It also shows the student what material should have been covered before embarking on the particular lesson in front of them. For example, if students have not covered the lesson on entities in Entity - Relationship (E - R) modelling, they are not ready for the lesson on relationships, thus the lesson on entities is a pre-requisite for the lesson on relationships. This should alleviate the problem of students undertaking lessons for which they are not prepared. An ill-prepared student can quickly become demoralised and irritated. The idea is not to prevent students from entering a particular lesson (holistic learners may like a preliminary browse) but to ensure the student understands what is expected of them. The learning outcomes show what the student will achieve once they have completed a particular lesson, they help the student to identify the goals that a lesson will provide and motivation can be increased if the student perceives a match between the lesson goals and their own goals.

6.3.2 Definitions

The **Definitions** are provided for any keywords or concepts the student may not have encountered before and enable the student to understand subsequent sections where the keywords or concepts are used or manipulated. This reduces the possibility of any ambiguities arising over the use of new words.

6.3.3 Rules

The **Rules** are presented to illustrate how the keywords or concepts may be manipulated, applied or used. For instance, how to diagrammatically represent entities or relationships in an E - R diagram.

6.3.4 Illustrative Examples

The **Illustrative examples** are given to demonstrate the subject area from many points of view, thus giving the student a fuller picture. For example, in the subject of E - R

70

diagramming, it is important the student understands both the diagrammatic and also the textual approach in a number of situations such as a library example or an insurance example to relate the abstract concept to concrete realisation. The examples also provide a means to illustrate both valid and invalid applications of the technique and help the student to identify where mistakes might arise. In learning terms, it is the equivalent of the student "learning by example" (Winston 1992). Winston advocates the provision of both positive, correct examples, and negative, incorrect examples, which allow the student to form an overall picture of the subject.

Within E-R Modelling, for example, a common mistake that students make is to incorrectly identify the membership of a relationship within an E-R diagram. Membership is represented by solid or dotted lines to depict a mandatory or optional relationship. These are often transposed to form an incorrect interpretation. Membership rules can be given to the students using a number of illustrative examples to highlight both the correct and incorrect interpretations.

Examples often tend to be passive and are used to illustrate a point or relate the abstract to the concrete. Within CBL it may be that examples may be solely textual or may utilise graphics, animation, sound or video or a combination of all the media elements. A particular idea may be made more understandable by adopting this approach e.g. illustrating the fetch-execute cycle of a CPU (central processing unit) is much clearer using an animation than either text or static graphic. Interactivity can be introduced by allowing the student the ability, for example, to start and stop an illustrative animation or entering text which is then incorporated into the example. For example, in a lesson on many to many resolution in E-R Modelling the student may enter the name of an intersection entity which is then used during the resolution process.

6.3.5 Problem Solving

Problem solving can be used in two ways. From the point of view of the student it can highlight weaknesses in their knowledge and in the application of their knowledge. From the point of view of the lecturer it can show how well the student has learned the material. Problem solving can be in a section on its own where problems relating to all the concepts in a particular lesson are covered, but problems may also be dispersed throughout the lesson to provide practice in this activity. In the body of the lesson problems may be context specific. However, in the problem solving section the problems may cover more

than one concept. Providing two opportunities to practice problem solving can re-enforce the knowledge acquisition within a particular topic.

In addition, the results from the problem solving exercise can provide the developer and the tutor with useful information. From the developer's point of view, consistent poor results in a particular part of the problem solving section can highlight possible weaknesses in the presentation of that material within the lesson and from the tutor's point of view, if there are only a few poor results this can pinpoint those students who may need additional help.

6.4 Conclusion

The model need not be exclusively used for CBL presentations. It can, for example be used in the preparation of many teaching / learning support materials or to provide a comprehensive presentation during a workshop or lecture, where problem solving opportunities may be provided by interacting with the audience. It does, however, provide a framework for developers to produce CBL material quickly, efficiently and within a sound learning environment.

The UDRIP model does not preclude the use of Human Computer Interface (HCI) standards nor the use of multi-media elements, it acts as a sort of skeletal structure which combined with the HCI, the multi-media and the subject material, form the whole CBL package. The UDRIP model is intended to be used in conjunction with any of the CBL development models in the stage where the developer is expected to "develop and select instructional materials" (Dick and Carey, 1996), "construct prototype" (Tripp & Bichelmeyer, 1990) etc. (see chapter 4).

It should be noted, however, that the UDRIP model is not intended to be overly prescriptive to the extent that developers are expected to include every section in every lesson. However, it is prescriptive to the point where they should have considered all the sections and elicited all the information from the subject expert to determine the final structure of the CBL lesson. If the decision is then made to omit a section or part of a section then this may be carried out in collaboration with the subject expert and is a conscious decision. An example might be where one lesson may just provide a brief overview of the topic as a whole in which the subject expert may only wish to provide pre-requisites, the topic overview and some illustrative examples. Also the sequence of the sections need not necessarily be rigid, for example some CBL lessons may provide the

problem solving section first as a diagnostic test. This particular structure may be used by the students as a form of self-assessment of their knowledge or by the tutor to determine if the lesson is warranted for individual students.

The aim has been to produce a CBL design and development model which supports the developer during the CBL project but which also supports the learner through the production of pedagogically valid lessons. It is also intended to ensure that both the tutor and the developer have a clear idea of what the lesson includes and to reach a consensus on the most appropriate structure for that lesson.

Chapter 7

CBL Implementation

7.1 Introduction

This chapter will discuss the application of the UDRIP model to the development of CBL material to teach Entity-Relationship (E-R) Modelling. Due to the nature of the template adopted throughout the W.I.S.D.E.N. consortium (see chapter 2), it was necessary to ensure the sections within a lesson had a coherent, logical structure. The UDRIP model was used to provide a logical order which the serialist type student could move through in a sequential fashion. It also provided individual coherent sections which addressed the needs of holistic learners to browse the topic for an effective learning experience.

As a test of the applicability of UDRIP, it was incorporated into the template to provide structured lessons in E-R Modelling. However, to be able to adopt both the template and UDRIP at the lesson level, it was necessary to refine each topic from its original high level aims, through broad objectives, to very low level learning outcomes to enable the "chunking" necessary to delimit individual lessons. This process also defined the structure of the topic as a whole since it provided an analysis of the material to reveal which part of the topic preceded and, hence, became a pre-requisite for any other. The refinement of the objectives was not an intuitive task and required a considerable amount of effort to reach the required granularity, however, only at this point can the scope of the topic, and individual lessons within that topic, be defined.

7.2 Refinement of Aims and Objectives (Task Analysis)

This top-down analysis activity is carried out to identify the low-level or "enabling" objectives which contribute to the successful completion of the overall aim. There are two types of task analysis, procedural task analysis is designed to highlight the steps necessary to perform a particular task e.g. changing the wheel of a car. This type of analysis usually results in a list of steps the learner must follow to be able complete a task. Learning task

analysis is used to reveal which objectives are pre-requisites for subsequent objectives. The outcome from the learning task analysis is a "learning hierarchy" (Gagné, 1992). Using the procedural task analysis to break down the high level aim of E-R Modelling into low level objectives results in the following:

High level Aim:

- To provide a basic understanding of the data modelling process to enable the construction, using a specified notation, of data models from a well defined system scenario with limited scope.

High Level Objectives:

- given a well defined system scenario/specification, to be able to carry out a data requirements analysis and construct a data model using a specified notation.

- given a data model for a system, to be able to demonstrate how (or not) the model meets the requirements of the specified system.

Medium Level Objectives

To be able to:

- define the following terms associated with data modelling: entity, relationship, attribute, relationship degree, relationship membership (optional, mandatory), candidate key, primary key, foreign key;

- identify and define entities, relationships and attributes from a specified system scenario;

- construct an entity-relationship data model of a system that meets it's processing requirements.

Low Level Objectives

From a system scenario/specification be able to:

Entities

- define entities;
- identify entities;
- label entities;
- identify entity types;
- identify entity instances;
- distinguish between an entity type and an entity instance(occurrence);
- use the correct modelling notation to diagrammatically represent entities;

Using the learning task analysis it can be demonstrated that the topic of E-R Modelling falls naturally into 3 sub-topics: Entities, Relationships and attributes. However, the resolution of many to many relationships, which falls within the relationships objectives, requires a knowledge of the details of entities, relationships and attributes and for this reason needs to be taught after all the others. Hence, a logical structure for the topic can be represented by the hierarchical structure Figure 7.1. This figure shows that the student, firstly, needs to learn about entities, then attributes and relationships and, finally, many to many resolution.

The diagram represents the sub-topics running from top to bottom so that, entities should be taught first, then either attributes or relationships (neither is a pre-requisite for the other for understanding) and finally, many to many resolution. Each level of the diagram, starting at the top, represents a pre-requisite lesson for subsequent lessons. It is not intended to force the student into this path through the lessons, holistic learners may benefit from the option to browse around the topics, but the structure is provided as an advisory one.
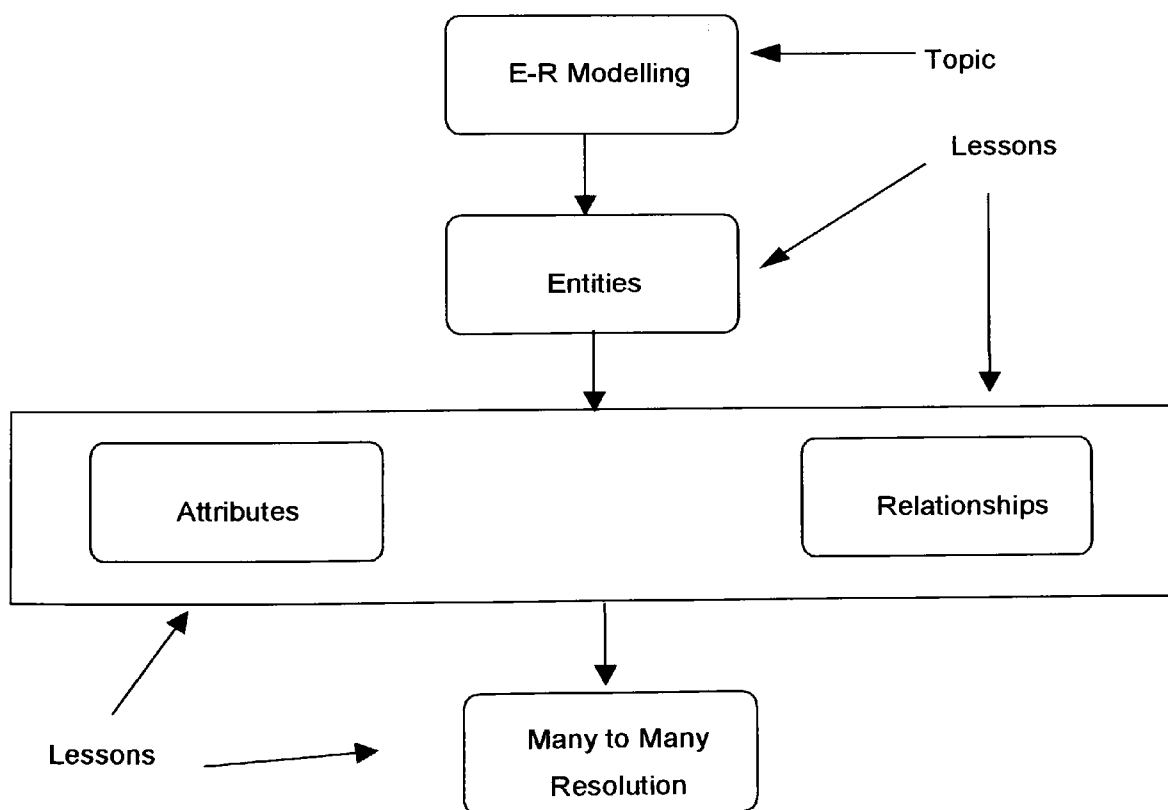
Figure 7-1. Topic Structure.

Many students may come to the topic of E-R Modelling with no previous experience, however, students who have studied the topic before could well be aware of Logical Data

Analysis or Data Modelling. They may be unaware that E-R Modelling or Diagramming is the same as Logical Data Modelling or Logical Data Structures. An overview is provided to give a synopsis of the subject so that all the students plus the tutor are sharing a common terminology. The overview is also used to provide a small case study which shows a "real life" situation where E-R Modelling may be used. The diagram of lesson hierarchy, thus, becomes as shown in Figure 7.2. The overview may be studied prior to the main lessons or as well as the main lessons. It does not become a pre-requisite for any other lesson but merely provides additional material such as, in this case, a case study or a summary of the topic as a whole.
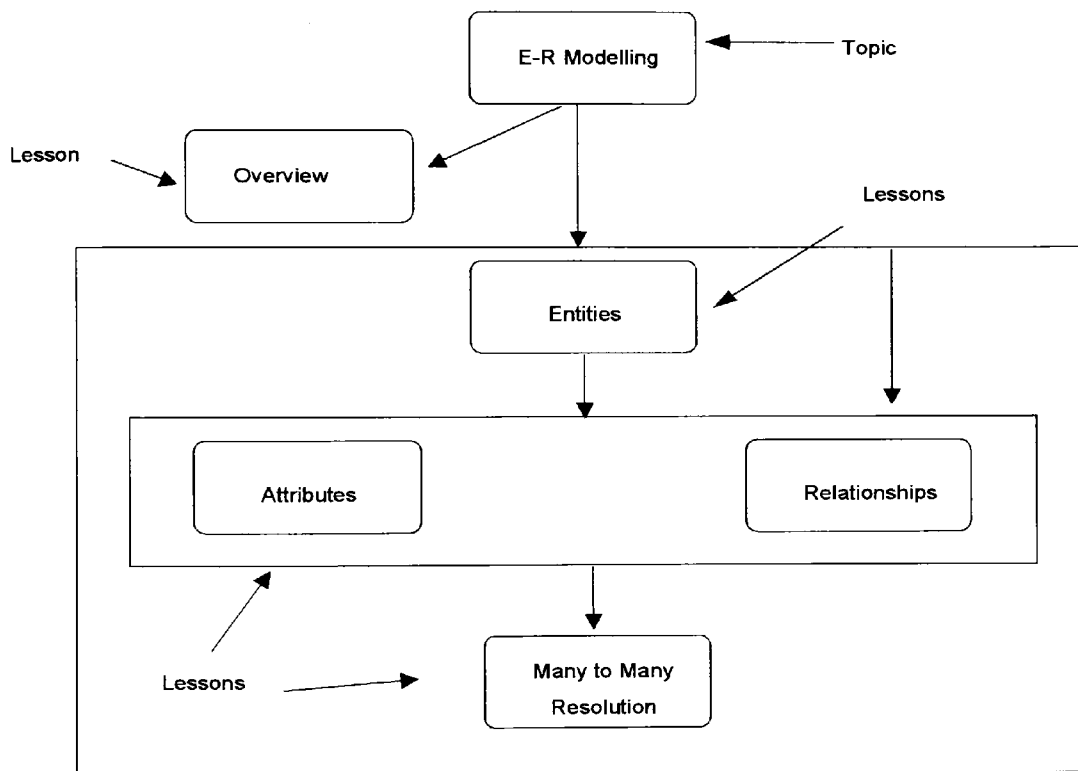
Figure 7-2. Lesson Hierarchy.

In this diagram, the overview is not shown as a pre-requisite to any of the main sub-topics for E-R Modelling, it can, however, be used as an introduction to the overall topic.

7.3 The Template

As mentioned previously, the consortium members agreed a number of standards and the lead site produced a template for the user interface. An example of the screen layout of the template is shown in Figure 7.3.
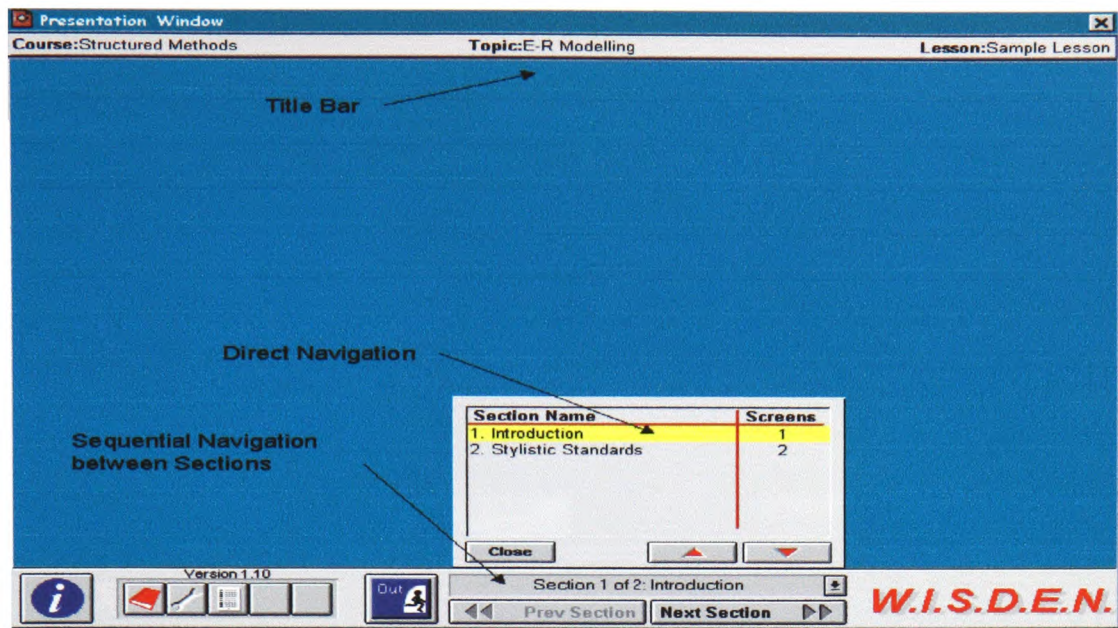
Figure 7-3. W.I.S.D.E.N. Template.

Thus the content for each lesson was to be displayed between the title bar and the navigation bar for each consortium member. Within the template itself various elements could be customised for each individual developer. These were: the section titles; the course name; the topic name and the lesson name. The help button provided the version; the author and an abstract, together with details of all the buttons on the screen, as in Figure 7.4.
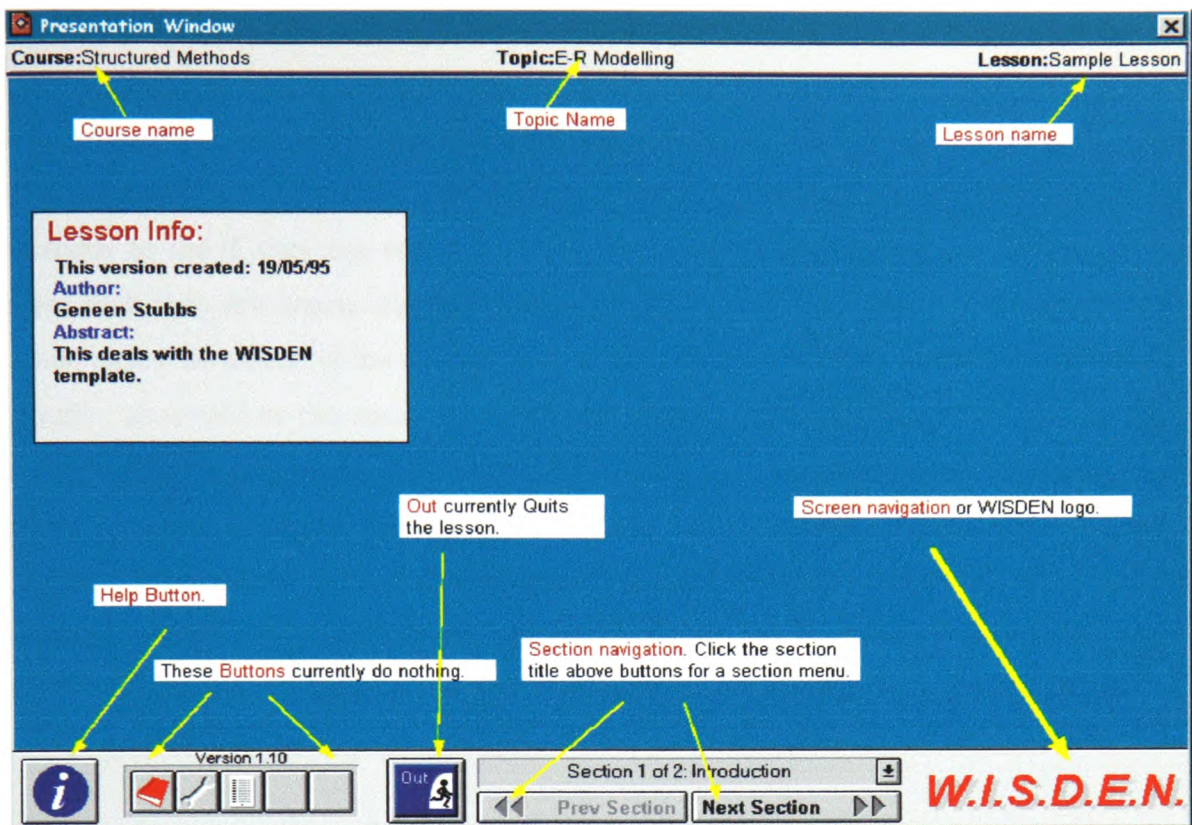
Figure 7-4. Help.

Each developer could present the content in the way that was felt most appropriate for that particular lesson. This still left the question of the best way to structure the content to take advantage of the composite navigation available in the template. It is at this level that the UDRIP model can be applied to give a logical approach to the development. Each section of the model can be implemented as one or more sections within the lesson i.e. the total lesson may have more or less than the five sections of the model, for example the rules for Entities may cover Entity type rules; occurrence rules; diagram rules and definition rules all of which may need a separate section. In the lessons developed at Glamorgan, the illustrative examples were used where and when necessary, mainly in the sections on definitions and rules. However, if a developer wished to have them in a separate section that is also possible.

The order of the sections is intended to be advisory rather than mandatory and should the developer wish to rearrange them, it is still possible for the student to have a beneficial learning experience. Take, for example, the scenario where the student is presented with a problem solving section first, if the student/learner is able to complete all or most of the problems correctly, the result can be used to advise the student on whether or not they need to study the lesson at all. The subject expert can advise the developer what level of

attainment is suitable for each of the lessons and once the student achieves this they may move on. This can help motivation by eliminating those areas where the student already has the requisite knowledge. Another problem solving section could also be incorporated towards the end of the lesson which contained those problems the student had answered incorrectly to see if they can register an improvement. This can increase motivation if the student feels they are improving their performance.

In some cases, however, if the student fails to answer a reasonable number of problems correctly , as would be the case if they were a novice, motivation can be decreased and the student can feel demoralised. Within the W.I.S.D.E.N. lessons the problem solving section was not used as a pre-test. However, it was prepared as a single section within the lesson to enable those students who wished to use it as a pre-test that opportunity. The key benefit from the UDRIP model is that it ensures the developer has considered all aspects of the topic and made reasoned decisions, in collaboration with the subject expert, before embarking on the actual implementation.

7.4   Implementation of UDRIP

7.4.1 Universal Picture (Pre-requisites and Objectives)

Another factor that can de-motivate students is attempting a lesson for which they do not have the necessary prior knowledge to be able to complete it. To eliminate this prospect, the first page the student sees within the lesson, outlines what prior knowledge is needed to understand the lesson. Also in this section is a page which outlines the objectives for the lesson to try and increase motivation by matching student goals to lesson objectives. The student was not, however, prohibited from accessing the lesson with or without the pre-requisite knowledge. Many learners like to browse lessons to get a feel for the subject, see Figure 7.5.
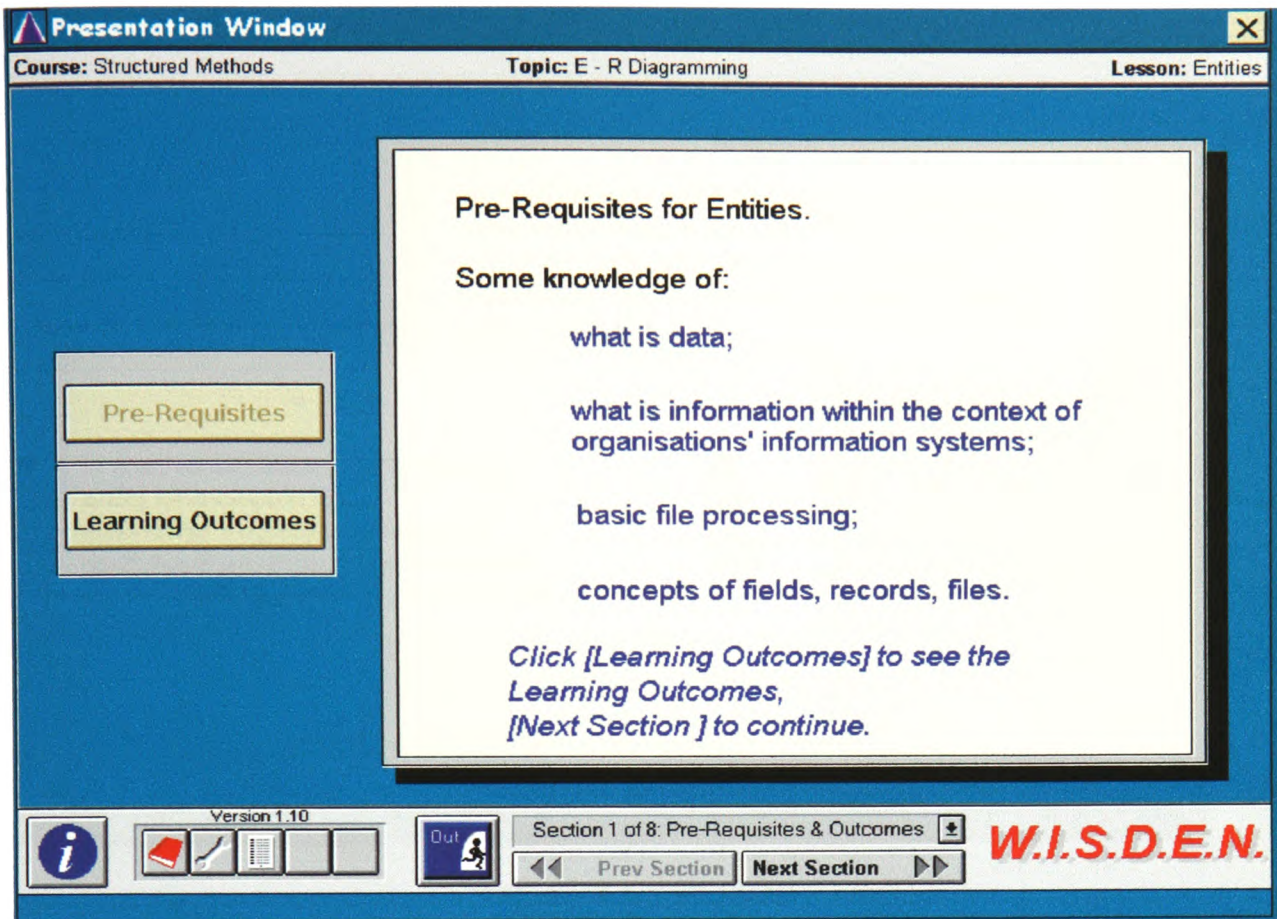
Figure 7-5. Pre-Requisites and Objectives.

Figure 7.5, shows the W.I.S.D.E.N. logo which can be seen instead of the screen navigation buttons, this was a standard agreed within the consortium so that where there is only one single screen the logo would be displayed rather than have two greyed out buttons. Buttons that were not available to the student were greyed out (as the section navigation button - "Prev Section" above) to conform to Microsoft's standards, details available on-line at (http://www.microsoft.com/win32dev/uiguide/default.htm, last updated 7/3/96).

7.4.2 Definitions

The definitions of keywords and/or concepts are necessary to clarify the rest of the lesson. Once they are defined they can be used with confidence, ensuring that both the student and the tutor are working with the same understanding. Throughout the sections on definitions and rules illustrative examples are used to give context to the abstract ideas, Figure 7.6.
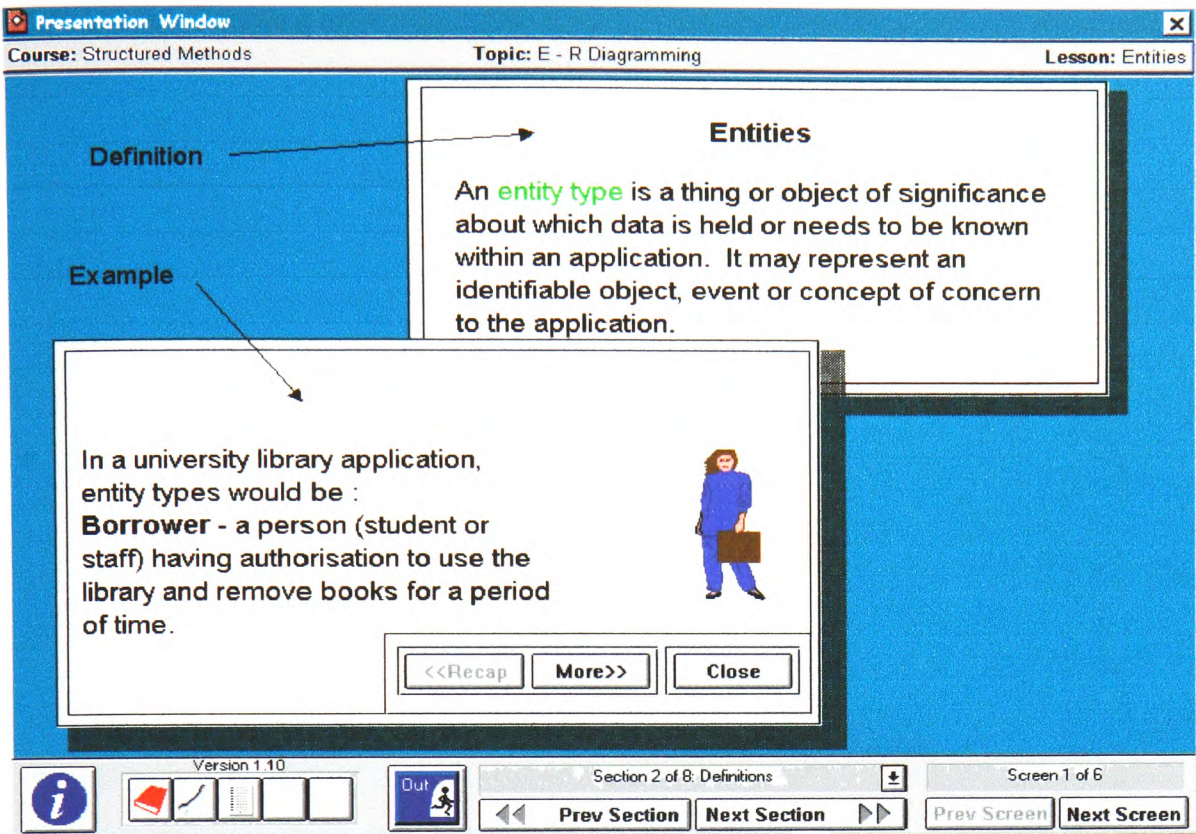
Figure 7-6. Definitions.

Similarly, in the section on the rules, Figure 7.7;

7.4.3 Rules
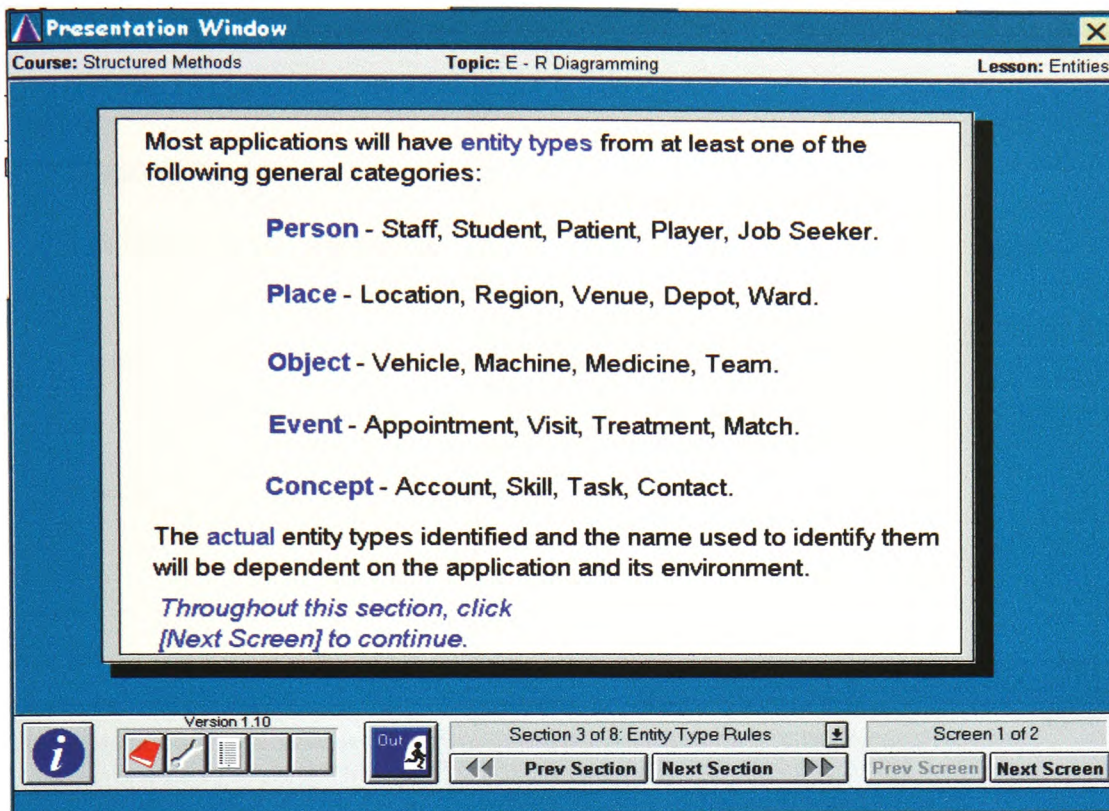
Figure 7-7. Rules.

The format of messages to the user was agreed within the consortium to be blue, italicised text with button names enclosed by square brackets (as per the [next screen] message figure 7.7). Background colours for the screen were chosen by the individual sites.

7.4.4 Problem Solving

This section provided practice or a self assessment facility for the student in the E-R Modelling lessons, figure 7.8. The score was not retained but if this feature was required or found to be desirable for a particular project, it could be easily implemented. Retention of scores can sometimes increase motivation by providing the student with evidence of improvement but it can also become the entire focus of the lesson. If the problems provided in this section are static i.e. the student sees the same problems each time they move through the lesson, it might be that they achieve the correct answer by remembering previous attempts instead of learning the topic under instruction. To limit this effect, the problems can be provided randomly from a problem bank, they can be comparable in terms of difficulty but can facilitate learning by providing a wide problem domain.
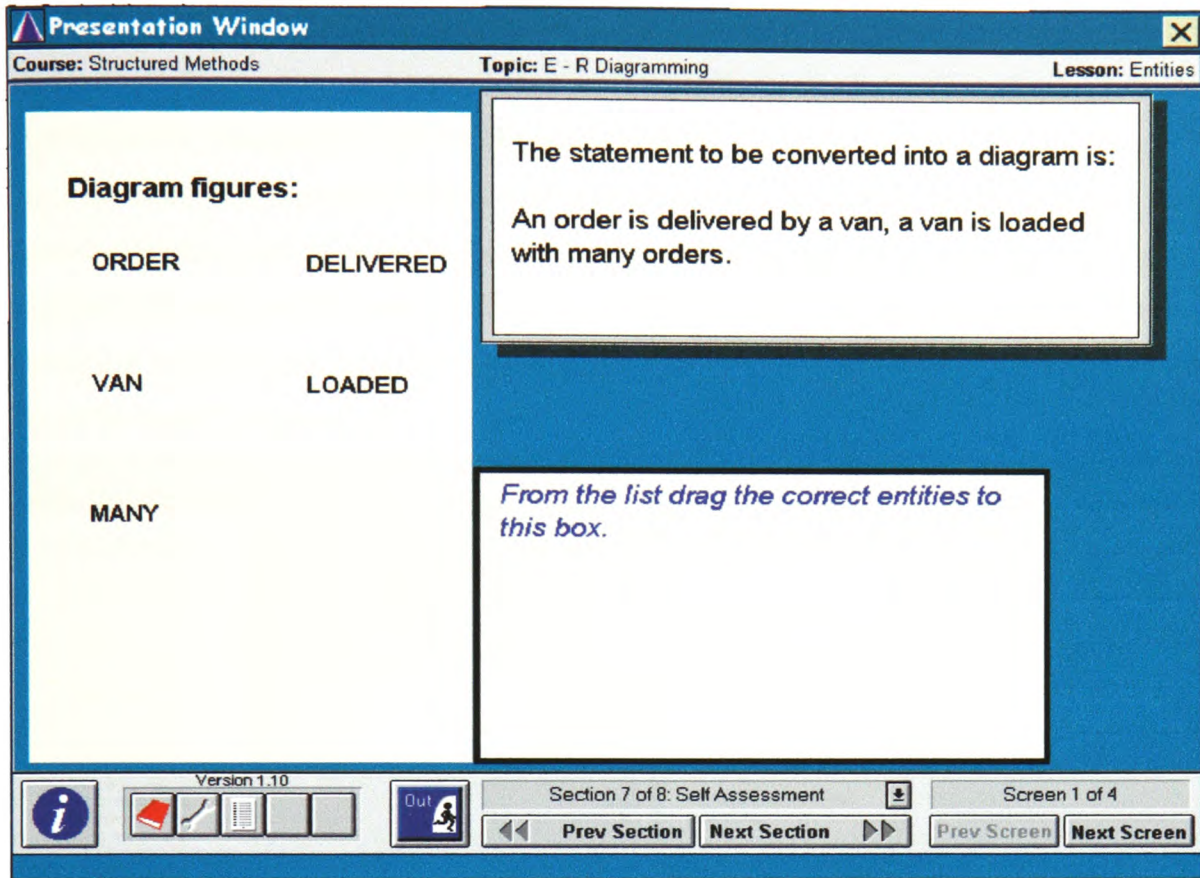
Figure 7-8. Problem-Solving Example.

In this section, a number of problems were used such as drag and drop (figure 7.8), multiple choice, true/false, text entry (completing or filling in the blanks) and quizzes. Feedback was provided immediately on incorrect answers and the student was allowed a number of tries, either a fixed number (usually used in text entry examples) or any number if there was a limited choice e.g. multiple choice or true/false, figure 7.9.
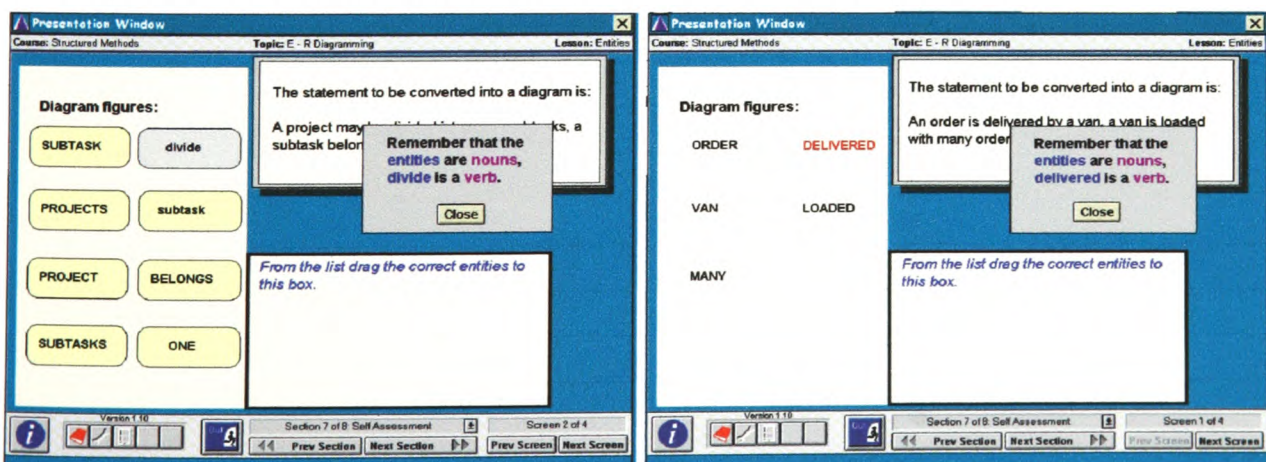
Figure 7-9. More Problem-Solving Examples.

In these particular problems the student has to close the feedback window before they can proceed, this is deliberate and is aimed at re-enforcing the learning by explaining incorrect choices and forcing the student to focus on the feedback window. The choice made is also highlighted by changing its colour (here the student chose divide and delivered figure 7.9), this associates the incorrect answer with the choice made, another re-enforcement feature. Another feature used to provide feedback was an animated person, in this case an animated student, this was used to provide some fun whilst re-enforcing incorrect answers, the animation would shake his head, wag his finger or in extreme cases his eyes would "pop out of his head", figure 7.10.
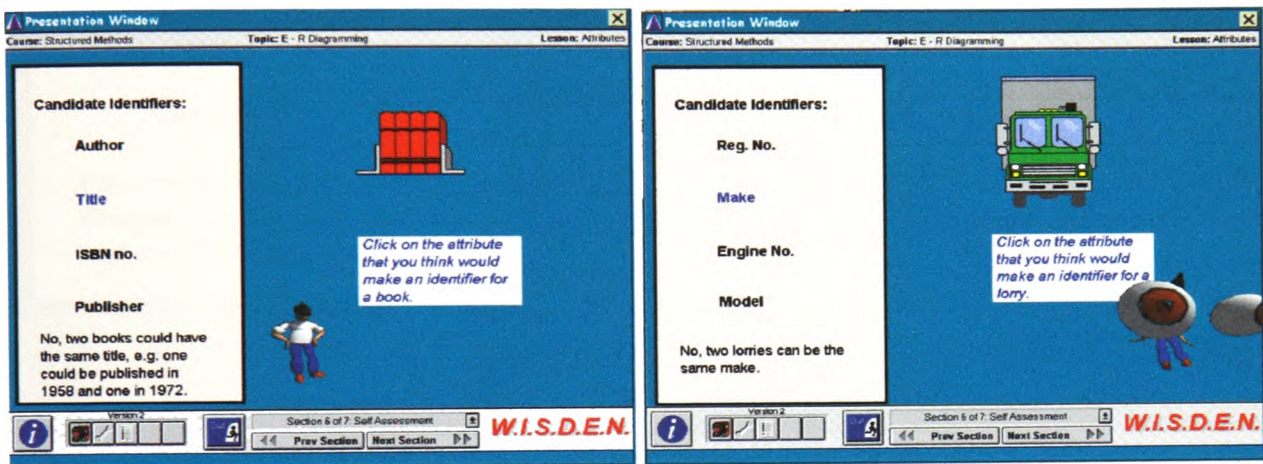
Figure 7-10. Animation for Re-enforcement.

When the correct answer was given the animation would clap and a well done message would appear, figure 7.11.
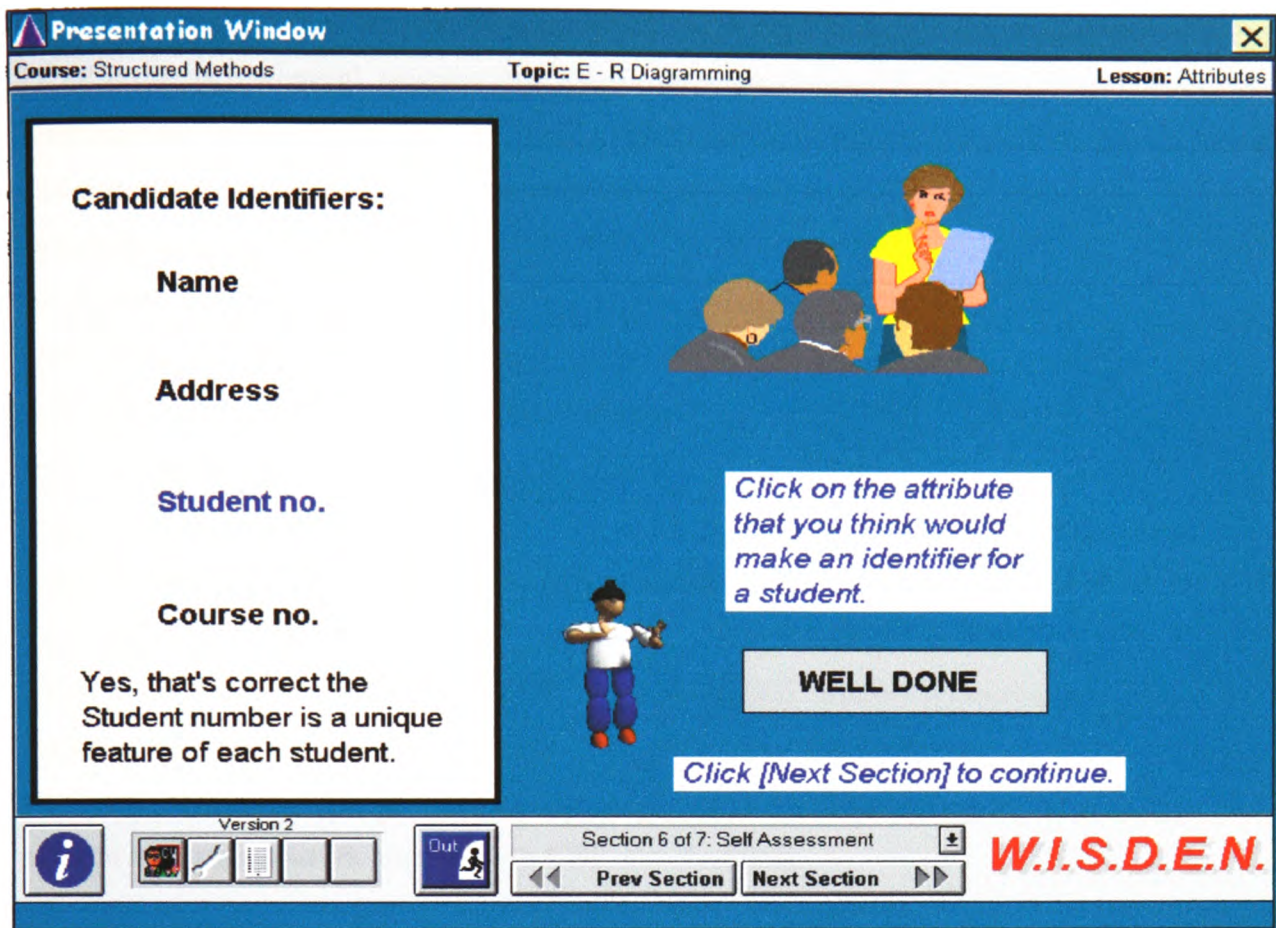
Figure 7-11. Correct Answer.

The student was also given feedback explaining the correct answer further, figure 7.11, this was another re-enforcing feature. The correct answer was again coloured to link the answer with the choice made. The animation was used in moderation to retain its impact, its use was also limited to the self-assessment or problem solving section as in the examples above.

7.5 Conclusion

The UDRIP model was initially used to structure a lesson on Entity-Relationship Modelling as explained above. Many of the features used such as the animated figure, the position of text or graphics, were not imposed by the model. The UDRIP model acted as a skeletal structure on which the "flesh and features" were moulded to form the overall lesson. The key aspect of the adoption of this model was that it ensured all elements of the lesson were considered before development began. It also provided a means of communication between the developer and the content/subject expert enabling the developer to request what was needed to construct the whole lesson. This frees the

The main drawback with developing CBL material is the time taken to develop it (Capell, 1995). Anything that can impact on that can only be seen as beneficial. To conclude, the UDRIP model offers an advisory working structure for the content of CBL lessons for developers, especially novice developers (evidence from the W.I.S.D.E.N. project, and TLTP projects in general, suggest that more experienced developers have already become comfortable with their own ad hoc model which they adopt again and again). It also offers a communication mechanism between the developer and the content expert where these are two separate participants in the project. The UDRIP model does not, however, constrain the creative abilities of the developer nor the media contributors.

The UDRIP model is based on pedagogic and didactic theories combined with the principles seen as beneficial within the Software Engineering area. UDRIP provides a skeleton structure for lessons which is independent of implementation. This skeletal structure may be completed using HCI standards, appropriate media and topic content, suitable examples and problem solving opportunities. Content may be included by consultation with the subject expert to ensure it contains material at the correct level and which is clear, complete and consistent. Overall, UDRIP enables a developer to focus on constructing CBL lessons by following clear guidelines. The instructional design methods may be followed and UDRIP may be integrated into the design/development phase to provide a comprehensive approach to CBL development.

developer to concentrate on choosing media elements, addressing HCI issues such as the position of text, graphics, animation or in cases where additional media elements are used the audio and/or video.

Chapter 8

Evaluation

8.1 Introduction

This chapter reports on the results of an evaluation exercise conducted on the CBL material produced using the UDRIP development model. Initial, informal evaluations were conducted on the W.I.S.D.E.N. prototypes by an educational Psychologist which resulted in the formation of a standards group and examination of the structure of the material to address criticisms of this early work. Research at Glamorgan lead to the derivation of a CBL development model UDRIP which was used to build CBL material to teach E-R Modelling. This material was evaluated, in the first place, with respect to its usability to ensure the users could access the material in a manner that suited their learning styles and needs. Subsequent evaluations were planned to test the teaching/learning effectiveness of this type of resource to ensure that it was a useful learning experience for the user. This chapter details the evaluation techniques available and those applied. The results of the evaluation exercise are presented together with their impact on the CBL material.

Within the W.I.S.D.E.N. consortium each member was instructed to set out the learning outcomes for their particular topic and asked to develop prototypes in that area. Evaluation of these early prototypes, by an independent educational Psychologist, revealed a diversity of styles and criticism of their engagement and effectiveness. Lack of standards with regard to navigation, fonts, colours etc. was an obvious feature of these early prototypes which needed to be addressed, as were the pedagogical issues which had been uncovered. It was also necessary to ensure the needs of the target audience would be met with respect to usability, effectiveness and level.

As a result of this review of the development process and subsequent prototypes, a standards group was formed which agreed several standards for consortium developers to apply within their CBL material, to ensure a common look and feel (see chapter 2, Introduction to WISDEN), also the adoption of the UDRIP model provided a clear

coherent structure for the courseware. Endorsement for the use of a teaching and learning model came from a Coopers & Lybrand report commissioned by TLTP, which stated that many projects had underestimated the complexity of the educational task and little regard had been paid to pedagogical issues. They determined that a characteristic of an effective CBL product was that it should be supported by a clear model of learning and teaching (Coopers & Lybrand, 1996).

With the agreement of the standards and the adoption of the UDRIP teaching and learning model, each member produced a refined prototype which was presented for initial assessment within the consortium group. There was agreement that a reproducible method for CBL development would be invaluable and each member gave the prototypes careful consideration. As has been described earlier, application of the UDRIP model provided a mechanism for structuring CBL lessons without constraining the creativity of the developer within their specialist subject area, as such it was readily accepted as a useful tool and subsequently, the model was used to underpin the CBL lessons produced throughout the consortium (Norcliffe, 1996).

One of the most interesting outcomes from the initial presentation of the CBL development model was the common areas which many members had incorporated into their design but which they had never formalised into a usable, reproducible method for CBL development. For instance, South Bank University had a section in their lessons called "About This Topic" which contained the pre-requisites and learning outcomes for that lesson. This concurred with the UDRIP Universal picture section which also specifies pre-requisites and learning outcomes.

8.2 Evaluation

The evaluation of a CBL product involves three activities:

1. understanding the product's teaching and learning objectives with respect to its intended audience (Laurillard, 1995);

2. gathering evidence regarding its "look and feel" and usability (Barker & King, 1993);

3. judgement of the product based on the evidence gathered from the previous two activities (Elthe, 1995/96).

Evaluation is required to assess the quality of the CBL product. Any CBL product has two main components:- the software component as it is a computer based activity; the didactic

component as its purpose is to instruct. Therefore, evaluation must address both facets if the quality of the final product is to be assured. Thus, the evaluation needed to address both the usability of the system, as per a traditional software product, and also its learning effectiveness. The learning effectiveness is influenced by both the usability of the system and the material contained within the CBL lessons, its content and form. If the material to be learned is incorrect or incomplete, the overall CBL system can be compromised no matter how good the usability. If the usability is not satisfactory the learning effectiveness can be undermined e.g. if a student misses a section on problem solving due to poor navigation structures this could impact on how well they learn the material since problem solving is propounded to be an essential element of the learning process. Evaluation exercises are, therefore, undertaken to examine all the relevant aspects of the CBL system. Each exercise needs to be focused to provide pertinent information from which conclusions about the final product can be drawn. The usability and learning effectiveness of the system are addressed by evaluating the material with student users but the content evaluation is conducted by subject experts.

There are two types of evaluation that may be conducted, these are: Formative - used during development to "form" the system; and Summative - used to test the system against normal practice (Kemp et al, 1996, Smith & Ragan, 1999). Formative evaluation is conducted throughout the development phase of the project. Its results provide feedback which is used to perfect the final product. This iterative approach is similar to the approach advocated in most software engineering methods (see chapter 4). Summative evaluation is conducted once the product is considered complete. It is carried out with representative users as the material is intended to be used i.e. with real users in a real environment.

8.3 Evaluation Techniques

For the CBL developer, only through evaluation will feedback be gained that will enable improvement in the quality of the CBL product. To this end, there are a number of evaluation techniques available to achieve the required outcome, these include but are not limited to: Questionnaires; Observation; Interviews; Video; Pre- & Post-tests; Expert evaluation (English, 1991, Laurillard, 1995). All of these techniques, with the exception of the expert evaluation are directed at the user's perception of the system. Before usability and effectiveness evaluations can commence, however, it is essential that there is a consensus amongst the subject experts that the content is correct. Therefore, the initial

evaluation is conducted by the subject experts and subsequent evaluations are conducted by students.

8.3.1 Expert Evaluation

Obviously, for a student to use CBL material with confidence, they must be assured that it will teach them the correct information, be unambiguous and consistent with other methods used to teach the same material e.g. lectures. An expert evaluation ensures these criteria are proven. This particular evaluation exercise is undertaken by a person who is an expert in the subject area of the CBL lesson. This can include the person who has commissioned the CBL material but should extend to people outside the project to ensure an objective view of the material and guarantee the content is complete, consistent, clear and correct. An added benefit of an expert evaluation which includes several subject experts is that they are then more amenable to including the CBL material as part of their teaching methods since they feel confident of its quality and, in part, its implementation. To conduct this evaluation, the subject experts are presented with the CBL lessons and are asked to comment on aspects of the content e.g. are the diagrams correct; are the definitions and rules correct and clear etc. Once these evaluations are complete, amendments can be made to ensure that subsequent evaluation results are not as a result of incorrect content e.g. the learning effectiveness is judged by the achievement of learning outcomes, if these learning outcomes are not achieved due to incorrect facts in the CBL lesson then the judgement of the effectiveness of the CBL to facilitate learning is directly affected.

8.3.2 Usability Evaluation Techniques

Usability is a key area that needs to be addressed in an evaluation exercise, if the content, examples, self-assessment facilities etc. are first class but the student does not know how to use the CBL lesson, e.g. they cannot navigate in order to find all the information or the sections within a lesson, then they can find it impossible to learn the material. The key facets related to the usability of a CBL lesson are the interface and navigation. In order to address usability, it is necessary to try and make the interface as transparent as possible and the navigation as useful but unobtrusive as possible to allow the student to focus on learning the material. This is a feature that has been recognised in the GUI (Graphical User Interface) environment of the PC where the familiar metaphor of the office and

desktop are adopted to try and make working on the computer easier for the user (Shneiderman, 1997). Of the techniques available for usability evaluation some of the more common are: Questionnaires; Observation; Video; and Interviews. Questionnaires may be constructed to provide both qualitative and quantitative data, observation allows for the use of the system to be examined either informally or formally; similarly video allows the system use to be monitored but for the observer to collate results after the evaluation exercise is over; and finally, interviews can elicit learners' opinions and assumptions.

Questionnaire

Questionnaires are used to try and elicit attitudes and opinions about the CBL system, they consist of questions that direct the user to examine specific parts of the system e.g. screen design, graphics etc. They may comprise open or closed questions where closed questions provide quantitative data and open questions qualitative data. Quantitative data allows the developer to have an overall feel for the item under consideration e.g. it allows for the result that 90% of all the participants felt the graphics were of the correct size. Qualitative data provides opinions in more depth e.g. "I particularly liked the drag and drop activities". The closed questions do not allow the user to express an opinion but do allow them to rate a feature, this can be achieved using a Lickert scale. The Lickert scale may be used in a number of ways for instance it might range from very poor to very good e.g. 1. Very poor, 2. Poor, 3. OK, 4. Good, 5. Very good. It can, also, be modified to allow the user to respond to questions such as "how relevant did you find this material to your course of study?" 1. Very relevant, 2. Relevant, 3. Little relevance 4. Not relevant. The Lickert scale may have any number of responses but is, generally, restricted to four or five to prevent the answers becoming overly complicated.

The open questions on the other hand allow the user to express their opinion in their own words and often provide the developer with more pertinent information. As an example consider the closed question which asks about the screen design, if a student rates the screen design as very poor on a Lickert scale, the developer may not find out why it is rated so poorly unless they provide an opportunity for a further response. In an open question the student may well explain why they find the design so poor, this helps to determine if there is a real problem over the whole group or an isolated problem with an individual and how serious that problem is. Certain findings require immediate action if they are reported by particular individuals, these individuals are those respondents who are

colour blind. Colour blindness is a relatively rare condition affecting approximately 10% of men and 1% of women and is likely to affect very few, if any, people in a single evaluation exercise, therefore, if responses relate to this condition they must be dealt with despite the fact that only a single respondent may have highlighted them. Since the evaluation exercise is designed to refine the CBL material from the point of view of usability, this will ensure all future users should have an equal opportunity to use the system to learn.

Thus, it can be seen that closed questions provide quantitative data whilst open questions provide qualitative data. Quantitative data allows the developer to provide a numerical or statistical analysis from the evaluation exercise which can be used to demonstrate quality whilst qualitative data allows the developer to judge the severity or triviality of any problems revealed during the evaluation activity and whether these warrant correction or not.

Observation, Interviews and Video

Observation

Observations are conducted to determine how the students are using the system, if there are any difficulties with respect to the use and what is affecting that use. Observation may be casual, informal or formal, casual observation has no clear structure and entails the evaluator, who may or may not be the developer, generally observing the students using the CBL material.

Casual Observation

This type of observation is used to detect learner behaviour during the use of the system and is not geared to particular areas or tasks but usually to general usage. To be able to learn from the system the learner must engage with it and want to learn, if this is not the case then no matter how good the system, the learner will not learn. Casual observation can identify if users are intentionally trying to use the system to learn or "playing" with the system with no serious attempt to learn. Their behaviour may be markedly different but if the casual observation was not conducted the developer might attribute poor results to the system and not to the students motivation. Casual observation can also detect areas where students may be having difficulty by noticing those users who ask questions of one another, for example, when problems arise but who do not mention them during interviews or in questionnaire answers. These observations may affect several areas such as questionnaire refinement or interface refinement or alterations to lesson structure or

provision of additional help messages etc. which can have a profound effect on the overall effectiveness of the system.

Informal Observation

Informal observation is used where the evaluator is looking to identify specific areas during the evaluation exercise. The evaluator could be looking at navigation, interaction between the user and the system, number of times the user needed help etc. In this case the evaluator is observing the same things for the group as a whole to get an overall impression of the results. They may use an informal reporting mechanism such as textual observation or they may have a checklist which is ticked as the responses are achieved. The informal observation may focus on particular areas such as navigation or help or may focus on particular tasks.

Formal Observation

Formal observation takes place where an individual or group is observed as they perform some pre-determined tasks, the results are noted and then the evaluation repeated with more users. The areas observed might be very similar to those mentioned in informal observation such as using the help facility to determine how to perform a particular task or navigating to particular sections of the system to test the effectiveness of the navigation strategy employed.

Interviews

Interviews may be conducted with individuals or groups of students or just used to record casual conversations. Opinions expressed by any one group of students are subjective and should be checked against any other information that has been collected about the use of the CBL system. However, interviews allow students to express their feelings about a system and to pinpoint any issues which they feel are particularly important. It is important the interviewer does not influence the individual or group and also that one individual in a group does not dominate. Interviews provide qualitative data about the CBL system which needs to be carefully checked for validity.

Video

Video enables the evaluator to secure a record of the students using the CBL system, it enables the evaluator to watch the session more than once and also to make notes of that session at their leisure. This is useful since the evaluator may easily miss something during the evaluation which can then be detected later. Care must be taken that the students are comfortable with the use of the video and that it does not affect the results, students who are ill at ease with the use of video may behave differently than if the video was not

used. Video can capture all the details of the student interactions, with one another, with the tutor and with the system.

8.3.3 Teaching and Learning Effectiveness

Effectiveness, with respect to a CBL system, is determined by the ability of the learner to meet the learning outcomes of the topic which themselves are derived from the objectives of the topic provided by the tutor. The achievement of the learning outcomes is often measured via some assessment vehicle such as an examination or in the case of CBL lessons via on-line tests, quizzes or problem solving exercises.

Within an evaluation activity, a mechanism for providing a measure of learning gain is the application of pre and post-tests, the pre-test is given before the student uses the CBL material and the post-test after. Pre and post-tests are based on the learning outcomes identified by the subject expert. Pre and post-tests provide a quantifiable comparison between what the learner knew prior to using the CBL material to what they know after using the system. Pre and post-tests provide additional information for the developer as they can identify weak areas of the CBL material, for example, if every student has difficulty answering a particular question in the post-test it might well be attributed to the CBL system which may need to be amended or corrected. Other evaluation techniques such as interviews used in collaboration with this type of testing can provide confirmation of a problem. This type of testing is criticised from the point of view of its effect on the outcome of the evaluation, administering the tests may itself affect the results since learners have an idea of the types of questions they will be asked before using the CBL material and may focus on what they perceive as the relevant areas. Despite this, pre and post-tests are used extensively to measure the difference in knowledge of students prior to use of a CBL package to after its use. They are seen as an indicator of the effectiveness of the learning experience and may be used by the developer to demonstrate to tutors what has been achieved through the use of specific CBL material.

To conclude, the methods outlined above provide a combination of approaches that can provide both quantitative and qualitative data with which an evaluator and hence a developer and tutor may judge, the content, usability and effectiveness of a CBL system.

8.4 Formative Evaluation

8.4.1 Expert Evaluation

Evaluation of the E-R Modelling CBL material produced with the aid of the UDRIP model, initially, focussed on the content of the lessons. Several subject experts were asked to use the CBL material and to provide comments on the content with respect to its clarity, consistency, completeness and correctness. The subject experts were people who were already teaching E-R Modelling to students but usually in a more traditional fashion i.e. using lectures, seminars, tutorials etc., thus, they were also able to judge how well the material would fit into traditional teaching strategies. The evaluations were conducted on an informal basis and uncovered various issues ranging from spelling and grammar to pinpointing material that needed further clarification. Overall, the subject experts were enthusiastic about the CBL material and expressed an interest in adding it to their own teaching resources. Feedback from this exercise enabled the CBL material to be amended and re-evaluated by the same subject experts who then approved the content paving the way for evaluations with students.

8.4.2 Usability Evaluation

Initial evaluations with students focused on the usability of the CBL material, these were conducted as formative evaluations to ensure the material could be used as expected and with no difficulty. This evaluation was carried out to refine the design of the CBL system, with a view to progressing to a full scale evaluation of both the usability and learning effectiveness of the material with students. It was decided that this early evaluation would be conducted using questionnaires, interviews and observation. Work began on devising a questionnaire which would provide useful and meaningful results. Observation and interviews were planned to be less formal to allow the students to express any opinions they had on the experience of using the CBL and any problems they found.

Questionnaire Design

The design of the questionnaire derived from work carried out in the EC DELTA project ILDIC where fifteen criteria for evaluating Multimedia systems were deduced (Barker & King, 1993). The findings from the Barker-King experiment were intended to be used as a guide for evaluators and to be tailored to individual experiments and requirements. Each criterion had suggested questions for that area which could be incorporated to form

customised questionnaires. The questionnaire, in this case, consisted of 26 questions adapted from those suggested in the report, 23 of which were graded 1-5 i.e. closed questions, two of which were open questions that asked the user to list the most attractive and unattractive features of the system and one question that was a combination question, part closed and part open which asked if the learners had any suggestions for improvements to the system. The areas assessed included navigation; screen layout; preferred learning style; assessment of prior knowledge and acquired knowledge; help facility; level of learning and overall impression of the system. A typical closed question had a 5 point Lickert scale e.g.

5. The screens within the courseware were attractive.

strongly agree	agree	undecided	disagree	strongly disagree
1	2	3	4	5

Figure 8-1. Lickert Scale.

Open ended questions were also used to allow the respondent to express an opinion or give more detail e.g. Is there one thing that stands out as being really good about this product? Once the questionnaire was complete, the evaluation could begin.

The material was presented to a group of students studying on the first year of a conversion MSc. course in Computer Studies. The student group consisted of 14 students, 12 males and 2 females. Each was asked to complete a questionnaire and to record any comments they had about any aspect of the courseware or its use, the students had little or no experience of Data Modelling.

Questions were asked on:

Navigation	paths through the courseware and button use;
Screen Layout	screens and graphics;
Learning	use of the courseware and the students preference in comparison with other methods e.g. books and lectures;
Help	help and feedback on mistakes;
Level	how the courseware compared to learning from books and lectures and whether the courseware was too easy;
Overall Impressions	this checked there were no elements of racism or sexism that might hinder the student use of the courseware through a

perceived prejudice that was not intended and if there were
more multimedia elements that the users felt would enhance
the presentation e.g. sound and video.

The students were also asked to list any ideas they had which had not been covered in the
questionnaire, and also to say what were the most attractive and unattractive features of the
courseware.

Observation and Interviews

Observation was conducted as the students used the CBL system and interviews were
carried out at the end of the experiment. Observation was useful to ensure the students
were using the system correctly and without difficulty. This aided the interpretation of the
results from the evaluation exercise. Interviews allowed the students to express their
opinions on the system without the constraint imposed by the closed questions found in the
questionnaire. Informal interviews were conducted with groups and individuals to try and
elicit their views of the CBL material.

8.4.3 Evaluation Results

The results outlined in the following sections reflect the order of the sections on the
questionnaire..

Navigation

Navigation through the CBL lesson was designed to meet the needs of both serialist and
holistic type learners (Pask, 1976, Clarke, 1989), to this end learners had the opportunity
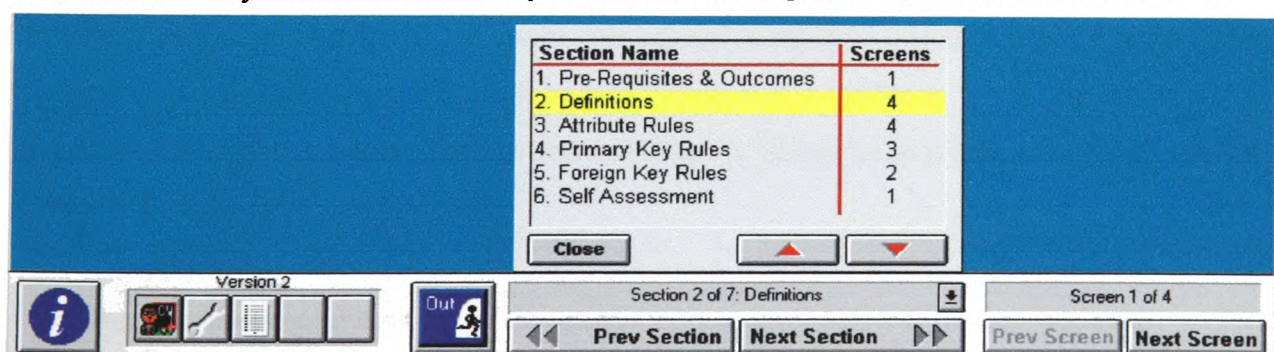to move screen by screen or section by section or to jump directly to a particular section.

Figure 8-2. Navigation.

The questionnaire asked four questions about navigation to check how learners had used
and felt about their level of control over the paths chosen to move through the CBL lesson
and how they felt about the buttons available to them.

Results from the questionnaire indicated that 13 out of the 14 (93%) of the learners felt that the functions available via the buttons were intuitive. The one learner who felt the buttons were not intuitive also felt that it was easy to become lost in the system and that (s)he was not always certain how to navigate through the material. A contradictory finding, however, was that they felt they could choose routes that suited them through the lesson. Despite the fact that the majority of the learners found no problem with the buttons a worrying 8 of the 14 (57%) felt that they were not always certain how to move through the lesson, though 6 of these were undecided and only 2 definitely had a problem.

In contrast, 11 of the 14 (79%) felt they could choose routes through the lesson that they felt were relevant to their learning needs, overall, 2 of the 14 (14%) felt it was easy to become lost within the lesson, with 3 undecided (21%) and the majority 9 out of 14 (65%) finding no problems with navigation. These results showed a conflicting attitude to the navigation provided. However, the learners found no problem with the buttons, therefore, it was decided to include more on screen messages to supplement the navigation controls e.g. Click [Next Screen] to continue where [Next Screen] was the name of a screen navigation button. From the observation results at this time, it was discovered that very few people used the direct navigation facility to jump to a particular section. Subsequent evaluations have shown that this facility is rarely used until the learner becomes confident using the CBL material and once they begin to use it more frequently.

Screen Layout

All learners felt the screens within the courseware were attractive and all felt the size of any graphics used were appropriate. The majority of the learners 12 our of 14 (86%) felt that everything was clearly laid out with one person undecided and the person who felt the buttons were not intuitive also felt the screen layout was a problem. In fact in the open questions, the most attractive feature was credited as being the graphics and screen design by the majority of the respondents.

Learning

Learning was the largest section of the questionnaire and focussed on the students' attitudes to the use of computer based material as part of their learning. The questions attempted to uncover where they felt this type of resource should fit in with existing methods such as tutorials, lectures etc. It also attempted to reveal how much they felt they had learned using the CBL material and if they would use it again.

There was a clear consensus that this type of resource was a useful supplement to existing teaching/learning resources and strategies such as text books and lectures and also that it

would provide a useful revision tool; all the learners supported these views. If there was a choice between using this type of material or conventional teaching/learning strategies only one person advocated the conventional methods with 6/14 (43%) undecided, 7/14 50% choosing CBL. It is clear that learners strongly support this type of resource as an additional approach rather than as an alternative approach.

Only one person had reservations about using the CBL material again, they were undecided, the majority 13/14 (93%) affirmed that they would like to use it again and all the learners felt the CBL material was a useful learning tool which is a very positive outcome. Only one respondent was undecided about whether they felt the CBL was boring but they did respond that they would use it again, also encouraging, the majority 13/14 (93%) felt that they had not been bored whilst using the CBL material, figure 8.3.

12. Working through this courseware bored me

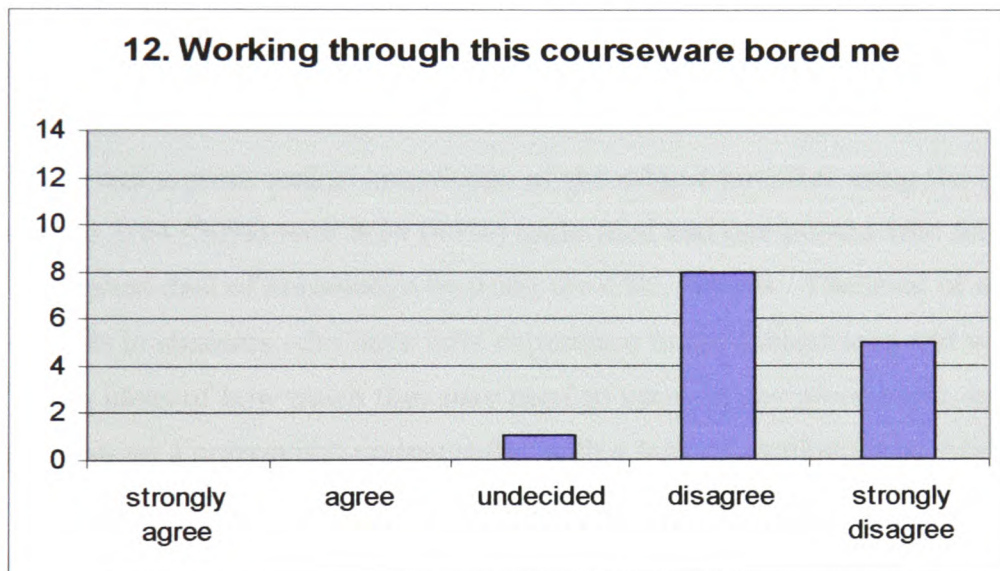Figure 8-3. Results from Question 12.

The learners had several chances to interact with the lesson, these were provided using drag & drop, clickable objects and text entry question and answers either via exercises or examples or problem solving opportunities. The most popular of these were clickable object (13/14 – 93%) and drag & drop (11/14 – 79%), text entry (6/14 – 43%) was the least popular, Figure 8.4.

14. The method I most preferred for making choices was
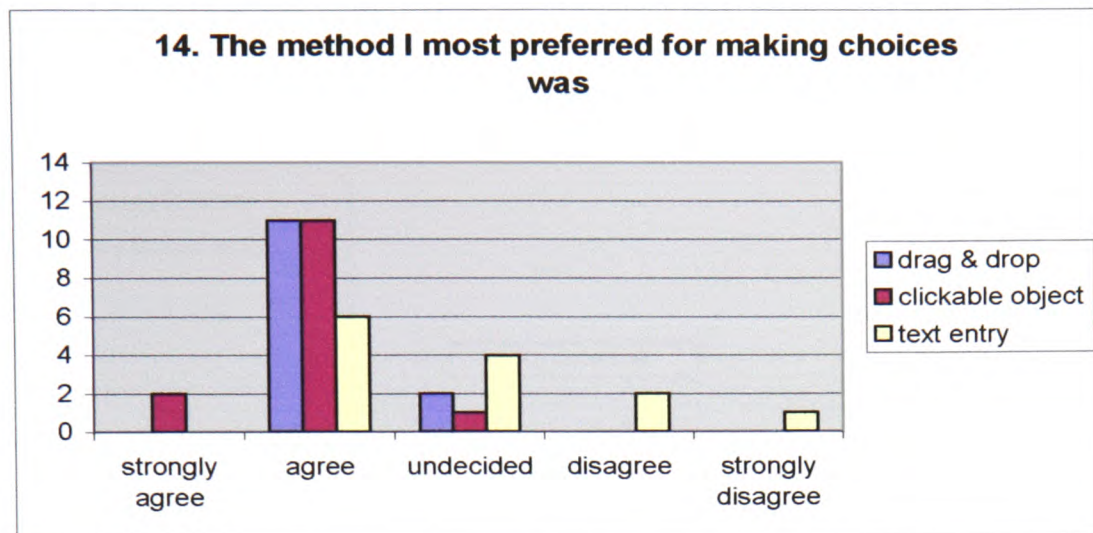
Figure 8-4. Results from Question 14.

Finally, the questionnaire probed whether the students felt their knowledge of E-R Modelling had increased after using the CBL lessons. Prior to using the material only one person felt they had a great deal of knowledge of the subject but after using the CBL, this had increased to 7/14 (50%) with 6/14 (43%) undecided and one person who felt they had not acquired a great deal of knowledge by using the CBL lessons. The level of indecision is understandable in students who have little experience in the subject area and who may have conflicting ideas of how much they may need to know in any given topic, many students studying on a conversion course suffer with a lack of confidence in their ability. It was very encouraging to have such an increase in those who felt that exposure to this type of learning resource had increased their knowledge.

Help

The help facility within the CBL material was designed to assist with navigation and screen layout, it explained what the buttons on the screen represented and what the headings covered.

101
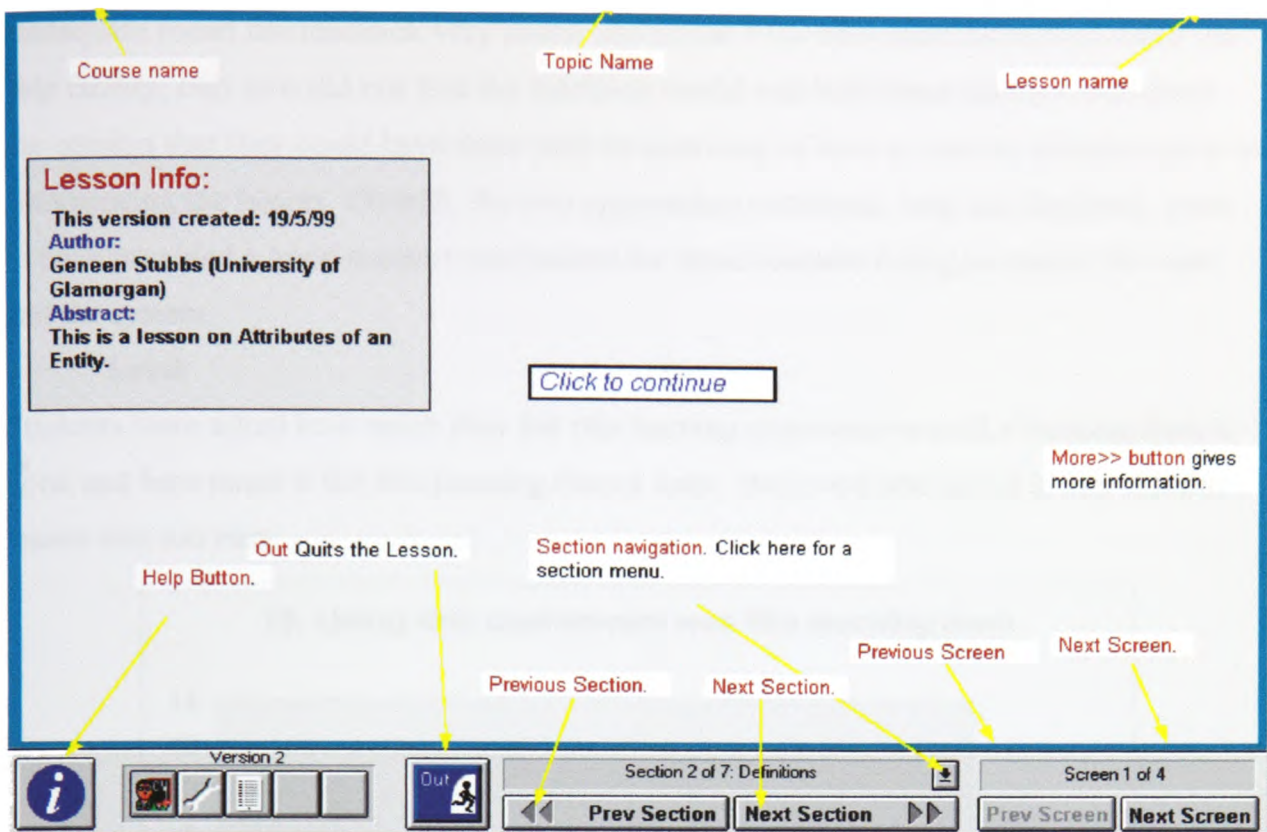
Figure 8-5. Help.

Feedback was also provided for learners in the problem solving section in the form of reasons why the answer might be incorrect and opportunities to try again.
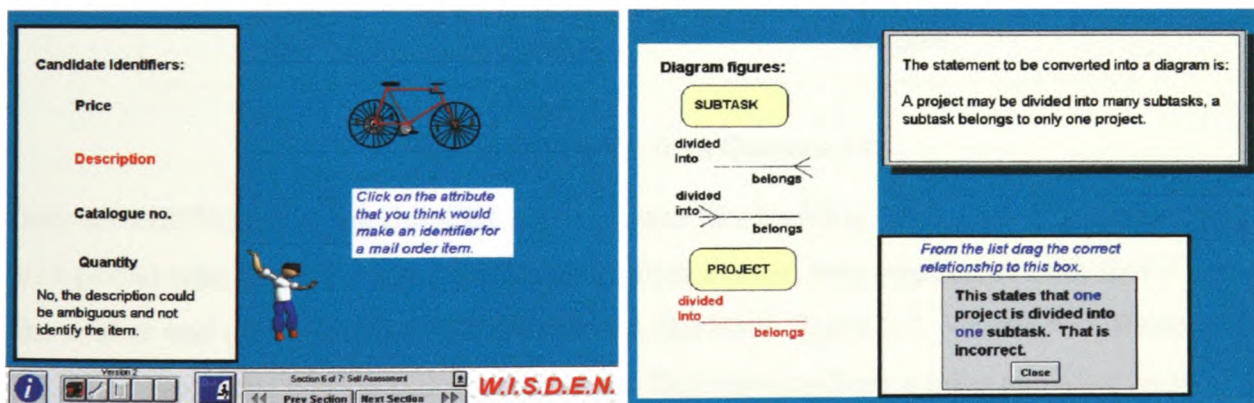
Figure 8-6. Problem-Solving Feedback.

The help facility was limited to screen elements and buttons and as such 8/14 (57%) were undecided as to its adequacy, with a further person finding it inadequate, this left 5/14 (36%) who were satisfied. This may be explained by the lack of context specific help i.e. help with the subject matter itself since the majority of the learners expressed no problems with the interface, to confirm this 11/13 (85%) felt the feedback provided useful explanations for mistakes and ways to move on. The person who found the help

inadequate found the feedback very useful and of the 57% who were undecided about the help facility, only two did not find the feedback useful and both these learners expressed the opinion that they could have done with an overview of how to use the software prior to commencing the lesson. Overall, the two approaches combined, help and feedback, seem to have provided a good support mechanism for those learners trying to master the topic and the system.

Level

Students were asked how much they felt this learning experience was like learning from a book and how much it felt like learning from a tutor, they were also asked if they felt the lesson was too easy.

19. Using this courseware was like learning from
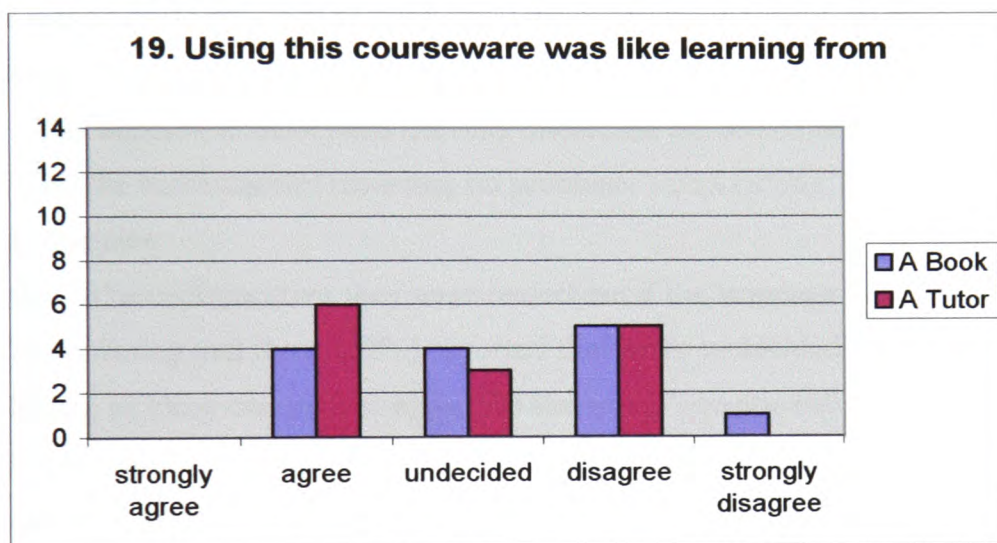
Figure 8-7. Results from Question 19.

Only 4/14 (29%) felt using this CBL material was like learning from a book as opposed to 6/14 (43%) who felt that it was like learning from a tutor, two respondents felt that it was like neither and one respondent felt that it was like both, figure 8.7. Of the rest, those who felt it was like using a book did not feel it was like learning from a tutor and vice versa. This was a very mixed reaction to the CBL material and can possibly be explained by the unfamiliarity of the users with this type of learning experience since the results do not show a clear-cut outcome. It is also fair to say that the CBL lessons were not intended to be like a book and, though also not intended to be like a tutor, it would be preferable to have it regarded more like a tutor than a book, so from that point of view this result is very positive.

The response to the question on how easy the learners found the CBL material was also very mixed, 7/14 (50%) did not find it too easy, one person undecided and 6/14 (43%)

reporting it as too easy. This type of mixed result is often found in module reviews where some students find the work easy and some found it ok or hard, the result would have been more problematic if all the learners had found the lesson too easy or all too hard.

Overall Impressions

Learners were asked four questions in this section, one related to the use of the mouse and keyboard, one related to the language and examples, one related to racism and sexism and one related to additional multimedia elements, in this case, audio and video.

Only 1/13 (8%) found that the use of the mouse and keyboard were not intuitive and in the open questions this user complained at having to move the mouse to so many places on the screen the comment was "you kept on having to click the mouse on different areas". One other user was undecided about how intuitive the use of the mouse and keyboard was and their comment was "it was not clearly stated that text entry should be followed by <enter>". This appears to have been the only dissent on the use of the mouse and keyboard with the vast majority reporting no problems 11/13 (85%), one person did not answer this question.

Two people (14%) reported that they were undecided if the language used in the CBL lessons was confusing and three (22%) reported they were undecided if the examples used were confusing, of these one person appended the word "sometimes" to their answer. Again, the majority 12/14 (86%) and 11/14 (79%) reported no problems with language nor examples, another very positive result.

No-one felt the CBL material had any elements of racism, one respondent was undecided if it contained any elements of sexism, though they did not elaborate on this in any of the open questions, overall, once again, a strong positive response from the respondents.

As to the issue of the inclusion of additional audio and video, once again a mixed result,

24. The courseware would have been more enjoyable and interesting if it had included
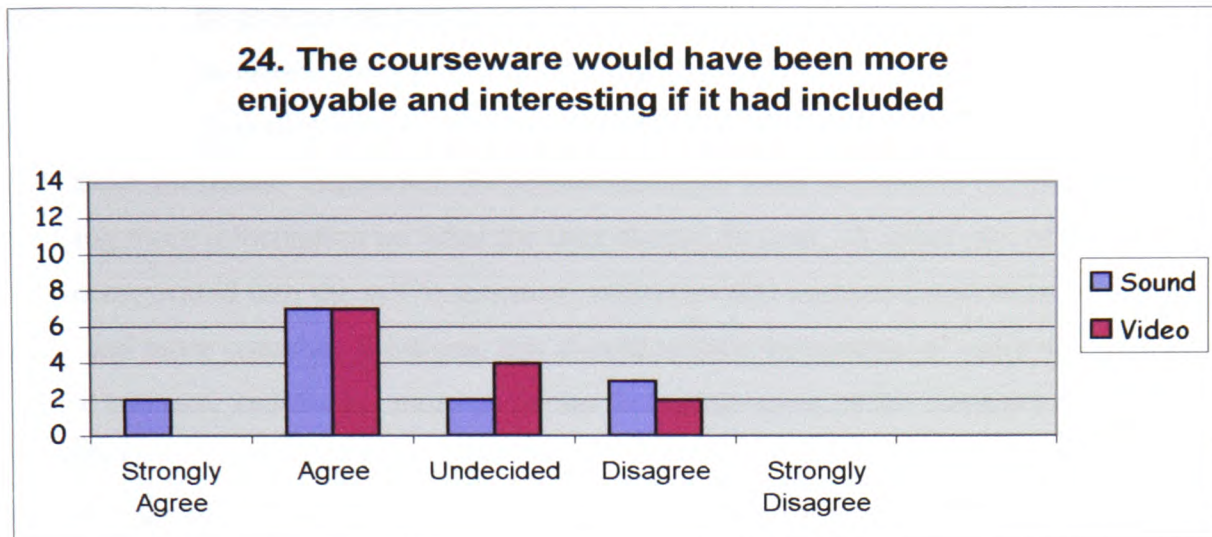
Figure 8-8. Results from Question 24.

9/14 (64%) favouring the inclusion of sound and 7/13 (54%) favouring the inclusion of video, however, 5/14 (36%) were undecided or against the inclusion of sound and 6/13 (46%) undecided or against the inclusion of video. More respondents were undecided about the inclusion of video than were undecided about the inclusion of sound. Whilst the inclusion of additional multimedia elements might enhance the look and feel of the CBL material there needs to be considerable research into its effects on learning to warrant its inclusion since the primary aim of this CBL material is to facilitate learning for the users. Unfortunately, the inclusion of multimedia elements such as these were ruled out within the W.I.S.D.E.N. project due to the need to distribute the software to other higher education institutions where the specification of the computers used might be too low to accommodate such elements.

8.5 Conclusions

Overall, reactions from the students were positive and constructive. During interviews they expressed a general approval of the use of courseware as an additional resource for teaching, stating that its "patience" was a plus factor, allowing them to revisit areas of uncertainty and attempting the questions in the self assessment (problem solving) sections many times until their confidence improved. Several students expressed their appreciation of the examples provided which illustrated various concepts. Some students said they felt there should be more. The screen layout and design was praised as being "easy on the eyes" or "attractive" and the graphics as being useful in explaining difficult concepts. Feedback was also commended, as was the opportunity to interact with the lesson and the

ability to navigate around the topic. One comment stated "graphics and animation very good, it is a great effort."

Areas pinpointed as needing attention were, on-screen messages, further examples and self assessment questions. Additional on-screen messages were included in the material providing more information on what the user should do next. A wider mix of questions were incorporated into the self assessment section so that learners could attempt both simple and more complex questions, this should reduce the number of users who find the material too easy and finally, more examples to explain some of the concepts were added to satisfy the request brought to light in the interviews. The scene was, now, set to progress onto the learning effectiveness evaluation to test if, indeed, users could learn from the CBL lessons. Further usability tests were also planned to ensure no further problems were uncovered that could affect the results of these effectiveness evaluations of the system.

Chapter 9

Refinement of the Development Method

9.1 Introduction

During the course of the design and application of UDRIP, research continued into the pedagogic theories. It soon became apparent that there was an omission in the CBL model. Research into learning styles was, initially, undertaken to ensure that as many types of learners as possible were considered during the construction of the CBL material from the point of navigation and interface design. An early finding indicated that learners might fall into two main categories, from the point of view of navigation, sequential learners who would need a sequential, linear type navigation strategy and holistic learners who would need a direct or browsing type navigation facility (Pask, 1976, Clarke, 1989). These findings influenced the types of navigation facilities employed within the CBL lessons and resulted in the construction of a sequential navigation facility moving from screen to screen and section to section, together with a direct navigation facility which allowed learners to jump directly to any individual section of the lesson. Further research uncovered more types of learning styles, some of which were addressed by elements of UDRIP but some of which were not. The initial idea of just two types of learner, serialist or holistic, was extended to cover more learners such as pragmatists, theorists etc. (see section 9.3). UDRIP was, therefore, also extended to ensure these additional learning styles could be catered for (see section 9.4).

9.2 Development

As the number of CBL lessons increased, feedback from the consortium members was positive. Initial evaluations began throughout the group and findings suggested the material was well received and appreciated by both tutors and students. Further evidence was gathered during the dissemination of these findings through workshops, seminars and

conferences. There was great interest in the idea of a CBL development model (see published papers in appendix).

9.3 Pedagogic Theories

As research continued, development progressed and formative evaluation began, deeper investigation into the area of learning styles uncovered a plethora of theories dealing with the way people prefer to learn. Work by David Kolb (Kolb, 1984) indicated that learners moved through a cycle of stages, each of which was needed to promote learning.
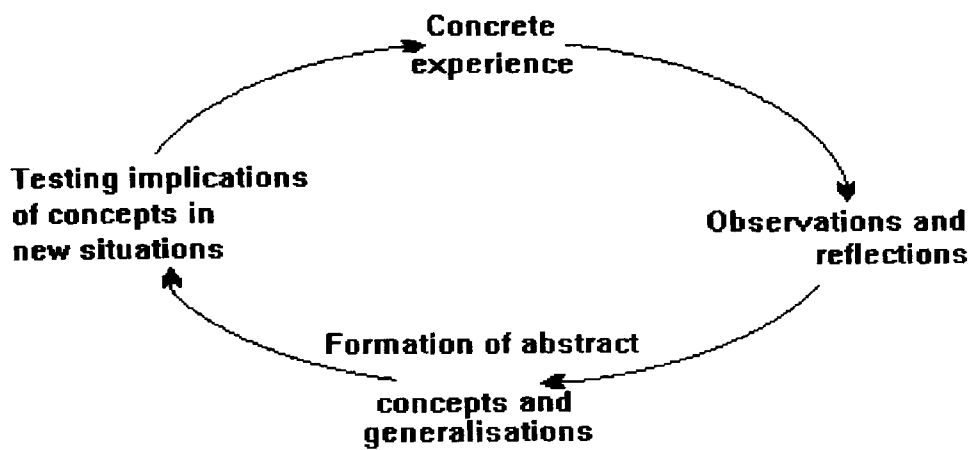
Figure 9-1. Kolb's Learning Cycle.

A learner moves through each of the stages as they attempt to learn. The stages, as outlined above, are: having an experience which affects the learner, reflecting on that experience, forming a hypothesis to explain what has happened and then testing out that hypothesis to deepen understanding by having a further experience and so the circle is repeated. Obviously, this theory illustrates experiential learning where students learn by doing. As far as the CBL material produced to teach E-R diagramming is concerned, the concept of learning by doing fits very well with the interactions and assessment questions used throughout the CBL lessons. Within the CBL lessons, students are encouraged to apply the knowledge taught through problem solving activities and exercises. However, less well addressed is the reflective aspect of the learning cycle. To reflect is "to mediate on; think about" or "to consider; remind oneself" (Oxford Encyclopedic English Dictionary, 1991). In this sense, the reflective activities are met, to some extent, through the navigation facility provided within the CBL lesson which allows the learner to re-visit

the sections within any of the lessons. However, this relies on the student being aware that they need to reflect on their learning, which is not always the case.

Additional research uncovered further references to the reflective aspect of learning (Schön 1987, Honey & Mumford, 1992, & Race, 1994). One in particular, Honey & Mumford, categorised learning styles with respect to the stages of the learning cycle. They identified four different preferences, or ways in which people prefer to learn, each related to a different stage of the learning cycle. These preferred "learning styles" they call Activist, Reflector, Theorist and Pragmatist.

ACTIVIST
Having an experience

PRAGMATIST
Planning the next steps

REFLECTOR
Reviewing the experience

Concluding from the experience

THEORIST
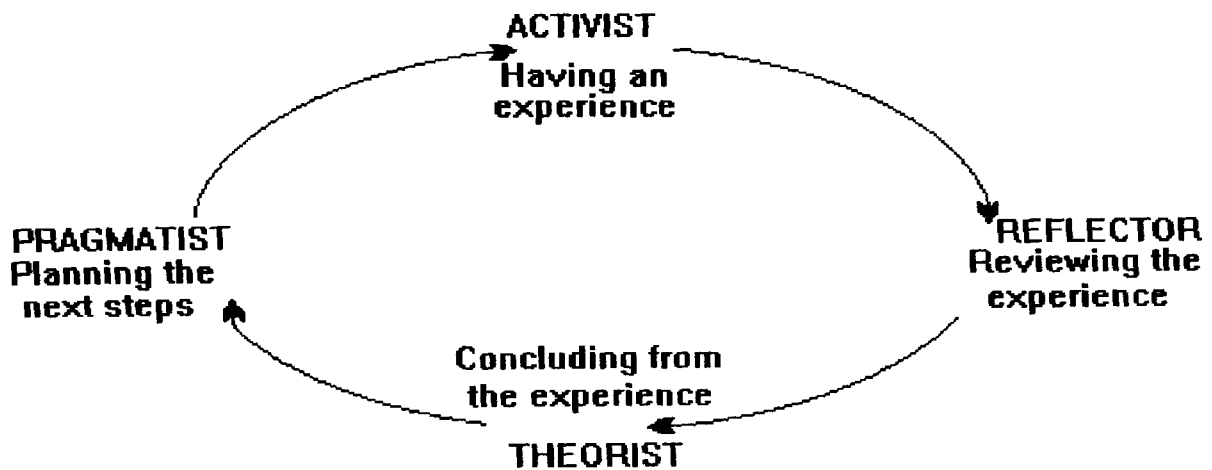
Figure 9-2. Honey & Mumford's Learning Cycle.

Many learners show a marked preference for one stage or another some preferring just one some two etc. Although having a preference for one mode over another does not appear to inhibit learning, its impact on CBL development is evident. There are many different types of learners who will use the CBL material to learn a topic. However, whilst the courseware is not designed to accommodate any one particular learning style, the aim is to provide an equal opportunity for as many different learning styles as possible to facilitate the learning process for as diverse a user population as is practicable. The different types of learners prefer different ways of being taught e.g. activists like new experiences and short applicable exercises, reflectors like to mull over activities and relish the opportunity to review what has happened, theorists learn best when they are in well structured situations with a clear purpose and pragmatists like the chance to apply the knowledge in practical situations i.e. they can try out and practise techniques with feedback to improve performance.

The different types of learning style are not mutually exclusive, many students can apply different strategies at different times and for different types of tasks. However, it is clear that any group of students need to have the opportunity to experience any of the stages and to that effect, any CBL material must accommodate them. Within the E-R diagramming CBL many of the activities enjoyed by the differing learners already exist and could explain why the CBL material was so enthusiastically welcomed. However, one area appears to be deficient and this is that needed by the reflector to review what has happened to them.

The reflective learner needs to extract the key points from the learning experience that they have just undertaken. They must also extract the principles involved and attempt to form hypotheses to explain what they have just learned. Also, as defined above, to reflect is "to remind oneself", and as such the learner needs to remind themselves of the learning that has been attempted or achieved. Through reflection, the learner should strengthen the knowledge which they have and deduce the knowledge that they are lacking. To this end, it was felt that the learners who were using the CBL material needed a summary section which outlined the key points from the lesson and which helped them to reflect on the activities and experiences they had just encountered. It was also anticipated that this type of section would allow the learner to pause and consciously start the reflective practice which is perceived as a useful activity to consolidate learning. As Donald Schön advocates learners should constantly reflect on "what they have done, why they have done it and how it might be done differently" (Schön, 1987).

9.4 UDRIPS

To accommodate the reflective activity within any CBL material developed, a further section was added to the UDRIP model to ensure the reflective learner had a chance to review what they had just learned within a CBL lesson. This reflective section took the form of a summary which highlighted the objectives that had just been addressed within the package. It became the last section of the model and was actually the last section of the CBL material to encourage an element of reflection before the learner moved on to new material. Thus UDRIP evolved into UDRIPS and the evaluation exercises continued with the material being adapted to include this additional section in all of the lessons on E-R Modelling.

The UDRIPS acronym thus becomes:

9.4.1 Universal picture:

Student - where should I be coming from, where am I going, what will I be able to do?

Developer – what are the pre-requisites & expected learning outcomes?

9.4.2 Definitions:

Student - what don't I know?

Developer – what are the keywords & concepts that need to be defined? This is the declarative knowledge "the facts that we know" (Anderson, 1980).

9.4.3 Rules:

Student - how are things applied or used?

Developer – how are the keywords and concepts applied and used? This is the procedural knowledge "the skills we know how to perform" (ibid).

9.4.4 Illustrative examples:

Student - are practical examples available?

Developer - provide embedded scenarios and solutions.

9.4.5 Problem solving:

Student - do I really understand?

Developer – is there an opportunity for self assessment?

9.4.6 Summary:

Student - what exactly did I learn?

Developer – provide a summary of the significant elements of the lesson, this could be a re-stating of the objectives that were introduced at the start of the lesson.

Alteration of the CBL material to accommodate this, involved the addition of one section to the structure for each individual lesson. Once this was in place, further evaluations were planned both to continue testing the usability of the CBL material and to test its learning

effectiveness. It was anticipated that students reservations about some of the aspects of the CBL material outlined previously should have been addressed and, thus, further evaluations should show a difference in attitude to the first evaluation exercise.
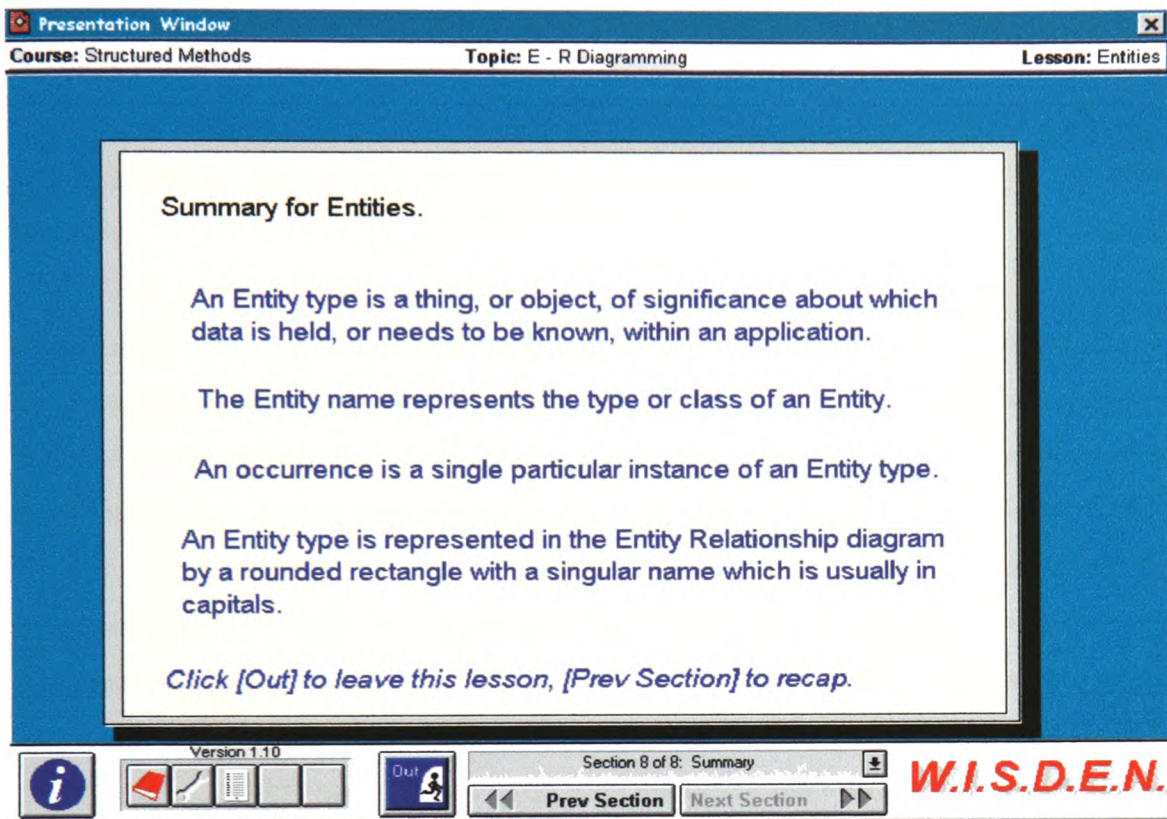
Figure 9-3. Summary for Lesson on Entities.

9.5 Conclusion

An investigation into the field of pedagogic theories uncovered an aspect of learning previously not addressed fully within the UDRIP model and, hence, omitted from the CBL material. To accommodate as many types of learners as possible, it is necessary to provide each with a chance to find themselves in their ideal learning situation. Learners such as activists, theorist and pragmatists were well served by the CBL material which was developed using the UDRIP model but reflectors had little encouragement to review what had happened during the use of that material. The reflective aspect of the CBL material was addressed by including a summary of the key points presented in the CBL lesson to allow the user to reflect on what had just been introduced to them. The summary presented the learning outcomes that had been proposed at the beginning of the lesson to consolidate the learning process. The amendment to UDRIP to form UDRIPS was also communicated

to the W.I.S.D.E.N. consortium members and the CBL material under construction amended appropriately.

The inclusion of a further section to address the reflective element of the learning process enabled the CBL material produced using the UDRIPS model to accommodate the four learning styles identified by Honey and Mumford. Its approach, now, consolidates the elements of the learning process perceived as beneficial and supports practices such as problem solving, knowledge application and hypothesis formation through the reflective practice.

The overall aim is to offer as complete an experience as possible to a diverse learner group. Whilst many learners prefer some learning modes more than others, all learners should be encouraged to participate in all the learning modes. This helps to mature their learning style. A more mature learner stands a better chance of learning from a wide range of experiences and, hence, this allows them to take full advantage of any learning opportunity offered. The material produced using the UDRIPS model should, therefore, not only provide learning practice in the modes the learner prefers but also encourage all learners to participate in those modes where they may feel less comfortable. This supports the learners as they learn and helps to foster a more mature approach to learning.

To conclude, UDRIP was refined to include an additional section which covered the reflective aspect of learning and, thus, became UDRIPS. This section took the form of a summary of the key points of the lesson and was intended to encourage the learner to reflect on what they had just learned in the lesson they had undertaken.

Thus, the CBL design and development model was complete. It was intended to be used in conjunction with existing CBL methods (see chapter 4). To illustrate how UDRIPS integrates with existing instructional design models, consider the Dick and Carey model (arguably, the most widely known and used model). UDRIPS may be used during the phase "Develop & Select Instructional Materials", shown shaded in figure 9.4. Once the structure of the lessons has been achieved using UDRIPS the developer can progress to the subsequent phases.
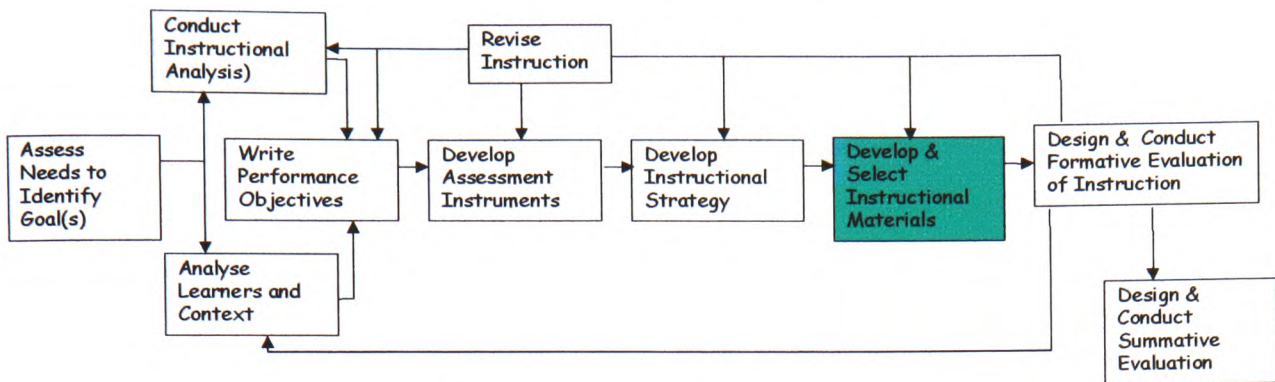
Figure 9-4. UDRIPS and Dick & Carey (Dick and Carey, 1996).

In fact, wherever an instructional design model contains a phase devoted to developing instructional material, UDRIPS can assist the developer by providing a clear structure for the lessons under development. Several models have phases such as:

Develop and select instructional materials, (Dick & Carey, 1996);

Develop and implement (Hannafin and Peck, 1988);

Select and/or develop materials (Knirk and Gustafson, 1986);

Construct prototype (Tripp & Bichelmeyer, 1990);

Determination of strategy (Gerlach & Ely 1980);

Content sequencing (Kemp, 1998).

UDRIPS may be used in all these situations to assist in the development of instructional material. At this stage, the material produced in this project has been in the form of a tutorial type system to teach E-R Modelling. Future work will consider the application of UDRIPS to additional domains both inside and outside the field of Computing.

Chapter 10

Further Evaluations

10.1 Second Evaluation

This chapter will present the findings from two further evaluations of the CBL material produced using the UDRIPS development model. The usability evaluation was conducted to ensure the learners could use the material without any problems. The learning effectiveness evaluation was used to verify that learners could indeed learn using the CBL material. The usability evaluation was conducted as before (see chapter 8) and the learning effectiveness assessed using pre and post-tests.

10.2 Introduction

Subsequent to the initial, formative, usability evaluation, certain alterations were made to the courseware to enhance the interface and alleviate one or two problems encountered, (see chapter 8). The courseware was then ready for further evaluations and both the usability and learning effectiveness were addressed during this testing phase.

The same questionnaire was used for the usability evaluation as in the first evaluation, with a multiple choice question test used for the learning effectiveness evaluation. Observation, informal interviews and video were used during the second evaluation but only observation and interviews during the third. Video was found to be very intrusive. Whereas the students were prepared to chat with one another and with the evaluator before the video camera was introduced, as soon as filming began, they fell silent and remained that way until the camera was taken away. The video tape produced at this time was, therefore, less than helpful.

In both evaluations, observation was conducted as the students used the CBL system and interviews were carried out at the end of the session. Observation was useful to ensure the students were using the system correctly and without difficulty. This aided the interpretation of the results from the pre and post-tests where it was important to know if

the students had taken the evaluation exercise seriously. Interviews allowed the students to express their opinions on the system and to expand on the answers given in the questionnaire to either the open or, more importantly, the closed questions asked. Informal interviews were conducted with groups and individuals to try and elicit their views of the CBL material.

10.3 Usability Evaluation

10.3.1 Questionnaire Results

The same questionnaire that was used in the first evaluation was again used in these later evaluations. It consisted of 26 questions, 23 of which were graded 1-5 i.e. closed questions, two of which were open questions that asked the user to list the most attractive and unattractive features of the system and one question that was a combination question, part closed and part open which asked if the learners had any suggestions for improvements to the system.

The study group consisted of 13 students, all male, enrolled on a Masters course in Computing. Again, areas addressed were: Navigation; Screen Layout; Learning; Help; Level and Overall impressions. The evaluation exercise comprised both a usability element and a learning effectiveness element. It was conducted in a single session with the students attempting the pre-test, using the courseware and then completing the post-test and filling in the questionnaire immediately afterwards.

Navigation

The navigation questions focussed on the buttons displayed on the screen together with how the learners moved through the CBL lesson. The majority of the users 10/13 (77%) felt the functionality of the buttons used was obvious with 3/13 (23%) undecided, of these 2 were novice computer users. Results from the questions on navigation gave a mixed picture, the majority 10/13 (77%), once again, felt that they would not get lost in the lesson but 1/13 (8%) were undecided and 2/13 (15%) felt they could get lost, both of these respondents reported getting confused over next section and next screen. However, perhaps surprisingly, both these respondents reported that they could choose routes through the lesson that were relevant to them which is a very positive result. With respect to choosing paths through the lesson, 10/13 (77%) felt they could choose relevant paths, 1/13 (8%) felt they could not and 2/13 (15%) were undecided. The most mixed result came from the final question in this section which asked if the users had been certain how

they could move through the lesson. Of the 13 users, 7/13 (54%) reported they were certain how to move through the lesson, 2/13 (15%) were undecided and 4/13 (31%) were not certain how to move through the lesson. Of these respondents, one reported problems with reading the on-screen messages as he was colour blind and the font was coloured blue, a W.I.S.D.E.N. adopted standard for on-screen messages, but the background was green, this was easily adjusted to make the text opaque so that the blue lettering had a white background, figure 10.1.

Figure 10-1. Text before and after evaluation.

This problem could have affected more of the users without their knowledge as not everyone is, necessarily, aware of any problems with colour blindness. Several users reported that they did not "like" some of the colour combinations though none of these reported that they were colour blind. This new foreground/background combination should elicit better results in further evaluations.

Screen layout

This section was aimed at detecting problems with screen layout and graphics. These questions provided a very positive endorsement of the interface design and layout. The users reported that 12/13 (92%) felt the screens were attractive, only one person (8%) did not agree with an added comment included that the colour combinations were sometimes difficult, again, this was rectified as mentioned previously. The size of graphics used were approved by 11/13 (85%) with two (15%) reporting they were undecided. The majority of the users 9/13 (70%) supported the screen layout, two (15%) were undecided but two (15%) felt the screen layout was not clear. Once again, several of these users reported difficulty with certain text and background colours.

Learning

This section examined how the users felt about the CBL material with respect to how they would like to see it incorporated into their learning, how useful they felt it was and also what interactions they preferred. It also looked at how much they, themselves, assessed they had learned from the CBL lessons.

The overwhelming majority of the students expressed the opinion that they would not prefer to use text books and lecture notes instead of the CBL material, 12/13 (92%). Only

one person (8%) was undecided on this issue. All 13 students felt that this CBL material would provide a useful supplement to text books and lectures and, also, that the CBL lessons would provide a useful revision tool. A very impressive endorsement for this type of learning resource as this reinforced the same finding from the first evaluation. All students agreed they would like to use the CBL material again and 12/13 (92%) felt the lessons were not boring, one person (8%) was undecided. This latter finding concurred exactly with the first evaluation but the strength of feeling was greater in the second evaluation that the CBL material was not boring, identified from the Lickert scale of disagree to strongly disagree, figure 10.2.

12. Working through this courseware bored me
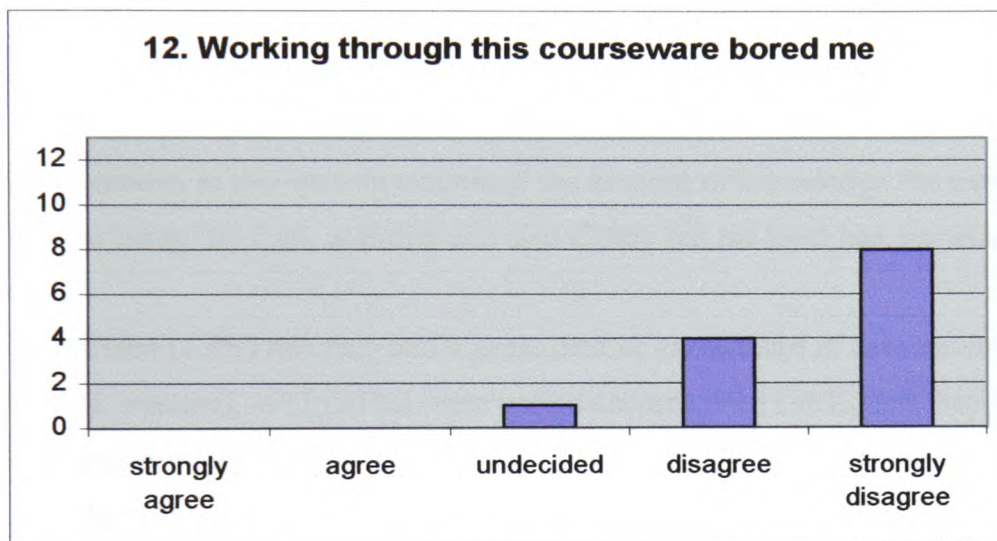
Figure 10-2. Results from Question 12.

All the students felt the CBL lessons were a useful learning tool, again concurring with the first evaluation result for this question.

The CBL material provided a number of interaction opportunities for the users, these were drag & drop, clickable objects or text entry. Preferences for these interactions were mixed, 10/12 (83%) expressing a preference for drag & drop, 7/13 (54%) with a preference for clickable objects and only 4/12 (33%) with a preference for text entry, again showing a high correlation with the findings of the first evaluation, figure 10.3.

14. The method I most preferred for making choices was

Legend: drag & drop, clickable object, text entry

X-axis categories: strongly agree, agree, undecided, disagree, strongly disagree

Figure 10-3. Results from Question 14.

The last two questions in this section examined the amount of knowledge the users felt they had prior to using the CBL material and also if they felt his level had increased after using the CBL.

Three of the thirteen (23%) felt they had a great deal of knowledge of data modelling prior to using the CBL material, 4/13 (31%) were undecided and 6/13 (46%) felt they had no knowledge of data modelling prior to using this CBL. Of those who felt they had a great deal of knowledge of data modelling prior to using the CBL, 2/3 (66%) felt this knowledge had increased with the use of the CBL and one was uncertain if it had increased. Of those who were undecide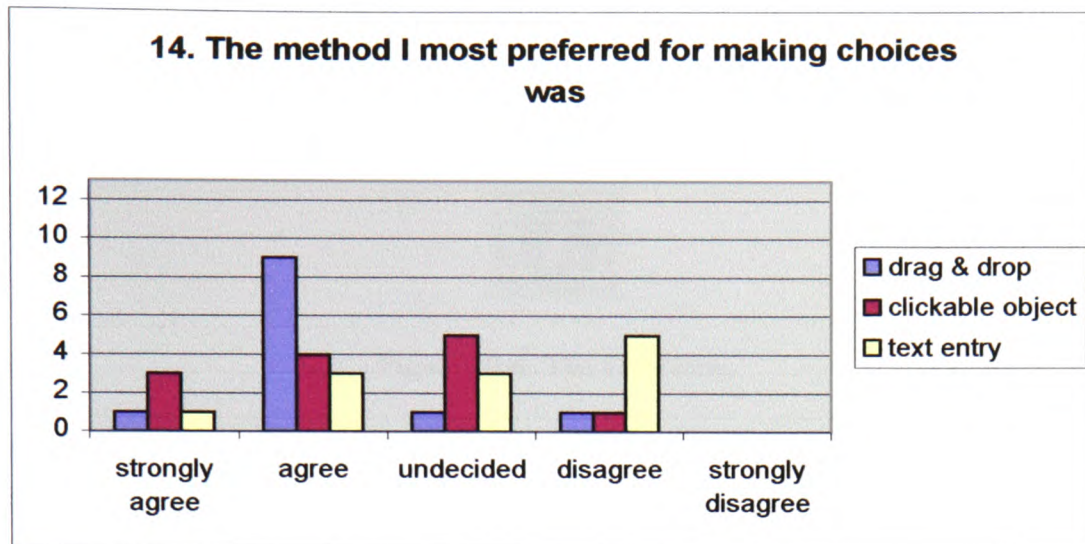d prior to using the CBL, 2/4 (50%) felt their knowledge had increased and 2/4 (50%) were still uncertain and of the final 6 who had no prior knowledge of data modelling, 5/6 (83%) felt they had increased their knowledge of the subject and 1/6 (17%) felt his knowledge had not increased. Overall the majority of people 9/13 (69%) felt there was an increase in their knowledge of the subject, 3/13 (23%) were undecided if their knowledge had increased or not and one person (8%) felt their knowledge had increased. An excellent endorsement considering the problems people had with the on-screen messages and some of the colour combinations. In interviews and from comments on the questionnaire, of those students who were undecided if their knowledge had increased or not, most expressed the opinion that their knowledge level had increased but they were unsure by how much.

Help

This section looked at the help available within the CBL lessons and also the feedback provided during the use of the system.

Figure 10-4. The Help Icon.

Help was provided via a button, figure 10.4. This help facility provided help on navigation controls and titles but not on the content. This icon was used as it was similar to a familiar symbol from the highway code where it is used on signs that indicate tourist information locations, the icon used in the WI.S.D.E.N. CBL material was also intended to indicate to the user that information was available.

Students were asked if they felt help was adequate and readily available when needed, 8/11 (73%) agreed whilst 3/11 (27%) were undecided, one user stated they did not use the help and one did not answer. This was an improvement in the results from the first evaluation where 8/14 (57%) were undecided as to its adequacy, with a further person (7%) finding it inadequate, leaving 5/14 (36%) who were satisfied. The feedback provided when users made mistakes was approved by 11/13 (85%) with only two people (15%) undecided over the benefit it provided, again an improvement over the first evaluation.

Level

This section aimed to uncover some opinions about how the users felt about learning from the CBL lessons and whether they felt the level the material was aimed at was to low i.e. that the material was too easy.

As in the first evaluation, there was a mixed response to the question on whether the CBL material was more like learning from a book or a tutor. The responses show that 3/13 (23%) felt it was like learning from a book, 4/13 (31%) were undecided and 6/13 (46%) felt it was not like learning from a book. 7/13 (54%) felt it was like learning from a tutor, 2/13 (15%) were undecided and only 4/13 (31%) felt it was not like learning from a tutor.
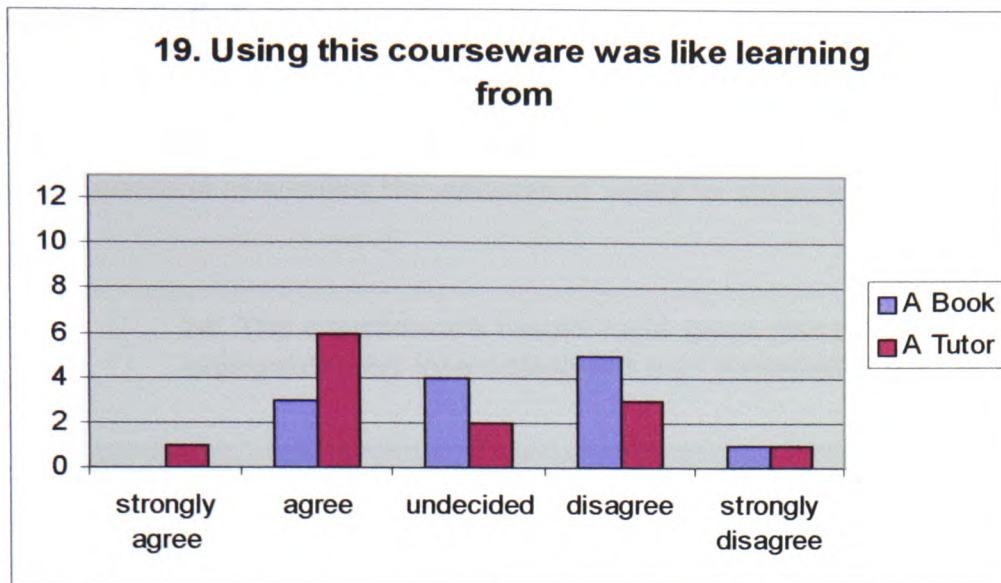
Figure 10-5. Results from Question 19.

3/13 (23%) felt it was like both a book and a tutor, the same number felt it was like neither, 3/13 (23%) felt that it was like a tutor and not like a book and no-one felt it was like a book and not like a tutor, figure 10.5. 9/13 (69%) did not feel that the CBL lessons were too easy, 3/13 (23%) were undecided and only one person (8%) felt it was too easy. This was an improvement over the first evaluation results where 7/14 (50%) did not find it too easy, one person undecided and 6/14 (43%) reporting it as too easy. This is the result that was anticipated as a wider mix of questions were incorporated into the self assessment section so that learners could attempt both simple and more complex questions as a result of the first evaluation. Also, the introduction of a reason for using the courseware i.e. trying the pre and post-tests, appeared to concentrate the learning and deepened the knowledge the students sought from the courseware, resulting in the positive responses to this question.

Overall Impressions

This section sought to find the users impressions with regard to the use of the mouse and keyboard; the language and examples used; whether the CBL material contained any racist or sexist elements; and attitudes to the inclusion of additional multimedia elements, in this case audio and video.

With respect to the use of the mouse and keyboard, 12/13 (92%) felt their use was intuitive with only one user (8%) undecided over this. All users felt the examples used were explanatory and aided understanding, similarly, 12/13 (92%) felt the language used presented no problems with only one person (8%) undecided. These results, again, show

an improvement over the first evaluation findings. 12/13 (92%) of the users found no evidence of racist or sexist elements with just one person (8%) undecided on these points. As far as the introduction of multimedia elements was concerned , there was a mixed response to the question of whether the inclusion of sound or video would have enhanced the courseware.
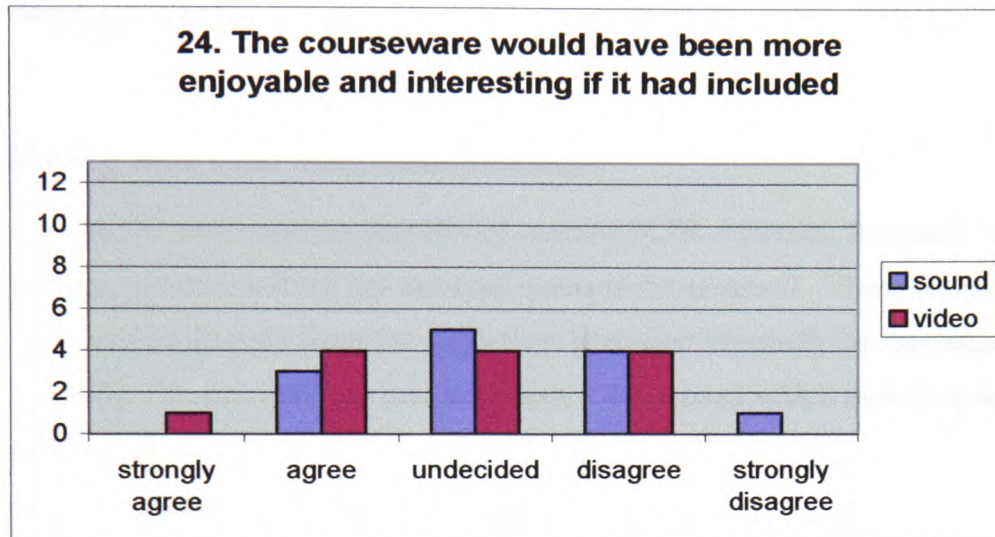
Figure 10-6. Results from Question 24.

3/13 (23%) users felt that the CBL would have been improved with the inclusion of sound, 5/13 (39%) undecided and the same number disagreeing that the inclusion of sound would improve the CBL material. Similarly with video, 5/13 (39%) would like to see video in the CBL, 4/13 (31%) undecided and the same number against the inclusion of video, once again a very mixed reaction, figure 10.6.

10.3.2 Conclusion

Some of the positive comments given by the students included the ease of use and the provision of interaction opportunities, being able to work at their own pace and the animated figure which appeared with feedback in the self-assessment sections. Conversely some of the criticism was levelled at the animated figure and also at the combinations of the text and background colours though some students had no criticisms of the courseware. The most important point that needed consideration was the colour combinations on certain screens, these were amended to provide more clearly contrasting options e.g. blue text on a white background and text boxes which, whilst having a variety of background colours, always provided a good contrast between foreground and background colours.

Once again, the response to the use of CBL material was very positive, users liked the ability to work at their own pace and being able to re-visit difficult areas to clarify understanding. The CBL material was improved in line with the suggestions and comments provided by the students and another usability and learning effectiveness evaluation activity planned. However, at this point the students involved in this exercise had undertaken the learning effectiveness evaluation and their results recorded and analysed.

10.4 Teaching and Learning Effectiveness

In order to judge the teaching/learning effectiveness of CBL material, the users' ability to meet the learning outcomes from the material needs to be assessed. These learning outcomes are derived directly from the objectives provided originally by the subject expert (Laurillard, 1995). To this end, pre and post-tests can be used which include questions related directly to those learning outcomes, in this way, the difference in the results from the pre-test to the post-test provide an indication of an improvement or a problem. An increase would indicate an improvement, a decrease or no increase would indicate a problem. The evaluator must be careful that the results relate directly to the use of the CBL material and that no external factor has influenced the result e.g. the use of text books helping to improve scores or the incorrect use of the CBL lessons which might affect any poor score. Thus, casual observation was used during the use of the CBL material to determine if the pre and post-tests were giving a true indication of any change. Both the pre-tests and the post-tests were the same i.e. they contained exactly the same questions. The pre and post-tests were constructed of multiple choice questions to provide an objective testing facility which was both quick to answer and simple to grade. The quantitative data produced could be compared to provide a correlation between the two sets of results (Alessi & Trollip, 1991).

Multiple choice questions are composed by providing a stem and a number of options one of which is correct, the incorrect answers are called the distractors (Race, 1994). Criticism of multiple choice questions include the fact that students can guess the correct answer since they have to choose one from four of the answers presented. The addition of confidence assessors to the test paper was designed to detect if the student had guessed the answer or had attempted to answer the question correctly (Gardner-Medwin, 1995). The hypothesis was that students who guessed the answer would be less confident that it was correct than students who believed they knew the answer. Thus, in a pre-test, where the

student had no knowledge, the confidence level was expected to be low but in a post-test where they had used the CBL material the confidence was expected to be much higher. A typical multiple choice question has a stem with four possible answers, other numbers of options may be used but four is proposed as optimum (Gibbs et al, 1988, Race, 1994), the confidence indicators were implemented as faces with different expressions ranging from a very sad face to a very happy face, this was intended to offer a 5 point Lickert scale from very under-confident to very confident. Each question was derived from the original learning objectives and included a confidence assessor, the test comprised twenty seven questions in total, see figure 10.7.

Original learning outcome: identify degree of relationships (1:1, 1:N, N:M);

16. For the following relationship definition, choose the correct degree classification. Each salesperson is allocated a company car, a company car is assigned to one salesperson:-

	Sales Person		Car
A	1	:	N
B	N	:	1
C	1	:	1
D	M	:	N

Figure 10-7. Example of multiple choice question.

10.4.1 Results from Teaching and Learning Effectiveness Evaluation

Results from the pre and post-tests were calculated as a percentage, the two were compared to detect any difference. The confidence levels were calculated by assigning a value of one to the very under-confident symbol and five to the very confident symbol, this gave a minimum mark of 27 and a maximum mark of 135.

MSC Evaluation (2)
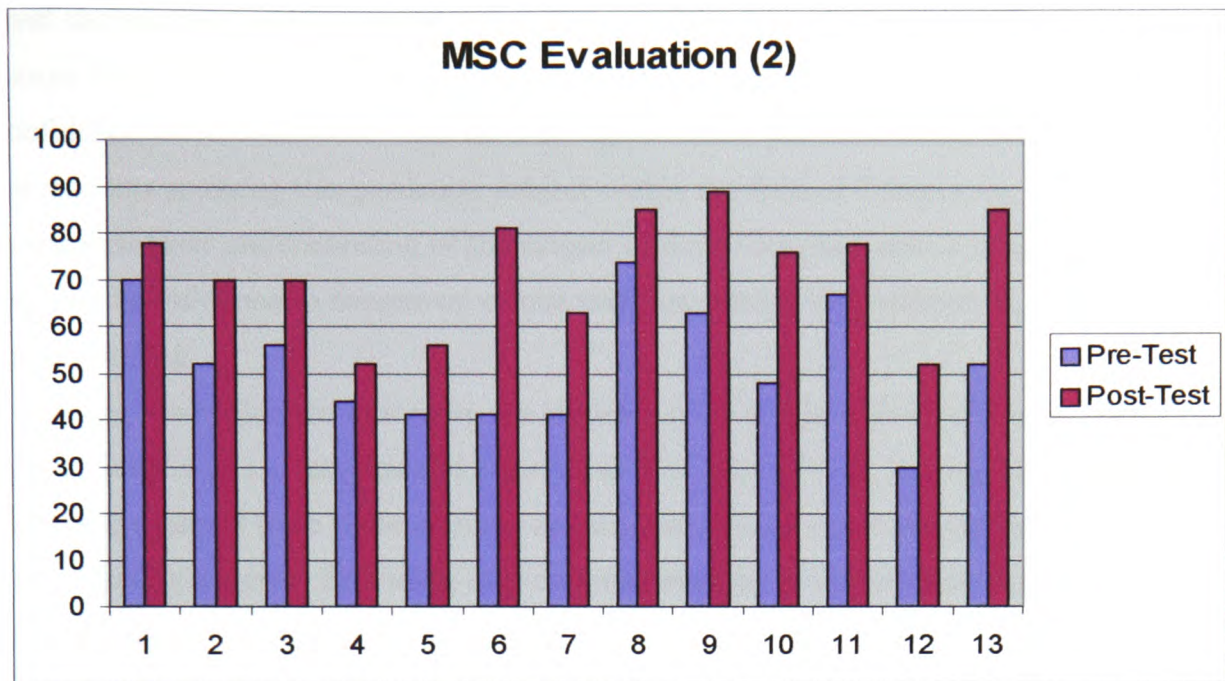
Figure 10-8.Pre and Post-Test Results.

Comparison of the results from the pre and post-tests did indeed show that all students had improved their scores, figure 10.8. Confidence levels reflected the questionnaire results where many students felt they had little knowledge of the subject prior to using the courseware, figure 10.9.

Confidence levels Pre and post-tests
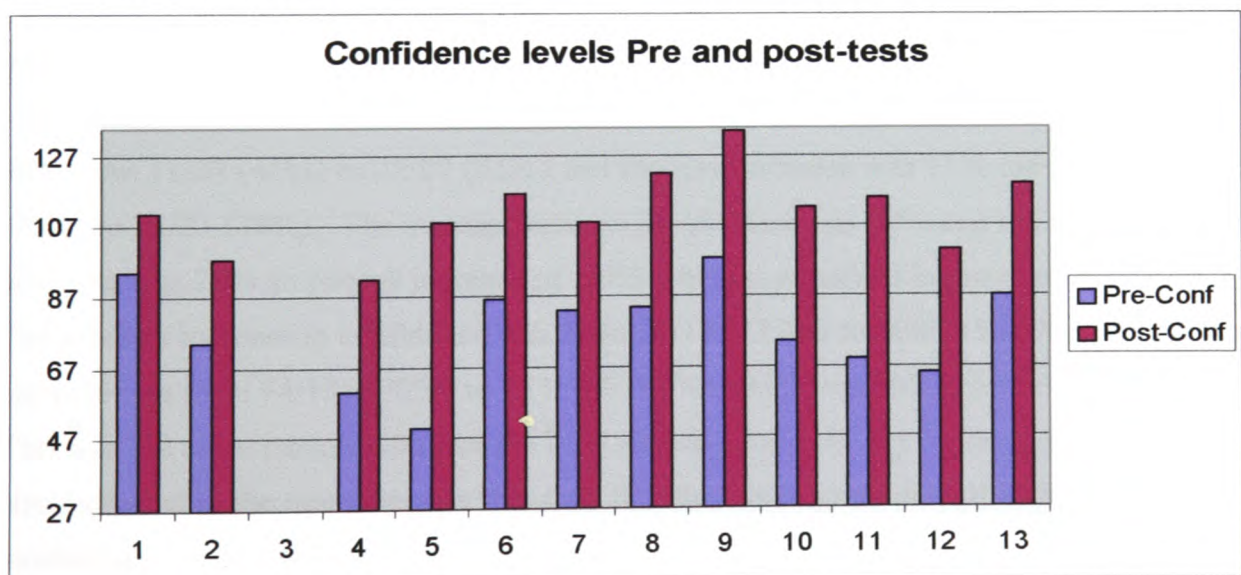
Figure 10-9. Pre and Post-Test Confidence Levels.

Student 5, in particular, felt very strongly that (s)he had no knowledge of the subject before using the courseware but that their knowledge had markedly improved once they had used it and the results from the pre and post-tests show a clear improvement. Their confidence

125

level showed the highest increase of all the students. Overall, the average increase in scores from the pre-test to the post-test was 42% and the average increase in the confidence levels was 48%. This level of improvement provides an excellent grounding for students studying this particular subject within the field of Computing, it enables them to increase their understanding of the subject at their own pace and to progress with an increased confidence to deepen or extend their knowledge and, ultimately, their understanding.

Following the evaluation exercises, the students were taught E-R modelling in the conventional way i.e. lectures and tutorials as in previous years, this was done so that knowledge gained from those lectures and tutorials would not influence the results from the pre and post-tests. Providing conventional teaching of this subject was also necessary as the quality of the final courseware was still under evaluation, the ultimate intention, however, is to provide both teaching methods to enhance the learning experience rather than to replace one with the other.

10.4.2 Conclusion

Overall, results from the usability evaluation showed one or two problem areas which were amended as indicated in section 10.3. These problems, however, do not appear to have adversely affected the teaching/learning effectiveness evaluation. Results from the pre and post-tests show a marked improvement in the scores obtained and a very favourable increase in confidence levels. The greatest increase from the pre-test to the post-test was 98% from 11/27 (41%) to 22/27 (81%) and the least increase was 11% moving from 19/27 (70%) to 21/27 (78%). The average score in the pre-test was 52% and the average in the post-test was 72% an overall increase of 20% which is a marked improvement.

The greatest increase in confidence was from 50/135 (37%) to 108/135 (80%) and the least increase was from 94/135 (70%) to 111/135 (82%) which was still a marked increase. This was the same person who had the least increase from the pre to the post-test and who also indicated in the questionnaire that they felt they had a great deal of knowledge of data modelling prior to using the CBL material but were uncertain if that had increased after using the CBL. The average confidence level in the pre-test was 77/135 (57%) and the average in the post-test was 111/135 (83%) an increase of 26%.

Obviously, this indicates that the students using the CBL material had an improvement in their knowledge of the subject from the pre-test to the post-test and at the same time, their confidence that they knew the answers also increased. This was a very positive outcome

from this evaluation exercise which was anticipated to be repeated in the subsequent evaluation which had been planned.

10.5 Third Evaluation

10.6 Usability Evaluation

In this evaluation exercise, the group consisted of 21 students, six females and fifteen males, all enrolled on a Masters course in Computing, however, only 15 students returned the questionnaire, five females and ten males. In this exercise, the group was given a pre-test, allowed to use the CBL material immediately afterwards but took the questionnaire away to fill it in and return it one week later when they would be given the post-test. The questionnaire was the same as that used in the previous two evaluations and the tests the same as those used in the second evaluation.

10.6.1 Questionnaire Results

Navigation

Results from the navigation section of the questionnaire showed a marked similarity to those found in the second evaluation, if anything, the results from the third evaluation were slightly more positive than the second evaluation. 13/15 (87%) of the students felt the functionality of the buttons was obvious with only 2/15 (13%) undecided, no-one felt they were not intuitive. Of the two students who were undecided, one of the respondents was undecided over all of the questions on navigation, he also expressed the opinion that the navigation was the most unattractive feature of the CBL material. 12/15 (80%) of users felt they would not get lost in the CBL lessons with 3/15 (20%) undecided, no-one felt they would get lost. 11/15 (73%) felt they could choose relevant routes through the material and 4/15 (27%) were undecided, no-one felt they could not choose relevant routes. Similarly, 11/15 (73%) felt they were always certain how to move through the CBL lessons with 4/15 (27%) undecided. Over these four questions on navigation no-one expressed a negative response to the navigation facilities, though some people were undecided on a number of issues. This was still, however, an improvement on the second evaluation and could possibly be attributed to the alteration in the on-screen messages mentioned as one of the features amended from the results of that evaluation.

Screen Layout

This section queried attitudes on screen layout and appearance, together with how the users felt about any graphics used. 14/15 (93%) of the users felt the screens in the CBL material were attractive only one person (7%) was undecided over this question, this person was also undecided over whether the size of the graphics used were appropriate, he was one of three (20%) though he did say that more graphics would have enhanced the package. 11/15 (73%) felt the graphics used were of an appropriate size with one person (7%) feeling they were not, though there was no comment made as to why he felt this way. 12/15 (80%) felt the screen was well laid out with 3/15 (20%) undecided, no-one expressed a negative reaction.

Learning

Surprisingly, 2/14 students (14%) expressed their preference for text books and lecture notes rather than the CBL material with 5/14 (36%) undecided, however, the remaining 7/14 (50%), preferred the CBL material, one student did not reply to this question. 14/15 (93%) felt the CBL material was a useful supplement to text books and lectures and the same number said they would use the CBL lessons again, in each case a sole student (7%) was undecided. Only one student (7%) felt the CBL material would not make a useful revision tool, this respondent was one of the two people who said they would rather use text books and lectures to learn rather than this CBL material. One other (7%) was undecided, this left 13/15 (86%) who felt it would be a useful revision tool. This was a slightly poorer result than those obtained on both the previous evaluations, however, only 5/15 (33%) had worked with computers prior to embarking on the course, thus, the majority 10/15 (67%) were novice computer users with no previous experience of this type of learning activity. 14/15 (93%) felt they would use the CBL material again with 1/15 (7%) undecided which could reflect their inexperience and need to familiarise themselves with this type of learning environment.

No-one felt bored working through the CBL material, as found in the previous exercise, though 2/15 (13%) were undecided. 14/15 (93%) agreed the CBL material was a useful learning tool, with a single student (7%) undecided, again, this was the same student who preferred books and lectures to this type of learning.

The preference for the various interaction mechanisms was mixed, nevertheless, clickable objects were preferred by the most students. 11/15 (73%) preferred clickable objects, 9/14 (64%) preferred drag & drop and only 2/14 (14%) preferred text entry.
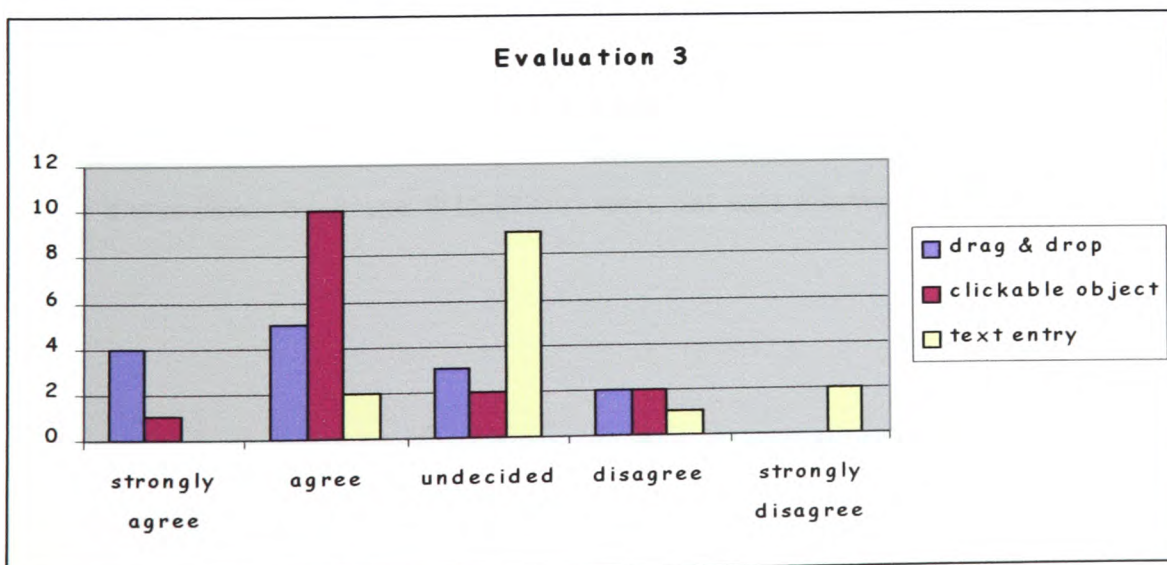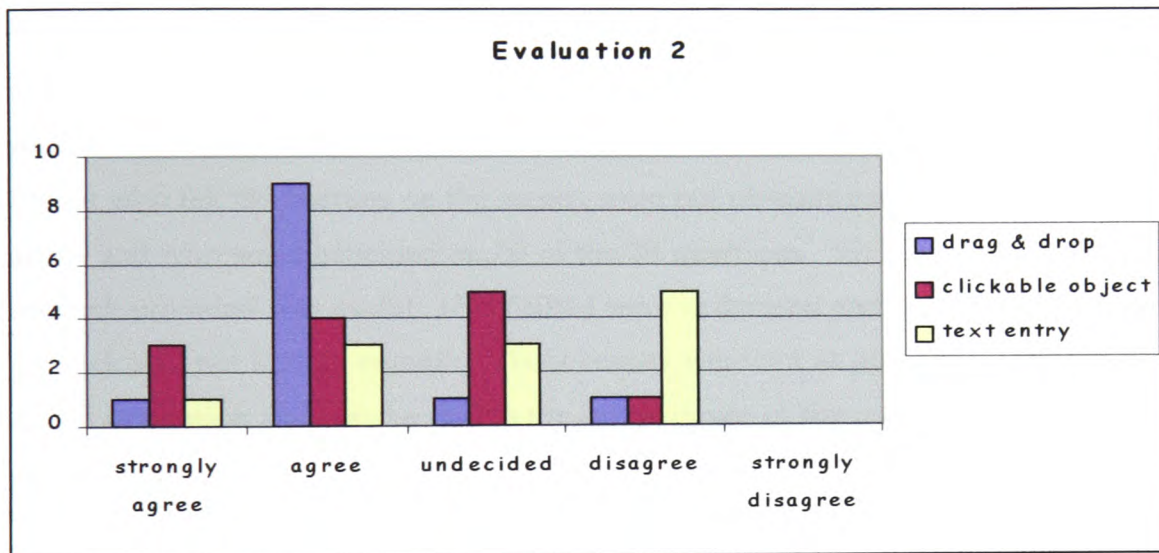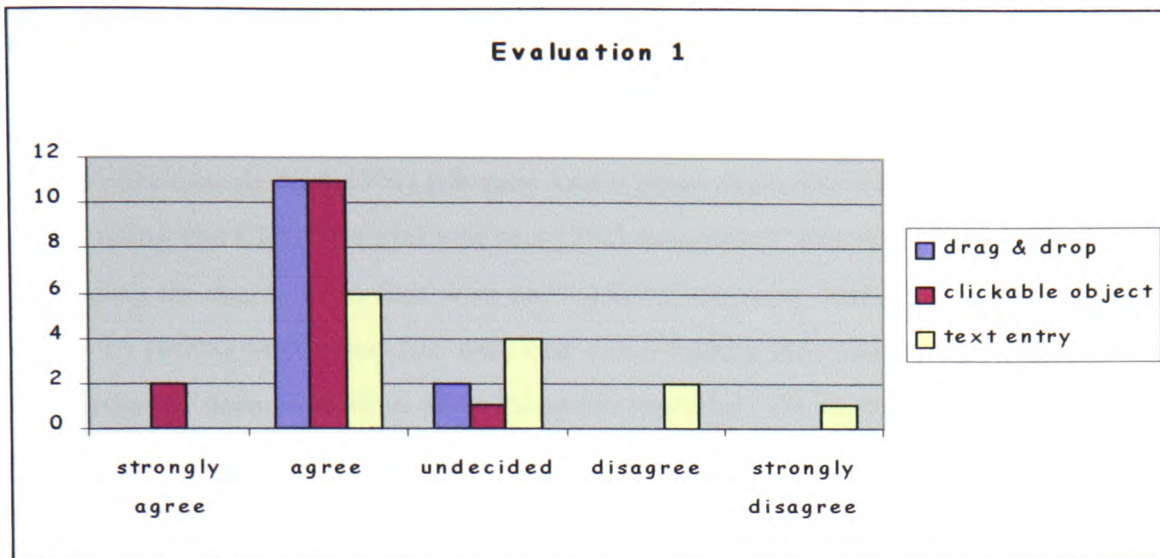
Figure 10-10. User Interactions.

This was very similar the first two sets of results for this question, where there was a mixed reaction , in general, to the interaction types available. In every evaluation, however, the least popular interaction type was text entry, figure 10.10.

Where only one student (7%) felt they had a great deal of knowledge of data modelling before using the CBL material and one (7%) undecided, this number rose dramatically after using the material so that 8/15 (53%) felt they, now, had a great deal of knowledge with 6/15 (40%) undecided and only one (7%) feeling they did not have a great deal of knowledge of data modelling after using the material. As in previous evaluations several students felt they were unable to judge how much their knowledge had increased but did feel it had increased from pre-CBL levels.

Help

The help facility, as in previous evaluations, only provided help on navigation controls and titles. 8/15 (53%) felt the help facility was adequate and always available, 6/15 (40%) were undecided and only one student (7%) felt it was inadequate, this student was the same student who felt the buttons on the screen were not obvious and disliked the navigation facility and who was undecided on 10 of the 24 questions. 11/15 (73%) agreed the feedback provided was useful, 3/15 (20%) were undecided and one person (7%) felt the feedback was not helpful enough. These results were not as good as those found in the second evaluation and may be due to the inexperience of the users with this type of learning activity or the lack of content specific help.

Level

Most of the students, 8/15 (53%) felt the CBL lessons was like learning from a tutor, with the majority, 11/14 (79%) affirming it did not feel like learning from a book, a welcome result since much of the criticism of CBL material has been directed at the fact that it, merely, represents a computerised page turning exercise. Once again, 3/14 (22%) were not sure is it was like a book and 5/15 (33%) were not sure if it was like a tutor, figure 10.11.
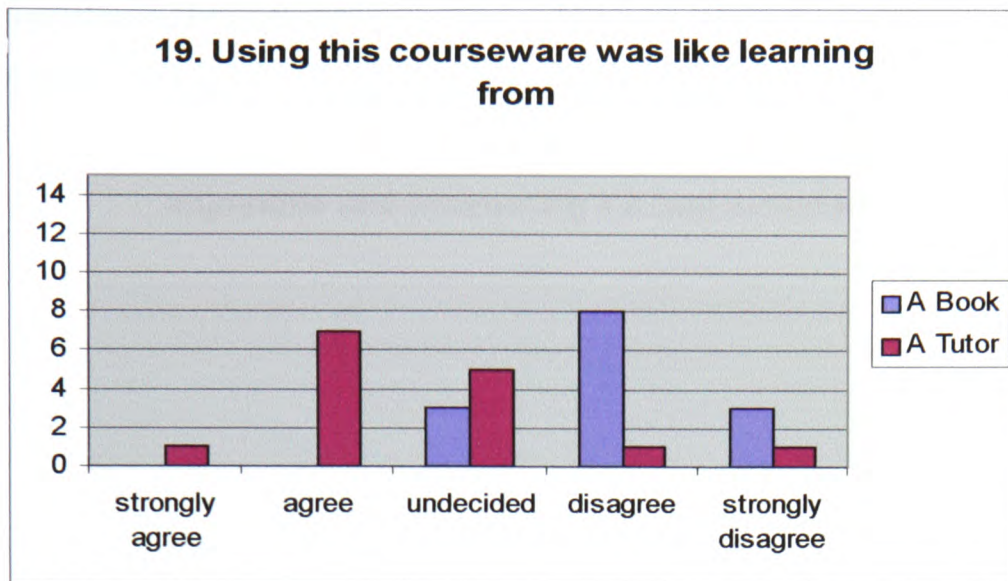
Figure 10-11. Results from Question 19.

Only one student (7%) felt the CBL material was too easy, with 4/15 (27%) undecided, the rest, 10/15 (66%) felt the CBL lessons were not too easy. This was a very similar result as the second evaluation and, again, an improvement over the first evaluation where 43% had reported the CBL material was too easy.

Overall Impressions

The use of the keyboard and mouse within the CBL lessons does not appear to have caused any problems with 10/15 (67%) concurring and 5/15 (33%) undecided. This, yet again, may be attributable to the inexperienced computer users.

The next few questions relating to language, examples, racism and sexism found a number of the students undecided, though no-one found there was any problem. The majority of the students 14/15 (93%) had no problem with the language, 13/15 (87%) had no problem with the examples used, 12/15 (80%) found no racism and 11/15 (73%) felt there was no sexism. One student was undecided over the language, two students over the examples, three over racism and four over sexism, this represented 6.7%, 13.3%, 20% and 26.7% respectively. This was a surprising finding since no other evaluations had uncovered any similar information and it, thus, requires further investigation to establish the validity of the data.

Finally, again, responses on the inclusion of multimedia were mixed, more students expressed the opinion that sound would have enhanced the CBL over video, figure 10.12. However, during the interviews and in some of the open ended questions, they stated that

the inclusion of sound in a laboratory setting could prove distracting and could understand why it was not included.

24. The courseware would have been more enjoyable and interesting if it had included
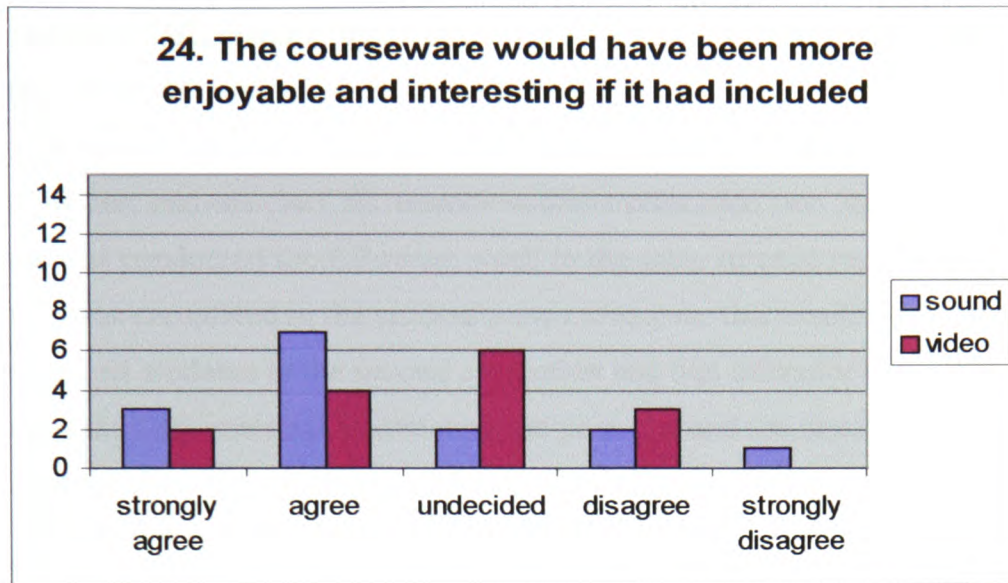
Figure 10-12. Results from Question 24.

10.6.2 Conclusion

One of the most positive comments on the CBL material and its use came from one of the female students undertaking this evaluation. She wrote "It was one of the most enjoyable learning experiences I've had since joining this course because I actually felt that I had managed to understand a great deal in a relatively short space of time." Other positive comments included the interactivity which most of the students enjoyed and felt was a plus, the ability to re-visit sections until understanding was clarified and the animated figure which accompanied feedback in the self assessment sections, though, this also came in for some criticism as tacky and patronising. Several students expressed no criticism of the CBL material and found the whole experience very positive. A problem was encountered in evaluation two, several students reported having difficulty moving through the CBL lessons. This was, to some extent, due to an inappropriate colour combination of the text and background colours for the user instructions, this problem did not appear in this evaluation suggesting that the changes made had been effective in alleviating the problem.

10.7 Teaching and Learning Effectiveness

The teaching and learning effectiveness was again judged by assessing the students before and after using the CBL material by means of a multiple choice pre and post-test which incorporated a level of confidence assessment. The pre and post-test were identical but the length of time between the two tests was extended to one week, this enabled the students to complete a pre-test and use the CBL lessons within a scheduled two hour tutorial session, the post-test was conducted the following week in the same tutorial session and the questionnaire was completed in the student's own time over that week. This format was adopted as several students in the second evaluation had had difficulty in completing the pre-test, using the CBL material, completing the post-test and the questionnaire within the time allocated for the evaluation exercise which was a normal two hour tutorial session.

10.7.1 Results from Teaching and Learning Effectiveness Evaluation

As before, the pre-tests were compared with the post-tests, as were the levels of confidence before and after using the CBL material. Results from the pre and post-tests were calculated as a percentage, the two were compared to detect any difference. The confidence levels were calculated by assigning a value of one to the very under-confident symbol and five to the very confident symbol, this gave a minimum mark of 27 and a maximum mark of 135.
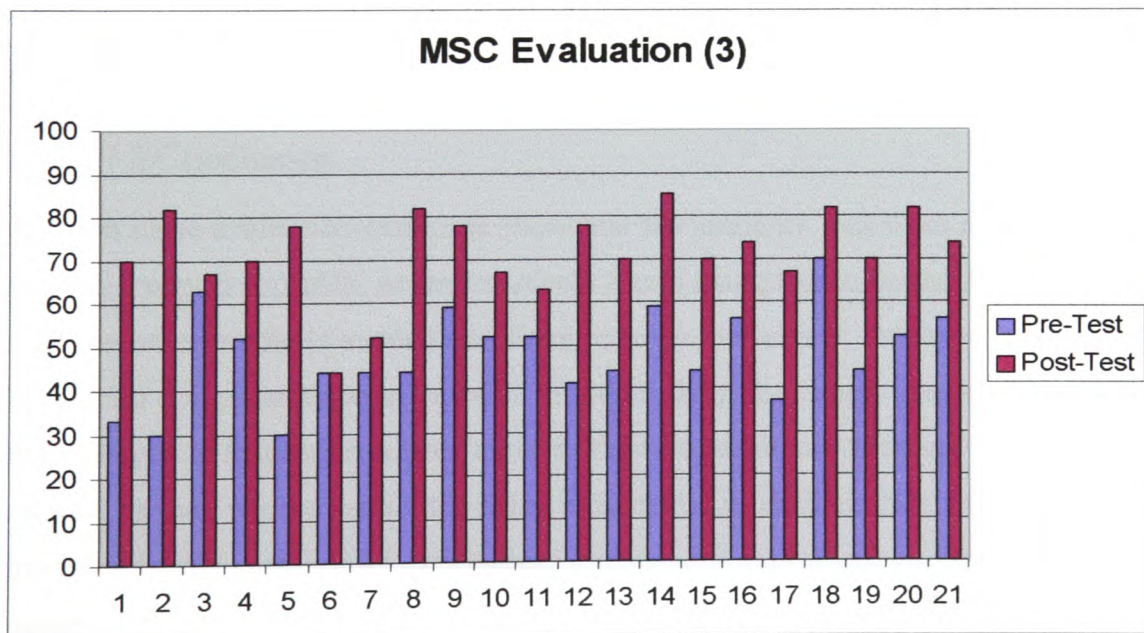
Figure 10-13. Pre and Post-Test Results.

133

This comparison revealed that three students had increased their scores from the pre-test to the post-test by over 100%, showing a dramatic increase in their knowledge, figure 10.13. Only one student did not register an improvement but despite this there was an increase in the confidence level for that student which suggested that (s)he was more confident of the answers given, figure 10.14.
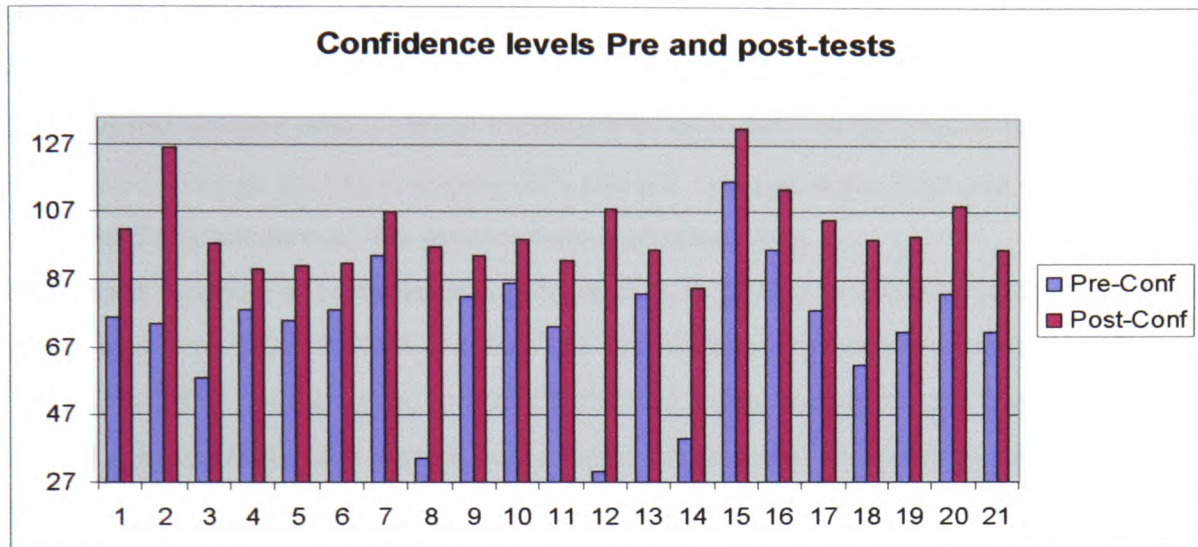
Figure 10-14. Pre and Post-Test Confidence Levels.

This graph shows that several students had a marked improvement in their confidence in the answers given in the post-test as opposed to the pre-test. Overall, the average increase from the pre-test to the post-test was 57% and the average increase in confidence levels registering as 53%.

10.7.2 Conclusion

Results from these evaluation exercises show that the students' reactions to using the CBL material were very favourable, several students found using the computer based material preferable to other methods available to them to learn the subject. Many of the students had little or no computer experience prior to undertaking the course they were studying and felt that this type of learning resource allowed them to make mistakes and progress as fast or as slowly as they wished without it being judgmental or critical, thus, they felt it provided a rewarding and useful experience.

The outcome from the pre and post-tests was that students could improve their ability to meet the learning outcomes as set out by the subject expert and their confidence in their knowledge increased after using the CBL material. There was a consensus that the CBL material could provide both a learning opportunity and a revision opportunity.

The greatest increase from the pre-test to the post-test was 160% from 8/27 (30%) to 21/27 (78%), surprisingly this result came from the same student who was the only person to claim to have a great deal of knowledge of E-R Modelling prior to using the CBL material and was unsure if their knowledge had increased. The least increase was recorded as no increase from the pre to the post-test for one student but for whom the confidence increase was from 78/135 (58%) to 92/135 (68%), this was surprising as the student scored 12/27 (44%) in the pre-test and the same in the post-test. This student, unfortunately, did not submit a questionnaire which makes it difficult to speculate on the reason for this finding. The average score in the pre-test was 48% and the average in the post-test was 72%, an increase of 24% which was 4% greater than evaluation two.

The greatest increase in confidence was from 30/135 (22%) to 108/135 (80%) and the least increase was from 94/135 (70%) to 107/135 (79%) from a person whose score only increased by 18%. Some of the lowest confidence levels during the pre-test phase were recorded against the female participants despite the fact that their scores were no worse than their male counterparts. The average confidence level in the pre-test was 73/135 (54%) and in the post-test was 102/135 (76%) which was a very gratifying increase of 22%. This increase was 4% less than the second evaluation but one participant had a pre-test confidence level of 116/135 (86%) with a score of 44% and a post-test confidence level of 132/135 (98%) with a post-test result of 70% which shows how subjective the confidence of individuals can be.

10.8 Overall Conclusions

Over the course of the evaluations, amendments to various aspects of the CBL material as a result of feedback from students and subject experts enabled the refinement of the material to eliminate any problems encountered. Initial, unstructured evaluations were conducted by subject experts who examined the content to ensure it was complete, consistent, clear and correct. The qualitative feedback generated from the expert evaluations enabled the correction and/or clarification of the content to pave the way for the usability evaluations.

Usability evaluations enabled refinement of the user interface in respect of fonts, colours, navigation and layout. Initial difficulties with certain colour combinations were identified and modified to improve background and foreground contrast. This led to a corresponding improvement in results from the questionnaires on navigation and screen design. Navigation was affected by the colour combinations as the on-screen instructions were in

blue but the background was green, a poorly contrasted combination. Subsequently, the blue messages were set on a white rectangle, and the combination set on the green background (see section 10.3.1.1). This alleviated any problems due to the learners finding difficulty in reading the on-screen messages and led to a much more positive attitude from the students.

More examples and problem solving opportunities were also requested and subsequently included which led to a drop in the number of respondents who felt that the CBL lessons were too easy. Interaction with the CBL material was a popular feature as was the ability to practice the new knowledge through exercises and problem solving opportunities.

The overwhelming response from the learners was that they felt the CBL material should be provided as a supplementary resource rather than as a replacement activity. This concurred with the findings of the Coopers & Lybrand report where they were "more convinced of effectiveness in cases where material was envisaged as an adjunct to, rather than as a replacement for, teacher/student interaction" (Coopers & Lybrand, 1996).

When questioned students felt they had learned more about the subject after using the CBL material compared to what they had known prior to using it. This was supported by the findings from the learning effectiveness evaluations.

During the learning effectiveness evaluations, all the students, except one, who used the CBL material registered an improvement in the post-test scores over the pre-test scores. The single exception had exactly the same pre-test score to post-test score. However, without exception, the confidence levels of the students rose considerably from their pre-test levels to their post-test levels. This indicated that the students were much more confident that the answers they were giving were correct rather than guessed. In several cases the increase in the confidence levels was very marked, the highest overall increase was from 22% in the pre-test to 80% in the post-test.

This evaluation has shown that the CBL material produced with the UDRIPS model enables students to improve their knowledge of the subject of E-R Modelling in line with the learning outcomes specified by the subject expert. They also acquire and refine that knowledge at their own pace and in a manner characteristic of their individual learning style.

In these evaluation exercises students who used the CBL material were, also, subsequently taught using conventional lectures, seminars and tutorials. This was felt to be the optimum way of teaching the subject of E-R Modelling, offering a self-paced introductory element which allowed the students to increase their knowledge and confidence in the basics of the

topic, prior to attending the lectures and seminars. This, effectively, offered two opportunities to cover the objectives for the topic, or allowed the tutor to proceed to teach the more complex concepts, secure in the knowledge the students had a good grounding in the basics.

Chapter 11

Conclusions and Future Work

11.1 Introduction

This chapter summarises the discussion, states the contribution and discusses some of the limitations of the work. It also outlines possible future work in this area.

11.2 Background to the Project

The original aim of the W.I.S.D.E.N. project was the development of interactive computer-based learning materials in the main branches of software systems analysis and design. The project members were a consortium based, geographically disparate group with backgrounds in a variety of areas including, but not limited to, software systems analysis and design. The objectives of the project were:

1. to develop and disseminate a wide range of interactive CBL material in the area of software specification and design;

2. to develop materials with a commonality of approach;

3. to critically evaluate materials produced for the purposes of establishing best practice for developing interactive CBL materials.

With the need for collaboration and co-operation between consortium members, it quickly became necessary to form standards to address the objective of developing material with a commonality of approach. As development progressed, however, it became apparent that there was no standard model or method available for developers to produce CBL lessons which had pedagogic validity but which also provided a sound software engineering approach focussed on software quality. Initial prototypes were built which were demonstrated to consortium members, but these early prototypes displayed a diversity with respect to navigation, structure and general interface standards. Working in a group highlighted the need for a mechanism whereby CBL material could be produced which demonstrated a "commonality of approach", an original aim within the project. Decisions

138

on standards took a considerable amount of time. The template adopted from Sheffield Hallam provided an environment which supported the sequential and direct navigation structures agreed within the consortium. However, there was no in-built model of what the sections within the template should contain and, hence, how the lesson should be structured. Individual sites were developing material in a wide variety of topic areas and work was progressing very slowly. A standard for the CBL lessons was urgently needed. This need to produce a model or method to address lesson structure lead to the definition of the UDRIPS design and development model.

11.3 CBL development Model

The developers and subject experts involved in the W.I.S.D.E.N. project were familiar with existing software design and development methods used to produce a variety of software systems. To this end, there was an anticipation that a corresponding method and its associated tools and techniques would be available for CBL development. Whilst several CBL development methods were discovered, there were very limited associated tools or techniques to complement them. This formed the impetus for the research detailed in this report and the subsequent derivation of UDRIPS.

To develop UDRIPS, it was necessary to distil the key principles from the two fields of software engineering and pedagogy. A CBL system differs from a conventional software system in that its primary aim is to facilitate learning for a user. It is still, however, a software system and, as such must be constructed so that it is efficient and effective. The hypothesis for UDRIPS was that a CBL development model could be built that combined the key principles from the software engineering field and the key principles advocated in the pedagogic theories. This would, then, lead to the development of CBL systems that facilitated learning whilst at the same time providing quality software systems that were efficient and effective from the point of view of the computer environment.

CBL development was seen as involving the collaboration between a software developer and a subject expert to produce didactic systems to satisfy a diverse population of learners. This appeared analogous to the development of more conventional software systems where a software engineer worked in collaboration with the end users to produce software systems for a diverse population of users. The key difference in the two systems produced was the didactic/pedagogic element that was not addressed in conventional systems. Thus, combining the two fields was perceived as the solution to the provision of additional techniques needed to complement existing CBL development methods.

11.4 Contribution to Knowledge

UDRIPS is an amalgamation of principles from the software engineering and pedagogic areas which provides a framework for developers to produce CBL material. The aim in building the model was to ensure that developers could use a structured software method with a pedagogical slant to build valid and quality CBL lessons. UDRIPS was derived to provide a structured CBL development model that would provide a mechanism for building CBL material that mimicked models and methods available in the software engineering field. UDRIPS is implementation independent, is structured, may be easily taught to others yet is flexible enough not to constrain the creativity of the developer. Evidence from the W.I.S.D.E.N. project suggests that this is applicable whether the project is a single developer project or a team based project.

An additional benefit from using UDRIPS is the communication facility afforded to developers in obtaining relevant subject matter for the lessons under construction from the tutors/subject experts. This is achieved since UDRIPS specifies that the contents of the lesson should include objectives, pre-requisites, definitions, illustrative examples, problems for self or tutor assessment and a summary. Thus, the developer knows, for example, that in any lesson (s)he needs some illustrative examples and also some questions for the self-assessment section which can be provided by the tutor.

It is proposed that UDRIPS should to be used in collaboration with other CBL methods. It is not intended as an alternative. For example in Figure 11.1 below, UDRIPS is intended for use in the development and selection of instructional material. In particular, UDRIPS can be used to structure CBL tutorial type systems that have been chosen as part of the instructional strategy in the previous phase. Its applicability to other instructional approaches such as lectures and tutorials is anticipated but, as yet, not tested.
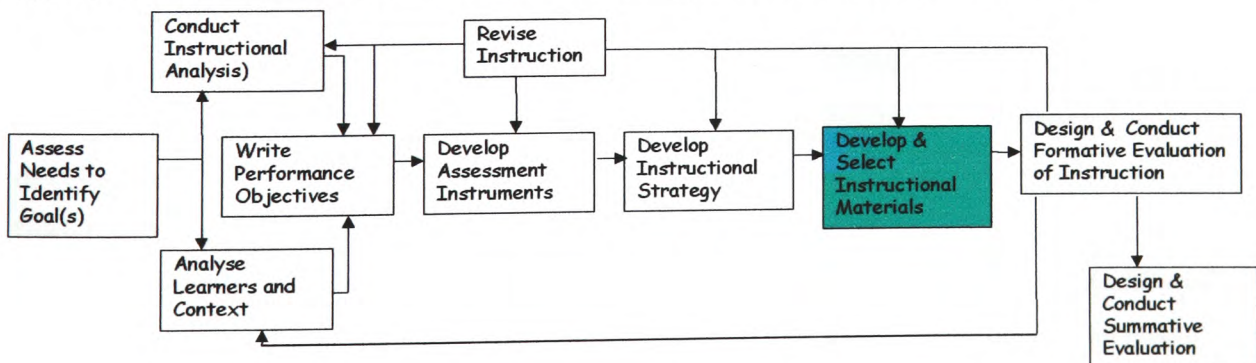

Figure 11-1. UDRIPS & Dick & Carey (Dick and Carey, 1996).

Similarly, in Rapid Prototyping (Tripp & Bichelmeyer, 1990) it can help with the construction of the initial prototype since it can help both the developer and the tutor to communicate their needs and requirements, see chapter 9 for more details.

Thus, UDRIPS may assist the developer who decides to adopt one of the CBL methods available, or indeed any method that has a phase devoted to developing instructional material. UDRIPS is intended as an addendum to the models to complement the overall process rather than as a replacement for them. To this end the models together with UDRIPS form a comprehensive approach to CBL development with UDRIPS providing a technique similar to those provided in the conventional software development field such as E-R Modelling, Data Flow Diagrams etc. in methods such as SSADM.

11.5 Application of UDRIPS to CBL Development

To investigate the efficacy of UDRIPS, it was applied to the development of CBL lessons to teach E-R Modelling. The aim was that UDRIPS would combine with existing CBL development methods (Dick & Carey, 1996, Gerlach & Ely, 1980, etc.) to form a comprehensive approach. To this end, the topic, E-R Modelling, was decomposed into very low level objectives which were then grouped to form sub-topics and hence the lessons. It was at the lesson level that UDRIPS was introduced. In addition, UDRIPS was introduced and demonstrated to other consortium members and subsequently adopted as the standard for lesson structure. One of the most beneficial aspects of adopting UDRIPS was its flexibility. Since lessons spanned many topics and sub-topics in the software design area, the decision to adopt UDRIPS also contributed to the aim of having a commonality of approach. Despite this commonality, it was still possible for the developers to bring their own creativity to the lessons resulting in some innovative and interesting interfaces, examples and assessment. This opportunity to include creative features helped to ease any fears developers had about adopting a prescriptive design model, such as it would lead to boring, repetitive lessons and boring repetitive development.

Initial prototypes were built when UDRIPS was introduced and demonstrated to the group as a whole. The consensus was that it gave a good underpinning structure but allowed creative freedom which suited everyone. Work began in earnest to produce the many lessons required to address the myriad of sub-topics. Early formative evaluations were conducted at member sites and results appeared very positive. It must, however, be emphasised that UDRIPS was not imposed on the consortium members rather it was

adopted voluntarily by them. Evidence of the adoption of UDRIPS can be seen in the many CBL lessons produced as a result of W.I.S.D.E.N. It can also be seen that there are a diverse range of lessons each with a similar structure but very different content. Feedback from the W.I.S.D.E.N. developers showed that with the introduction of a structured CBL development model, the pace of production increased and prototypes could be formatively evaluated and the results fed back to the group. Developers used the UDRIPS model in the early stages of the project and also to produce CBL material in further topics. Benefits associated with the use of UDRIPS included: the ability for a wide range of CBL material to appear as a coherent whole; the ability to increase production in the early part of the project and the ability to obtain the requisite content from the subject experts as and when needed.

11.6 Usability Evaluation

In Glamorgan, formative evaluation showed similar positive results with students expressing their satisfaction with this type of instruction, but, emphasising that they wanted CBL systems to be additional resources rather than a replacement for conventional teaching methods. During the evaluation sessions, learners were observed using the CBL material, interviews were conducted to obtain opinions and questionnaires were distributed to elicit feedback and additional data. The questionnaires comprised closed questions using a Lickert scale, which provided quantitative data, and open questions which allowed the students to express their opinions of the system thus providing qualitative data. These formative evaluations formed the basis of the usability assessment, refining and improving the interface, navigation facilities and assessment strategy of the CBL products.

Early evidence indicated that students who had poor self-confidence found a computer based lesson to be an invaluable tool for assisting in deepening their knowledge in an unfamiliar topic. They were often reluctant to voice their uncertainty in lectures or seminars but felt they could use the CBL material repeatedly until they had mastered the topic. Even students who were more experienced felt the CBL material gave them a good grounding in the topic and both inexperienced and experienced users found the CBL lessons a useful revision tool prior to examinations. Evidence also emerged that the structure of the CBL material allowed the users to choose appropriate routes through the lessons and that this enhanced the learning experience.

11.7 Learning Effectiveness Evaluation

As confidence grew in the usability of the system, evaluation of the learning effectiveness of the lessons began. This took the form of pre and post-tests based on the original objectives to test how well the learners had learned the topic. These pre and post-tests were composed of multiple choice type questions which were very similar to questions asked in the problem solving section of the CBL lessons. The intention was to prepare the learners for the tests by providing some examples of the type of questions they would encounter and to give them practice at answering such questions. This would allow the learners to focus on answering the test questions without worrying about how to answer the questions i.e. what to answer and not how to answer.

Results from these evaluation exercises showed that the majority of learners' scores improved from the pre-test to the post-test but perhaps more revealing was the increase in the confidence expressed by the learners that their answers were correct in the post-test. This was a marked increase indicating that learners felt they, now, had a better knowledge of the subject than in the pre-test. Overall, the results from the evaluations showed that the CBL material produced using the UDRIPS model was usable and could be used by learners to increase their knowledge in a particular topic and also increase their confidence in that knowledge.

11.8 Limitations of the Current Research

11.8.1 Product/Process

Evaluation of the CBL material produced using the UDRIPS model shows that there is extensive support for this type of learning experience. However, the evaluations that were undertaken examined the products developed using UDRIPS rather than the UDRIPS model itself. It evaluated the product but not the process. Subsequently, investigation has begun to rectify this. In order to look at the benefits of applying UDRIPS to CBL development, an experiment has been conducted to examine how developers, given UDRIPS, apply the model to their particular subject. Early results from this indicate that developers, in this case novice developers, find the model both useful and intuitive in structuring their own CBL lessons. This in itself has also been a limitation, further investigation of the usefulness of UDRIPS is needed to confirm its applicability to individuals and groups of developers with diverse experience of CBL development.

Novice developers often find new or innovative approaches to systems development difficult to grasp and "require procedural details of how to apply them" (Vessey & Conger, 1994). To date, novice developers have found the UDRIPS model intuitive to apply yet flexible enough to accommodate varying styles of development and various media elements. Vessey & Conger report that whilst novice developers experience difficulty in adopting certain approaches to systems development, more experienced developers may "fare even more poorly" since they are already comfortable with an existing approach. Thus, UDRIPS needs to be provided to experienced developers and its application monitored.

11.8.2 Domain

All three subject areas which have been addressed at Glamorgan have featured topics which are inherently procedural in nature. The three areas cover topics which teach how to design software, in particular, Information Systems. In order to build these types of systems the methods have clearly defined steps and structures which aid the software developer to build well structured, easily maintainable systems. In these areas, UDRIPS has been found to provide a useful tool for building a teaching/learning system both from the developers and the learners point of view. This domain is limited, however, and UDRIPS application in other domains, both procedural and conceptual, both in Computing topic areas and outside the field of Computing, needs investigation. Only through this research can the extent of the usefulness of the adoption of UDRIPS within a CBL project be assessed. This is a wide-ranging undertaking which may take many years to complete.

11.9 Future Work

In order to provide more tangible results of the benefits of adopting UDRIPS a number of further evaluations are planned. To date, much of the investigation into the adoption of UDRIPS has resulted in qualitative rather than quantitative feedback. UDRIPS has been used within the W.I.S.D.E.N. project and within student projects at Glamorgan.

This research has, to date, uncovered a number of interesting findings, future work will focus more precisely on how provision of a structured CBL development model impacts on CBL development. This will include examination of the impact of the introduction of a such a model on length of time taken to develop CBL material, cost of development (linked to time) and quality of the products produced. Time taken to develop CBL material appears to be positively influenced by the introduction of a structured model to

underpin the process. Evidence from W.I.S.D.E.N. appears to show a positive effect on development time with the pace increasing after the adoption of UDRIPS. However, this must be more formally addressed to provide more conclusive evidence.

Additional research will also focus on the extent to which UDRIPS is used as it stands or is amended or altered to suit particular situations, needs and requirements. Obviously, care must be taken to minimise the impact of adopting a novel approach in any situation until the result of its application can be forecast with some authority. Factors outside the remit of the research may emerge which have serious consequences for the participants. For example the result of providing UDRIPS in particular student projects and not providing it in others is difficult to predict and may have a serious impact on the assessment of those projects. Similarly, in an industrial project it would be very difficult to apply UDRIPS to certain forms of development and not to others, especially in a project where there may be some necessity to provide materials that are coherent or components of a larger project. To counteract this, it may be possible to provide UDRIPS to a whole group of developers in one project and to allow ad-hoc development in another. Through observation, interviews and questionnaires together with more quantitative data such as time taken to produce a design, for example, it may be possible to determine more conclusively the efficacy of UDRIPS. This investigation requires industrial contacts and willing participants and is a more major undertaking than the academic research conducted to date.

11.9.1 Further Academic Evaluations

Further academic work is planned to perfect the techniques used to deduce the usefulness of UDRIPS. This work will investigate the use of a CBL model within student groups but is intended to continue over a number of years. The impact of UDRIPS on time taken to achieve a design and also the extent to which UDRIPS was amended, altered or used as it stands will be examined. This will be conducted with groups that are asked to produce CBL material in the same subject areas but who develop the material over a period of years. Initially, no model will be recommended for the whole group and in later years UDRIPS will be introduced as the design model for the whole group. Results will be collected using observation of the process together with feedback from the developers themselves. The groups will be asked to reflect on the production of the CBL material to provide comments and/or criticisms on the development process and the ease or difficulty of structuring the CBL material, whether this is with or without UDRIPS. A comparison of the CBL material will also be made to elicit the underlying structure of material

produced with UDRIPS and without it. The CBL material produced over those years will, then, be used to form the basis of a number of evaluation exercises to determine the usability and learning effectiveness of both sets of systems to ascertain the benefits associated with the application of UDRIPS. The topics will be chosen to form CBL material to teach Computing topics at various levels to include, HND, Degree and Masters courses, the students studying on these courses will then be the subjects for the usability and learning effectiveness evaluation exercises in subsequent years.

11.9.2 Scale of Development

An additional problem that merits consideration for future work is the scale of development. Providing the model to any group with a limited time scale for their development may have a serious impact on findings which is not related directly to UDRIPS. For example, a group of students developing CBL material as part of the assessment for a single module are severely restricted in terms of both time and scale. A CBL prototype takes a considerable amount of time to develop and as such any material produced in 10-12 weeks, the time given in CBL assignments, is on a very small scale. The assignments are also individual activities requiring the students to act in many capacities e.g. developer, graphic designer, evaluator etc. However, adopting UDRIPS in these types of situations may have considerable impact on the limitations of the undertaking. UDRIPS outlines a clear, coherent structure for CBL lessons which provides the developer with guidelines for the content. Experience so far with novice developers producing CBL material, has shown that they find the structuring of CBL lessons to be a very difficult task requiring considerable research. The adoption of UDRIPS can underpin the development process and should positively affect the length of time it takes to produce CBL material. If this happens, the developer should be able to devote more time to evaluation and refinement which in turn affects quality and benefits the learner.

11.9.3 Domains

In future, work will also need to be undertaken to compare the development of CBL systems which teach procedural topics with those that are non-procedural or more conceptual in nature. Initially, however, it is intended that CBL material will be produced in a variety of Computing topics. This will test the usefulness of UDRIPS within this particular domain prior to testing in domains outside Computing such as Business or Maths, for example. Firstly, it is intended to investigate the application of UDRIPS more

formally within the areas covered by the W.I.S.D.E.N. project. Application of UDRIPS to CBL to teach Object Orientation and Real-time Systems Development has apparently proven beneficial, this needs to be verified more formally to confirm these findings. Subsequent evaluations are planned which address areas outside the field of Computing. To achieve this partners will be sought, in the first place from other departments within Glamorgan. This will, eventually, be extended to partners from other universities and/or industry.

11.10 Conclusion

The areas of software engineering and pedagogical theory were investigated and principles from both were identified and combined to produce the UDRIPS design and development model. This model was used to produce CBL material in a variety of topics by a number of developers. Evaluation feedback indicates that the material produced had a positive effect on the learners who used it. At Glamorgan, the author adopted UDRIPS to develop CBL lessons in Entity-Relationship Modelling. These CBL lessons were then used as an additional resource to teach the subject to groups of Masters students studying the area of Structured Methods. Both usability evaluations and learning effectiveness evaluations were conducted on the CBL material to deduce how students used it and whether it could facilitate learning. Results from these evaluations enabled the refinement of the material and yielded very positive results with respect to its effectiveness and use.

On a larger scale, adoption of the UDRIPS model by developers in the W.I.S.D.E.N. consortium enabled CBL material to be produced which had a commonality of approach, as per the original aim, but which demonstrated that material need not be constrained in its look and feel to meet its objective of enhancing teaching/learning. This flexibility meant that the developers were more positive about the adoption of a design and development model since it provided a valuable structure without dominating the development process. Informal evaluation of the adoption of UDRIPS showed that developers were very positive about its structure and usefulness and that it had provided a practicable framework for CBL development which facilitated communication, both between developers and between developers and the subject experts.

Subsequently, at Glamorgan, UDRIPS has been used to develop material within two more topic areas, Normalisation and Entity Life Histories, with the CBL lessons ready to be evaluated with learners in the same way as that undertaken with the E-R Modelling CBL material. Work is currently underway with the usability evaluation, and learning

effectiveness evaluation. Pre and post-tests are also under construction. Early evidence from these additional development activities appears to confirm that the adoption of UDRIPS has a beneficial impact on the time taken to produce CBL material. This needs to be compared to the findings from further work to prove this hypothesis.

UDRIPS has been used to produce CBL material which is effective in the facilitation of learning and which may be used as a learning tool and as a revision tool. Whilst these evaluations provide positive feedback further work is planned to consolidate these findings to determine the effectiveness of the UDRIPS model itself and its impact on development. The aim is to strengthen and confirm the findings from the W.I.S.D.E.N. project, and the evaluation results, of the benefit that UDRIPS affords to provide a technique which, not only aids the developer, but also aids the learner.

References

Alessi, S. M., Trollip, S. R., Computer Based Instruction: Methods and Development, 2nd ed., Prentice Hall, Englewood Cliffs, NJ., (1991).

Anderson, J. R., Cognitive Psychology and Its Implications, Fourth Edition, W.H. Freeman and Company, New York, (1995).

ARL Collaboratory, Applied Research Laboratory, Pennsylvania State University, USA, http://quark.arl.psu.edu/training/tr-menu.html (2000).

Atkinson, R. L., Atkinson, R. C., Smith, E. E., Bem, D. J., Introduction to Psychology, Eleventh Edition, Harcourt, Brace, Jovanovich, Fort Worth, TX., London, (1993).

Ausubel, D. P., Novak, J. D., Hanesian, H., Educational psychology, a cognitive view, 2nd ed., Holt, Rinehart and Winston, New York, (1978).

Barker, P. & King, T., Evaluating Interactive Multimedia Courseware - A Methodology, Computers Educ., Vol. 21, No. 4, pp 307-319, (1993).

Barker, P., Richards, S., Banerji, A., Intelligent Approaches to Performance Support, ALT-J, Vol. 2, No. 1, pp 63 – 69, (1994).

Barr, A. & Feigenbaum, E. A., The Handbook of Artificial Intelligence, Volume II, Addison-Wesley, Reading, Massachusetts, (1982).

Bauer, F., NATO Science Committee Conference, (1969) (quoted by R.S. Pressman, "Software Engineering: A Practitioner's Approach," McGraw-Hill, New York, 1997).

Beard, R. M., Healey, F. G., Hollo, P. J., Objectives in Higher Education, 2nd ed., Society for Research into Higher Education, London, (1974).

Bentley, C., Introducing PRINCE: The Structured Project Management Method, NCC Blackwell, Oxford, (1992).

Beynon-Davies, P., Analysing IS Failures; A Practical Approach, Technical Communications Ltd., Ptiman/THES Publications, London, (1996).

Beynon-Davies, P., Information Systems Development: an Introduction to Information Systems Engineering, Macmillan, London, (1998).

Birchall, D., & Smith, M., Applying Groupware in Management Education including Distance Learning, Active Learning, No. 5, pp 48 – 52, (1996).

Black, J.B. and McClintock, R.O., An Interpretation Construction Approach to Constructivist Design, In B. Wilson (Ed.) Constructivist learning environments, Educational Technology Publications, Englewood Cliffs, NJ (1995).

Bloom, B. S., Taxonomy of Educational Objectives: The Classification of Educational Goals: Handbook 1: Cognitive Domain, Longmans, New York, (1956).

Boehm, B.W., Software Engineering, IEEE Transactions on Computers, 25(12), (1976).

Booch, G., Object Oriented Design, with Applications, Benjamin/Cummings, California, (1991).

Boyle, T., Margetts, S., The CORE Guided Approach to Acquiring Programming Skills, Computers Educ., Vol. 18, No. 1-3, pp127 - 133, (1992).

Boyle, T., Gray, J., Wendl, B. & Davies, M., Taking the Plunge with CLEM: The Design and Evaluation of a Large Scale CAL System, Computers Educ., Vol. 22, No. 1/2, pp. 19-26, (1994).

Boyle, T., Design for Multimedia Learning, Prentice Hall, London, (1997).

Braxton, S., Instructional Design Methodologies And Techniques, George Washington University, http://tangle.seas.gwu.edu/~sbraxton/ISD/isd_homepage.html (1995).

Briggs, L. J., Instructional Design: Principles and Applications, 2nd Edition, Englewood Cliffs, NJ: Educational Technology Publications, (1991).

Britton, C. & Doake, J., Software System Development A Gentle Introduction, McGraw-Hill, London, (1993).

Bruner, J.S. Toward a Theory of Instruction, Belknap Press of Harvard University Press, Cambridge, Mass, (1966).

Buzan, A. & Buzan B., The Mind Map Book, BBC Books, London, (1993).

Capell, P., Report on Distance Learning Technologies, Software Engineering Institute, Carnegie Mellon University, (1995).

Clarke, J. A., Cognitive Style and CAL: What Can You Do About It?, Proceedings of the Seventh Annual Conference of the Australian Society for Computers in Learning in tertiary Education, (1989).

Coad, P. & Yourdon, E., Object Oriented Analysis, Prentice-Hall, London, (1991).

Cornford, T. & Smithson, S., Project Research in Information Systems, Macmillan Press Ltd., Basingstoke, (1996).

Coopers & Lybrand, University of London's Institute of Education & Tavistock Institute, Evaluation of the Teaching and Learning Technology Programme, HEFCE Ref. M 21/96 (1996). http://www.niss.ac.uk/education/hefce/pub96/m21_96.html

Coopers & Lybrand (b), University of London's Institute of Education & Tavistock Institute, Evaluation of the Teaching and Learning Technology Programme, Executive Summary, Active Learning No. 5, pp 60-63, (1996).

Davies, P., Brailsford, T., New Frontiers of Learning, Guidelines for Multimedia Developers, Volume 1, ITTI (Information Technology Training Initiative), (1994).

Davies, M. L. & Crowther, D. E. A. The Benefits of Using Multimedia in Higher Education: Myths and Realities, Active Learning 3, (1995).

Dick, W., & Carey, L., The Systematic Design of Instruction, Fourth Edition, Harper Collins Publisher, New York, (1996).

DSDM Consortium, Dynamic Systems Development Method, Version 2, Tesseract Publishing, (1995).

Duffy, T. M., and Jonassen, D. H. editors, Constructivism and the Technology of Instruction: A Conversation, Lawrence Erlbaum Associates Publishers, Hillsdale, NJ, (1992).

Eberts, R. E., User Interface Design, Prentice-Hall, Englewood Cliffs, N.J., (1994).

English, T., An Evaluation Toolbox, Avon TVEE Evaluation, School of Education, University of Bath, (1991).

Evaluation of Learning Technology in Higher Education (Elthe) Workshops, (1995 & 1996).

Farance, F. & Tonkel, J., Learning Technology Systems Architecture (LTSA) Specification, IEEE 1484 Learning Technology Standards Committee (LTSC), (1998).

Gagné, R. M., Briggs, L. J. & Wager, W.W., Principles of Instructional Design, Harcourt, Brace, Jovanovich Publishers, Fort Worth, TX., (1992).

Gardner-Medwin, A.R., Confidence Assessment in the Teaching of Basic Science, ALT-J, Vol. 3, No. 1, pp 80-85, (1995).

Gentleman, W. M., Definitions of Software Engineering, Software Engineering Group, Institute for Information Technology, National Research Council, Canada, (1990). http://wwwsel.iit.nrc.ca/sedefn/SEdefn.html

Gerlach, V.S., and Ely, D. P., Teaching and Media A Systematic Approach, 2nd Edition, Prentice-Hall, Englewood Cliffs, NJ, (1980).

Gibbs, G., Habeshaw, S. and Habeshaw, T., 53 interesting ways to assess your students, Technical and Educational Services, Bristol, (1988).

Goodland, M. & Slater, C., SSADM Version 4: A Practical Approach, McGraw Hill, London, (1995).

Hannafin, M. J., & Peck, K.L., The Design, Development and Evaluation of Instructional Software, Macmillan Publishing Company, NY, (1988).

Herbart, J. F., in Bigge, M.L., Learning Theories for Teachers, Sixth Edition, Longman, NY, (1982).

Hobbs, P., Managing Educational Software Development, An ALT Training Workshop, Bristol, (1995).

Honey, P., Mumford, A., The Manual of Learning Styles, P. Honey, Maidenhead, (1992).

Hooper, R., An Introduction to the National Development Programme in Computer Assisted Learning, British Journal of Educational Technology, Vol. 8, No. 3, (1977).

Ince, D. and Andrews, D., The Software Life Cycle, Butterworths, London, (1990).

Institute of Computer Based Learning, Queens University of Belfast, Understanding Computer Based Learning, ITTI, (1995).

Jackson, M. A., System Development, Prentice-Hall, Englewood Cliffs, NJ, (1983).

Jonassen, D.H. Mayes, J.T. and McAleese, R., A Manifesto for a Constructivist Approach to Technology in Higher Education, in T. Duffy, D. Jonassen, & J. Lowyck (Eds), Designing constructivist learning environments, Springer-Verlag, Heidelberg, (1994).

Kemp, J.E., Morrison, G.R., and Ross, S.M., Designing Effective Instruction, Second Edition, Prentice Hall, Englewood Cliffs, NJ, (1998).

Kent, P., Ramsden, P. & James, M., Mathematical Microworlds for Undergraduates: Mathematica-based learning environments, The SAC (Symbolic and Algebraic Computation) Newsletter, Number 2 (1997).

Knirk, F.G., and Gustafson, K.L., Instructional Technology A Systematic Approach to Education, Holt, Rinehart, and Wilson, New York, (1986).

Kolb, D., Experiential Learning, Prentice Hall, Englewood Cliffs, NJ, (1984).

Laurel, B.(ed), Mountford, S. J., The Art of Human-Computer Interface Design, Addison Wesley, Reading, Mass., (1990).

Laurillard, D., Rethinking University Education, Routledge, London, (1995).

Macnab, D. S. & Cummine, J. A., Teaching Mathematics 11-16: A Difficulty-Centred Approach, Blackwell, Oxford, (1986).

Mager, R., Preparing Instructional Objectives, Belmont, California, (1984).

Martin, J., Rapid Application Development, Macmillan Pub. Co., New York, (1991).

Mayes, J.T. Cognitive Tools: a suitable case for learning. In P. Kommers, D. Jonassen & J.T Mayes (Eds) Cognitive Tools for Learning, Springer-Verlag, Berlin, (1992).

McNiff, J., Lomax, P. & Whitehead, J., You and Your Action Research Project, Routledge, London, (1996).

Merrill, M.D., Learner control in computer based learning, Computers and Education, 4, 77-95, (1980).

Merrill, M.D., Li, Z., & Jones, M. K., Instructional Transaction Theory, An Introduction, Educational Technology, 31(6), 7-12, (1991).

Miller, P., Courseworks: an extended concept mapping tool, Active Learning, No. 3, pp 50 – 53, (1995).

Norcliffe, A., The TLTP W.I.S.D.E.N. Project - A Progress Report, 4th Annual Conference on the Teaching of Computing, (1996).

Oliver, R., & Grant, M., Distance Education Technologies, A review of Instructional Technologies for Distance Education and Open Learning, Edith Cowan University, Perth, Australia, (1996).

O'Shea, T., & Self, J., Learning and teaching with computers: artificial intelligence in education, Harvester, Brighton, (1983).

Oxford Encyclopaedic English Dictionary, Clarendon Press, Oxford, (1991).

Pagliaro, L., The History and Development of CAI: 1926-1981, An Overview, The Alberta Journal of Educational Research, Vol. XXIX, No. 1, (1983).

Pask, G., Styles and Strategies of Learning, British Journal of Psycholgy, Vol. 46, (1976).

Patton-Bennington E., A Glossary of Terms from Instructional Technology & Distance Education, {http://www.ind.net/IPSE/disted.html}, (1997).

153

Pavlov, I. P., Conditioned Reflexes, Oxford University Press, Oxford, (1927).

Pressey, S. L., A Simple Apparatus which gives Tests and Scores - and Teaches, School and Society, 23, (1926).

Pressman, R. S., Software Engineering: A Practitioner's Approach, Fourth Edition, McGraw Hill, London, (1997).

Race, P., The Open Learning Handbook, Second Edition, Kogan Page Ltd., London, (1994).

Reigeluth, C. M. (Ed.), Instructional-Design Theories and Models: A New Paradigm of Instructional Theory, Lawrence Erlbaum Associates, Hillsdale, NJ, (1999).

Rettig, M., Interface Design When You Don't Know How, (1992), Communications of the ACM, Vol 35, No. 1, pp 29-34, (1992).

Rieber, L. P., Computers, Graphics & Learning, Brown & Benchmark, Madison, Wisconsin, (1994).

Rist, R., Institute of Computer Based Learning, Heriott-Watt University, {http://www.icbl.hw.ac.uk/ctl/patrick/rslides1/roger004.html-roger011.html}, (1996).

Rist, R., & Hewer, S., What is Learning Technology? Some Definitions, Learning Technology Dissemination Initiative (LTDI), ISBN: 0 9528731 0 9, (1996).

Ross, T. W., Bloom and Hypertext: Parallel Taxonomies? Ed-Tech Review, pp 11-16, (1993).

Rumbaugh, J., Jacobson, I. and Booch, G., The Unified Modelling Language Reference Manual, Addison Wesley, Reading, Mass., (1999).

Schank, R.C., Active Learning Through Multimedia, IEEE Multimedia, (1994).

Schön, D. A., Educating the Reflective Practitioner, Jossey-Bass Inc., San Francisco, (1987).

Shneiderman, B., Designing the User Interface: Strategies for Effective Human-Computer Interaction, third edition, Addison-Wesley Publishers, Reading, Mass., (1997).

Siviter, D. & Brown, K., Hypercourseware, Computers Educ., Vol 18, No. 1-3, pp 163-170, (1992).

Skinner, B. F., The Behaviour of Organisms, Appleton-Century-Crofts, New York, (1938).

Skinner, B. F., Teaching Machines, Science, 128, pp 969-977, (1958).

Skinner, B. F., Why We Need Teaching Machines, Harvard Educational Review, 31, (1961).

Skinner, B. F., The Technology of Teaching, Prentice-Hall, Englewood Cliffs, NJ, (1968).

Smith, P. L. and Ragan, T. J., Instructional Design, Second Edition, Prentice Hall, Englewood Cliffs, NJ, (1999).

Sommerville, I., Software engineering, 3rd ed., Addison Wesley, Wokingham, England, Reading, Mass., (1989).

Stubbs, G., Watkins, M., Davies, R. A., & Berrow, T. C., UDRIP, A model for CAL Specification and Design, Ed-Media '95 - Graz, Austria, (1995a).

Stubbs, G., Watkins, M., Davies, R. A., & Jones, T. C., Instructional Design Issues Addressed, AAIM, Asheville, North Carolina, USA, (1995b).

Stubbs, G. & Watkins, M., CBL Evaluation: The Why's, What's & How's, Ed-Media '96, Boston, USA, (1996a).

Stubbs, G. & Watkins, M., Evaluation of Computer Based Learning Material, 5th Annual AAIM Conference, Multimedia In Education & Industry, Charleston, South Carolina, USA, (1996b).

Stubbs, G. & Watkins, M., Re-Engineering CBL Development, IEEE FIE '96, 1996 Frontiers in Education Conference, Salt Lake City, Utah, USA, (1996c).

Stubbs, G. & Watkins, M., Developing CBL in W.I.S.D.E.N.: The Use of a Teaching and Learning Model, 5th Annual Conference on the Teaching of Computing, Dublin City University, (1997).

Suppes, P., Impact of Computers on Curriculum in the Schools and Universities, R. Taylor (Ed.), The computer in school: Tutor, tool, tutee (pp. 236-247), Teachers College Press New York, (1980).

Tessmer, M. and Richey, R.C., "The Role of Context in Learning and Instructional Design." Educational Technology Research and Development 45, no. 2, 85-115 (1997).

Thorndike, E., Educational Psychology, Teachers College Press, New York (1913).

Travers, J. F., Elliott, S. N. & Kratochwill, T. R., Educational Psychology, Effective Teaching, Effective Learning, Brown & Benchmark, Madison, Wisconsin, (1993).

Tripp, S.D., and Bichelmeyer, B., Rapid Prototyping: An Alternative Instructional Design Strategy, Educational Technology Research and Development, Vol. 38, Part 1, pp. 31-44, (1990).

Vessey, I. & Conger, S. A., Requirements Specification: Learning Object, Process and Data Methodologies, Communications of the ACM, Vol. 37, No. 5, (1994).

Ward, P., and Mellor, S.J., Structured development for real time systems.; Implementation modeling techniques, Yourdon Press, New York, (1986).

Wilson, B. G., Jonassen, D. H., & Cole, P., Cognitive approaches to instructional design. In G. M. Piskurich (Ed.), The ASTD handbook of instructional technology (pp. 21.1-21.22), McGraw-Hill, New York, (1993).

Winston, P. H., Artificial Intelligence, Third Edition, Addison Wesley, Reading, Mass., (1992).

Wood, B., Pethia, R., Gold, L.R. & Firth, R., A guide to the Assessment of Software Development Methods, Technical Report CMU/SEI-88-TR-8, Carnegie Mellon Software Engineering Institute, (1988).

Woolley, D. R., PLATO, The Emergence of On-line Community, Computer-Mediated Communication Magazine, Vol. 1, No. 3, (1994).

Yourdon, E., Modern Structured Analysis, Prentice-Hall, Englewood Cliffs, NJ, (1989).

Yourdon, E. & Constantine, L., Structured Design, Prentice-Hall, Englewood Cliffs, NJ, (1979).

Project Addresses:

Kent, P. and Ramsden, P., The METRIC Project, Mathematics Department, Imperial College, 180 Queen's Gate, London, SW7 2BZ.

URLs

ALT:
http://www.warwick.ac.uk/alt-E/

Altavista:
http://www.altavista.com/

CBL Unit, Leeds:
http://www.cbl.leeds.ac.uk/~www/about/default.html

EdMedia Conference:
http://www.aace.org/conf/edmedia/

IEEE Learning Technology Standards Committee:
http://grouper.ieee.org/p1484/

Instructional Design Methodologies And Techniques:
http://www.seas.gwu.edu/student/sbraxton/ISD/isd_homepage.html

Macromedia:
http://www.macromedia.com/

TLTP Projects:
http://www.ncteam.ac.uk/projects/tltp/index.htm

TLTP Phase 3:
http://www.niss.ac.uk/education/hefce/pub98/98_20.html

APPENDIX A

Usability Questionnaire

User Evaluation Questionnaire.

Name -

M / F (circle which is appropriate)

Do you work in an environment that entails working with computers? Y / N

Time taken to complete courseware -

On the scale shown below, please circle the number that you think is most relevant for each of the following statements:

strongly agree	agree	undecided	disagree	strongly disagree

Navigation

1. It was obvious what each button on the screen was for.

    1   2   3   4   5

2. It was easy to become lost within the courseware.

    1   2   3   4   5

3. I could choose routes through the courseware that were relevant to me.

    1   2   3   4   5

4. I was always certain how to move through the courseware.

    1   2   3   4   5

Screen Layout

5. The screens within the courseware were attractive.

    1   2   3   4   5

6. If there was a graphic, the size was OK.

    1   2   3   4   5

7. Everything was always clearly laid out.

    1   2   3   4   5

strongly agree	agree	undecided	disagree	strongly disagree

Learning

8. I would prefer to use text books and lecture notes for learning rather than this courseware.

    1   2   3   4   5

9. The courseware would provide a useful supplement to text books and lectures.

    1   2   3   4   5

10. The courseware would provide a useful revision tool.

    1   2   3   4   5

11. I would use this courseware again.

    1   2   3   4   5

12. Working through the courseware bored me.

    1   2   3   4   5

13. This courseware was a useful learning tool.

    1   2   3   4   5

14. The method I most preferred for making choices was:
    drag & drop;

    1   2   3   4   5

    clicking a relevant area or object;

    1   2   3   4   5

    text entry.

    1   2   3   4   5

15. Prior to this activity, I had a great deal of knowledge of Data Modelling.

    1   2   3   4   5

16. After completing this activity, I have acquired a great deal of knowledge of Data Modelling.

    1   2   3   4   5

strongly agree	agree	undecided	disagree	strongly disagree

Help

17. The help facility was adequate and always available.

    1   2   3   4   5

18. If I made a mistake the feedback told me why and how to move on.

    1   2   3   4   5

Level

19. Using this courseware was like learning from:
    a book;

    1   2   3   4   5

    a tutor;

    1   2   3   4   5

20. The courseware was too easy.

    1   2   3   4   5

Overall Impressions

21. The use of the mouse and keyboard was intuitive within the courseware.

    1   2   3   4   5

22. This courseware was confusing because of:
    the language;

    1   2   3   4   5

    the examples.

    1   2   3   4   5

23. The courseware had elements of:
    racism;

    1   2   3   4   5

    sexism.

    1   2   3   4   5

strongly agree	agree	undecided	disagree	strongly disagree

24. The courseware would have been more enjoyable and interesting if it had included:

sound;

1	2	3	4	5

video.

1	2	3	4	5

Please list anything else you think would have made the courseware more enjoyable and interesting.

Please answer the following questions in your own words:
25. what were the most attractive features of the courseware.

26. what were the most unattractive features of the courseware.

APPENDIX B

Pre & Post-Tests

Instructions for Entity - Relationship Test

Pre Test

Name:

Please circle the correct answer to the test questions, mark only **ONE** choice, e.g.

How am I feeling today ?

A   fine
(B)   ecstatic
C   depressed
D   happy

At the end of each question you will see five faces

please indicate how confident you are that your answer is the correct one by marking the appropriate face, mark only **ONE** face, e.g.

this indicates that you are <u>very</u> confident.

1. A systems investigation will identify the data that comprise the system. Which of these is used to model that data?
   A   data flow diagram
   B   decision table
   C   entity relationship diagram
   D   program variables

2. Why does a systems analyst use E-R diagrams?
   A   to show links between groups of data
   B   to show the flow of data in a system
   C   to show individual items of data
   D   to show processes in the system

3. The Entity Relationship diagram is
   A   A logical model
   B   A physical model
   C   A conceptual model
   D   An organisational model

4. Data characteristics of an entity type are called?
   A   Fields
   B   Columns
   C   Attributes
   D   Properties

5. An individual occurrence of a particular entity type is called?
   A   An entity instance
   B   An entity appearance
   C   An entity class
   D   An entity record

6. An association between entity types is called?
    A   A relation
    B   A relationship
    C   A link
    D   A franchise

7. Which of the following is a valid notation for an entity?

    A | garage |

    B | GARAGE |

    C | GARAGE |

    D | GARAGES |

Questions 8 - 10 relate to the following scenario
A hockey club has about 20 players who take part in a number of matches throughout the season.

8. Which is the degree of the relationship 'plays in' between PLAYER and MATCH?

	Player		Match
A	1	:	1
B	1	:	N
C	M	:	N
D	N	:	1

9. In each match one player will be captain.
    Which is the degree of relationship 'captains' between PLAYER and MATCH?

	Player		Match
A	1	:	1
B	1	:	N
C	M	:	N
D	N	:	1

10. Which correct membership for the captains' relationship?
    A   PLAYER is optional, MATCH is mandatory
    B   PLAYER is optional, MATCH is optional
    C   PLAYER is mandatory, MATCH is mandatory
   D   PLAYER is mandatory, MATCH is optional

    'A CD-TITLE is a reference to a music CD, at least one copy of which has been acquired by the library for loan to its members'.

11. What other information does the definition provide?
    A   The library stocks every music CD produced
    B   There may be more than one copy of some CDs
    C   There is only one copy of each CD
    D   There is more than one copy of all CDs

12. An initial E-R diagram is shown, however it has been found that several invoices are often associated with one order.

    ORDER ——————— INVOICE
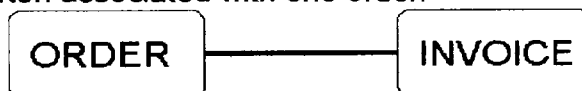
    Which change is required to correct the E-R diagram?
    A   rename the INVOICE entity as INVOICES
    B   introduce a new entity BILL
    C   introduce a second relationship
    D   change the degree of the relationship

13. An initial E-R diagram is shown, however it has been found that for some book titles the   library has no book copies at all.

    BOOK-TITLE ——————< BOOK-COPY

    Which change is required to correct the E-R diagram?
    A   change the many end of the relationship to one
    B   change the one end of the relationship to optional
    C   change both ends of the relationship to optional
    D   do nothing

Questions 14 -15 relate to the following scenario.

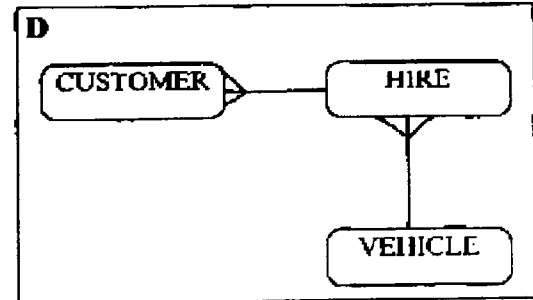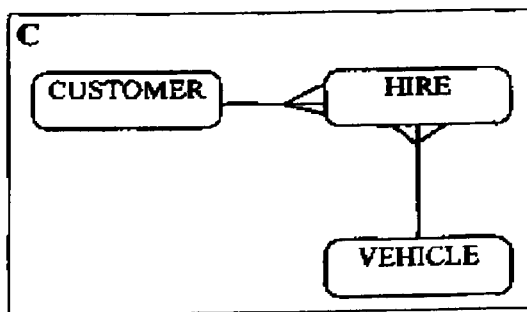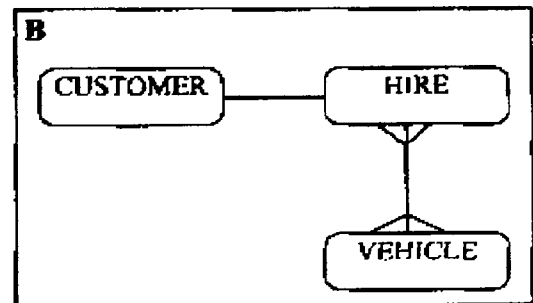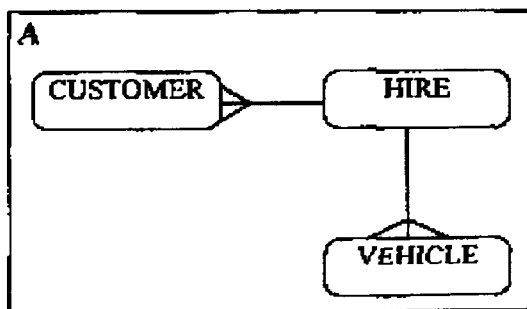A new entity *hire* has to be developed in the E-R diagram shown, it also has attributes of *date* and *day's-out.*

CUSTOMER >―――< VEHICLE

14. Which set of tables matches the new situation?

A customer(**account no**, name, date)
   hire(**reg no, account no**)
   vehicle(**reg no**, model, day rate,
                 days out)

B customer(**account no**, name)
   hire(**reg no, account no**)
   vehicle(**reg no**, model, day rate,
                 date,days out)

C customer(**account no**, name)
   hire(**date**, days out)
   vehicle(**reg no**, model, day rate)

D customer(**account no**, name, date)
   hire(**reg no, account no**, date, days out)
   vehicle(**reg no**, model, day rate)

15. Which is the correct E-R Diagram

A
CUSTOMER >― HIRE
HIRE ―< VEHICLE

B
CUSTOMER ― HIRE
HIRE ―< VEHICLE

C
CUSTOMER ―< HIRE
HIRE ―< VEHICLE

D
CUSTOMER >― HIRE
HIRE ― VEHICLE

16. For the following relationship definition, choose the correct degree classification. Each salesperson is allocated a company car, a company car is assigned to one salesperson:-

	S'Person		car
A	1	:	N
B	N	:	1
C	1	:	1
D	M	:	N

17. For the following relationship definition, choose the correct degree classification. Each department has at least one employee, an employee works in one department:-

|	Dept		Employee
A	1	:	N
B	1	:	1
C	M	:	N
D	N	:	1

18. For the following relationship definition, choose the correct degree classification. Each project involves one or more employees, an employee may work on more than one project: Project

	Employee		
A	1	:	N
B	M	:	N
C	N	:	1
D	1	:	1

19. Which of the following is the correct representation of the following relationship definition: A client may place one or more contracts, a contract must be made for one client.

A CLIENT ———<CONTRACT

B CLIENT >——— CONTRACT

C CLIENT ——— CONTRACT

D CONTRACT >········· CLIENT
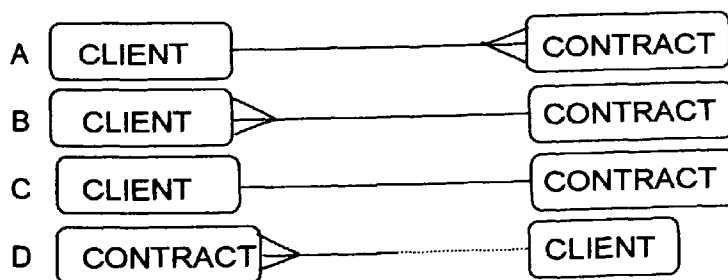
The entity BUS will have many attributes. Some of these are: type of bus; number of seats; registration number; date of purchase; purchase price; supplier code.

20. Which of these would you choose as the primary key?
  A type of bus
  B number of seats
  C registration number
  D date of purchase

21. Which of these is a candidate to be a foreign key?
  A date of purchase
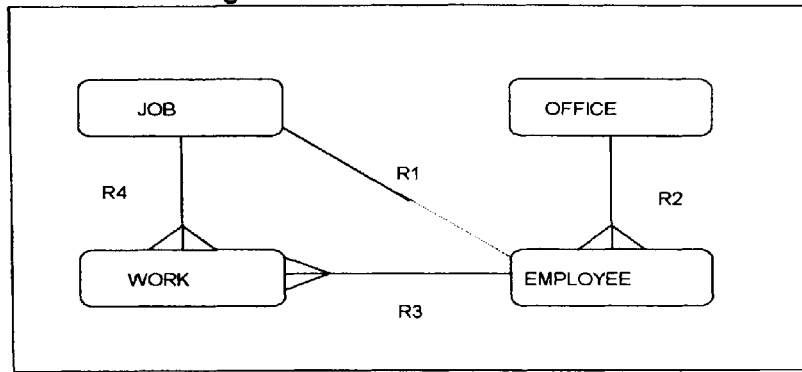 B supplier code
  C number of seats
  D type of bus

22. Meeting rooms in an IT department are available for booking by staff and students.
Booking(room no, bookee id, date, time, duration, purpose)

Which would be a primary key for Booking?
  A room no
  B room no/date/time
  C room no/date/duration
  D room no/bookee id

JOB(job no, title, location, client)
OFFICE(Code, address, tel no)
EMPLOYEE(Staff id, Name, Tel no, grade)
WORK(job no, staff id, hours)

23. Which is the correct definition of the *manages* relationship R1?
    A   All employees manage jobs and all jobs are managed
    B   Some employees manage many jobs, a job has one manager
    C   Each employee may manage one job, each job must have a
         manager
    D   Some jobs are not managed and all employees manage a job

24. Which is the correct amendment to implement R1?
    A   JOB(job no, title, location, client, staff id)
    B   EMPLOYEE(staff id, name, tel no, grade, job no)
    C   OFFICE(Code, address, tel no, staff id, job no)
    D   No change

25. which is the correct amendment to the attribute tables to establish the *works from*
     relationship R2?   JOB(job no, title, location, client)
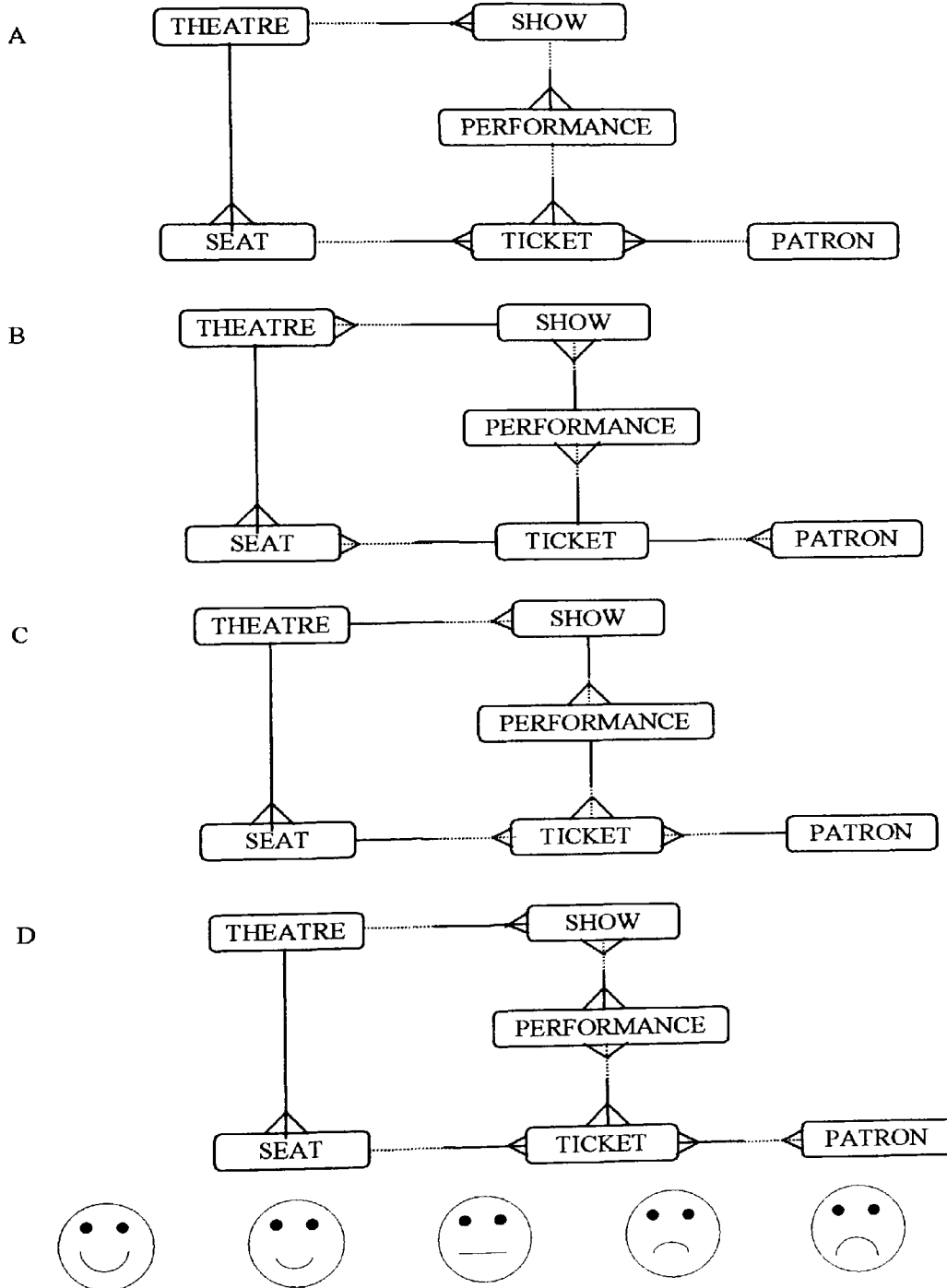   OFFICE(Code, address, tel no)
                             EMPLOYEE(Staff id, Name, Tel no, grade)

     A   OFFICE(Code, address, tel no, Staff Id)
     B   EMPLOYEE(Staff Id, Name, Tel No, grade, Office Code)
     C   OFFEMP(Office Code, Staff Id)
     D   No change

A system is required for the purpose of tracking the sale of tickets of several theatres. A theatre may have many shows but each show will only be staged at one theatre. A show may have many performances e.g. one performance can be on April 7th at 12 noon and another can be on April 7th at 8 p.m. All seats are available for all performances. Each seat is identified by a row number and a seat number. Tickets to these seats are sold to patrons. Each ticket has a unique serial number. Information regarding the price of each ticket is also kept. In order to facilitate credit-card payments, the system would require the patron's name and credit-card number to be kept.

Which of the following diagrams represents the correct E-R model for this scenario.

A

| THEATRE |······—◁ | SHOW |

| PERFORMANCE |

| SEAT |·············◁ | TICKET |▷······ | PATRON |

B

| THEATRE |▷······ | SHOW |

| PERFORMANCE |

| SEAT |▷······ | TICKET |—◁ | PATRON |

C

| THEATRE |······◁ | SHOW |

| PERFORMANCE |

| SEAT |······◁ | TICKET |▷······ | PATRON |

D

| THEATRE |······◁ | SHOW |

| PERFORMANCE |

| SEAT |······◁ | TICKET |▷······◁ | PATRON |

Tele-mail is a mail order company which sends out mail catalogues. The company receives customer orders through the mail. Customers who are interested in the products may order these by filling in the necessary order form. Together with the product descriptions, the interested customer should supply his name, address, telephone number, quantity required and mode of payment, credit-card number and expiry date (if payment is by credit). Upon receipt of the product, customers have a 21-day period to decide whether they wish to purchase the product or to return it. Non credit-card customers should send their payments to Tele-mail for the products purchased. Customers wishing to return the product(s) should mail these together with a completed return form to Tele-mail within the stipulated period. Partial returns are acceptable.

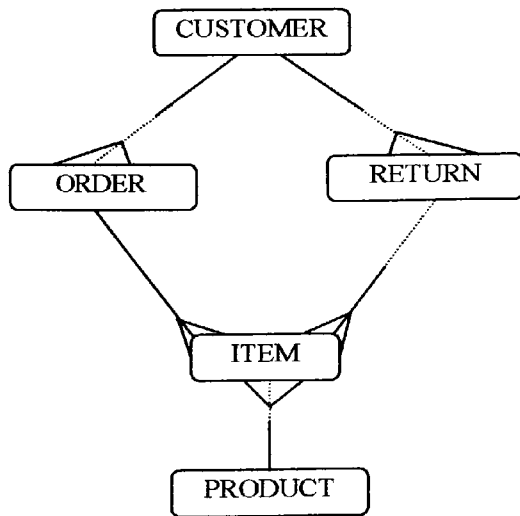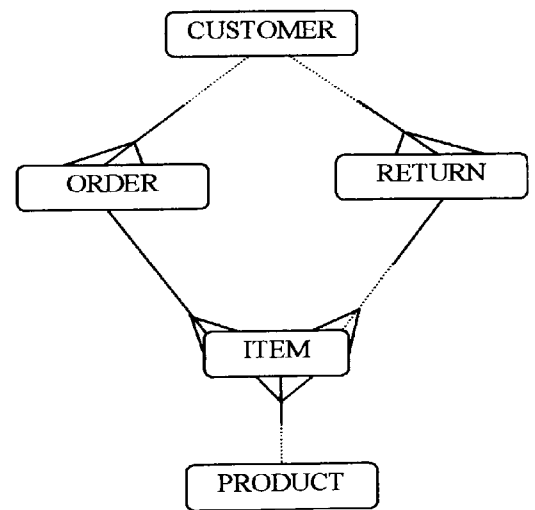Which of the following diagrams represents the correct E-R model for this scenario.

A
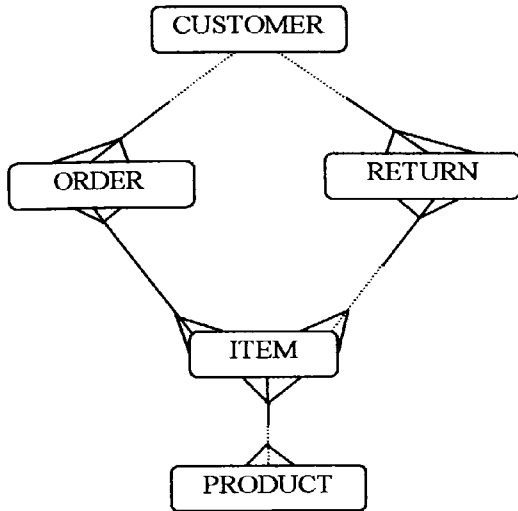
B
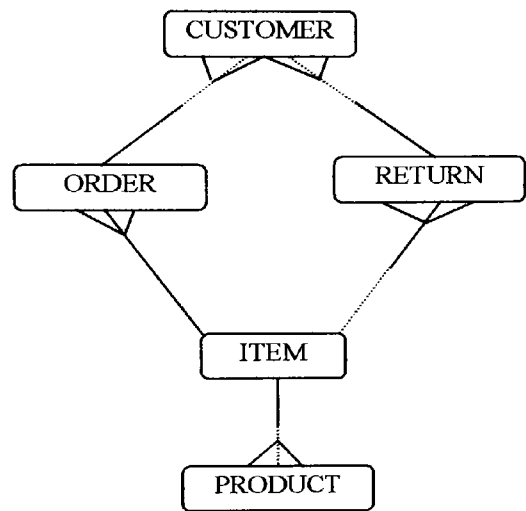
C

D

APPENDIX C

Published Papers

G. Stubbs , M. Watkins, R.A. Davies &
TC Berrow. UDRIP, A model for CAL
Specification and Design. Ed-Media '95
- Graz, Austria, (1995a).

UDRIP, A model for CAL Specification and Design.

G. Stubbs , M. Watkins & R.A. Davies,
Department of Computer Studies,

T.C. Berrow,
Business School,

University of Glamorgan, Pontypridd, Mid Glam, CF37 1DL. Telephone: (0443) 480480
GSTUBBS@COMP.GLAMORGAN.AC.UK

1. Introduction

This paper presents the UDRIP model which provides a structured methodology for the development of Computer Assisted Learning (CAL) materials. UDRIP has been developed at the University of Glamorgan, as a result of being an active member of the W.1.S.D.E.N. TLTP II consortium. The consortium is currently over one year into the development of CAL material which covers various software project analysis and design methods. The authoring languages used for the project are Toolbook and Authorware Professional, the development platforms are PC and MAC. The courseware currently under development here at Glamorgan covers the subject of structured methods, E-R modelling in particular.

2. The Learning Model

Like a software project, a poor CAL design without an understandable structure will not provide a reliable nor robust product. UDRIP is a structured generic model for CAL design and presentation based on cognitive and behavioural learning theories [Ausubel 1980, Gagné & Briggs 1979]. As with recent computing paradigms, three main benefits have been identified when using the UDRIP model:
* it offers a structured approach;
* it offers consistency and reliability;
* it is generic, in that, it can be applied to many subject areas and developments using learning materials other than CAL.

The UDRIP acronym represents the constituent parts of the model from the learner's perspective:

Universal picture	where have I been, where am I going - Pre-Requisites & Objectives;
Definitions	what don't I know - Keywords & Concepts;
Rules	what is legal, what is illegal - Application & Usage of Keywords & Concepts;
Illustrative examples	are all aspects covered - Embedded Scenarios & Solutions;
Problem solving	do I really understand - Self Assessment.

The development of UDRIP resulted from the need to amalgamate positive aspects of current learning models, and to tailor them to the specific environment of CAL. The use of the UDRIP model does not preclude the use of Human Computer Interface (HCI) standards nor the use of multi-media effects, it acts as a sort of skeletal structure which combined with the HCI, the multi-media and the subject material, form the whole CAL package.

3. References

[Ausubel 1980]. Ausubel, D., Schematic, Cognitive Structures and Advance Organisers,(1980), American Educational Research Journal, pp400 -404.
[Gagné & Briggs 1979]. Gagné, R. & Briggs, L., Principles of Instructional Design, (1979), New York, Holt, Rinehart and Winston.

G. Stubbs , M. Watkins, R.A. Davies &
T.C. Berrow. Instructional Design Issues
Addressed. AAIM, Asheville, North
Carolina, USA, (1995b).

Instructional Design Issues Addressed

G. Stubbs (1), M. Watkins (1), R.A. Davies (1) & T.C. Jones (2)
1. Department of Computer Studies, 2. Business School,
Address: University of Glamorgan, Pontypridd, Mid Glam, CF37 1DL
Telephone: (01443) 480480 ext 2263, Fax: (01443) 482715
E-Mail Address: GSTUBBS@COMP.GLAMORGAN.AC.UK

INTRODUCTION

The Wide-ranging Integrated Software Design Education Network (W.I.S.D.E.N.) is a three year project funded under the Teaching and Learning Technology Programme, phase Two (TLTP-2). The main object of the project is the development of a wide range of interactive computer-based learning materials in the area of software design, with the primary aim of making teaching and learning more productive and efficient in the mainstream areas of undergraduate computing courses. Objectives of the project are:

1. to develop and disseminate a wide range of interactive CBL material in the area of software specification and design;
2. to develop materials with a commonality of approach;
3. to critically evaluate materials produced for the purposes of establishing best practice for developing interactive CBL materials.

The development of the W.I.S.D.E.N. CBL material was distributed amongst the seven consortium universities, project management was centred at the lead university site and a commercial partner. Each consortium member took responsibility for development of CBL material in a specific topic area of software development methods, Glamorgan and South Bank universities had responsibility for the Structured Methods topics, E - R Diagramming and Data Flow Diagramming, respectively. Other consortium members covered such topics as Formal Methods, Real Time Design and Object Oriented Methods. This distributed development required a need for close collaboration between the consortium members and regular review of produced work to seek to ensure some consistency of level and style.

The development of the W.I.S.D.E.N. CBL material has needed to address several specific CBL issues, these include:

interface standards and navigation issues;
content and learning models.

INTERFACE STANDARDS AND NAVIGATION ISSUES

To concur with the objective of the project to produce material with a commonality of approach, both the interface standards and the navigation issues were researched by a working party and their recommendations adopted universally by all group members. These aspects of CBL authoring have been widely addressed, screen design [Shneiderman 87, Eberts 94, Rettig 92], navigation [Siviter & Brown 92, Ross 93].

The interface standards addressed issues such as text colour, fonts & font sizes and background colours. Navigation involved adopting a dual approach, both a sequential path and a quick access or browse facility. A template was developed and distributed to each member, this obviated the possibility of duplication of effort. The template offered:

Navigation aids (pop-up menus, help button);
Course Title, Topic Title & Lesson Title;
Forward and Backward navigation, both section and screen navigation.

Once the standards were agreed this left the individual sites free to decide on screen layout, images, animation and subject content.

Content and Learning Models

The research activity at the University of Glamorgan has focused on existing learning models and has resulted in the development of the UDRIP learning model. This development resulted from the need to amalgamate the positive aspects of current learning models, and to tailor them to the specific environment of CBL.

The rationale behind the UDRIP model stems from the expertise in Structured Methods of the developers at Glamorgan. The process of CBL development appears to mirror very well the process of software development but the purpose of CBL is quite different to the purpose of more generic software. The objective of software development is to computerise an existing system or produce a new system for which a specific need has been identified. The objective of CBL development is to produce a software package that will assist student learning, and two issues need to be addressed in the quest for quality

*   the process of producing the CBL;
*   the educational core of the final product.

As the number of CBL projects continues to grow, an important question needs to be addressed, "is CBL development analogous with software development in general" and if this proves to be the case, adopting a more structured approach to CBL development is an obvious step to take. However current software structured approaches are inadequate as far as educational needs are concerned and a more suitable method is desirable.

UDRIP is a structured generic model for CBL design and presentation based on cognitive and behavioural learning theories. The UDRIP acronym represents the constituent parts of the model from the learner's perspective:

Universal picture	where have I been, where am I going - Pre-Requisites & Objectives;
Definitions	what don't I know - Keywords & Concepts;
Rules	what is legal, what is illegal - Application & Usage of Keywords & Concepts;
Illustrative examples	are all aspects covered - Embedded Scenarios & Solutions;
Problem solving	do I really understand - Self Assessment.

The model need not be exclusively used for CBL presentations, it can, for example be used in the preparation of many visual teaching / learning support materials. It is also aimed at giving a framework for developers to produce CBL material quickly, efficiently and within a sound learning environment.

The UDRIP model does not preclude the use of Human Computer Interface (HCI) standards nor the use of multi-media effects, it acts as a sort of skeletal structure which combined with the HCI, the multi-media and the subject material, form the whole CBL package. Application of the model to CBL development within Glamorgan has seamlessly meshed with the W.I.S.D.E.N. template. Other consortium members have adopted UDRIP in their own presentations and have given positive feedback.

The collaboration of Glamorgan and South Bank has enabled evaluation by the developers of each other's CBL development strategies. The two sites had different CBL development experience. Glamorgan used Authorware while South Bank chose to build on their expertise in Toolbook. Both centres approached their development by agreeing a breakdown of the topic areas into very low level learning outcomes. The content presentations have different screen layouts that reflect the individual backgrounds, however, analysis of both sets of material against prescribed criteria has shown that they both exhibit common user involvement and activities, e.g. presentations combine passive information delivery with interactive re-inforcement and discovery. Our experiences have shown that using the UDRIP model to provide a pedagogic structure does not stifle individual creativity of the developer.

REFERENCES

[Shneiderman 87]. Shneiderman, B., Designing The User Interface. 2nd Edition, (1992), , Addison Wesley, Reading, Mass.

[Eberts 94]. Eberts, R.E., User Interface Design, (1994), Prentice-Hall Inc., New Jersey.

[Rettig 92]. Rettig, M., Interface Design When You Don't Know How, (1992), Communications of the ACM, Jan 1992, Vol 35, No. 1, pp 29-34.

[Siviter & Brown 92]. Siviter, D. & Brown, K., Hypercourseware, (1992), Computers Educ., Vol 18, No. 1-3, pp 163-170.

[Ross 93]. Ross, T.W., Bloom and Hypertext: Parallel Taxonomies? (1993). Ed-Tech Review, Autumn/Winter 1993, pp 11-16.

G. Stubbs & M. Watkins. CBL

Evaluation: The Why's, What's & How's

Ed-Media '96, Boston, USA, (1996a).

CBL Evaluation: The Why's, What's & How's.

G. Stubbs & M. Watkins,
Department of Computer Studies,
University of Glamorgan, Pontypridd, Mid Glam, CF37 1DL. Telephone: (01443) 482263
GSTUBBS@GLAMORGAN.AC.UK

Introduction

Development & usage of CBL is a commonplace activity today, whereas evaluation of the effectiveness of the CBL material as a teaching / learning medium is a key and often neglected issue. The evaluation of a CBL product involves three activities: understanding the product's teaching and learning objectives and its intended audience; gathering evidence regarding its use and effectiveness; judgement of the product based on the evidence. As experienced developers of Computer Based Learning (CBL) materials for use in undergraduate computing modules, we have, during the academic year 95/96 undertaken some trials of CBL material with groups of students that has involved evaluation of the student learning experience.

Features of Evaluation

Evaluation is required to assess the quality of the CBL product. Any CBL product has two main components: the software component as it is a computer based activity; the didactic component as its purpose is to instruct. Therefore, evaluation should address both facets if the quality of the final product is to be assured. For the CBL developer, only through evaluation will feedback be gained that will enable improvement in the quality of the CBL product. Evaluation of the CBL product involves a two pronged approach, firstly from the point of view of a content expert who is concerned with the totality and credibility of the subject matter and secondly, from the point of view of the potential student user. While we are aware of the former this paper is primarily concerned with the latter.

In evaluating a CBL product we need to elicit from the student user: the teaching / learning effectiveness of the product; the usability of the product with respect to ease, consistency & clarity of use; the usage of the product with respect to users wishing to have access to it. The approach taken has been to use: a short test paper that addresses the learning objectives derived directly from the CBL material [Laurillard, 1993]; a questionnaire that addresses navigation, usability and other HCI issues[Barker & King, 1993]; video to record student experiences with the material; student interviews to provide additional open-ended feedback [Elthe, 1995/96]. The test paper was used twice, firstly as a pre-test and secondly as a post-test and the student performances compared. The paper took the form of a number of multi-choice questions that covered the learning objectives of the CBL material, students were also required to indicate their confidence when making a particular choice[Gardner-Medwin, 1995].

Conclusions from Evaluation

In conducting the evaluation with different but typical student groups a number of problems materialised: student attitudes to CBL; availability of access to CBL material; evaluation timetable. We believe that the problems experienced are not unique to Glamorgan or the particular CBL material content and raise question about conclusions drawn from any evaluation. By being aware of the problems that may arise, the evaluation events can be planned to minimise their effect and thus provide more credible conclusions. After addressing the problems uncovered the evaluation produced some very positive initial results regarding the student learning process.

References:
[Barker & King, 1993] Barker, P. & King, T.(1993). Evaluating Interactive Multimedia Courseware - A Methodology, Computers Educ., Vol. 21, No. 4, pp 307-319.
[Laurillard, 1993] Laurillard, D. (1993). Rethinking University Teaching, Routledge.
[Gardner-Medwin, 1995] Gardner-Medwin, A.R. (1995). Confidence Assessment in the Teaching of Basic Science, ALT-J, Vol. 3, No. 1, pp 80-85.
[Elthe, 1995/96] Elthe Workshops, (1995 & 1996).

G. Stubbs & M. Watkins. Evaluation of Computer Based Learning Material, 5th Annual AAIM Conference, Multimedia In Education & Industry, Charleston, South Carolina, USA, (1996b).

Evaluation of Computer Based Learning Material

G. Stubbs & M. Watkins
Department of Computer Studies,
University of Glamorgan,
Pontypridd,
Mid Glam, CF37 1DL
Telephone: (01443) 482263,   Fax: (01443) 482715
E-Mail Address: GSTUBBS@GLAM.AC.UK

Presentation Type: Research.
Duration: 45 minutes.
Technology area: Determining Effectiveness of CBL and Multimedia, Design & Development of Professional Multimedia Presentations.
Anticipated Audience Level: All.
Hardware: Screen Projector, Multimedia IBM machine.

Introduction

The use of CBL has become commonplace today in all levels of education from primary through to higher, at home and at work, whereas development of CBL products has largely been the province of higher education or commercial organisations. This flurry of activity has been fuelled by:-

(i)   the dramatic increase in performance coupled to a similar cost reduction in microcomputer products enabling true multimedia applications, providing computing for all!
(ii)   the increased usage and availability of the WWW, but is it WWW for all?
(iii)   the government view that CBL usage in education will lead to reduced staffing requirements;
(iv)   an increased demand for distance learning facilities.

In the UK there have been a number of significant publicity "hypes" regarding computing in schools using multimedia technology and access to the Internet e.g. Liverpool Schools Initiative had widespread national coverage, locally to Glamorgan University there have been a number of Local Education Authority funded computing in Schools projects. At Glamorgan University, our Internet access is widespread, in terms of student use, across all disciplines and new students arrive with high expectations of Internet availability. This publicity hype coupled to our observation of Internet usage has led us, as CBL developers, to rigorously address the educational benefits of CBL usage. This is necessary in order to combat existing misconceptions that the principal characteristics of CBL seem to be "it's interesting, it's time consuming, keeps users quiet, like watching paint dry."

Within this context of CBL usage, evaluation of the effectiveness of the CBL material as a teaching / learning medium is a key and often neglected issue.

The department of Computer Studies at the University of Glamorgan has been actively involved, for the past three years, in development of Computer Based Learning (CBL) materials for use in undergraduate computing modules. This work, together with that of other academic partners, comprises the W.I.S.D.E.N. TLTP-2 project in the UK. During the academic year 1995/96 the development team have undertaken some trials of CBL material with groups of students that has involved evaluation of the student learning experience.

Our evaluation method for CBL products has involved three activities:-
(i) understanding and clearly specifying the product's teaching and learning objectives;
(ii) gathering evaluation evidence regarding its use and effectiveness;
(iii) judgement of the product based on all the evidence collected viz. learning objectives met, HCI and navigation issues.

The first activity is an integral part of our CBL development process and is necessary to effectively perform the other two activities that occur post CBL development and can provide feedback to the development process.

Rationale for Evaluation

Evaluation is required to assess the quality of the CBL product. Any CBL product has two main components:-
(i) the software component as it is a computer based activity;
(ii) the didactic component as its purpose is to instruct.

Therefore, evaluation should address both facets if the quality of the final product is to be assured.

In addition the process of evaluation must:-
(i) convince any external user, who has not been involved in the development of the CBL product, of its quality;
(ii) demonstrate the veracity of the content to the satisfaction of all the development team members;
(iii) provide users with the knowledge that the educational outcomes resulting from using the CBL material are comparable to those achieved by more traditional methods e.g. lecturing.

For the CBL developer, only through evaluation will feedback be gained that will enable improvement in the quality of the CBL product.

Evaluation of the CBL product involves a two pronged approach, firstly from the point of view of a content expert who is concerned with the totality and credibility of the subject matter and secondly, from the point of view of the potential student user. While we are aware of the former this paper is primarily concerned with the latter. This evaluation can be used in two ways, from the point of view of the student it can highlight any possible deficiencies in knowledge of which the student may be unaware. From the point of view of the lecturer it can, through an analysis of student performance, highlight any possible deficiencies in the CBL material of which the developer or lecturer is unaware.

In evaluating a CBL product we need to elicit from the student user:-
(i) the teaching / learning effectiveness of the product, "have I learned anything?";
(ii) the usability of the product with respect to ease, consistency & clarity of use, "can I find my way around it?";
(iii) the usage of the product with respect to users wishing to have access to it, "will I use this again and how?".

The approach taken has been to use:-
(i) a short test paper that addresses the learning objectives derived directly from the CBL material (Laurillard, 1993);
(ii) a questionnaire that addresses navigation, usability and other HCI issues(Barker & King, 1993);
(iii) video to record student experiences with the material;
(iv) student interviews to provide additional open-ended feedback.

The test paper was used twice, firstly as a pre-test and secondly as a post-test and the student performances compared. The paper took the form of a number of multi-choice questions that covered the learning objectives of the CBL material, students were also required to indicate their confidence when making a particular choice(Gardner-Medwin, 1995).

Evaluation Experience

Evaluations have been conducted using a variety of student groups. These have ranged from small size groups of mature post-graduate conversion students with limited computing experience to larger size groups of computing undergraduates. Initially, the evaluations were conducted in a very "loose" way, i.e. the students were given a pre-test, then allowed to use the CBL material in their own time and lastly given a post-test, this process was accomplished over a number of weeks. Later evaluations were conducted in a very controlled way, here, the students were given a pre-test, then observed using the CBL and finally given a post-test, this process taking place over a number of hours rather than weeks.

In conducting the evaluation with different but typical student groups a number of problems materialised, such as:-
(i)   student attitudes to CBL usage;
(ii)   availability of access to CBL material;
(iii)   evaluation timetable;

We believe that the problems experienced are not unique to Glamorgan or the particular CBL material content and raise question about conclusions drawn from any evaluation.
By being aware of the problems that may arise, the evaluation events can be planned to minimise their effect and thus provide more credible conclusions.

Our presentation will address:-
(i)   our own experiences of formally evaluating CBL material;
(ii)   suggestions for questionnaire design;
(iii)   examples of pre & post-test questions;
(iv)   some preliminary results;
(v)   guidelines that we believe are necessary for carrying out such an evaluation.

References:

Barker, P. & King, T., Evaluating Interactive Multimedia Courseware - A Methodology, Computers Educ., Vol. 21, No. 4, pp 307-319, 1993.

Laurillard, D., Rethinking University Teaching, Routledge, 1993.

Gardner-Medwin, A.R., Confidence Assessment in the Teaching of Basic Science, ALT-J, Vol. 3, No. 1, pp 80-85, 1995.

Elthe Workshops, 1995 & 1996.

G. Stubbs & M. Watkins. Re-Engineering CBL Development, IEEE FIE '96, 1996 Frontiers in Education Conference, Salt Lake City, Utah, USA, (1996c).

Re-Engineering CBL Development

Geneen Stubbs & Mike Watkins

Department of Computer Studies,
University of Glamorgan,
Pontypridd,
Mid Glamorgan,
CF37 1DL
UK
email:- gstubbs@glam.ac.uk

Abstract

This paper describes a method for the development of Computer Based Learning (CBL) applications that mirrors those available for traditional software development with an added dimension specifically devoted to educational excellence, such that it:

- *is based on learning theories and incorporates sound principles of education;*
- *embodies a learning model;*
- *offers a structured approach;*
- *offers consistency and reliability;*
- *is generic, in that, it can be applied to many subject areas and developments using learning materials other than CBL;*
- *can be taught.*

Background

The lack of a sound learning basis has been widely recognised in the production of CBL, a survey by the Computers in Teaching Initiative (CTI) of the many projects under their auspices showed that "it is certainly true that 'a theoretical base in learning and instruction is sometimes missing or not well applied in the development and implementation of computer mediated instruction'". In fact, by the Autumn of 1987, 36 of the 106 CTI projects which had then been in existence for at least a year reported difficulties in the "educational" domain [1]. However, this practise must be viewed with caution, it is known that the venture into educational theory must be carefully controlled so that it does not become the primary exercise in the CBL development process.

The idea of a CBL development model stems directly from the Structured Methods expertise already well established in the field of computing, however, the provision of a pedagogical adjunct within a CBL development method should go some way to providing a solution to the problems highlighted above. As with traditional software engineering techniques, any CBL

development method will only be of use if it is reproducible, can be quickly taught and can be applied in various diverse situations. Although there exists much advice both in the field of Educational Psychology on learning and that of CBL development on structure and content, to date, no amalgamation exists to offer the same method to the process of CBL development as Data Flow Diagramming, for example, does for Information Systems Development.

Pedagogy

Research into the field of educational learning theories [2], [3], [4], [5], [6], [7], [8] has uncovered five areas which appear, by general agreement, to be essential for learning to transpire or for learning to be more meaningful. These areas are:

- pre-requisites;
- clear learning objectives;
- illustrative examples;
- problem solving;
- reflection.

The Software Engineering Methods

Software projects over the past twenty five years have expended much effort in seeking to impose some structure to the process of software project design and development. The introduction of software engineering and its associated activities reflect a determined effort to bring software production into line with mainstream engineering by addressing such issues as standards and quality, while attempting to impose an engineering discipline to software projects. The Data Oriented Design [9], Object Oriented Design [10] and Function Oriented Design [11] paradigms are three examples of imposing structure to the design process.

Each of these methods is concerned with the important items within a system and the interaction between them.

As far as the CBL development model is concerned the identification of the "entities", "objects" or "functions" can be utilised by defining any keywords or concepts about which the user is expected to learn, the "relationships", "processes" or "interactions" are the equivalent of the procedural knowledge of the use of the concepts, i.e. the rules of application of the concepts. This gives two more components which complete the CBL development model:

- define any keywords or concepts that are to be taught in the lesson;

- outline the rules that govern the use or application of the concepts.

UDRIPS: A CBL Development Model

To summarise, research into a variety of learning theories and structured software engineering methods has identified several important factors. These are:

- pre-requisites;

- clear learning objectives;

- illustrative examples;

- problem solving;

- reflection;

- define any keywords or concepts that are to be taught in the lesson;

- outline the rules that govern the use or application of the concepts.

These factors constitute the CBL development model, UDRIPS.

The UDRIPS acronym represents the constituent parts of the model from the learner's perspective:

Universal picture	where have I been, where am 1 going? - Pre-Requisites & Learning Outcomes;
Definitions	what don't I know? - Keywords & Concepts;
Rules	what is legal, what is illegal? - Application & Usage of Keywords & Concepts;
Illustrative examples	are all aspects covered? - Embedded Scenarios & Solutions;
Problem solving	do I really understand? - Self Assessment.
Summary	what exactly did I learn? - Summary of significant elements of the lesson.

The UDRIPS model does not preclude the use of Human Computer Interface (HCI) standards nor the use of multi-media effects, it acts as a sort of skeletal structure which combined with the HCI, the multi-media and the subject material, form the whole CBL package.

Experience with UDRIPS

UDRIPS was applied to build an initial CBL prototype to teach undergraduates the basic concepts of Entity-Relationship Modelling. This prototype enabled an evaluation of:

- the method;

- the CBL issues of navigation and interface standards;

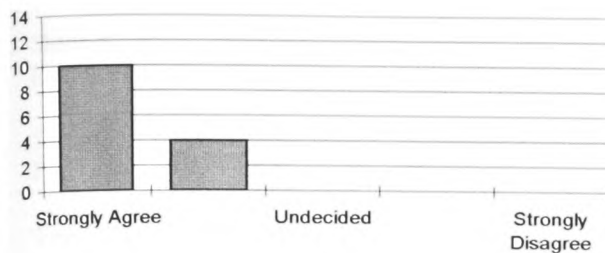- the student learning experience.

The Method

The method provided a discipline for the development team and gave them a structured template that bridged communication between the content provider and the CBL developer. Its use increased the speed of development, with each lesson being structured and built to pre-determined guidelines. Further evidence gathered from application of UDRIPS to other subject areas indicates that it can offer the same benefit to all CBL development and, thus, should address one of the primary criticisms of this activity i.e. the time taken to complete the projects.
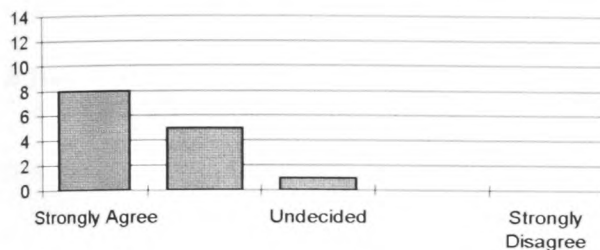
Navigation and Design Issues

Initial evaluation results show that the navigation structures were clear and unambiguous. users appreciated the opportunity to navigate in a variety of ways and expressed the opinion that they would use the CBL material again.

**The courseware would provide a useful revision tool.**

I would use this courseware again.

Student Learning Experience

Subsequent evaluations have also considered the didactic effectiveness of the CBL material. This has been addressed by means of pre- and post-tests based on questions derived to test the original learning objectives specified at the start of the CBL development process [12]. The paper took the form of a number of multi-choice questions that covered the learning objectives of the CBL material, students were also required to indicate their confidence when making a particular choice [13].
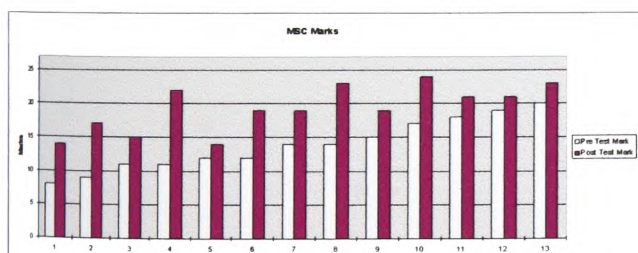

This evaluation was conducted with a small group of students who had no previous knowledge of the subject. The students were asked to complete a pre-test, then immediately given the CBL material and finally asked to complete a post-test. Both the pre-test and the post-test contained identical questions. These results show the increase in the students' performance over that period.

Future work aims to reproduce this result with larger groups from a variety of computing schemes. It is also

anticipated that the UDRIPS development method could

be used to produce open-learning materials to a high, consistent standard and work is underway to test this hypothesis.

An interesting addendum to this evaluation was the difference in the confidence levels of the students from the pre- to the post-test.
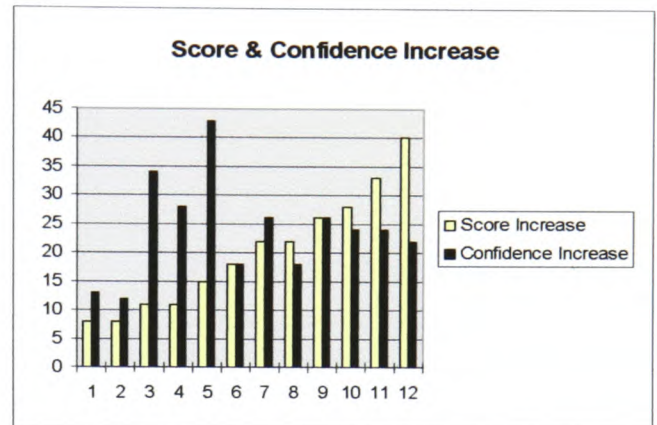
The interesting feature of this result is the lack of correlation between high increases in post-test scores and the corresponding confidence level. Some students had little increase in confidence but a high increase in score whereas, conversely, others had a high increase in confidence and a low increase in score.

Overall, every student had some increase in both their score and confidence after using the CBL material.

Future work will include evaluation of the retention of the material after a delayed period and also, a comparison of confidence levels after this time.

Bibliography

1. Gardner, N., Darby, J., Using Computers in University Teaching: A Perspective on Key Issues, Computers Educ., Vol. 15, No. 1-3, pp27 - 32, (1990).

2. Ausubel, D. P., Novak, J. D., Hanesian, H., Educational psychology, a cognitive view, 2nd ed., (1978).

3. Boyle, T., Margetts, S., The CORE Guided Approach to Acquiring Programming Skills, Computers Educ., Vol. 18, No. 1-3, pp127 - 133, (1992).

4. Gagné, R. & Briggs, L., Principles of Instructional Design, (1979).

5. Skinner, B. F., Teaching Machines, Science, 128, pp 969-977, (1958).

6. Skinner, B. F., Why We Need Teaching Machines, Harvard Educational Review, 31, (1961).

7. Skinner, B. F., Notebooks, Englewood Cliffs, Prentice-Hall, (1980).

8. Travers, J. F., Elliot, S. N. & Kratochwill, T. R., Educational Psychology, Brown & Benchmark, (1993).

9. Hawryszkiewycz, I.T., Relational Database Design, An Introduction, Prentice Hall, (1990).

10. Booch, G., Object Oriented Design, with Applications, (1991).

11. Sommerville, I., Software Engineering, 3rd ed. (1989).

12. Laurillard, D., Rethinking University Teaching, Routledge, (1993).

13. Gardner-Medwin, A.R., Confidence Assessment in the Teaching of Basic Science, ALT-J, Vol. 3, No. 1, pp 80-85, 1995.

G. Stubbs & M. Watkins. Developing CBL in W.I.S.D.E.N.: The Use of a Teaching and Learning Model, 5th Annual Conference on the Teaching of Computing, Dublin City University, Ireland, (1997)

Developing CBL in W.I.S.D.E.N.:
The Use of a Teaching and Learning Model.

Geneen Stubbs & Mike Watkins

University of Glamorgan, Pontypridd, South Wales, UK

email: gstubbs@glam.ac.uk

ABSTRACT: In this paper we will describe the teaching and learning model used to underpin the material produced by the TLTP-2 project W.I.S.D.E.N. This was a consortium based project charged with producing CBL material in the field of software design covering formal methods, structured methods and object oriented methods. The W.I.S.D.E.N. consortium consisted of seven universities and a commercial partner, the involvement in the project offered the opportunity to review the development strategies of consortium members. An initial evaluation of the prototypes, produced early in the project, revealed a diverse set of approaches to design with little evidence of the application of a methodology.

KEY WORDS: Teaching/Learning Model, Evaluation

----

1. Introduction

This paper will describe the teaching and learning model used to underpin the material produced by the W.I.S.D.E.N. project. The TLTP-2 W.I.S.D.E.N. consortium of seven, geographically disparate universities and a commercial partner had the responsibility of producing Computer Based Learning (CBL) material in the area of software design at undergraduate level. Initially, the subject area was divided into topics and allocated to the consortium members. It was necessary to refine each topic from its original high level aims, through broad objectives, to very low level learning outcomes to enable the "chunking" necessary to delimit individual lessons. This process also defined the structure of the topic as a whole since it provided an analysis of the material to reveal which part of the topic preceded and, hence, became a pre-requisite for any other. The refinement of the objectives was not an intuitive task and required a considerable amount of effort to reach the required granularity, however, only at this point can the scope of the topic, and individual lessons within that topic, be defined.

Each member was instructed to set out the learning outcomes for their particular topic and asked to develop prototypes in that area. Evaluation of these early prototypes, by an independent educational Psychologist, revealed a diversity of styles and criticism of their engagement and effectiveness. Lack of standards with regard to navigation, fonts, colours etc. was an obvious feature of these early prototypes which needed to be addressed, as were the pedagogical issues which had been uncovered. It was also necessary to ensure the needs of the target audience would be met with respect to usability, effectiveness and level.

The review of the development process resulted in the application of standards throughout the consortium material to give a common look and feel and the adoption of a teaching and learning model which had been developed at Glamorgan and was a synthesis of pedagogical and software engineering principles.

The model has the following features: it is based on learning theories and incorporates sound pedagogical principles; it offers a structured approach to CBL development; it offers consistency and reliability; it is generic, i.e. it can be applied to many subject areas; it can be taught; it is independent of implementation and it encourages modularisation.

The teaching and learning model does not directly address the preferred learning style of each individual student, rather it addresses the structure of the subject matter. The preferred learning style of the individual is addressed via the navigation strategies employed within the lesson. The navigation template used throughout the W.I.S.D.E.N. material provides the student with a composite navigation structure to allow for a serialistic or holistic learning approach (Clarke, 1990). The learning model, however, provides a logical path through the lesson without undermining the preferred learning style of each individual.

Endorsement for the use of a teaching and learning model came from a recent Coopers & Lybrand report commissioned by TLTP, which stated that many projects had underestimated the complexity of the educational task and little regard had been paid to pedagogical issues. They determined that a characteristic of an effective CBL product was that it should be supported by a clear model of learning and teaching.

Within W.I.S.D.E.N., the teaching and learning model (UDRIPS) was used to produce a prototype which was presented for initial assessment by the consortium members. There was agreement that a reproducible method for CBL development would be invaluable and each member gave the prototype careful consideration. Subsequently, the model was used to underpin the CBL lessons produced throughout the consortium (Norcliffe, 1996). One of the most interesting outcomes from an initial evaluation of the CBL development model was the common areas which many members had incorporated into their design but which they had never formalised into a usable, reproducible method for CBL development. South Bank University had a section in their lessons called "About This Topic" which contained the pre-requisites and learning outcomes for that lesson, this concurred with the UDRIPS Universal picture section which also specifies pre-requisites and learning outcomes.

2. UDRIPS: A Teaching/Learning Model for CBL Development

The UDRIPS acronym represents the constituent parts of the model from the learner's perspective:

Universal picture	where have I been, where am I going - Pre-Requisites & Learning Outcomes;
Definitions	what don't I know - Keywords & Concepts;
Rules	what is legal, what is illegal - Application & Usage of Keywords & Concepts;
Illustrative examples	are all aspects covered - Embedded Scenarios & Solutions;
Problem solving	do I really understand - Self Assessment.

Summary   what exactly did I learn ? - Summarise significant elements of the lesson.

3. Benefits of UDRIPS

The model provides a discipline for the development team and gives them a structured template that bridges communication between the content provider and the CBL developer. Its use increases the speed of development, with each lesson being structured and built to pre-determined guidelines. Further evidence gathered from application of UDRIPS to other subject areas indicates that it can offer the same benefit to all CBL development and, thus, should address one of the primary criticisms of this activity i.e. the time taken to complete the projects.

To illustrate the application of this model to CBL material, an example from the three topic areas within the Structured Methods field produced at Glamorgan will be used. The three areas are: Entity-Relationship Modelling; Normalisation; Entity Life Histories.
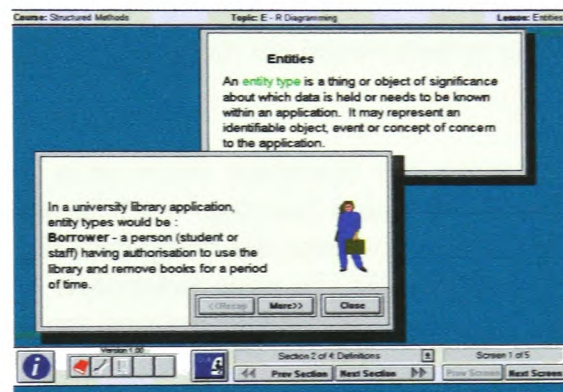
4. Material

The particular CBL material used will be E-R Modelling.

- The **universal picture** is presented by informing the user of the pre-requisites and learning outcomes for a particular lesson within a topic.
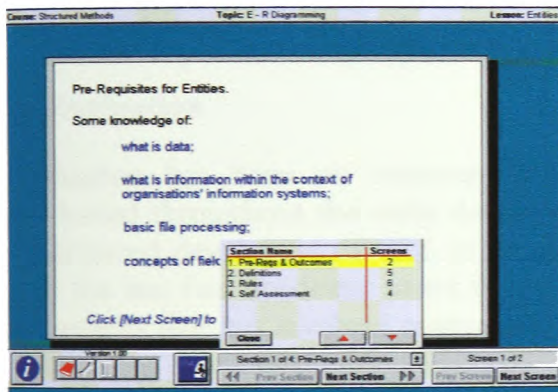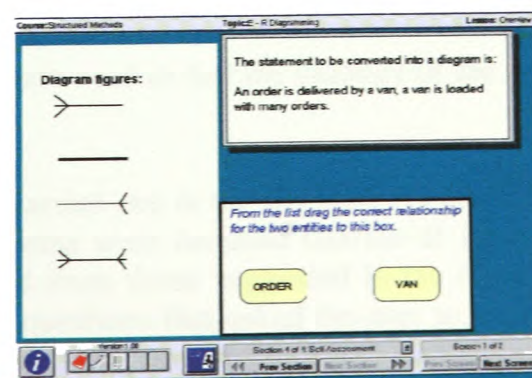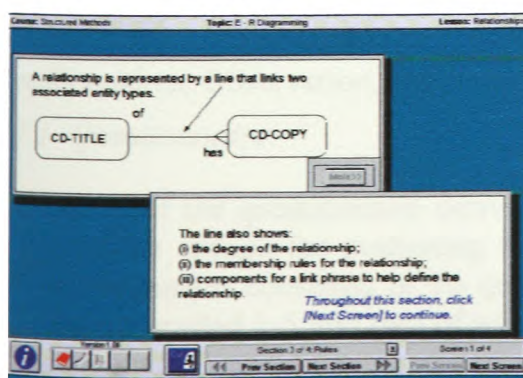

Pre-requisites   Definition and Examples

- The **Definitions** are of any keywords or concepts to clarify the new situation the student is faced with, so that, in future, the keywords or concepts can be used or manipulated with confidence.

- The **Rules** are presented to illustrate what are the "legal" situations for the use or application of the subject to be learned.

- The **Illustrative examples** are given to demonstrate the subject area from many points of view.
- The **problem solving** can be used in two ways. From the point of view of the student it can highlight weaknesses in their knowledge and in the application of their knowledge. From the point of view of the lecturer it can show how well the student has learned the material.
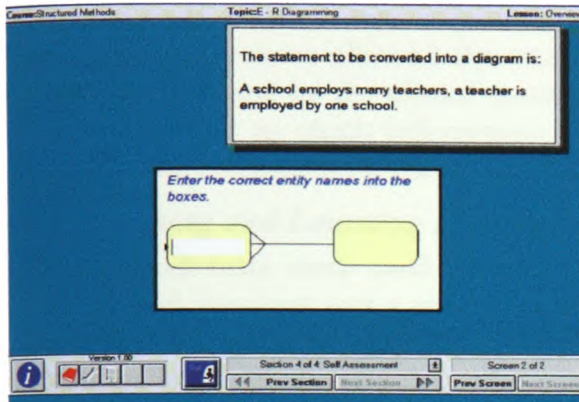
Problem
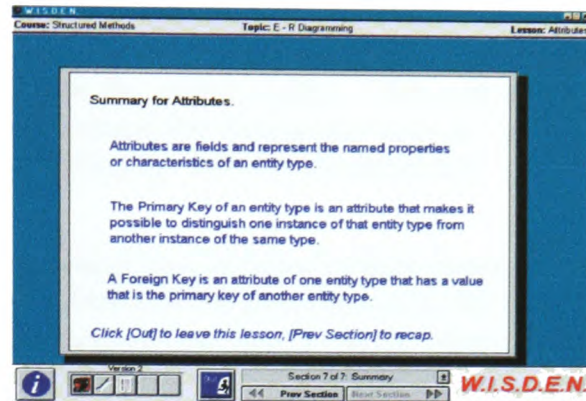
Summary

- The **summary** provides the opportunity for the student to reflect on what was important in the lesson. It provides a synopsis of the fundamental issues which the lesson tried to convey.

5. Evaluation

Evaluation has been an important activity for consortium members, the process has been conducted throughout the early development phase of the project and is an ongoing activity. As experienced developers of CBL materials for use in undergraduate computing modules, we have, over the last two academic years 95/96 and 96/97 undertaken trials of CBL material with groups of students that has involved evaluation of the student learning experience.

5.1 Evaluation Techniques

For the CBL developer, only through evaluation will feedback be gained that will enable improvement in the quality of the CBL product. To this end, there are a number of evaluation techniques available to achieve the required outcome, these include but are not limited to:
Questionnaires; Observation; Interviews; Video; Pre- & Post-Tests; Expert evaluation.
All of these techniques, with the exception of the expert evaluation are directed at the user's perception of the system.

5.2 Usability

Questionnaires, Observation, Interviews and Video were used to test the usability of the system.

5.2.1 Questionnaire

The design of the questionnaire derived from work carried out in the EC DELTA project ILDIC where fifteen criteria for evaluating Multimedia systems were deduced (Barker & King, 1993). The questionnaire consisted of 26 questions adapted from those suggested in the report, 24 of which were graded 1-5 and two of which were open questions that asked the user to list the most attractive and unattractive features of the system. The areas assessed included navigation; screen

layout; preferred learning style; assessment of prior knowledge and acquired knowledge; help facility; level of learning and overall impression of the system.

5.2.2 Observation, Interviews and Video

Observation was conducted as the students used the CBL system and interviews were carried out at the end of the experiment. Observation was useful to ensure the students were using the system correctly and without difficulty. This aided the interpretation of the results from the pre and post-tests where it was important to know if the students had taken the evaluation exercise seriously. Interviews allowed the students to express their opinions on the system without the constraint imposed by the closed questions found in the questionnaire. Although video was used to try and elicit as much information as possible about the evaluation activity, it proved to be intrusive and affected how the students behaved during the time it was employed.

5.3 Teaching and Learning Effectiveness

Pre and post-tests were used to evaluate the teaching and learning effectiveness of the system. The pre and post-test was a short test paper that addressed the learning outcomes derived directly from the CBL material (Laurillard, 1993). The questions on the test paper were multiple choice questions but with the addition of a confidence assessor (Gardner-Medwin, 1995). Results, and conclusions drawn from the evaluations will be shown at the presentation of this paper.

6. References

Barker, P. & King, T., (1993), Evaluating Interactive Multimedia Courseware - A Methodology, Computers Educ., Vol. 21, No. 4, pp 307-319.

Clarke, J.A., (1990), Designing CAL Programs to Cater for Different Learning Styles, Computers in Education, pp 351-8.

Gardner-Medwin, A.R., (1995), Confidence Assessment in the Teaching of Basic Science, ALT-J, Vol. 3, No. 1, pp 80-85.

Laurillard, D., (1993), Rethinking University Teaching, Routledge.

Norcliffe, A., (1996), The TLTP W.I.S.D.E.N. Project - A Progress Report, 4th Annual Conference on the Teaching of Computing.

A recent review of all the TLTP products by the Accountants Coopers & Lybrand can be found at http://back.niss.ac.uk/education/hefce/pub96/m21_96.html

Details of TLTP projects can be found at TLTP Home Page: http://www.icbl.hw.ac.uk/tltp/