COMPUTER AIDED LEARNING FOR ENTRY LEVEL ACCOUNTANCY STUDENTS

KINSHUK

THESIS SUBMITTED IN PARTIAL FULFILMENT

OF THE REQUIREMENT OF DE MONTFORT UNIVERSITY

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

JULY 1996

PAGE NUMBERING AS ORIGINAL

Computer Aided Learning for Entry Level Accountancy Students

Kinshuk

Abstract

The thesis is concerned with the design, development and evaluation of cost-effective CAL modules to teach procedural knowledge, but facilitating the acquisition of conceptual knowledge, in numeric subject disciplines. Three in-house Project Byzantium CAL introductory accounting modules: Capital Investment Appraisal, Absorption Costing and Marginal Costing have been used in this study as an adjunct to traditional teaching.

The modules adopt mixed-initiative approach and provide intelligent tutoring by inferring information about students from their interactions through the interface and changing the tutoring strategies, feedback and knowledge granularity for each individual user. Implementation of the modules is in Microsoft Visual C++, where the author's main contribution has been the conceptual interface design and interface development.

The effectiveness of the in-house intelligent tutoring modules as an alternative to tutorials was examined by means of a multi-institutional evaluation study using mainly qualitative techniques. An initial two group parallel trial at one university, involving the Capital Investment Appraisal module, validated aspects of CAL module content and provided feedback on the *interface design*. This study also validated the measurement techniques and the questionnaire design. The prospective study (total sample size = 513) included a two group comparison of CAL and traditional tutoring (total 161 students) at two universities using Capital Investment Appraisal module and testing of all three CAL modules by a random sample of about 40 students at each of six institutions.

This study found traditional and CAL tutoring methods comparable for numeric parts of accounting and revealed that the students with no previous computer training, no confidence in operating computers and no feeling of enjoyment with computers, were less supportive of the CAL. The thesis also finds the Honey & Mumford learning style measurement instrument unsuitable in the context of CAL in accounting. The findings of the thesis are discussed in the context of their applicability in other numeric disciplines.

Acknowledgements

Thank you God,

for helping me through times

of joys and sorrows.

I take great pleasure in this opportunity to express my sincere gratitude and appreciation to each and every person who has made the work for this project possible.

To Professor Derek Teather, for giving me an opportunity to work under his supervision and for his unreserved support and guidance during the period of this research project.

To Mr. Ashok Patel, for his extended help from the beginning to the end of my research, specially at the times of urgencies.

To Professor Greg Wilkinson-Riddle, for providing every possible administrative support.

To Professor Martin Denscombe, for his constructive comments at early stages of my research, and for helping me in the research methodology programme.

To Mr. A. Iqbal, for his kind support and assistance throughout my work, especially in the literature search. He also provided me with parts of his own unpublished research for reference purposes.

To numerous staff at various universities for helping in the user testing of in-house developed packages.

To my parents, for allowing me to pursue higher studies while keeping all their emotions aside for the sake of my future.

To Dorota, for taking all the pains and giving me support at the time, when I was too busy to take care of myself.

Last, but not the least, to all others, whose co-operation has been invaluable in completion of my research work.

Contents

Figures

Author's Research Papers Related to the Byzantium Project

1. Knowledge characteristics: Reconsidering the design of Intelligent Tutoring Systems. *International Conference: Knowledge Transfer - '96,* London, July 22-26, 1996 (Co-authored with Ashok Patel)
2. Applied Artificial Intelligence for Teaching Numeric Topics in Engineering Disciplines. *Third International Conference on Computer Aided Learning and Instruction in Science and Engineering,* Donostia - San Sebastian, Spain, July 29-31, 1996 (Co-authored with Ashok Patel)
3. Artificial Intelligence in Learning and Assessment - A supportive software for the teaching of Accounting. *The 1996 East-West International Conference on Human-Computer Interaction,* Moscow, Russia, August 12-16, 1996 (Co-authored with Ashok Patel)
4. Intelligent Tutoring Tools - Redesigning ITSs for adequate knowledge transfer emphasis. *International Conference on Intelligent and Cognitive Systems '96,* Tehran, Iran, September 23-26, 1996 (Co-author: Ashok Patel)
5. Intelligent Tutoring Tools - A problem solving framework for learning and assessment. 1996 *Frontiers in Education Conference - Technology-Based Re-Engineering Education,* Salt Lake City, Utah, USA, November 6-9, 1996 (Co-authored with Ashok Patel)

Introduction - Computer Aided Learning in Accounting

1.1 Computer Aided Learning - An Adjunct to Traditional Teaching?

Computer aided learning (CAL), which is becoming widely available as an instructional medium, seeks to individualise the teaching and learning process so learning is more effective (Askar et. al., 1992). The aim is to provide the learners with an environment that is tailored to their learning needs and goals (Clancey & Soloway, 1990).

The increasing demand for students to acquire multi-disciplinary knowledge and skills, encouraged by modular approaches to education, has led to increased pressure on academics to ensure transfer of adequate skills to all students - some of whom may be less motivated than others. The business disciplines in general and accounting in particular have witnessed dramatic rise in cohort sizes. While staff resource for lecturing is relatively less affected by this growth (large lecture theatres), the *tutorial* aspects of teaching requires large amounts of staff resource that can not be met due to insufficient funding. Many institutions are moving to alternate week tutorials and increased reliance on self-study, again posing no great problems for well-motivated students. It is the remaining students for whom the lecturer has to ensure an adequate level of knowledge transfer. This creates a real need for cost-effective software tutors that are designed to provide an adequate level of knowledge to all or most of the students rather than simply enhance the learning experience of well motivated students. The resource intensive tutorials generally deal with the operational (*know-how*) aspect of knowledge and are suitable for software tutors (Kinshuk & Patel, 1996).

Thus this thesis is concerned with the design, development and evaluation of cost-effective intelligent tutoring modules to teach procedural knowledge, but facilitating the acquisition of conceptual knowledge, in numeric subject disciplines. The domain of accountancy forms the focus for the research which involves a multi-institutional quantitative evaluation.

1.2 Computer Aided Learning - Historical Developments

The introduction of computers in classroom teaching and technical training began in the 1950s (Megarry, 1983). In the 1960s and 1970s, more teaching machines and teaching assisted software packages were developed for teaching and job-training. The period of 1980s is considered as the second wave of interest in CAL. This is probably due to the increasing popularity of personal computers (Law & Maguire, 1993). The lower cost of hardware equipment and the availability of low cost educational software stimulated the spread of CAL to many learning centres including traditional classrooms. The issues related to the spread of computer aided learning are discussed in Chapter 2 of this thesis where various factors of CAL implementation are discussed. Several advantages and disadvantages are reviewed and good and bad practices in the CAL software development are also analysed.

1.3 Advances in CAL - Intelligent Tutoring Systems

While there have been many successes, traditional CAL software packages are bound by numerous limitations and drawbacks (Epstein & Hillegeist, 1990). For instance, an average of two hundred hours is required to produce one hour of quality, interactive CAL time (Law & Maguire, 1993). This lengthy development time is extremely expensive because it is difficult and costly to find human experts to do the implementation.

Early CAL software packages were primarily built on a set of pre-defined steps. These steps were incorporated into computer programs which presented knowledge to learners through a specific order in a graphical and/or text form. The knowledge of a CAL software package did not go beyond the information stored in its library. The rigid and unresponsive nature made the instructional effectiveness less than satisfactory. On the other hand, a human instructor always uses heuristics to teach and to interact with students. As a consequence, after the introduction of Artificial Intelligence (AI) to CAL, AI researchers in the 1970s started to transfer expertise from human professionals to a machine accessible form and then redirected this knowledge back to other human beings (Barr & Feigenbaum, 1982). AI researchers attempted to organise the problem-solving skill and expertise in a way which was suitable for teaching and consulting. To develop such an *intelligent* teaching software system, AI techniques such as knowledge representation, user modelling, and natural language processing have widely been incorporated into Intelligent Tutoring Systems (ITSs). Chapter 3 of this thesis reviews the evolution of ITSs, various structures of ITSs available in literature, pedagogues related to the ITSs and current trends in the development of ITSs.

1.4 Factors in the Successful Implementation of Intelligent Tutoring Systems

An important factor in the success of an intelligent tutoring system is the design of its components, particularly the *inference engine* and *interface*. Where the inference engine gives all the desired internal functionality in the system, the interface plays an important role in the implementation of ITSs. The whole of the user's interaction with the inference engine and other parts of an ITS is through the interface. O'Malley (1990) argued that the tutoring component of any learning environment requires an explicit representation of itself through the interface. Chapter 4 of this thesis reviews different approaches towards interface design and examines the role of interface design in various educational environments.

The success of an intelligent tutoring system also depends on whether the students like it or not. Students have different attitudes towards their learning methods and have different cognitive styles, and they may accept or reject the tutoring system depending on whether or not the system fits their learning attributes. Every subject area has its own characteristics and students with different cognitive styles may have different aptitude towards a subject.

Different cognitive styles have been described in the literature and many different measurement tools are available for these styles. Chapter 5 of this thesis presents a summary of these learning styles and measurement methods. The chapter discusses the appropriateness of Honey & Mumford Learning Style Questionnaire (Honey & Mumford, 1986) in context of computer aided learning and the categorisation of people according to their learning styles.

1.5 Suitability of CAL packages for Knowledge Acquisition

The approach utilised in CAL packages for knowledge transfer to the students is an important factor in the success of computer aided learning. In practice, there are two types of educational software: the first type is tutoring software based around simple interaction involving dialogue between a student and the software. Constrained by the absence of natural language processing, this dialogue requires atomising of knowledge. Although, this approach supports the acquisition of knowledge in small doses, there is a danger of fragmentation creating obstacles in the mental construction of the full picture. The interaction may be enhanced by employing video and audio. The second type of software is a facilitator involving hypertext and hypermedia. Here the software acts as a friendly librarian rather than a tutor and learning depends on the initiative of a student (Patel & Kinshuk, 1996a).

1.6 Computer Aided Learning in Accounting

The Accounting profession has arisen from the practical skills of the 18th century book-keeper. Modern accounting discipline covers many specialised branches of knowledge and practice and new areas of specialisation, for example, Environmental Accounting, are emerging. At the core of all branches of accounting are the skills of recording, organising, analysing, interpreting and presenting information. The practical nature of accountancy, especially audit, is a justification frequently used for articleship - an apprenticeship approach to train aspiring accountants.

The issues relating to the degree to which accountancy can be learnt in an academic institution and needs to be learnt in an actual work environment are discussed in Chapter 6. The possibility of decomposing large tasks into sub-tasks is considered and the benefits to the process of learning identified. In particular such an approach enables our limited short-term memory to grasp the workings of each component, facilitating learning from much simple interactions at the component level, making the use of computers in accounting education feasible.

Chapter 6 also describes the historical developments and current trends in the use of computers for accounting education. The need to acquire practical competences and the role of practice in this process is discussed in Chapter 7, which considers one way in which the computers can be harnessed for educational purpose. The chapter also describes the in-house approach to designing tutoring systems.

1.7 The Byzantium Project - Objectives and Practical Issues

Despite a number of attempts to develop computer aided learning material in accounting (see chapter 6) the need still remains to find an efficient and cost-effective alternative to tutorials. To fulfil this need, a prototype intelligent tutoring system was developed at De Montfort University to teach Marginal Costing, and subsequently, a consortium of six universities (TLTP Byzantium) was awarded with funding from Higher Education Funding Councils of United Kingdom to develop four management accounting and two financial accounting modules. The objectives of Byzantium project are (Patel & Kinshuk, 1996b):

- to employ useful software tools within the overall learning environment consisting of human teachers and educational technologies;
- to add intelligence to the software tools to provide a degree of support to students, enabling them to work by themselves;
- to let intelligent tutoring evolve from practically useful applications, in a bottom up fashion rather than be designed top down;
- to understand the economies of the learning environment and be concerned with assessment and course management as they consume substantial human resource;
- to appreciate the economies of software production and to recognise that tutoring software is a joint cognitive system (Dalal & Kasper, 1994) and at times it is more economic to let the student explore the working of tutoring software than to spend huge efforts and expenses in designing an ITS that attempts to comprehend all the mental processes of different students with diverse personalities and backgrounds;
- to acknowledge that various educational technologies have their own strengths and that a learning environment benefits synergistically from an appropriate use of multiple resources - the converse also being true in the learning environment becoming suboptimal through inappropriate use of the educational technologies.

These objectives are further discussed in chapter 7 along with the design strategy adopted in developing the software modules. The development was started in October 1993 with Capital Investment Appraisal module in Clipper language under Windows environment, but soon was moved to Microsoft C/C++ to achieve greater flexibility and functionality at user as well as programmer level. All successive modules were developed using graphical user interface development techniques as described in chapter 8. The author's main contribution in the development of these modules has been the conceptual interface design, interface development and integration of various routines into the programs.

1.8 The Byzantium Project - Conceptual Issues

The in-house accounting CAL modules are different from generally available CAL accounting packages in the sense that they provide intelligent tutoring by adapting their tutoring strategies and feedback for individual user. These modules adopt mixed-initiative strategy to teach procedural skills of accounting and related concepts.

The *Basic Concepts* part of these packages provide understanding of the subject concepts, both initially and as a focused inquiry after attempting to solve a problem in the *Project* part of the packages. The *Project* part presents the problem structured to introduce the concepts with increasing complexity.

As described in chapter 7, the packages are designed in such a way that the system infers the information about the students' understanding of the concepts by their interactions through the interface. The mixed-initiative approach of the packages allows the students to select alternative paths to solve a problem and the system changes the tutoring strategies and feedback according to students' actions and the inferred understanding of the concepts so far encountered (see section 7.4.6).

1.9 The Research Issues

Although, the development of essential skills and understanding of concepts of a discipline may require hands-on practice in real world situations, favouring apprenticeship teaching methods, it is necessary to grasp the core or basic skills and domain knowledge before concepts can be utilised and extended to new situations. This is particularly true for the accounting discipline as it contains a major practical dimension.

This thesis is based on the above approach to teaching accounting in which basic skills must first be acquired. The approach adopted by project Byzantium is reviewed, where computer aided learning modules are developed to be used in an interactive way for the tutorial purposes. As explained in chapter 7 (section 7.4.6), the modules adopt mixed-initiative approach and provide intelligent tutoring by inferring information about students from their interactions through the interface and changing the tutoring strategies and feedback for each individual user, similar to a human tutor. The knowledge granularity is also modified as and when necessary.

While CAL modules free up human resources by replacing some mechanical aspects of teaching, their use should come in practice only after determining their effectiveness. There is a clear need for well designed evaluation to examine comparability of CAL modules to the human tutor. Does the CAL modules' adaptation of its tutoring strategies to the individual student accomplish efficient knowledge transfer?

The evaluation of CAL in accounting has not received much attention in the literature. Although there are some instances of small evaluations which have been completed within single institutions, little work has been done on large scale evaluations conducted across several universities.

Thus this research has involved a multi-institutional evaluation of the effectiveness of in-house developed intelligent tutoring systems as an alternative to tutorials in teaching. The subjective views of the students-users towards the functionality and effectiveness of the interface of these packages is also sought. The results of this evaluation are not limited to the in-house accounting modules, but may provide insight in other numeric disciplines where students learn the use of numeric interrelationships of the particular subject matter. The aim of the evaluations is to demonstrate whether the CAL modules work in practice. Initial evaluation studies carried out by other researchers to investigate the effectiveness of CAL modules (Legree et. al., 1993; Murray, 1993; Mark & Greer, 1993) have been, mainly quantitative approach. A quantitative approach has also been adopted in this thesis, although qualitative views of students were also obtained to examine student opinion towards the in-house CAL modules. The evaluation has been carried out at several institutions on three management accounting modules. Although the evaluation study was laboratory based control testing, which may not provide a fully accurate picture as to how the students would behave in real teaching environment, the multi-institutional nature of this study brings it near to a field trial.

The thesis reviews the differences in students performance at various institutions, which provides an overview of suitability of CAL modules in different teaching scenarios. For example, the thesis addresses the question as to whether the in-house modules are more suitable to new universities (previous polytechnics), where students receive training for day to day real world problems requiring adequate practical skills or to traditional universities where conceptual knowledge is viewed more essential than practical skills.

A further research question addressed in this work is whether the CAL modules are equally effective for students with different learning attitudes. The views of student-users are also examined to analyse their reactions towards CAL modules, specially with regard to the interface and the integration of procedural skills of subject matter with the interface. These views are examined in the light of differences in students' attributes such as gender, learning styles and computer background.

Chapter 9 deals with the evaluation approach and statistical techniques used for evaluation in this study, whereas chapter 10 provides the actual analysis of the evaluation study. The overall summary, conclusion and recommendations of the study are reported in chapter 11 of this thesis.

1.10 Summary

The study is concerned with the design, development and evaluation of software tutors which replace resource intense tutorial aspect of introductory teaching of numerical aspects of accounting. The mixed-initiative approach has been used in the software tutors which allows them to infer the information about the students understanding through interactions with interface and modify the feedback and tutoring strategies accordingly. These tutors aim to provide the teachers with a cost-effective means of facilitating the learning of numeric concepts in a large population of entry level students. This research seeks to examine the overall effectiveness of these software tutors which provide an integration of traditional teaching (lectures) with educational technology (tutorials) to enhance the student learning in today's resource scarce environment.

CAL - An Alternative Learning Environment

The introduction of personal computers in the academic curriculum, was accompanied by predictions that Computer Aided Learning (CAL) would change the whole attitude of the educational world and the way by which students acquire knowledge (Wong, 1994). Teachers were expected to tailor their own lessons by using an authoring program. Wong (1994) also found that, not many commercial CAL packages were available in the market and whatever was available, was confined to so small subject units that it was not sufficient to fulfil the needs of an eager educator to cover a substantial amount of syllabus. As the number of computers increased in classrooms, teachers declined preparing their own lessons and started relying on commercial software. Major reasons for this change include the refinement in the quality of available CAL software packages and complexity in replacement of outdated textbook content from self- tailored lessons prepared using authoring packages (Wong, 1994).

2.1 CAL- What Does it Really Mean!

The term *Computer Aided Learning* is generally meant to include the use of any special or general purpose software or a combination of the two, to enable utilisation of information technology for educational purpose. Jensen & Sandlin (1992) offer a definitive taxonomy:

CAT - (Computer Aided Teaching) includes in class computerised teaching aids and involves teaching with the instructor interacting live with students either in class or on a network;

CAI - (Computer Aided Instruction) refers to self-learning on a computer and includes drill lessons, tutorials and interactive dialogues software;

CMI - (Computer Managed Instruction) provides for management of student tests, lesson tracking and performance reports and includes drill lessons, tutorials, computer administered examinations and interactive dialogue between the student and the computer.

Their consideration of CAL as a meta-acronym for CAT, CAI and CMI, has been criticised by some authors (Williams & Newton-Ingham, 1994), on the grounds of its underlying inference that teaching and instruction is synonymous with learning. Objections of these nature have prompted many authors to use terms like Computer Based Learning, Computer Mediated Instruction, Computer Aided Education, Course Management Software etc. in an attempt to define more precisely the nature of software under consideration.

2.2 Current Implementations of CAL

A wide variety of CAL packages are already available in the market and there is considerable discussion on the suitability of these packages in various education curriculum. Gardner (1990) pointed out that the most routine and simple tutorial (question and answer) or multiple choice style courseware can be as effective in stimulating students to study, discuss and assimilate learning as any sophisticated 'What if ...?' modelling system. Having said that, it is hard to beat a good piece of simulation or modelling courseware which challenges students and is interesting and meaningful to them! *Steamer* (Hollen et. al., 1984) was a good historical example which simulated the steam propulsion plant of a large ship and was used to train operators. The good thing about the *Steamer* was that it used graphics and colour to show dials and process change, which is very similar conceptually to what happens in the world. *Guidon* (Clancey, 1987), the medical student teaching program, is another good example of modelling courseware of its time, with the uniqueness that it was the first attempt to adept a pre-existing expert system into an intelligent tutor. Recent research includes B. Woolf's *Cardiac Tutor* (Kaplan & Rock, 1995) which provides training for advanced cardiac life support in an emergency room environment. The qualities of *Cardiac Tutor* include the provision of graphic interfaces that let instructors change the topics, the tutoring strategies, and the content. Hammond et. al. (1992) pointed out why CAL is taking longer to be a part of the classrooms. The reasons include the quality and nature of the available materials and the educational context in which the technology is to be used. The instructional goals embedded within a piece of courseware all too often fail to match the goals of the course it purports to support. Yet other reasons concern the organisational and political context: institutions and departments, whether by intent or default, may give little assistance to those wishing to exploit innovative approaches. Indeed, there is some argument that the key factors required to develop educational computing in universities are organisational, and some researchers advocate a strategic approach through centralised but co-operative management (Bidin et. al., 1990). Certainly models of the adoption of computing into the university curriculum based solely on characteristics of individual lecturers tend to have little predictive value (Mudd et. al., 1988).

In spite of the above facts, CAL is still coming in the classrooms and has its own advantages and disadvantages over traditional teaching and learning (Gallagher & Letza, 1991). These are now considered.

2.3 Advantages of CAL

- Instructors are better equipped to capture the attention of the current generation of students who have grown up with television, videos and electronic games (Jensen & Sandlin, 1992). Wilson (1990) of the Texas Education Corporation points out:

  > *" When potential students sit in on a class session, they realise how easy it has been made for them to approach learning, and that completely changes the psychology of the classroom. For many, this is perhaps their first positive learning experience. The result is that they stay in class ... Improved learning results since students' attentiveness is held at a high level throughout the class [emphasis added]."*

- CAL provides a suitable alternative for lecturers in response to the current trend of increased number of students and higher teacher/student ratios and the subsequent pressures resulting on the lecturers (McDonough et. al., 1994a; Gladwin et. al., 1992; Darby, 1992a).
- CAL is well suited to offer help in the area of remedial teaching because of its flexibility, i.e. course modules incorporating varying levels of expertise, drill and practice exercises (McDonough et. al., 1994b).
- CAL allows the students to work at their own pace. Different students may have difficulty with different concepts. Fast learners can go ahead (McDonough et. al, 1994a).
- The students are allowed to spend more time on areas with which they have difficulty. Students may concentrate on specific areas without holding up the rest of the group (Gallagher & Letza, 1991).
- Small departments can benefit from CAL by reducing certain costs, e.g. expensive laboratory experiments can be replaced by simulations and there may be less need to have human experts constantly available (Gladwin et. al., 1992).
- Use of multimedia in CAL attracts even quite young students, and captures the interest of reluctant learners (Lambart, 1990).
- CAL ensures a more consistent course delivery and eliminates the need to cope with different tutoring styles and personality clashes between teacher and student (Gallagher & Letza, 1991).
- Teachers are free to do other necessary work, hence increasing educational productivity (Woodhouse & McDougall, 1986).
- Feedback may be immediate so increasing its effectiveness. If there is a time lag between productivity of work and its assignment, the impact of such work and elements of learning may be lost. Delayed feedback will not be so effective as the students retention of knowledge will diminish as time progresses (Wong, 1994).

- CAL is attractively flexible as students can learn in their own free time without the need of constant tutor guidance. They can skip sections covering topics which they already have sufficient knowledge (Gallagher & Letza, 1991).
- CAL forces active participation of student. It works on one-to-one basis so that students must answer questions. Whereas in a classroom, students can just keep quite and let others answer (Gallagher & Letza, 1991).
- When CAL is used to replace traditional teaching, its cost is justified due to high student usage and reusability in various classes (McDonough et. al., 1994a).
- Integration of computers in subject curriculum helps prepare the students for their subsequent careers by familiarising them with information technology and PCs (Shaoul, 1989).
- Because of the size of computer disks, the space required to store hard copies is not needed (Jensen & Sandlin, 1992).

2.4 Disadvantages of CAL

- The best CAL software cannot imitate the subtle attributes of a well- trained, ambitious, conscientious and gifted lecturer (Villiers et. al., 1992).
- The effort and cost of developing a CAL course may be four times that required for instructor led courses (Wehr, 1988). The cost of developing CAL software not only involves salaries and training of personnel as well as the purchase price of the assisting software and hardware but may also include hidden costs such as student use fees, distribution fees, annual maintenance fees and the commercial marketing of the courseware (Collins, 1989).
- Normally CAL is used to supplement traditional teaching rather than to replace it, in which case the cost of CAL must be added to the normal teaching costs (Hendley & Jurascheck, 1992).
- The developer may be more interested in what machine can do as opposed to the specific needs of the user (Rowntree, 1987).

- CAL may be poor at developing teamwork skills, communication and interpersonal skills being a one-to-one system whereas these skills can be enhanced only in group (Gallagher & Letza, 1991). Recently some researchers (Dillenbourg & Self, 1992; Repman, 1993; Wan, 1994) have started to explore the area of collaborative learning which has the potential of promoting these skills. These computer-based learning environments are discussed later in the chapter.
- The availability of hardware is a constraint on the utility of CAL (McDonough et. al., 1994a).
- Teachers' fear of being replaced by machinery, may lead to opposition of CAL as a teaching method (Beilby, 1987).
- A large number of CAL experts agree that most CAL software is of a questionable quality. Whatever is available, the scope of that tends to be topics rather than modules or courses (Darby, 1992b).
- As the students have more control over the learning process, they may skip important areas because of boredom, lack of time or simply unawareness of its importance (Gallagher & Letza, 1991).
- Most of the CAL programs do not care for students state of motivation and attention, that is possible in human teaching situations (Dixon, 1988).
- Computer discs are easier to damage or lose and more difficult to identify than a book or paper file (Gallagher & Letza, 1991).
- The computer may result in being nothing more than a highly expensive electronic "page turner" if the students are faced with page after page of indigestible text (Gallagher & Letza, 1991).

It has been found in the current project that one needs a good knowledge of the abilities of target group of students, the subject to teach, and sound background of didactic principles, to develop CAL packages. Besides that, one also needs to be good programmer. It is very difficult, if not impossible, to do this without a development team. Although there are many tools available nowadays to aid in the development of CAL programs and some are easy to use without requiring good programming skills, these programs require a very high specifications in terms of hardware. Also, the level of flexibility achieved at programmer level is not the same as compared with developing CAL programs from scratch.

However, Faiola (1989) has warned that too large a team may result in communication deficiencies.

2.5 Good and Bad Practices in CAL Software Development

- The primary focus of CAL should be on material that truly aids learning. A surprising amount of learning material, both computer and other, is vague in content (Anoskey & Catrambone, 1992).
- Good CAL software requires a high quality of academic input from discipline specialists for its design (Darby, 1992a).
- Many CAL programs currently being developed are heavily influenced by the hardware and software limitations. Technology, not education, has dominated (Villiers et. al., 1992).
- Overuse of colours and frequent changes in colour scheme should be avoided (Jensen & Sandlin, 1992).
- CAL material should present text and graphic materials to learners in a co-ordinated manner and use the exercises and questioning techniques in a way so that learners remain active during the learning process (Abouserie & Moss, 1992).
- A critical problem is the failure to realise the potential of the computer as an interactive medium. Poor software packages, even multimedia ones, display too much for students to read on the screen before doing anything (Wong, 1994).
- Friendly dialogues should be used instead of constant repetition of stock phrases as students progress through the program.

- ► Long messages should be avoided in a CAL program as it is found that students do not read them fully. Brevity and accuracy should be balanced to retain the context (Wong, 1994). Messages should appear at the same place to retain consistency (Gallagher & Letza, 1991).
- ► A second offender is the "help" facility, a critical capability of good computer-based learning material. Often a procedural help is repeated instead of a proper context-sensitive help (Dix et. al, 1993).
- ► Poorly prepared computer material provides only a single learning path, forcing everyone through the same approach in a lecture like fashion (Villiers et. al., 1992).
- ► Advanced learning modules contain "front ends" that pre-validate common errors of typing slip-ups in student input preventing its mix-up with conceptual errors.
- ► Immediate feedback results in better retention of subject matter than delayed feedback (Azevedo, 1995).
- ► CAL programs should allow students to exit the program at will to avoid frustration (Wong, 1994).

Bell (1992) pointed out the experiences of Coventry Computer Based Learning Project and gave a few suggestions.

- ♦ The most effective and efficient model for the management of computer-based learning is obtained when the computers are grouped together in a dedicated room under the supervision of a learning centre administrator.
- ♦ The room chosen for the learning centre should be large enough to accommodate the computers and other resource materials and also provide space for students to work away from the computers.
- ♦ CAL should be used whenever possible within a structured learning management system.

- The best applications of CAL make full use of the special characteristics of the medium:
    - ⇒ Individual learning
    - ⇒ Automatic learning management
    - ⇒ Immediacy of the learning package
    - ⇒ Animation and graphics capabilities
    - ⇒ Privacy of the learning process
    - ⇒ Inter-activity providing immediate feedback
- The use of the computer has been found to be a stimulus for communication; the students spontaneously discuss among themselves their experiences while using the computer. This is particularly true when the students are working together on software packages such as business simulations.
- The individualised nature of the lessons and the immediate feedback provided combine to increase the concentration level of those with learning difficulties, and the variety of possible input methods make the learning process easier for those with co-ordination difficulties.
- It is important that the courseware designers are involved in the testing and validations of the software they have developed.

It was also analysed during initial stages of in-house software evaluation that many students who do not have previous exposure to computers need some time to familiarise themselves with the system. This slows down the process of actual learning.

2.6 Classifications of CAL

In the literature (Moonen et. al., 1983; Alessi et. al., 1985), there are a number of descriptions of the form that CAL may take. A classical approach is to divide CAL into three main streams: *drill and practice* (or exercise programs), *tutorial* (or instructional programs), and *simulation*. Apart from these types of CAL, Jong et.

al. (1992) distinguished three more areas; problem *solving* (specific learning goal), *testing* (examine the knowledge), and *databases* (extracting information).

Another classification of CAL programs comes from the level of control that can be exercised over the sequence taken by the learner through the program. At one extreme it is the program that fully decides the steps to be taken through the courseware. The other extreme would be that in which the learner may choose any part of the courseware at any time. Most usual is control in which the learner may choose a path at specific points in the program. These points may be different topics, but also, for example, the ability to ask questions or seek clarifications at any time. Often systems contain an advisory route from which the student is allowed to diverge in certain ways (Jong et. al., 1992).

2.7 Computer-Based Learning Environments

Computer-based learning environments are newly emerging examples of the use of computers to help student explore and develop their own potential. They are quite different from other forms of CAL. Hsu et. al. (1993) noted the uniqueness of computer-based learning environments in that they allow students to work toward objectives that are not specifically set and paced by a computer program. Bentley (1991) states that computer-based learning environments harness the true power of computer and link it with a human approach which focuses on the psychology of learning. Accordingly, there are four inherent features in which computer-based learning environments outperform other forms of learning environments. The first is the speed with which the computer can respond to individual learners' need. The second is the way that the computer can offer, and respond to, a wide range of learner interaction. The third is the potential to provide information in a wide scope of formats from text to video; and the fourth is the opportunity to provide an almost unlimited choice of learning paths. These four features of the computer-based learning environments can only be replicated by the most extensive development of the traditional approach - and even then the choice is limited to one or two carefully pre-defined paths.

Dillenbourg & Self (1992) described an emerging field of human-computer collaborative learning (HCCL). In HCCL, a human learner and a computerised learner (a model of learner generated by the computer which tries to find what experiences human learner has gained using the system) collaborate to learn from experience. The computerised learner is called the *co-learner*. Both learners share the problem solving experience they have with a computer generated micro-world. An HCCL includes five components: (i) a micro world; (ii) the human learner; (iii) a computerised 'co-learner'; (iv) the interface through which learners interact with the micro-world; (v) the interface between the two learners (figure 1).

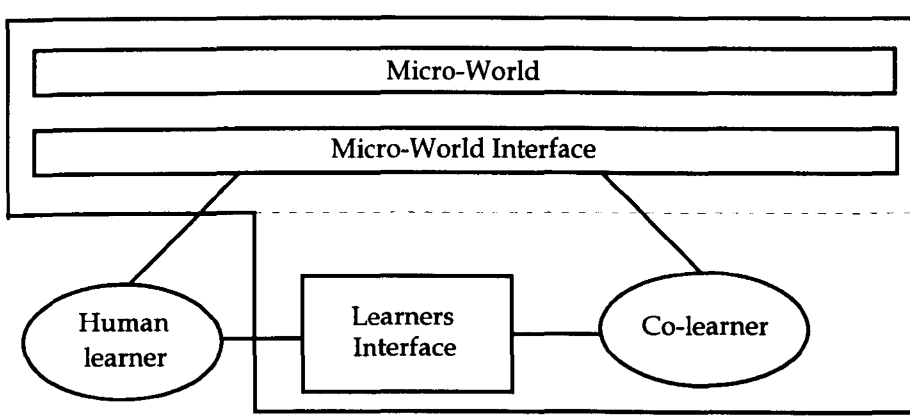

Figure 1. Components of an HCCL system (Dillenbourg & Self, 1992)

To some extent, any interactive software could be seen as a HCCL system. However, a collaborative system supposes a symmetrical interaction among learners: the same range of interventions must be available for both learners.

Repman (1993) showed different views towards computer-based collaborative environments. According to Repman (1993), in computer-based collaborative environment, students learn in a group with the help of computers while interacting with each other. Repman (1993) suggests that collaborative computer-based learning promotes more and better work, more successful problem solving, and higher performance on factual recognition, application and problem-solving tasks. Although warning that, in practice, computer-based collaboration involves one student "expert" who takes an active role in entering information or experimenting with different tactics to solve the problem, while the remainder of the group acts as passive observers, Repman (1993) suggests that the use of structured protocols to direct student to student interactions should minimise both off-task and passive behaviour, while providing more opportunities for all students, regardless of ability level, to give and receive elaborated explanations.

Wan (1994) devised a computer-supported learning environment, called CLARE, that facilitates meaningful learning through collaborative knowledge construction. CLARE stands for *Collaborative Learning And Research Environment*. It consists of three essential components:

- A knowledge representation language called RESRA that serves as a meta-cognitive framework for understanding scientific research literature and learners' perspectives;
- A process model called SECAI that prescribes a systematic procedure to guide learners in interpreting, discussing and relating knowledge elements derived from the selected research artefact; and
- A computer-based environment that integrates RESRA and SECAI into a consistent, hypertext-based interface.

Bentley (1991) commented that, to create truly effective computer-based learning environments the designers have to be learners themselves. It is essential, therefore, that the people designing, writing and building the program are definitely *not* the so-called 'subject-matter experts'. The designers of effective learning environments need to have access to 'subject-matter experts', but unless the designers themselves are ignorant of the subject area it is very difficult for them to take the position of the learner.

When people know a subject well they tend to make broad assumptions about what others know, or need to know. They find it virtually impossible to take the point of view of a complete novice. They do, therefore, usually take the position of a person who is already well advanced in learning the subject. This prevents them from developing programs of exploration and discovery (Bentley, 1991).

2.8 Conclusion

This chapter discussed various aspects of implementation of computer aided learning in real academic environment and reviewed the benefits and limitations of the implementation. Various classifications of CAL were also discussed with an overview of emerging technique of computer-based learning environments.

Next chapter provides a detailed overview of Intelligent Tutoring Systems (ITSs), a technique of computer aided learning which been a major area of research in recent years. Besides reviewing the history of ITSs, the next chapter also outlines the current research work and future expectations of this field.

Intelligent Tutoring Systems - A New Approach to Teaching

"It is not often that any man can have so much knowledge of another, as is necessary to make instruction useful."

Samuel Johnson (1752)

Intelligent Tutoring Systems are computer programs that instruct the student in an intelligent way (VanLehn, 1988). They are designed to incorporate techniques from the artificial intelligence community to provide tutors which know *what* they teach, *who* they teach and *how* to teach it (Nwana, 1990).

The goal of an ITS is to apply artificial intelligence methods and techniques to develop highly individualised computer-based instructional (CBI) environments in which the student and computer tutor can have a flexibility that closely resembles what actually occurs when student and human tutor sit down one-on-one and attempt to teach and learn together (Seidel & Park, 1994). Such flexibility is important because without it, the system cannot be fully adaptive to the individual student's on-going learning needs during instruction.

3.1 Comparative Study of ITSs and Traditional CBI Systems

Seidel and Park (1994) described several intelligent features which are intrinsic to ITS that are difficult or impossible to include in traditional CBI using ordinary programming techniques.

- ITS can generate knowledge, rather than selecting pre-programmed frames containing knowledge, to present to the student spontaneously according to the student's on-going needs during the instructional process.

- ITS allows both the system and student to initiate instructional activities by applying AI techniques. This "mixed initiative" approach is an important intelligent feature to simulate the live one-to-one tutoring process.
- ITS can make inferences in interpreting the student's inputs, diagnosing misconceptions and learning needs, and generating instructional presentations on the basis of what is available that time.
- ITS can monitor, evaluate and improve its own performance by applying AI techniques commonly used in machine learning.
- Modelling of the student learning process and qualitative decision-making of instruction are intelligent features of ITS.

A comparison between ITS and traditional CBI systems (Seidel & Park, 1994) is presented in table 1.

Issue	*CBI*	*ITS*
Development Goal	Practical use for instructional improvement	Exploration of new technology (AI) in instruction
Theoretical Base	Existing theories and principles of learning and instruction	Information processing theories and cognitive science
System Structure and Process	Frame-oriented static and single structure; prespecified algorithmic system-initiative process	Process-oriented dynamic and modular structure; generative, mixed initiative process
Instructional Principles	Various, but mostly program-initiated expository approach	Mostly student-centered discovery approach
Knowledge Structuring Methods	Mainly task analysis for identifying subtasks and content elements	Knowledge representation techniques for structuring knowledge into the system
Student Modelling Method	Binary judgement or quantitative assessment	Reasoning and qualitative evaluation
Instructional Formats	Mostly expository tutorials drill and practice, games and simulations	Mostly inquiry tutorials, games, and simulations
Subject Matter Areas	Virtually every area	Limited to well-structured areas
Development Process and Team	Systems approaches; instructional designer, subject matter expert, and programmer	Prototyping approach; Mostly AI expert (or knowledge engineer) only

Evaluation	Formative and summative evaluation; instructional effectiveness	Validation of functional running; mainly technical debugging
Hardware and Software	Mostly general-purpose hardware and software or authoring tools	AI-purpose hardware and software (LISP and PROLOG)

Table 1. A comparison between ITS and traditional CBI (Seidel & Park, 1994)

Seidel and Park (1994) also identified the need to incorporate some of the useful features of CBI into ITS. They found that there has been continuous research and development in the area of ITSs over a long period of time. The next section discusses the development of ITS from the very beginning.

3.2 Development of ITSs in Context of CAL

In the 1950's, Skinner (1958) proposed simple 'Linear Programs', based on the principle of operant conditioning. Material in these programs was presented step by step to the student in a series of frames. Most frames had simple questions with immediate feedback. Programs proceeded regardless of students' understanding of previous frames. These programs did not provide individualisation, all students received exactly the same material in exactly the same sequence irrespective of their abilities, background, previous knowledge of the domain. Carbonell (1970) commented that with this type of systems '*the computer does little more than what a programmed text book can do*'.

Crowder (1959) proposed using student's responses to control the material shown to the student. The 'branching programs' that resulted still had a fixed number of frames, but were able to comment on a student's response and then use it to choose the next frame. The main features of these programs were that they offered corrective feedback, and adapted the selection of teaching material to the students.

Pattern matching techniques allowed alternative answers to be treated as acceptable or partially acceptable, rather than totally correct or incorrect as demanded by Skinnerian systems. However, the teaching material became too large to be manageable through straight forward programming and so a special breed of programming languages, called 'authoring languages', were developed for creating CAI material (Nwana, 1990).

In the late 1960's and early 1970's 'generative systems' or so-called 'adaptive systems' came into picture in which meaningful problems could be generated and solved by the computer. The intention was to do away with all the pre-stored teaching material, problems, solutions and associated diagnostics, and actually generate them. This drastically reduced the memory usage and systems could generate and provide as many problems as the student needed to some desired level of difficulty. Uhr (1969) implemented a series of systems which generated problems in arithmetic that were 'tailor-made' to a student's performance. Suppes (1967) and Woods & Hartley (1971) produced systems with similar abilities. These programs were restricted to drill-type exercises in the domain as well as structure. They did not possess any real knowledge of the domain and they could not answer questions. The gap between the student's cognitive processes and the internal workings of the programs was too wide (Sokolnicki, 1991). Only parametric summaries of behaviour were used to guide problem generation, rather than an explicit representation of the student's knowledge (Sleeman & Brown, 1982b). The model of the student was primitive, sometimes consisting of only an integer to indicate the level of the student's competence. Yazdani (1986) notes that

> *'none of these systems (CAI) has human-like knowledge of the domain it is teaching, nor can it answer the serious questions from the students as to "why" and "how" the task is performed.'*

Hawkes et. al. (1986) noted that CAI systems were lacking of many reasons.

- They attempted to produce total courses rather than concentrating on building systems for more limited topics.
- They had severe natural language barriers which restricted user interaction with them.
- They had no 'knowledge' or 'understanding' of the subject they tutored or of the students themselves.
- They tended to be static rather than dynamic. There was little experimentation with systems in order to improve them.

Self (1974) argued that a computer tutorial program should have a representation of what is being taught, who is being taught and how to teach the student. ITSs should dynamically analyse the solution history and use principles to decide what to do next, rather than requiring solutions to be anticipated by the author of the program. Providing a truly 'intelligent' system was recognised to be a non-trivial task which needed experts from several disciplines. Some of the major features of ITSs are discussed next.

- ITSs provide a clear articulation of knowledge for a limited domain.
- ITSs have a model of student performance which is dynamically maintained and is used to drive instruction.
- The ITS designer defines the knowledge and the inferable rules, but not the teaching sequence, which is derived by the program.
- ITSs provide detailed diagnostics of errors rather than simply drill and practice.
- Students can pose questions to an ITS.

Table 2 provides some of the historically prominent ITS efforts (Kaplan & Rock, 1995).

ITS	*Developer*	*Year*	*Domain*	*Key ITS issues*
SCHOLAR	Carbonell	1970	Geography	1st ITS, N. L. Dialogue
WHY	Stevens, Collins	1977	Meteorology	Socratic Dialogue, Tutoring
SOPHIE	Brown, Burton	1977	Electronics	NLP Interface, Black-Box
WUSOR	Goldstein	1979	Game Strategy	Overlay Structure
GUIDON	Clancey	1981	Mycin Tutor	Expert Systems, Glass-Box
WEST	Burton	1981	Game Strategy	Coaching, Example-based
BUGGY	Brown	1981	Arithmetic	Represent incorrect knowledge
DEBUGGY	Burton, VanLehn	1982	Arithmetic	Off-line diagnostics
STEAMER	Stevens, Hollan	1983	Navy boiler design	Simulation, Mental models
LMS	Sleeman	1984	Algebra	MAL-Rules
MENO	Woolf	1984	Meteorology. Programming	Discourse management
PROUST	Johnson	1984	Programming	Intention diagnosis
ACTP	Anderson	1984	Lisp tutor	Cognitive modelling
SIERRA	VanLehn	1987	Arithmetic	Bug prediction
SHERLOCK	Lesgold, Katz	1991	AF Electronics	Cognitive apprenticeship

Table 2. Some of the historically prominent ITS efforts (Kaplan & Rock, 1995)

Seidel and Park (1994) made seven specific recommendations for the development of ITSs:

1. The AI methods and techniques should be applied for improving instructional functions in ITSs rather than for demonstrating functional features.
2. More emphasis should be given on the prescriptive process of instruction.
3. ITSs development should be a multi-disciplinary team effort, including instructional psychologists and subject matter experts.
4. The development of ITSs should combine the use of instructional task analysis (particularly, cognitive aspects) and AI knowledge representation techniques.
5. More considerations should be given to differences among individual students.

6. Instructional effectiveness and efficiency should be evaluated while developing ITSs.
7. More development and delivery should be done using micro-computers.

3.3 Structure of ITSs

Various architectures are proposed in literature for ITS systems and it is almost impossible to find two ITSs based on same architecture (Nwana, 1990). There are currently no clear cut guidelines available regarding the general structure of the ITSs architecture (Yazdani, 1986, 1987). Previously, there was some consensus that the architecture of an ITS consists of at least three basic components (Barr & Feigenbaum, 1982; Bonnet, 1985).

- The expert knowledge module.
- The student model module.
- The tutoring module.

Afterwards, one more component has been identified and has been added as fourth one in the list (Wenger, 1987; Burns & Capps, 1988; Mandl & Lesgold, 1988).

- The user interface module.

Figure 2 shows an idealised structure of a ITS (Brusilovskiy, 1994).

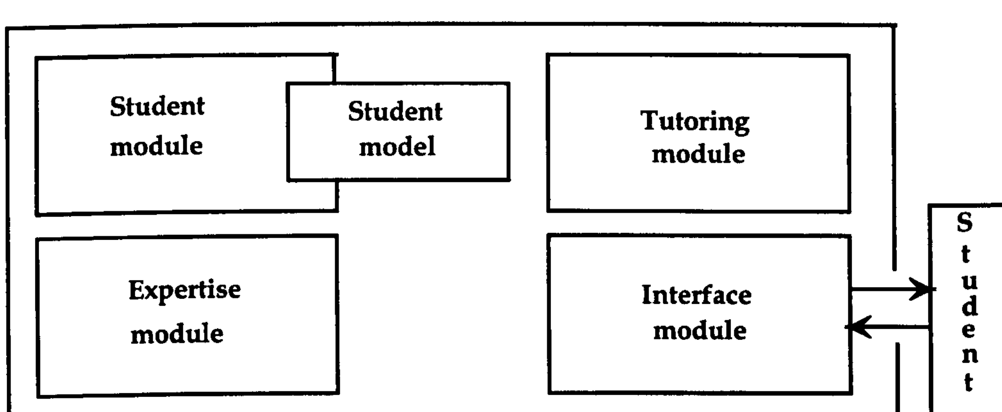

Figure 2. Idealised structure of an intelligent tutoring system (Brusilovskiy, 1994)

Expert Knowledge Module

The expert knowledge module comprises the facts and rules of the particular domain to be conveyed to the student, i.e. the knowledge of the experts, which is generally derived from people who have years of experience in the domain. Currently expert knowledge in ITSs is represented in various ways, including semantic networks, frames and production systems, and includes both surface knowledge (e.g. the descriptions of various concepts that the student has to acquire), and the representational ability which is a critical part of expertise (Mandl & Lesgold, 1988).

Expert knowledge modules range from completely opaque or 'black-box' representations, whereby only final results are available (a classic example is tutor SOPHIE I), to fully transparent or 'glass-box' ones, where each reasoning step can be inspected and interpreted (e.g. SOPHIE III). These modules, on one hand, serve as the source of knowledge to be presented to the student, which includes generating questions, explanations and responses, and on other hand, they provide a standard for evaluating the student's performance by generating comparable solutions to problems in the same context as the student. The modules must also be able to detect common systematic mistakes, and any resulting gap in the student's knowledge. The expert modules must also be able to generate sensible, and possibly multiple, solution paths to compare intermediate steps, to provide student monitoring.

The expert knowledge module by necessity embodies a specific view of the domain of the designer. Thus, tutoring can be compromised if the student does not understand the system's instruction or if the system cannot interpret the student's behaviour in terms of its own view of the knowledge (Wenger, 1987). Of course, human teachers also have their own views and styles of teaching which students may find difficult to understand, but these human teachers have the incredible ability to adopt the styles and way of expressions which students can understand.

<u>*Student Model Module*</u>

The student model is the component of an ITS that represents the students' current state of knowledge (VanLehn, 1994). The student model module represents the dynamic representation of the emerging knowledge and skill of the student. Most researchers agree that a student model is an integral part of an ITSs (Hartley & Sleeman, 1973; Self, 1974, 1987a, 1987b, 1988; Rich, 1979; Sleeman, 1985; Tobias, 1985; Zissos & Witten, 1985; Ross et. al., 1987; Glimore & Self, 1988; Wachsmuth, 1988). Ideally, this model should include all those aspects of the students' behaviour and knowledge that have possible repercussions on their performance and learning. But as Self (1990) commented

> "*student modelling is not about building exact cognitive models. If it were, we would have to solve all the problems of cognitive science, and teach a machine to be a cognitive scientist, before we could build a student model. We only need to model the student to the level of detail necessary for the teaching decisions we are able to take.*"

Self (1988) identified twenty different uses that had been found for student models in existing ITSs. After analysis, he noted that the functions of student models could generally be classified into six types.

- *Corrective*: to help eradicate bugs in the students' knowledge.
- *Elaborative*: to help correct 'incomplete' student knowledge.
- *Strategic*: to help initiate significant changes in the tutorial strategy other than the tactical decisions of 1 and 2 above.
- *Diagnostic*: to help diagnose bugs in the students' knowledge.
- *Predictive*: to help determine the students' likely response to tutorial actions.
- *Evaluative*: to help assess the student or the ITS.

Wenger (1988) and Nwana (1990) explained two *super* functions of student models: acting as a *source of information about the student*, and serving as a *representation of the student*. In achieving these functions, they act in roles including corrective, elaborative, strategic, diagnostic, predictive and evaluative.

Evidence to generate the student model can be obtained in four ways (Barr & Feigenbaum, 1982). These are:

1. Implicitly, by monitoring the students' problem- solving behaviour.
2. Explicitly, extracted from dialogue with the students.
3. Historically, i.e. assumptions based on the students' experience and
4. Structurally, reflecting the inherent difficulty of the material.

Techniques for building student models fall into two groups: methods for *inferring single facts* and methods for *inferring clusters of facts* (Rich, 1983). One of the aims of student modelling is to evaluate and extract information about the student.

Ohlsson (1986) called student modelling "cognitive diagnosis", as the purpose of student model is to know something about the cognitive state of the students, what they know and how they think, and preferably how they learn. Accordingly, the student model includes: *performance measures*, which indicate the proportion of the subject matter known by the student; *error descriptions* which are the distorted or misconceived "knowledge units"; and *simulations*, which are executable and allow predictions to be made about student performance.

One critical issue of student modelling is, whether the student really knows a skill after only demonstrating it in an isolated context. The opposite question is also an issue. Is it always true that some piece of knowledge is absent just because it has not been used in a certain situation? Incorrect or suboptimal behaviour may be due to incorrect versions of the target knowledge rather than from incomplete knowledge. In *Buggy* (Brown & Burton, 1978) approach, a more formative student model provides explicit representations of the students' incorrect versions of the target knowledge for remediation purposes. The student model then, must be able to indicate the abilities of students with respect to the domain being taught, as well as student preferences for particular methods of tutoring. Most ITSs model this type of information by utilising the knowledge in the expert module.

The relationship of the student model to the expert module

The construction of the student model is generally based on the model of the subject matter or the expert's knowledge model. According to the relation between this model and the student model, student models are usually classified into three types in the literature (Sleeman & Brown, 1982a; Kass, 1989): *overlay, differential* and *perturbation* models. The *overlay model* assumes that all differences between the students' behaviour and the behaviour of the expert model can be explained by the lack of skills on the part of the student (Goldstein & Carr, 1977; Goldstein, 1982). *Differential modelling* divides the class of things of which the system believes the student does not know into those which the student should know (if they are to behave like the expert), and those things which the student could not be expected to know. Both overlay and differential models assume that knowledge of the student is only a subset of the knowledge of the expert, and therefore do not represent any belief of the user which is outside the domain knowledge of the expert. In *perturbation models* the student model can differ, on the basis of small perturbations, from some of the knowledge in the expert model. A lack of knowledge is just another form of perturbation- for instance, common bugs or misconceptions.

<u>*Tutoring Module*</u>

The tutoring module, or instructional module (Vasandani & Govindaraj, 1995) is the part of the ITS that designs and regulates instructional interactions with the students. In other architectures, this module is referred to as the teaching strategy or the pedagogic module. It is closely linked to the student model, using knowledge about the student, and its own tutorial goal structure to decide which pedagogic activities will be presented: hints to overcome impasses in performance, advice, support, explanations, different practice tasks, tests to confirm hypotheses in the student's model, etc. (Self, 1988).

The tutoring modules in existing ITSs have a wide range of systems. At one end, there are systems that monitor the students' every activity closely, adapting their actions to the students' responses but never relinquishing control. *Steamer* (Hollan et. al., 1984) which teaches how to operate a steam plant in a ship; and *Quest* (White & Frederiksen, 1985) which gives qualitative understanding of electrical system trouble-shooting, are example of this type of systems. On the other extreme, there are guided-discovery learning systems where the student has almost full control of the activity, and the only way the system can direct the course of action is by modifying the environment. Examples of this type include *West* (Brown et. al., 1982) which provides on line coaching for a mathematics game; and *Proust* (Johnson & Soloway, 1984) which is a system for diagnosing nonsyntactic student errors in Pascal Programs. In the middle are mixed-initiative systems where the control is shared by the student and the system as they exchange questions and answers. *Scholar* (Carbonell, 1970) which had the knowledge base about South American geography; and *Why* (Collins, 1976) which tutors not just factual knowledge but the principles of rainfall as well, correcting student's misconceptions concerning the casual models underlying rainfall, are the examples of mixed-initiative systems. The existence of this wide spectrum clearly highlights the fact that tutoring is an art that requires great versatility which is still extremely difficult to articulate and present in an ITS. Nevertheless, some progress is being made, albeit limited.

ITSs also aim at explicitly representing the knowledge found in the tutoring module. This creates the potential to adapt and improve strategies over time (as in the case of self-improving tutors), and for the same strategies to be used for other domains.

The User Interface Module

The user interface module is the communicating component of the ITS which controls interaction between the student and the system. In both directions, it translates between the system's internal representation and an interface language that is understandable to the student. Because the user interface can make or break the ITS, no matter how 'intelligent' the internal system is, it has become customary to identify it as a distinct component of its own. In fact, it would be a mistake to consider it a secondary component of the ITS for two main reasons. First, when the ITS presents a topic, the interface can enhance or diminish the presentation. Since the interface is the final form in which the ITS presents itself, qualities such as ease of use and attractiveness could be crucial to the student's acceptance of the system. Secondly, progress in multimedia technology is increasingly providing more and more sophisticated tools whose communicative power heavily influences ITS design.

Current ITSs provide user interfaces which, for the input, range from the use of fixed menus with multiple-choice answers to a fairly free treatment of a pseudo-natural language. For the output, they range from the mere display of pre-stored texts typical of CAI, to the use of fairly complicated generic frames. Within these two ends of the spectrum there is also a varying flexibility (Wenger,1988).

3.4 Some Approaches of ITSs Development in Last Decade

- *Effects of specific instructional strategies in ITS* - Recently some ITS developers have started to examine what factors make a difference in the instructional value of an ITS and what kind of advantages ITS has for manipulating these factors. For example, Katz & Lesgold (1994) identified nine candidate instructional strategies which might have contributed to the success of *Sherlock,* including a wide range of factors from functionally relevant embedded learning to meta-cognitive skills and heuristics that leverage the domain knowledge.

  Seidel & Park (1994) suggested that the systematic integration of various instructional strategies through repeated studies, by adding one strategy at a time, holds the promise that the characteristics of effective instructional program can be defined in a generalised nature.

- *Flexibility of instructional strategy research* - Research on ITS is being focused on the application of the intelligent features to the manipulation of specific instructional strategies that are not feasible to implement in traditional CBI. For example, Shute & Glaser (1990) used *Smithdown* to investigate whether effective interrogatory skills are teachable and how tutorial guidance should be provided to teach the skills. Seidel & Park (1994) pointed out that this kind of research would be extremely difficult to conduct without the flexible environment of the ITS in which the students were allowed to exercise wide range of self-initiated learning activities. This provides a glimpse at the potential benefits of ITS research.

  Reiser, and his associates used the *GIL* (Graphical Instruction in Lisp) tutor to investigate the guiding effects of the dynamically constructed casual explanations and their visual representations on the student's reasoning. This is another example that shows how the intelligent function of ITS can be used to investigate the advantages of specific instructional strategies (i.e., dynamic graphical constructions in GIL) which are difficult to implement in a traditional CBI environment (Reiser et. al, 1989; Reiser et. al., 1991).

- *Simulating human learning and cognition* - There are efforts to develop learning theories and instructional strategies by simulating human learning processes using ITS features. For example, Ohlson (1991a), Ohlson (1991b) and Ohlson & Rees (1991) simulated the different learning processes on the same topic using the intelligent features of ITS in order to develop learning and instructional-design theories. VanLehn and his associates investigated the effects of self-explanation by simulating the student learning process in a computer model, *Cascade*, which has capability to acquire both domain knowledge and derivational knowledge (i.e. application of general knowledge) (VanLehn, 1994; VanLehn et. al., 1991). Seidel & Park (1994) commented that although this approach to the development of learning theories and instructional strategies requires a difficult assumption to accept that human cognitive learning process is the same as the computational (i.e. machine learning) process, the results provide useful information for the research and development of future ITSs.

- *Creating multimedia environments* - Many current ITS and CBI systems have capability to control the presentation of multimedia types of information, including text, sounds, high-resolution graphics, interactive video, and even animation of virtual reality, for instruction (Woolf & Hall, 1995). This technological advancement provides a great potential for ITS to be far more effective than human tutors since human tutors do not have the ability to generate or control the presentation of multimedia information. Current research includes the development of Hydrive (Hydraulics Interactive Video Experience) (Kaplan & Rock, 1995) which is an ITS that incorporates multimedia to troubleshoot an Air Force F-15 hydraulics system. Although

Hydrive uses an external laser disk to supply video images on an on-demand basis, the contents of the laser disk (motion sequences and still images) are represented in the knowledge base used by the Hydrive.

Okamoto (1994) suggested a possibility to integrate technology between ITSs and various integrated learning environments such as distributed and co-operative computer supporting environments in the facilities of multimedia and networking.

3.5 ITSs of Today and Tomorrow

Kaplan & Rock (1995) noted that ITSs are making significant contributions in variety of domains, including job training, defence hardware maintenance, help desks, legal research, and so on. Powerful, sophisticated systems can now be created on less expensive microcomputers, and the technology is beginning to move into, even in schools. In order to be successful, however, designers of these systems need to work closely with classroom teachers in order to avoid repeating many of the problems experienced in the past when attempting to integrate CBI into the classroom (Steuck, 1992).

Ohlsson (1986) pointed out that for many years, CBI and ITS have followed divergent paths, one concerned with questions of instruction and the other with "hacking" together systems that contain AI components. However, rather than considering this as a dichotomy between CBI and ITSs, it may be more useful to view both as existing on a continuum that is anchored at one end by traditional, linear CBI developed from an instructional systems design perspective, such as the tutorials found in many training settings, and at the other end by the "ideal" ITS.

The bottom line of high costs of development, delivery, and maintenance schedules continue to stare at the ITS community. The tutor/learner interaction that tries to anticipate everything ahead of time is a very expensive, if not a hopeless goal. Non-traditional ITS methods are beginning to be explored to reduce the high costs of development, delivery, and maintenance. These include reusable, component-based software, where small independent modules with their own local intelligence are linked together (Kaplan & Rock, 1995); synthetic agents which exchange information and services with other programs and thereby solve problems that cannot be solved alone (Genesereth & Ketchpel, 1994); nonsymbolic computation which automatically defines functions to generate, categorise, and measure misconceptions (Kaplan & Rock, 1995); and concept-based retrieval in which ITSs are constructed for the assistance of the concept formation at the analogy level using the analogical inference engine (Okamoto et. al., 1994).

A great deal of attention has also been paid to the interface design of ITSs in recent years due to the fact that the interface provides the user with all the functionality of the software. A major factor of users' liking of the ITSs is the comfort in interacting with the interface of the ITSs. The next chapter is focused on the interface design of ITSs where the principles and various approaches of the interface design are discussed with special focus on interfaces for learning environments.

Interface Design - Applications in Learning Environments

Human computer interface (user interface) design is the process of designing interface software so that computer systems are efficient, pleasant, easy to use and do what people want them to do. Plessis et. al. (1995) comment that the interface for a CAL program should be sensitive to the age of students, cultural backgrounds, computer literacy level and other related factors. Good interface design is based on four general principles namely *clarity, consistency, common sense* and *comfort*. These principles are to be applied in four key aspects of human computer interface: *environment, appearance, support* and *interaction*. These four parts make it possible for the system developers and users to work together towards making systems easy to use and easy to learn.

There is no clear-cut classification of interfaces available in the literature. Hutchins (1987) has provided a four-way categorisation of styles or modes of interaction: conversational interfaces, declaration interfaces, model-world interfaces and collaborative manipulation interfaces, but there is considerable amount of overlap among these styles.

Conversational interfaces are those characterised most closely by command-language styles of interaction. They are based on a 'conversational metaphor' where interaction with the system is via some intermediary language. Input in such interfaces need not necessarily be via typed input. Menu driven systems can also be conversational in style. Traditional approaches to ITS embody this conversational style of interaction in the computer-as-tutor paradigm (this includes directive teaching as well as coaching and Socratic methods).

Declaration type of interfaces are more direct than these conversational interfaces and employ certain aspects of command languages or menu designs. Examples include the use of abbreviations for certain Unix commands, such as 'dw' for 'delete word'.

Model-world interfaces do not use any kind of language which, by virtue of its flexibility of expression, entails the possibility of making errors. They contain a model world which constrains the set of possible actions so as to render making an error almost impossible. However, these constraints also reduce the 'directness' of the interaction.

Collaborative manipulation interfaces capture the best features of declaration and model-world interactions. This involves giving the system access to the world in which the user operates, to enable meta-level discussion of the domain. The 'display controller' selects display and display components, which the operator may then reject, indicating that they are not relevant in the current context. The display controller learns about the operator's preferences in the same way it has learned about the system's behaviour, by observation. The display is thus a shared world of action which can be collaboratively manipulated by the user and the display controller. O'Malley (1990) commented that, of the four styles of interaction described by Hutchins (1987), the collaborative manipulation style is probably most suited to the goals of guided learning systems.

4.1 Problems in Traditional Interface Design

Interface design principles have been present in the computer science literature and industry for more than two decades. Examples of interface design principles can be found in Martin (1973). People have complained for a long time that computer systems are difficult to use. Landseadel (1995) provides various reasons why there have been obstacles in traditional interface design practices in developing good user interfaces.

- *Development strategies*

  The traditional development approach was to develop the functionality first, and then slap a user interface on it. The main focus was on the budget and schedule performance, not on better quality of interface. No usability testing was done until later, when it was already too late for any changes, and it was too expensive to fix.

  A couple of problems with this design approach are directly related to usability. First, a user interface, even a graphical user interface (GUI), does not ensure usability. If the design is not suited to the user or to the natural flow of the task, using windows, icons, touch screens, or any other bells and whistles is not enough to make the product usable. The second problem is that because the developer is not a user, the developer may not foresee all of the usability issues. Even with years of experience and sometimes because of years of experience, the developer cannot think as a user does. The result has been many feature-laden systems which were not designed for usability.

- *Beware of software reuse*

  Potential usability issues may arise from plugging old code into a new system. For example, the error handling software may have been implemented differently, resulting in a mismatch in the functionality. There are many benefits of proper software reuse, but they could be carefully weighed against any drawbacks associated with interface inconsistency.

- *Organisational barriers*

  Many organisations do not include human factors specialists, and the user interface is often designed using the intuitions of the programmer. Even in organisations that include human factor specialists, their analysis is often applied to the design after implementation is well in progress and too late for strategic changes.

- *Previous implementations*

  In the case where a similar product has been fielded, the users may be resistant to changes, even to changes that could potentially make their jobs easier. It is harder to learn a new way to do something than to learn it for the first time.

4.2 Designing the User Interface

Interface design has become as important as the functionality of the applications, users have demanded and used pleasant, attractive and easy to use software. Computing systems are also becoming increasingly interactive. As they do so, the amount of code which is written for input and output (i.e. interface) has risen. Various technologies and tools are being developed to create better interfaces. The advent of interactive systems for developing user interfaces seems to be one of the most promising current developments in the area of user interface development of software. Various methodologies and design principles are also being established to develop user interfaces. Pangalos (1993) has laid out some design principles and methods for user interfaces.

- *User interface consistency*

  There are at least three different aspects of consistency across all applications in the user interface area, which are important to the user: *semantic consistency, syntactic consistency*, and *physical consistency*. *Semantic consistency* refers to the meaning of the elements that make up the interface, such as the result of invoking a particular command. *Syntactic consistency* refers to the words used for the commands and the sequence of the appearance of the elements comprising the interface, such as the particular word used to invoke a function. Finally, *physical consistency* refers to the hardware and how it is used, e.g. what key is pressed for a particular command.

To achieve *semantic consistency,* all aspects of the objects which are visible to the user and the actions which can be performed on these objects must be consistent. To achieve *syntactic consistency,* the same terms should be used to express the same meaning. Syntactic consistency allows users to develop strategies in terms of the common terms that can be directly applied to each application. Without syntactic consistency, users may develop strategies using common concepts but they will usually have to use different commands for similar strategies in different applications. *Physical consistency* is finally also important in simplifying the use of multiple applications. To achieve physical consistency, the same sequence of interactions should be used to request the same functions.

➢ *Direct manipulation*

Direct manipulation is enabled by the provision of dragging and picking operations for all graphical objects. In direct manipulation techniques, physical actions are included as button presses or mouse actions whose results appear immediately on the screen. The basic idea is to avoid syntactic forms or command languages, even the simpler ones, and to require only simple training. Error messages are unnecessary in many cases because the results of the actions are obvious and are easily reversed. The general approach of direct manipulation is reflected in the phrase *what you see is what you get.* Direct manipulation is attractive to the users because it is comprehensible, natural, rapid and even enjoyable.

➢ *Rapid Prototyping*

The development of a user interface for an application should pass through several iterations of prototyping, testing, evaluation and redesign before it is possible to achieve the quality and usability required. A tool that assists the design and construction of user interfaces should therefore provide methods that allow changes to take place quickly and simply, shortening the time required for each phase of the interaction.

- *Semantic feedback*

  Semantic feedback is feedback that depends on information given by the application. In the applications using semantic feedback, the flow of dialogue is usually controlled by the user interface, limiting the way in which the application can intervene to provide feedback. The application is only called in response to a user action and cannot directly affect the decisions concerning output and feedback.

4.3 Approaches to Interface Design

Wallace & Anderson (1993) defined four major approaches to interface design. The first is the *craft approach;* here each design project is viewed as unique, solutions evolve, under the guidance of a skilled human factors expert, to suit the circumstances. The second approach is *enhanced software engineering;* in this case attempts are made to introduce HCI techniques into the repertoire of traditional systems engineering. The third school can be called *cognitive engineering;* this is the most theoretical approach to interface design. It attempts to apply cognitive psychology to the problems facing designers to facilitate optimal design. Finally, there is *technologist approach;* which has had more and more impact in recent years, this approach tries to provide software tools to solve the problems of interface design. Each of these approaches is briefly summarised in table 3.

	Craft approach	*Enhanced software engineering*	*Cognitive engineering*	*Technologist approach*
Philosophy	Craft-oriented design through skill and experience	Incorporate HCI into software engineering	Apply the psychological knowledge base to achieve optimal design	Quantify and automate the design process
Character	Monolithic Evolutionary	Structured transformation	Structured transformation	Black box generation
Focus	Specification Design	Specification	Specification Evaluation	Implementation
Role of practitioner	Craftsman/artist Multi-disciplinary collaboration	Traditional analyst Broadening the scope of software engineering	Psychologist Ergonomist Human-factors specialist	Software tools developer
Tools	Brainstorming Prototyping User evaluation	Software engineering methods CASE tools	Task analysis methods GOMS, CCT, KAT	UIMS Mathematics Formal grammars

Table 3. Approaches to interface design (Wallace & Anderson, 1993)

4.4 Designing Graphical User Interface

Graphical applications are becoming more commonplace nowadays in human interaction intensive areas. Marchisio et. al. (1993) commented that graphical interfaces are known to stimulate user creativity and increase user productivity. These interfaces are successful because of simple-to-use input devices, icons, windows, and menus. Therefore it makes sense to move away from hard-to-remember, command-line-only interfaces to these, more attractive interfaces, which have dramatically widened the explicit communication channel between the user and computer.

Graphical user interfaces (GUI) provide intuitive and easy means for users to communicate with computers. However, construction of GUI software requires complex programming that is far from being intuitive. Because of the 'semantic gap' between the textual application program and its graphical interface, the programmers themselves must conceptually maintain the correspondence between the textual programming and the graphical image of the resulting interface. Miyashita et. al. (1992) proposed a programming environment based on the *programming by visual example* (PBVE) scheme, which allows the GUI designers to 'program' visual interfaces for their applications by 'drawing' the example visualisation of application data with a direct manipulation interface.

Many software tools have emerged in last few years which assist in the development of graphical user interfaces. Most of these tools, however, concentrate on the implementation of GUIs rather than on their design. In addition, some of the tools which assist in the GUI design process do not encourage the practice of participatory design, which attempts to make interfaces more usable by involving users in the design process. Examples include NeXT Interface Builder (Myers, 1992) and WindowsMAKER (BlueSky, 1991). Miller et. al. (1992) developed TelePICTIVE, which is an experimental object-oriented software prototype designed to allow naive as well as expert users to work together in designing GUIs. TelePICTIVE is designed to support designers with diverse expertise in the collaborative design of GUIs.

Beck et. al. (1994) described a methodology for the derivation of graphical user interfaces from the object model of computer applications. The design starts with a conceptual user interface model consisting of data objects as well as objects related to the interactive computer application.

4.5 Recommendations for Better Graphic Interface Design

Harrison (1995) and Iannella (1992) suggested some points for better graphic interface design. These suggestions are more related to the screen design than to design methodology.

- A key principle of attractive design is smooth contrast drop, e.g. white lettering on a black background gives too much of a drop.
- Colours also vary in brightness according to the background with which they are contrasted. Therefore, a good screen design will contain a stepped

contrast drop, e.g. black border, blue or grey background and then white lettering rather than white on black.

- Good interface should avoid:
  - using colours for no particular purpose
  - bright colours
  - hot colours (e.g. pink and magenta appear to pulse on the screen; they are good for highlighting)
  - too many colours
  - non complementary colours e.g. red and yellow, green and blue
- Good interface should aim for:
  - consistent meanings for colours
  - not more than four colours on the screen (except graphics)
  - pale, pastel colours
  - low-contrast background colour, such as grey
  - colour contrast between character and background, e.g. white lettering on grey
  - an overall style, e.g. with graded changes in contrast from background to text and highlights
- It is a good idea to keep the screen components in function areas, e.g.
  - avoid text which wraps around graphics
  - build up complex graphics in stages, perhaps with a silhouette
- A few guidelines for readable text on the screen:
  - Use as little text as possible
  - Use short lines
  - Present text in natural blocks, not a page at a time
  - Do not indent
  - Use ragged, right hand justification, which is easier to read
  - Avoid text wrapping around graphics
  - Avoid too many fonts

- Windows should be placed in the same place if possible so that the learner knows where is to expect it.
- In Windows environment, the Windows conventions should be used as much as possible.

4.6 Interface Design for Learning Environments

For a learning environment, whether it is 'non-intelligent' computer assisted learning or 'intelligent' tutoring systems, the interface is the way in which the learner has access to the functionality of the system. The implication is that it is no longer acceptable to design the nuts and bolts of the system and simply 'tack on' an interface afterwards. The underlying functionality of the system (not the particular implementation, but the actual functional behaviour) cannot be completely separated from the user interface. Functionality 'leaks through' into interaction. The development of the learner's model of the system and the domain being taught are shaped by what the learner-as-user can see, hear and do via the interface. To exploit full functionality of the system-interface interaction in a tutoring system, Chen (1995) proposes a model of interface, where a portion of user interface penetrates into system components, namely, to be distributed into the entire tutoring system. Apple Computer (Hefley, 1995) has explored the use of anthropomorphic agents as a part of interface to actually lead the user through an application. The in-house Byzantium approach considered the importance of interface integration with system functionality.

O'Malley (1990) argued that careful attention needs to be paid to three types of representations at the interface. The first is the representation of the semantics of the domain to be learned. The second is the representation of the system's functionality. This refers to the actual operations the system performs in representing the domain. It is treated as a separate type of representation, since it may not map completely onto the domain semantics. The third type of representation concerns what the user is expected to do in interacting with the system. This can be regarded as the 'affordance' the system offers for interaction.

O'Malley (1990) further noted that the interface is also the only way in which the tutoring component has access to the activities of the learner. The implication of this is that the tutoring or guidance system must have some representation of the learner machine interaction itself in order to model the development of the learner's understanding. This applies as much to active diagnosis (assessment) as it does to passive diagnosis (modelling behaviour). This issue of what needs to be represented even for a sufficiently accurate model (as opposed to an ideal model) is problematic: the learner machine interface is not bounded by the computer screen alone but includes the physical and social context in which the learner machine system is embedded.

4.6.1 Role of the Interface in Educational Environments

➢ *Interface as knowledge representation*

The design of the user interface to a piece of educational software is of vital importance to its educational effectiveness. This applies whether one is designing ITS, microworlds, simulations or any other form of computer-based learning activity.

O'Malley (1990) distinguished between two sources of knowledge representation in tutoring systems: an internal representation of the domain, and an external representation at the interface. The interface does more than map meaning onto a set of symbols; the external representation can actually compete with the internal representation as a source of domain knowledge. The external representation does not always play the ancillary role traditionally associated with user interfaces, but can be the driving force of the design of a tutoring system. For example, the design of *Guidon* (Clancey, 1983) took the traditional path from the construction of an internal model (domain representation), to augmenting the system with visual displays in *Guidon-Watch* (Richer and Clancey, 1985), whereas the design of Steamer (Hollan et. al., 1984) went from an external representation (a graphical interface) to an internal representation (using qualitative process theory), which was specifically designed to incorporate the conceptual content of the visual display into the internal representation.

O'Malley (1990) argued that the learning environments need to be designed 'outside-in' rather than 'inside-out'. There are two main reasons for this. The first is to ensure that the resulting design is usable to the extent that the interface 'disappears' for the users so that they concentrate on the domain itself. The second is to ensure that the representation by the tutoring system of the learner-as-user is sufficiently accurate.

➢ *Making the interface transparent*

If people are using computers to learn about something their task should not be made more difficult by having to struggle with learning and using the computer system itself. Educational interfaces should be easily learnable and usable, but the argument is a little more subtle than this. If the learner has to concentrate on using the interface, then attention is drawn away from the domain being taught.

4.6.2 Representing the System to the User

In order to provide guidance in discovery learning environments, the system needs to model the processes whereby the student learns from interacting with the system. Even if the theories of learning can be developed which are detailed enough for student modelling, they could not be applied to specific situations and adapted for particular learners, without being attached to domain semantics. Therefore, we need to understand how learning takes place via interaction with the system.

➢ *Representing domain semantics*

In order to teach something new, it is necessary to find out what student already knows. One approach is to represent the domain to be taught in the form of simulation. An example is *Steamer* (Hollan et. al., 1984). Another approach is to design the interface with appropriate metaphors, as in such microworlds as *Shopping on Mars* (Hennessy et. al., 1988), or the *Alternate Reality Kit* (ARK) (Smith, 1986).

There are two main advantages of using metaphors in the interface. One is that judicious use of metaphors allows users to draw analogies with their own experience. This enhances the learnability of the system because the concepts and techniques necessary for using the system are already familiar. Another advantage in using a consistent and uniform metaphor at the interface is that it enables users to make correct predictions about the behaviour of the system, even in novel or unfamiliar circumstances.

➢ *Representing system functionality*

The choice of appropriate metaphors to represent domain semantics (whether those domains are concrete or abstract) is one issue. The way in which the interface represents those metaphors is a separate issue. Ideally, there should be a direct mapping between the two, but the constraints of software and hardware technology often require an extra translation step. For example, file deletion concept can be represented by using the metaphor of throwing a document in the wastebasket, as in Windows 95 interface.

The design of metaphors such as the 'desktop' is usually achieved via graphical or iconic means. Command-based systems which use textual interaction can also convey metaphors. Moreover, the visual modality is not the only source for presenting a metaphor. Auditory icons, such as the sound of breaking glass to convey deletion of a file, can also be powerful sources of metaphors.

Icons do not simply represent (refer to) domain semantics, they also represent to a greater or lesser degree the functionality of the underlying system. For example, an icon representing a folder in the Windows 95 interface also suggests (via the user's understanding of the semantics being represented, which may be more or less successful) that items such as documents can be placed in or removed from the folder.

Successful icon representation depends on the types of representation used in mapping from symbol to referent, such as whether it's an abstract or metaphorical relationship. But the icons that go to make up metaphors such as the desktop involve sets of symbols that are related in some way, so interpretation may also depend on mappings between icons. Rogers (1988) suggested three interacting factors which contribute towards effective design of iconic interfaces: the directness of the visual link between an icon and its referent; the extent to which the referents can be differentiated within a set of icons; and the degree of compatibility between the relations of graphic elements and the common characteristics of the referents.

4.6.3 Representation of the User in the System

➢ *Defining the boundaries of the interface*

The *User-computer interface* has been viewed by different people in different way in the HCI literature. For some, it is restricted to the visual display (i.e. what the user sees), for others it includes the hardware (e.g. input devices), and for others it includes the social context in which the computer is being used.

The user-computer interface is probably best considered as a distributed system. It is partly represented in the mind of the user, in the sense that the user constructs a model of the system during the course of the interaction, which may or may not accord with what the designer had in mind. It is partly represented in the functionality of the software, partly in the appearance of the graphical objects on the screen, partly in the documentation accompanying the system, and so on. Furthermore, if the users work with computers together in communities, the interface is also represented in the community.

The slipperiness of the concept of user-computer interface increases when one considers the dynamics of user-computer interaction. Just as the interface is distributed over space (while extending the notion of space to include 'cognitive space'), so it is distributed across time, and it changes over time. This applies both at the 'low' level, in the sense that visual displays are dynamics, and at 'higher' levels, in the sense that computer systems are appropriated by communities and given particular cultural interpretation.

Decisions about the location of the boundaries of the interaction have crucial implications for the possibility of modelling the student's understanding of the domain. As Wenger (1988) said,

> "*The interface becomes epistemically removed from the system's actual domain expertise to the degree that it takes advantage of the student's ability to read more into its presentations that the system is aware of*".

➢ *Implications for student modelling*

Interfaces which are designed with a uniform and consistent metaphor can help in learning and using the system because they provide constraints within which the user can predict the effects of actions. However, the ability of a guidance system to collaborate with the user about the interaction presupposes that the system has some representation of the interface with which to reason and collaborate.

In this respect, discovery learning environments pose several problems for student modelling. Under the 'model-world' metaphor, although the learner's interaction with the system is quite constrained to prevent errors, it is still far less constrained than in traditional CAL. Whilst simple and self-contained microworlds may be fairly easy to model, modelling the functionality of a much more general and powerful environment is much more of a challenge. The problems are not simply to do with representing domain knowledge. The need to model the interaction, and have some notion of the sense that the user is making from interacting with the system, compound the problems. These issues are not specific to the learning environments, but they become sharply focused when the aim is to provide adaptive learning environments with some sort of guidance built into the system.

The data-driven nature of interaction with graphical and direct manipulation interfaces implies that the system should be able to reason not only about its own state at any moment, but also about how that state appears to the user. That is the system needs to have, in some sense, a representation of what's happening on the screen at any moment, as well as what's happening in terms of the domain being represented.

4.7 Student Preferences Towards Interfaces in Learning Environments

Cusack (1993) wrote,

> *"The area of communication that intrigues the computer industry is at the human-computer interaction level, where the machine meets the human mind, and where information is exchanged. The presentation layer, or user interface, to computer programs acts as a shield to a far more complex array of processes. Yet by simple navigational and functional commands, the user should be able to manipulate the computer program and achieve the desired results quickly. This is not always the case, however, and*

> *deficiencies in both usability and information access continue to plague 'ordinary' computer users."*

The most important consideration in interface design is how well it simplifies access to the program, how 'natural', or 'intuitive' it makes computer use. When computers were used primarily by engineers and technicians, 'ease-of-use' was not such a concern. But if computers are to be used in schools, the user interface must easily allow young, non-technical, easily frustrated users to interact with the system.

Hazari & Reaves (1994) conducted an evaluation study of Graphical User Interface Vs Command Line Interface on 102 undergraduate students enrolled in multiple sections of a Technical Writing course offered at a large, state-supported university in the south-eastern United States. Students working on the command line interface used DOS based Microsoft Word 3.0 word processing software while students working on graphical user interface used Microsoft Word 4.0 on the Macintosh platform. Hazari & Reaves (1994) found that students preferred Graphical User Interfaces (GUI) over other types of interfaces. They commented that the greater potential for enhancing teaching and learning seems to rest with GUI, not because of its short term effect on writing quality, but because it more quickly helps students use the technology and gives them more time to focus on learning. GUI users do not have to memorise complicated keystroke commands since the mouse input is used to automate routine tasks. Because strict software development standards force programmers to use a consistent design for dialogue boxes, buttons, tool palettes, and menu structures, users do not have to go through a new learning curve while learning different application programs.

4.8 Discussion

This chapter discussed the designing of user interface, problems associated with traditional interface design methods, various design principles and recommendations. Interface requirements for learning environments were also discussed, and various features affecting the learning environment interfaces have been identified.

As the Windows environment was used to develop the in-house modules, iconic metaphor used by Windows was considered better than command language based "conversational metaphor". Since the software is to be used by the novice students, pictorial representation alone was not sufficient to represent the ideas. Therefore initially, both picture and text were used to give the meaning of a picture and then afterwards, pictures were used to represent the meaning.

O'Malley (1990) argued that the design of an interface used in an educational environment implicitly embodies a domain representation. The implementation of the tutoring strategies take place primarily through the interface. This factor was considered very important in the in-house development. The approach ensured that there wasn't just a loose linkage between the main functionality and the interface of the programs, and therefore the interface design was an integral part of the module development from the beginning. The interface of in-house modules allows the system to infer the information about the students through their interactions and to present the appropriate tutoring strategies through the interface.

Various user interface consistencies (Pangalos, 1993) have been considered and interface provides a consistent approach in semantics, syntax and physical expression throughout the program.

Good design principles for graphical user interfaces suggested by various researchers (Harrison, 1995; Iannella, 1992; Marchisio et. al., 1993) have been applied while developing the interface. All the screen-shots referred in the following discussion are enclosed at the end of this chapter.

As suggested by Harrison (1995), colours have been used purposefully. For example, yellow colour on text indicates the hot keys and dark cyan indicates that information is for reference only and cannot be changed. Bright colours are avoided and similar colours are used on screens to make them restful as shown in screen-shot figure 3 and screen-shot figure 4. Number of colours on one screen are kept to a minimum. Same font is used everywhere, as is suggested in literature.

Same layout is used throughout a package (Iannella, 1992) and as much as possible, among different packages to helps students finding the information more easily as can be seen in screen-shot figures 3, 4 and 5. This helps students getting familiarised with style adopted in the packages and gives more time to learn the subject matter than to find their way in the program.

Phase I evaluation study revealed that students prefer more diagrams in Basic Concepts part of the program. Good design principles (Harrison, 1995) also recommend the sensible use of graphics to teach a concept. Therefore, many meaningful pictures and replicas of parts of project screens are incorporated in the tutorial text of Basic Concepts part of the packages as shown in screen-shot figures 6 and 7.

As suggested in good interface design principles (Marchisio et. al., 1993), icons are used in the packages to convey the information, but they are also supplemented with textual information to increase fast understanding.

Feedback is an important part of interface and a good interface provides feedback in a very helpful manner so that the cognitive overload is minimum on the students (Iannella, 1992). Therefore bottom right part of the screens is reserved in each package for interactive feedback messages and student does not have to look anywhere else for the messages any time. Messages are kept concise and to the point to make them clear and avoid confusions (screen-shot figure 5).

Good design principles also stress on keeping the amount of information as less as possible on a screen and prevent the clustering of information at any part of screen. Therefore only very necessary information is provided on screens in the in-house packages and zoom-in facility is available wherever necessary so that the students can go to deeper level of details as shown in screen-shot figure 6.

Another example of good interface design principles implementation is the provision of graphs in the packages for better understanding of concepts (screen-shot figure 7).

As O'Malley (1990) recommended, the learning of use of computers is kept to a minimum to provide more time for subject learning. To accomplish this, only two types of controls are used throughout the programs, buttons and scrollbars. The use of these controls is also made considerably intuitive, for example, buttons contain both pictorial and textual information of their tasks, and are placed constantly at one place so that user does not have to look for them everywhere.

New technologies such as multimedia were also considered while developing the interfaces. But it was found that these technologies are at their development phase and require high specification hardware and software which is not available in the student laboratories. Therefore, it was envisaged that while there is a possibility in future for inclusion of these technologies in the software modules, in current situations, it is not feasible to use high-memory demanding real-time graphics, video clips and sound effects in introductory CAL modules.

While designing the interface of a CAL program, one important consideration is the learning attributes of the learner using the program and whether the interface is catering for those attributes. The next chapter is focused on various methods of measuring those learning attributes. The techniques used in this project for determining learning styles are also described.

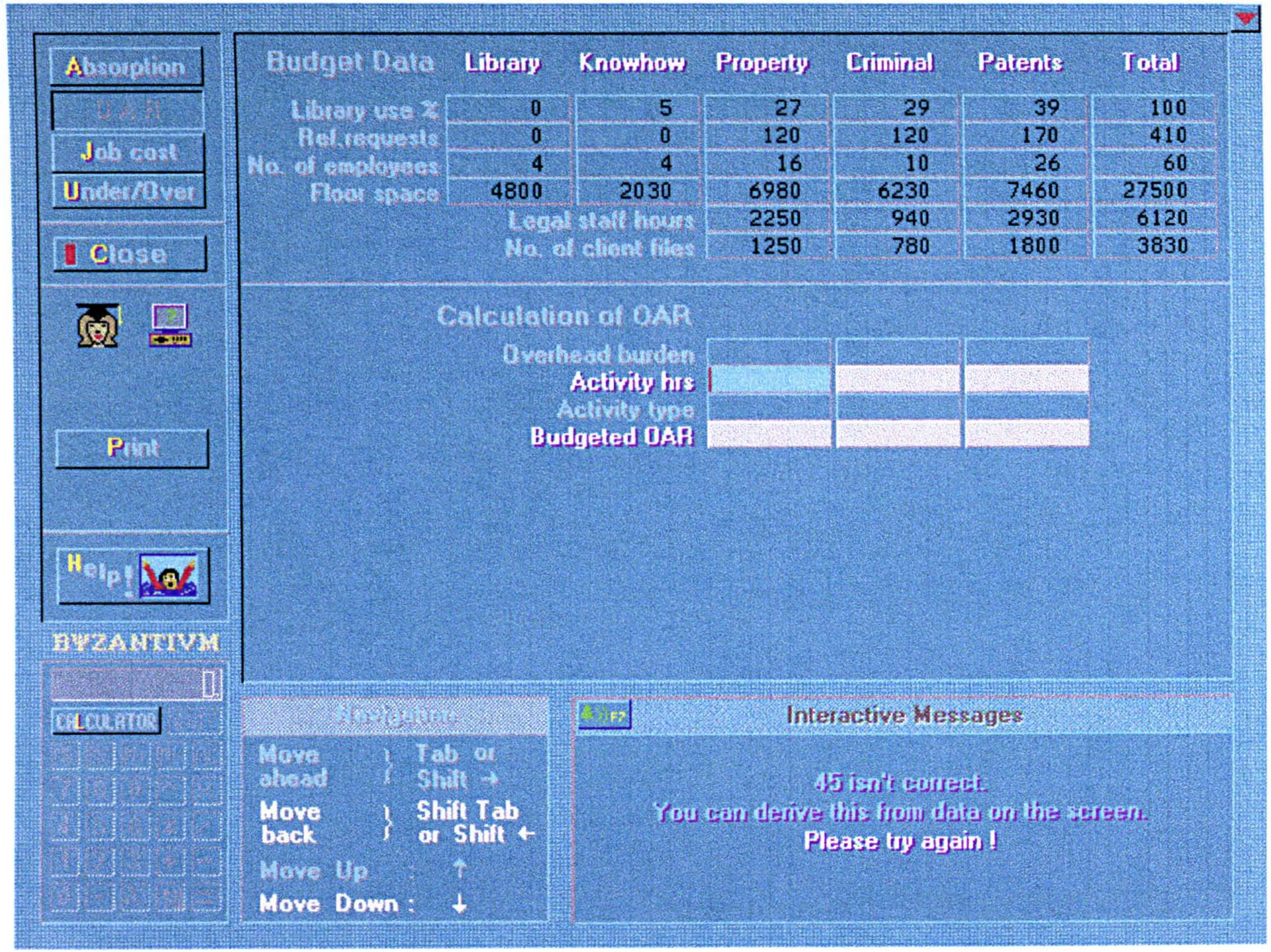

Figure 3. Screen-shot of OAR Screen of Absorption Costing Package

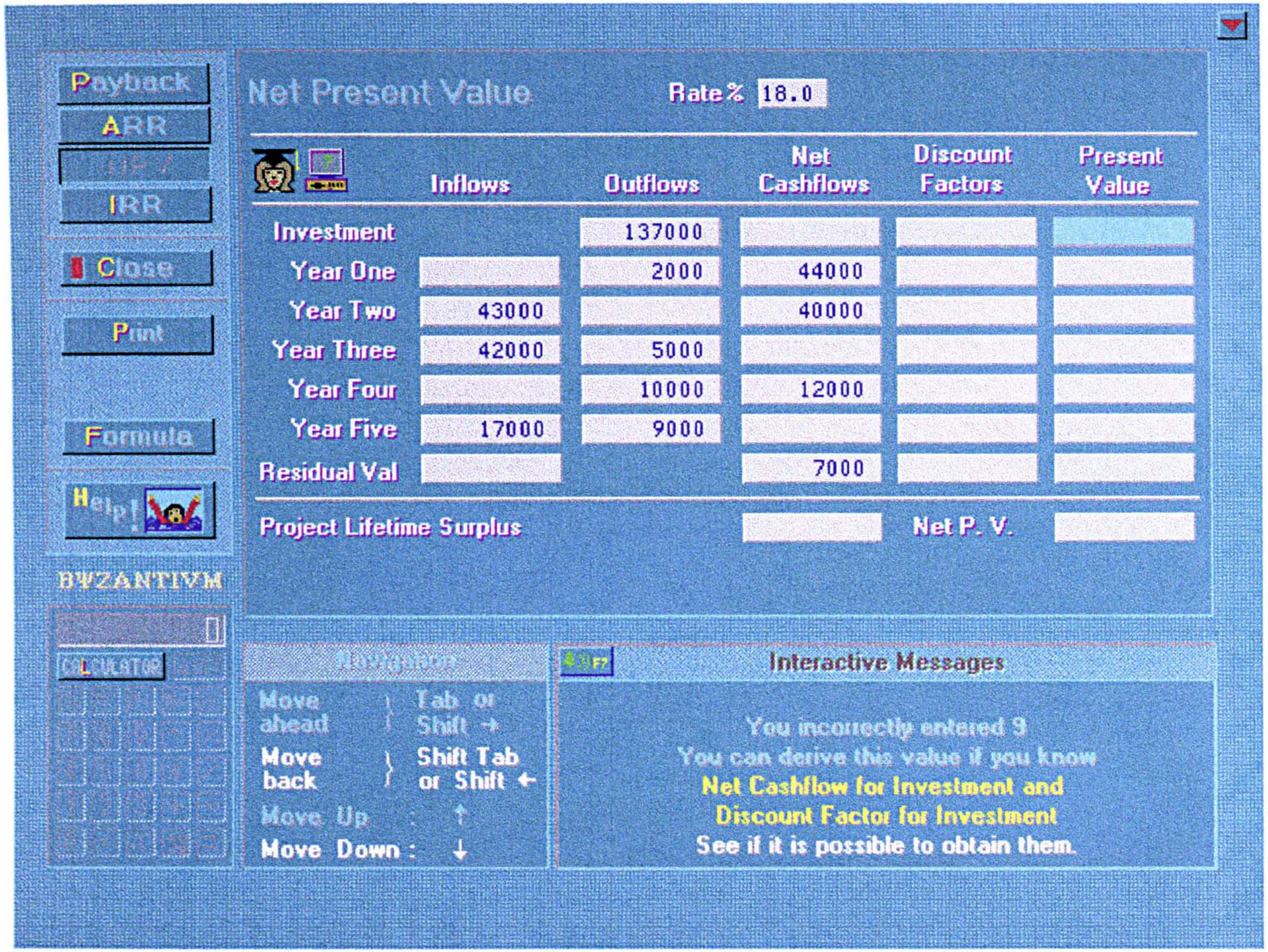

Figure 4. Screen-shot of NPV screen of Capital Investment Appraisal package

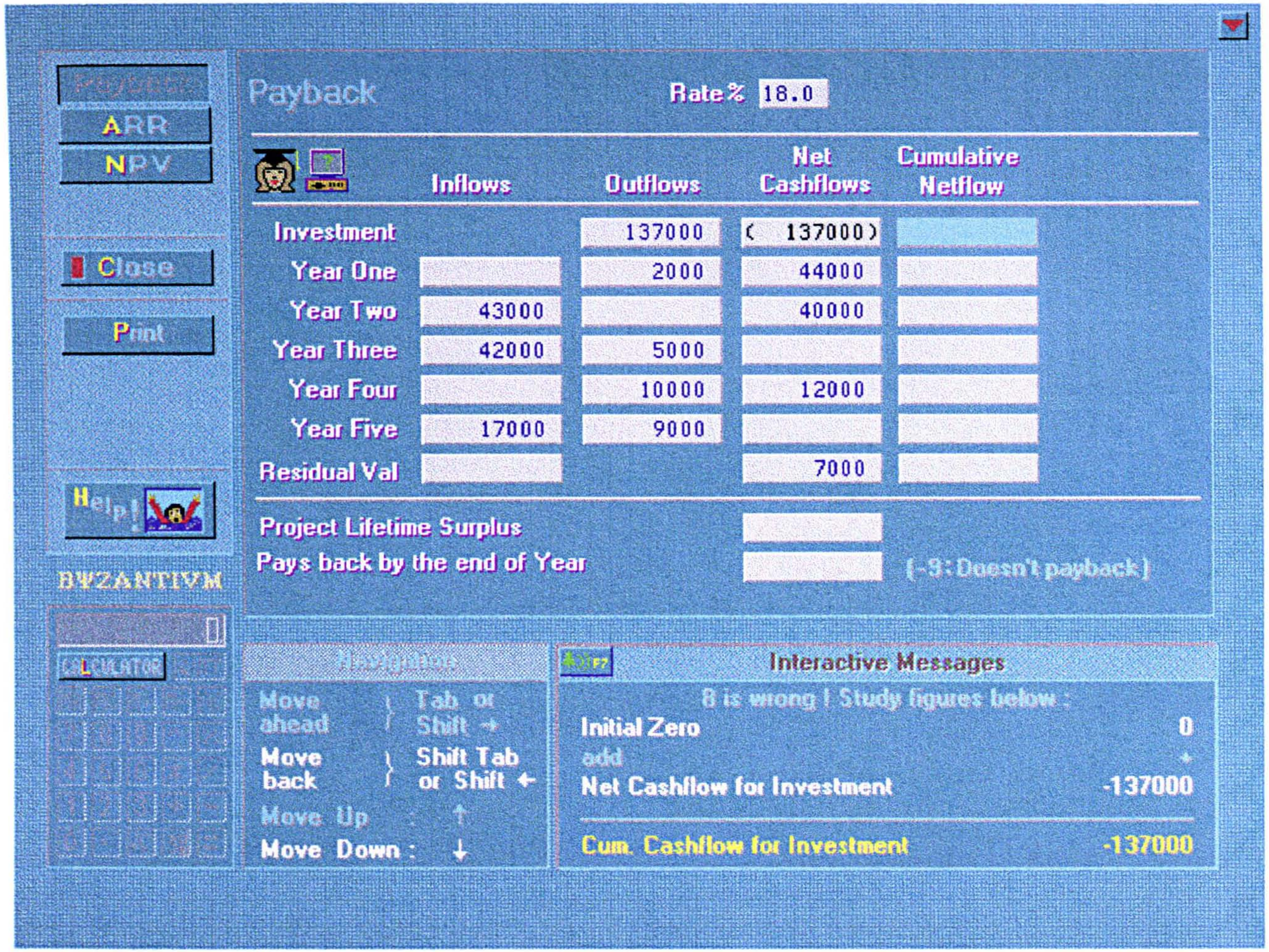

Figure 5. Screen-shot of Payback screen of Capital Investment Appraisal package

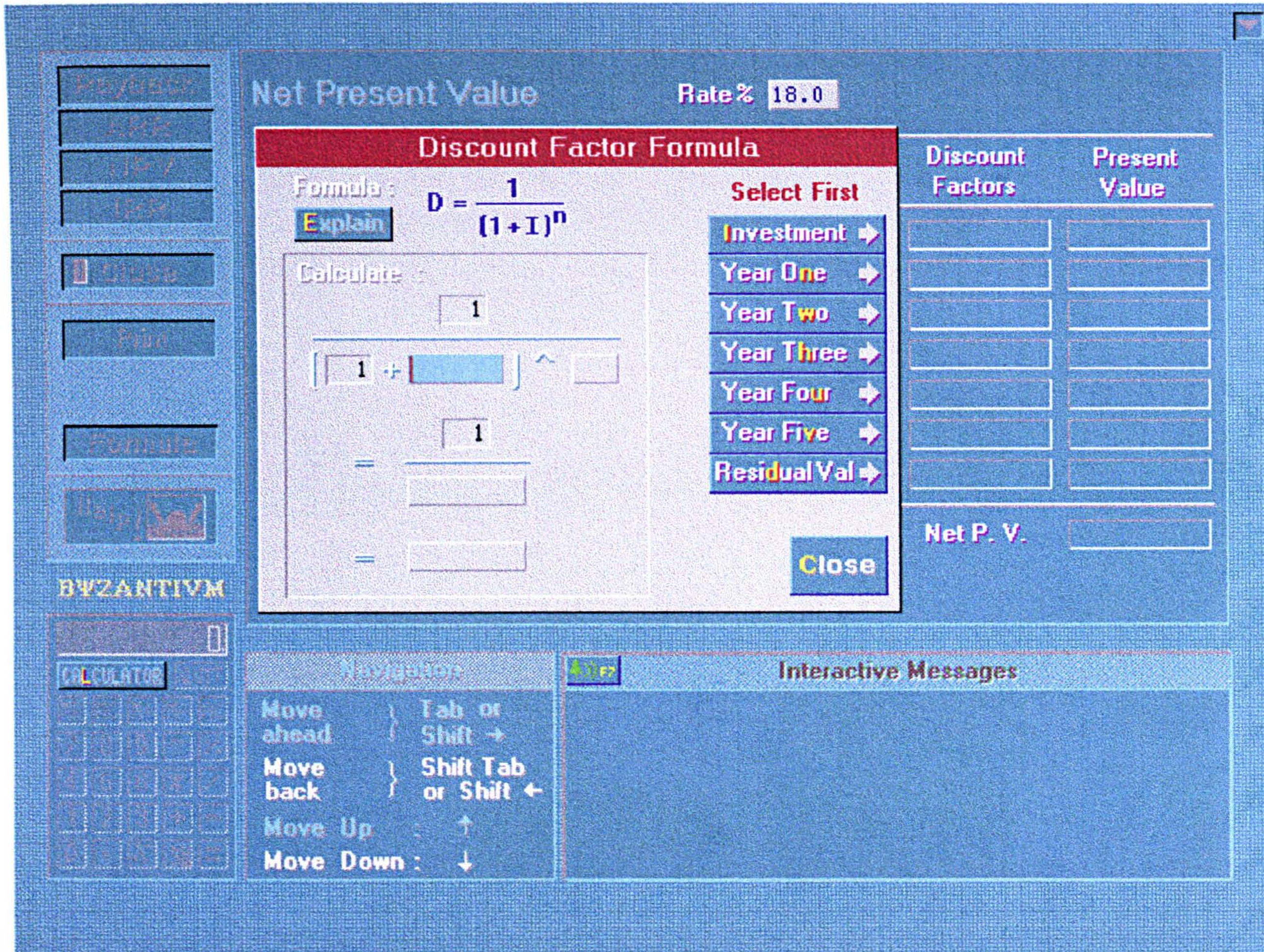

Figure 6. Screen-shot of Formula screen of Capital Investment Appraisal package

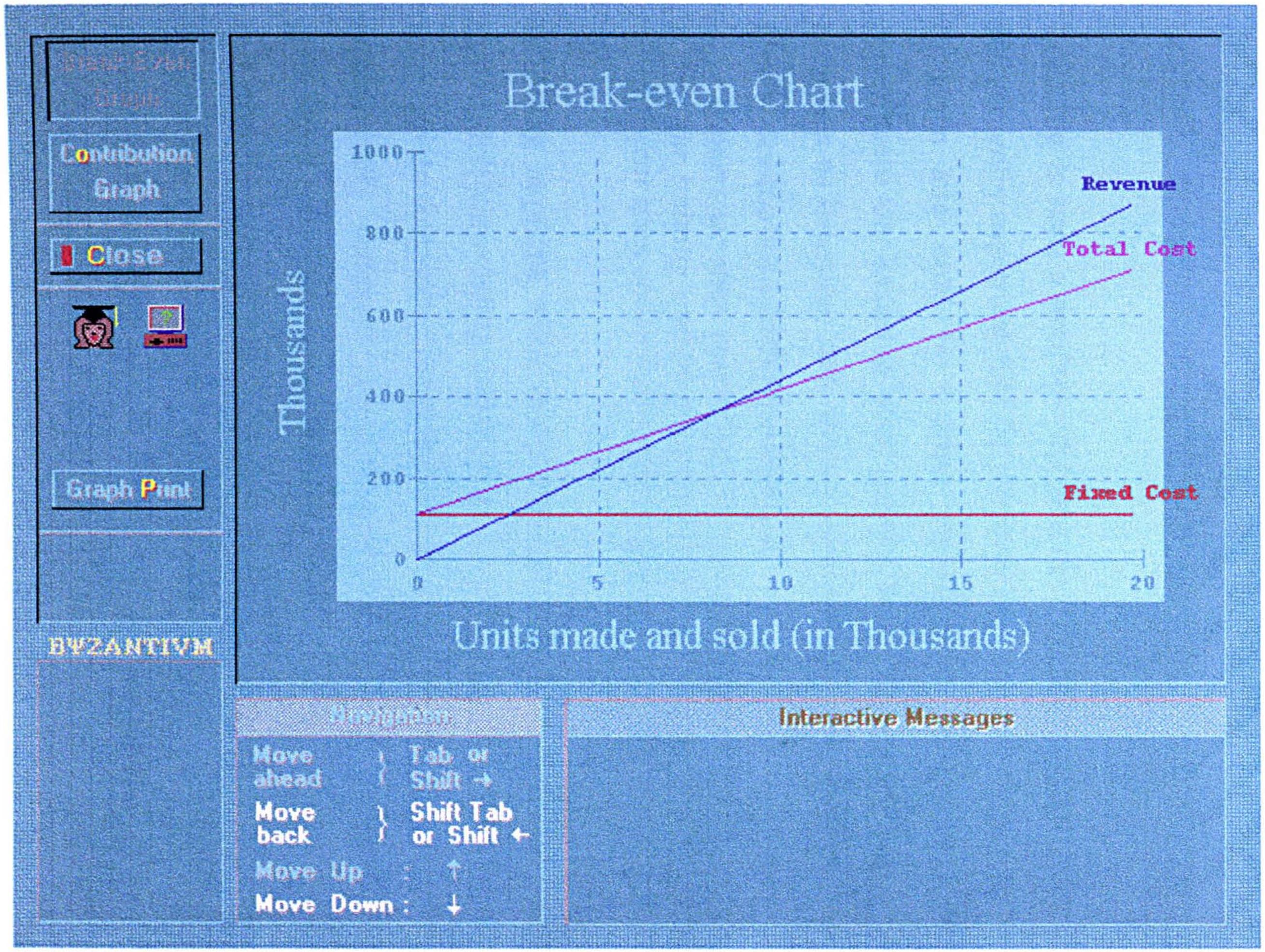

Figure 7. Screen-shot of Graph screen of Marginal Costing package

Theory of Learning Styles

"Learning is not finding out what other people already know but solving our own problems for our own purposes by questioning, thinking and testing until the solution is part of our life."

Prof. Charles Handy

(Osborne, 1994)

Learning is such a fundamental process that many people take it for granted, conveniently assuming that by the time they are adults they have learned how to learn and need no further assistance in the process. Thus lecturers concentrate on lecturing and assume students are skilled at such learning activities as listening, note taking, researching, essay writing and revising. As Honey & Mumford (1986) commented,

> *"Trainers too often assume that learners are empty buckets waiting to be filled by the training method the trainer favours. The fact that the buckets are of different sizes, and/or leak and/or upside down is conveniently overlooked."*

Learning happens when people can demonstrate that they know something they didn't know before (insights and realisations as well as facts) and/or when they can do something they couldn't do before (skills). Learning happens in two substantially different ways. Sometimes we are 'taught' through formal structured activities such as lectures, case studies and books. We also learn from our experiences, often in an unconscious, ill defined way. The word "Learning" is generally associated with the acquisition of facts rather more than with the messier process of learning from day to day experiences. In fact, learning style preferences have implications of all types of learning.

5.1 History of Learning Styles

The subject of how to help people to learn how to learn effectively, has been an active research area over last decade (Mumford, 1982). Individuals have different learning styles which indicate preference for particular learning experiences. Messick et. al. (1976) defined 19 different dimensions. Witkin's (1976) work on *field dependent* and *field independent* cognitive styles concentrates on the differences in the way an individual structures and analyses information. Pask & Scott (1972) identified *holist* and *serialist* strategies in problem solving. Pask argued that the holist and serialist strategies are the manifestations of the underlying differences in the way people approach learning and problem solving. Miller & Parlett (1974) described *cue-consciousness* and identified two distinct group of students. The first group is respective to, and actively seek out, cues and hints from their tutors regarding forthcoming examinations, these they termed as *cue-seeking*; whereas the second, who are less sophisticated strategists and do not pick up on available hints, are termed *cue-deaf*. Dunn et. al. (1977) point out, a person's learning orientation is perhaps the most important determinant of his or her educational attainment. Logically, the greater its congruence with the teaching method used, the greater the chance of success (Allinson et. al., 1988). Consequently, some instruments are available which seek to measure learning styles. In past years, a number of writers have examined the concept of learning styles (Delahaye & Thompson, 1991). Marton & Saljo (1976a) believed that students' approaches to learning tasks could be categorised into two broad areas that they labelled as '*deep approaches*' or '*surface approaches*'. Deep approaches involved an active search for meaning underlying principles, structures that linked together different concepts or ideas and widely applicable techniques. Surface approaches, on the other hand, relied primarily on attempts to memorise course work, treating the material as if different facts and topics were unrelated.

Follow-up studies by Marton & Saljo (1976b), and Svensson (1977), demonstrated that most students were somewhat versatile in their choice of learning approach. Their choice depended on such factors as their interest in the topic, the nature of their academic motivation; the pressure of other demands on their time and energy; the total amount of content in the course; the way in which a task is introduced, and their perceptions of what will be demanded of them in subsequent evaluations or applications of the material. Recent work in the field is more expansive, in that issues in assessment, instruction, personality and evaluation as they relate to learning styles and strategies are comprehensively addressed (Weinstein, Goetz & Alexander, 1988; Ginter et. al., 1989; Green, Snell & Parimanath, 1990).

Training programmes for learning strategies require a reliable and valid means for measuring students' deficits and progress. A number of researchers (Schmeck, Ribich & Ramaniah, 1977; Entwistle, 1981; Weinstein, Schulte & Cascallar, 1985) have provided measures for successful implementation and evaluation of these programmes. Their inventories - *Inventory of Learning Processes*; *Approaches to Studying Inventory*; and *Learning and Study Strategies Inventory* respectively - were designed largely to gain insight into the varied styles and strategies employed by students in their internalisation of cognitive material. However, the model which has stimulated most debate and research is that of Kolb (1976). More recently, Honey & Mumford (1986) developed an instrument, the Learning Style Questionnaire (LSQ) which they claim is based on Kolb's model.

5.2 Kolb's Theory of Learning

Kolb's (1976, 1984) model combines the two bipolar dimensions of cognitive growth distinguished by many psychologists: the active-reflective dimension and the abstract-concrete dimension. The first ranges from direct participation to detached observation. The second ranges from dealing with tangible objects to dealing with theoretical concepts. Kolb used these polar extremes to define a four-stage cycle of learning. It begins with the acquisition of concrete experience (CE). This gives way to reflective observation (RO) on that experience. On this basis, theory building or abstract conceptualisation (AC) occurs. The theory is then put to the test through active experimentation (AE). The cycle thus recommences since the experimentation itself yields new concrete experiences (figure 8).

Each stage of the cycle requires different abilities, and the learner must decide which ones to apply in any situation. Most people, it is argued, tend to be more skilled in some abilities than others. For this reason, they are inclined to favour a particular learning style. Kolb classifies styles according to a four-fold taxonomy based on the two cognitive dimensions. Divergers reflect on specific experiences from a number of different perspectives; Assimilators develop a theoretical framework on the basis of that reflection; Convergers test the theory in practice; and Accommodators use the results of that testing as a basis for new learning.

5.3 The Learning Style Inventory

Kolb measures learning styles by means of a self-description questionnaire, the Learning Style Inventory (LSI) (Kolb, 1976). It comprises nine sets of four words, and requires respondents to rank the words within each set according to how well they characterise their learning orientation. This produces scores relating to each of the four learning abilities CE, RO, AC and AE respectively. Using these, two other scores are computed: AE minus RO, indicating the respondent's position on the active-reflective dimension; and AC minus CE indicating his or her position on the abstract-concrete dimension. Thus it becomes possible to identify a person's preferred learning style.

The utility of the LSI has been called into question, however. At the most basic level, it seems that the scores may not have the normal, continuous distribution usually associated with behavioural variables. Histograms suggest that each scale is, in fact, bimodal (Jervis, 1983). Furthermore, evidence of the reliability of the instrument is mixed. Even allowing the fact that some temporal instability is inevitable owing to situational variations in learning style modes, reported test-retest correlations are moderate (Kolb, 1981).

The construct validity of the LSI is also in dispute. If Kolb's learning theory is correct, it should be possible to confirm the existence of the two underlying dimensions of cognitive growth through factor analysis. Freedman & Stumpf (1978), in a study of business students in the USA, found some support for the model in that LSI items loaded on two bipolar factors in the expected way. The loadings were low, however, the total variance accounted for by the factors being only 20.6 percent, some of which was an artefact of the scoring method. Wilson (1986), in a survey of participants on British management courses, analysed three versions of the LSI (the original inventory and two modified versions), and found that the emergent factors were not even equivalent to those predicted by Kolb. Indeed, the only factor common to all versions was a Doing-Thinking one, but this was not clearly defined and accounted for less than 20 percent of variance at best.

Finally, the face validity of LSI, an important consideration in determining its acceptance by practising managers, appears to be unsatisfactory. James (1980) was not satisfied with the classification of some items; Juch (1980) suggested that certain words within each set are open to different interpretations; and Wilson (1986) admitted to difficulty in understanding a number of the terms used.

Thus, although Kolb's basic model of learning may be regarded as plausible, it would seem that there is a need for a more reliable and valid measure of learning styles than the LSI. Lewis & Margerison (1979) have drawn attention to the Myers-Briggs Type Indicator (Myers, 1979), an instrument based on Jung's theory of psychological types, which assesses the individual's preferred ways of interacting with the environment. Sugarman (1985) argues, however, that in situations in which learning styles are of particular concern, it may be unnecessarily long and sophisticated, its complexity possibly causing confusion.

A more promising alternative may be a measure developed by Honey & Mumford (1986), the Learning Styles Questionnaire (LSQ). Although LSQ has also been criticised by some researchers for its failings in construct validity and has failed on some occasions to show significant correlations among its four learning styles (Goldstein & Bokoras, 1992; Tepper et. al., 1993), this has been the most favoured learning style instrument in literature for evaluation of CAL modules (Allinson & Hayes, 1988, 1990; Hayes & Allinson, 1988; Furnham, 1992). It has also been used for students of business courses (Tepper et. al., 1993).

5.4 Honey and Mumford Approach

The Kolb model is the theoretical background to Honey & Mumford's (1986) Learning Style Questionnaire, which has four styles- Theorist, Activist, Reflector and Pragmatist. The Kolb model describes learning as a continuous process which can be described in an endless loop.

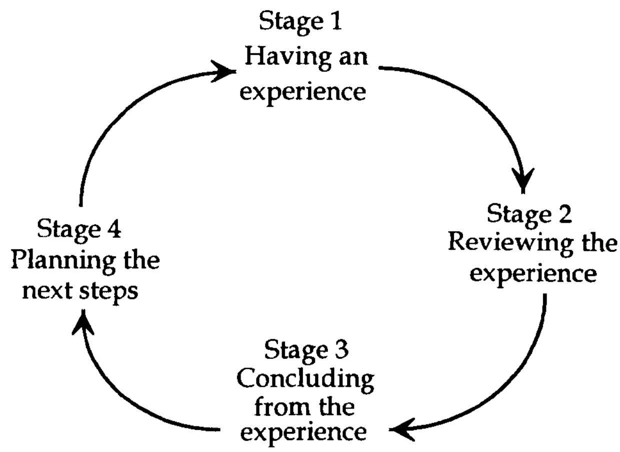

Figure 8. The Kolb Model

A learner can start anywhere on the cycle because each stage feeds into the next. A person could start, for example, at stage 2 by acquiring some information and pondering it before reaching some conclusions, stage 3, and deciding how to apply it, stage 4.

The four stages, experiencing, reviewing, concluding and planning are mutually supportive. None is fully effective as a learning procedure on its own. Each stage plays an equally important part in the total process, though the time spent on each may vary considerably. Most people, however, develop preferences which give them a liking for certain stages over others. The preferences lead to a distortion of the learning process so that greater emphasis is placed on some stages to the detriment of others. Here are some typical examples:-

➢ Preferences for experiencing such that people develop an addiction for activities to the extent that they cannot sit still but have to be rushing around constantly on the go. This results in plenty of experiences and the assumption that having experiences is synonymous with learning from them.

- Preferences for reviewing such that people shy away from first hand experiences and postpone reaching conclusions for as long as possible whilst more data are gathered. This results in an 'analysis to paralysis' tendency with plenty of pondering but little action.
- Preferences for concluding such that people have a compulsion to reach an answer quickly. This results in a tendency to jump to conclusions by circumventing the review stage, where uncertainty and ambiguity are higher. Conclusions, even if they are wrong ones, are comforting things to have.
- Preferences for seizing on an expedient course of action and implementing it with inadequate analysis. This results in a tendency to go for 'quick fixes' by overemphasising the planning and experiencing stages to the detriment of reviewing and concluding.

5.5 Learning Styles

Description of four learning styles described by Honey & Mumford (1986) is as follows (Figure 9):-

- *Activists*

  Activists involve themselves fully and without bias in new experiences. They enjoy the here and now and are happy to be dominated by immediate experiences. They are open-minded, not sceptical, and this tends to make them enthusiastic about anything new. Their philosophy is: "*I'll try anything once*". They tend to act first and consider the consequences afterwards. Their days are filled with activity. They tackle problems by brainstorming. As soon as the excitement from one activity has died down they are busy looking for the next. They tend to thrive on the challenge of new experiences but are bored with implementation and longer term consolidation. They are gregarious people constantly involving themselves with others but, in doing so, they seek to centre all activities around themselves.

- *Reflectors*

  Reflectors like to stand back to ponder experiences and observe them from many different perspectives. They collect data, both first hand and from others, and prefer to think about it thoroughly before coming to any conclusion. The thorough collection and analysis of data about experiences and events is what counts so they tend to postpone reaching definitive conclusions for as long as possible. Their philosophy is to be cautious. They are thoughtful people who like to consider all possible angles and implications before making a move. They prefer to take a back seat in meetings and discussions. They enjoy observing other people in action. They listen to others and get the drift of the discussion before making their own points. They tend to adopt a low profile and have a slightly distant, tolerant, unruffled air about them. When they act it is part of a wide picture which includes the past as well as the present and others' observations as well as their own.

- *Theorists*

  Theorists adapt and integrate observations into complex but logically sound theories. They think problems through in a vertical, step by step, logical way. They assimilate disparate facts into coherent theories. They tend to perfectionists who won't rest easy until things are tidy and fit into a rational scheme. They like to analyse and synthesise. They are keen on basic assumptions, principles, theories, models and systems thinking. Their philosophy prizes rationality and logic. "*If it's logical it's good*". Questions they frequently ask are; "*Does it make sense?*" "*How does this fit with that?*" "*What are basic assumptions?*" They tend to be detached, analytical and dedicated to rational objectivity rather than anything subjective or ambiguous. Their approach to problems is consistently logical. This is their 'mental set' and they rigidly reject anything that doesn't fit with it. They prefer to maximise certainty and feel uncomfortable with subjective judgements, lateral thinking and anything flippant.

- *Pragmatists*

  Pragmatists are keen on trying out ideas, theories and techniques to see if they work in practice. They positively search out new ideas and take the first opportunity to experiment with applications. They are the sort of people who return from management courses brimming with new ideas that they want to try out in practice. They like to get on with things and act quickly and confidently on ideas that attract them. They tend to be impatient with ruminating and open-ended discussions. They are essentially practical, down to earth people who like making practical decisions and solving problems. They respond to problems and opportunities 'as a challenge'. Their philosophy is : "*There is always a better way*" and "*If it works it's good*".

Each style is associated with a stage on the continuous learning cycle. People with Activist preferences, are well equipped for experiencing. People with Reflector approach, with their predilection for mulling over data, are well equipped for Reviewing. People with Theorist preferences, with their need to tidy up and have 'answers', are well equipped for Concluding. Finally, people with Pragmatist preferences, with their liking for things practical, are well equipped for Planning (Honey & Mumford, 1986).

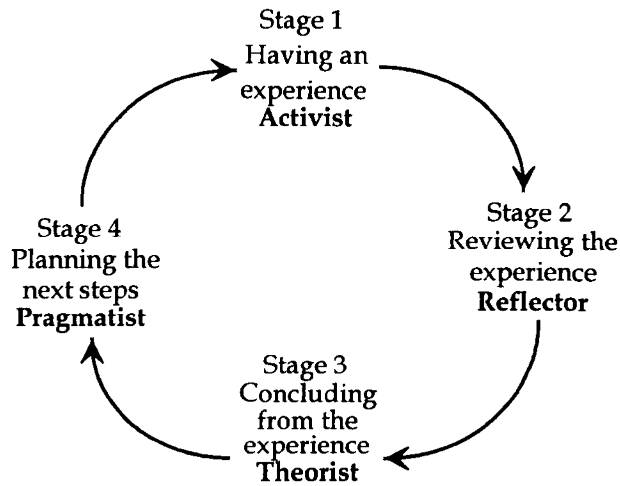

Figure 9. Honey & Mumford Learning Styles

5.6 The Learning Styles Questionnaire

Whilst Honey & Mumford (1986) found Kolb's four-stage learning cycle acceptable, they were less satisfied with the LSI, questioning the use of one-word descriptors as a basis for attributing style, and expressing concern over the face validity of the styles themselves. In developing the LSQ, their approach was to concentrate on observable behaviour rather than the psychological basis for that behaviour, and to identify learning styles that are meaningful to the managerial population. Allinson & Hayes (1988) found LSQ preferable to the LSI for many reasons. First, they found LSQ more capable of actually measuring something. Whereas the LSI apparently has no clear factor structure, the LSQ was able to distinguish similar cognitive dimensions in two independent samples. Second, the distribution of its scores is closer than that of the LSI to what might be expected theoretically. Third, it may be more reliable. Temporal stability coefficient for the LSQ appear to be superior to those reported for the LSI. Finally, it has better face validity. Whilst a number of the LSI items do not inspire confidence as indicators of learning style, the behavioural statements contained in the LSQ at least look as though they measure what they are supposed to be measuring.

5.7 Structure of LSQ

The LSQ comprises 80 statements which respondents are asked to tick or cross indicating broad agreement or disagreement respectively. The aim is to discover general behavioural trends, and no item carries more weight than another. The 80 statements comprise four subsets of 20 randomly-ordered items, each subset measuring a particular learning styles. A copy of LSQ is enclosed in Appendix F for reference.

5.8 Scoring and Interpreting the Questionnaire

Scoring the questionnaire is quite straightforward. The score key 'unscrambles' the items and lists those that probe Activist tendencies, Reflector tendencies and so on. The LSQ is scored by awarding one point for each ticked item and no points for crossed items. Thus the maximum possible score for each learning style is twenty. Raw scores are meaningful only when viewed in the context of normative data. The norms are calculated by analysing the actual scores of people who have completed the questionnaire. Honey & Mumford (1986) analysed the scores of well over one thousand people by 1982 and divided them into five bands:-

- **A** the point at which 10% of the scores are above and 90% are below.
- **B** the point at which 30% of the scores are above and 70% are below.
- **C** the middle 40% of scores with 20% above and 20% below the mean.
- **D** the point at which 70% of the scores are above and 30% are below.
- **E** the point at which 90% of the scores are above and 10% are below.

Each of the five bands arrived at in this way is indicative of a person's learning style preferences:-

- Any scores in the A band indicate a very strong preference since statistically only 10% of the scores fall into this band.
- Scores in the B band indicate strong, but not very strong, preferences.

- Scores in the C band indicate moderate preferences.
- Scores in the D band indicate low preferences.
- Scores in the E band indicate very low preferences since statistically 90% of the scores are above this band.

These norms are referred as general norms because they are based on a large population of scores. Honey & Mumford (1986) commented that the calculations with data collected since 1982 have confirmed the accuracy of the general norms, even though the population upon which they are based had risen to 3500. While the general norms are convenient, norms vary for different occupational groups, and proper occupational norms should be used for more precise results. Honey & Mumford (1986) have provided occupational norms for seventeen different groups.

In this thesis, the subjective views of students are examined for the evaluation of in-house developed CAL packages, the occupational norms for students are used as to analyse the learning styles. These norms are as shown in table 4.

	Band A	Band B	Band C	Band D	Band E
Activist	17-20	15-16	10-14	7-9	0-6
Reflector	19-20	17-28	13-16	10-12	0-9
Theorist	16-20	14-15	9-13	6-8	0-5
Pragmatist	16-20	14-15	10-13	8-9	0-7

Table 4: Occupational norms for students (Honey & Mumford, 1986)

5.9 Use of Learning Styles

Learning Style Questionnaire has been used successfully in many different countries and has given evidence of the reliability and validity of the instrument (Allinson & Hayes, 1988, 1990; Hayes & Allinson, 1988). This has resulted in improved learning process by altering the course structure slightly for individuals according to their learning preferences.

Tepper et. al. (1993) compared LSQ with their own learning style measurement instrument, called Problem Solving Style Questionnaire or PSSQ. 14-item PSSQ had two 7-item subscales (abstract conceptualisation/concrete experience and active experimentation/reflective observation). They used both PSSQ and LSQ on a sample of 227 undergraduate students enrolled in business courses at a large, public university located in the south. Approximately 62% of the respondents were male and their median age was 21 years old. They found that LSQ Reflector correlated significantly with LSQ Activist and LSQ Theorist, but did not correlate significantly with LSQ Pragmatist.

Furnham (1992) used LSQ to compare it with Whetten & Cameron (1984) Cognitive Style Instrument (CST) and Kolb (1981) Learning Style Inventory (LSI). They used LSQ on 60 students of which 40 were female. Their mean age was 22.8 and the range 18-30 years (SD 3.31 years). All students were volunteers. The analysis showed the clear evidence of the existence of four learning styles defined in Learning Style Questionnaire.

Goldstein & Bokoras (1992) also compared LSQ with Kolb (1985) Learning Style Inventory. The sample of students consisted of 44 undergraduate and graduate students and faculty from Central Connecticut State University and Manchester Community College. Subjects were asked to complete both of the questionnaires. Subsequently, all subjects were given written feedback sheets indicating their scores on each instrument along with information explaining how to interpret their scores. The analysis did not show any dramatic magnitudes of correlations.

5.10 Conclusion

According to Mumford (1987), Learning Styles Questionnaire results and an understanding of their meaning can be used in the selection of individuals for particular programmes. Just as some individuals are heavily dominated by one learning style, or are particularly weak in one style, so some learning activities are dominated by explicit or implicit assumptions about learning styles. Generally courses are based on the learning styles of course runners not the learners. By reviewing the literature about the programme and preferably by discussion with the course runners, it is possible to establish whether the nature of the course suits the preferred learning style of the individual. The learning styles can also be used to help learners respond to those aspects which are foreign to their preferred learning style.

Mumford (1987) recognised that unless the styles of learning and learning methods are assessed together, the likelihood of effective learning is much diminished and owes more to luck than we should allow.

This chapter reviewed various learning style measurement instruments and their efficacy in the evaluation of CAL. Honey & Mumford Learning Style Questionnaire was found to be the most favoured method in the literature despite some of its failings. This instrument has also been used in the area of business and therefore considered suitable for the purpose of the evaluation study presented in this thesis. However, in reflecting on the evaluation results in chapter 11, the thesis will also examine the instrument's underlying assumption that it is possible to categorise students according to their learning styles and hence the appropriateness of Honey & Mumford Learning Style Questionnaire for the entry level accountancy students.

The aim of this thesis is to evaluate the in-house accounting CAL modules with the help of students' views towards them. The effect of learning styles of students is an important factor in deciding what type of students are likely to be benefited with these modules. Various different approaches are possible for the development of computer aided learning modules and it is possible that students with different learning styles may get more benefit from one type of modules than from another. Next chapter discusses various approaches used in the development of accounting CAL modules. The history of CAL in accounting is reviewed with the current developments in this field. The next chapter also provides a brief review of various CAL software packages available in the field of accounting.

CAL in the Teaching of Accounting Subjects

The accounting profession is largely based on practical skills and requires the acquisition of basic skills and domain knowledge before applying them to real problems. This chapter discusses the need to learn practical skills in accounting and reviews the tutorial and situated learning approaches. The chapter reviews various efforts to develop computer aided learning material in accounting and provides an overview of in-house adopted approach for intelligent tutoring.

6.1 The Accounting Profession

Jamous and Peloille (1970) argued that professional knowledge includes (a) technical elements that can be mastered and communicated in the form of rules, and (b) indeterminate elements that escape rules and are attributed to the 'virtualities' of the producers. The professional knowledge can therefore be seen as made up of the technical skills and judgement. While technical skills can relatively easily be acquired and are sometimes hived off to technicians (e.g. Accounting Technicians), Perks (1993) observed,

> *"Professionals who lack effective technical skills are in danger of being seen on a par with priests or witch doctors ... meaning that the professional has no effective answer to particular problems..."*.

This is especially the case for Accounting profession that arose from practical skills. Hines (1989) saw professionalisation in accountancy as a social process by which members aspiring to professionalisation laid claim to general quantities such as common sense, diligence, respectability, honesty, independence, ... penmanship, arithmetic and calculation rather than a formal body of knowledge. The search for a formal body of knowledge came later as a defence of its professional status and privileges (Perks, 1993).

While accountancy has now become an umbrella term covering many specialised branches of knowledge and practice and new areas of specialisation, for example, Environmental Accounting are emerging, at its core lies the skills of recording, organising, analysing, interpreting and presenting information. Those who study introductory accounting topics can subsequently choose any accountancy related career such as Accounting Technician, Financial Accountant, Management Accountant, Systems Accountant, Public Accountant, Auditor or Taxation Practitioner in diverse commercial, industrial, public service, charitable and professional environments. Under a modular scheme of studies, even the students who might not want an accountancy related career may want to learn about the basics of accounting and many students are increasingly doing so, for instance, combining Accounting minor to Chemistry major.

6.2 Acquisition of Accounting Skills

Considering the predominantly practical dimension of the accountancy profession, it should be considered whether learning in an academic institution is suitable for those aspiring to the profession or should apprenticeship in the form of an articleship be regarded as the best form of training. The situated learning school of thought would favour the apprenticeship. Brown, Collins & Duguid (1989a) observed,

> *"All knowledge, is ... like language. Its constituents parts index the world and so are inextricably a product of the activity and situations in which they are produced. A concept, for example, will continually evolve with each new occasion of use, because new situations, negotiations and activities inevitably recast it in a new, more densely textured form. So a concept, like the meaning of a word, is always under construction. This would also appear to be true of apparently well-defined, abstract technical concepts. Even these are not wholly definable and defy categorical description; part of their meaning is always inherited from the context of use."*

They further noted,

> *"To explore the idea that concepts are both situated and progressively developed through activity, we should abandon any notion that they are abstract, self-contained entities. Instead, it may be more useful to consider conceptual knowledge as, in some ways, similar to a set of tools. Tools share several significant features with knowledge: They can only be fully understood through use, and using them entails both changing the user's view of the world and adopting the belief system of the culture in which they are used."*

While these ideas cannot be completely refuted, the practical problems need to be considered. If learning can only occur in the context of use, students will need to prematurely commit themselves to a career and train in the workplace. Even then they would have to rely on some basic skills learnt at School. Unless the idea is to train from infancy onwards, some division of labour has to be accepted in the process of education. The apprenticeship argument also fails to satisfactorily address a change from profession to industry or even within a discipline, say, from an Engineering firm to a Chemical Plant in case of Management Accounting. All these changes involve some new learning and a lot of adaptation but there is a core of basic skills and domain knowledge that transfers and enables faster re-orientation for an accountant with a different background then, say, a butcher who wants a change in occupation.

This is not to deny that there are aspects of learning that are strongly situated or that learning progressively develops through activity, however, there is a need to consider that there may be many aspects of learning, especially the core skills, that may be relatively weakly situated and more general in nature. The activity for progressing this type of learning, therefore, need not be carried out in actual work environment in the first instance. The actual work environment can later provide the necessary context for adaptation, enrichment and refinement. The type, extent and duration of work experience required, in addition to qualifying examinations, by some accountancy bodies like the Management Accountants, is to satisfy them that a student has had a sufficient breadth and depth of such enriching experience to qualify for the professional membership. On the other hand, auditors consider the essential skills for auditing to be strongly situated and insist on articleship in a professional firm. It is however debatable whether an auditor trained in commercial environment can adequately audit in an industrial environment without any re-orientation and therefore the insistence on articleship may also be interpreted as providing a source of cheap labour.

Thus, while recognising that learning is progressive and a life-long process and that some aspects of learning are strongly situated, there does not appear any need to take an extreme view that all learning cannot be decomposed and decontextualised. As Anderson et. al. (1996) observe,

> *"It does not require recondite research to demonstrate the near-decomposability of human tasks. Every page of a good cookbook contains examples of assumed component procedures as do the how-to books .... Fortunately for us human beings, with our very limited short-term memories, the workings of each component can be understood without simultaneous awareness of the details of all the other components."*

They also observe,

> *"While some context will often be required to assess a component, there are always bounds on how complex such a context need be. It is a well-documented fact of human cognition that large tasks decompose into nearly independent subtasks (Simon, 1981; Card, Moran & Newell, 1983), so that only the context of the appropriate subtask is needed to study its components."*

They claim that it is better to train independent parts of a task separately because fewer cognitive resources will then be required for performance, thereby reserving adequate capacity for learning. This approach is of particular relevance to the introductory students.

The possibility of decomposing a large task and the need only for the context of the sub task in learning these components, makes it possible to learn from much simple interactions at the component level. It should be possible to move the procedural aspects of any tasks to a computer application and similarly it should be possible to learn these aspects by interacting with a computer application. If the computer application provides intelligent tutoring and feedback to each individual student, it should be possible to both teach procedural skills and facilitate the acquisition of conceptual knowledge.

Currently too much of the teacher student interaction is used up for the mechanical aspects of the subject matter, rather than on the process of applying accounting concepts and techniques. Williams & Newton-Ingham (1994) observed that,

> *"a great deal of courseware that has been developed for tertiary education in the accounting and business area has been driven by the development of technology not by fundamental educational or subject based concepts."*

Bhaskar (1982) predicted that the impact of computers in curricula would fundamentally change the theory and teaching environment of accounting. Despite the enormous publicity given to the silicon chip, microprocessors and the technical revolution, the accounting profession seems to be adopting a somewhat reticent stance. According to Gallagher & Letza (1991), the use of CAL to assist the teaching of accounting may benefit both students in higher education, and those studying towards professional exams.

6.3 Historical Developments

Early examples of testing the effectiveness of computer aided learning can be found in the accounting education literature in the USA. McKeown (1976) reported on the development of a large CAL project dealing with (1) transaction processing, the closing of accounts process, investments and fixed assets and (2) cost accounting topics, capital budgeting and funds flow analysis. At the University of Illinois, McKeown had available some 800 special purpose computer terminals in 1972. These terminals and the whole system were called

PLATO IV (Programmed Logic for Automated Teaching Operations). The terminals had many special features and were designed for CAL use. McKeown tested the effectiveness of this teaching technique by means of control group and experimental group of 375 students each. Control group students were given a written examination whereas experimental group students were subject to on-line examination. Both examinations were multiple choice examinations. The studies strongly supported the hypothesis that the CAL students could be brought to a good performance level with less class time and significantly less total student time using PLATO as compared to conventional teaching.

McKeown's paper was considered to be the seminal work by Bhaskar (1983) but Collins (1983) suggested that the testing was fundamentally flawed. Collins (1983) argued that three of the four tests did not yield statistically significant results, experimental design was not fully tested and no references were made to the voluminous literature on education research. What is clear is that McKeown was the first attempt to test the effectiveness of Computer Aided Learning in accounting on an undergraduate programme. The difficulties he encountered are repeated throughout the later literature. For example, the problems of scientifically controlling a classroom learning environment in a test situation is still common today (Tonge et. al., 1994).

A second paper concerning the PLATO IV systems was that by Burton et. al. (1978) and it dealt with the generation and administration of examinations on interactive computer systems. The results of their experiments indicated that the measurements of student achievement obtained by CAL are consistent with those which are obtained through conventional means, and that the security of a CAL examination appeared to be good in that the 'same' examination could be administered at different times without significantly affecting the scores of students who took the examination later. McKeown (1976) commented on the ease with which 500 students could be simultaneously marked; notably, two minutes after the last student completed the examination, the examiners could be provided with the mean, standard deviation and the individual grades.

Er & Ng (1989) identified seven areas of IT usage in accounting education that included CAL. They did not consider CAL to be effective due to the high cost of the local research effort required but suggested that it was a development with potential to support weaker students with remedial lessons. At that point CAL was not sufficiently advanced to offer the real learning flexibility that teachers require in an independent learning situation. However an important opportunity was identified: "*a research problem in this area is how to program a computer to understand the mistakes made by students so that relevant remedial lessons can be automatically generated to suit the needs of individual students*" (Er & Ng, 1989).

Reaction to and interaction with users is clearly an important attribute for CAL if it is to be a responsive and effective primary learning source. Evaluations/Research must address this issue. Next section reviews the steps in the current development trends of CAL in accounting.

6.4 Current Development Trends in the CAL Uses in Accounting

Increasing number of students pursuing business studies as well as the trend of embedding business disciplines, especially accounting, in science and engineering degrees, has prompted implementation of concepts like *(1) Open Learning,* where the learners are given relatively free access to learning material on a self-paced basis (Hendley & Jurascheck, 1992) and there are no defined and delivered courses leading to qualifications (Lewis, 1990); and *(2) Computer Based Learning,* where students learn with the help of computer based tutorials and practice exercises either in booked tutorial sessions or in their own time (Hendley & Jurascheck, 1992). The increasing pressure on the academics, especially at the introductory level of accounting, provides a secondary (but increasingly becoming dominant) purpose to employing computers in accounting education.

The first reaction of many academics is to use computer based learning to rectify the situation. For more than a decade, accounting academics have utilised various software packages in the accounting curriculum, such as spreadsheets e.g. VisiCalc and SuperCalc (Thomas, 1984), Lotus 1-2-3 (Vickers et. al., 1991; Wattam, 1993; Wing-Shing, 1993; Rouse, 1993), and Multiplan (Wu, 1984); databases e.g. dBase III (Thomas, 1984), and dBase III Plus (Shaoul, 1989); expert system shells e.g. Crystal (Sangster & Wilson, 1991), and Payroll Processing Expert System Shell (Dorr et. al., 1988); and accounting packages e.g. SAGE and PEGASUS (Sangster & Wilson, 1991). This trend is encouraged by publishers who provide spreadsheet templates to accompany their textbooks - all standard textbooks used for management accounting now have an accompanying set of manuals and disk with spreadsheet templates, examples include Needles et. al. (1990) and Smith (1990). This practice coupled with the suppliers' pricing policy of providing an incremental licence copy of the general purpose software very cheaply, has seen a growth of spreadsheet based learning in last four years. Whigham & Houston (1995) recently introduced a computer aided learning package MARKUP, which provides learning of interactive appreciation of the principles and applications of average cost plus markup pricing strategies with the help of spreadsheet modules.

The use of spreadsheets has been augmented by tutoring software packages that are designed around multiple choice tests such as IVY software (Holmes, 1991) and Screenwise by Financial Training Company (Nicholson, 1993); authoring packages based on hypertext/hypermedia such as GALLOP (Hoey et. al., 1994); and recently introduced intelligent tutoring systems such as Byzantium (Tonge et. al., 1994) and BITE (Williams et. al., 1993).

There are five distinct types of software packages available in the literature, which are being used in accounting for computer aided learning purposes. These are briefly discussed in the following subsections.

6.4.1 Accounting Software Packages

Though the use of accounting packages had some merit in terms of learning how to set up accounts structure and design coding strategies for efficient information reporting, the increasing sophistication in software design is fast eroding this advantage. The affordable range of accounting software is designed for experts and comes equipped with default structures and interfaces which mimic physical representation e.g. you fill a cheque on the screen and the system debits the creditor and credits appropriate bank account. Examples include ExpenseMaster from BearaTech (O'Kane, 1995) and WinForecast from PASE (Newing, 1995). Looking at the effect of a set of transactions, a student might learn about liquidity, turnover etc. but does not learn accounting. Similar views were expressed in a study conducted by Sangster (1992) who used a commercial accounting package PEER Statements of Standard Accounting Practice (SSAPs) from Systems Dynamics Ltd. - with second year accounting students.

A danger in using this type of approach is that the students become over-reliant on the packages, use them as a blackbox and not having adequate practice, cannot spot any incorrect results when they arise.

6.4.2 Databases

Shaoul (1991) considered the significance and value of a database of a company's actual management accounts to be twofold; firstly it provides students with a realistic set of data for technical exercises as well as for managerial decisions and for comparison with actual outcomes and secondly it provides a structure for simulation by altering basic inputs. Manchester

University has developed such a database from the records of a carpet industry company as a study resource and source of information (Shaoul, 1990). A database relating to a footwear industry was developed at Salford University (Gee, 1988) in order to integrate the teaching methods and materials across several related disciplines as part of the *Computers in Teaching Initiative* (Gershuny & Slater, 1989).

This approach has some major limitations. Firstly it makes learning too specific to a situation. Though it is recognised that learning and cognition are fundamentally situated (Brown et. al., 1989a) and that experience is known in relation to a particular context, the teaching practices that encourage abstraction from experience do not have to subscribe to an epistemology that places knowledge *solely inside heads* (Laurillard, 1993). The carpet or footwear industry database, though it satisfies the argument of Brown et. al. (1989b) against de-contextualising of knowledge by teaching abstractions, fails to provide the multiple contexts to enable students to learn abstractions.

Secondly, using the database for simulation purpose by altering basic inputs may produce imbalanced data. Such a use in itself negates any pedagogic utility of actual data. It would be more productive to provide hypothetical but balanced multiple contexts.

6.4.3 Expert System Shells

The use of expert systems for learning accounting is increasing, both in volume and sophistication (Undergraduate Cost Accounting Expert System, Goldwater & Fogarty, 1993; Insolvency Permit Expert System, Muggridge & Lymer, 1993; Ince, 1988; Humpert & Holley, 1988; Wilson, 1989). A number of researchers have found that accounting students can acquire specialist expert skills by studying the simulation of such skills in practitioner expert systems (Sangster & Wilson, 1991). An expert system could also be written by a lecturer for the use of the students and this might provide a lecturer with a fresh insight into the subject area (King & McAuley, 1991).

Unlike a human expert, however, an expert system cannot initiate a dialogue, constraining the interaction to a student's initiative and persistence. The only way a lecturer can ensure adequate use of such a system is to set coursework based on the knowledge encapsulated in the system, thereby increasing the assessment workload.

Learning and retention is, however, far enhanced if students have to study a topic and its rules and construct an expert system. This raises the problems of learning how to use the expert system shell and the time spent on constructing an expert system. Though expert system shells are relatively easy to learn for very basic expert systems, Sangster & Wilson (1991) found in a study of 65 second year accounting graduates that expert system approach was significantly less efficient in terms of student time, since the students had to practise a large number of problems to get familiar with the system and doing so caused the approach to be over-demanding of student time. They also found that the efficiency suffered even further if a student had to consult more than one expert system in a tutorial, due to the time spent in exiting one system and loading another. In terms of syllabus coverage, therefore, this approach is inefficient and may create distraction by the lectures and tutorials getting out of synchronisation. King & McAulay (1991) developed an expert system for standard costing and found it difficult to accommodate the wide knowledge base necessary to answer standard costing examination questions.

6.4.4 Spreadsheets

Hendry & Green (1994) characterised spreadsheets as having very low requirements for looking ahead (low *premature commitment*); virtually no user-defined abstractions; ability to re-arrange material swiftly (low *viscosity*), but at the price of *hidden dependencies* (viz. say, *c* is based on *b* which is based on *a*; formula in *c* refers to *b* but the connection to *a* isn't apparent without first going to cell *b*) and poor *visibility* (data could be scattered anywhere on the spreadsheet). These characteristics have a bearing on the use of spreadsheet as a learning package.

Low premature commitment and low viscosity encourage quick and dirty approaches to the solution as a student is not forced to plan and execute a well thought out approach. This could easily result in badly laid out data which hinders the learning process in subsequent cycles of simulation.

Though low viscosity permits easy re-arranging of data for an expert, this could still be difficult for a student, especially if any absolute references are used. If the spreadsheet is bigger than a single screen or is badly designed with data clusters spread all round, the poor visibility cancels out any advantage of immediate feedback through automatic recalculation. It would be quite difficult for a student to make sense of changes occurring in different parts of a spreadsheet - only some of them directly visible. Learning is also made very difficult by the hidden dependencies as they obscure the chains of relationships.

Yet another obstacle to learning is the uni-directional structure of a spreadsheet. This means that the dependencies are determined at the time of model creation, for example, if $c=a+b$, then the spreadsheet model will incorporate this formula in cell *c* and as soon as *a* and *b* are available, *c* is calculated. You cannot, in the same model, ask what will be *a* if *b* and *c* are given, though the solution depends on the same relationship. A student might actually erase the formula in cell *c* by directly entering a value.

Learning and retention is improved if students construct the model of the topic studied, but this is inefficient in the same way an expert system is - even more so because the spreadsheet requires the logic to be expressed in its own syntax. It also creates distraction if students get more interested in learning various features of a spreadsheet rather than the topic studied. Spreadsheet templates created by tutors or textbook authors can speed up the process, but they make learning more passive, the only excitement being the opportunity of watching different outcomes based on input variations. The problems of hidden dependencies and low visibility become especially significant in the case of ready-made templates. To a novice student, who cannot fully understand the spreadsheet syntax, there is no way to easily understand the inter-relationships of a model's variables.

Even in the case of models constructed by a student, subsequent use of the model does not reinforce the knowledge of relationships between variables except in a passive manner through observing the outcomes. Spreadsheets are a powerful aid to an accountant's work but they have a very limited scope for learning accounting.

6.4.5 Intelligent Tutoring Systems

Software packages such as accounting software packages, databases, expert system shell and spreadsheets provide problem solving approach but they do not adapt themselves to the need of individual students, as a human tutor does. To provide efficient guidance and tutoring, the software should have some idea of student's cognitive state and should be able to adapt to the situation as student progresses. This phenomenon generates a need for the tutoring software to have some kind of a model of a student which is updated progressively with the student learning. Intelligent Tutoring Systems provide this facility and therefore, are suitable for computer aided learning in accounting. Since the accounting is a procedural and numeric discipline, and is

less 'exception-prone' (generally rules are well established), the technique of intelligent tutoring should be effective in learning accounting.
The concept of intelligent tutoring is not very old in the field of accounting. Recently, some researchers have started taking interest in this area under the Teaching and Learning Training Programme (TLTP) of Higher Education Funding Councils of UK. BITE (Williams, 1993) and BYZANTIUM (Wilkinson-Riddle & Patel, 1993) projects are two known examples of this approach although both projects have different approaches towards interaction with students.

The BITE Project is based on a game based metaphor to aide and draw the students through the learning material (Kaye & Harrop, 1994). The metaphors, when understood well, provide learning at a faster pace because they serve as a 'conduit' for learning directly rather than deriving the knowledge from experience. Pinker (1994) noted "*Metaphors and humour are useful ways to summarise the two mental performances that go into understanding* ...". The BITE Project harnesses both these mental performances by employing '*funny creatures in a plaza*' metaphor. Pinker (1994) has however cautioned "*the metaphor is misleading*". It may unduly emphasise some aspects of what is being learnt. It also suffers from the drawback that people with different cultural backgrounds may draw different meanings from a given metaphor. Given the potential advantages and limitations of metaphors, it will be useful to see the evaluation results of the BITE Project as they become available.

The Byzantium approach does not deliberately employ any metaphor, excepting those inherent in a direct manipulation interface, but benefits from '*Fill in the blanks*' exercises performed at school. Since the screen shows a series of edit boxes, some with values, it is instinctive for students to attempt filling the blank ones. Similarly, the software introduces students to the spreadsheet metaphor by its information layout and focus on relationships. The in-house developed Byzantium modules are described in the next two chapters.

6.5 Brief Review of CAL Software Packages in Accounting

There are many CAL software packages in accounting subjects sighted in the literature (Gallagher & Letza, 1991; Nicholson, 1994a; Nicholson, 1994b; King & Whittaker, 1990; Vickers et. al., 1991). Some of them have been quite popular and are of historic interest. Research work is well in progress and more and more of these packages are coming in the market. This section reviews some of these packages. More details are available in Appendix B. Although there is no clear-cut dividing line, Understand ACCOUNTS, EQL Interactive Taxation Tutor, Stella II, and Financial Management can be considered clearly state of the art as they use latest software and hardware technology and provide advanced graphical and audio interfaces available nowadays. No known intelligent tutoring system is available in accounting domain except the in-house modules which are described in next chapter.

➢ **Accounts Trainer**

(G. J. Wilkinson-Riddle and B. E. Barker; Pitmansoft, 1988)

Topics covered: Introduction to basic double entry bookkeeping, "T" accounts with trial balance and final accounts.

Known Users: De Montfort University and University of Westminster.

The construction is in Basic language and is somewhat dated, and the screens rather cluttered and not always easy to follow. The booklet is set out in simple English. Objectives are set at the beginning of each chapter. Prompts in the software are sequential and slow. The cursor moves according to a set pattern and there is apparently no way to skip to the required cell on the screen, which could prove frustrating as the student's ability improves.

➢ **Financial Accounting for Non Accountants**

(IVY Software Computer Assisted Learning; IVY Software (Holdings) Ltd)

Topics covered: Nature of accounting, the balance sheet and profit and loss account, recording transactions and the trial balance, funds flow statements and ratio analysis.

Known Users: Bradford Management Centre, University of Surrey and Ashridge Management College (for pre-reading in MBA course).

The software is well constructed with a reasonable amount of flexibility enabling the student to repeat elements of the course without having to start again completely; however, there is no facility to move backwards within an element, for example to review the contents of the previous screen.

➢ **PINSTRIPE**

(R. Hornsby, P Thompson and A Ramsdale; Humberside Business School)

Topics Covered: Basic accounting concepts, fundamentals of balance sheet and transfer of balances to profit and loss account or balance sheet with calculation of ratios.

Known Users: Humberside Business School.

The package is easy to use as a book, with facility to exit from the procedure at any stage. Although the software is good for practising the construction of profit and loss account and balance sheet from a trial balance, there are pages full of text presented in complicated language.

➢ **PLATO** **Financial Statements: Structure and Content.**

(G Carr; Control Data, 1985)

Topics covered: Background to, and functions of, financial reporting, introduction to the balance sheet layout, description of assets and their valuation, shares, reserves and long term loans, concept of funds flow as a cause of the change in balance sheet values from one year to the next, and introduction of the accruals (matching) and the realisation concepts. There are practical exercises to get hands-on practice of various concepts.

Known Users: The software is very old and there is currently no known user for this software.

The package needs a reading of manual before attempting anything on computer. Screen layout is effective, there is no cluttering of text on the screen. It is difficult to switch over to different sections of the software as there is no direct access from one part to another parts of software. The details of subject content is too brief to be used this software as main instructional software in curriculum.

➢ **TRAINING PACKAGE** (Currently known as **EQL Bookkeeping Package)**

(Malton P, Gray I and Barker B E; Institute of Chartered Accountants of Scotland, 1989)

Topics covered: Introduction to accounting, double entry in balance sheet, maintenance of "T" accounts, construction of trial balance, double entry in profit and loss account, trading account, cost of sales and gross profit.

Known Users: Nottingham Trent University, University of Loughborough, University of Huddersfield and **all** Scottish HE institutions.

The text is easy to understand and to follow. Colours are used consistently throughout the software. Forward and backward movement is easily possible. Although help is possible at all times, interactive feedback is limited to one standard message.

➢ **QSB+ (Quantitative Systems for Business Plus)**

(Yih-Long Chang and Robert S Sullivan; Prentice-Hall, 1989)

Topics covered: Linear programming; Mixed integer programming; Transportation problems; Network models, critical path analysis and PERT; Dynamic programming; Inventory theory: deterministic and probabilistic; Queuing theory; Decision and probability theory: weighted mean and variance analysis, bayesian analysis, payoff tables, decision trees and Markov process;

Time series forecasting: moving averages, exponential smoothing, linear regression and Winter's model.

Known Users: University of East Anglia.

The software is straight forward to use with little training and provides adequate facilities for editing problems and displaying solutions. Where appropriate the graphical displays present a further insight into the problem being solved. The main disadvantage of this software is its inflexibility in installation. The correction of typing errors is difficult as an immediate mechanism for cancelling the current operation is not always provided.

> **Microcomputer models for Management Decision - Making**

(Terry L Dennis and Laurrie B Dennis; West Publishing, 1988)

Topics covered: Linear programming; Mixed integer and binary programming; Goal programming; Transportation problems; Assignment problems; Network models and PERT; Forecasting: moving averages, exponential smoothing and regression analysis; Simulation: Monte-Carlo, queuing and inventory models; Decision theory; Markov analysis.

Known Users: There is no feedback available on the use of this software in HE institutions.

Software is a set of easy-to-use programs which allow students to solve a number of management decision models. The main drawback is that the software does not provide any graphical views of solutions.

> **Understand ACCOUNTS**

(Produced by Interact; distributed by EQL International Ltd, 1994)

Topics covered: Introduction to accounts, balance sheet, profit and loss account, accounting fundamentals, profitability, cash and liquidity, business and gearing risk, investor ratios.

Known Users: All Scottish HE institutions.

The software is one of the best pieces of CAL software available in the market. It is well presented and challenging. The content is well presented and dealt with in a logical progression. Although the objectives are not explicit in the software itself, the accompanying booklet intended for the tutor or installer does describe the objectives for each module.

➢ **EQL Interactive Taxation Tutor**

(EQL International Ltd, 1994)

Topics covered: Introduction to the UK tax system, personal income tax, Schedule D income tax, capital allowances, corporation tax, capital gains tax and value aided tax.

Known Users: This package is recently released and till now it is in evaluation phase. None of the institutions have put order to purchase the package.

The software is one of the latest piece of software of its publisher, who is well known in the field of CAL packages. The screens of this software are clear, interesting and colourful and the material is well presented. Questions are particularly well handled in this package. For non numeric questions there are various interesting ways of responding to the question.

➢ **Stella II Version 3.0 (Windows Version)**

(High Performance Systems Inc., 1995)

Topics covered: Multi-level, hierarchical modelling environment which can be used for financial modelling, management accounting, finance and accounting, strategic management, economics, production and operations management and information systems.

Known Users: This package is recently released and till now it is in evaluation phase.

The objectives of Stella include creation of an environment of active classroom learning, focusing discussion and keeping all students involved in the learning process. No previous knowledge and experience of modelling is required.

Screen layouts are well presented and easy to read. Colours and graphics are sensibly used. Errors in the formula are detected automatically. The software is easy to install and is available for Apple Macintosh, MSDOS and Windows environments.

➢ **Financial Management**

(Interactive Services Ltd., 1995)

Topics covered: A broad coverage of financial management topics.

Known Users: This package is recently released and till now it is in evaluation phase.

The software runs both under DOS and Windows environments. It consists of 12 modules which open with a menu of the lessons contained within them. Lessons are divided into topics, and within each topic, sub-divided into sections. Other features of the software include practice questions, glossary and good illustrations.

➢ **BITE's Pandora Courseware**

(Univ. of East Anglia, 1995)

Topics covered: A broad coverage of business management, accounting and finance topics. The modules covered in Pandora are Performance Reports, Working Capital Control, Budgetary Planning and Control, Absorption versus Variable Costing, Cost-Volume-Profit / Break-even Analysis, and Investment Appraisal.

Known Users: This package is recently released and till now it is in evaluation phase.

Pandora comprises six modules based on the metaphor of a shopping and business plaza. The courseware has high quality colour graphics and is designed to enhance the quality, thoroughness and speed of student learning by enabling students to manage their own learning experiences. The courseware is user friendly and does not assume students having any prior computer or subject knowledge.

6.6 Discussion

The chapter discussed the practical skills that gave rise to the accounting profession and noted the various specialisms covered by the term 'Accounting', and hence various career paths open to a student of introductory accounting. It also discussed issues in the teaching and learning of accounting subjects and the suitability of using computers for accounting education. Historical developments and the current trends in the use of CAL in accounting are reviewed, identifying five distinct types of software used in accounting education.

Out of five types of learning packages, databases and accounting packages have limited benefits in terms of student learning, although they are software tools of accounting discipline by nature. Their knowledge is essential for an accountant, but for an accounting student, they can help only if the student already knows the basic techniques of accounting. Expert system shells and spreadsheets can provide good learning if students construct the expert system and spreadsheet models themselves, but this requires considerable input of time and effort and burdens a student with twin objectives of learning the use of software along with learning the accounting topic. The use of ready-made expert systems and spreadsheet templates make learning relatively passive and student learning depends on a student's initiative and perseverance. Intelligent tutoring systems are the most recent introduction in the field of accounting. Recently, some projects have started in UK under Teaching and Learning Technology Programme (TLTP) with the help of government funding to develop computer aided learning programs, some of which are developing intelligent tutoring systems in the field of accounting. In-house software development is one of the examples of this scheme.

There are many CAL packages available in the field of Financial Accounting, but Management Accounting is still in its infancy. Much of the efforts in accounting CAL developments have attempted to replace the lecture and not the tutorial. A lecturer can easily lecture to a large group of students but in tutorials, where there is a real need of one-to-one interaction, CAL packages can be found very useful. The CAL packages give homogeneity in subject content presented to the student and allow students to keep their own pace of learning. The accounting CAL packages available till now have concentrated on the theoretical part (lecture) rather than the practical/numerical part (tutorial) of the curriculum. Efforts have been made in-house to develop CAL modules which can assist the numerical part of Management Accounting curriculum. The next chapter describes the approach adopted by the Byzantium project and discusses how this approach helps in building up the essential skills while improving the conceptual understanding.

Intelligent Tutoring Tools - In House Approach

Johnson-Laird (1988) explains learning as a relatively permanent change that occurs when, as a result of experience, one is able to do something one could not do before or is able to do it better. One can learn facts, general concepts and skills and the learning can occur in various regimes, for example through trial and error, through following instructions or through imitating an expert. Initially the learners have to be attentive to all the parts of their performance but as they become more practised, they need to monitor only the trickier parts. This increase in automatic performance shows that the brain can do different things in parallel: one part is devoted to the skill whilst another part mediates conscious experience. The philosopher Alfred North Whitehead (1911) attributed human advancement to this ability, observing, "*Civilisation advances by extending the number of important operations which we can perform without thinking about them*".

The ability to carry out automatically as many basic performances as possible, therefore, increases the scope of any discipline's practitioner. The automated procedures ensure that the memory does not get overloaded with only the mechanical aspects of the problem under consideration. Students aspiring to any discipline need competence acquired through practice and the support for this view is reflected in both the argument for apprenticeship as well as in the academic practice of tutorials following a lecture. Whereas the apprenticeship model allows different rate of progression based on the student's learning efficiency over a range of skills, the current tutorial model based on human instructor is time bound and therefore quite constrained in adequately addressing the issue of natural differences among students in their working patterns, preferences and abilities. If computers can be used in an interactive way for the tutorial purpose, they can greatly help in relieving this bottleneck as computer labs can easily be made available for needy students outside the scheduled tutorial sessions.

Wilkinson-Riddle & Patel (1993) noted that in UK Business Schools academics there is no widespread appreciation of the benefits that computer aided learning can bring to them and to their students. There was very little literature on this topic in the UK prior to 1992 (chapter 6 provides a review of available literature) and until July 1992 there was little funding encouragement to develop or install computer aided learning material in UK universities. Within this context Wilkinson-Riddle & Patel (1993) worked for three years to:

- define computer aided learning standards,
- develop a program methodology which can deliver them, and to
- develop a working software model embodying the standards.

They consider that the main aims of using computer aided learning are to:

- Enhance the quality, speed and thoroughness of student learning.
- Make more efficient use of teaching resources, particularly staff time.
- Make numerate disciplines more accessible to non- specialist students and to integrate IT awareness and abilities into courses.
- Improve access to University Education by promoting flexible learning opportunities.

To achieve these aims, Wilkinson-Riddle & Patel (1993) developed prototype "*Human Emulation Software*" to teach Marginal Costing under Management Accounting Education. They considered that, to be capable of assisting students, as a human tutor would, computer aided learning software must have the following characteristics:

- Require no fixed sequence of data entry and be able to accept any suitable data.

- The system should solve any entered problem for itself; rather than reproduce a solution held on a file.
- The system should recognise when a student has taken a valid but different route to a solution and be capable of helping the student along that different route.
- Be able to generate numerical examples based on random values so the students can practise on their own.
- Be able to operate in a test mode where it does not offer any guidance to the student.
- The system should allow key values to be held in case of narrative examples so that the two distinct aspects of interpreting data and understanding interrelationships can be tested separately.
- The system should incorporate automatic marking facilities to save staff time and these facilities must apply to randomly generated examples without the need for staff input.
- It should provide utilities like pop up calculators and facilities for reducing keystrokes which ease system use and the entire program should require no prior student computer knowledge.
- The system should be able to graphically represent the numeric relationships involved.

The prototype software employed an applied inference engine, *KNOwledge Storage And Recall Technique* (KNOSART). The engine, along with its interface to the application is called *Specific Domain Applied Artificial Intelligence* (SDAAI) methodology (Wilkinson-Riddle & Patel, 1993). The inference engine is based on the common overlay architecture suggested by various researchers (Barr & Feigenbaum, 1982; Bonnet, 1985; Wenger, 1988; Burns & Capps, 1988; Mandl & Lesgold, 1988) and consists of four basic components, namely, expert knowledge module, student model, tutoring module, and user interface module. The inference engine is considerably *domain independent* and has been used later in TLTP Byzantium products. The structure of engine is explained in section 7.4.

The prototype software provides the clear mathematical relationships of marginal costing by creating an entirely interactive approach for problem setting and problem solving. The prototype software was evaluated at various institutions in UK and results were compiled in Tonge et. al. (1993), Dungworth & Russell (1994), Tonge et. al. (1994).

The success of the prototype resulted in a consortium of six universities led by De Montfort University (TLTP Byzantium) to bid for funding under the Teaching and Learning Technology Programme (TLTP) of the Higher Education Funding Councils of United Kingdom. Funding in excess of £700,000 was secured, starting from October 1993 spreaded over 3 years. Development of six modules was proposed with four modules in management accounting and two in financial accounting. A Teaching Support Tool (TST) for marking and reporting of scores was also proposed.

7.1 The Byzantium Model

An approach to computer integration in learning environment is outlined below as the Byzantium model of Computer Integrated Learning Environment (CILE):

- to use computers and human for what they are (*currently*) good at;
- to employ useful software tools within the overall learning environment consisting of human teachers and educational technologies;
- to add applied intelligence to the software tools to provide a degree of support to students, enabling them to work by themselves;
- to let intelligent tutoring evolve from practically useful applications, in a bottom up fashion (through vertical and horizontal integration) rather than be designed top-down;

- to understand the economics of the learning environment and be concerned with assessment and course management as they consume substantial human resource - *The system should provide economics of staff time in all areas of the teaching cycle, e.g. exposition, example setting and grading work* (Wilkinson-Riddle & Patel, 1992);
- to appreciate the economics of software production and recognise that tutoring software is a joint cognitive system (Dalal & Kasper, 1994) involving a student, tutoring software and a human teacher whose involvement may be greater or lesser depending on capabilities of the software and the manner of its integration into the curriculum. In such joint cognitive systems, sometimes it is more economic to let the student explore the working of tutoring software than to expend huge effort and expense in trying to design a comprehensive ITS that attempts to comprehend the mental processes of students with different personalities and backgrounds;
- to acknowledge that various educational technologies have their own strengths and therefore a learning environment benefits synergistically from an appropriate use of multiple resources - the converse also being true in the learning environment becoming suboptimal through inappropriate use of the educational technologies.

The Byzantium model of CILE divides the subject area into three distinct knowledge levels (Figure 10):

1. *Introductory application level,* where the formation of a mental map of interrelated conceptual atoms takes place and the student learns the use of basic tools of a subject discipline.
2. *Advanced application level,* where the integration of conceptual objects takes place.
3. *Actual application approximation level,* which attempts to simulate the real world problems in their simplicity and the students learn to correlate introductory tools while dealing with behavioural and environmental

factors. Probabilistic models, which include real life attitudes to risk and uncertainty of situations, start affecting the whole scenario at this level.

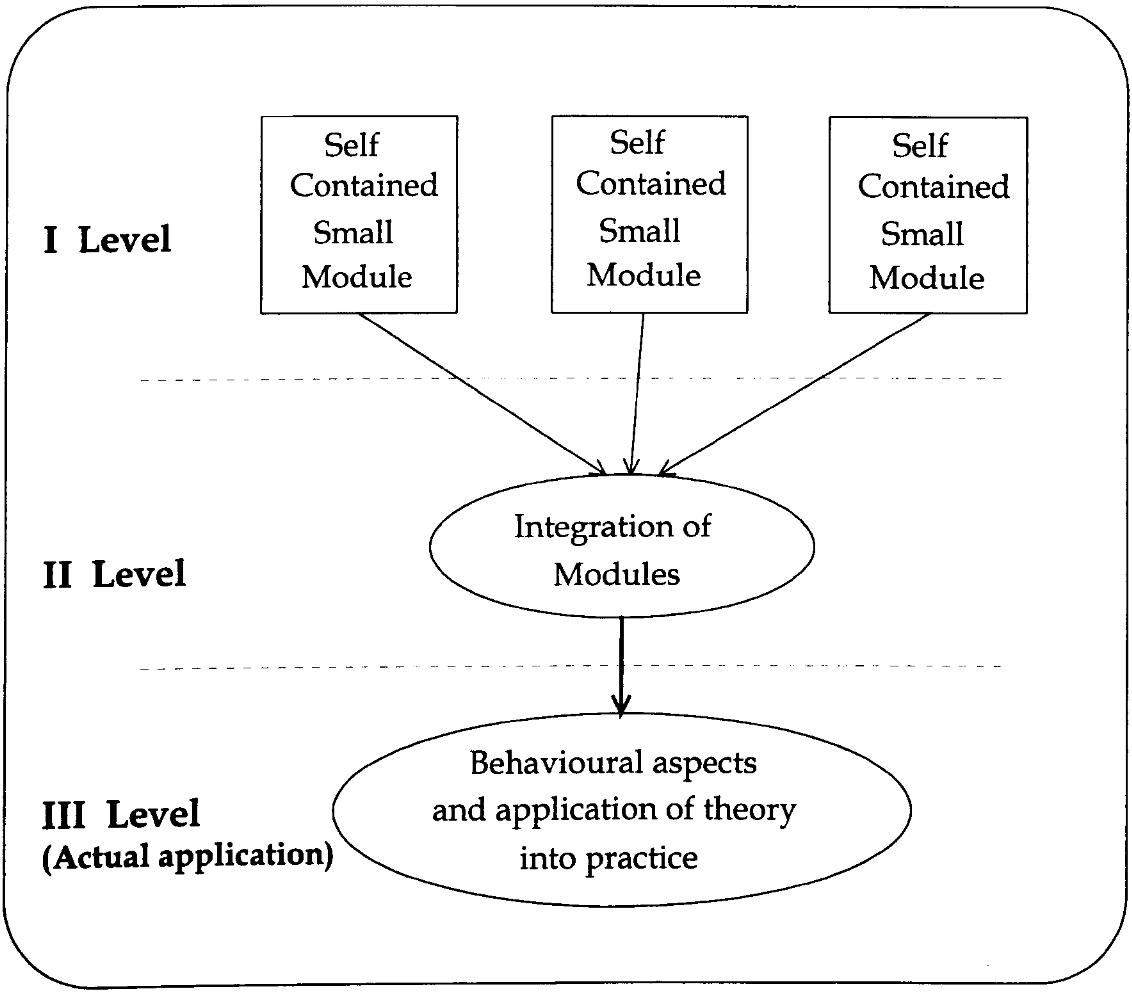

Figure 10. Levels in module integration

The level of complexity increases with the increment in the level. The inter-connection of these three levels is uni-directional and sole purpose of first and second level is to prepare the students for next level.

It is recognised that the integration of the small self-contained *introductory application* level Intelligent Tutoring Tools (ITTs, which are mixed-initiative intelligent tutoring systems, each covering only one elementary topic of curriculum) at the *advance application* level will provide full knowledge of the subject. The integration at *advance application* level assumes that the students have already gained the knowledge contained in ITTs. It does not care how students acquired this knowledge. The process can be traditional teaching or teaching by Computer Aided Learning modules. In this way, the same *advance application* level module can be applied to all in-coming students from different sources.

Current efforts include the software development of first level (Intelligent Tutoring Tools) in the domain of accounting and defining an approach for future development of second and third level (Intelligent Tutoring Tools Network).

7.2 Intelligent Tutoring Tools

TLTP Byzantium has the aim of producing Intelligent Tutoring Tools (ITTs) that extend a lecturer's scope, by horizontally partitioning some of the teaching activities e.g. supervising the development of operational skills, and assigning them to a computer tutor. It is recognised that computers, though quite powerful, are only one of the educational technologies that can be applied in a learning environment. This recognition allows ITTs to be mixed and matched with other technologies (e.g. video) as well as human teachers in various configurations of Computer Integrated Learning Environment (CILE) to suit *class-room, open* and *distance* learning.

7.3 Integrating the ITTs

It is recognised that a usable and substantial ITS is, intrinsically, a complex system. It is envisaged that when sufficient ITTs are ready, a second level of integration will be feasible - an Intelligent Tutoring Network (ITTN). This integration will occur in two directions. *Vertical integration* will allow holding and comparing results of different instances of an ITT, e.g. comparing four different investment proposals; adding behavioural considerations like attitude to risk. *Horizontal integration* will allow use of multiple tools to solve a given problem viz. using marginal costing ITT to provide contribution figures, which are sought to be maximised in a linear programming ITT.

An individual ITT is, thus, sought to be made as an autonomous entity possessing rudimentary intelligence and then, hopefully, to be connected in a decentralised network enabling emergence of a more sophisticated tutoring system. Even if this vision fails to materialise, the individual ITTs have their own utility and therefore the whole effort cannot be lost.

7.4 Applied Intelligence in ITTs

The ITT is *mixed-initiative* system with an *overlay* type of *applied* inference engine which stores and processes knowledge rules. It is *applied* because it does not learn these rules during the student learning. The architecture of the ITT is shown in figure 11 (Patel & Kinshuk, 1996b). The system learns about a student's understanding of the procedural and conceptual knowledge by inferring the information from the student's interactions with the interface and provides the advice according to the knowledge gained by that student so far. Section 7.4.6 describes how the system infers the information from a student's interactions to update the student model and to provide appropriate tutoring strategies.

7.4.1 Domain Knowledge Representation and Teaching Strategy

The main features of the software are:

a) **Knowledge base** containing the conceptual rules and information about how to process those rules. The knowledge base is divided into two sections (Patel & Kinshuk, 1996c):

   i. ***Common didactic knowledge,*** which is based on the variables, their relationships and tolerances. Since Accounting is a procedural discipline and requires sequential processing of rules, it stores procedural knowledge. It also contains a rule precedence hierarchy to ensure guidance to the student by the simplest possible rule where a solution is possible in more than one ways.

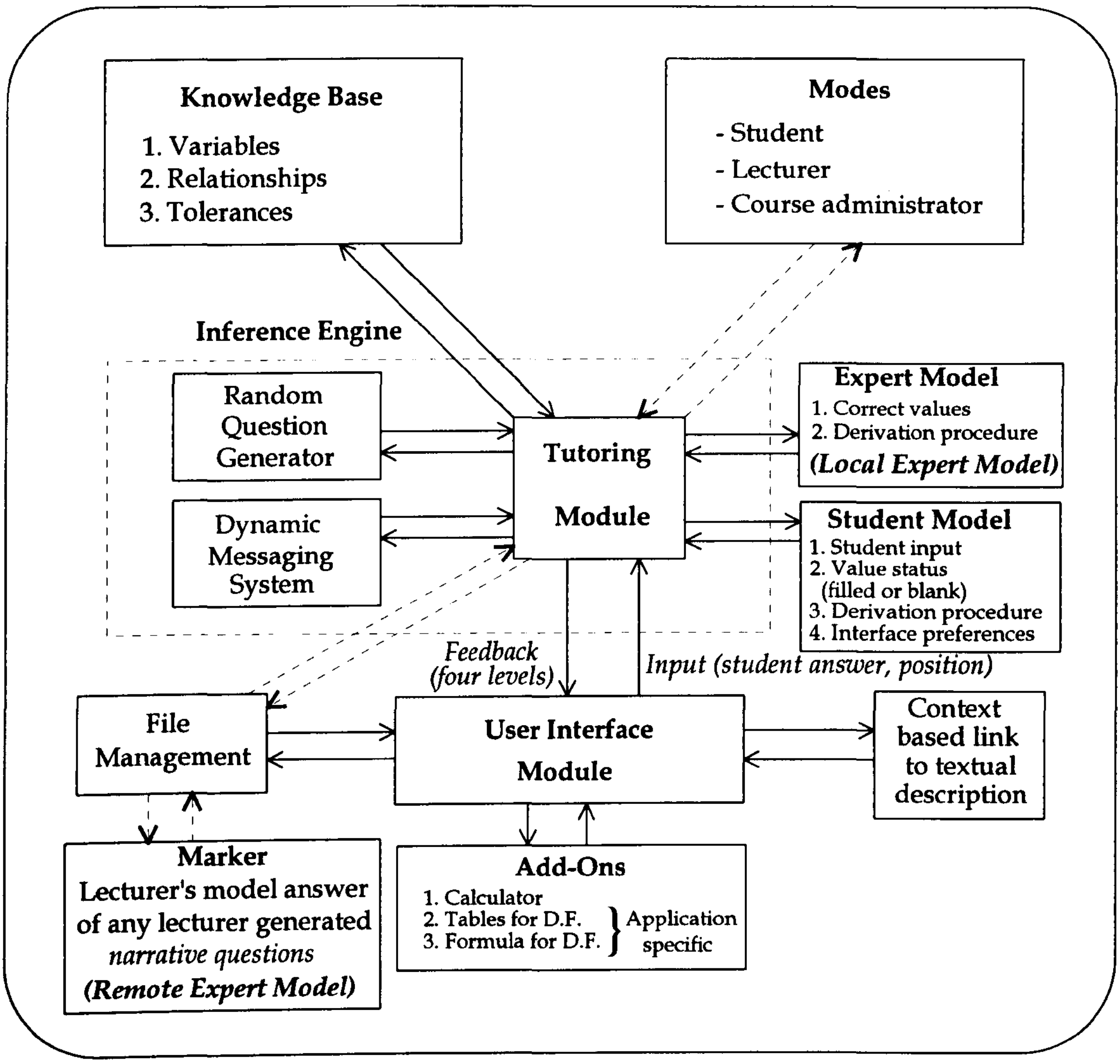

Figure 11. Architecture of an Intelligent Tutoring Tool (Patel & Kinshuk, 1996b)

A partial view of the structure of common didactic knowledge is shown below:

```
Var_desc[...] = Description of variables
Operator[...] = Description of operators
Relationship_type[...]  =  Type  of  possible  relationships  for
          calculation of each variable
Variable[...]_procedure = location of relationships to be used in
          calculation of a particular variable
First_var_location[...]  =  location  of  first  variable  for  a
          particular relationship
Last_var_location[...]  =  location  of  last  variable  for  a
          particular relationship
Tolerance[...] = tolerances to be used for a particular variable
          for a particular relationship
```

ii. ***Domain didactic knowledge,*** which contains conditional rules of the particular domain of ITT. These rules are not generalised and therefore applied through specific logical chain processes. For example, in 'Capital Investment Appraisal' ITT, a student is guided to the correct Internal Rate of Return through an iterative process supported by a procedure contained in this part of knowledge base.

Since the operators can be defined, the knowledge representation architecture allows most of the domain didactic knowledge also to be held in the form of the common didactic knowledge. This would, however, produce convoluted handling mechanism at times and make the code less reusable. In practice, a range of operators are kept constant over all the ITTs and these are supplemented by a small range of domain specific operators that are used frequently within the domain and benefit from the localised generalisation. For example, operators *Plus, Minus, Multiplication,* and *Division* are defined as operator numbers 1, 2, 3, and 4 in common didactic knowledge base. In Capital Investment Appraisal module, operator number 5 is "add column values up to and including" which is specific to this module and hence, is defined in domain didactic knowledge base.

b) **Student model** which records a student's progress towards a complete solution. The student model is explained in detail in section 7.5.1;

c) **Expert model,** which is linked to the knowledge rules and records correct outcomes and how they were derived by the system. *Local expert model* contains correct answer and its derivation procedure. In case of narrative questions, the *local expert model* contains the solution derived from the independent variables which have been put in by the student after interpreting the narrative question. Whereas the *remote expert model* is based on the data provided by the lecturer and therefore contains correct solution. The integrated working of these two models is explained in section 7.4.8;

d) **Tutoring module**, linking student to the various parts of engine through user interface, to advise the student on the basis of work done so far. It allows a student to adopt a different route to the solution than the expert model;

e) **User interface module**, which provides interaction among user and various parts of an ITT. This is explained in section 7.4.5.

f) **Level Selector** which determines the functionality available to user through the interface. The levels are:

i) **Student level** enabling the use of the ITTs for learning without being able to create examples for others to use.

ii) **Lecturer level** allowing users to create examples or templates for use of the students and to save model answers used by the Marker (see section 7.6). Marking of students' assignment work is also possible at this level, but global file management tasks are not available. A unique option of "***auto-solve***" is provided at this level where system solves whatever it can as each independent variables is input and shows the partial solution on screen to allow adjusting of inputs to obtain a desired scenario.

iii) **Administrator level** at which the global management of marking schemes and students' data is possible. This functionality was subsequently added in the Marker to enable it to provide summarised results over a number of ITTs.

7.4.2 Tutoring Mechanism of Inference Engine

A flowchart of tutoring mechanism of an ITT's inference engine is shown in figure 12.

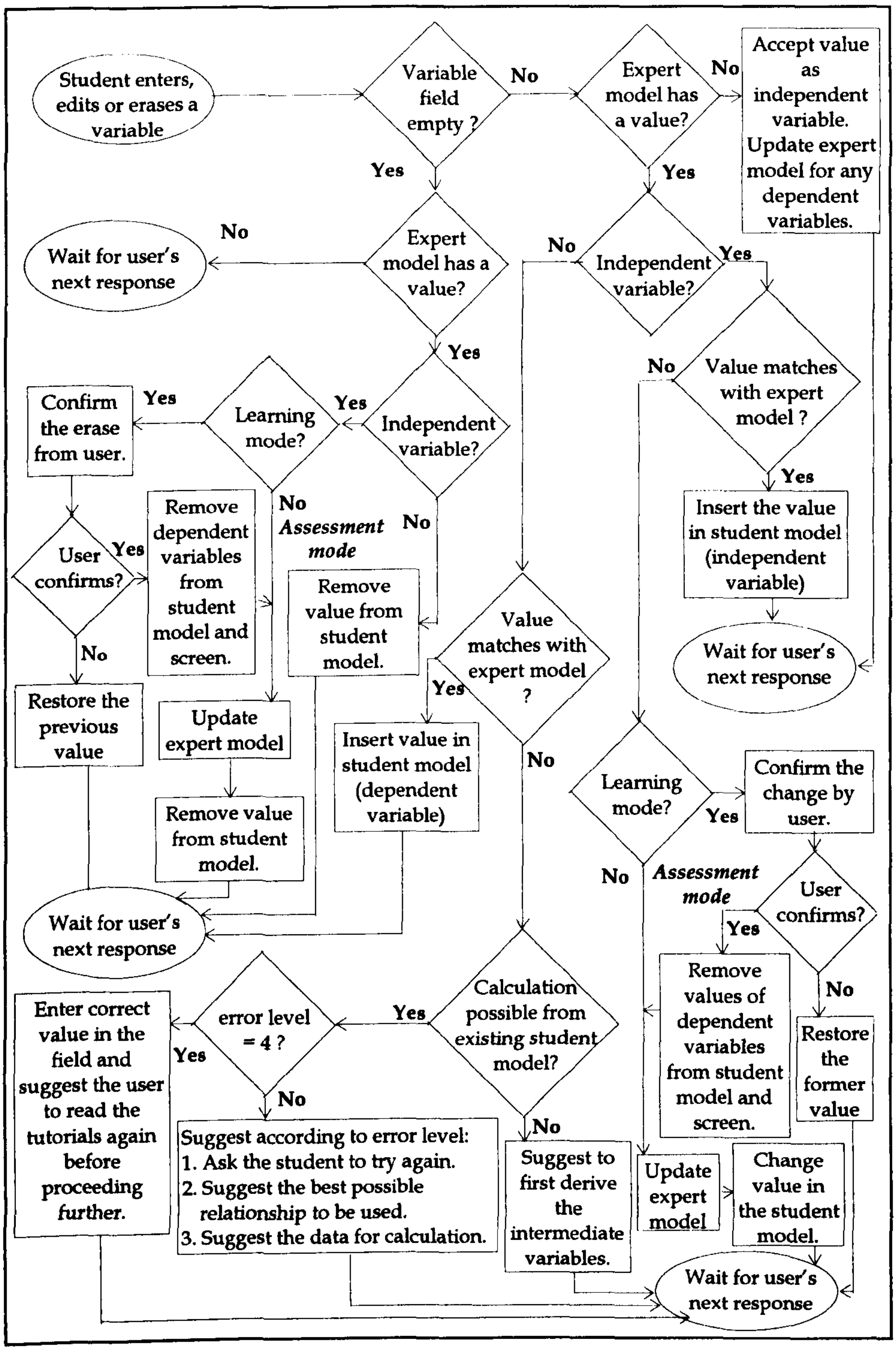

Figure 12. Tutoring mechanism of an ITT

When student enters, edits or erases a value of a variable on screen, the interface send the value and position of the variable to the inference engine.

If student has erased the value, and the variable is dependent, then the engine removes the value from the student model and returns control to the interface and waits for the user's next response.

If the variable is independent, then the engine updates both, expert and student model for all dependent variables and also updates the screen after user confirmation. User confirmation is necessary because change in independent variable affects all dependent variables. This situation does not arise in randomly generated questions where all independent variables are given to the student in read-only mode. In assessment mode, the engine returns control to interface without modifying any dependent variables either in student model or on screen leaving them to the user's discretion.

If the user has entered or edited a variable, and the variable is not empty (user did not erase the value), then, in case of independent variable, if the expert model is empty, the engine updates the expert model for any dependent variables and returns the control to interface. This situation arises only when users enter their own questions. Interpreted question is an example of this mode.

If the expert model has the solution, then after comparing the student input with expert model, the engine updates the expert model, student model and the screen appropriately and provides the dynamic feedback according to the modes, user status and error level. Error level increases with each wrong input of the user for a particular variable. If student has filled wrong input for a variable consecutively four times, the engine enters the correct value of that variable in student model and on the screen, and suggests the user to read the corresponding tutorials again before proceeding further. At this stage, user may abandon the exercise, and go back to tutorials or may proceed further for calculation of remaining variables. Feedback is explained in section 7.4.7.

7.4.3 Enhanced Features of Inference Engine

The Inference Engine contains some enhanced functionality such as:

a) **Random Question Generator** which randomly generates and solves the questions, whenever the tutoring module requires it to do so. It randomly picks variables and assigns random values within specified limits and then derives the solution by applying its knowledge rules. Thus, an ITT *need not contain any data bank* but a lecturer can create a bank of favourite questions to be used by the students. Once the question is generated, Random Question Generator passes the question to user through the interface.

b) **Dynamic Messaging System** which generates dynamically the message strings for the feedback on the basis of information received from tutoring module and passes it to user through the interface. The feedback provided by the ITTs is explained in section 7.4.7.

7.4.4 Dynamic and Adaptive Teaching Strategy

As a part of teaching strategy, considerable scaffolding is given to students at beginners level in the form of help. The system adapts to the progress of student and increases the level of complexity when the student has shown considerable expertise at one level. For example:

a) If student calculates correct Net Present Value, tutoring module tells the interface to provide button for access of Internal Rate of Return screen.

b) If student calculates correct Internal Rate of Return, tutoring module tells the interface to provide button for access of IRR Graph screen.

7.4.5 User Interface Module

The user interface module allows the user to interact with various parts of the ITT. It includes various features, such as:

a) Context based help link to textual description of topics

b) File management

c) Add-ons

i) General functionalities, such as Calculator etc.

ii) Application specific, such as Tables and Formula interfaces for discount factors' calculation in Capital Investment Appraisal module.

7.4.6 Tutoring Strategy

According to the terminology used by Self (1988), *corrective*, *elaborative* and *evaluative* aspects of student model are used in Byzantium. The advice given by the software to the student is based upon the extent to which that student has been able to solve that problem so far and, in view of that, what that student should best do next.

The *corrective* behaviour of the software tells the students whenever they make mistakes in calculations. If they make the mistake a second time, *elaborative* behaviour comes into action, which suggests the various relationships by which certain value can be calculated. The suggestion depends on what the student has done so far. If enough information is not available on the screen to calculate certain value, the software guides the student to fill other values before attempting to fill that particular value. The *evaluative* aspect of software is featured in ***Assignment mode,*** which is designed to test the gain in student's knowledge.

Diagnostic, predictive and *strategic* aspects of student model are not considered to any significant degree in the software. The reason is that learning topic has been sliced in so small modules that the modules are the most basic units of that topic. A '*Road to London*' paradigm is adopted which is more concerned with '*what to do next*' rather than '*how did I come here*' (Patel, 1995). Coupled with dynamic feedback which corrects any mistakes immediately, this paradigm keeps on reinforcing the correct connections through the network of inter-relationships and eliminates the need for *diagnostic, predictive* and *strategic* aspects.

The system applies the tutoring strategies mainly through the interface as all of the user's interaction is through the interface, and the system is able to select the tutoring strategies by inferring the information about the user from interactions. The mixed-initiative approach of the modules is also reflected in the interface, and the students can decide at various points in the program how to proceed further. The students can choose their own paths to reach the solution on the basis of their own understanding of what they have learnt. The system then infers from students' interactions the information about students level of understanding in terms of both procedural and conceptual knowledge and tries to identify gaps still remaining in students' grasp of the subject matter. If the student moves along the chosen path without showing any deficiencies in their knowledge, the system infers that the student has correct understanding. If the student finds it difficult to progress further and start making mistakes, the system takes control, and advises the student to go to higher level of granularity where the concepts are deconstructed into their components. At this moment, the mixed-initiative strategy of the system allows the student to take the decision as to whether or not to pursue the finer granularity, or return to the original path way.

The system infers information regarding mis-conceptions in student's knowledge about the subject topic from user interactions. This inference is of three types:

i. Basic mis-conceptions, where the calculation of particular variable is possible from data available on the screen, and the student has used wrong relationships of variables.
ii. Missing conceptions, with the help of mis-conceptions, where the wrong value has been entered for a particular variable, when it was not possible to calculate that variable without calculating intermediate variables. This mis-conception arises due to missing knowledge about intermediate relationships.
iii. If student enters wrong value for some complex variables, which are in themselves an output of another set of variables, the system infers that mis-conception and advises the student to use finer grain knowledge. If the student pursues finer grain knowledge through the interface, the interface then captures the student's mis-conceptions in term of one or more component variables. The system's advice regarding the mis-conceptions in component variables eventually lead to rectification of the mis-conception of complex variable.

For example, in Capital Investment Appraisal module, if the student selects the calculation of discount factors by formula, and enters a value for a discount factor, the systems compares the value with expert model. If the value matches, the system infers that the knowledge of calculating discount factor is within student's knowledge boundaries. But, if the values does not match, the system infers that the student's level of knowledge regarding calculation of discount factor is lower than the current level of teaching, and the system advises the student to use finer grain knowledge as shown in figure 6 (enclosed in chapter 4). If the student still has difficulty in understanding the concept of discount factor, the system provides the detailed explanation of the calculation process in the form of explanation interface accessible through 'explain' button.

7.4.7 Feedback

The tutoring module provides feedback to the student whenever student enters any value. The messages for the feedback are generated by *Dynamic Messaging System* of the engine which generates context sensitive messages to prevent monotony. There are several possibilities:

- **If student input is right**

  The Inference Engine compares the student's answer with the correct answer held in the expert model. If the values match, the tutoring module tells the student, through the interface, using dynamic messaging system that the value is correct and student can proceed further.

- **If student input is wrong**

  (a) ***If value can be calculated from information already provided by the student***

  Inference engine gives four level help:

  (i) Hint that value is incorrect

  (ii) Suggestion of the best possible relationship (*based on student and expert model*)

  (iii) Suggestion of the data for calculation (*based on student and expert model*)

  (iv) Correct answer

  (b) ***If enough information has not been provided by the student to calculate the variable***

  The inference engine suggests the best possible relationship student should use from the expert model and the student's answers so far to calculate corresponding intermediate variable(s).

7.4.8 Testing of Student's Interpretation

In case of an examination type question - *given in a narrative form and requiring a student to interpret data, identify given values and derive a solution,* the lecturer's model answer held on the ***Marker*** TST serves as the overall (*remote*) expert model conveying correct interpretation and correct method while the application (*immediate*) expert model tests correct method, their combination enabling the ***Marker*** to give partial score for an answer based on correct method but a wrong interpretation of given data. This clearly demonstrates emergence of a higher degree of overall intelligence from the connection of the rudimentary intelligence of an ITT to the ***Marker*** TST.

7.4.9 Modes of Operation

The Byzantium ITTs have two main mode of operation. In the ***Interactive mode,*** the software does not let a student enter a wrong value and provides immediate feedback, as found in a student modelling technique known as model tracing (Anderson, 1988) - the difference being that a student's outcome is monitored rather than the process (rule) employed. Such dynamic feedback is considered essential for learning as it prevents *at source* any incorrect mental associations being made by a student. Outcomes are important in any applied numeric discipline and students must develop the ability to judge the correctness of the process i.e. rules applied by looking at the outcomes.

In the ***Assignment mode,*** meant for testing, there is no immediate feedback and the software allows any values to be entered. Feedback on their correctness is given after the work is marked. This is static feedback. As Routen (1992) observes, "*There are advantages with both forms of student monitoring. Static feedback perhaps is less obtrusive ... while dynamic feedback prevents students from making gross errors and getting completely lost.*" An ITT is designed to benefit from both the dynamic and static types of feedback under the two modes.

7.5 Structure of an ITT

Patel (1995) describes the variables in an ITT as empty containers, that are connected in a network of inter-relationships. Any value can be entered in any variable, provided the whole network remains consistent.

Intermediate variables need not be filled in. This approach allows mental calculations and does not force a rigid path to the solution. Provided the value being entered is correct, the ITT assumes the intermediate steps to be well within a student's conceptual knowledge boundaries. If the value is incorrect, then the ITT guides the student in a graded manner suggesting intermediate steps if necessary. This advise is not based upon some hard coded set of instructions. Student model is applied here to improve the student learning.

7.5.1 Student Model in ITT

A student model might include various classes of information including the student's knowledge in the area being taught, their learning styles, the task they are currently engaged in and their preferred forms of user interfaces. The first of these, student knowledge, is particularly important for CAL modules: the student needs to use existing knowledge as a foundation upon which new knowledge can be built. Moreover, misconceptions in existing knowledge can block new learning (Kay, 1991).

Student model of an ITT consists of mainly four components:

i. values entered by a student;
ii. value status, for example, differentiating a zero value because of a blank variable, calculated zero or data not processed due to infinity condition;
iii. status of the variables filled by the students, such as independent variable, dependent variable or variable calculated through observation (for example, Pay Back Year is the year by the end of which initial investment has been fully paid back);

iv. user preferences of *generative* aspects of the interface e.g. in Capital Investment Appraisal ITT, students can choose whether the discount factors are given by the system, filled by them through inspection of 'Tables' (interface provided) or calculated by them using a formula (interface provided).

When the student attempts to calculate a variable on the screen, the information recorded in the first three components of the student model is used to infer the accuracy of the student input, since the tolerances of the results differ if different routes are used to calculate the same value. If the value does not match with the expert model, the feedback is based on the best suitable relationships of variables, which depend on the variables filled by the student so far.

If student attempts some other action than filling the variables, such as, requesting the assistant interfaces, the first three components of the student model are used to decide the best feedback strategy based on the values filled so far. It is to be noted that the student model does not keep a record of all student movements, and hence, is not capable to provide any feedback based on the sequence of the student moves. As explained in the section 7.4.6 of tutoring strategy, the sequence of student moves are not considered necessary in the modules.

The fourth component of the student model, the preferences towards *generative* aspects of the interface, are used to infer the cognitive state of the user, which then helps in deciding the level of understanding of the user. For example, the user who decides to get the discount factor given by the system, is a naive user. If tables are requested for selection of discount factors, the level of understanding is intermediate. But if the user demands the formula interface, it is possible only if user has full understanding of the concepts. If a naive user demands the formula interface, and is unable to calculate the discount factors, system guides the user to go back to the relevant tutorials.

7.5.2 Metaphors

The Byzantium ITT does not deliberately employ any metaphor, excepting those inherent in a direct manipulation interface, but benefits from '*Fill in the blanks*' exercises performed at school. Since the screen shows a series of edit boxes, some with values, it is instinctive for students to attempt filling the blanks one. Similarly, the software introduces students to the spreadsheet metaphor by its information layout and focus on relationships.

7.5.3 Functional Considerations

The Byzantium ITT is designed with the consideration of high visibility, logical decomposition and visible dependencies. It does not provide ready solutions and waits for a student to execute an action before providing a hint. The ITT accepts a correct entry, regardless of the state of student outcomes model, allowing a student to carry out some operations mentally. It forces an intermediate step calculation only in case of an invalid entry - signifying that a student hasn't fully grasped the chain of relationships.

7.6 Marker - A Teaching Support Tool

A Marker has also been developed as a TST to mark students' assignments and to deal with the bulk handling and reporting of those marks. The Marker provides a facility of creating various schemes for marking and penalties can be set for late submissions. Different weights can be assigned for variables within an ITT. The Marker also provides facility for holding a model answer for the questions in a narrative form and a tutor can set the fractional weights given for incorrect answer but correct method.

7.7 Apparent Weaknesses - Are These Real Weaknesses?

The adopted approach of Byzantium model of tutoring system has some weaknesses. Although some of these weaknesses are inherent from the underlying methodology, Byzantium model tries to overcome them to some extent.

7.7.1 Outcomes Orientation

A weakness in the Byzantium approach is the outcomes orientation with respect to a student's work. This creates two types of problems:

- ***Correct outcome through a wrong process*** - a student might get a correct outcome by chance, though employing a wrong process. Consider a subset of a network, where C = A divide by B and that C has no relationship whatsoever with D and E. If given values are: A = 480, B = 16, D = 0.5 and E = 60, a student under false impression that C is a product of D and E obtains a correct outcome of C = 30. The ITT will assume application of correct relationship.

  Since a student will practice on more than one example, this problem should be of little significance in practice. Nevertheless, it is a theoretical weakness of the outcomes orientation.

- ***Incorrect outcome through a correct process*** -in the above scenario, if a student correctly applies the relationship but makes a mistake in dividing A by B and say, gets an outcome of C = 40, the ITT will assume application of a wrong relationship, and infuriate the student by suggesting use of the relationship that was employed in the first place.

  This problem is found in all tutoring systems that allow *direct entry* of data rather than selection of a rule. The reason is that the tutoring system is not able to infer the source of error except guessing wildly about the student's cognitive process in obtaining that wrong answer. Most of them, however, only provide a feedback on whether the data entered is correct or incorrect and therefore the feedback is appropriate regardless of the source of error - application of a wrong rule or a calculation mistake. The Byzantium ITT's assumption that an application of an incorrect rule is the sole contributor to a wrong solution cannot be sustained in case of calculation errors. The implementation of a graded system of feedback, however, behaves like a conventional tutoring system in the first instance and merely points out the incorrectness of the data entered. The possibility of this type of problem, therefore, only arises if the entered data is incorrect for a second consequent attempt.

7.7.2 Dangers Inherent in the Tools Approach

Declaring a system as a 'tool' is a convenient way for the tool creators to limit their responsibility for its sensible use - success depends not so much on the tool's capabilities as the skill of those employing it. The obvious danger of employing tools in CILE is that the bolt on approach may produce sub-optimal results where the strengths of traditional teaching are exchanged for the weaknesses of educational technology. But it is not to say that a teaching 'tool' cannot provide efficient and effective learning. It is the responsibility of a tool maker in the province of educational technology, to give adequate consideration to its integration and suggest a *modus operandi* - for distance learning, open learning and class based learning. Careful implementation of these tools is very important. Project Byzantium is fully aware of this fact, and each ITT is accompanied by a student manual and a teacher manual to make the implementation task simpler. Course organisers are advised to use these ITTs to supplement the human teacher in a class based environment, and not to replace them. In distance and open learning, the students are advised to use these ITTs to practice the applications of the concepts, once they have learnt the concepts with the help of other educational means such as books.

7.7.3 Problems in Second Level Integration of ITTs

- ***Extension possible only by enlarging a closed system***: When more than one ITTs are linked in a solution, the second level application will have to be a closed system to provide a cohesive expert model. The second level application will, therefore, require identification of different classes of problems requiring use of multiple ITTs - that may be linked through limited Client/Server communications or through Object Linking and Embedding (OLE). It should be noted, however, that open-ended applications may be possible in future, employing techniques similar to the intelligent software agents - packets of code (similar to a virus) which might roam on internet to collect information or might use AI and embedded knowledge to help us in our work (Wayner & Joch, 1995), although they seem to be long way off.

- ***Precision/Tolerance***: Students cannot be expected to work with a very high degree of precision and should be allowed to round numbers up or down to nearest significant digit e.g. a discount factor which is worked out by a calculator as 0.23446946 may be employed as 0.2344 or 0.2345 by a student and this does not affect the overall decision.

  In the outcomes based ITT, however, this requires definition of acceptable tolerances. On a long chain of relationships, these tolerances can either cancel each other out or accumulate. Networking of ITTs creates longer chains and will require some corrective mechanism which does not confuse the user.

7.8 Conclusion

The Byzantium model of CILE does not attempt to radically change the process of education. It recognises the strengths of a human teacher and is concerned with extending a teacher's scope by providing intelligent tutoring tools that will efficiently teach introductory material and take some of the pain out of assessment. This approach should ensure better and more qualitative utilisation of the human teacher's time and effort. It is realistic in its demand on the computer software and therefore more likely to succeed in integrating computers into the curriculum. Nevertheless, it still maintains a vision of growing a much more complex tutoring system out of small and useful ITTs.

The Byzantium ITT interface, which is discussed in next chapter, is also quite simple. It does not use a menu system, but uses a panel of pushbuttons, which at places looks like a traditional menu, because the pushbuttons have text on them indicating the associated action. The pushbuttons work by a single left click of the mouse or by pressing a letter (highlighted on the pushbutton) on keyboard. The only other action oriented control is the scroll bar used for scrolling text in ***Basic Concepts***, a part of the software that explains, in a text form, the topic studied. All data entry is done through an edit control, suitably masked to prevent entry of illegal data.

Due to very few types of control used, the Help facility is tied to text based tutorials rather than procedural help on how to use various features of the software. There is a summarised '*How to use the software*' text which can be accessed from the ***Main Menu*** of the software.

The next chapter deals with the description of various ITT modules which have been developed as a part of this research. Structures of three modules, namely Capital Investment Appraisal, Absorption Costing and Marginal Costing are discussed, and the development of various custom controls is outlined. Subject specific unique interface features of different ITT modules are also explained.

Design and Development of Byzantium ITTs

Although many software packages have been used for the purpose of computer aided learning in accounting, such as spreadsheets, databases and expert systems, they do not provide adaptability according to individual user, and hence fail to provide context sensitive feedback, as they do not keep any record of what the user has learnt and what is still remaining. Intelligent Tutoring Systems fulfil this need, but very little work has been done till now in the field of Intelligent Tutoring Systems in accounting. The in-house development of CAL modules is one of the pioneering works in this field where efforts have concentrated on conveying procedural knowledge of accounting in such a way as to support the acquisition of conceptual knowledge.

8.1 Acquisition of the Concepts

The knowledge transfer takes place in two phases. First, the student grasps the understanding of accounting concepts with the help of textual explanations in *Basic Concepts* part of the package, and then obtains procedural knowledge and numeric skills of the subject in *Project* part by solving numeric problems. The textual explanations are divided into sub-topics to break down the complexity of the subject concepts. Besides providing the knowledge of concepts, the textual explanations also include numeric examples to give a concrete dimension to these abstract concepts.

8.2 Articulation of the Concepts

After acquiring concepts and representations of procedural knowledge in *Basic Concepts* part of the system, students are presented a problem space that structures a problem progressively into its conceptual components to introduce simpler concepts first. The problem space attempts to address the whole problem in a single model but where it becomes necessary, model progression is based on models that are self sufficient (e.g. alternative techniques) or those that provide a logical basis for aggregation to a coarser granularity representation in the following model(s). When the system generates a problem, some of the conceptual components are assigned values, which are sufficient to address the whole problem. The student has to derive the values associated with the remaining components applying the conceptual and procedural knowledge acquired from the *Basic Concepts* part. Since throughout the schooling process, the students are extensively exposed to the 'Fill in the blanks' metaphor, the empty containers of the remaining components are perceived as a challenge and the students are well-motivated to solve the problem.

8.3 Bridging the Conceptual Gaps

At any stage in the problem solving part of the system, the knowledge of the concepts can be refreshed or gaps in the knowledge bridged by returning to the *Basic Concepts* through the *Help* button. These subsequent returns to the conceptual explanations will be in the form of focused inquiry and therefore provide a deep learning experience. Alternatively, a student may also attempt to 'figure out' the situation from the name tags of the conceptual components employing natural intelligence. This process also provides a deeper learning experience.

The modules adopt mixed-initiative strategy, where the system allows the students to move along any of the different paths available to solve a problem, according to students' understanding of what they have learnt by that time. Since the whole problem space is presented to the students, the cognitive load of remembering the values associated with various conceptual components is reduced. It also provides a stimulus to think '*What to do next*'. This approach provides the benefits of motivation and planning activities of the discovery learning process, though the system takes control and instructs the student whenever it infers student misconception. As Anderson et. al. (1996) observed,

> *"When, for whatever reason, students cannot construct the knowledge for themselves, they need some instruction. The argument that knowledge must be constructed is very similar to the arguments that discovery learning is superior to direct instruction. Discovery learning, even when successful in acquiring the desired construct, may take a great deal of valuable time that could have been spent practising this construct if it had been instructed."*

The system reflects its functionalities in the interface as well as its tutoring strategies, which allows it to infer the students' procedural and conceptual knowledge from the interactions in the same fashion as a human tutor does.

The remaining chapter describes the design and development of three in-house modules used in this study with special focus on interface design. Various features of interface are reviewed. The chapter also describes the contents and working of these modules.

8.4 Designing the Interface of In-House Modules

Considering the increasing popularity and use of Windows operating system, the technical team of project Byzantium decided to develop ***Capital Investment Appraisal*** and successive modules for Windows environment. Graphical user interfaces were used for these modules due to the clarity they provide in the appearance, interaction and working environment of the interface, as described in Chapter 4. Interface development started at the same time that the development of first module commenced. All the functionality of the program was tightly linked to the interface so that the user can access the full functionality of the program without any hindrance caused by the interface.

It was envisaged that the spreadsheet metaphor is well established in the field of accounting and a novice student would use the modules easily and instinctively if the modules are based on this metaphor. The approach adopted in the modules was also problem-solving, and therefore it was decided that after studying the basic concepts in the tutorial part of the software, students would be given a screen of boxes, similar to spreadsheet, where they would be able to practise their skills of problem-solving of the subject domain.

Reusability of the code was considered while developing successive modules, but no compromises were made if re-coding was necessary to provide accurate functionality in a particular module. For example, in many cases, the mechanism of displaying messages needed changes to provide exact feedback to the user.

The approach used for the development of the interfaces of the modules could be classified as 'Craft Approach' (Wallace & Anderson, 1993) as described in chapter 4. It was started with the brainstorming sessions where the design of interface was laid out. A prototype was developed and evaluated by about 3000 students and about 20 lecturers at various institutions in UK and overseas. Many improvement suggestions were implied and further evaluated to bring the interface in its proper form.

Principles for good interface design from the literature review (refer chapter 4) were applied while designing the interface. Efforts were made to provide a consistent interface throughout the program in every possible sense, such as semantics, syntax and so on. Navigation was also made as simple and intuitive as possible.

The role of interfaces in educational environments were also considered and it was considered helpful to give the users all relevant values already entered by them on the screen while calculating further variables. In this way, a part of student model was made available to the user. Many error-prone actions, which may be obstacle to the subject learning, were eliminated from the interface. For example, downloading and uploading of values from and to the on-screen calculator was made automatic to prevent typing mistakes.

The next section focuses the practical aspects of the interface design and describes the development of various interfaces of in-house modules.

8.5 Development of Byzantium ITTs Interface

Following the success of Marginal Costing prototype package, team members initially decided to keep the framework and strategy of all the modules to be same as of Marginal Costing. The Marginal Costing package was DOS based software written in the Clipper language. It was decided to use the same language as it gives almost same flexibility of the good programming language while keeping most of the features of a database package. Authorware packages were also considered but none could give flexibility at user as well as programmer level, without increasing the cost of hardware. If extensive DLLs (*Dynamic Link Libraries* are the files which have collection of functions to be used by the programs at run time) have to be written, it was considered preferable to program from scratch and escape the pain of circumventing an authoring language's limitation. As only DOS version of Clipper was available at that time, it was decided to use a third party library called "Dolce Vita" which purported to provide the necessary functions to emulate Windows flexibilities. Very quickly, the technical team started feeling frustrated because of the limitations in the design and features of the output. Dolce Vita was a general purpose library and was not able to provide the flexibility and functionality required in CAL programming.

8.6 A Move Towards More Flexible Programming Environment

The software tools initially selected were not satisfactory for dedicated CAL programming. Therefore, the team decided to move to more flexible programming environment and the work was reinitiated in Microsoft C/C++ v7.0. Various small functions and objects were produced which were used afterwards in all six modules. Work was initially moved to Visual C++ v1.5 and subsequently to 32 bit Visual C++ v4.0. The applications became increasingly class-based as these were faster to develop, and easier to maintain.

8.7 Development of Custom Controls

It was found that CAL interface programming is quite different from other software development. The technical team faced quite a number of problems while developing these CAL interfaces and this resulted in the development of various custom controls. Some of the problems encountered, are as follows:

- It was found that, although most of the student laboratories are equipped with Windows environment, the presence of a working mouse is not always available. Therefore, it was decided to develop all proposed CAL modules so that they could be used with or without mouse (only keyboard) with same level of simplicity.

- All Windows controls are configurable by nature. Although this is very useful feature, it is not desirable in dedicated CAL programs where software is to be applied in laboratories and students are on the mercy of previous users on a particular terminal in the context of colours etc. Although every student of elementary accounting course is not expected to know the use of advance Windows facilities, some experienced users may configure Windows with poor colour combinations etc. Therefore, all the controls inside the CAL modules have their pre-determined features, and are not configurable externally.

- *Editbox:* It is not possible to restrict user entry to a particular format in the editboxes provided by Windows. Numeric only or restricted ranges cannot be defined, preventing any sort of pre-validation. This would unnecessarily increase the inter-action traffic and overload the post entry validation which is mainly concerned with the task of verifying the entry within the knowledge network and provide feedback. It is also not possible to individually change the foreground, background and boundary colours of editboxes to provide grouping or special effects.

It was decided to design a configurable custom editbox which provides these facilities and all other desired features, such as navigation and validation functions within editbox.

- *Scrollbar:* An aesthetically pleasing scrollbar, narrower than the Windows' default one, was needed to match with interface screen. A fully configurable custom scrollbar control has been created which has typical features required in a CAL package.

- *Buttons:* A button control was needed with four modes of status: up, down, focus and unavailable. In unavailable mode, it was supposed to return all messages to the parent window for processing. Handling of keyboard was also necessary while the button is in focus. Windows' default button functionality could not cope with all the requirements and hence, custom control buttons were created which provide many other features besides handling all these requirements.

- *Viewbox:* There was a requirement of a read only editor, which can be used to read tutorials in the Basic Concepts part of the software. Windows does not provide any ready-made utility which can show text, graphics, tables and other reading material on screen with desired changes in colours, borders, shadows etc. Also, it was not possible to get full mouse and keyboard control over scrolling of the text and other editor features. So, once again, it was decided to write a custom control for viewbox which can retrieve a file and can show it on the screen in a desired format and structure.

- *Painting of screen:* Painting of various graphic objects is not an easy task in Windows. Technically, whenever any area of screen has to be modified, a lot of care has to be taken to what and where repainting is needed. The Project Team circumvented this problem by writing a "canvas" routine which is basically creation of a bitmap image of the screen. Whenever any graphic object is added or removed from the screen, it is to be done in that bitmap image only. It also prevented the flickering on the screen which is a usual phenomenon while repainting Windows.

8.8 Additional Features of Interface

Some additional features are available in the CAL modules to enhance the simplicity and usability of the interface. These features are particular requirements of CAL packages and are made interactive to other parts of the applications so that the end-user can take advantage of these features while using the application in normal routine.

- *Calculator:* A calculator control is created which is an emulation of pocket calculator. It is customised in size and colour to match the interface of the application. The advantages of the calculator control over pocket calculator is that it is interactive with the other controls of the application. Values from editboxes can be uploaded to the calculator by one mouse click and the same applies to the downloading of the results. A very important feature of '*Sum multiple values*' is added to the calculator which allows the summation of multiple values of selected editboxes.

- *Marker interface:* A marker is provided for the assessment of improvement in the student's knowledge. It consists of two modes. "Tutor mode" allows the selection and modification of marking schemes. Tutor can set the weights to various sections of a problem. Since marker works off-line and creates all reports in password secured tutor mode, there is little chance of student's getting any confidential data.

"Student mode" of marker allows students to see their marked work. Students can see correct answers against their wrong answers and can obtain a measure of their overall performance. Marker also marks the method adopted by the students using student modelling techniques for "interpreted problems" where model answers are supplied by the tutor. The students can see where they have gone wrong in calculations, in spite of adopting correct problem solving pattern.

The marker is linked to all in-house modules through *data driven programming* paradigm (where the flow of the data is defines the order of operations implicitly, and the structure of a software system is derived by mapping system inputs and outputs), as each ITT, through its file structure, informs the marker about its data.

- *Graph representation:* A graph creation utility is provided for the representation of the results of problems, wherever appropriate, so that students can visualise their answers for comparative studies. For example, in ***Capital Investment Appraisal*** module, graph facility has been provided to show the inter-relationship of *internal rate of return* and *net present value*.

8.9 Description of Capital Investment Appraisal Module

The ***Capital Investment Appraisal*** software consists of one floppy disc and one manual for students. The software assumes no prior knowledge of the subject and almost no computer knowledge. It aims to teach various techniques of Capital Investment Appraisal, namely Accounting Rate of Return, Payback and

Net Present Value with Internal Rate of Return. The software only deals with various techniques and not with the mixing of techniques. There are five tutorials in the software which are a cut-down version of the manual. These are basically for revision purposes just before doing any exercise on the package. These tutorials are:

1. Introduction to Capital Investment Appraisal
2. The Payback Method of CIA
3. Net Present Value
4. Discounting to find the Internal Rate of Return
5. Accounting Rate of Return

The software is push button driven. The architecture is shown in figure 13. Various modes are explained below in detail with student instructions.

➴ ***Learning***

Options available are :

- Select Basic Concepts from the first menu.
- Read Tutorial One.
- At the end of Tutorial One, access the examples by going into self testing mode as instructed and complete them.
- Read Tutorial Two, access and complete its examples and do the same for all five tutorials.

➴ ***Self testing***

This mode is for student practice and for further student learning once they have been introduced to the subject by the tutorials. Various steps in self testing interactive mode are:

- Select the interactive mode from the main menu.

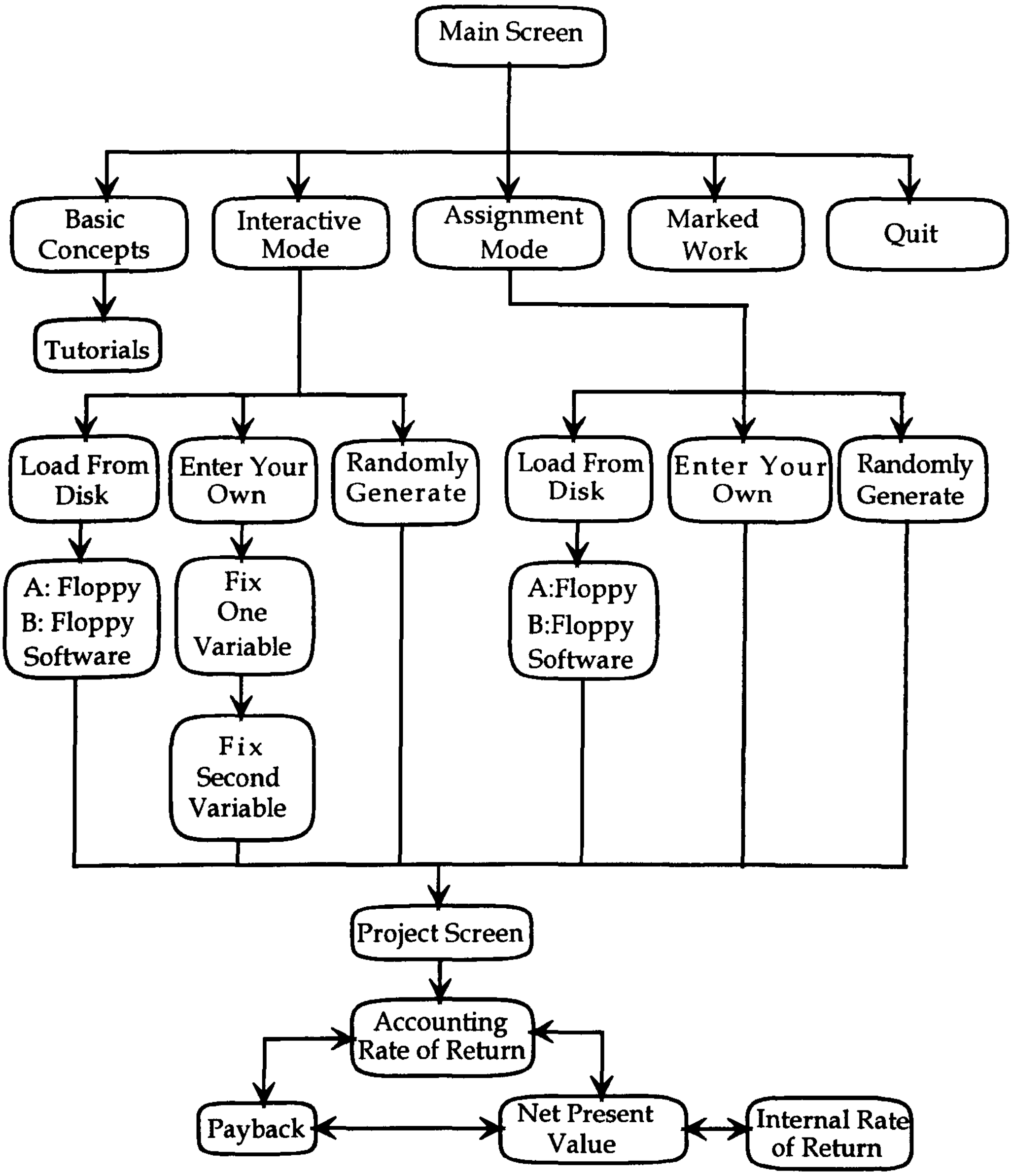

Figure 13. Architecture of Capital Investment Appraisal Module

- This gives another menu of following choices:
  - Load question from disc
  - Enter your own questions
  - Randomly generate questions

- Select the Assignment mode from the main menu.
- On the menu, there are three choices, same as interactive mode.
- Complete all the screens for all modes of appraisal.
- Students may save partly completed assignments and re-load them for later completion. Once work is saved ready for automatic marking it cannot be re-accessed or changed.

- ***View Marked work***
- Once the work has been marked, the disc is returned to the student.
- To view the marked work, students should select *View Marked Work* from the main menu.
- Marked work is screened showing what is correct and incorrect in each method of appraisal, and the software shows a summary of the marks awarded for each question attempted.

8.10 Description of Absorption Costing Module

The ***Absorption Costing*** module has a similar interface as of ***Capital Investment Appraisal*** module.

There are five options on main screen: Basic Concepts, Interactive Mode, Assignment Mode, View Marked Work and Quit. Basic Concepts provides a general introduction of following topics.

1. Costing Information - Product cost, Types of cost, Direct and Indirect costs
2. Overhead Absorption and its method
3. Applications of Absorption Costing
4. Control of Overheads
5. Alternative Costing Methods

- Complete all the screens for all modes of appraisal. In case of errors, *Messaging System* provides help.
- When students are confident that they are able to solve the exercises easily, they are ready for the assignment mode.

In self testing mode, students always receive assistance from the unique *Human Tutor Messaging System* of the *Inference Engine.* Although the *Inference Engine,* working behind the interface, has its own solution for each variable after calculating the dependent variables from independent variables, it does not force its own solution on the student. In Capital Investment Appraisal, it is possible to have different values for intermediate variables although the overall solution remains same. Therefore, the software tries to solve the problem with the user input to find out if another solution is possible with student's value of that particular variable. If a solution is possible, it takes that solution into account and provides further help according to that solution. But if no feasible solution exists with student's value, then it gives a hint to the student how that value can be obtained. If student still enters a wrong value, it shows the actual calculation to be done at that stage. If student's input is still wrong, the software enters correct value in the box and then suggests the student to go to Basic Concepts part of the software to read the tutorial again. Students are encouraged to use this mode in order to assure themselves they understand the concepts fully and can approach the assignment mode with confidence.

Assignment mode

This mode is for testing students' understanding. It provides no help to students. It allows incorrect input and gives no message feedback. Students are asked to save their work and hand in disks for automatic marking. The lecturer then marks those disks using password protected marking mode of the software and returns the marked disks back to students. Student results are automatically recorded for the lecturer. Various steps in assignment mode are:

Interactive and Assignment modes have the same structure, the only difference is that there is no interactive guidance to students in Assignment mode. There are four screens in these modes.

- Information Screen, which contains the initial data regarding products and services.
- Overhead Absorption Rate Screen, where student is able to calculate the rate of overhead absorption on some particular basis.
- Over and Under Absorption Calculation Screen.
- Product Cost Calculation Screen.

8.11 Description of Marginal Costing Module

Marginal Costing software, which was developed as a prototype, has been rewritten for Windows environment. It aims to give students a grounding in identification, calculation, and use of marginal costing information. The students are introduced to the concepts of:

- Unit Revenue
- Unit Variable Cost
- Unit Contribution
- Total Revenue
- Total Variable Cost
- Total Contribution
- The Contribution to Sales Ratio
- Break Even Points
- Margins of Safety

In addition, students learn the basic uses of marginal costing to:

- Evaluate extra orders and alternative strategies
- Find break even points
- Find margins of safety
- Inform make or buy decisions

✎ *Basic Concepts*

Selecting it displays the tutorials. There are seven tutorials:

1. Variable & Fixed Costs
2. Contribution & C/S Ratio
3. Profitability of Alternatives
4. Profitability of Extra Orders
5. Profitability & Breaking Even
6. Profits and Margin of Safety
7. Profitability : Make or Buy

✎ *Interactive and assignment modes*

Interactive and assignment modes contain only one screen, where fourteen boxes are laid out. In randomly generated questions, four out of 14 variables are generated by computer as independent variables and students are supposed to calculate rest 10 dependent variables.

8.12 Summary

Byzantium ITTs are designed for the novice users who will have limited computer knowledge. Users must be able to concentrate on learning activity rather than interacting with the system. These factors have informed to design the interface. That's why, users are provided with a restricted *easy to use and easy to learn* interface.

This chapter demonstrated the development of ITT interfaces under project Byzantium. Unique features of these interfaces were also discussed with focus of the problems encountered. Structures of three modules used in this research were explained with respect of subject matter and student instructions.

After the development of CAL modules, they are to be evaluated for their efficiency and effectiveness on the user community. The next chapter deals with the evaluation methods and techniques available in the literature for computer aided learning packages, and the approach adopted in the Byzantium project.

Evaluation of CAL and Statistical Techniques

Heller (1991) commented that instructional software, like all other educational material, should be evaluated before it is used in the classroom or research laboratory. The challenge is to decide what to evaluate, how the process should be carried out and by whom. There have been many attempts in medical domain to provide a methodology of evaluating the expert systems (Miller & Sittig, 1990; Spiegelhalter, 1983; Rossi-Mori et. al, 1990) which parallels with the approach needed in the field of education. The literature in CAL evaluation suggests that initially, any intelligent system should be evaluated for its overall effectiveness and usability. Once the system is found acceptable in initial evaluation, the efficacy of its components should be determined in the real environment. Quantitative methods are favoured to find overall effectiveness of the systems (Legree et. al., 1993; Mark & Greer, 1993) and the qualitative methods are recommended to find the internal efficiency of the whole system and its individual components (Murray, 1993). Wyatt & Spiegelhalter (1990) suggested laboratory testing as a most suitable method for initial evaluation with stress on field trials at later stages. The multi-institutional studies more closely resemble field trials as the likely involvement of multiple evaluators weakens the control and also as they cover a far wider variety of subjects. This chapters provides a detailed discussion of these evaluation techniques. It also provides the history of evaluation of CAL, available evaluation techniques and statistical procedures suitable for these evaluations. It then focuses on the evaluation and statistical techniques used in the project.

9.1 Evaluation

Evaluation, as Baker (1991) pointed out, looks back at what has been accomplished and accordingly asks "*Where have we been and what do we know?*".

Furthermore, Iqbal (1994) has defined evaluation as, "*the process of deciding the worth of things and their costs in relation to that worth.*" Hence this is related to deciding whether one particular system is worth having, or in choosing between two different systems. One way of measuring this is to determine how effective the teaching/learning system is as a whole.

Evaluating Computer Aided Learning packages is costly and time consuming. After building PROUST, an ITS which finds non-syntactic bugs in student's PASCAL programs and then tutors the students about these errors, Littman et. al. (1988) raised the question that if evaluation of CAL programs is so costly, why do it at all? Wouldn't it be better just to finish one CAL program and then build the next one, perhaps letting the marketplace determine survival? They commented that evaluation pays off by helping to answer two questions that are central to cognitive science, artificial intelligence, and education:

1. What is the educational impact of a CAL program on students?
2. What is the relationship between the architecture of a CAL program and its behaviour?

They found that these questions lead to a different perspective of evaluation than that from traditional educational evaluation. Traditional evaluation works on techniques called *Formative* and *Summative* evaluation techniques. *Formative evaluation* provides feedback into the design process in order to refine and form the design (Byrum, 1992). This type of evaluation is more likely to require qualitative information in order to pinpoint design flaws. The *Summative evaluation* approach, however, seeks to provide an overall assessment of the final product. This approach finds quantitative information more useful than qualitative information. The merger of these two approaches is necessary as the design of products continue even after their introduction into the user market (Hewett, 1986). Littman et. al. (1988) defined two different classes of evaluation for the analysis of their two evaluation questions. One class consists of methods based on recent progress in student modelling. From these methods, researchers can study how a CAL program affects students and changes their knowledge and problem solving skills. Methods in this case can be used for 'external evaluation', so called because it assesses an effect external to the CAL, namely the student's learning. The second class defined by Littman et. al. (1988) can be used to construct an accurate picture of the relationship between the architecture of a CAL program and its actual behaviour. These methods are the basis of 'internal evaluation', which is concerned with the inner workings of the CAL program.

There is a variety of purposes which evaluation can serve and a complex range of issues can be involved in any evaluation project (Ridgway, 1986). Social aspects are the neglected areas in the evaluation of CAL program. Ridgway et. al. (1984) found that microcomputers show a major role in enabling teachers to shift the social dynamics of classrooms. Quite different kinds of interaction take place when pairs or groups of users, interact with a computer, compared to individual, tutorial CAL. Systems issues relate to the whole environment within which CAL takes place. In school environment, systems issues will involve the personal dynamics which determine priorities amongst users. In university contexts, systems issues will include times of access of staff and students; levels of machine provision; staff support during tutorials; and decisions about whether CAL is compulsory or optional.

Coutaz et. al. (1993) suggested that evaluation techniques for interactive systems can be divided into two broad categories. *Predictive methods* are applicable during the design phase. They do not require any system implementation, and nor do they need effective users. At the opposite end, *experimental techniques* rely on the existence of a physical apparatus, ranging from mock-ups of the real system up to the full implementation of a running prototype.

Gardner (1990) stated that when dealing with a complex issue like teaching and learning, it is almost impossible to produce the incontrovertible evidence, for or against, that many would wish for. The tendency has been towards searching for some form of experimental approach (control vs. trial groups) which would yield some degree of rigorous, incontrovertible empirical evidence about the 'worth' of the activities. Every control group has its weak points and every teaching/learning situation is dynamic and prone to change. This is not to say that experimental approaches should not be used.

Qualitative techniques provide information as a function of personal interaction and perception (Murray, 1993). The qualitative evaluation of CAL packages does not imply a lack of thoroughness (Heller, 1991) but may give an insight to the aspects of learning and performance that are impossible to apprehend by other means. Qualitative evaluation is primarily about identifying the characteristics of a situation or setting, and quantitative analysis is about finding causes and consequences (Shute & Regian, 1993). Henwood & Pidgeon (1993) noted,

> *"Alan Bryman (1988) has argued that a good deal of confusion exists .... what constitutes qualitative and quantitative research. Part of the confusion comes from the narrow association of qualitative methodology either with particular modes of data gathering .... or its non-numeric character .... However, method is more that data alone. The gathering, analysis and interpretation of data is always conducted within some broader understanding of what constitutes legitimate inquiry and warrantable knowledge."*

Murray (1993) suggested several characteristics typical to qualitative research:

- The researcher is the key instrument - subjective judgement is involved in collecting the data, be it note taking, photographs, videotapes, etc.

- It is descriptive - the data collected is primarily in the form of words or pictures rather than numbers, and anecdotal evidence is common.
- It is more concerned with process than outcomes or products.
- Data tend to be analysed inductively - abstractions and theory are built "bottom up".
- "Meaning" is of essential concern - the researcher is interested in the beliefs, attitudes, and perceptions of the participants/subjects.

The traditional quantitative evaluation methods have been favoured by Legree et. al. (1993) at the initial stages of implementation of the CAL modules, as quantitative procedures provide an estimate of the overall product effectiveness of a CAL application. Once the overall effectiveness is ascertained, the careful evaluation of the internal components of an intelligent system is recommended through a combined quantitative and qualitative evaluation procedure.

Murray (1993) suggested that quantitative methods are most appropriate to prove that an ITS system 'works'. Once it is proved that the system works, qualitative analysis is necessary to determine which aspects work, which don't work, and descriptive discussions of how and why.

Mark & Greer (1993) suggested that quantitative evaluation is useful to find out whether or not ITSs will be accepted and used. They may also be interesting as measures of the long term impact of ITS on attitudes and education.

Wyatt & Spiegelhalter (1990) provided a detailed overview of the evaluation of medical expert systems, and suggested that once the system is developed, it is necessary to perform a careful preliminary evaluation before the system is introduced into the practice. This *laboratory testing* phase is necessary to show that the system is 'safe' and has at least the potential to benefit its users. Once the initial usability of the system is determined, the field trials can be used to analyse the value of the systems to their users. Wyatt & Spiegelhalter (1990) warned that for an expert system to be successful, its 'usability' is a vital aspect for study and evaluation. Intelligent tutoring systems have much in common with expert systems in terms of their structure and user interaction, the initial *laboratory testing* phase can be considered crucial for their implementation in the academic environment.

Since the aim of the evaluation study presented in this thesis is to examine the overall effectiveness of the in-house CAL modules, mainly quantitative methods are used in the evaluation, as suggested by various researchers (Legree et. al., 1993; Murray, 1993; Mark & Greer, 1993). Once this is determined through quantitative analysis that the approach adopted in in-house modules works, further research will be needed to employ qualitative techniques for more detailed analysis of changes in students' behaviour due to application of CAL modules and long term impact of such CAL material in university accounting education. This view has been supported by Bryman (1989) who observed that the qualitative research emphasises on the perspective of the individual being studied. Bryman (1989) noted,

> *"the qualitative researcher seeks to elicit what is important to individuals as well as their interpretations of the environments in which they work through in-depth investigations of individuals and their milieux."*

Some qualitative views from students have also been sought in the evaluation study presented in this theses but the data obtained from various sources was not uniform and was merely in the form of observations and open-ended questions (asked in the subjective questionnaires). Though this data was of limited value for statistical analysis, it provided an indication of the students' receptivity towards the CAL modules and the perceived strengths and weaknesses of the individual packages. Although, the evaluation study presented in this thesis is a laboratory testing of the packages, its multi-institutional nature brings it near to the field trials.

Gardner (1990) recommended a kind of "mixed economy" of evaluation techniques with every opportunity for so-called "rich" information/evidence to be gleaned from observation, participation and enquiry especially directed at the "end-users", the students.

9.2 Evaluation Techniques

Howard & Murray (1987) suggested that it is possible to identify five main types of HCI evaluation after a wide-ranging literature search of some eighty references. These are general Human Computer System evaluation methods. These are:

- Expert based
- Theory based
- Subject based
- User based
- Market based

These HCI evaluation methods are described below. As described later, the *Subject based* evaluation is widely used for the evaluation of CAL packages. It is discussed in detail in section 9.3 of this chapter.

Expert based

In this situation a domain expert may employ expert knowledge, scientific principles and intuition to evaluate the interface. Examples of this form of evaluation exist in the work of Hammond et. al. (1984). This evaluation can, if necessary, be conducted quickly and at any time during the design cycle. It enables diagnostic analysis. An expert system capable of disseminating human factors information to the designer may be useful in reducing the need for an (expensive) human factors expert (Wilson et. al., 1986). Currently, however, it does require the evaluator to have extensive knowledge of the user, is open to possible unchecked bias and tends to be incomplete.

Theory based

This type of evaluation consists of *a model of the user* and *a model of the interface* and occurs where the mapping relationships between formal representations of the user and the device are examined with a view to identifying any mismatch. Examples of this form of evaluation exist in the work of Card, Moran & Newell (1983) and Kieras & Polson (1985). The formal representation of the system provided by theory-based evaluation can be used as a basis for simulation, allowing predictions to be made about system performance. Theory-based evaluation is also capable of fine-grained diagnostic analysis. It does, however, require considerable expertise and the user/system task representations may need extensive tailoring from one situation to the next, which brings its applicability into question. It is also a problematic exercise to choose the most appropriate user/system representation. These techniques have a narrow focus, many assumptions.

Subject based

Subject based evaluation is directly based on the user's judgement (Reiterer & Oppermann, 1993) and is the most widely used type of evaluation method for CAL packages (Howard & Murray, 1987). This consists of four main components: system, task, subject and metric. Data collection may involve running naive or practised subjects under laboratory conditions while individual techniques range from traditional reaction-time type experiments to multi-user attitude surveys.

Data on a user may be collected at four different levels: *affective, cognitive, behavioural* and *physiological*. Data from one level of the user alone is likely to be insufficient to draw relevant or appropriate inferences from. For example, it is possible to maintain performance in the light of varying task demands, with respect to some criteria, by varying the amount of effort invested in the task.

Measures of primary task performance alone would not be sensitive to this. Data should be collected from at least two, if not three, levels of the user.

Subject based techniques generally present less constraints on the data that can be collected than the previous two approaches and are less open to bias. This, however, may not be true of some questionnaires, structured interviews and the like. The great variety of techniques - and the limited advice available regarding selection of a technique - can result in an inappropriate application which can affect the worth of the data collected. The results of good subject-based evaluations are, however, very persuasive for designers and useful for comparing systems against high-level criteria such as other systems.

User based

This consists of a user and a system and relates to personal evaluation by the user reflected in terms of patterns of system use relating to the task and the overall environment. The data yielded by user evaluation is a by-product of system use and, as such, open to influence by a large variety of other variables. This form of evaluation can be considered as the ultimate step in an evaluation procedure. Although interpretation of the data is difficult, they can be useful in developing future systems and they are regarded as important because the final decision of whether or not to use the system lies with the user.

Market based

This consists of a market and a system and relates to the final evaluation conducted by the market place reflected in terms of market performance. The data obtained by this type of evaluation is open to influence by other variables e.g. organisational policy and the state of the market. Again final decision of whether or not to use the system lies with the user.

9.3 Subject Based Evaluation Techniques

Subject based evaluation techniques are being widely used for the evaluation of CAL packages (Daroca, 1986; Simpson, 1986; Gallagher & Letza, 1991; Kachelmeier et. al., 1992; Iqbal, 1993; Tonge et. al., 1994). These are described below.

➢ *Observation*

Although a number of techniques are observational in nature in that data is emitted by the subject, not elicited by the experimenter, the term *observational* is being used to denote, not a class of techniques, but an individual technique distinguishable from others. The distinction is based upon three dimensions: the data collection agent and the twofold nature of the data collected. Observation, in this project, is considered to be the collection of emitted non-verbal data, either manually or with the use of video tapes (only manual observations were performed in the actual evaluation studies). A number of papers refer to this form of observation (Neal & Simon, 1983; Bewley et. al., 1983). This technique is frequently used alongside verbal protocol analysis where no distinctions or attempts to assess the strengths and weaknesses of each are made.

Bewley et. al. (1983) report on an extensive evaluation of the Xerox Star workstation. Three experiments were designed to assess the number of buttons, and meaning of each, required on a mouse; to identify the significant parameters in the shape of icons; and to assess the workstation's suitability for making line drawings. Amongst the data collection techniques used where observational video recordings, on-line logging, verbal protocol analysis, interviews and questionnaires.

The video recordings were used to assess the workstation for making line drawings - the least specified part of the evaluation. The cameras showed both the users and the screen along with the time of day. The evaluators also conducted manual observation, rating critical incidents which could later be categorised with a view to identifying sources of problems and difficulties. Bewley et. al. (1983) summarise that the main advantage of the observation technique is that it is unobtrusive and complete. It is convincing to designers and is useful for a first pass evaluation or for supporting other techniques. The major disadvantage relate to problems of data reduction and technical knowledge required.

➢ *On-line Logging*

On-line logging is the collection of emitted non-verbal data by the system. It should be distinguished from such methods as on-line questionnaire administration which would be considered to be the elicitation of verbal data. Penniman & Dominick (1980) distinguish between 'monitoring' and 'evaluation'. Monitoring refers to the collection of data on the functioning and usage of the system. Evaluation is the process of analysing that data so that decisions regarding system functioning can be made. On-line logging is regarded to be automated monitoring in the terms of Penniman & Dominick (1980).

Howard & Murray (1987) suggested that on-line logging may be conducted for three main reasons:

- to provide information on a system with a view to assessing its usability, functionality, efficiency etc.;
- to provide information necessary for the system to function, as in an adaptive or tutoring system;
- to provide the user with feedback on the interaction.

The main advantage of on-logging data collection technique is that it is unobtrusive. Few resources are required for data collection and it provides a permanent 'complete' record of the interaction. The disadvantages include the problems of data reduction since this technique generates large amounts of low level data.

It is possible to collect a variety of data with this technique. Atherton (1972) reported on the collection of search term data whereas Mittman & Dominick (1973) reported the collection of real time, CPU time, number and type of user errors, number of records scanned, number of print reports and user comments as well as search term data.

➢ *Questionnaires*

There is a sense in which questionnaire and interview techniques are simply variations of the same technique. If a series of questions are put to a subject via a written sheet, and the same questions are read aloud, then it may be said that questionnaires and interviews respectively, have been used to collect the same data.

Differences between the techniques clearly point to situations where each would be appropriate. The main advantage which questionnaires have over interviews is in terms of resources: they are easy to apply and do not require the presence of an experimenter. As a result, they are an inexpensive tool. One major disadvantage of questionnaire relative to interviews is that of inflexibility. When a questionnaire is developed it is important to know the type and amount of information required, for, unlike interviews, changes cannot be made to the type of data to be collected whilst in operation. This requirement clearly increases in importance with the specificity of the questions. A general directive might be to use interviews when resources are many and the domain of interest ill-defined. When resources are restricted and the interest domain is well-defined then questionnaires may be more appropriate. The use of general questions can to some extent circumvent problems arising from a fuzzy interest domain. Root & Draper (1983) gave some good recommendations for questionnaire design.

Questionnaires provide a means of collecting cognitive and affective data quickly and easily. They use few resources. The practical advantages of this approach diminish sharply, however, when increasing amounts of rigour are brought to their development in an attempt to ensure validity and reliability.

➤ *Interviewing*

Much of that which is true for questionnaires approaches to interface evaluation is also good for interview approaches. However, there are technical differences including the necessity of interviewer to be present at the time of interview. Interview design is rarely subjected to the same psychometric rigour as questionnaire design. The flexibility provided by an interview reflects the amount of discretion that an interviewer has over the course of the interaction. This discretion can be a cause of subjective bias and can be divided in four sources of interviewer bias:

- *Halo effect:* If an interviewee is rated as good on a dimension considered important by the rater then they may erroneously be considered good on all dimensions.
- *Stereotyping:* Attributing a variety of traits to an often arbitrary label (e.g. expert, novice).
- *Projection:* The interviewer perceives characteristics in the interviewee that are actually part of their own personality.
- *Contrast and assimilation:* An act on the part of the interviewer of judging the interviewee as being closer to themselves on some scale than they actually were. If in reality they were not far away, then that is termed 'assimilation', while if they were further, that is termed 'contrast'.

All of these effects are possible sources of bias in using the interview for evaluation purposes. However, this will also be true to some extent of the interviewee's perception of the interface.

Another aspect may be the nature of the questions themselves. Closed-ended questions may miss a lot of information but make for easy analysis (in extreme case simply counting up yes/no responses). Open-ended questions have the opposite characteristics. The more structure that is placed upon an interview the more it is possible to assess its validity and reliability but also the more it grows to resemble a verbally delivered questionnaire, and thus the advantage of flexibility is lost.

9.4 Evaluation Techniques for CAL Packages

The most common techniques in literature for evaluation of CAL packages are two group trial studies. Webb et. al. (1991) evaluated an intelligent educational system, the Unification Tutor, for teaching the unification of terms from the Prolog programming language. The evaluation of the Unification Tutor consisted of a comparison between two versions of the Tutor, identical except that in the Modelling condition (MOD) the full Tutor was used, whereas in the No-Modelling condition (NOMOD) the Feature Based Modelling (FBM) student model was not used. In all, 46 students used the Tutor, 21 in MOD and 25 in NOMOD. the focus of the evaluation was on two questions: (i) whether the sequence of problems presented to a student is especially effective for that student, and (ii) the usefulness of the comments provided by the software to the students. The analysis showed that there was no significant difference between results of two groups.

Kachelmeier et. al. (1992) evaluated the effectiveness of a computer-intensive learning aid for teaching pension accounting with the help of between-subjects treatment-control design over sequential semesters at a large public university.

Semester 1 was the control group semester (n=137), in which students completed the pension section of a financial accounting course without computer assistance. Semester 2 was the treatment semester (n=88), in which students completed a set of pension homework problems using the computerised learning aid. The ANCOVA findings of the analysis supported the hypothesis that the pension learning aid accounted for a significant improvement in comprehension of the pension material after controlling for general differences in overall performance.

Simons & De Jong (1992) used two group study for the domain of vocabulary learning. The students were asked to learn foreign words with the help of a keyword. Two keyword conditions were compared between a control group of 14 students and experimental group of 30 students. Simons & De Jong (1992) found the two group study well suited for the purpose. The results showed no significant difference between two groups. This may be attributed to the small sample size which may have resulted in small differences in the results and reduced the chances of detecting any significant differences between the two groups.

Wang & Sleeman (1993) compared the relative effectiveness of computer assisted instruction and conventional methods for teaching an operations management course in a business school. 38 males and 14 females of two undergraduate level classes in the school of business at a university of central Connecticut were randomly assigned into two groups: an experimental (CAI) group and a control (CI) group. Both groups were taught by using a lecture/discussion strategy over a five weeks period. However, the CAI students were allowed a twenty-minute period to review the content utilising an individualised computer drill and practice after each lecture/discussion. All students in both groups were pre and post tested to analyse the gain in their performances. The results of analysis showed no significant difference among the performances of students of two groups. The ANCOVA results suggested that there was neither a positive nor a negative effect on student performance and retention level when using CAI drill and practice program as a supplemental tool in teaching.

Ruf et. al. (1994) analysed the relationship between the cognitive abilities (spatial relations, spatial scanning, mathematics, and verbal) and accounting test performance with the help of a non-computer assisted group and a computer-assisted group. Non-business undergraduate students in the USA were recruited through an advertisement directed at students taking psychology courses. They were divided into two groups. The traditional group had 44 students and the Computer-assisted group had 42 students. Subjects did not know that there were two experimental groups. Sessions were alternatively held in one of two classrooms: (a) a classroom in which traditional group subjects sat at desks without computers, and (b) a classroom in which computer-assisted group subjects sat at desks with computers. All subjects were post tested and their Scholastic Aptitude Test (SAT) scores were used as pre-test scores (Scholastic Aptitude Test is the entrance test for any degree in United States). The results showed that the accounting test performance differs for both instructional methods for students with different cognitive abilities (spatial relations, spatial scanning, mathematics, and verbal).

Forrester (1995) describes an evaluation study of learning using X-Windows based hypertext system. Students involved in the study (who were learning the programming language FORTRAN on a course specifically designed for non computing science degree students, i.e. mostly from business and management courses) were randomly assigned to three groups prior to course commencement. One group, a 'hypertext only' group learned the complete course using only the hypertext system provided. A second group was taught using the traditional method of learning FORTRAN (i.e. through lectures and session exercises). The third group attended the lecture course and had access to the hypertext course system. Students in all three groups completed detailed pre- and post- study questionnaires. Furthermore, logging data was collected for the two groups of students who used the hypertext system. The results showed that the students who were taught using solely hypertext based material in an 'on-line' context performed as well as students who were taught using the traditional lecture method.

Magnuson-Martinson (1995) used multitrait, multimethod, and experimental evaluation method to compare a variety of computer aided learning packages. Computerised activities were used as treatment in one section of the class, and in the same term, as a control, another section of the class was taught in traditional manner. There were no significant difference between the performance of two groups.

9.4.1 Research Design Considerations: Internal Validity

Duncan (1993) suggested that there are several possible sources of invalidity in the research on the efficacy of software. One problem associated with the use of between-group designs, for example, is the possibility of selection bias due to the non-random assignment of subjects. This possibility increases when different experimental treatments are applied to different sections of a course (or to different classes in a university department). The members of one (or more) section of a course receive CAI while the remaining section or sections serve as no-treatment or other controls. Personal experience in teaching multiple sections of courses suggests that class performance, interests in the subject, and related variables can vary significantly in the absence of different treatment, even when class size is fairly large (35-50 students) and when a class comprises either those majoring in one particular discipline or a mixture of disciplines.

The random assignment of subjects to experimental conditions has been incorporated in much of the research on the efficacy of software (Hannafin & Carney, 1991; Thomas & Bostow, 1991), but the possibility of subjects' bias remains. Some researchers have taken measures of subject variables considered relevant to check for possible differences between the groups of subjects in an experiment. For example, Liefeld & Herrmann (1990) selected variables for analysis: academic major, number of previous courses in the major, an English aptitude test, semester grade average, and other variables selected to ensure academic equivalence among the groups.

If matching to control for learner characteristics is attempted as in Varnhagen & Zumbo (1990), the question is which of the student's characteristics should be assessed (such as measures of ability, aptitude, or motivation; prior knowledge of the subject) and what instruments should be used for such assessments. There is little consistency in the literature (Duncan, 1993). The choice of outcome measures partly determines the variables selected for matching, and these measures may vary considerably from one investigator to another.

9.4.2 Comparison of Various Instructional Activities

Another design feature to be considered is the type of compared educational activity. Since some studies (Grabe, Petros & Sawler, 1989; Petty & Rosen, 1990) used only a no-treatment control group, the possibility exists that other instructional activities not requiring the use of computers would produce similar or even larger gains in performance. In other cases, because of the difficulty of equating different educational treatments, more than one factor is permitted to vary. For example, Thomas & Bostow (1991) found no significant differences in post test scores between groups to whom rational-emotive therapy concepts were presented in either 300 written statements with key concepts replaced by blanks requiring the students to fill in an accompanying answer sheets or the same 300 statements on a computer screen. Feedback was also presented in the computer-based condition. However, only the CAI group performed significantly better than they did on the pre-test, in comparison with controls who did not actively participate while completing the assigned reading. Although this experiment made an attempt to systematically compare techniques, the presentation of feedback to only CAI group allowed more than one factor to vary (answer sheet Vs computer and no feedback Vs feedback) and made it difficult to draw definite conclusions.

9.4.3 Outcome Variables

Another critical component of a good research design is the selection and measurement of outcomes variables. The choice of the optimal dependent variable or variables to be used in the evaluation of educational technology has been a subject of much debate.

Outcome measures are necessarily a function of the goals of the software's developers and users. Although test performance is a reasonable measure of acquired information, the performance assessment structure of most college courses makes the use of this measure alone inadequate. Hannafin & Carney (1991) conducted an investigation into the effect of review strategies that were designed to influence student's cognitive processing of the content of an instructional software package. The effects of behavioural strategies emphasising recall of information presented in each section were compared with the effects of cognitive review strategies designed to promote meaningful integration with existing schemata. No differences were found in the college students' performance on a 24-item, short answer test that contained both factual and inferential questions. There were differences, however, in the other dependent variables: (1) The total number of words used in answers, a measure of the quality of the response, was significantly greater in the cognitive condition; (2) the number of personal elaborations, a measure of *far-transfer* of information, was also greater in the cognitive condition; and (3) the behavioural strategy resulted in significantly more ideas related to the specific content of the program (a *near-transfer* measure). This study made a dual contribution to the issue of evaluation by focusing on the use of different cognitive instructional strategies and by broadening the type of dependent measure utilised.

There are, of course, other outcome measures of interest. It is important to understand the effect of CAI on students' attitudes toward computer-based learning and toward the particular software package employed. A number of studies (Cordell, 1991; Duncan, 1991) have included this outcome measure. Most of them have reported positive attitudes.

Until now, most research on the efficacy of educational software has been experimental or quasi-experimental. Experimental studies in which computer-based learning is compared with other instructional techniques have been productive methods of investigation (Kulik & Kulik, 1991; McNeil & Nelson, 1991).

9.4.4 Sample Size Determination

In any experimental design, the number of subjects, called sample size, is a very important factor. While testing any hypothesis on a sample, the sample size should be large enough to ensure that the differences in results are large enough and are acceptable to reject the fact that the variability of sample is resulting that difference. The sample size should be such which can give a high chance of detecting, as statistically significant, a worthwhile effect, if it exists, and thus to be reasonably sure that no such benefit exists if it is not found in the testing. Many of the studies described in the section 9.4 of this chapter had small sample size and hence had low chances of detecting differences. Most of these studies concluded with no significant differences which may be attributed to their small sample sizes.

Sample size is based on the quantity called standardised difference, which is the ratio of the difference of interest to the standard deviation of the observations.

In the comparison of two methods of teaching the introductory subjects in the study, the difference in the means of gains obtained by students is used as a basis for comparing two teaching approaches. A real difference of 10% in means of gains is taken as representing an important difference between the performance of two teaching methods.

As is shown in the next chapter, the standard deviation in our study is varying between 7.4 and 14.7. Therefore, with the maximum value of standard deviation as 14.7, the value of standardised difference comes out to be 0.68. According to the Nomograph for calculating sample size or power provided by Altman (1991), the power level of 0.95 can be achieved with a total sample size of 110 for significance level of 0.05. This value of power is large enough to draw firm conclusions. In our study, the total sample size for all packages are well above 110 students.

9.5 Evaluation Techniques Used in the Project

As discussed in this chapter, subject based evaluation techniques have been found to be effective techniques for CAL evaluation. They are based directly on the user's judgement and data collection is possible under laboratory conditions. The data is less open to bias if the collection method has carefully been chosen. This research, therefore, employed these techniques for evaluation of in-house developed software. Two types of subject based evaluation techniques were used: questionnaires and observations. Since questionnaires contained both structured and open ended questions, it was quite easy to elicit large amount of specific information quickly and easily. Also, users were free to provide detailed opinion about the packages in open-ended questions. Students were also observed by the author and respective tutors/technicians of various institutions, the collected information provided a great deal of insight about the students' responses towards the navigational procedures of packages and other interface related matters. Since it is difficult to establish the effectiveness of a computer aided learning system simply by observing, specially when study is taking place at many institutions at the same time, this method of evaluation has been kept as a supplementary to the questionnaire method.

A two group trial study was used in this research for the evaluation of in-house packages as it provides the researcher necessary controls over what happens to the participants. The researcher can influence events and investigate the effects of the intervention. Since stronger inferences can be made from experimental studies than from observational studies, experimental studies have been used for this research.

The object of the proposed research was to evaluate the interface of various in-house developed CAL modules for introductory subjects. This requires a comparison between the progress of students who use these modules with those who did not. Therefore, it was essential to use a control group study in which CAL group students could be compared with traditionally studied students.

The research questions addressed in this thesis for statistical analysis, are as follows:

1. Are the gains in students procedural knowledge of subject matter from each intelligent tutoring system comparable to traditional methods of tutoring for numeric components of subject matter?

2. Are the gains in students knowledge consistent across different packages?

3. Are the gains in students knowledge consistent across different institutions?

4. Is the Honey & Mumford Learning Style Questionnaire a valid instrument for the purpose of CAL evaluation? Are the gains in students knowledge consistent across students with different learning styles?

5. What are the views of student-users towards the interface and the integration of procedural skills of subject matter with the interface, for the in-house CAL modules, according to following factors:

   a) Gender
   b) Learning styles
   c) If they are previously computer trained
   d) If they have confidence in operating computers
   e) If they enjoy using computers

6. Are there any differences in the performance of students who did not have any previous computer training, who did not have confidence in operating computers and who did not enjoy computers, with those who had these attributes?

The in-house developed packages were evaluated in two phases. The table 5 shows various phases of the evaluation.

Phase	Activity	Analysis
Phase I (May 1995) **"Two group parallel trial study"** at one university on Capital Investment Appraisal module	* Pre test of both Control and CAL groups * Learning styles obtained from both groups via Honey & Mumford LSQ * Teaching of Control group by an experienced teacher in traditional manner * Exposure of CAL module to CAL group students * Post testing of both groups * Subjective questionnaires completed by CAL group students * Observations were taken by the author	Comparative study of two groups provided: * Initial validation of all aspects of CAL modules * Initial validation of the interface design of CAL modules from subjective questionnaires * Validation of measurement techniques used * Validation of questionnaire design
Phase II (Session 1995-96) **"Two group parallel trial study"** at two universities and CAL exposure at four other universities on Capital Investment Appraisal, Absorption Costing and Marginal Costing modules	* Prospective study * Two group study at two universities on Capital Investment Appraisal module * CAL exposure at other universities on a random sample of about 40 students each (Capital Investment Appraisal at 2 universities, Absorption Costing at 3 universities and Marginal Costing at 4 universities) * Observations were taken by the author and the tutors at various universities	The phase II study gave insight to the research questions raised in this project.

Table 5. Various phase of evaluation

The First Phase evaluation took place at one university where Capital Investment Appraisal CAL module was evaluated in a two group parallel trial. Control group students were taught by an experienced traditional teacher whereas the CAL group was exposed to the Capital Investment Appraisal ITT. Both groups were pre and post tested and their learning styles were acquired by the Honey & Mumford Learning Style Questionnaire. Group comparison provided the initial validation of all aspects of ITT, whereas the observations and subjective questionnaire feedback from CAL group validated the interface design adopted for all in-house developed modules. This study also provided a validation of the measurement techniques and the design of the questionnaire. The data collected from the study is analysed in next chapter.

Phase II study was the prospective study, which took place at various institutions and all three CAL modules, Capital Investment Appraisal, Absorption Costing and Marginal Costing were used by different groups of students for this purpose. A two group study was organised at two universities on Capital Investment Appraisal module. At other institutions, the testing of ITTs was done on a random sample of about 40 students, as it was not feasible to test all students at other institutions. The study gave insight to the research questions raised in this project.

The analysis and conclusion obtained from the collected data of various studies is summarised in the next chapter. Various questionnaires used in this study are enclosed in the appendices for reference. These are:

Appendix C: Capital Investment Appraisal Module - Pre and Post Test
Appendix D: Absorption Costing Module - Pre and Post Test
Appendix E: Marginal Costing Module - Pre and Post Test
Appendix F: Learning Style Questionnaire
Appendix G: Subjective Questionnaire

The pre test questionnaires for each package is designed to identify the level of student knowledge of each area before formally teaching that subject in this study. All questions in the questionnaires need calculations to be recorded on the questionnaires, which helps to investigate the correctness of the method used by the student in solving the question. The questionnaires start with very simple questions which can be solved even by common-sense, and gradually progress towards more difficult subject-specific questions.

The post test questionnaires contain only subject-specific questions and are designed solely to find out the level of student knowledge of corresponding subject matter. Again, students are required to show all calculation work on questionnaires to provide information on the correctness of the methods used by students in solving various problems. The questionnaires start with simple formula based questions and gradually progress towards more decision-based questions.

The same subjective questionnaire is used for all packages. It is divided into three parts and contains a total 129 questions. The first part contains information regarding biographical data of users. Second part is related to the users' experience with general computing. The information provided in these two parts provided the basis for the division of students into various subgroups according to their background. This ensured the homogeneity of the students in a particular subgroup. Basically these parts of the questionnaires provide following information:

- Name, course and university
- Age
- Present occupation
- Gender
- Experience with computer at home or at office/university
- Any computer related training
- Any experience with any type of CAL packages

- Any experience with CAL packages in accounting

The third section of the questionnaire is the assessment of the tutoring system. It contains 113 closed-ended statements and one open-ended question. This section provides information relevant to the evaluation of the interfaces of the CAL modules. It consists of two types of statements:

1. "Purely interface" related statements
2. "Integration of interface with subject matter" related statements

Each closed-ended statement is either in favour or not in favour of the package. The number of favoured statements are almost equal to the not-favoured statements (44% favoured statements and 56% not-favoured statements), so that the questionnaire is unbiased and balanced. All statements have five point agree-disagree Likert Scale (figure 14), which facilitates an easy and reliable analysis.

1	--------	2	--------	3	--------	4	--------	5
Strongly Agree		Agree		Uncertain		Disagree		Strongly Disagree

Figure 14. Likert Scale

Instead of providing five boxes with each statement with facility of ticking the desired box, only one box is provided so that students are required to fill the box with correct number. This forces students to think about the choice and they cannot just tick the box in a serial fashion (without even reading the statement!).

The open-ended question is related to students' opinion about the strengths and weaknesses of the packages and recommended improvements in their own words.

Once the data is collected from the evaluation studies, appropriate statistical analyses need to be completed. The next section describes the techniques utilised in this project.

9.5.1 Statistical Analysis

The data obtained from the students was in the form of pre and post tests, learning style questionnaires, subjective questionnaires and observations. Since it was not possible to obtain the *observation data* in a prescribed form from different institutions, it was used only to see a general trend, and was not used in statistical analysis. To address the research questions raised in this thesis, various statistical procedures were applied on the collected data.

The table 6 shows a summary of the statistical analysis used in this study. Initially, percentage gains were obtained for different institutions, where the two group trial study was carried out. 2 way ANOVA analysis was applied to percentage gain data to investigate whether gain in student knowledge was consistent for both teaching methods. Interaction between modes of instruction and centres was also analysed and Least Significant Difference method was employed to investigate which centres have significantly different results (Altman, 1991).

Then, a 2 way ANOVA was employed to investigate the consistency among the percentage gains obtained by the students for different packages at various centres where in-house CAL packages were used.

To investigate the effect of learning styles of students on their percentage gains for different packages at various centres, 2 way ANOVA procedure was employed for all CAL centres dividing them according to the packages used. Interactions was also analysed among learning styles and centres. Where the interaction was found significant, the Least Significant Difference analysis was carried out to investigate which centres are significantly different.

Data	Analysis	Results
Learning Style Questionnaires	These questionnaires were used to obtain the learning styles of students.	These learning styles were subsequently used for statistical analysis with Pre and Post Tests and Subjective Questionnaires to investigate the research questions raised in this thesis.
Pre and Post Tests	* Percentage gains (Post test results - Pre test results) for institutions with two group trial study * 2 way ANOVA analysis on percentage gains allowing for the centre effect * Interaction between modes of instruction and centres * Least significant Difference analysis to find the centres with significantly different results	* The consistency of students knowledge gain for both teaching methods
	* 2 way ANOVA analysis on percentage gains for different packages at all CAL study centres, allowing centre effect	* The consistency of students knowledge gains for various CAL modules
	* 2 way ANOVA analysis on percentage gains for different learning styles, allowing for centre effect * Interactions among learning styles and centres * Least Significant Difference analysis for the significant interactions	* The consistency of students knowledge gains for different learning styles
Subjective questionnaires	* The division of the questionnaire into various groups according to centres and packages * Mantel-Haenszel chi-square statistic on following parameters: ➢ Gender ➢ Learning styles ➢ Previous computer training ➢ Confidence in operating computers ➢ Enjoying computers	* To investigate students views towards the packages according to various parameters * The consistency of students views towards the packages across different packages

Table 6. Statistical analysis used in this study

To investigate the student-users views towards the packages, the data collected in the form of subjective questionnaires was analysed. The questionnaires were divided into various groups according to centres and packages, and since the data was categorical, Mantel-Haenszel chi-square test was used to analyse any differences in the views of the students allowing for centres according to following parameters:

a) Gender
b) Learning styles
c) If they are previously computer trained
d) If they have confidence in operating computers
e) If they enjoy using computers

9.6 Summary

This chapter focused on the importance of evaluation of Computer Aided Learning packages. The importance of initial laboratory testing phase of the evaluation was discussed and necessity of quantitative evaluation methods to analyse the overall effectiveness of the modules was emphasised. Various evaluation methods were discussed and subject based evaluation was identified from the literature as most widely used method for evaluation of CAL packages. The chapter also discussed various evaluation techniques and pointed out various problems associated with research on efficacy of CAL software packages. Finally, the evaluation and statistical techniques used in the Byzantium project were explained.

The next chapter deals with the analysis of data obtained in the evaluation of Byzantium ITTs. The summary of statistical analysis and justification of techniques used is also dealt with and conclusion of the study is also presented.

Evaluation of Byzantium CAL Modules

The evaluation was conducted at six UK institutions using the Capital Investment Appraisal, Absorption Costing and Marginal Costing software packages. A pilot study was completed at one university in phase I on Capital Investment Appraisal package. Subsequently all three packages were used at various institutions.

Although the same interface design is used in all three packages, there are differences in the nature of the subjects, which may attribute towards the students' preferences for one package to another. For example, in Marginal Costing, almost all variables are related to each other. Therefore the system needs to provide a whole picture of the problem. There are no iterations in the processes. In Capital Investment Appraisal, there are three different techniques available to solve a problem and none of techniques are dependent on each other. Therefore any of the techniques can be used without utilising the others. This module is an example of a composite model. In Absorption Costing, there are strong dependencies amongst the variables. The process is sequential and functions are broken down into small calculations for ease of learning. The results of one part are used to calculate values in other parts of the problem. There is some iteration and many variables need re-calculation.

Four universities out of six (universities A, B, D and E) are recent polytechnic-converted-universities, and the other two (universities C and F) are traditional universities. One new university (university E) used the packages in their open learning programs, whereas at other universities, the packages were used in general tutorial settings.

Table 7 lists the number of students who participated in the evaluation at the different universities.

	Capital Investment Appraisal	*Absorption Costing*	*Marginal Costing*
University A (I Phase)	Trad. Students=40 CAL Students=40		
University A (II Phase)	Trad. Students=40 CAL Students=40	CAL Students=38	CAL Students=38
University B (II Phase)	Trad. Students=41 CAL Students=40		CAL Students=38
University C (II Phase)			CAL Students=39
University D (II Phase)		CAL Students=38	CAL Students=38
University E (II Phase)	CAL Students=41	CAL Students=40	
University F (II Phase)	CAL Students=42		

Table 7. Summary of sample sizes at various universities

10.1 Information Gathered From Users

The following information was gathered from the students in the research study:

1. Pre Test
2. Learning Style Questionnaire
3. Post Test
4. Subjective Questionnaire
5. Observations

10.2 Analysis of Percentage Gains

1. Two way ANOVA analysis was applied to percentage gains of pre and post test results of the two group parallel trial study on Capital Investment Appraisal package at two universities. At one university, this study was

carried out in two phases (code 1 and 2). The first phase study was a pilot study. Table 8 shows the mean percentage gains of various universities.

Universities	*Traditional Teaching*	*CAL Teaching*	*Total mean gain*
1. University A (Phase I)	52.1	55.3	54.0
2. University A (Phase II)	65.2	66.8	66.0
3. University B (Phase II)	65.0	66.1	65.6
Total mean gain	60.8	62.7	

Table 8. Mean percentage gains for various two group parallel trial studies

The results of ANOVA analysis are as follows:

Source	DF	ANOVA SS	F Value	Pr > F
MODE	1	220.77802485	1.06	0.3045
CENTRE	2	7833.43015790	18.79	0.0001
MODE*CENTRE	2	55.23924312	0.13	0.8760

For p=0.05, the above analysis can be summarised as follows:

a) The difference between the gains of traditional and CAL teaching is not significant.

b) The difference between the gains of various centres is significant.

c) There is no significant interaction between teaching modes and centres.

Since the two group trial study was completed at two universities in which one university conducted the study twice, first in phase I and then in phase II, least significant difference test was employed to investigate whether or not the difference is between the universities or between phase I and II. The results of the analysis are as follows:

Comparisons significant at the 0.05 level are indicated by '***'.

Comparison	Lower Confidence Limit	Difference Between Means	Upper Confidence Limit	
2 - 3	-4.014	0.470	4.954	
2 - 1	7.838	12.336	16.834	***
3 - 2	-4.954	-0.470	4.014	
3 - 1	7.382	11.867	16.351	***
1 - 2	-16.834	-12.336	-7.838	***
1 - 3	-16.351	-11.867	-7.382	***

The results above, clearly show that phase I gains at university A are significantly different from other gains. This concludes that the differences are between the gains of phase I and phase II evaluation studies. This difference from phase I and phase II can be attributed to the changes in the package after phase I study. There were various suggestions for improvement observed in phase I study and subsequent major modifications include re-structuring of *help* section, inclusion of *beep* and *print* buttons, and changes in feedback messages.

2. Once it was established that there is no significant difference among the percentage gains of different universities except for the results of phase I, the analysis was extended to all the universities in phase II where CAL study took place. 2 way ANOVA analysis was employed on all these centres to analyse any differences present among the percentage gains for different packages. The table 9 shows the mean percentage gains.

Universities	*Capital Investment Appraisal*	*Absorption Costing*	*Marginal Costing*	*Total mean gain*
1. University A	66.8	64.2	81.1	70.7
2. University B	66.1		84.6	75.1
3. University C			69.2	69.2
4. University D		69.6	79.8	74.7
5. University E	67.2	65.6		66.4
6. University F	68.0			68.0
Total mean gain	67.1	66.4	78.6	

Table 9. Phase II studies of CAL

The results of the analysis are as follows:

Source	DF	ANOVA SS	F Value	Pr > F
PACKAGE	2	13897.2231881	69.96	0.0001
UNIV	5	4628.5081648	9.32	0.0001
PACKAGE*UNIV	3	1182.8153323	3.97	0.0083

The above analysis shows that the differences in percentage gains are significant among different packages and among different centres and there is significant interaction between packages and centres.

To find out, for which packages, the differences in gains are significant, the least significant difference test was applied on packages. The results are as follows:

Comparisons significant at the 0.05 level are indicated by '***'.

Comparison	Lower Confidence Limit	Difference Between Means	Upper Confidence Limit	
3 - 1	9.393	11.598	13.803	***
3 - 2	9.789	12.201	14.613	***
1 - 3	-13.803	-11.598	-9.393	***
1 - 2	-1.777	0.603	2.982	
2 - 3	-14.613	-12.201	-9.789	***
2 - 1	-2.982	-0.603	1.777	

The above analysis shows that the Marginal Costing package results are significantly different from other packages. This difference can be attributed to the difference of about 10% in percentage gains between Marginal Costing and other packages.

To find out, for which universities, the differences in gains are significant, the least significant difference test was applied on universities. The results are as follows:

Comparisons significant at the 0.05 level are indicated by '***'.

UNIV Comparison		Lower Confidence Limit	Difference Between Means	Upper Confidence Limit	
2	- 4	-2.755	0.403	3.560	
2	- 1	1.615	4.483	7.352	***
2	- 3	2.064	5.906	9.748	***
2	- 6	3.340	7.089	10.838	***
2	- 5	5.653	8.761	11.869	***
4	- 2	-3.560	-0.403	2.755	
4	- 1	1.190	4.081	6.972	***
4	- 3	1.645	5.503	9.362	***
4	- 6	2.920	6.686	10.453	***
4	- 5	5.230	8.358	11.486	***
1	- 2	-7.352	-4.483	-1.615	***
1	- 4	-6.972	-4.081	-1.190	***
1	- 3	-2.204	1.422	5.048	
1	- 6	-0.922	2.606	6.133	
1	- 5	1.441	4.277	7.114	***
3	- 2	-9.748	-5.906	-2.064	***
3	- 4	-9.362	-5.503	-1.645	***
3	- 1	-5.048	-1.422	2.204	
3	- 6	-3.173	1.183	5.539	
3	- 5	-0.963	2.855	6.673	
6	- 2	-10.838	-7.089	-3.340	***
6	- 4	-10.453	-6.686	-2.920	***
6	- 1	-6.133	-2.606	0.922	
6	- 3	-5.539	-1.183	3.173	
6	- 5	-2.053	1.672	5.397	
5	- 2	-11.869	-8.761	-5.653	***
5	- 4	-11.486	-8.358	-5.230	***
5	- 1	-7.114	-4.277	-1.441	***
5	- 3	-6.673	-2.855	0.963	
5	- 6	-5.397	-1.672	2.053	

The above analysis shows that the universities B and D are significantly different from universities A, C, E and F, whereas university A is significantly different from university E. The differences of universities B and D can be attributed to the higher percentage gains obtained by the students at these universities, and the differences between universities A and E can be attributed to the lower percentage gains obtained by the students of university E (a new university where the software was used in open learning programs).

3. 2 way ANOVA analysis was applied on percentage gain and learning style for all CAL centres dividing them according to the packages used. The following tables show mean percentage gains for each centre for each package.

Capital Investment Appraisal

Universities	*Activist*	*Reflector*	*Theorist*	*Pragmatist*	*Mixed*	*Total mean gain*
1. University A	60.1	65.1	63.8	70.4	68.2	66.8
2. University B	67.8	58.2	69.4	64.2	67.6	66.1
3. University E	72.2	69.9	66.6	64.2	66.2	67.2
4. University F	67.8	69.3	72.2	49.6	69.6	68.0
Total mean gain	65.9	66.9	67.2	63.1	68.1	

Table 10. Gains for various learning styles for Capital Investment Appraisal package

The results of the analysis are as follows:

Source	DF	ANOVA SS	F Value	Pr > F
L_STYLE	4	383.33256155	1.09	0.3617
CENTRE	3	79.94751832	0.30	0.8222
L_STYLE*CENTRE	12	1684.09810267	1.60	0.0969

The above analysis shows that there is no statistically significant effects.

Absorption Costing

Universities	*Activist*	*Reflector*	*Theorist*	*Pragmatist*	*Mixed*	*Total mean gain*
1. University A	61.3	59.4	73.0	65.0	65.4	64.2
2. University D	70.5	68.4	71.0	69.7	69.3	69.6
3. University E	66.8	68.6	69.8	58.7	63.7	65.6
Total mean gain	64.9	66.6	70.9	63.9	66.3	

Table 11. Gains for various learning styles for Absorption Costing package

The results of the analysis are as follows:

Source	DF	ANOVA SS	F Value	Pr > F
L_STYLE	4	366.67337794	1.36	0.2547
CENTRE	2	615.52682153	4.55	0.0128
L_STYLE*CENTRE	8	447.67601183	0.83	0.5803

The analysis shows that there is significant difference among the results of various universities, but no significant difference in gain of students with different learning styles.

To find out, for which universities, differences in gains are significant, the least significant difference test was applied on universities. The results are as follows:

```
Comparisons significant at the 0.05 level are indicated by '***'.

                        Lower      Difference     Upper
                     Confidence     Between     Confidence
       Comparison       Limit        Means         Limit

      2     - 3         0.397        4.065         7.734     ***
      2     - 1         1.758        5.474         9.189     ***

      3     - 2        -7.734       -4.065        -0.397     ***
      3     - 1        -2.260        1.408         5.077

      1     - 2        -9.189       -5.474        -1.758     ***
      1     - 3        -5.077       -1.408         2.260
```

The above analysis shows that university D is significantly different from other universities.

Marginal Costing

Universities	*Activist*	*Reflector*	*Theorist*	*Pragmatist*	*Mixed*	*Total mean gain*
1. University A	66.7	81.3	88.9	79.6	79.5	81.1
2. University B	91.7	83.3	88.1	86.7	81.6	84.6
3. University C	75.0	63.3	66.7	66.7	68.6	69.2
4. University D	66.7	78.6	83.3	81.7	80.4	79.8
Total mean gain	76.9	77.3	84.5	79.3	77.5	

Table 12. Gains for various learning styles for Marginal Costing package

The results of the analysis are as follows:

```
Source                 DF          ANOVA SS      F Value        Pr > F

L_STYLE                 4      638.36265157         1.23        0.3028
CENTRE                  3     5115.84915720        13.10        0.0001
L_STYLE*CENTRE         12     1633.04279479         1.05        0.4116
```

The analysis shows that the differences among the gains of various universities are significant but there is no significant difference in gains of students with different learning styles.

To find out, for which universities, differences in gains are significant, the least significant difference test was applied on universities. The results are as follows:

Comparisons significant at the 0.05 level are indicated by '***'.

Comparison	Lower Confidence Limit	Difference Between Means	Upper Confidence Limit	
2 - 1	-1.674	3.509	8.692	
2 - 4	-0.358	4.825	10.007	
2 - 3	10.269	15.418	20.568	***
1 - 2	-8.692	-3.509	1.674	
1 - 4	-3.867	1.316	6.499	
1 - 3	6.760	11.910	17.059	***
4 - 2	-10.007	-4.825	0.358	
4 - 1	-6.499	-1.316	3.867	
4 - 3	5.444	10.594	15.743	***
3 - 2	-20.568	-15.418	-10.269	***
3 - 1	-17.059	-11.910	-6.760	***
3 - 4	-15.743	-10.594	-5.444	***

The above analysis shows that university C is significant different from other universities with mean percentage gains less than the other three universities.

10.3 Subjective Questionnaire

Mantel-Haenszel chi-square statistic was applied on subjective questionnaire results for various parameters such as gender, students' learning styles and their attitude towards computers. The analysis is enclosed in appendix H. The summary of results is as follows:

The overall response of the users towards the system was mixed, although most of the students agreed that the packages do not require any prior knowledge of accounting and computing. However, there were less favourable results in terms of the student view as to how well the software compares to other traditional medium of learning, whether it provides helpful error messages and whether or not the software is easy to follow. Table 13 shows students' opinions towards various packages.

Questions	Capital Investment Appraisal	Absorption Costing	Marginal Costing
	% Agree	% Agree	% Agree
1. The software does not require any prior knowledge of computing.	67.5	72.4	70.9
2. The software compares well with other traditional medium of learning.	42.3	44.0	44.4
3. The software does not impose a rigid structure of learning.	46.6	51.7	51.7
4. The software gives good knowledge of the topic covered.	59.3	62.1	58.3
5. The software does not require any prior knowledge of accounting.	56.8	58.6	56.3
6. The software provides helpful and meaningful error messages.	42.9	43.1	59.6
7. The software provides helpful interactive guidance to individual students.	51.5	53.4	55.0
8. The software instructions are clear and easy to follow.	43.6	49.1	50.3
9. The software lets the students learn at their own pace.	59.5	65.5	66.2
10. The software enables good learning and retention of what is learnt.	55.2	60.3	58.9
11. The software has an adequate help facility.	57.1	58.6	57.6
12. The software is easy to follow and uses instinctive and unambiguous controls.	28.8	29.3	30.5
13. The software provides a useful support facility through its calculator.	52.8	50.9	61.9
14. The software lays out information in clear and meaningful way.	72.4	72.4	76.0
15. Overall perception of the user about this system is positive.	53.7	52.6	53.0

Table 13. Students' opinion towards various packages

The results of the analysis of the association of students' view with the factors, gender, learning style, computer exposure and confidence are given below.

a. ***Gender***

The only significant association of user opinion and gender was for the Marginal Costing package where the male students were less supportive to the question that the software provides a useful support facility through its calculator.

b. ***Learning styles***

There were no significant associations of user opinion and learning styles for any of the packages.

c. ***Computer training***

The Mantel-Haenszel test for associations between user response and computer training revealed only one significant association for Capital Investment Appraisal package. Students with no previous computer training were less supportive of the statement that the software does not require any prior knowledge of accounting. No significant associations were found for any other package.

d. ***Confidence in operating computers***

The significant associations of user opinion and confidence in operating computers were found for Capital Investment Appraisal package. The students with no confidence in operating computers were less supportive of the following statements:

"The software gives good knowledge of the topic covered."

"The software enables good learning and retention of what is learnt."

"The software has an adequate help facility."

"The software provides a useful support facility through its calculator."

e. ***Enjoying computers***

There were no significant associations between user opinion and computer enjoyment for Absorption Costing package. For Capital Investment Appraisal package, only one significant association was found. Students who do not enjoy computers were less supportive of the statement that the software has an adequate help facility. For Marginal Costing package, students who do not enjoy computers were again less supportive of following statements:

"The software compares well with other traditional medium of learning."

"The software gives good knowledge of the topic covered."

"The software has an adequate help facility."

4. Once it was established that the students who did not have any previous computer training, did not have confidence in operating computers or did not enjoy computers provided less supportive attitude towards CAL teaching, analysis was carried out to investigate if there is any difference between the performance of these students with those who had these attributes. The results of the analysis are enclosed in Appendix K. The analysis revealed that the performance of students with or without any previous computer training was not significantly different. There was no significant difference between the performance of students with or without confidence in operating computers and similarly the performance of students who enjoyed computers was not significantly different with those who did not enjoy computers.

5. *Responses to the open-ended question of subjective questionnaire*

Most of the students responded to the open-ended question of the questionnaire. There were three parts of the question. First part was related to the strengths of the packages, second was limitations of the packages, and third part was related to the recommendations of improvements in the packages.

A large number of students provided positive feedback towards the user interface of the packages and appreciated the error handling and ease of navigation. They found the packages easy to understand and use. Many students agreed that the layout of the screens made the programs easy to follow.

Many students provided the feedback about the problem in using calculator facility and suggested improvements in its interface. Some students wanted improvements in the saving routines of the packages.

The results of the open-ended questions are summarised in appendix I.

6. Observations

The students were observed at various institutions while using the in-house developed packages. Students' comments were also recorded by the author and respective tutors. Students generally did not have any problems in using the packages and could find their way through the programs without any external help, although some students had problems in saving their work on floppies. Students found it easy to learn the subject matter with the help of packages and some of them even complained about not having *internal rate of return* exercise in assignment section of Capital Investment Appraisal module, which they had practised in interactive section. The results of the observations are summarised in appendix J.

10.4 Summary

This chapter discussed the statistical analysis used in the evaluation of the in-house modules and the findings of the analysis. In general, there was no significant difference between the results of traditional teaching and CAL teaching methods. The results at different centres were different in terms of percentage gains obtained by the students. Students with different learning styles responded equally well to the in-house developed CAL packages.

Students' opinions towards the packages were also obtained in the form of subjective questionnaires and were analysed with respect to gender, learning styles and their background and attitudes with computers. It was found that the students who did not have any previous computer training, had no confidence in operating computers and did not enjoy computers were less supportive of the packages.

The findings of the analysis are discussed in more details in the next chapter which provides a summary of the whole thesis. Possible extensions of the research are also discussed in the next chapter.

Conclusions and Discussions

This chapter provides an overview of the research problems addressed in this thesis. The research issues are reviewed and the research methodology employed to address these issues is briefly re-described. The conclusions of this research are then presented and their effects on the implementation of computer aided learning packages in numeric subject disciplines are discussed.

11.1 The Objectives of the Research

In disciplines like accounting which contain a major practical component, learning of concepts requires a great degree of practice of procedural skills. Understanding of concepts is directly related to the application of concepts in these disciplines. The procedural skills and related basic concepts can be learnt in academic environment with the help of tutorials. The application of this basic knowledge in real world problems and understanding of complex concepts requires understanding of the situational interpretation of concepts which can be obtained through situated learning such as in-job training and apprenticeship, as discussed in chapter 6.

Traditional tutorials are the usual way of providing the basic procedural skills, but they are resource intense, and lack of funding has forced universities to seek other alternatives such as alternative week tutorials and increased reliance on self-study. This poses no great problem for well-motivated students, but does not ensure adequate knowledge transfer to remaining students (Patel & Kinshuk, 1996a).

This creates a real need for software tutors that are designed with a knowledge transfer emphasis and can provide intelligent tutoring by adapting their tutoring strategies and feedback according to individual user.

The accounting discipline, although quite rich in the use of software packages for learning purposes, lacks the software which can provide intelligent user-adaptable tutoring. The commonly used packages such as spreadsheets and databases, are mainly suited for a problem solving approach and do not ensure the learning of both procedural skills and the application of concepts for each individual student.

This thesis has provided an overview of in-house project Byzantium approach where ITT's act as an adjunct to traditional teaching. The work is one of the pioneering efforts to address the need for software tutors for the domain of accounting. These software tutors provide intelligent tutoring of procedural skills to facilitate the learning of conceptual knowledge. The students interactions with the interface are used to infer information about the student's learning of the domain concepts. Based on student's inferred understanding, the modules present the knowledge at different granularity levels via the interface. The software tutors adapt themselves to the need of individual students, as a human tutor does. They build a cognitive model of the student and update it progressively with student learning (see chapter 7). The modules were developed in Microsoft Visual C++ after a careful analysis of available development platforms, where the author's main contribution has been the conceptual interface design, interface development and integration of various routines into the programs.

The thesis has also examined the effectiveness of CAL software with particular consideration of in-house modules. In particular the effectiveness of Byzantium CAL modules, as an alternative to tutorials, for introductory numeric disciplines is considered. The laboratory testing has provided an initial validation of the approach adopted by project Byzantium. Since the study was completed across many institutions, it can be considered as near to a field trial.

11.2 The in-house CAL Development

The CAL modules developed in this study cater only for the numeric part of the subject matter. The students learn the conceptual knowledge of subject matter in the *Basic Concepts* part of the packages and then grasp the procedural knowledge of basic relationships of accounting by the handling of the numeric data in *Project* part of the module. Since the students are exposed to the problem scenario through a suitable interface and the system provides context sensitive dynamic feedback and flexible intermediate paths to the solution, learning of conceptual knowledge of the subject topic is facilitated to some degree. The *help* button available on the project screens gives access to the conceptual knowledge of the subject matter and provides deeper learning in context of focused queries.

The modules are intended to teach very basic concepts of accounting. They do not address any behavioural or human factors associated with the subject discipline. For example, in Capital Investment Appraisal module, though the student can produce four sets of figures for four competing projects, the concept of management attitude to risk is not addressed. However in the *Basic Concepts* part of the package, the students are advised of the need to consider human factors in decision making.

Hypertext and multimedia aids have not been used in the modules although their use in *Basic Concepts* part of the modules would have enriched the learning experience. It was found that hardware and software in institutional computer laboratories is not equipped to accommodate the multimedia at present and therefore, multimedia extensions were left for the future. It should be noted however that hypertext links, which allow users to deviate their attention from main problem to secondary learning issues, may lead the student to get lost in the hyperspace (Whalley, 1993).

The in-house packages developed as a part of this research infer the problem-solving approach adopted by the student by looking at the responses entered by the student. It is possible that the system may incorrectly infer the type of mistake made by the student. If the student has got the correct value by the application of wrong relationship, the system may infer it as the application of correct relationship. On the other hand, if the students has employed correct relationship but made a calculation error and got wrong value, the system will infer it as the application of wrong relationship and may infuriate the student by advising to use the relationship which was used by the student at first place.

The packages intend to teach only a small amount of subject matter and do not provide knowledge about the other alternatives available. For example, the Absorption Costing package does not provide information on Activity Based Costing techniques which has received much attention recently.

11.3 Evaluation and Analysis of in-house CAL Modules

Once the packages were developed, evaluation studies were designed and conducted at various institutions in the UK. Although mainly quantitative evaluation techniques were used in the study due to their feasibility at various institutions, some qualitative views of the students were also obtained to get an insight of general trends of students' opinion towards CAL modules. A two group parallel trial study was carried out at two institutions and the CAL packages were also evaluated at four other universities. The students' knowledge of the subject was measured with the help of pre and post tests. Pre and post tests comparisons provided the measurement of mainly procedural skills although one question in post test also tested the decision-making skills of students. The learning attitudes of students were obtained with the help of Honey & Mumford (1986) learning style questionnaires. The information about students' background and their opinion towards the packages were obtained through subjective questionnaires. Students were observed by the author and respective instructors/technicians at various institutions while using the CAL modules.

The collected data was analysed with the help of various statistical techniques. Since the qualitative data collected from various institutions was not uniform, the statistical analysis techniques were employed only on quantitative data. The percentage gains resulted from pre and post tests were analysed through 2 way ANOVA procedure taking into consideration of any interactions. Significant results were identified using least significant difference test. The effects of learning styles of students on their gains were also analysed.

The data collected through subjective questionnaires was analysed using Mantel-Haenszel chi-square statistics to investigate students' opinion towards the interface design of the packages and the integration of subject matter with the interface of the packages. Students' opinions were analysed with regard to various attributes such as gender, learning styles of students, and their attitude towards computing. Conclusions from these analyses are discussed in the next section.

11.4 Conclusions

Besides design and development of CAL modules, a key issue this research addressed is, whether the Byzantium approach to CAL in accounting provides a means of teaching procedural skills that is of comparable effectiveness to human tutor.

The study presented in this thesis reveals that the means of percentage gains obtained by the students of traditional teaching group (mean = 60.8) and CAL teaching group (mean = 62.7) are almost equal and statistically, the difference in results are not significant. Since the sample size of this study gives power level more than 0.95, the study suggests that the traditional and CAL tutoring methods are comparable. The pre and post tests mainly tested the procedural skills of the subject matter, though one question in post test examined the decision skills of the students.

The research work also focused on the differences in the knowledge transfer due to the nature of the subject matter. The means of percentage gains of students were found statistically significantly different for different packages (Capital Investment Appraisal = 67.1, Absorption Costing = 66.4 and Marginal Costing = 78.6). This difference may be attributed to the nature of the subject matter covered in these packages (Kinshuk, 1995). Capital Investment Appraisal is an iterative topic with various independent techniques to solve the problem. Absorption Costing contains strong dependencies amongst the variables and is sequential in nature. In Marginal Costing, fewer variables are involved and these are closely related to each other. This allows the whole problem to be displayed on one screen and the user visually maintains a full picture of the problem. Unlike Marginal Costing, Capital Investment Appraisal and Absorption Costing packages spread out the problem questions into a number of screens by breaking the problems into small parts for ease of learning. But this increases the cognitive load on students as they have to relate the variables of a particular screen with the screens which are not visible. Though critical variables may be reproduced on the current screen, a novice user has to retain a mental map of the variables in the previous screen to maintain the semantic link and may have to move between the screens to refresh this link until the critical variables are fully grasped.

Another important difference which favours the Marginal Costing package, is the smaller amount of overall information students have to process as compared to Capital Investment Appraisal and Absorption Costing packages. Marginal Costing has only 14 variables in total, as compared to 48 variables in Capital Investment Appraisal and 114 variables in Absorption Costing. This has a psychological effect on students' performance as they perceive the relative size of the problem to be considerably larger in Capital Investment Appraisal and Absorption Costing as compared to Marginal Costing. These reasons may explain better performance of students in Marginal Costing package compared to other packages.

The efficacy of the in-house CAL modules also need to be studied in relation to the varying curriculum structures of different institutions to investigate whether the CAL modules are equally effective in different academic environments. The study revealed that there are differences in the students' percentage gains across different institutions. The gains were found higher in ex polytechnics than in traditional universities except one ex polytechnic, where open learning took place. In ex polytechnics, where the students receive training for day to day real world problems and have better grasp of procedural skills, the in-house modules may provide additional enhancement in the learning process due to their procedural nature. In the traditional universities, where the emphasis is more on conceptual knowledge transfer than practical skills, the modules provided knowledge acquisition comparable to traditional tutorials, but not as much as they did in ex polytechnics.

A further research question examined was whether the modules were equally effective for different students. Students were categorised in different groups for this purpose with the help of Honey & Mumford Learning Style measurement instrument. The allocation of students to various learning styles was a difficult task, since a large number of students had more than one predominant learning styles. This resulted in creation of group called 'mixed' which was larger than any other group.

The Honey & Mumford approach seeks to divide the students into four distinct groups of learning styles, whereas in this study, it was found that the students may have more than one predominant learning style. The Honey & Mumford approach does not cater for this situation, and hence the instrument was not found very useful in context of CAL in accounting.

In this study, the differences among the gains of students with different learning styles were not significant. Since small number of students were allocated to the individual categories of learning style and most of the students fell into the 'mixed' group, these findings are almost certainly affected by small sub-sample sizes.

The evaluation of in-house CAL modules sought the views of student-users in the form of subjective questionnaire. The statistical analysis of the information available through the questionnaires provided an insight into the students' opinion towards the interface of the modules and towards the integration of subject matter with the interface. The analysis was carried out by categorising the students in various groups according to their background attributes such as gender, learning styles and computer background.

The study revealed that there were no significant differences in the opinion of students of different gender and learning styles. However, the study revealed that the students who did not have any previous computer training, had no confidence in operating computers and did not enjoy computers were less supportive of the packages. These students are likely to be either not familiar with the general computer operations or if familiar, not interested in these operations. This may adversely affect the students operation of the CAL software, since any Windows application demands at least some familiarity with the Windows environment and basic operations needed to run a program. This suggests the need for some basic IT training before CAL software is introduced.

11.5 Discussion

This thesis provides insight into the design, development and evaluation of CAL modules as an adjunct to traditional teaching and an alternative to resource-intense tutorials for teaching numeric part of three topics of management accounting: Capital Investment Appraisal, Absorption Costing and Marginal Costing. The study validates the approach adopted for the tutoring strategies in which the modified feedback is provided for individual users by inferring information about them through their interactions with the in-house CAL modules interface.

The evaluation of in-house modules comprised mainly quantitative methods which provided validation of the CAL approach for the acquisition of procedural skills and related basic concepts. Further research is necessary to employ qualitative approaches for more detailed analysis of usability of particular aspects of the modules. This could provide an insight of how the students use the software in the real academic environment, and how their behaviour changes with the application of CAL modules. This analysis could provide insight into the long term effects of the CAL modules on educational environment.

Action type research may be also useful in examining the effectiveness and usability of various parts of the modules allowing designers/developers to pinpoint the areas which have less effect or perhaps negative effect on learning. These areas may not be identified by a quantitative analysis based on some simple overall measure of effectiveness.

The thesis presented an approach to the provision of intelligent tutoring of procedural skills in accounting. There are few examples in literature of operational for intelligent tutoring in accounting, and only recently, has software been developed for the accounting domain. As discussed in chapter 6, there are two known examples of intelligent tutoring systems to date, BITE and in-house Byzantium project. Whereas the Byzantium approach provides procedural skills and some related conceptual knowledge, the BITE project seeks to provide the learning of concepts through the metaphor approach. The accounting discipline is predominantly a practical discipline but requires both procedural skills and the knowledge of concepts. Project Byzantium concentrates on developing practical skills.

Real world applications in accounting demand the grasp of more complex concepts in decision making. This requires the understanding of the applicability of concepts to different situations. Metaphors are the ideal way to teach at this level as they provide direct learning rather than acquiring knowledge through 'learning by experience'. The BITE project is based on this view of knowledge transfer and puts more focus on acquisition of conceptual knowledge. Both approaches have a role in accountancy training.

The findings of this thesis may prove useful for intelligent tutoring systems in other numeric disciplines since the approach has shown successful implementation for three different numerical topic areas. The approach has also proved successful for *Spring Design* in mechanical engineering discipline (Patel & Kinshuk, 1996b; Patel & Kinshuk, 1996d). The approach should be most suitable to disciplines where the relationships among various concepts is largely straight forward, and the application of concepts is mostly numeric. Procedural skills are then necessary to understand the concepts and numeric values associated with the concepts provide a new interpretation of the concepts, just like in situated learning.

11.6 Byzantium Project - Learning from Experience

The design, development and evaluation of Byzantium CAL modules has provided insight into the kind of problems in the CAL development and implementation for numeric disciplines and a number of lessons can be learned from this research.

The development of Byzantium modules was started in Clipper programming language but soon was transferred to Microsoft Visual C++ to enable greater flexibility provided by the Windows environment. The choice of this platform has enabled the development of user interfaces to provide the novice user with necessary functionality and required ease of use while keeping an open the path for future developments. This suggests that the CAL development requires a development platform which should provide the necessary flexibility at programmer level to provide facilities and ease of use required at novice user level.

The design of in-house CAL modules also required provision of small utilities such as 'print' and 'beep' buttons which, although they do not add much to the interface, provide great comfort to the user and hence give a psychological effect towards the appreciation of the CAL modules.

The mixed-initiative approach adopted by the modules allows the students to choose alternative paths towards the solution of the problem. The system is then able to infer students' knowledge from their interactions through the interface and to provide appropriate tutoring strategies and feedback to individual students.

The domain knowledge is presented in the modules in the form of relationships among concepts, and complex concepts are de-composed to their finer grain knowledge representation. This allows the system to infer from student's interactions, exactly at what location the mis-conception occurred. If the mis-conception occurs at any complex concept, the approach facilitates the system to advise the student to pursue at higher level of granularity, where the complexity of the concept is broken down in its components.

Analysis of evaluations data from this study revealed that the performance in gains were higher for Marginal Costing package, which had least number of inter-related variables and lower for other packages where the number of variables was quite large and the problems were divided into several screens to reduce complexity. This suggests that when the complexity of the problem is broken down into small calculations and divided into various screens, though it reduces complexity, it actually increases cognitive load as the student have to relate variables of one screen to another. This may result in missing or wrong inter-linking of information among various screens and hence reduce the performance of students. This suggests that the approach presented in this thesis should be avoided for very large problems. When the students fail to grasp the inter-relationships of variables between screens, some additional help is needed to reduce cognitive overload. This help may take the form of graphic display of the concepts already presented in the previous screens, or as a concepts map, where the students can refer and relate the currently presented concepts with previous screens. Some repetition of data from previous screens may also provide better understanding of the conceptual inter-relationships.

The evaluation of Byzantium modules also suggested that it is better to provide some basic IT training to the students who do not have any experience of computers before they are introduced to CAL software. This would give confidence to these students and would remove the fear of 'breaking' the computer. Once they have gained some confidence in basic operations, they may feel more at ease with CAL teaching and gain more out of it.

11.7 Future Research

Although the research in this thesis has investigated an important area of design, development and evaluation of intelligent tutoring in accounting, which was almost untouched till now, there are many research issues of interest, which could be the subject of further investigation. These are now discussed.

11.7.1 Integration of ITTs

The research in this thesis has divided the whole subject area of a particular discipline into three distinct knowledge levels as described in chapter 7. At the *introductory application* level, student learns the use of basic tools of the subject matter. The *advanced application* level aims at the integration of conceptual objects, and the *actual application approximation* level aims at the correlation of introductory tools while dealing with behavioural and environmental factors. The aims of second and third level are unproved at this stage. Further research in this area should take account of the limitation of the in-house approach (see section 11.5) that the effectiveness of the modules reduces considerably for the problems which are large and must be divided into several screens. The current research has been focused on the development of basic ITT modules and the definition of a methodology for their integration (Patel & Kinshuk, 1996d).

The future research could include the development of practical applications by integrating the basic ITTs. The integration can take place at two levels. At the **advanced application** level, the vertical and horizontal integration of conceptual objects can take place. *Vertical integration* would involve a comparison of the results of multiple use of the same tool e.g. comparing four different investment proposals; adding behavioural considerations like attitude to risk. *Horizontal integration* would allow use of multiple tools to solve a given problem viz. using marginal costing ITT to provide contribution figures, which are sought to be maximised in a linear programming ITT. Though the basic ITTs can be used for various sub-tasks, an intelligent application providing a suitable interface for (i) holding and comparing the results of multiple instances of an ITT and (ii) connecting various ITTs, would be able to guide a student through the whole task. This approach will require a careful design for effective student learning, as the effectiveness of in-house approach has been found to be best for small size domain problems comprising of a limited number of inter-related concepts.

The basic technical difficulty in the networking of the ITTs would be the standardisation of the communication interface and protocol for enabling message passing between the ITTs. Another difficulty would be to decide which ITT is to be used to solve a particular problem. This may require concept-based indexing (Brusilovskiy et. al. 1996) and a mechanism for adding new ITTs to such index as they evolve.

The **actual application approximation** level would aim to simulate a simplification of the real world problems. Here the students would need to learn how to account for behavioural and environmental factors. Networking of ITTs at this level would require that the *basic* and *advanced application* ITTs addressing the problem have the ability to handle qualitative, probabilistic and noisy data.

The three levels must be progressive and the overall complexity increase with level. It is expected that the modular approach of the ITT network can help in breaking down the complexity. The approach needs a careful evaluation to examine if it provides adequate understanding of the conceptual relationships. Since the ITTs used in the three levels do not provide the acquisition of 'all' concepts, the approach is still an adjunct to traditional teaching but is expected to replace tutorials at advanced stages of subject learning. Whilst it is a long term vision and the task is massive, the definition of communication standards among various ITTs would facilitate the development of many ITTs by different developers via the Internet. This will help in dealing with complex problems through part solutions provided by these ITTs. Work at a very early stage is being undertaken in this direction by the Byzantium project team.

11.7.2 Internet Portability

A further area of future research may take advantage of recently emerging Internet languages such as Java (Sun Systems Inc.) and ActiveX (Microsoft Inc.). The basic ITTs could be transported onto Internet by converting them into *applets* (executable objects created using Java which run on clients' computer) or *web-objects* (executable objects created using Java which run on host server). The main consideration here would be to clearly identify what proportion of application functions should be transferred and run on client's computer and how much should be kept on the host server, so that the whole application could be run with minimum traffic on network and minimum use of client's computer memory.

11.7.3 Evaluation Factors

The evaluation study in this thesis has been one of the largest quantitative studies conducted in the area of CAL in accounting, and has investigated the effects of a number of different factors such as gender, students' learning styles and their attitude towards computing. There is still a great need to investigate the deeper areas of cognitive aspects, such as the effects of visual-spatial ability of the students on their opinions towards the packages, specially the financial accounting modules where the amount of data is enormous, and the changes at one level of data structure affect many different levels of data.

11.8 Summary

Whilst there is an increasing number of simulation, hypertext and multimedia based tutoring systems becoming available and many of these systems enhance the learning experience, there is a need for software that enables the application of the principles learnt and reinforces the inter-relationships of the factors involved. The in-house ITT modules provide intelligent tutoring for procedural skills embedded in basic conceptual knowledge of the discipline by inferring information from student's interactions. The tutoring strategies and feedback are modified for each individual students just as an Intelligent Tutoring System does. These modules, therefore, are useful to disciplines where procedural skills are required with the basic domain knowledge before the application of concepts in real world situations is grasped. Disciplines, such as accounting and engineering, which comprise a large practical dimensions, are suitable for this approach. This research also provides a broad framework for the emergence of more complex tutoring systems from individual ITTs in a bottom-up fashion (Patel & Kinshuk, 1996b) as opposed to the top-down design generally adopted by traditional tutoring systems.

The evaluation of the ITTs has also been a major part of this research and after identifying a lack of multi-institutional evaluation studies in accounting field, this research provides a useful contribution towards the use of CAL in the numeric disciplines. A number of important research questions have been raised in the research and findings suggest that the students' attitude towards CAL learning for numeric disciplines is positive for the students who either have some previous computer training, or have confidence in operating computers or they enjoy working on computers.

References

- **Abouserie R. & Moss D.** (1992). Cognitive style, gender, attitude towards computer-assisted learning and academic achievement. Educational Studies, 18(2), pp151-160.
- **Alessi S. M. & Trollip S. R.** (1985). Computer Based Instruction. Prentice Hall, Englewood Cliffs, N. J.
- **Allinson C. W. & Hayes J.** (1988). The learning style questionnaire: an alternative to Kolb's inventory? Journal of Management Studies, 25(3), May 1988, pp269-281.
- **Allinson C. W. & Hayes J.** (1990). Validity of the Learning Style Questionnaire. Psychological Reports, 67, pp859-866.
- **Altman D. G.** (1991). Practical Statistics for Medical Research. 1st ed., Chapman & Hall, London.
- **Anderson J. R.** (1988). The expert module. Foundations of Intelligent Tutoring Systems (Eds. M. C. Polson & J. J. Richardson), Lawrence Erlbaum, Hillsdale, NJ, pp21-53.
- **Anderson J. R., Reder L. M. & Simon H. A.** (1996). Applications and Misconceptions of Cognitive Psychology to Mathematics Education. Internal paper, Carnegie Mellon University, Pittsburgh, USA.
- **Anoskey A. M. & Catrambone R.** (1992). Text and graphics in instructional design. Lecture Notes in Computer Science, 602, pp74-86.
- **Askar P., Yavuz H. & Koksal M.** (1992). Students' perceptions of computer assisted instructional environment and their attitudes towards computer assisted learning. Educational Research, 34(2), Summer, pp133-139.
- **Atherton P. et. al.** (1972). Free text retrieval evaluation. Syracuse University, School of Library Science, NY, Report No. RADC 72 159.
- **Azevedo R.** (1995). Assessing the effects of feedback in computer-assisted learning. British Journal of Educational Technology, 26(1), pp57-58.

- **Baker E. L.** (1991) in Intelligent Tutoring Systems - Evolution in Design (Eds. Burns H., Parlett J. W. & Redfield C. L.), Lawrence Erlbaum Associates, NJ.
- **Barr A. & Feigenbaum E. A.** (1982). The Handbook of Artificial Intelligence, 2, Kaufmann, Los Altos.
- **Beck A., Janssen C., Weisbecker A. & Ziegler J.** (1994). Integrating object-oriented analysis and graphical user interface design. Software Engineering and Human-Computer Interaction, Proceedings of ICSE'94 Workshop on SE-HCI: Joint Research Issues Sorrento, Italy, May, pp127-140.
- **Beilby M.** (1987). Experience of computer assisted education in a university department. Trends in Computer Assisted Education (Eds. R. Lewis & E. D. Tagg), Blackwell, London, pp174-179.
- **Bell M.** (1992). The Coventry Computer Based Learning Project. Programmed Learning & Educational Technology, 22(3), pp218-223.
- **Bentley T.** (1991). Intelligent tutoring. Management Accounting, April, pp14.
- **Bewley W., Roberts T., Schroit D. & Verplank W.** (1983). Human factors testing in the design of Xerox's 8010 "Star" office workstation. Proceedings of CHI' 83, Human Factors in Computer Systems, Boston, December 12-15, 1983 (Ed. A. Janda), ACM, New York.
- **Bhaskar K. N.** (1982). Use of Computers in accountancy Courses. Accounting and Business Research, Winter 1982, pp3-10.
- **Bhaskar K. N.** (1983). Computers and the choice for accountancy syllabuses. Accounting and Business Research, Spring 1983, pp83-93.
- **Bidin A. R. H. & Drabble G.** (1990). An organizational approach towards the development of educational computing in a university environment. Computers and Education, 14, pp137-143.
- **Blue Sky Software Corporation** (1991). WindowsMARKER. 2375 East Tropicana Ave., Suite 320, Las Vegas, NV 89119.
- **Bonnet A.** (1985). Artificial Intelligence: Promise and Performance. Prentice-Hall, London.

- **Brown J. S., Collins A. & Duguid P.** (1989a). Situated cognition and the culture of learning. Educational Researcher, 18(1), pp32-42.
- **Brown J. S., Collins A. & Duguid P.** (1989b). Debating and situation: A rejoinder to Palincsar and Wineburg. Educational Researcher, 18(4), pp10-12.
- **Brown J. S., Burton R. R. & deKleer J.** (1982). Pedagogical, natural language and knowledge engineering techniques in SOPHIE I, II and III. Intelligent Tutoring Systems (Eds. D. Sleeman & J. S. Brown), Academic Press, New York, pp227-282.
- **Brown J. S. & Burton R.** (1978). Diagnostic models for procedural bugs in basic mathematical skills. Cognitive Science, 2, pp155-192.
- **Brusilovskiy P. L.** (1994). The construction and application of student models in intelligent tutoring systems. Journal of Computer and Systems Sciences International, 32(10), pp70-89.
- **Brusilovsky P., Schwarz E. & Weber G.** (1996). A tool for developing hypermedia-based ITS on WWW. ITS'96 Workshop on Architectures and Methods for Designing Cost-Effective and Reusable ITSs, Montreal, June 10th 1996.
- **Bryman A.** (1988). Quantity and Quality in Social Research. Unwin Hyman, London.
- **Bryman A.** (1989). Research Methods and Organization Studies. Unwin Hyman, London.
- **Burns H. L. & Capps C. G.** (1988). Foundations of intelligent tutoring systems: an introduction. Foundations of intelligent tutoring systems (Eds. Polson M. C. & Richardson J. J.). Lawrence Erlbaum, London, pp1-19.
- **Burton E. J., McKeown J. C. & Sholsberg J. L.** (1978). The generation and administration of examinations on interactive computer systems. The Accounting Review, LIII (1), January 1978, pp170-178.
- **Byrum D. C.** (1992). Formative evaluation of computer courseware: An experimental comparison of two methods. Journal of educational computing research, 8(1), pp69-80.

- **Carbonell J. R.** (1970). AI in CAI: an artificial intelligence approach to computer-assisted instruction. IEEE Transactions on Man-Machine Systems, 11, pp190-202.
- **Card S. K., Moran T. P. & Newell A.** (1983). The Psychology of Human Computer Interaction. Lawrence Erlbaum, NJ.
- **Chen Z.** (1995). Interacting with software system components. Decision Support Systems, 14, pp349-357.
- **Clancey W. J.** (1983). GUIDON. Journal of Computer-Based Instruction, 10(1), pp8-14.
- **Clancey W. J.** (1987). Knowledge based tutoring: the Guidon Program. MIT Press, Cambridge.
- **Clancey W. J. & Soloway E.** (1990). Artificial intelligence and learning environments: Preface. Artificial Intelligence, 42, pp1-6.
- **Collins A. M.** (1976). Processes in acquiring knowledge. Schooling and the acquisition of knowledge (Eds. R. C. Anderson, R. Spiro & W. E. Montague), Lawrence Erlbaum Associates, Hillsdale, NJ, pp339-363.
- **Collins R.** (1983). Computers and accountancy courses: A comment. Accounting and Business Research, Autumn 1983, 13(52), pp313-315.
- **Collins M. A. J.** (1989). The hidden costs of computer courseware authoring tools. Educational Technology, 29(6) pp46-47.
- **Cordell B. J.** (1991). A study of learning styles and computer-assisted instruction. Computers & Education, 16, pp175-183.
- **Coutaz J., Salber D. & Balbo S.** (1993). Towards automatic evaluation of multimodel user interfaces. Knowledge-Based Systems, 6(4), pp267-274.
- **Crowder N. A.** (1959). Automatic tutoring by means of intrinsic programming. Automatic Teaching: The State of Art, Wiley, NY, pp109-116.
- **Cusack M. W.** (1993). Efforts to simplify human-computer communication. IEEE Transactions of Professional Communications, 36, pp14-19.

- **Dalal N. P. & Kasper G. M.** (1994). The design of joint cognitive systems: the effect of cognitive coupling on performance. International Journal of Human-Computer Studies, 40, pp677-702.
- **Darby J.** (1992a). Computers in teaching and learning in UK higher education. Computers & Education, 19, pp1.
- **Darby J.** (1992b). The future of computers in teaching and learning. Computers & Education, 19(1/2), pp193-197.
- **Daroca F. P.** (1986). Introducing Microcomputers into the Classroom: Student Learning and Perceptions. Kent Bentley Review: Accounting and Computers, Vol. 2, Fall 1986, pp67-78.
- **Delahaye B. & Thompson B.** (1991). Learning styles - What do they measure? Asia Pacific Human Resource Management, Winter, pp60-68.
- **Dillenbourg P. & Self J.** (1992). People Power: A human computer collaborative learning system. Lecture Notes in Computer Science, v608 (Eds. C. Frasson, G. Gathier & G. I. McCalla), "Intelligent Tutoring Systems", Second International Conference, ITS'92, Montreal, Canada, June, Springer-Verlag, Berlin, pp651-660.
- **Dix A., Finlay J., Abnod G. & Beale R.** (1993). Help and Documentation. Human Computer Interaction. Prentice-Hall, London, pp401-420.
- **Dixon M.** (1988). Can you really learn from computers ? Training Officer, September 1988, pp16-18.
- **Dorr P., Eining M. & Groff J.** (1988). Developing an accounting expert system decision aid for classroom use. Issues in Accounting Education, 3(1), Spring, pp27-41.
- **Duncan N. C.** (1991). CAI-enhanced exam performance in a research design course. Behavior Research Methods, Instruments, & Computers, 23, pp324-327.
- **Duncan N. C.** (1993). Evaluation of instructional software: Design considerations and recommendations. Behavior Research Methods, Instruments, & Computers, 25(2), pp223-227.

- **Dungworth S. & Russell D.** (1994). The evaluation of interactive C.A.L. in management accounting: Further evaluation of a management accounting package in practice. <u>5th Annual CTI-AFM Conference</u>, April 94, Coventry.
- **Dunn R., Dunn K. & Price G.** (1977). Diagnosing learning styles: a prescription for avoiding malpractice suits. <u>Phi Delta Kappan</u>, 58, pp418-420.
- **Entwistle N. J.** (1981). <u>Styles of Learning and Teaching</u>. Wiley, New York.
- **Epstein K. & Hillegeist E.** (1990). Intelligent instructional systems: Teachers and computer-based intelligent systems. <u>Educational Technology</u>, Nov., pp13-19.
- **Er M. C. & Ng A. C.** (1989). The Use of Computers in Accountancy Courses: A New Perspective. <u>Accountancy and Business Research</u>, 19(76), pp319-326.
- **Faiola T.** (1989). Improving courseware development efficiency: the efforts of authoring systems on team roles and communication. <u>Educational Technology</u>, 29(8) pp16-19.
- **Forrester M.** (1995). Interpreting 'Hyper-Entrails' in a computer-based learning environment. <u>TLTP Newsletter</u>, Summer, pp8-9.
- **Freedman R. D. & Stumph S. A.** (1978). What can one learn from the Learning Style Inventory? <u>Academy of Management Journal</u>, 21(2), pp275-282.
- **Furnham, A.** (1992). Personality and learning style: A study of three instruments. <u>Personality and Individual Differences</u>, 13(4), pp429-438.
- **Gallagher I. D. & Letza S. R.** (1991). An evaluation of computer aided accounting learning- a student's perspective. <u>Paper presented at the Reviewing Accounting Courseware conference</u>, Manchester university, 26 March 91.
- **Gardner J.** (1990). Computers in higher education teaching and learning. <u>Proceedings from the symposium held at The Queen's University of Belfast</u>, June 1990, pp1.1-1.3.
- **Gee K.** (1988). A one term financial modelling course for second year accounting undergraduate using the Salford 'NOESIS' database. <u>Paper</u>

presented at the Sharing Accounting Courseware Conference, March 1988, Manchester.

- **Genesereth M. R. & Ketchpel S. P.** (1994). Software agents. Communications of the ACM, 37(7), July, pp48-53 & 147.
- **Gershuny J. J. and Slater J. B.** (1989). Computers in Teaching Initiative Report, University of Bath, 1989.
- **Ginter E., Brown S., Scalise J. & Ripley W.** (1989). Perceptual learning styles: their link to performance, sex, age and academic standing. Perceptual and Motor Skills, 68, pp1091-1094.
- **Gladwin R. P., Margerison D. & Walker S. M.** (1992). Computer assisted learning in chemistry. Computers and Education, 19, pp17.
- **Glimore D. & Self J. A.** (1988). The application of machine learning to intelligent tutoring systems. Artificial Intelligence and Human Learning: Intelligent Computer- Aided Instruction (Ed. J. A. Self), Chapman & Hall, London, pp179-196.
- **Goldstein I. P.** (1982). The genetic graph: a representation for the evolution of procedural knowledge. Intelligent Tutoring Systems (Eds. D. H. Sleeman & J. S. Brown), Academic Press, New York, pp51-77.
- **Goldstein I. P. & Carr B.** (1977). The computer as coach: An athletic paradigm for intellectual education. Proceedings of ACM77, pp227-233.
- **Goldstein M. B. & Bokoros M. A.** (1992). Tilting at windmills: Comparing the learning style inventory and the learning style questionnaire. Educational and Psychological Measurement, 52(3), pp701-708.
- **Goldwater P. L. & Fogerty T. J.** (1993). The development of an expert system for accounting education. International Journal of Applied Expert Systems, 1(1), pp41-57.
- **Grabe M., Petros T. & Sawler B.** (1989). An evaluation of computer assisted study in controlled and free access settings. Journal of Computer-Based Instruction, 18, pp77-82.

- **Green D., Snell J. & Parimanath A.** (1990). Learning styles in assessment of students. Perceptual and Motor Skills, 70, 363-369.
- **Hammond N., Gardner N., Heath S., Kibby M., Mayes T., McAleese R., Mullings C. & Trapp A.** (1992). Blocks to the effective use of information technology in higher education. Computers and Education, 18(1-3), pp155-162.
- **Hammond N., Hinton G., Barnard P., MacLean A., Long J. & Whitefield A.** (1984). Evaluating the interface of a document processor: a comparison of expert judgement and user observation. Paper presented at the Interact' 84, First IFIP conference on Human-Computer Interaction, Amsterdam.
- **Hannafin M. J. & Carney B. W.** (1991). Effects of elaboration strategies on learning and depth of processing during computer-based instruction. Journal of Computer-Based Instruction, 18, pp77-82.
- **Harrison N.** (1995). Practical Instructional Design for Open Learning Materials. Second Ed., The McGraw-Hill Training Series (Series Ed. Roger Bennett), London.
- **Hartley J. R. & Sleeman D. H.** (1973). Towards more intelligent teaching systems. International Journal of Man- Machine Studies. 5, pp215-236.
- **Hawkes W. L., Sharon J. D., Kandel A. & Taps Project Staff** (1986). Fuzzy expert systems for an intelligent computer based tutor. Technical Report No. 86-5. Learning Systems Institute, Centre for Educational Technology, Florida State University.
- **Hayes J. & Allinson C.** (1988). Cultural differences in the learning styles of managers. Management International Review, 28, pp75-80.
- **Hazari S. I. & Reaves R. R.** (1994). Student preferences toward microcomputer user interfaces. Computers and Education, 22(3), pp225-229.
- **Hefley W.** (1995). Helping users help themselves. IEEE Software, March, pp93-95.
- **Heller R. S.** (1991). Evaluating software: A review of the options. Computers & Education, 17(4), pp285-291.

- **Hendley R. J. & Jurascheck N.** (1992). Cascade: Introducing AI into CBT. Computers and Education, 18(1-3), pp71-76.
- **Hendry D. G. & Green T. R. G.** (1994). Creating, comprehending and explaining spreadsheets: a cognitive interpretation of what discretionary users think of the spreadsheet model. International Journal of Human-Computer Studies, 40, pp1033-1065.
- **Hennessy S., Evertsz R., Ellis D., Black P., O'Shea T. & Floyd A.** (1988). Design specifications for '*Shopping on Mars*', a Computer-Based Educational Activity. Technical Report No. 29, Centre for Information Technology in Education, Open Press, Milton Keynes.
- **Henwood K. L. & Pidgeon N. F.** (1993). Qualitative research and psychological theorizing. Social Research - Philisophy, Politics and Practice (Ed. M. Hammersley), SAGE, London, pp14-32.
- **Hewett T. T.** (1986). The role of iterative evaluation in designing systems for usability. Proceedings of HCI '86, "People and Computers : Designing the Interface", University of York, 23-26 September 1986 (Eds. Harrison M. D. & Monk A. F.), pp196-214.
- **Hines R. D.** (1989). Financial accounting knowledge, conceptual framework projects and the social construction of the accounting profession. Accounting, Auditing and Accountability Journal, 2(2), pp72-92.
- **Hoey A. & Gregg B.** (1994). The development of a computer-based teaching aid in public procurement law. Account, 6(2), pp29-31.
- **Hollan J. D., Hutchins E. L. & Weitzman L.** (1984). STEAMER: an interactive inspectable simulation-based training system. AI Magazine, 5(2), pp15-27.
- **Holmes G.** (1991). Evaluation of IVY Software. Account, 3(1), pp27-29.
- **Honey P. & Mumford A.** (1986). The Manual of Learning Styles. 2nd ed., Maidenhead, Berks.
- **Howard S. & Murray D.** (1987). A survey and Classification of Evaluation Techniques for HCI. NPL Report DITC 94/87, July 87.

- **Hsu J. J., Chapelle C. A. & Thompson A. D.** (1993). Exploratory learning environments: What are they and do students explore? Journal of Educational Computing Research, 9(1), pp1-15.
- **Humpert B. & Holley P.** (1988). Expert systems in financial planning. Expert Systems, 5(2), pp78-103.
- **Hutchins E. L.** (1987). Metaphors for interface design. ICS report no. 8703, Institute of Cognitive Science, University of California, San Diago, California.
- **Iannella R.** (1992). Designing 'Safe' User Interfaces. The Australian Computer Journal, 24(3), August, pp92-97.
- **Ince D.** (1988). Financial and business applications of artificial intelligence. Information Technology Briefings, pp9-14.
- **Iqbal A.** (1993). Survey of a Tutoring System. University College, London.
- **Iqbal A.** (1994). Assessing the value and effectiveness of an ITS - development of evaluation metrices. Working Paper, University College, London.
- **James K.** (1980). The development of senior managers for the future. Advances in Management Education (Eds. J. Beck & C. Cox), Wiley, London, pp55-71.
- **Jamous H. & Peloille B.** (1970). Changes in the French university-hospital system. Professions and Professionalization (Ed. J. A. Jackson), Cambridge University Press, Cambridge, pp113.
- **Jensen R. E. & Sandlin P. K.** (1992). Why do it? Advantages and Dangers of new waves of computer- aided teaching/instruction. Journal of Accounting Education, 10, pp39-60.
- **Jervis P.** (1983). Analysing decision behaviour: learning models and learning styles as diagnostic aids. Personal Review, 12(2), pp26-37.
- **Johnson, S.** (1752). The Rambler, pp87.
- **Johnson L. & Soloway E.** (1984). PROUST: Knowledge based program debugging. Proceedings of the Seventh International Software Engineering Conference, pp369-380.

- **Johnson-Laird P.** (1993). The Computer and the Mind. Second Ed., Fontana press, London.
- **Jong T. D., Andel J. V., Leiblum M. & Mirande M.** (1992). Computer assisted learning in higher education in the Netherlands: A review of findings. Computers and Education, 19(4), pp381-386.
- **Juch A. H.** (1980). A general process model for personal development. PhD Thesis, University of Bath.
- **Kachelmeier S. J., Jones J. D. & Keller J. A.** (1992). Evaluating the effectiveness of a computer-intensive learning aid for teaching pension accounting. Issues in Accounting Education, 7(2), Fall, pp164-178.
- **Kaplan R. & Rock D.** (1995). New directions for intelligent tutoring. AI Expert, February, pp31-40.
- **Kass R.** (1989). Student Modelling in ITSs- Implications for User Modelling. User Models in Dialogue Systems (Eds. A. Kobsa & W. Wahlster), Springer-Verlag, Berlin, pp386-410.
- **Katz S. & Lesgold A.** (1994). Implementing post-problem reflection within coached practice environments. Paper presented at the East West International Conference on Computer Technologies in Education, September, Crimea, Ukraine.
- **Kay J.** (1991). An explicit approach to acquiring models of student knowledge. Advanced Research on Computers in Education (Eds. R. Lewis & S. Otsuki), Elsevier Science Publishers B. V., North-Holland, pp263-268.
- **Kaye G. R. & Harrop R. L.** (1994). Is Accounting an Appropriate Subject for a Hypermedia Teaching Tool?, Paper presented at the British Accounting Association Annual Conference, 23-25 March, King Alfred's College, Winchester, UK.
- **Kieras D. & Polson P. G.** (1985). An approach to the formal analysis of user complexity. International Journal of Man-Machine Studies, 22, pp365-394.

- **King M. & McAuley L.** (1991). A standard costing knowledge base: building and using an expert system in management accounting education. Issues in Accounting Education, 6(1), Spring, pp97-111.
- **King R. & Whittaker J.** (1990). Integrating Computers into Accounting Courses. Working Paper, Loughborough University Business School.
- **Kinshuk** (1995). The influence of interface design upon the effectiveness of computer aided learning programs in entry level accounting subjects. MPhil-Phd transfer document, De Montfort University, Leicester, UK, January 1995.
- **Kinshuk & Patel A.** (1996). Intelligent Tutoring Tools - Redesigning ITSs for adequate knowledge transfer emphasis. International Conference on Intelligent and Cognitive Systems '96, Tehran, Iran, September 23-26, 1996.
- **Kolb D. A.** (1976). Learning Style Inventory: Technical Manual. McBer, Boston, Mass.
- **Kolb D. A.** (1981). Experiential learning theory and the learning style inventory: a reply to Freedman and Stumpf. Academy of Management Review, 6(2), pp289-296.
- **Kolb D. A.** (1984). Experiential Learning. Prentice-Hall, Englewood Cliffs, NJ.
- **Kolb D. A.** (1985). Learning Style Inventory: Technical Manual. McBer and Company, Boston.
- **Kulik C. L. C. & Kulik J. A.** (1991). Effectiveness of computer-based instruction: An updated analysis. Computers in Human Behavior, 7, pp75-94.
- **Lambart C.** (1990). The electronic tutor. Harvard Magazine, 93(2), pp42-51.
- **Landseadel P.** (1995) Methodologies in user interface design. IEEE Aerospace and Electronic Systems Magazine, 10(8), pp15-20.
- **Laurillard D.** (1993). Rethinking university teaching - a framework for the effective use of educational technology. Routledge, London.
- **Law R. C. H. & Maguire R. B.** (1993). A critical survey of intelligent tutoring systems. Congressus Numerantium, 92, pp55-64.

- **Legree P. J., Gillis P. D. & Orey M. A.** (1993). The quantitative evaluation of intelligent tutoring system applications: Product and process criteria. Journal of Artificial Intelligence and Education, 4(2/3), pp209-226.
- **Lewis R.** (1990). Computers in higher education teaching and learning - some aspects of research and development. Proceedings from the symposium "Computers in higher education teaching and learning" held at The Queen's University of Belfast", June 1990, pp3.1-3.8.
- **Lewis R. & Margerison C.** (1979). Working and Learning - Identifying your preferred ways of doing things. Personnel Review, 8, pp25-29.
- **Liefeld J. P. & Herrmann T. F.** (1990). Learning consequences for university students using computerized mastery tessting. Educational Technology, Research & Development, 38, pp19-25.
- **Littman D. & Soloway E.** (1988). Evaluating ITSs: The Cognitive Science Perspective. Foundations of Intelligent Tutoring Systems (Eds. Polson M. C. & Richardson J. J.), Lawrence Erlbaum Associates, Hillsdale, NJ, pp209-242.
- **Magnuson-Martinson S.** (1995). Classroom computerisation: Ambivalent attitudes and ambiguous outcomes. Teaching Sociology, 23, January, pp1-7.
- **Mandl H. & Lesgold A.** (1988). Learning Issues for Intelligent Tutoring Systems. Springer-Verlag, London.
- **Marchisio L., Ronco E. & Saracco R.** (1993). Modeling the user interface. IEEE Communications Magazine, 31(5), pp68-74.
- **Mark M. A. & Greer J. E.** (1993). Evaluation methodologies for intelligent tutoring systems. Journal of Artificial Intelligence in Education, 4(2/3), pp129-153.
- **Martin J.** (1973). Design of Man Computer Dialogues. Prentice-Hall, Englewood Cliffs, NJ.
- **Marton F. & Saljo R.** (1976a). On qualitative differences in learning, 1: outcome and process. British Journal of Educational Psychology, 46, pp4-11.

- **Marton F. & Saljo R.** (1976b). On qualitative differences in learning, 2: outcome as a function of the learner's conception of the task. British Journal of Educational Psychology, 46, pp115-127.
- **McDonough D., Strivens J. & Rada R.** (1994a). University courseware development: differences between computer-based teaching by users and non-users. Computers and Education, 23(3), pp211-220.
- **McDonough D., Strivens J. & Rada R.** (1994b). Current development and use of computer-based teaching at the University of Liverpool. Computers and Education, 22(4), pp335-343.
- **McKeown J. C.** (1976). Computer-Assisted Instruction for Elementary Accountancy. The Accounting Review, January 1976, pp123-130.
- **McNeil B. J. & Nelson K. R.** (1991). Meta-analysis of interactive video-instruction: A 10 year review of achievement effects. Journal of Computer-Based Instruction, 18, pp1-6.
- **Megarry J.** (1983). Thinking, learning and educating: The role of the computer. Computers and Education, (Eds. Megarry et. al.), Nichols Pub. Co., NY, pp15-28.
- **Messick et. al.** (1976). Individuality in Learning. Jossey Bass, CA.
- **Miller C. M. & Parlett M. R.** (1974). Up to the mark: a study of the examination game. SRHE, London.
- **Miller P. L. & Sittig D. F.** (1990). The evaluation of clinical decision support systems: What is necessary verses what is interesting. Medical Informatics, 15(3), pp185-190.
- **Miller D. S., Smith J. G. & Muller M. J.** (1992). TelePICTIVE: Computer-supported cpllaborative GUI design for designers with diverse expertise. Proceedings of the ACM Symposium on User Interface Software and Technology, Monterey, California, November 15-18, 1992, pp151-160.
- **Mittman B. & Dominick W. D.** (1973). Developing monitoring techniques for an on-line information retrieval system. Information Storage & Retrieval, 9(6), pp297-307.

- **Miyashita K., Matsuoka S., Takahashi S., Yonezawa A. & Kamada T.** (1992). Declarative programming of graphical interfaces by visual examples. Proceedings of the ACM Symposium on User Interface Software and Technology, Monterey, California, November 15-18, 1992, pp107-116.
- **Moonen J. and Gastkemper F.** (1983). Computer Assisted Instruction. Het Spectrum, Utrecht.
- **Mudd S. and McGrath K.** (1988). Correlates of the adoption of curriculum-integrated computing in higher education. Computers and Education, 12, pp457-463.
- **Muggridge L. & Lymer A. M.** (1993). Assessing professional expertise using expert systems: expert systems in the professional accounting domain. International Journal of Applied Expert Systems, 1(3), pp229-238.
- **Mumford A.** (1982). Learning style and learning skills. Journal of Management Development, 1(2), pp55-65.
- **Mumford A. C.** (1987). Learning Styles and Learning. Personnel Review, 16(3), pp20-23.
- **Murray T.** (1993). Formative qualitative evaluation for "Exploratory" ITS research. Journal of Artificial Intelligence and Education, 4(2/3), pp179-207.
- **Myers B. A.** (1992). State of the Art in User Interface Software Tools. Advances in Human-Computer Interaction (Eds. H. Rex Hartson & Deborah Hix), 4, Ablex Publishing, Norwood, NJ, pp1-31.
- **Myers I. B.** (1979). Manual: The Myers-Briggs Type Indicator. Consulting Psychologists Press, Palo Alto, CA.
- **Neal A. S. & Simons R. M.** (1983). Playback: a method for evaluating the usability of software and its documentation. Paper presented at the CHI' 83, Human Factors in Computer Systems, Boston, December 12-15, 1983.
- **Needles, Anderson & Caldwell** (1990). Principles of Accounting. 4th Ed., Houghton Miffin, Boston.
- **Newing R.** (1995). Software for better budgeting. Management Accounting, October, pp14.

- **Nicholson A. H. S.** (1993). The Source Book. Publishers: CTI Centre for Accounting Finance and Management, pp44.
- **Nicholson A. H. S.** (1994a). Software review: Understand ACCOUNTS. Account, 6(2), Summer 1994, pp38-40.
- **Nicholson A. H. S.** (1994b). Software review: EQL Interactive Taxation Tutor. Account, 6(3), Autumn 1994, pp45.
- **Nwana H. S.** (1990). Intelligent Tutoring Systems: An Overview. Artificial Intelligence Review, 4, pp251-277.
- **O'Kane B.** (1995). Sign here, please. Certified Accountant, September, pp44-45.
- **O'Malley C.** (1990). Interfaces issues for guided discovery learning environments. Guided Discovery Tutoring (Ed. Mark Elsom-Cook), Paul Chapman, London, pp24-41.
- **Ohlson S.** (1991a). System hacking meets learning theory: Reflections on the goals and standards of research in artificial intelligence and education. Journal of Artificial Intelligence, 2(3), pp5-18.
- **Ohlson S.** (1991b). Artificial intelligence : a method for relating learning theory to instructional design. Learning Research and Development Center, University of Pittsburgh, Pennsylvania.
- **Ohlson S. & Rees E.** (1991). The function of conceptual understanding in the learning of arithmatic procedures. Cognition and Instruction, 8, pp103-179.
- **Ohlsson S.** (1986). Some principles of intelligent tutoring. Instructional Science, 14, pp293-326.
- **Okamoto T., Morihiro K., Matsuda N. & Takuma S.** (1994). Application of analogical reasoning and extraction of tutoring rules for concept-formation learning. Electronics and Communications in Japan, Pt. 3, 77(3), pp75-86.
- **Okamoto T.** (1994). The current situations and future directions of intelligent CAI research/development. IEICE Transactions of Information & Systems, E77-D(1), January, pp9-18.

- **Osborne C.** (1994). Independent learning at Middlesex University. Draft text for International Symposium on Independent Study and Flexible Learning, Cambridge, 6-9 Sept. 1994.
- **Pangalos G. J.** (1993). Designing the user interface. Computers in Industry, 22(2), pp193-200.
- **Pask G. & Scott B. C. E.** (1972). Learning strategies and individual competence. International Journal of Man-Machine Studies, 4, pp217-253.
- **Patel A.** (1995). Intelligent Tutoring Tools:An evaluation of TLTP Byzantium's approach to software design for a computer integrated learning environment in introductory accounting, MSc Project Report, De Montfort University, UK, September.
- **Patel A. & Kinshuk** (1996a). Knowledge characteristics: Reconsidering the design of Intelligent Tutoring Systems. International Conference: Knowledge Transfer - '96, London, July 22-26, 1996.
- **Patel A. & Kinshuk** (1996b). Applied Artificial Intelligence for Teaching Numeric Topics in Engineering Disciplines. Third International Conference on Computer Aided Learning and Instruction in Science and Engineering, Donostia - San Sebastian, Spain, July 29-31, 1996.
- **Patel A. & Kinshuk** (1996c). Artificial Intelligence in learning and assessment - A supportive software for the teaching of Accounting. The 1996 East-West International Conference on Human- Computer Interaction, Moscow, Russia, August 12-16, 1996.
- **Patel A. & Kinshuk** (1996d). Intelligent Tutoring Tools - A problem solving framework for learning and assessment. 1996 Frontiers in Education Conference - Technology-Based Re-Engineering Engineering Education, Salt Lake City, Utah, November 6-9, 1996.
- **Penniman W. D. & Dominick W. D.** (1980). Monitoring and Evaluation of On-line information system usage. Information Processing and Management, 16(1), pp17-35.
- **Perks R. W.** (1993). Accounting and Society. Chapman & Hall, London.

- **Petty L. C. & Rosen E. F.** (1990). Increase in mastery levels using a computer-based tutorial/simulation in experimental psychology. Behavior Research Methods, Instruments & Computers, 22, pp216-218.
- **Pinker S.** (1994). Then Language Instinct : How the mind creates language. William Morrow, New York.
- **Plessis J. P. du, Van Biljon J. A., Tolmie C. J. & Wollinger T.** (1995). A model for intelligent computer-aided education systems. Computers & Education, 24(2), pp89-106.
- **Reiser B. J., Beekelaar R., Tyle A. & Merrill D.** (1991). GIL: Scaffolding learning to program with reasoning-congruent representations. CSL Report 48, Princeton University Cognitive Science Laboratory, Princeton, New Jersey.
- **Reiser B. J., Ranney M., Lovett M. C. & Kimberg D. Y.** (1989). Facilitating student's reasoning with casual explanations and visual representations. Proceedings of the Fourth International Conference on Artificial Intelligence and Education (Eds. D. Bierman, J. Breuker & J. Sanberg), Springfield, Virginia.
- **Reiterer H. & Oppermann R.** (1993). Evaluation of user interfaces: EVADIS II - a comprehensive evaluation approach. Behaviour & Information Technology, 12(3), pp137-148.
- **Repman, J.** (1993). Collaborative, computer-based learning: cognitive and affective outcomes. Journal of Educational Computing Research, 9(2), pp149-163.
- **Rich E.** (1979). User modelling via stereotypes. Cognitive Science, 3, pp329-354.
- **Rich E.** (1983). Users are individuals: individualising user models. International Journal of Man Machine Studies, 18, pp199-214.
- **Richer M. H. & Clancey W. J.** (1985). GUIDON-WATCH: a graphic interface for viewing a knowledge-based system. IEEE Computer Graphics and Applications, 5(11), pp51-64.

- **Ridgway J., Benzie D., Burkhardt H., Coupland J., Field G., Fraser R. & Phillips R.** (1984). Investigating CAL? Computers and Education, 8(1), pp85-92.
- **Ridgway J.** (1986). Development and Evaluation of CAL Materials. Proceedings of 6th Computers in Higher Education Conference, April 86, University of Lancaster. Trends in Computer Assisted Education (Eds. Lewis R. & Tagg E. D.), Blackwell, London, pp203-212.
- **Rogers Y.** (1988). Pictorial representations of abstract concepts in relation to human computer interaction. Unpublished PhD thesis. University of Wales.
- **Root R. W. & Draper S.** (1983). Questionnaires as a Software Evaluation Tool. Paper presented at the CHI' 83, Human Factors in Computer Systems, Boston, December 12-15, 1983.
- **Rossi-Mori A., Pisanelli D. M., Ricci F. L.** (1990). Evaluation stages and decision steps for knowledge-based systems in medicine. Medical Informatics, 15(3), pp191-204.
- **Ross P., Jones J. & Millington P.** (1987). User Modelling in Intelligent Teaching and Tutoring. Trends in Computer Assisted Instruction (Eds. R. Lewis & E. D. Tagg). Blackwell, London, pp32-44.
- **Rouse P.** (1993). Constructing Monte Carlo simulations in Lotus 1-2-3. Journal of Accounting Education, 11, pp113-132.
- **Routen T.** (1992). Reusing formalisations of legislation in a tutoring system. Artificial Intelligence Review, 6, pp145-159.
- **Rowntree D.** (1987). CBT = Computers Bamboozle Trainers ? Training Officer, November 1987, pp24-32.
- **Ruf B. M., Brown R. M. & Crawford H. J.** (1994). Computer-assisted versus traditional instruction: Differential moderating influences of math, verbal and visual-spatial abilities on learning. British Accounting Review, 26, pp197-210.
- **Sangster A. & Wilson R. A.** (1991). Knowledge-based learning within the accounting curriculum. British Accounting Review, 23, pp243-261.

- **Sangster A.** (1992). Computer-based learning in UK accounting education: to support or to supplant? British Journal of Education Technology, 23(2), pp136-146.
- **Schmeck R., Ribich F. & Ramaniah N.** (1977). Development of Self-report Inventory for assessing individual differences in learning processes. Applied Psychological Measurement, 1, pp413-431.
- **Shute V. J. & Regian J. W.** (1993). Principles for evaluating intelligent tutoring systems. Journal of Artificial Intelligence and Education, 4(2/3), pp245-271.
- **Seidel R. J. & Park O.** (1994). An historical perspective and a model for evaluation of intelligent tutoring systems. Journal of Educational Computing Research, 10(2), pp103-128.
- **Self J. A.** (1974). Student models in computer-aided instruction. International Journal of Man-machine Studies, 6, pp261-276.
- **Self J. A.** (1987a). The application of machine learning to student modelling. Artificial Intelligence and Education 1: Learning Environments & Tutoring Systems (Eds. R. Lawler & M. Yazdani). Ablex, Norwood, pp267-280.
- **Self J. A.** (1987b). Realism in Student Modelling. Alvey-IKBS Research Workshop Tutoring Systems. University of Exeter.
- **Self J. A.** (1988). Student models: what use are they? Artificial Intelligence Tools in Education (Eds. P. Ercoli & R. Lewis). North Holland, Amsterdam, pp73-86.
- **Self J. A.** (1990). Bypassing the intractable problem of student modelling. Intelligent Tutoring Systems: At the Crossroads of AI and Education (Eds. C. Frasson & G. Gauthier). Ablex, Norwood, pp107-123.
- **Shaoul J. E.** (1989). Sharing Accounting Courseware - Some of the issues and a only few solutions. Paper presented at the conference "Sharing Accounting Courseware", University of Manchester, 22 March 1989.
- **Shaoul J. E.** (1990). Teaching accounting with computers - before and after CTI. CTISS File, 9, February 1990.

- **Shaoul J. E.** (1991). Using a database for teaching management accounting. Paper presented at the Reviewing Acounting Courseware Conference, University of Manchester, 26 March 1991.
- **Shute V. & Glaser R.** (1990). A large scale evaluation of an intelligent discovery world: Smithdown. Interactive Learning Environments, 1, pp51-77.
- **Simons P. R. & De Jong F. P. C. M.** (1992). Self-regulation and Computer Aided Instruction. Applied Psychology: An International Review, 41(4), pp333-346.
- **Simon H. A.** (1981). The Sciences of the Artificial. 2nd Ed., MIT Press, Cambridge, MA.
- **Simpson S. M.** (1986) Review of Computer Aided Instruction AB-Initio Teaching in Accounting. M. Phil Dissertation, Hatfield Polytechnic.
- **Skinner B. F.** (1958). Teaching Machines. Science, 128, pp969-977.
- **Sleeman D. H.** (1985). UMFE: a user modelling front-end subsystem. International Journal of Man- Machine Studies. 23, pp71-88.
- **Sleeman D. H. & Brown J. S.** (1982a). Intelligent Tutoring Systems, Academic Press, New York.
- **Sleeman D. H. & Brown J. S.** (1982b). Introduction: Intelligent Tutoring Systems. Intelligent Tutoring Systems (Eds. D. H. Sleeman & J. S. Brown), Academic Press, New York, pp1-11.
- **Smith G. N.** (1990). Electronic spreadsheet applications for accounting principles. 2nd Ed., South-Western, Cincinnati.
- **Smith R. B.** (1986). The Alternate Reality Kit: an animated environment for creating interactive simulations. Proceedings of the IEEE Workshop on Visual Languages, IEEE Press, Washington, DC, pp203-210.
- **Sokolnicki T.** (1991). Towards knowledge-based tutors: a survey and appraisal of Intelligent Tutoring Systems. The Knowledge Engineering Review, 6, pp54-95.
- **Spiegelhalter D. S.** (1983). Evaluation of clinical decision aids with an application to a system for dyspepsia. Statistics in Medicine, 2, pp207-216.

- **Steuck K.** (1992). The initial evaluation of an intelligent tutoring system for pre-algebra word problems. Paper presented at the annual meeting of the American Educational Research Association, April, San Francisco, CA.
- **Sugarman L.** (1985). Kolb's model of experiential learning: touchstone for trainers, students, counselors, and clients. Journal of Counseling and Development, 64(4), pp264-268.
- **Suppes P.** (1967). Some theoretical models for mathematics learning. Journal of Research and Development in Education, 1, pp5-22.
- **Svensson L.** (1977). On qualitative differences in learning, 3: study skills and learning. British Journal of Educational Psychology, 47, pp233-243.
- **Tepper B. J., Tetrault L. A., Braun C. K. & Romero J. E.** (1993). Discriminant and convergent validity of the problem solving style questionnaire. Educational and Psychological Measurement, 53(2), pp437-444.
- **Thomas A. L.** (1984). North American college and university uses of micro computers in teaching accounting - 1983 survey. Journal of Accounting Education, 2(2), Fall, pp31-41.
- **Thomas D. L. & Bostow D. E.** (1991). Evaluation of pre-therapy computer-interactive instruction. Journal of Computer-Based Instruction, 18, pp66-70.
- **Tobias S.** (1985). Computer Assisted Instruction. Adapting Instruction to Individual Differences (Eds. M. C. Wang & H. J. Waldberg). McCutchan, Berkeley, CA, pp139-159.
- **Tonge A., Beacham N. & Gallagher D.** (1994). The Implementation and Evaluation of Computer Based Learning in Accounting Education. Paper presented at the International Symposium on Independent Learning and Flexible Study, Cambridge, September 1994.
- **Tonge A., Johnston K., Dungworth S. & Russell D.** (1993). "The only truly effective C.A.L. is interactive" - An evaluation of a management accounting C.A.L. package in practice. Paper presented at the 4th Annual CTI-AFM Conference, March 93, Leicester.

- **Uhr L.** (1969). Teaching machine programs that generate problems as a function of interaction with students. Paper presented at the 24th National Conference of Education, Virginia, April 2-4, 1969.
- **VanLehn K., Jones R. M. & Chi M. T. H.** (1991). Modelling the self-explanation effects with Cascade3. Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society (Eds. K. Hammond & D. Gentner), Erlbaum, Hillsdale, New Jersey.
- **VanLehn K.** (1988). Student Modelling. Foundations of Intelligent Tutoring Systems (Eds. Polson M. C. & Richardson J. J.). Lawrence Erlbaum Associates, Hillsdale, NJ, pp55-78.
- **VanLehn K.** (1994). Learning events in the acquisition of three skills. Proceedings of the Eleventh Annual Conference of the Cognitive Science Society (Eds. G. Ohlson and E. Smith), Erlbaum, Hillsdale, New Jersey.
- **Varnhagen C. K. & Zumbo B. d.** (1990). CAI as an adjunct to teaching introductory statistics: Affect mediates learning. Journal of Educational Computing Research, 6 pp29-40.
- **Vasandani V & Govindaraj T.** (1995). Knowledge organisation in intelligent tutoring systems for diagnostic problem solving in complex dynamic domains. IEEE Transactions on System, Man, and Cybernetics, 25(7), July, pp1076-1096.
- **Vickers S. P. & Kerrison M. A.** (1991). A Teacher's Guide to Courseware in Management Accounting and Courseware in Finance. Paper presented at the Reviewing Accounting Courseware Conference, University of Manchester, March 1991.
- **Villiers C. de, Pistorius M. C., Alexander P. M. and Plooy N. F. du (1992).** Constraints on computer - assisted instruction in a distance education environment. Computing & Control Engineering Journal, January 1992, pp11-13.
- **Wachsmuth I.** (1988). Modelling the knowledge base of mathematical learners: situation- specific and situation- non-specific knowledge. Learning

Issues for Intelligent Tutoring Systems (Eds. H. Mandl & A. Lesgold). Springer-Verlag, London, pp63-79.

- **Wallace M. D. & Anderson T. J.** (1993). Approaches to interface design. Interacting with Computers, 5(3), pp259-278.
- **Wan D.** (1994). Experiences with CLARE: a computer-supported collaborative learning environment. International Journal of Human-Computer Studies, 41, pp851-879.
- **Wang S. & Sleeman P. J.** (1993). A comparison of the relative effectiveness of computer assisted instruction and conventional methods for teaching an operations management course in a school of business. International Journal of Instructional Media, 20(3), pp225-234.
- **Wattam S. I.** (1993). Evaluation of Lotus 1-2-3 student edition release 2.3 and associated volumes. Account, 5(1), Spring, pp39-41.
- **Wayner P. & Joch A.** (1995). Agents of Change - beyond the hype, software is getting smarter at helping us solve real problems. BYTE, March, pp95.
- **Webb G. I. & Cumming G.** (1991). Educational evaluation of feature-based modelling in a problem solving domain. Advanced Research on Computers in Education (Eds. R. Lewis & S. Otsuki), Elsevier Science Publishers B. V., North Holland, pp101-108.
- **Wehr J.** (1988). Instructor-led or Computer based: Which will work best for you ? Training and Development Journal (US), June 1988, pp18-21.
- **Weinstein C. E., Goetz E. T. and Alexander P. A.** (1988). Learning and Study Strategies. Academic Press, San Diego, CA.
- **Weinstein C. E., Schulte A. C. & Cascallar E. C.** (1985). The learning and studies strategies inventory (LASSI): Initial design and development. Technical Report, US Army Research Institute for the Social and Behavioural Sciences, Alexandria, VA.
- **Wenger E.** (1987). Artificial Intelligence and Tutoring Systems. Morgan Kaufmann. Los Altos, CA.

- **Wenger E.** (1988). Artificial Intelligence and Tutoring Systems. Morgan Kaufmann, Los Altos, California.
- **Whalley P.** (1993). An alternative rhetoric for hypertext. Hypertext - A Psychological Perspective (Eds. C. McKnight, A. Dillon & J. Richardson), Ellis Horwood, New York, pp7-17.
- **Whetten D. & Cameron K.** (1984). Development Management Skills. Foreman, Scott, London.
- **Whigham D. & Houston J.** (1995). MARKUP - a Computer-based Learning package to allow interactive appreciation of the principles and applications of Average Cost plus Markup Pricing Strategies. Paper presented in CTI-AFM Conference, Glasgow, 28-29 March 1995.
- **White B. Y. & Frederiksen J. R.** (1985). QUEST: Qualitative Understanding of Electrical System Trouble-shooting. ACM SIGART Newsletter, 93, pp34-37.
- **Whitehead A. N.** (1911). Introduction to Mathematics. London press, London.
- **Wilkinson-Riddle G. J. & Patel A.** (1992). Applied research in the development of a software methodology enabling the use of unstructured data in management accounting computer aided learning systems. Paper presented in March 92 at the CTI Conference, Bradford, UK.
- **Wilkinson-Riddle G. J. & Patel A.** (1993). "Human Tutor" Emulation in Teaching Numerical Business Subjects - a Software Breakthrough. Proceedings of the tenth International Conference on Technology and Evaluation, Massachusetts, March 21-24, pp174-176.
- **Williams B. C.** (1993). BITE: Business education with Information TEchnology. The CTISS File, 15 (April 1993), pp67.
- **Williams B. C. & Newton-Ingham G.** (1994). Teaching and Learning: A new approach to Courseware Development in Accounting. Paper presented at the 17th Annual Congress of the European Accounting Association, Venice, April 6-8.

- **Wilson A.** (1989). Accounting systems in industry and public practice. The Accountants Magazine, November, pp76-79.
- **Wilson D. K.** (1986). An investigation of the properties of Kolb's learning style inventory. Leadership and Organization Development Journal, 7(3), pp3-15.
- **Wilson J.** (1990). Hypergraphics: The classroom of the future. THE Technical Horizons in Education Journal, March 90, pp6-8.
- **Wilson M. et. al.** (1986). Using an Expert System to convey HCI information. Proceedings of HCI' 86, "People and Computers: Designing the Interface" (Eds. Harrison M. & Monk A.), Cambridge University Press, London, pp482-497.
- **Wing-Sing C.** (1993). The use of computers in undergraduate accounting education in Hong Kong. Paper presented at the Fifth Annual Conference of Accounting Academics, Hong Kong, 7-8 April, 1993.
- **Witkin H. A.** (1976). Cognitive style in academic performance and in teacher-student relations. Individuality in Learning (Eds. S. Messick et. al.), Jossey-Bass, CA, pp38-72.
- **Wong Shirley M.** (1994). Lessons learned from authoring - Computer Assisted Instruction. Business Education Forum, 48(4), pp39-41.
- **Woodhouse D. & McDougall A.** (1986). Computers: Promise and challenge in education. Blackwell Scientific, Melbourne.
- **Woods P. & Hartley J. R.** (1971). Some learning models for arithmetic tasks and their use in computer-based learning. British Journal of Educational Psychology, 41, pp35-48.
- **Woolf B. P. & Hall W.** (1995). Multimedia Pedagogues: Interactive Systems for Teaching and Learning. Computer, May, pp74-80.
- **Wu F. H.** (1984). Teaching managerial (cost) accounting with electronic spreadsheet software. Issues in Accounting Education, 7(3), pp81-97.
- **Wyatt J. & Spiegelhalter D.** (1990). Evaluating medical expert systems: what to test and how? Medical Informatics, 15(3), pp205-217.

- **Yazdani M.** (1986). Intelligent tutoring systems survey. Artificial Intelligence Review, 1, pp43-52.
- **Yazdani M.** (1987). Intelligent Tutoring Systems: an overview. Artificial Intelligence and Education (Eds. Lawler R. & Yazdani M.), Ablex, Norwood, pp182-201.
- **Zissos A. Y. & Witten I. H.** (1985). User Modelling for a Computer Coach: a case study. International Journal of Man- Machine Studies, 23, pp729-750.

Appendix A

Glossary

AI	Artificial Intelligence
CAL	Computer Aided Learning
CBI	Computer Based Instruction
CILE	Computer Integrated Learning Environment
DLL	Dynamic Link Libraries
Expert System	A computer program that uses a knowledge base and inference procedures to act as an expert in a specific domain. It is able to reach conclusions very similar to those reached by a human expert.
GUI	Graphical User Interface
HCI	Human-Computer Interaction
ICAI	Intelligent Computer Assisted Instruction, synonym for Intelligent Tutoring System.
ITS	Intelligent Tutoring System, a computer program that (a) is capable of competent problem solving in a domain, (b) can infer a learner's approximation of competence, and (c) is able to reduce the difference between its competence and the student's through application of various tutoring strategies.
ITT	Intelligent Tutoring Tool
ITTN	Intelligent Tutoring Tools Network
LSI	Learning Style Inventory
LSQ	Learning Style Questionnaire
TLTP	Teaching and Learning Training Programme
TST	Teaching Support Tool, developed under project Byzantium, such as Marker

Appendix B

Review of Some CAL Software Packages in Accounting

The first few examples are of historical interest, and these are followed by the current state of the art systems. Although there is no clear-cut dividing line, the last four examples namely: Understand ACCOUNTS, EQL Interactive Taxation Tutor, Stella II, and Financial Management can be considered as clearly state of the art as they use latest software and hardware technology and provide advanced graphical and audio interfaces available nowadays. No intelligent tutoring systems could be reviewed, as no intelligent tutoring system in accounting domain is known till date except the in-house modules which are described in Chapter 7 and 8.

➢ **Accounts Trainer**

Source: G. J. Wilkinson-Riddle and B. E. Barker (Leicester Polytechnic)

Release Date: 1988

Details: The courseware consists of two discs (one tutorial, one interactive), and a booklet. The booklet consists of seven chapters, at the beginning of each of which, objectives are set and clearly stated.

Tutorial One commences with an optional keyboard familiarity exercise and is mainly instructive with a large amount of text to read. The accounting equation is introduced and a short exercise at the end prepares the students for *Interactive Mode One*. A facility to produce a printout of the result of the exercise is available. In *Interactive Mode One* the task is to create a balance sheet from a list of opening assets and creditors affected by a number of transactions. Following the worked example there are five further exercises which may be attempted. Printouts giving a summary of transactions and a final balance sheet are available at the end of each exercise. Text recognises the concept of profit as distinct from capital in *Tutorial Two*. "T" accounts and the profit and loss account are introduced at this stage and the tutorial concludes with a double entry exercise. The task in *Interactive Mode Two* is to create a balance sheet in "T" account form. The figures are same as those in example one. *Chapter Four* introduces the trial balance and explains how to construct the trading and profit and loss account from the trial balance. *Chapter Five* gives a brief introduction to accounting conventions and principles. *Interactive Mode* Three in conjunction with *Chapter Six* requires the student to sort a trial balance into the categories: Asset; Creditor; Expense; Capital; Income; and make adjustments for depreciation, accruals, closing stock and bad debts. *Chapter Seven* finally explains provision accounts for depreciation and bad debts.

Features

- The booklet is set out in simple English.
- Objectives are set at the beginning of each chapter and the contents of the chapter are relevant and comprehensive.
- The computer provides a skeleton framework for data to be entered from the book, whereas some courseware has examples built into the software so limiting the number of exercises which may be attempted.
- The teacher may construct his own exercises to illustrate relevant points.

Limitations

- The sequence in which the courseware should be executed is not clearly indicated.
- It is difficult for the student to skip sections of the tutorial if he is not aware of what each page consists of.
- The tutorials are mainly instructive and the student is expected to read many pages of text - a method which may not successfully hold the students interest and hence reading from book may equally, if not more, be effective. The tutorials are in monochrome.
- Each transaction entered into the computer must be identified by a given reference number and categorised by one of seven transaction codes. It may

be difficult for the student to memorise meaningless codes in addition to learning new concepts.

- The balance sheet is presented in a way which does not confirm with the recommendations of the Companies Act 1985. This does not adequately prepare the students to subsequently deal with long term and current liabilities, or to interpret financial statements which are likely to be in different format.
- Chapter four refers to "movement of funds statement" which adds opening capital to net profit, retained profit and new capital introduced, to give closing capital. This could lead to a confusion of terms if the student subsequently encounters "Funds Flow Statements".

➢ **Financial Accounting for Non Accountants**

Source: IVY Software

Release Date: 1989

Details: The courseware consists of four discs and a booklet. *Chapter One* in the booklet gives a general introduction to financial accounting. Fundamental concepts and principles are explained, and the requirements of the various users of financial accounts are identified.

Each of the following chapters may be read from the book or the screen. Chapter Two gives a brief introduction to financial statements. It follows a progress test consisting 27 questions. The next stage involves a more detailed explanation of the components of the profit and loss account, which is gradually constructed. Four further progress tests may be attempted, each following the introduction of a new concept. In *Chapter Three* ledger accounts are introduced together with the concept of double entry. There are three progress tests which may be attempted during the course of the chapter. *Chapter Four* begins with a graphical illustration of the flow of funds in and out of a business. There are three progress tests afterwards and a worked example. *Chapter Five* is concerned with the analysis of financial statements. Each one of 28 accounting ratios is explained in detail and relationships between them are highlighted. A progress test is followed by a company case study which can only be read from the booklet. This is followed by a detailed explanation of the actions which should be taken to overcome the problems experienced by the company.

Features

- The instructive text on the disc for chapters 2, 3, 4 and 5 can also be found in the booklet. This gives the students the choice to work at the computers or away from it.
- At the end of each progress test the users are advised of their scores and percentages, together with the length of time taken to complete the test. This may be a useful function not only for self-assessment by the students, but also if the teacher sets target for the students to achieve before progressing to the next section.
- Colour is used to highlight key concepts and to distinguish between profit increasing and reducing items. Colourful graphics are also used occasionally, thus enhancing presentation. The colour of the text and background changes frequently thus preventing monotony.
- It is possible to move from one section to another at almost any time, sections can be skipped thus a student could use the progress tests for revision purposes.

Limitations

- At some stages, the student is faced with pages full of text to read, which is not a desired technique.

- There is a standard response to incorrect answers in the progress tests regardless of the reason why the answer is wrong, thus the mistake anticipated may not be the mistake made.
- Considering the fact that this package is designed for non accountants as its title suggests, the course includes too much detail for beginners.
- The course can be repetitive by placing too much emphasis on a single concept.
- The chapters are split into short sections with progress tests at the end of each one. Consequently, some of the tests are too easy as they refer directly to the preceding text and the answers are obvious.
- Chapter Four refers to two approaches to funds flow analysis. This may prove to be confusing to the beginners.

➢ **PINSTRIPE**

Source: R. Hornsby, P Thompson and A Ramsdale (Humberside Business School)

Release Date: Not Known

Details: The courseware consists of three discs. The first disc "DEBIT" begins with an introduction to BACON (Basic Accounting CONcepts) followed by a section entitled "How we suggest you use this package". This basically sets out the purpose of each section whilst "The aims of DEBIT" sets out the objectives. The next stage is a tutorial consisting of six sections and a quiz. The instructive sections introduce the ledger and the concept of duality. The quiz consists of questions in the form of multiple choice. The next section of the course is a case study. The background of the case study is given and the student is to assume the role of management who require calculations of various "mixed value measures" from trial balance totals. Subsequent quiz consists of five transactions. Immediate feedback is given from the quiz in terms of message "correct" or "incorrect - please try again". The progress test consists of five similar transactions, but here feedback is delayed. A result screen may be viewed comparing the students answers with the correct answers.

The second disc "BALANCE" is structured in a similar way to "DEBIT". There is an introduction to BACON followed by instructions on how to use the package. The objectives are set out in "The aims of BALANCE" which leads to a description of the fundamentals of the balance sheet. The third disc "SITCOM" differs in structure and content from the first two discs. SITCOM does not use BACON, but HASLA (Humberside Algorithmic System of Learning Accounting) in which an algorithm is constructed to show the flow of accounting information. The student is required initially to enter the figures into a trial balance and subsequently transfer the balances to either the profit and loss account or balance sheet as appropriate, choose a value for closing stock, calculate totals for the profit and loss account and balance sheet, and also calculate ratios. The results of the ratios may be displayed and compared in order to determine which scenarios give the best results.

Features

- The package allows the use of the "Page Up" and "Page Down" keys and hence it is easy to use as a book.
- The student may exit from the procedure at almost any stage and save work if necessary.
- The codes used in DEBIT are easy to remember as they represent the initial letter of the transaction type.
- SITCOM provides good practice for constructing a profit and loss account and balance sheet from a trial balance.

Limitations

- The student is often faced with full pages of text presented in complicated language.

- The package uses colour reasonably well for the exercises and illustrations, but the instructive text is mainly in black and white.
- The multiple choice quiz questions consist of four false statement and one true statement, this may confuse the student who has previously been learning facts from the screen.
- Whilst immediate feedback is supplied for the quiz questions and tutorial examples, the results of the progress tests are not given until the end of the exercise. It breaks the consistency of the approach.
- Overall the course differs from conventional accounting courses to the extent that it may not be accepted by the teachers and may confuse students with some previous accounting knowledge.

➢ **PLATO** **Financial Statements: Structure and Content.**

Source: G Carr

Release Date: 1985

Details: The first stage is to read the text in the booklet, consists of four main sections. *Section One* is concerned with the background to, and functions of, financial reporting. *Section Two* gives a concise introduction to the balance sheet layout, followed by a more detailed description of assets and their valuation, shares, reserves and long term loans. *Section Three* introduces the concept of funds flow as a cause of the change in balance sheet values from one year to the next. There follows a practical exercise to construct a funds flow statement from two years' balance sheet details. *Section Four* is concerned with profit and introduces the accruals (matching) and the realisation concepts.

The next stage of the course can be found on the disc. There is an optional introduction to the keyboard followed by the first exercise. The first exercise, "Key Terms", consists of a series of questions relating to the material in the booklet. Exercise two, "Concepts", displays a balance sheet and there follow eight questions on balance sheet items, and a further fifteen questions on balance sheet items, and a further fifteen questions relating to the general text of the booklet, but with a bias towards balance sheet items and transactions. The third exercise is in the form of a tutorial and consists of four parts: Balance sheet; Profit and Loss; Funds flow; and Funds flow interpretation. The fourth exercise on the disc is entitled "Balance Sheet". A balance sheet is displayed at the top of the screen and questions relating to it are displayed at the bottom of the screen.

Features

- The screen layout is reasonably effective in the sense that good use is made of colour to highlight key words. The screen fills up slowly so that the student is not overwhelmed with pages full of text.
- Progress reports are given whilst executing the first two sections of the main menu e.g. "You have scored 7 out of 10". The student must re - attempt the questions which were originally answered incorrectly. Initially an incorrect answer will not prompt an explanation, rather an indication as to in which section of the booklet the correct answer may be located. However, if at the second attempt the student still answers incorrectly, the correct answer will be supplied, together with brief explanation and the relevant section of the booklet.

Limitations

- The students are given no indication as to what section of the course they are working on. There are two menus - a main menu and, from option 3 of the main menu, a tutorial menu. However, the students are not advised when they have completed one section and proceeded to the next.
- The major criticism of the courseware is that it covers a lot of detail too briefly, and as such may be of more use for revision purposes.

- There is no "help" facility available to the student, but the procedure is easy to follow, even though questions and instructions are not always located in the same section of the screen.

➢ TRAINING PACKAGE

Source: Institute of Chartered Accountants of Scotland.

Release Date: 1989

Details: The courseware consists of one disc. The disc holds details of five lessons, a second disc holding a further five lessons must be obtained in order to complete the course. *Lesson One* is entitled "Double Entry - Balance Sheet" and commences with a comprehensive introduction to accounting, in which the purpose of financial accounts, and the steps involved in their preparation, are identified. *Lesson Two* "Balancing Off Accounts" demonstrates how "T" accounts are balanced off and their balancing figures carried down and used to construct a trial balance. *Lesson Three* "Double Entry - Profit and Loss Account" introduces the profit aspect and the use of expense accounts and income accounts is incorporated into the transactions. *Lesson Four* "Profit and Loss Account and Balance Sheet" introduces the trading account, cost of sales and gross profit. *Lesson Five* "Profit and Loss And Balance Sheet - Further Considerations" builds on the concepts introduced in the previous four lessons.

Features

- The students have control over the speed at which the screen fills up, i.e. the following piece of text is not displayed until the students carry out the command "Hit any key to continue".
- The text is easy to understand and to follow.
- At the beginning of each lesson the students are made aware of what should have been learned in the previous section, and what is to be learned in the

current section. If they feel the objective of previous lesson has not been achieved, they can return to it before proceeding to the next section.

- Colour is used consistently throughout the course in the sense that accounts are colour coded and key words and phrases relating to these accounts are highlighted in the appropriate colour. Account identification codes are meaningful and they are always displayed on the screen when they are required, hence the student does not have to spend extra time learning or referring to the text for the codes.
- Help is always available throughout the course and a message at the top left of the screen advises the student of his current position in the course. Sections may be skipped or revisited at any time.
- The examples in the first four lessons follow on from each other, hence the students can see the accounts of three businesses gradually build up. This continuity does not however prevent the students from skipping a section as each individual concept is clearly explained as distinct from all of the others.
- The questions are of good standard and relevant to the preceding text.
- There is no booklet accompanied with the software so students do not require divert their attention from the screen.

Limitations

- As the course is entirely disc based, the student cannot study in his own time unless he has access to a computer.
- If a question is answered incorrectly, a standard message appears on the screen. Students can know that they are wrong, but do not get the explanation of the mistake.

➢ **QSB+ (Quantitative Systems for Business Plus)**

Source: Yih-Long Chang and Robert S Sullivan (Prentice-Hall)

Release Date: 1989

Details: Software package contains the most popular problem - solving algorithms in management science. Manual is also included for guidance. The topics covered are: Linear programming; Mixed integer programming; Transportation problems; Network models, critical path analysis and PERT; Dynamic programming; Inventory theory: deterministic and probabilistic; Queuing theory; Decision and probability theory: weighted mean and variance analysis, bayesian analysis, payoff tables, decision trees and Markov process; Time series forecasting: moving averages, exponential smoothing, linear regression and Winter's model.

Features

- Most of the modules have an optional facility for displaying detailed solution procedures which will assist students in understanding the operation of the algorithms.
- Menu system is easy to use and input procedures are relatively friendly.
- Modules are generally self-explanatory with on - screen instructions and overview; little, if any, reference to the manual is necessary.
- Manual includes an illustration of each module and several example problems with solutions.
- Linear programming modules support files in MPS format.
- Graphical solution to two variable linear programming problems can be displayed.
- Critical path method includes a crash analysis facility.

Limitations

- An immediate mechanism for cancelling the current operation is not always provided which makes it more difficult to correct errors.
- Free format data entry for linear programming problems is rather cumbersome.

- Regression analysis does not calculate standard errors or coefficient of determination.
- Graphics is only supported in CGA low resolution mode.
- Software is rather awkward to install on a hard disc or file server since it requires the modules to be in the default directory and has no provision for the user to change directory.
- Manual contains no background or explanation to the techniques being used.

➢ **Microcomputer models for Management Decision - Making**

Source: Terry L Dennis and Laurrie B Dennis (West Publishing)

Release Date: 1988

Details: Software teaches a number of decision models in common use within business. The topics covered are: Linear programming; Mixed integer and binary programming; Goal programming; Transportation problems; Assignment problems; Network models and PERT; Forecasting: moving averages, exponential smoothing and regression analysis; Simulation: Monte-Carlo, queuing and inventory models; Decision theory; Markov analysis.

<u>Features</u>

- Menu structure and data entry procedures are fairly easy to use.
- Options to display intermediate results during solution process make better understanding of the subject.
- Manual contains good explanations of how to use each module together with a number of sample problems with solutions.
- Deterministic inventory models are provided to take into account backorders and discounts.
- Probabilistic inventory models are available for discrete, continuous and single period demand problems.

Limitations

- It is difficult to cancel an option selected from the menu.
- No graphical displays of solutions provided.
- The software is copy protected and this is not mentioned anywhere other than on the disc itself which is sealed on purchase.
- Software is rather awkward to install on a hard disc or file server since it requires the program to be in the default directory although it does allow a path when specifying filenames.
- Manual contains no background or explanation to the techniques being used.

➢ **Understand ACCOUNTS**

Source: Interact (Distributed by EQL International Ltd.)

Release Date: 1994

Details: The software is a Windows based package designed to be used without manuals or workbooks. It consists of a set of tutorial modules followed by an assessment module. A separate reporting module is available to the tutors who can use this to track student use of the software and their performances in the assessment.

The tutorial modules cover the topics: Introduction to accounts; The balance sheet; The profit and loss account; Accounting fundamentals; Profitability; Cash and liquidity; Business and gearing risk; Investor ratios. The objectives are not explicit in the software itself though the accompanying booklet intended for the tutor or installer does describe the objectives for each module. No prior knowledge of company accounts is assumed. The learner is given the opportunity to perform arithmetic calculations, to build up graphs and to exercise judgement.

Features

- Learning is reinforced by a review section at the end of each module.
- The content is well presented and dealt with in a logical progression.
- Colours are used well for highlighting and for emphasis, but a colour monitor is essential.
- The layout is very imaginative with good use of graphics and some simple animation.
- The system accepts different forms of the same numerical answer e.g. both 6000 and £6,000 are accepted forms.

Limitations

- No calculator is available within the package. Many of the calculations to be performed cannot easily be done mentally and a calculator to hand is a must.
- Glossary is accessible only from the Main Menu screen. Although this can be reached from any page, but it is not possible to return to the page learner has come from.
- User has limited control over the flow of the program. Although it is possible to skip through the module, each step takes the learner to predetermined points.
- No learner documentation is provided, although there is very little need.

➢ **EQL Interactive Taxation Tutor**

Source: EQL International Ltd.

Release Date: 1994

Details: The package aims to familiarise users with personal and business tax in the UK. There are nine modules and a reporting module which lists all users and records their performance. For each user the system records the modules completed, started but not completed and those not yet started. The date of last access and the time for which the user is within the system is noted. Also recorded is the percentage correct of the interactions attempted.

The modules included are: Introduction to the UK tax system; Personal Income Tax; Schedule D Income Tax; Capital Allowances; Corporation Tax; Capital Gains Tax; Value Added Tax.

Features

- The screen are clear, interesting and colourful and the material is well presented.
- An on-line calculator is available and a glossary can be accessed from the menu bar.
- Three options are available when a question is answered wrongly for the first time, 'try again', 'hint' and 'explain'. When wrong the second time the hint option is replaced by 'show answer', and the user can view the answer and the working required to arrive to it.

Limitations

- Software needs at least 9 Mb free hard disk space to load the files.

➢ **Stella II Version 3.0 (Windows Version)**

Source: High Performance Systems Inc. (distributed by Cognitus)

Release Date: 1995

Details: The software provides a multi-level, hierarchical modelling environment which enables model construction and simulation to take place using the systems dynamics methodology. The software comes with a vast armoury of documentation including the following:

- a getting started booklet covering the essential elements of model building;
- a technical manual which explains how to use the software in more detail;

- an introduction to systems thinking;
- an applications manual which provides illustrations on how the software might be used using the systems dynamics methodology;
- an authoring manual which provides guidance for the model builder in terms of designing a user friendly environment.

Stella has been accessible on the Apple Macintosh for last seven years and recently has become available under MSDOS and Windows operating systems. Models developed within a Macintosh environment can be translated into a Windows environment, and vice versa, with ease.

Stella is of particular value to model users in terms of communicating and making visual the relationships created between variables within a model (Ballantine J A & Whittington M, 1994). The software could be used by both novice and expert builders. Stella II lowers the technical and conceptual burden on modellers, by relying on 'graphical tools to support conceptualisation, construction, analysis and communication activities'. The objectives of this software are:

- it can be used to create an environment of active classroom learning, focusing discussion and keeping all students involved in the learning process;
- it provides a discipline that helps students understand text book information;
- it helps students understand the interplay of variables through time in terms of gaining an understanding of systems dynamics;
- it helps develop critical thinking and modelling skills through the process of model construction, extension and testing;
- it helps facilitate the development of understanding within the classroom context.

Features

- Software has wide usability area and can be used for financial modelling, management accounting, finance and accounting, strategic management, economics, production and operations management and information systems. In addition, discrete event simulation modelling is possible.
- No previous knowledge or experience of modelling or system concepts are required for the use of this software.
- Screen layouts are well presented and easy to read.
- The software makes considerable use of icons to perform common tasks (for example, deleting objects) and selecting system objects (for example, connectors and converters).
- Different colours and shapes are used by the software to clearly distinguish between different elements of the system, thereby providing greater clarity.
- Software automatically checks for errors in formula input, by ensuring that the elements outlined in the conceptual systems map agree with the elements used in the formula created by the modeller.
- Alterations to formulae and input data are easy, data input being facilitated through either graphics or tables.
- Software prompts the users to save their work on exiting from the system.
- Stella is easy to install, with clear instructions provided by the software vendors.
- Stella enables the user to copy and paste results in the form of tables to other Windows products.

Limitations

- Software has a more limited range of functions than found in more recent spreadsheet versions.
- On screen instructions for operators are minimal, and where provided, are not particularly informative when used in isolation of the accompanying documentation.

- There is no on-line help available in the software and user has to rely entirely on printed documentation.

➢ **Financial Management**

Source: Interactive Services Ltd. (jointly developed with the Institute of Chartered Accountants in Ireland (ICAI).

Release Date: 1995

Details: The software provides 12 modules covering basic concepts in financial management. Software consists of 6 disks for installation. Minimum specification needed is VGA monitor, 286 processor, 2Mb RAM, and 12 Mb free disk space. The program comprises a *course*, which students interact with, and a *management* module which records students' progress through the course.

Each of the 12 modules opens with a menu of the lessons contained within it, and as students work through a lesson, the box to the right of the lesson title changes colour to indicate partial or full completion. Lessons are divided into topics, and within each topic may be sub-divided into sections. The student is not obliged to follow a linear sequence through the material, any item can be selected at any stage, except for the mergers/takeovers modules where a pre-requisite is stated.

Navigation within a module is helped by a standard set of options to reverse/continue, return to the menu, Quit, or access the global Glossary. Each section of the course contains questions to enable students and their teachers to assess their progress.

<u>Features</u>

- The program runs both under DOS and Windows environment.

- The program can be used either stand-alone or networked, and networking instructions are included in the documentation.
- The coverage of financial management topics is very broad and all significant subject are covered to some extent.
- The glossary is accessible at any point, enabling the user to look up any unfamiliar terms.
- The illustrations in the program are quite useful and do not require high-specification equipment.

Limitations

- No installation instructions for Windows are included with the package.
- Anyone working through the program needs to equip themselves with pen, paper and calculator, although this is not specified either on the screen or in the documentation.
- Program does not retain a marker at the point of last access, so it is up to the students to remember where they stopped. As there is a menu structure at each level through modules, lessons, and topics, it is quite possible to become confused about the last point of access.
- Some theoretical issues are rather superficially covered, and some complex issues, such as dividend policy, are dealt with rather simplicity. For example, theory of foreign exchange rate movement is covered without any evaluation of the theories.
- There is no overall index.
- There are inconsistencies in the glossary. For example, *Cum dividend* is there, but *ex dividend* is not and yet the text refers to both.
- Users cannot skip the questions without answering them even if they want to read the tutorials again before attempting those questions.
- Students themselves have no means of recording their progress through the material.
- There is no means to print the user guide and practice examples.

➢ **BITE's Pandora Courseware**

Source: Univ. of East Anglia

Release Date: 1995

Details: A broad coverage of business management, accounting and finance topics. The software is distributed via Internet, CD ROM and floppy disk and is available in Windows and Macintosh versions.

Pandora comprises six modules based on the metaphor of a shopping and business plaza. The plaza, which is the base of the content in the courseware, has been produced as a mathematical model. The modules covered in Pandora are Performance Reports, Working Capital Control, Budgetary Planning and Control, Absorption versus Variable Costing, Cost-Volume-Profit / Break-even Analysis, and Investment Appraisal. In addition, there is a computer desktop glossary or terms for use while revising, essay writing etc. The courseware is developed in a purpose built authoring system in a highly sophisticated programming environment.

Features

- The courseware has taken a lot of advantage of user feedback during its design and development stage.
- The courseware has a high quality colour graphics with a three dimensional feel and is designed to enhance the quality, thoroughness and speed of student learning by enabling students to manage their own learning experiences.
- The courseware is user friendly and does not assume students having any prior computer or subject knowledge.
- Navigation to any part of the program is possible at any time through the *Security* feature. Back and forth navigation is also possible in the program.
- Help is available at all time throughout the program.

- Context sensitive hypermedia controlled tutorials are available corresponding to the mouse position.
- The program adopts an exploratory game approach which enables it to provide learning to different types of users.
- Assessment of learning is also possible with the help of tutorial examples.
- The courseware is available at nominal charge which makes it accessible to a large student community.

Limitations

- The program needs multimedia system with a large amount of memory.
- The hypermedia approach adopted by the program inherits the deficiency of getting the user lost in the links (Pinker, *ibid.*).
- The game approach may focus attention away from the subject matter and a student might become more interested in attending the 'game objective' as opposed to 'tutoring objective'!

Appendix C

Capital Investment Appraisal module

Pre Test **Post Test**

NAME :

COURSE :

Capital Investment Appraisal
Knowledge Acquisition Test

Question 1:
If you invest £20,000 in a project and you invest further £5,000, what is your net cash outflow?

Answer:

Question 2:
If you have a cash inflow of £12,000 during January and a cash outflow of £7,000, also during January, what is the amount of net inflow or net outflow?

Answer:

Question 3:
A project costs £20,000 in year 1983 and earns the following returns:-

Year	Return
1983	£10,000
1984	£ 5,000
1985	£ 6,000
1986	£ 6,000
Residual value	£ 0

(a) During which year will the investment be returned?

Answer:

(b) What is average investment for the project?

Answer:

(c) What profit (return) does the project provide at the end of 1986?

Answer:

(d) What profit (return) does the project provide at the end of 1986 if interest rates are 10% p.a.

Answer:

(e) What percentage return does the project provide if interest rates are 10% p.a.

Answer:

NAME :

COURSE :

Capital Investment Appraisal
Post Knowledge Acquisition Test

The following information is available for two projects:

	Project A		Project B	
	Inflows	Outflows	Inflows	Outflows
Investment		60000		60000
Year 1	22000	6000	33000	3500
Year 2	25000	2000	35000	4500
Year 3	25000	4000	21000	8000
Year 4	20000	3000	14000	7000
Year 5	17000	4000	10000	7800
Residual Value	0		0	

Interest Rate (Required Rate of Return) 12% per annum

Question 1
Calculate the following values:

	Project A	Project B
Average Investment		
Cumulative Netflow at the end of year 5		

Question 2

Calculate the following values:

	Project A	**Project B**
Payback		
Net Present Value		
Accounting Rate of Return		

Question 3

State, which project would you choose for investment and briefly give reasons for your choice.

Appendix D

Absorption Costing module

Pre Test **Post Test**

NAME :

COURSE :

Absorption Costing
Knowledge Acquisition Test

Data

Budgeted information relating to two departments in Rydons Tables Limited for the next period is as follows:

Department	*Production overhead*	*Direct material cost £*	*Direct labour cost £*	*Direct labour hours*	*Machine hours*
1	27,000	67,500	13,500	2,700	45,000
2	18,000	36,000	100,000	25,000	300

Further data

During the period, job number 9287 is carried out by Rydons Tables Limited. Production data is as follows:

Direct material cost	- department 1	£30
	- department 2	£10
Direct labour	- department 1	4 hours at £5 per hour
	- department 2	9 hours at £4 per hour
Machine hours	- department 1	65 hours
	- department 2	1 hour

Individual direct labour workers within each department earn differing rates of pay, according to their skills, grade and experience.

Question 1:

What is the most appropriate production overhead absorption basis for Department 1? (e.g. Direct material cost, Direct labour hours etc.)

Answer:

Question 2:
What is the most appropriate production overhead absorption basis for Department 2?
Answer:

Question 3:
a) Calculate the appropriate OAR for department 1.
Answer:

b) Calculate the appropriate OAR for department 2.
Answer:

c) Calculate the department job cost for job 9287 in department 1.
Answer:

d) Calculate the department job cost for job 9287 in department 2.
Answer:

e) What is the total production cost of job 9287 using overhead absorption rates based on your selection in questions 1 and 2?
Answer:

NAME :

COURSE :

Absorption Costing
Post Knowledge Acquisition Test

Data

Ryton Dentists employ 5 people in total, 2 in reception and 3 within the surgery. They have the following floor space and budgeted for the following direct labour hours:

	Reception	*Stores*	*Surgery*	*Total*
Floor space	15	10	25	50
Direct labour hrs.	50	0	100	150
Indirect labour				1,500
Light/Heat				675
Rent/Rates				460

Question 1

a1) What is the Indirect Labour Cost apportioned to reception?

a2) What is the Rent/Rates Cost apportioned to Surgery?

b1) What is the Stores Cost apportioned to Reception?

b2) What is the Stores cost for apportioned to Surgery?

Question 2

a1) What is the total O/H Burden for Reception?

a2) What is the total O/H Burden for Surgery?

b1) What are the Production Hours on which the Reception Overhead will be absorbed?

b2) What are the Production Hours on which the Surgery Overhead will be absorbed?

c1) What is the OAR for Reception?

c2) What is the OAR for Surgery?

Question 3

Determine whether or not there is any under or over absorption if the actual direct labour hours are:

- Reception	65
- Stores	0
- Surgery	110

and the actual overhead incurred is 2950.

Appendix E

Marginal Costing module

Pre Test **Post Test**

NAME :

COURSE :

Marginal Costing
Knowledge Acquisition Test

A company makes microwave ovens and sells them in its own retail outlets at a sale price of £ 99.99 each, before charging VAT. In the month of September 1995, 11025 units were manufactured and sold. Variable cost per unit was £ 86.74 and the fixed cost for the month was £ 93810.

Required: (please show all the workings)

1. What was the total sales revenue of the company for September '95 ?

2. How much contribution (towards the fixed costs and profit of the company) was generated by these sales in September '95 ?

3. What was the monthly profit of the company for September '95 ?

4. At least how many units should have been sold in September '95 to ensure that the company did not incur any monthly loss ?

5. The variable cost per unit and the fixed costs are not likely to alter in the following month of October '95, but the last week of September saw a decline in the demand and it is felt that the October sales might drop by around 30%. By how many units (as compared to September) can the sales drop, before the company incurs a monthly loss for October '95 ?

6. It is, however, believed that the current level of sale i.e. 11025 units can be maintained in October '95, if the sale price was dropped by 13.25%. What will be the monthly profit/loss for October, if this strategy was adopted ?

NAME :

COURSE :

Marginal Costing
Post Knowledge Acquisition Test

A company manufactures Black and White Television sets. It sells them in its own retail shops at a sale price of £ 75.99 each, before charging VAT. In the month of July 1995, 1760 units were manufactured and sold. Variable cost per unit was £ 36.74 and the fixed cost for the month was £ 52595.

Required: (please show all the workings)

1. What was the total sales revenue of the company for July '95 ?

2. How much contribution (towards the fixed costs and profit of the company) was generated by these sales in July '95 ?

3. What was the monthly profit of the company for July '95 ?

4. At least how many units should have been sold in July '95 to ensure that the company did not incur any monthly loss ?

5. The variable cost per unit and the fixed costs are not likely to alter in the following month of August '95, but the last week of July saw a decline in the demand and it is felt that the August sales might drop by around 30%. By how many units (as compared to July) can the sales drop, before the company incurs a monthly loss for August '95 ?

6. It is, however, believed that the current level of sale i.e. 1760 units can be maintained in August '95, if the sale price was dropped by 51.65%. What will be the monthly profit/loss for August, if this strategy was adopted ?

Appendix F

Learning Style Questionnaire

***

Learning Styles Questionnaire

***

This questionnaire is designed to find out your preferred learning style(s). Over the years, you have probably developed learning 'habits' that help you benefit more from some experiences than from others. Since you are probably unaware of this, this questionnaire will help you pinpoint your learning preferences so that you are in a better position to select learning experiences that suit your style.

It will probably take you 10-15 minutes to complete this questionnaire. The accuracy of the results depends on how honest you can be. There are no right or wrong answers. If you agree more than you disagree with a statement put a tick by it ☑. If you disagree more than you agree put a cross by it ☒. Be sure to mark each item with either a tick or cross.

Your name..

Your course..

☐ 1. I have strong beliefs about what is right and wrong, good and bad.

☐ 2. I often act without considering the possible consequences.

☐ 3. I tend to solve problems using step-by-step approach.

☐ 4. I believe that formal procedures and policies restrict people.

☐ 5. I have a reputation for saying what I think, simply and directly.

☐ 6. I often find that actions based on feelings are as sound as those based on careful thought and analysis.

☐ 7. I like the sort of work where I have time for thorough preparation and implementation.

☐ 8. I regularly question people about their basic assumptions.

☐ 9. What matters most is whether something works in practice.

☐ 10. Actively I seek out new experiences.

☐ 11. When I hear about a new idea or approach I immediately start working out how to apply it in practice.

☐ 12. I am keen on self discipline such as watching my diet, taking regular exercise, sticking to a fixed routine etc.

☐ 13. I take pride in doing thorough job.

☐ 14. I get on best with logical, analytical people and less well with spontaneous, 'irrational' people.

☐ 15. I take care over the interpretation of data available to me and avoid jumping to conclusions.

☐ 16. I like to reach a decision carefully after weighing up many alternatives.

☐ 17. I am attracted more to know well, unusual ideas than to practical ones.

☐ 18. I don't like disorganised things and prefer to fit things into a coherent pattern.

☐ 19. I accept and stick to laid down procedures and policies so long as I regard them as an efficient way to getting the job done.

☐ 20. I like to relate my actions to a general principle.

☐ 21. In discussions I like to get straight to the point.

☐ 22. I tend to have distant, rather formal relationships with people at work.

☐ 23. I thrive on the challenge of tackling something new and different.

☐ 24. I enjoy fun-loving, spontaneous people.

☐ 25. I pay meticulous attention to detail before coming to a conclusion.

☐ 26. I find it difficult to produce ideas on impulse.

☐ 27. I believe in coming to the point immediately.

☐ 28. I am careful not to jump to conclusions too quickly.

☐ 29. I prefer to have as many sources of information as possible - the more data to think over the better.

☐ 30. Flippant people who don't take things seriously enough usually irritate me.

☐ 31. I listen to other people's points of view before putting my own forward.

☐ 32. I tend to be open about how I am feeling.

☐ 33. In discussions I enjoy watching the manoeuvrings of others participants.

☐ 34. I prefer to respond to events on spontaneous, flexible basis rather than plan things out in advance.

☐ 35. I tend to be attracted to techniques such as network analysis, flow charts, branching programmes, contingency planning, etc.

☐ 36. It worries me if I have to rush out a piece of work to meet a tight deadline.

☐ 37. I tend to judge people's ideas on their practical merits.

☐ 38. Quiet, thoughtful people tend to make me feel uneasy.

☐ 39. I often get irritated by people who want to rush things.

☐ 40. It is more important to enjoy the present moment than to think about past or future.

☐ 41. I think that decisions based on a thorough analysis of all the information are sounder than those based on intuition.

☐ 42. I tend to be a perfectionist.

☐ 43. In discussions I usually produce lots of spontaneous ideas.

☐ 44. In meetings I put forward practical, realistic ideas.

☐ 45. More often than not, rules are there to be broken.

☐ 46. I prefer to stand back from a situation and consider all the perspectives.

☐ 47. I can often see inconsistencies and weaknesses in other people's arguments.

☐ 48. On balance I talk more than I listen.

☐ 49. I can often see better, more practical ways to get things done.

☐ 50. I think written reports should be short and to the point.

☐ 51. I believe that rational, logical thinking should win the day.

☐ 52. I tend to discuss specific things with people rather than engaging in social discussion.

☐ 53. I like people who approach things realistically rather than theoretically.

☐ 54. In discussions I get impatient with irrelevancies and theoretically.

☐ 55. If I have a report to write I tend to produce lots of drafts before settling on the final version.

☐ 56. I am keen to try things out to see if they work in practice.

☐ 57. I am keen to reach answers via a logical approach.

☐ 58. I enjoy being the one that talks a lot.

☐ 59. In discussions I often find I am the realist, keeping people to the point and avoiding wild speculations.

☐ 60. I like to ponder many alternatives before making up my mind.

☐ 61. In discussions with people I often find I am the most dispassionate and objective.

☐ 62. In discussions I'm more likely to adopt a 'low profile' than to take the lead and do most of the talking.

☐ 63. I like to be able to relate current actions to a longer term bigger picture.

☐ 64. When things go wrong I am happy to shrug it off and 'put it down to experience'.

☐ 65. I tend to reject wild, spontaneous ideas as being impractical.

☐ 66. It's best to think carefully before taking action.

☐ 67. On balance I do the listening rather than the talking.

☐ 68. I tend to be tough on people who find it difficult to adopt a logical approach.

☐ 69. Most times I believe the end justifies the means.

☐ 70. I don't mind hurting people's feelings so long as the job gets done.

☐ 71. I find the formality of having specific objectives and plans stifling.

☐ 72. I'm usually one of the people who puts life into a party.

☐ 73. I do whatever is expedient to get the job done.

☐ 74. I quickly get bored with methodical, detailed work.

☐ 75. I am keen on exploring the basic assumptions, principles and theories underpinning things and events.

☐ 76. I'm always interested to find out what people think.

☐ 77. I like meetings to be run on methodical lines, sticking to laid down agenda, etc.

☐ 78. I steer clear of subjective or ambiguous topics.

☐ 79. I enjoy the drama and excitement of a crisis situation.

☐ 80. People often find me insensitive to their feelings.

Appendix G

Subjective Questionnaire

Analysis of a tutoring system

Questionnaire

Please go through the questions in this document
and try to answer them as carefully as possible

**Most of the questions relate to
the computer program you have just used.**

All information will be treated with the strictest confidence

On the following two sections are a list of questions about yourself and your familiarity with computers.

Please respond either by **ticking** the appropriate box on the right-hand side of the page or by **writing** the relevant answer on the dotted lines provided.

A. BIOGRAPHICAL DATA

1. Please indicate your name here...

2. Your University...

3. Your department or course...

4. What is your age (in years)?...

5. What is your present occupation? 5.

Undergraduate student	☐
Postgraduate student	☐

If other, please specify..

6. Please specify your gender. 6.

Male	☐
Female	☐

B. EXPERIENCE WITH COMPUTING

7. (a) Do you have a computer at HOME? 7 a.

Yes	☐
No	☐

(b) If YES, how much do you use it? 7 b.

Daily	☐
Weekly	☐
Monthly	☐
Rarely	☐
Never	☐

8. (a) Do you have access to a computer at COLLEGE/WORK? 8 a.

Yes	☐
No	☐

(b) If YES, how much do you use it? 8 b.

Daily	☐
Weekly	☐
Monthly	☐
Rarely	☐
Never	☐

9. On the whole, can you operate it without help of a user manual or other person? 9.

Not applicable	☐
Yes	☐
No	☐

10. How much do you enjoy using a computer? 10.

Not applicable	☐
Not at all	☐
Not much	☐
Unsure	☐
Quite a lot	☐
Very much	☐

11. How often did you use a computer at school? 11.

Daily	☐	
Weekly	☐	
Monthly	☐	
Rarely	☐	
Never	☐	

12.(a) Have you received any computer related education/training? 12 a.

Yes	☐	
No	☐	

(b) Which type(s) of education/training did you receive? 12 b.

	Not applicable	☐	
Education -	'O' level Computer Science	☐	'O'
	'A' level Computer Science	☐	'A'
	HND Computer Science	☐	HND
	BTEC Computer Science	☐	BTEC
	BSc Computer Science	☐	BSc
Software Packages -	Database	☐	Data
	Spreadsheet	☐	Spread
	Word processing	☐	Word

If other, please specify..

13. (a) Can you program in any computer languages? 13.

Yes	☐	
No	☐	

14. (a) Have you used any computer aided learning package before? 14 a.

Yes	☐	
No	☐	

(b) If yes, was it in business subjects? 14 b.

Not applicable	☐	
Yes	☐	
No	☐	

Please specify the topic(s) covered..

1	2	3	4	5
Strongly Agree	Agree	Uncertain	Disagree	Strongly Disagree

C. ASSESSMENT OF THE TUTORING SYSTEM

Given below are a number of statements about the Capital Investment Appraisal system, please indicate your response for each statement, **using the scale above**. For example, if you feel that you **strongly agree** about the statement fourteen, then write **1** in the box next to statement fourteen. However if you **strongly disagree** about statement number fourteen then write **5** in the box. There are no right or wrong answers, please give the response closest to your own feelings.

COMPUTING KNOWLEDGE

15. This program does not require any prior knowledge of computing. [ ] 15.

16. I couldn't learn anything of subject because I was spending all my time getting to know how to use a computer. [ ] 16.

17. After using this system I became more confident about learning from a computer. [ ] 17.

18. It was difficult enough to tackle the accountancy problems without having to fight with the computer. [ ] 18.

19. The system can be used sufficiently well without any manuals. [ ] 19.

20. Normally I find this type of subject very simple, but I couldn't see what the computer wanted me to do. [ ] 20.

COMPARISON WITH OTHER MEDIA

21. I would prefer to learn from a human tutor than from this system. [ ] 21.

22. It would have been a big improvement to have a tutor close by, so that I could ask any questions. [ ] 22.

23. I liked understanding the subject with this system, because it's like being on a one-to-one basis with your tutor. [ ] 23.

24. I was left with a lot of unanswered questions after using the system. [ ] 24.

25. This form of teaching was really clear, because unlike a lecturer, the program does not miss out a lot of steps. [ ] 25.

1	2	3	4	5
Strongly Agree	Agree	Uncertain	Disagree	Strongly Disagree

26. It is easier to get involved solving accounting problems using this system than in a classroom tutorial. [ ] 26.

27. I prefer an accountancy book because you can flip it backwards and forwards. [ ] 27.

28. It's much easier to understand the subject by this system than reading an accountancy book because this system is interactive. [ ] 28.

29. It was good to be able to discover the relationships between variables, which I never got from books. [ ] 29.

30. I would rather learn accountancy by working out problems using pencil and paper. [ ] 30.

31. I like doing accountancy this way because you can see how it's done, rather than asking somebody to explain it. [ ] 31.

INDIVIDUAL FLEXIBILITY

32. This system is only appropriate for advanced students of accounting. [ ] 32.

33. The computer got harder just as I got better at the problems. [ ] 33.

34. I knew how to get the results, but the system persisted in following its long and laborious steps. [ ] 34.

35. The system kept adjusting its advice according to my needs and progress. [ ] 35.

36. I couldn't jump into the more demanding material without having to go through the easier stuff again. [ ] 36.

ACCOUNTANCY KNOWLEDGE/TEACHING

37. I didn't need any previous knowledge of accounting to use the system. [ ] 37.

38. I have gained a good introduction to the concepts of Capital Investment Appraisal by using this system. [ ] 38.

39. I think the program is only useful for people who are already familiar with accountancy and want to improve. [ ] 39.

1	2	3	4	5
Strongly Agree	Agree	Uncertain	Disagree	Strongly Disagree

40. At the beginning I was not sure what Capital Investment Appraisal really is, but once I got the hang of it, I became quite an expert. ☐ 40.

41. It is difficult to learn the basics of Capital Investment Appraisal with this system. ☐ 41.

42. The system gives a first class grounding in the use of Capital Investment Appraisal information. ☐ 42.

43. Most people would find it hard to work out Capital Investment Appraisal problems using this system. ☐ 43.

44. The system allows you to fully test your understanding of Capital Investment Appraisal. ☐ 44.

45. At the end I still did not know anything about Capital Investment Appraisal. ☐ 45.

46. After using this system I can easily use my knowledge to work out Capital Investment Appraisal of a given project. ☐ 46.

47. I would find it easy to use this system to teach someone else about Capital Investment Appraisal. ☐ 47.

48. The computer has helped me by guiding me through the learning. ☐ 48.

49. I would like to learn other accounting topics using this system. ☐ 49.

50. The computer system allowed me to learn an accurate picture of Capital Investment Appraisal and corrected my mistakes. ☐ 50.

51. It allowed me to think about the relationships behind Capital Investment Appraisal. ☐ 51.

52. Even now, I do not understand the relationships between the different parts of Capital Investment Appraisal. ☐ 52.

ERROR MESSAGES AND DOCUMENTATION

53. The computer messages were good enough to help me when I got stuck. ☐ 53.

54. The error messages were unhelpful. ☐ 54.

1	2	3	4	5
Strongly Agree	Agree	Uncertain	Disagree	Strongly Disagree

55. I liked the program because it advised me what to do whenever I made any mistakes. ☐ 55.

56. The program should not allow you to enter any incorrect values. ☐ 56.

57. The program is so unforgiving, it doesn't allow you to do anything else, until you provide the correct answer. ☐ 57.

58. There should be an error message if I enter an incorrect negative or positive numbers. ☐ 58.

59. I like the way it told me precisely what I did wrong. ☐ 59.

60. Program directly told me what to do without giving me another chance. It would be better to have another chance with a small hint. ☐ 60.

EXPLANATION (FEEDBACK) INSTRUCTIONS

61. The system provides good feedback only when you have entered an answer. ☐ 61.

62. The system didn't give enough praise for the correct answer. ☐ 62.

63. The system helped me with every step I took in solving the problem. ☐ 63.

64. Through the feedback I immediately knew whether I had got it right or wrong. ☐ 64.

65. The explanations given by the program were hard to understand. ☐ 65.

66. The explanations showed clearly why I had got the wrong answer. ☐ 66.

67. The program gives the same type of feedback at all times regardless of what you have done. ☐ 67.

68. The program gives individual feedback to each student depending upon how far they have progressed to the solution. ☐ 68.

69. It would be better if a clearer explanation of the aim of the exercise and the required information to complete the exercise would have been given. ☐ 69.

1	2	3	4	5
Strongly Agree	Agree	Uncertain	Disagree	Strongly Disagree

ROUTE THROUGH PROGRAM

70. I never knew what all the items on various buttons meant. [ ] 70.

71. The instructions given on how to proceed through a tutorial were unclear. [ ] 71.

72. I like learning like this, because you can check over things and do not make the same mistakes again and again. [ ] 72.

73. Sometimes I found it hard to keep track which bits I had done. [ ] 73.

74. It allowed me to solve a problem in any order. [ ] 74.

75. The system is unable to recognise a different route to the solution. [ ] 75.

76. I would have found it more helpful to be given a suggested route through the program. [ ] 76.

77. The prompts, asking me to enter text, weren't clear from the screen. [ ] 77.

78. I want to be able to stop the exercise and go back to the tutorial, and then return where I left off. [ ] 78.

79. On-screen tutorials are difficult to follow. [ ] 79.

80. It provided little understanding of what you are doing or how to do it. [ ] 80.

CONTROL OF LEARNING

81. I felt the system allowed me to work at my own pace and direction. [ ] 81.

82. I found the system was too slow because I could do the exercises much more quickly with pencil and paper. [ ] 82.

83. It was too quick, I would like more control of how much time I was given to solve the problems. [ ] 83.

84. I wanted slower speed in the beginning and quicker when I got some experience, but system kept on at same speed which was annoying. [ ] 84.

1	2	3	4	5
Strongly Agree	Agree	Uncertain	Disagree	Strongly Disagree

85. I learnt a lot about Capital Investment Appraisal by trying different things with the system. [ ] 85.

RETENTION AND LEARNING

86. I am sure I will retain most of what I have learnt. [ ] 86.

87. I learnt most from this system, by trying to answer the various problems. [ ] 87.

88. If I use this program again, I could solve similar problems much quicker. [ ] 88.

89. I would not be able to solve problems if they will be presented in different format than this system. [ ] 89.

HELP FACILITY

90. I would prefer more help from the system. [ ] 90.

91. The help facility should have given a few hints, rather than the correct answer. [ ] 91.

92. Program does not give any help at many points where it is most wanted. [ ] 92.

93. Program gave a general help everytime instead of context sensitive help. [ ] 93.

FUNCTIONS OR KEYS

94. I found it very difficult to know which buttons/keys corresponded to the command I wanted. [ ] 94.

95. When proceeding through the program I found I was clear which options I was allowed to select. [ ] 95.

96. I hated some of the buttons/keys, I sometimes got lost because of them. [ ] 96.

97. When I was going through the program, pressing the same key would produce a different command. [ ] 97.

98. I got confused because same command was available through different keys in different screens. [ ] 98.

1	2	3	4	5
Strongly Agree	Agree	Uncertain	Disagree	Strongly Disagree

99. It was easy to exit from a particular route through the program. ☐ 99.

100. I couldn't skip some parts of the exercise. ☐ 100.

101. It was simple to delete any answer which I knew was wrong. ☐ 101.

102. I wasn't allowed to restart with the same data. ☐ 102.

CALCULATOR FACILITY

103. The calculator facility was easy to use. ☐ 103.

104. I found it hard to use the calculator to transfer the answer to the program. ☐ 104.

105. I had to type all the numbers by hand even when they were available on the screen. I could easily pick them by mouse but program didn't allow it. ☐ 105.

106. Using the calculator is frustrating - it doesn't do what it's supposed to do. ☐ 106.

107. Calculator should have some more functions which are needed in the exercise. ☐ 107.

PRESENTATION OF INFORMATION ON SCREEN

108. The screen layout meant that it was easy to communicate with the computer. ☐ 108.

109. Sometimes I was confused by the large amount of information on the screen. ☐ 109.

110. The way the problem is presented on the screen makes it easy to work out the solution. ☐ 110.

111. There was not enough information on the screen for me to solve the problems. ☐ 111.

112. The screen was attractive and clear. ☐ 112.

113. The colours on the screen were very confusing. ☐ 113.

1	2	3	4	5
Strongly Agree	Agree	Uncertain	Disagree	Strongly Disagree

114. The program looked very dull and boring. [ ] 114.

HOLISTIC SYSTEM

115. I did not like the system on the whole. [ ] 115.

116. I really liked using this system because it was fun to use. [ ] 116.

117. I enjoyed it but it could be improved a lot more, so I would be tempted to use it again. [ ] 117.

118. I simply used the computer to read what is on the screen. [ ] 118.

119. The program went on too long. [ ] 119.

120. It is easy to use and does not require much practice to be familier. [ ] 120.

121. I would be happy, if I could be given assignments using this program. [ ] 121.

122. The program did not manage to convince me that I could learn something by using it. [ ] 122.

123. I liked the way it helped me when I was in trouble. [ ] 123.

124. It is a good way to revise something I have forgotten. [ ] 124.

OVERALL PERCEPTION

125. Most students are not interested in using such a system. [ ] 125.

126. I solved the problems but I was unclear why the answer was correct. [ ] 126.

127. I would like to know what other possible ways may be to solve a particular problem. [ ] 127.

128. I would like to have more examples. [ ] 128.

PLEASE TURN OVER

129. Apart from any of the issues discussed previously,
what other features did you find contributed to the:-

(a) STRENGTH of this system ....

....

....

(b) WEAKNESS of this system ....

....

....

130. If there was **one** outstanding improvement you could recommend to this system, what would it be?

....

....

PLEASE CHECK YOU HAVE ANSWERED ALL THE QUESTIONS.

THANK YOU VERY MUCH FOR YOUR CO-OPERATION.

Appendix H

Mantel-Haenszel chi-square statistic on subjective questionnaire statements

The results presented below are from the phase II evaluation study. Phase I study results were found to be significantly different then phase II study results with the differences being attributed to the changes made in the package after phase I study (chapter 10). Significant p values in the following tables are shown in **bold** font for the three packages. The significant associations are then described. The parameters considered in this appendix are: gender, learning styles, computer training, confidence in operating computers, and enjoying computers.

1. Gender Analysis - p values

Questions	Capital Investment Appraisal	Absorption Costing	Marginal Costing
1. The software does not require any prior knowledge of computing.	0.243	0.869	0.435
2. The software compares well with other traditional medium of learning.	0.495	0.703	0.584
3. The software does not impose a rigid structure of learning.	0.504	0.592	0.692
4. The software gives good knowledge of the topic covered.	0.465	0.404	0.446
5. The software does not require any prior knowledge of accounting.	0.705	0.702	0.471
6. The software provides helpful and meaningful error messages.	0.923	0.985	0.574
7. The software provides helpful interactive guidance to individual students.	0.239	0.344	0.59
8. The software instructions are clear and easy to follow.	0.736	0.207	0.47
9. The software lets the students learn at their own pace.	0.283	0.15	0.135
10. The software enables good learning and retention of what is learnt.	0.992	0.306	0.277
11. The software has an adequate help facility.	0.592	0.131	0.939
12. The software is easy to follow and uses instinctive and unambiguous controls.	0.347	0.963	0.345

13. The software provides a useful support facility through its calculator.	0.068	0.23	**0.003***
14. The software lays out information in clear and meaningful way.	0.242	0.376	0.086
15. Overall perception of the user about this system is positive.	0.623	0.311	0.672

Table 14. p-values for gender analysis

The only significant p-value is for question 13 for Marginal Costing package, although the value for the same question for Capital Investment Appraisal package is also near the limit of 0.05. The students' opinions for Marginal Costing package for question 13 are as follows:

* *Question 13: The software provides a useful support facility through its calculator.*

	Disagree	Agree
Females	26.3	73.7
Males	50.7	49.3

This analysis shows that in general, there is no difference between the opinions of students of both genders, except with respect to the calculator, for which female students provided more supportive opinion than male students.

2. Learning styles analysis - p values

Questions	Capital Investment Appraisal	Absorption Costing	Marginal Costing
1. The software does not require any prior knowledge of computing.	0.661	0.772	0.892
2. The software compares well with other traditional medium of learning.	0.447	0.973	0.978
3. The software does not impose a rigid structure of learning.	0.946	0.944	0.912
4. The software gives good knowledge of the topic covered.	0.971	0.979	0.917

5. The software does not require any prior knowledge of accounting.	0.996	0.773	0.403
6. The software provides helpful and meaningful error messages.	0.97	0.966	0.74
7. The software provides helpful interactive guidance to individual students.	0.619	0.86	0.976
8. The software instructions are clear and easy to follow.	0.824	0.862	0.806
9. The software lets the students learn at their own pace.	0.984	0.997	0.67
10. The software enables good learning and retention of what is learnt.	0.972	0.761	0.953
11. The software has an adequate help facility.	0.66	0.705	0.802
12. The software is easy to follow and uses instinctive and unambiguous controls.	0.998	0.703	0.68
13. The software provides a useful support facility through its calculator.	0.947	0.234	0.452
14. The software lays out information in clear and meaningful way.	0.636	0.906	0.635
15. Overall perception of the user about this system is positive.	0.697	0.756	0.992

Table 15. p-values for learning style analysis

There is no significant p-value for any of the packages which shows that the views of students of all learning styles are similar.

3. Computer training analysis - p values

Questions	Capital Investment Appraisal	Absorption Costing	Marginal Costing
1. The software does not require any prior knowledge of computing.	0.279	0.912	0.523
2. The software compares well with other traditional medium of learning.	0.258	0.708	0.821
3. The software does not impose a rigid structure of learning.	0.605	0.873	0.703

4. The software gives good knowledge of the topic covered.	0.23	0.622	0.977
5. The software does not require any prior knowledge of accounting.	**0.046***	0.396	0.458
6. The software provides helpful and meaningful error messages.	0.938	0.975	0.947
7. The software provides helpful interactive guidance to individual students.	0.536	0.468	0.31
8. The software instructions are clear and easy to follow.	0.142	0.6	0.696
9. The software lets the students learn at their own pace.	0.239	0.794	0.955
10. The software enables good learning and retention of what is learnt.	0.212	0.761	0.887
11. The software has an adequate help facility.	0.065	0.399	0.573
12. The software is easy to follow and uses instinctive and unambiguous controls.	0.737	0.86	0.704
13. The software provides a useful support facility through its calculator.	0.287	0.481	0.064
14. The software lays out information in clear and meaningful way.	0.376	0.6	0.613
15. Overall perception of the user about this system is positive.	0.214	0.734	0.673

Table 16. p-values for computer training analysis

The only significant p-value is for question 5 for Capital Investment Appraisal package, although the value question 11 for Capital Investment Appraisal package is also near the limit of 0.05. The students' opinions for Capital Investment Appraisal package for question 5 are as follows:

* *Question 5. The software does not require any prior knowledge of accounting.*

	Disagree	Agree
Students with no computer training	54.7	45.3
Students with computer training	37.6	62.4

The analysis shows that in general, there is no difference in opinions among the students with or without computer training. The only significant difference, which exist, is towards the requirement of prior subject knowledge. Students without computer training were less supportive with respect to whether the package does not require knowledge of accounting.

4. Confidence in operating computers analysis

Questions	Capital Investment Appraisal	Absorption Costing	Marginal Costing
1. The software does not require any prior knowledge of computing.	0.731	0.763	0.9
2. The software compares well with other traditional medium of learning.	0.458	0.56	0.414
3. The software does not impose a rigid structure of learning.	0.424	0.352	0.747
4. The software gives good knowledge of the topic covered.	**0.034***	0.156	0.47
5. The software does not require any prior knowledge of accounting.	0.324	0.826	0.572
6. The software provides helpful and meaningful error messages.	0.368	0.58	0.748
7. The software provides helpful interactive guidance to individual students.	0.364	0.834	0.152
8. The software instructions are clear and easy to follow.	0.156	0.597	0.177
9. The software lets the students learn at their own pace.	0.128	0.464	0.208
10. The software enables good learning and retention of what is learnt.	**0.026***	0.411	0.765
11. The software has an adequate help facility.	**0.045***	0.801	0.671
12. The software is easy to follow and uses instinctive and unambiguous controls.	0.808	0.879	0.994
13. The software provides a useful support facility through its calculator.	**0.019***	0.515	0.492
14. The software lays out information in clear and meaningful way.	0.702	0.791	0.919
15. Overall perception of the user about this system is positive.	0.469	0.797	0.784

Table 17. p-values for confidence in operating computers analysis

The p-value is significant for four questions for Capital Investment Appraisal package, question 4, 10, 11 and 13. The students' opinions for Capital Investment Appraisal package for these questions are as follows:

* *Question 4: The software gives good knowledge of the topic covered.*

	Disagree	Agree
Students with no confidence in operating computers	60.7	39.3
Students with confidence in operating computers	35.8	64.2

* *Question 10: The software enables good learning and retention of what is learnt.*

	Disagree	Agree
Students with no confidence in operating computers	64.3	35.7
Students with confidence in operating computers	40.7	59.3

* *Question 11: The software has an adequate help facility.*

	Disagree	Agree
Students with no confidence in operating computers	60.8	39.2
Students with confidence in operating computers	40	60

* *Question 13: The software provides a useful support facility through its calculator.*

	Disagree	Agree
Students with no confidence in operating computers	68	32
Students with confidence in operating computers	43	57

The above analysis shows that for all significant differences, students with no confidence in operating computers were less supportive of the tutoring systems.

5. Enjoying computers analysis

Questions	Capital Investment Appraisal	Absorption Costing	Marginal Costing
1. The software does not require any prior knowledge of computing.	0.489	0.856	0.973
2. The software compares well with other traditional medium of learning.	0.216	0.335	**0.043***
3. The software does not impose a rigid structure of learning.	0.757	0.691	0.894
4. The software gives good knowledge of the topic covered.	0.076	0.112	**0.036***
5. The software does not require any prior knowledge of accounting.	0.276	0.538	0.509
6. The software provides helpful and meaningful error messages.	0.269	0.594	0.954
7. The software provides helpful interactive guidance to individual students.	0.554	0.541	0.939
8. The software instructions are clear and easy to follow.	0.278	0.644	0.121
9. The software lets the students learn at their own pace.	0.438	0.569	0.637
10. The software enables good learning and retention of what is learnt.	0.115	0.988	0.999
11. The software has an adequate help facility.	**0.031***	0.134	**0.038***
12. The software is easy to follow and uses instinctive and unambiguous controls.	0.952	0.727	0.82
13. The software provides a useful support facility through its calculator.	0.808	0.939	0.441
14. The software lays out information in clear and meaningful way.	0.425	0.815	0.912
15. Overall perception of the user about this system is positive.	0.253	0.937	0.601

Table 18. p-values for enjoying computers analysis

The p-value is significant for question 11 for Capital Investment Appraisal package and for questions 2, 4 and 11 for Marginal Costing package. The value of question 4 for Capital Investment Appraisal is also near the limit of 0.05. The students' opinions for the significant p-value questions are as follows:

Capital Investment Appraisal

* *Question 11: The software has an adequate help facility.*

	Disagree	Agree
Students not enjoying computers	54.1	45.9
Students enjoying computers	37.5	62.5

Marginal Costing

* *Question 2: The software compares well with other traditional medium of learning.*

	Disagree	Agree
Students not enjoying computers	65.9	34.1
Students enjoying computers	49.1	50.9

* *Question 4: The software gives good knowledge of the topic covered.*

	Disagree	Agree
Students not enjoying computers	53.1	46.9
Students enjoying computers	35.6	64.4

* *Question 11: The software has an adequate help facility.*

	Disagree	Agree
Students not enjoying computers	55.3	44.7
Students enjoying computers	36.6	63.4

All of the above analysis shows that the students who do not enjoy computers are less supportive.

Discussion on significant associations

Various questions with significant associations for at least one parameter are shown below:

Capital Investment Appraisal

Significant associations are shown in **bold** font and the opinions having more disagreement than agreement are shown in *italic* font.

Questions	Gender				Learning style										Computer training				Confidence in operating computers				Enjoying computers			
	Female		Male		Active		Reflect		Theory		Prag.		Mixed		No		Yes		No		Yes		No		Yes	
	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.
4. The software gives good knowledge of the topic covered.	38.2	61.8	42.6	57.4	35.6	64.4	46.2	53.8	38.7	61.3	38.7	61.3	36.9	63.1	47.1	52.9	36.7	63.3	***60.7***	***39.3***	**35.8**	**64.2**	49.2	50.8	34.9	65.1
5. The software does not require any prior knowledge of accounting.	45.6	54.4	41.5	58.5	35.6	64.4	42.8	57.2	44.1	55.9	44.1	55.9	41.7	58.3	***54.7***	***45.3***	**37.6**	**62.4**	*53.8*	*46.2*	41.8	58.2	49.2	50.8	39.8	60.2
10. The software enables good learning and retention of what is learnt.	44.9	55.1	44.7	55.3	36	64	46.5	53.5	39.1	60.9	44.5	55.5	43.6	56.4	*51.8*	*48.2*	41.3	58.7	***64.3***	***35.7***	**40.7**	**59.3**	*52.8*	*47.5*	40.4	59.6
11. The software has an adequate help facility.	40.6	59.4	44.7	55.3	36	64	46.5	53.5	44.5	55.5	*55.5*	*44.5*	38.8	61.2	*53.8*	*46.2*	38.6	61.4	***60.8***	***39.2***	**40**	**60**	***54.1***	***45.9***	**37.5**	**62.5**
13. The software provides a useful support facility through its calculator.	39.2	60.8	*53.2*	*46.8*	43	57	50	50	44.5	55.5	*55.5*	*44.5*	45.9	54.1	*53.8*	*46.2*	44	56	***68***	***32***	**43**	**57**	47.5	52.5	46.1	53.9

Table 19. Students' views towards Capital Investment Appraisal package

The above analysis clearly shows that the students with no computer training, no confidence in operating computers and not liking of computers are less supportive. Some male students and students with pragmatist learning style also have similar attitude.

Absorption Costing

No significant associations have been observed for any of the questions for Absorption Costing package.

Marginal Costing

Significant associations are shown in **bold** font and the opinions having more disagreement than agreement are shown in *italic* font.

Questions	Gender				Learning style										Computer training				Confidence in operating computers				Enjoying computers			
	Female		Male		Active		Reflect		Theory		Prag.		Mixed		No		Yes		No		Yes		No		Yes	
	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.	Disag	Agr.
2. The software compares well with other traditional medium of learning.	*57.5*	*42.5*	*53.5*	*46.5*	50	50	*54.1*	*45.9*	*53.5*	*46.5*	*58.9*	*41.1*	*55.3*	*44.7*	*57.2*	*42.8*	*56*	*44*	*63.2*	*36.8*	*54.6*	*45.4*	***65.9***	***34.1***	**49.1**	**50.9**
4. The software gives good knowledge of the topic covered.	43.8	56.2	39.4	60.6	33.6	66.4	38.4	61.6	38.4	61.6	34.4	65.6	41.5	58.5	40.5	59.5	41.3	58.7	47.6	52.4	39.4	60.6	***53.2***	***46.8***	**35.6**	**64.4**
11. The software has an adequate help facility.	42.5	57.5	42.3	57.7	38.7	61.3	42.4	57.6	46.5	53.5	31.3	68.8	43	57	38.1	61.9	43.1	56.9	36.7	63.3	43.1	56.9	***55.3***	***44.7***	**36.6**	**63.4**
13. The software provides a useful support facility through its calculator.	**26.3**	**73.7**	***50.7***	***49.3***	38.8	61.2	34.7	65.3	*58*	*42*	32.3	67.7	37.5	62.5	50	50	33	67	47.3	52.7	36.5	63.5	43.4	56.6	35.3	64.7

Table 20. Students' views towards Marginal Costing package

For Marginal Costing package, almost all students did not support the statement that the software compares well with other traditional medium of learning. Students who do not enjoy computers also were of the opinion that the software does not provide good knowledge of topic and does not have adequate help facilities. Some male students and some students with theorist learning style did not like calculator facility.

Appendix I

Open-ended Questions in Subjective Questionnaires

Students commented on many strengths and limitations of the packages and provided many useful recommendations in the open ended question of the subjective questionnaire. This appendix gives a summary of most commonly sighted comments:

Capital Investment Appraisal

Strengths

1. The software provides good interaction with the user. It offers correction and help facilities where needed.
2. The package was both easy to use and understandable, it made the subject easier to learn and generally appeared to be an attractive facility to use.
3. During tutorial, the system told you exactly what mistake you made.
4. Easy to understand and use.
5. Facility to go back and repeat things you are unsure about.
6. Layout of numbers easy to follow and a good screen.

Limitations

1. The calculator seemed slow to use in comparison with personal calculators, there were also some general difficulties in using the facility regarding the keys, and more complicated calculations such as percentages or negative numbers.
2. The colours were deemed unattractive and boring.
3. There appeared to be no save button on the screen (the close button seems as though you are exiting the system and thus, students did not want to use it before saving the work.)
4. Couldn't understand how to save your work in the middle of a question.
5. Reloading work once you had started was not easy.
6. More explanations required when mistakes have been made.
7. More difficult examples required.

8. Very boring and it is easy to complete the assignment without actually knowing or learning what Capital Investment Appraisal actually is. Therefore it encourages the reader not to do the tutorials.

Improvements

1. Different colours should be used to draw your attention to different icons or other areas.
2. Choice should be given to choose your own colour schemes.
3. A positive sound should be provided when a correct answer is entered, rather than just a negative sound when a wrong answer is entered.
4. A 'save' button should be provided on the screen which should be visible all the time.
5. 'Close' procedure should be amended to make it simpler.
6. The subject manual should have workbook type structure, so that examples could be practised on it.
7. The system should have more facts about Capital Investment Appraisal in the tutorials rather then just telling you when you are wrong.

Absorption Costing

Strengths

1. Quick and easy to use.
2. A good variation in the types of examples provided.
3. Feedback was improving with each mistake, so that you have more chances to get the answer right from you own brain.
4. Software very user-friendly.
5. A good help facility available while solving the problems and an on-screen calculator.

Limitations

1. Tended to crash for some of students.
2. No personal tutoring and help, you cannot return to the previously saved examples to make amendments.
3. Tutors don't care. They are not present at all.
4. If the student details are filled wrongly by mistake for first example, system does not allow to correct it in all examples.
5. The error messages were not very helpful.
6. It took too long to finish the assignment.

Improvements

1. The feature to check previously completed examples to make sure that there are not any errors.
2. Tutorial lectures to be available if there was something you didn't understand about the system.
3. Better calculator facility.
4. More exercises should be available using different data to test you knowledge more thoroughly.
5. More appreciation from system, when the answer is right.

Marginal Costing

Strengths

1. Good clear day to day English in the text - much easier to understand.
2. Far more user friendly.
3. The human computer interface was excellent.
4. Clear instructions were available on where to fill the data.
5. Good systematic approach which made it seem much easier.
6. System provides the formula which could be used in that particular situation.
7. Infinity sign was quite helpful. It saved from lot of problems.

Limitations

1. The calculator screen was very small.
2. Screen colours are poor. No consideration towards HCI.
3. Error messages were not helpful.
4. Same type of example again and again. Very boring.

Improvements

1. The calculator should allow the user to divide figures.
2. Colours could be improved.
3. On-line help should have more explanations.
4. System should be made less monotonous.
5. Screen colours should be toned down to less bright ones.

Appendix J

Summary of Observations taken during Analysis

Students were observed by the author and respective tutors and technicians at various institutions. Besides comments from observers, comments from students were also recorded. Summary of the most common observations is as follows:

Students' comments

Capital Investment Appraisal

1. "A very good user friendly package. Perhaps too spoon-fed at times. The data input became a matter of process without necessarily having a full understanding of the concepts which underlay them."
2. Because the program is so progressive it is doubtful whether one would understand it in a real life situation. Once you have done a few examples the input becomes like a machine, not thinking about the concept or trying to understanding it. It does make it easy to understanding though."
3. "Would be useful to flip back and forth to the tutorial pages."
4. "The formula for discount factor is presented as $\frac{1}{(1+i)}n$. It is not clear initially whether this is (1+i) to the power of n (which it is) or $\frac{1}{(1+i)}$ times n."
5. When entering the value into boxes from calculator (e.g. ARR), percentages are not rounded e.g. 7.29% is entered as 7.2%."
6. "Calculator interface is very uncomfortable."
7. "I am not sure if I have saved the example correctly. Is there any way I could find if the example is really on the disk?"
8. "Tutorials were good. I like the way screens were shown in the tutorials. I did not find it difficult to work on problem screen afterwards."

Absorption Costing

1. "The problem with the system is that you do need a tutor to clarify points."
2. "It is better than just an accountancy book, but is not a replacement for a tutor."
3. "The feedback is good."

4. "Repetitive questions - boring."
5. "I should be able to amend my work."
6. "I could not use enter your own feature. It was disabled."
7. "Using calculator is frustrating. It goes off every time you upload a value."
8. "Difficult to save, you have to exit example to save and go to the next."

Marginal Costing

1. "I am suspicious that the exam questions will be presented differently thus confusing as the software only gave one style."
2. "Very good method of learning, but the problems with number rounding meant that the computer wouldn't accept the right answers."
3. "MCC assignment is too close to exam - limited amount of computer time. MCC assignment should be a small part of our examination."
4. "It got a little boring after a while, lost concentration. The set topic and the question that followed were good."
5. "MCC was not difficult to work with and I found no problems with example questions once I properly understood the example question and the information at the beginning, which explains the concept of marginal costing was comprehensible but did take time to go through."
6. "I thought it was a different type of learning, something I had never done before. A worthwhile exercise."
7. "Once you reach a certain point all the rest is same."
8. "MCC was a good package, but tutorials were not as helpful. I had to understand the concept from the handbook before doing it. Other than that there were lots of useful examples."
9. "Too simplified. Saving and handling problem. User interface ridiculous - written for 5 year olds."
10. "The program was a very good way of learning the topic /material."

11. "The main problem was having access to computer, not having one to use at home. I found that it was problem removing people who were not supposed to be there."

Observations on all three packages

1. Students generally had no problems in using software, they use it even after the booked sessions, without any difficulty.
2. Saving of the questions was a bit troublesome. Students were hesitant to use close button without saving their questions first, although written instructions were issued to them regarding this.
3. Some students were confused by the rounding problems, specially with the values transferred from the calculator.
4. Many students were disappointed and asked why Internal Rate of Return screen is not available in assignment mode of Capital Investment Appraisal package, when they have already done so much practice on it in interactive mode.
5. None of the student had any trouble finding their way in the programs. The navigation was quite intuitive.
6. Students were not sure if they have saved the examples correctly on the floppy. They were not much worried about partially saved examples, but they wanted to be sure about examples saved for marking.

Appendix K

Students Analysis for Training, Confidence and Enjoyment in Computer Operations Attributes

The analysis of subjective views of student-users towards the packages showed that the students who did not have any previous computer training, did not have confidence in operating computers or did not enjoy computers provided less supportive attitude towards CAL teaching.

The analysis was further extended to find out the performance between the students with and without these attributes.

1. Analysis of students with and without computer training

Tables 21, 22 and 23 show the mean gains of the students with and without any previous computer training.

Capital Investment Appraisal

Universities	*Students without computer training*	*Students with computer training*	*Total mean gain*
University A	66.0	67.1	66.8
University B	69.3	64.7	66.1
University E	69.4	65.7	67.1
University F	68.4	67.8	68.0
Total mean gain	68.5	66.3	

Table 21. Gains of students with and without computer training for Capital Investment Appraisal package

Absorption Costing

Universities	*Students without computer training*	*Students with computer training*	*Total mean gain*
University A	62.3	64.6	64.2
University D	65.8	70.7	69.7
University E	64.4	66.3	65.6
Total mean gain	64.3	67.2	

Table 22. Gains of students with and without computer training for Absorption Costing package

Marginal Costing

Universities	*Students without computer training*	*Students with computer training*	*Total mean gain*
University A	81.3	81.1	81.1
University B	84.8	84.7	84.7
University C	68.1	69.8	69.3
University D	79.5	80.0	79.8
Total mean gain	78.0	78.9	

Table 23. Gains of students with and without computer training for Marginal Costing package

The ANOVA analysis was applied on the percentage gains of the students and the p-values obtained for Capital Investment Appraisal, Absorption Costing and Marginal Costing were 0.1992, 0.0987 and 0.6760 respectively. This shows that there is no significant difference between the performance of the students with and without computer training for any of the packages.

2. Analysis of students with and without confidence in operating computers

Tables 24, 25 and 26 show the mean gains of the students with and without confidence in operating computers.

Capital Investment Appraisal

Universities	*Students without confidence in operating computers*	*Students with confidence in operating computers*	*Total mean gain*
University A	65.4	67.2	66.9
University B	70.0	65.5	66.1
University E	70.4	65.8	67.1
University F	53.1	69.6	68.0
Total mean gain	66.6	67.1	

Table 24. Gains of students with and without confidence in operating computers for Capital Investment Appraisal package

Absorption Costing

Universities	*Students without confidence in operating computers*	*Students with confidence in operating computers*	*Total mean gain*
University A	59.7	65.0	64.2
University D	60.1	70.8	69.7
University E	69.3	64.9	65.6
Total mean gain	63.4	66.9	

Table 25. Gains of students with and without confidence in operating computers for Absorption Costing package

Marginal Costing

Universities	*Students without confidence in operating computers*	*Students with confidence in operating computers*	*Total mean gain*
University A	83.3	80.8	81.1
University B	83.3	84.9	84.6
University C	66.7	69.4	69.3
University D	78.6	80.1	79.8
Total mean gain	80.2	78.4	

Table 26. Gains of students with and without confidence in operating computers for Marginal Costing package

The ANOVA analysis was applied on the percentage gains of the students and the p-values obtained for Capital Investment Appraisal, Absorption Costing and Marginal Costing were 0.7752, 0.1012 and 0.5219 respectively. This shows that there is no significant difference between the performance of the students with and without confidence in operating computers for any of the packages.

3. Analysis of students with and without enjoyment in computers

Tables 27, 28 and 29 show the mean gains of the students with and without enjoyment in computers.

Capital Investment Appraisal

Universities	*Students without enjoyment in computers*	*Students with enjoyment in computers*	*Total mean gain*
University A	64.5	67.5	66.8
University B	65.3	67.1	66.1
University E	69.7	65.7	67.2
University F	67.9	68.1	68.0
Total mean gain	66.9	67.1	

Table 27. Gains of students with and without enjoyment in computers for Capital Investment Appraisal package

Absorption Costing

Universities	*Students without enjoyment in computers*	*Students with enjoyment in computers*	*Total mean gain*
University A	62.2	64.7	64.2
University D	70.8	69.4	69.6
University E	66.0	65.3	65.6
Total mean gain	65.9	66.6	

Table 28. Gains of students with and without enjoyment in computers for Absorption Costing package

Marginal Costing

Universities	*Students without enjoyment in computers*	*Students with enjoyment in computers*	*Total mean gain*
University A	83.3	80.4	81.2
University B	81.5	85.6	84.6
University C	68.9	69.4	69.2
University D	82.2	78.3	79.8
Total mean gain	78.2	78.8	

Table 29. Gains of students with and without enjoyment in computers for Marginal Costing package

The ANOVA analysis was applied on the percentage gains of the students and the p-values obtained for Capital Investment Appraisal, Absorption Costing and Marginal Costing were 0.8507, 0.6888 and 0.7586 respectively. This shows that there is no significant difference between the performance of the students with and without enjoyment in computers for any of the packages.