COMPUTER AIDED ARCHITECTURAL EVALUATION AND DESIGN
- A COST MODELLING EXPERIMENT.
(Volume I)

by
Tami A. BELHADJ

Thesis submitted to the University of Sheffield for the Degree of Doctor of Philosophy in the Faculty of Architectrural Studies.

Department of Architecture.
July 1989

Thesis title : COMPUTER AIDED ARCHITECTURAL EVALUATION AND DESIGN
- A COST MODELLING EXPERIMENT.

Abstract

Tami A. BELHADJ

This dissertation addresses the problem of Computer Aided Architectural Evaluation and Design, with particular reference to building cost evaluation and cost modelling. It identifies the lack of building evaluation tools in the CAAD environment, and suggests an integrated approach to building modelling and cost modelling. The interaction of **elemental** and **spatial** descriptions of a design solution is considered as an original contribution to the field of computer aided building modelling and evaluation. It demonstrates the potential of CAAD and Bills of Quantities intergration to give an extra dimension to cost modelling at early design stages.

Essentially, this reseach project advocates a larger overlap in the use of computers for the generation and evaluation of design. It asserts that any computer aided solution evaluation system must be able to converse with the designer during the highly integrated iteration of briefing, analysis, synthesis and evaluation stages of design. A working example is produced after software specification and implementation, to demonstrate to possibilites and/or limitations of such an approach.

To my Parents

Acknowledgements

My sincere gratitude is due to Pr B. R. Lawson, for his invaluable assistance during the corse of this research. He made extensive, precise and valuable criticisms throughout this project, and I have done my best to act upon them. Much that is in this thesis has been learned from him, though its interpretation and form is entirely my own responsibility.
I also most appreciate the financial support provided by the Algerian Ministry of Higher Education, without which it would have been difficult for me to stay abroad and carry out this research.

My colleagues, Joongseok Ryu, Barrie Bowden and Abu Hasan Ismail, have considerably helped this project, by discussing many of its ideas. I also wish to express my thanks to the software development team at GABLE CAAD system Ltd., as well as to the support staff of the Department of Architecture and Melvyn Broady in particular.

Finally, I would like to thank my family and friends for their endless support end encouragements.

TABLE OF CONTENTS

VOLUME I

VOLUME II

Chapter 1 : INTRODUCTION.

> "Computers have come to stay; they are changing the world whether we like it or not, and gradually they will find their way into the offices of architects and the schools of architecture all over the world."(p 1).
>
> Sudbo, B. (1988)

This quote was extracted from a report of an international forum held in Zurich in 1987, on 'architectural education and the information explosion'. It sets the background of the general feeling concerning computers 'overture' in architectural practice and education. There are several reasons to support such a view. Some are related to developments in computers technology, whereas others are associated with our understanding of the computer's role in design.

Computing technology has achieved in the last two decades a tremendous advance; for example processing speed and circuit density have increased by order of magnitude. The software field has also progressed considerably, with new software development tools, programming languages and methodologies. This new powerful computing environment is packaged and made available to individual users in the form of 'Personal Computers', and to engineers or designers in the form of the new generation of 'Graphical Workstations'.

During the same period, continuous research and development in the computer aided design field, has enabled to harness this computer power, and provide designers with new tools. Computer based design tools offer significant advantages over traditional design practice. In fact, they allow to perform design operations in ways that have never been possible before.

The broader concern of this thesis is to make the best possible use of computers in architectural design. Not just by accelerating the processing of design information, but by amplifying the intellectual abilities of the designer. The process of architectural design since its first inception has aimed at the production of 'good' buildings. The process itself, the teaching of design, and the evaluation methods and criteria, are subject to change and depend on the state-of-the-art in research and technology. Abstractions, in Schmitt's (1987) terms, support both the generative and the analytical phases of design. The concept of abstraction (or modelling) in architectural design is historically linked to the level of complexity of the design product. It involves the representation of existing structures and can be described as an attempt to recreate reality. Developments in the field of computer modelling, computer graphics, and more recently in cognitive psychology and artificial intelligence provide the theoretical basis to build fundamentally new tools to support the architectural design process, in particular for design abstraction and evaluation. The first generation design tools that have emerged from the early computer technology development have mainly concentrated on model building for drawing production, with the exception of few design evaluation systems. These were underpining the graphic characteristics of traditional architectural representation, where generally the abstraction hides most of the meaning of the drawing -not to the designer, but to the computer. It is the ability of a computer aided design system to 'understand' the implicit information contained in a drawing, that gives it the 'competence' to support 'intelligent' modelling and evaluation.

Among the large number of computer aided design tolls available to designers, very few offer support for the **evaluation** of design. Nearly all will assist in model visualisation for aesthetic examination, but there capabilities for computer aided building evaluation remain very limited.

Whilst architects can now produce many views and drawings of their designs, they can still do relatively little about evaluating the structural, functional, environmental, and financial performances of their buildings. It is the objective of this research to understand the reasons for such a situation, and explain it. In the process of doing so, the review of computer aided design developments is undertaken, and the theory behind it investigated. The lack of building evaluation tools in the CAAD environment is identified, and an attempt is made to explain as well as suggest solutions for it.

As an example for the computerisation of building performance evaluation, **cost modelling** was selected among different building design parameters. After reviewing the historical developments of quantity surveyors use of computers, this research project suggests an original approach to the integration of CAAD and cost modelling. The **elemental** and **spatial** description of a building computer model are combined to generate building quantities, which are subsequently used for cost analysis and cost modelling.

The computer processing speed and the nature of the data processed, have allow the performance of building cost evaluation in such a way that a larger overlap may be expected between the synthesis and the evaluation stages of design. From a design methodology point of view this last matter can be considered as a significant contribution to the computer closer 'involvement' in the design process.

As far as the organisation of the thesis is concerned, it is physically divided into two volumes; the first describes in general the work undertaken and demonstrates it in use, whereas the second volume consists of a listing of the program source code written during this research. In this first volume the introduction describes the context in which the research project was undertaken. It also defines which particular area, of the increasingly growing field of Computer Aided Architectural Design, is examined.

Volume one is presented in four main parts; first of all, Part A consists of a review of the theoretical background relevant to the understanding of the current situation of CAAD research and development. For instance, Chapter 2 gives an overview of design theory and design methodology, with particular emphasis on events sequence. It also reviews psychological work on human thinking and its relevance to problem solving in design. On the other hand, Chapter 3 outlines the historical development of computer technology, and gives some details on the basic computing techniques involved on computer aided design. Chapter 4 deals with the general issue of using computers in architectural design, and addresses the question of computer/designer interaction.

The next section is Part B, which consists of defining the problem, or in other words describing the aims and objectives of the study. In its Chapter 5, the importance of the evaluation stage of design is underlined, as well as its centrality in the design process. This is done using Markus's model of building modelling. Building cost evaluation is used as an example to support the experimental study. In the same section Chapter 6 introduces the notion of cost modelling in a CAAD environment. A brief historical review is done of the use of computers in cost estimating and cost modelling, and the suggested approach for the following software development is outlined.

The main section of the thesis is Part C. It describes the software application development involved in putting into practice the suggested approach to integrated Cost Modelling and Computer Aided Architectural Design. In Chapter 7, GABLE (Graphical Aids for Building Layout and Evaluation) system is described, and its building modelling system is explained using a simple architectural model. A cost modelling experiment is done on that building model, to illustrate the capabilities of the application.

Finally, Part D discusses very broadly the implications of the latest developments in CAAD, and speculates on the potential effects these might have on architectural design. The requirement for a system to link graphics, words and numbers is explained, and its relevance justified.

The second Volume contains the appendices and program listings.

PART A :

THEORETICAL BACKGROUNG.

Chapter 2 : DESIGN OVERVIEW.

2.1 DESIGN IN ARCHITECTURE.

> " We could investigate the architect's contribution to architecture in several ways-by looking at what they have designed...,or by looking at architect's themselves..."(p 1).
>
> Broadbent, G (1973)

The word architecture can be directly associated with two things. Firstly the product, or in other words the building, produced from the work of the architect, and secondly the process of producing such architecture. It is very difficult to envisage the study of architecture dissociating the product from the process. Historians often have the crude task of criticising buildings with little knowledge of the conditions in which they were designed. Similarly, theorists tend to discuss design processes without having experienced the use of buildings designed in that manner.

2.1.1 ARCHITECTURAL DESIGN.

Heath (1984) asks the question "What is architecture ?", and his answer inevitably refers to both the activity of designing and the artifact. The close relationship between product and process appears in early forms of architecture, where the same person or group of people conceived, realised and often used a building. Attempts were made to arrive at an 'anthropology' of architecture, Broadbent (1973) has enumerated four distinct ways of generating three-dimensional form that he describes as Pragmatic, Iconic, Analogic and Canonic, in chronological order of application. He argues that this chronology implies an increasing sophistication, with pragmatic design as the most primitive way of designing and canonic as the most intellectual. A good example of the 'increasing sophistication' referred to by Broadbent, is the 15th century Brunelleschi cathedral in Florence (Prager, F..D. & Scaglia, G. 1970, Mainstone, R. 1977), where the design of the building and especially the dome were so innovative in terms of construction techniques, that no builder could undertake the job merely relying on his own skills. Consequently, the builders had to obey the architects instructions to construct the sophisticated parts of the 'cupola'. This was, I believe, the start of the separation of designing from making. It is worth noticing here that Brunelleschi's ingenuity was helped by the use of scaled models at design stage, we will come back to this point later in section 3.2.1..

The separation of designing from making was accentuated in western industrial society, where the mass production of housing and other elements of the built environment have increased the remoteness of architects from builders and/or users (Lawson 1980). The teaching of architecture has itself reinforced such separation, in fact, the history of architectural education shows an increasing demand for abstraction and 'rationalism'. Bruno Zevi (1964) identifies three methods employed in the education of architects.

The first called the 'bottega' method (which means 'office' in Italian), where a master is selected and the teaching is done by working in his office. This teaching method still perhaps exist in some schools where there are few students and a great personality among the teachers, but the problem of mass education does not allow this any more.

The second educational system is the Beaux-Arts, where the design teaching is directed towards a 'style', probably many schools are still-run with this method.

The last system of teaching architecture suggested by Bruno Zevi is the 'historical' -that is to say, scientific- method , which he sees as a Bauhaus type of school with historical consciousness. Arguing this he says:

> " ...the idea that art is something purely intuitive, irrational, something that has to do only with feelings, is outdated. Art is a conscious act, a process which can be controlled and verified throughout."(p 15).

He adds :

> " ..., we should recognize that in the very few, exceptional works of art that are creative, there is a process that we can grasp and demonstrate and verify, just behind the lyrical or poetic aspect which appears irrational."(p 16).

This type of statement is bound to lead us into philosophical discussions on the difference between feeling and reason or art and science, which is not really the object of this thesis. However, Bruno Zevi's point of view is clear, and if creativity is purely a private process going on in the inaccessible recesses of the mind, then it would be purely subjective, and consequently, impossible to assess and educate (Best, D. 1985).

In conclusion, it is clear that architects are urged to be more explicit about their design process. We have seen that the study of architecture involves the combination of design process and design product, and that the growing complexity of the man-made world has put designers away from builders . Also that the education of the design itself is evolving, with more and more attention given to the design process. The next section will investigate the design methods and strategies of the generally accepted design theories.

2.1.2 THEORY OF DESIGN AND METHODOLOGY.

> "...designing should not be confused with art, with science, or with mathematics. It is a hybrid activity which depends, for its successful execution, upon a

> proper blending of all three."(p 10).
>
> Jones,J.C.(1970)

Architects have always been associated with artists because of their reliance on drawings to represent designs. Another reason for relating architecture to the arts is the sharing of intuitive design methods. Until recently architects relied almost exclusively on a 'philosophy' of design based on intuition. This consists of giving mystical explanations to design. Jones (1970) described it as the "Black-box" method, where intuition is the only guide through the design process. This of course promotes individualism, which means it will probably persist in practice.

Certainly, there have been advocates for 'rational' design. In fact, since the early nineteenth century, studies for classification of 'styles' were undertaken to discover the principles by which to design architecture (Viollet-le-Duc 1863). However, the theory of design had to wait for new scientific developments made during the 1950's in the fields of cybernetics (Gerardin, L. 1968) and information theory (Shannon, C.E. and Weaver, W. 1949) to put forward systematic design methods. Rittel (1972), an early operation researcher, argues that social science in a state of 'crisis' has turned to 'methods' to find a solution to its problems. He argues that it is by borrowing from para-military, post Second World War project strategies, like NASA programs in the USA, that design methodology got into civilian applications. This consisted of applying a rational approach, based upon system theory as a tool for reaching an optimum design solution (Foque, R. 1982).

It obviously means that design is seen from a 'pragmatic' perspective, and will rely considerably on scientific foundations. Yet, for a better understanding of the borrowing made from science; systems theory concepts and mathematical modelling will be looked at in Chapter 3.

The term 'design methodology' refers to the study of methods of designing, it suggests that design as an activity may be the subject of scientific investigation. This does not necessarily mean that design is exclusively a scientific activity. Also, the term 'design methods' refers to the procedures of the act of designing. According to the Design Methods Group (DMG), design methods are :

> "...step-by-step, teachable-learnable, repeatable and communicable procedures to aid the designer in the course of designing,"
>
> (DMG,1978)

Most significant analysis of the design process suggests a separation of the two crucial ingredients of design; **logical analysis** and **creative thinking**. The articulation of these two activities is generally used to describe the design process (Tovey, M. 1986).

The most popular design strategies among design methodologists (Cross, N. 1985) are the following :

- Sequential design (Asimow, M. 1962)
- Holestic design (Jones, J.C. 1970).
- Convergent design [P-D-I] (March, L. 1976).
- Flexible design (Alexander, C. 1964).
- Divergent design (Halprin, L. 1969).
- Focused design (Archer, B.L. 1965).

It is generally accepted that the overall structure of the design process include the three following steps :

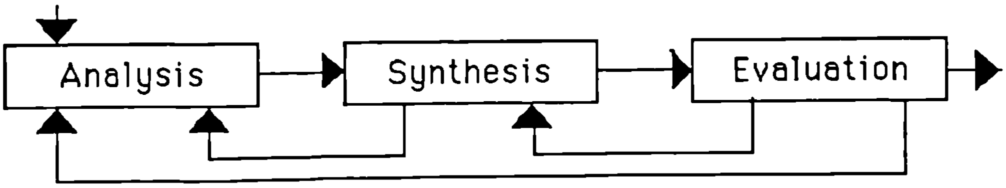


Figure 2.1 - The three stages design process.

These, in a cyclic process, should carry on until the identification of the design problems, and subsequently the production of a solution.

In this respect, Markus (1967), has noted that there are two distinct structures in designing; a 'vertical' one of the sequential phases, and a 'horizontal' one of iterative cyclic processes. The 'vertical' structure attempts to regulate the development of a design, usually from outline to detailed proposal. Whereas, the 'horizontal' structure is usually in the form of a number of steps in a design process with frequent recycling over a number of these steps. The core of these structure frequently consists of four steps:- **Analysis, Synthesis, Appraisal** and **Decision**. A design process may consist of hundreds, or may be thousands, of such sequences put together in ways that are determined not by some abstract flowcharts but by the nature of the design task itself (Markus 1967).

One of the major issues in the field of design methods is the 'direct' participation of users in the decision making process of design. In this respect it is believed (Broadbent 1979) that there are three generations of design methods.

This is put very clearly by Nasar (1980), who argues that the first generation methods of design broke problems into parts, analysed and solved the parts, and recombined them into a synthesis solution. This way the solutions evolved logically from an appropriate stating of the problem by the designer or the 'expert methodologist', to the production of the 'best' solution. The sequential design strategy (Asimow 1962) and the 'pattern language' design method (Alexander 1964), can be considered as first generation methods. The criticism of these methods is mainly in the assumption of stating exhaustively the problem characteristics at the start of the design process, whereas many requirements are likely to occur to the designer and user during the synthesis stage (March 1976, Lawson 1980).

The second generation methods came partly as a response to the first. Methodologists stepped down amongst the people and presented them with means to plan for themselves. The medical faculty building of Louvain University, built on the outskirts of Brussels in the early 70's (Kroll, L. 1975), is one of the few buildings designed and built in that manner, but with little success.

The reason for its failure is due, according to Broadbent (1979), to participatory decision making having produced a building with technical oversights making the product unacceptable to users. Relatively more successful applications of this method were made in Third-World countries where the economical frame work and the professional institutions are less resistant to users participation to the design and construction process (Hassan, F. 1973).

A third generation of design methods is identified by methodologists, this one would incorporate both participatory decision making and the expert designer or methodologist (Broadbent 1979).

A way in which this method might proceed is suggested by Nasar (1980). The user might undertake a preliminary design using scale models. The methodologist or professional designer might then describe adjacencies, setting characteristics, and behavioural flows as portrayed in the model. On the basis of that description the participants might revise the model to improve certain connections or to correct some unforeseen behaviour problems. Later, another expert might assess the cost implications of their plan, and they might decide to revise the model. This method seeks cooperative endeavours rather than isolated efforts. One of the objectives of this research is to contribute in the achievement of such cooperation.

2.2 DESIGN AND INFORMATION PROCESSING.

2.2.1 DESIGN AND PSYCHOLOGY.

> "...the world of design is the thought in the heads of designers, plus the skills of designers in externalising their thoughts;..."(p 3).
>
> BIJL, A. (1987)

A literature survey of the design theory (Jones, C.J. 1970, Hillier *et al* 1972, Broadbent, G. 1973, Lawson, B.R. 1980, Akin, O. 1986, Rowe, P.G. 1987) reveals that designing is a specific form of human thought, of which the study falls in the area of the psychology of thinking.

Historically a number of theoretical approaches have been taken to the topic of thinking. The principle early approaches were those of introspectionism, behaviourism, Gestalt theory and neo-behaviourism (Gilhooly, K.J. 1982). Recently, the work in this field has been dominated by the information processing approach. To put the current developments of the design theory and design thinking in perspective, a brief historical overview of the above approaches will be given.

ASSOCIATIONIST AND INTROSPECTIONISM.

Associationism prevailed toward the end of the nineteenth century. It is described as a mechanistic type of doctrine using irreducible lawlike relationships (Rowe 1987). It postulated that ideas took the form of elements analogous to basic physical entities and that these elements were hooked together to form thoughts or insights about problems. Associated with models of atomic structure in the physical world and their law of contiguity, to account for the association of elements of thoughts (Newell, Shaw and Simon 1958). Such analysis was to be made by means of the classical introspective report. The main reason for the decline of such methods was that the question of thinking without images aroused considerable controversy between the introspectionists (Gilhooly, K.J. 1982).

BEHAVIOURISM.

The behaviourist approach came as a reaction to the self observed mental experiences of introspectionism. Watson, J.B. (1913), an early behaviourist felt that introspection was a futile approach, he argued that;

> "Psychology, as the behaviorist view it, is a purely objective, experimental branch of natural science which needs introspection as little as do the sciences of chemistry and physics."(p 176).

At this point psychology discards all references to consciousness, and is defined in terms of 'Stimulus-Response'. This will advocate the 'black-box' view of designing (Jones, J.C.1970). Behaviourism claims that behaviour should be the sole subject matter of psychology, and only by studying what people do -their behaviour- is an objective science of psychology possible (Atkinson, R.C., Atkinson, E.R. and Hilgard, E.R. 1953). This definition helped shape the course of psychology during the first half of this century, until the Gestalt theory.

GESTALT.

A contrasting view was put forward by the Gestalt psychologists who objected to the different forms of 'elementarism' presented by the classical introspectionist and the early behaviourist (Gilhooly, K.J. 1982). According to Kohler, W. (1947), it is absolutely impossible to develop psychology as a science using experiment alone, if the processes underlying experience are merely a dynamic province of a much larger functional whole. Illustrating the notion of whole, He argues ;

> 'No-body can understand a game of chess through watching only the moves in one corner of the board.'(p 248).

The word 'Gestalt' has no exact English translation -from German- though "form", "configuration" or "pattern" come close. It emphasizes that the whole affects the way in which the parts are perceived. For this reason, it is sometimes said that 'The whole is different from the sum of its parts', a favourite principle of Gestalt psychologists (Atkinson, R.C., Atkinson, E.R. and Hilgard, E.R. 1953). In dealing with thinking and problem solving, the Gestalt theorists emphasised the way in which the problem was perceived as a determining factor in the task difficulty. The solving process was described as one of perceptual restructuring in which the problem comes to be seen in such a way that the solution is obvious. However, the Gestalt explanations did not get transported to many fields of psychology apart from visual perception. This gave the way to a behaviourist counter-reformulation.

NEO-BEHAVIOURISM.

The neo-behaviourists, incorporated some of the Gestalt findings into their theory (Koestler, A. 1964). They suggested that richer internal processes have to be postulated within the behaviourist theory. This allowed for the concept of directed thinking (Gilhooly, K.J. 1982). However, by the time this work developed, the information processing theory that still dominates **cognitive*** psychology had begun.

INFORMATION PROCESSING THEORY.

According to Gilhooly (1982) the main stimulus for the development of the information processing approach to thinking was the arrival of the computer.

---

cognition* = is the mental process of perception, memory and information processing by which the individual acquires information, makes plans, and solve problems.

He believes that this approach takes the 'computer' as its key metaphor for the mind. The basic idea being that, in his cognitive aspects, man can be regarded as a computer-like system that codes, stores and transforms information. This is very often the caricature made of the influence of information theory on the psychology of thinking. A more accurate picture should include developments made in the information measurement and communication engineering fields (see Shannon, C.E. and Weaver, W. 1949, Garner, R.W. 1962).

Newell, Shaw and Simon (1958) have described a theory of human problem solving behaviour in terms of what they call 'information processes', which they believe can have applications in the theories of learning, of perception, and of concept formulation. In this respect they say:

> "The real importance of the digital computer for the theory of higher mental processes lies not merely in allowing us to realize such processes "in the metal" and out side the brain, but in providing us with a much more profound idea than we have hitherto had of the characteristics a mechanism must possess if it is to carry out complex information-processing tasks."(p 163).

Both, the mapping of traditional design processes on the computer, and the exploration of design mechanisms will be investigated in respectively Chapter 3 and 4.

It must be said at this point that the evolution of our understanding of intellectual processes described above is proper to a group of countries sharing the same language and/or intellectual background. Different groups of countries mainly French speaking and Russian speaking have had different approaches to the study of thinking, which interestingly enough has shown some remarkable parallels and convergences with the work of the English speaking psychologists (Berlyne, D.E. 1965).

2.2.2 PROBLEM SOLVING AND INFORMATION PROCESSING.

> " A problem arises when a living creature has a goal but does not know how this goal is to be reached."(p 1).
>
> Duncker, K.(1945)

Problem solving and logical reasoning have been given more attention than undirected thinking or 'daydreaming'. This is perhaps because directed thinking is technically easier to investigate. Psychologists have described a theory of information processing for problem solving (Newell and Simon 1972), which sees thinking as a hierarchical organisation of elementary processes carried out one at a time. It introduces the concept of 'task environment' or in other words 'design situation', which is admittedly impossible to describe in advance, because in architecture it is very complex and continuously changing. The 'problem-space' is the internal representation of the designer task environment, and is searched by the problem solver (i.e. designer) using three methods : recognition, generate-and-test and heuristic search.

Recognition is the basic procedure of unselfconscious design; the problem is reduces to a point at which a known procedure is applied (Heath, T. 1984). The generate-and-test approach is a variant of 'trial-and-error', were the results of tests are explicitly used to guide subsequent attempts to generate solutions (Rowe, P.G. 1987). The heuristic is any procedure, or device that contributes to reduce in the search for a satisfactory solution, also referred to as 'rule of thumb' (Perkins, P.N. 1981).

There have been several attempts to apply Newell and Simon theory to architectural design. Pfefferkorn's (1975) experimental work on the Design Problem Solver is an example of such borrowing. More recently Akin (1986) describes a Design Information-Processing System (DIPS) which he sees operating on two environmental sources of information: the first studying the problem and the second developing the solution, respectively 'receptor' and 'effector'. Three modules characterize the DIPS, one is what Akin calls 'external representations', which combines knowledge acquisition and information representation of the real world (Receptors + Effectors). The second is a combination of 'design processes' or 'Processors' in Newell and Simon terms, these consist of tests and operations chained to each other by links that simply indicate the passage of control. The last one is the 'organisation of memory', and this is done by using three basic methods of search : recognition; generate-and-test and heuristic search. Akin being obviously very close to the Carnegie-Mellon group of information processing theorists, has drawn heavily from this theory in developing his model of the design process.

The theory of information processing is itself under criticism from psychologists rejecting its serial assumption, as it clashes with the parallel thought processes view of thinking which believes that undirected thinking is sometimes concerned with anticipating future problems (Gilhooly, K.J. 1982). In a new speculative thinking about the working of the mind, Robert Ornstein (1987) reveals the human mind as a hybrid system composed of many 'small minds' that work on their own and even conflict, simultaneously processing feelings, fantasies, fixed routines, interpersonal responses and bodily skills.

The design process is certainly not sequential as resent research in cognitive strategies in architectural design have shown (Lawson 1979). This means that the applicability of Newell and Simon information processor to architectural design remain to be proved.

More recently, design is seen as a procedure in which an emphasis is made on the identification of whatever will allow the generation of a solution conjecture of the design problem at the earliest possible stage of the design process, in other words before the problem has been fully analysed and understood. The design conjecture in the form of undetailed design concept is used to give direction to the analytical thinking, which will either simultaneously or subsequently propel the development of the design proposal (Tovey, M. 1986).

2.2.3 UNDERSTANDING DESIGN.(1)

> "Attempts have been made to break the design process down into a chain of activities. This has not been too successful in determining how we work, but has instead given rise to many ideas about how we might or even should work."(p 87).
>
> Lawson, B.R. (1970)

The relevance of cognitive psychology to the understanding of design is not questionable. It certainly gives an insight about the human mental inner-processes during the design activity. Using this approach, Hillier *et al* (1972) have produced a view of design about which they argue :

> "...design problems are essentially pre-structured both by constraints and by the designer's own cognitive map ..."(p 29-3-1).

The design solutions (or 'conjectures'), pre-existing in the designer's cognitive capabilities, are seen as to be originating from :

> " ..- Knowledge of the instrumental sets, solution types and informal codes, and occasionally from right out side - an analogy perhaps, or metaphor, or simply what is called inspiration." (p 29-3-10).

In this situation, the purpose of design analysis is primarily, to test conjectures, or in other words, **evaluate** solutions.

Supporting this last point, Lawson (1972 and 1979), investigating cognitive strategies in architectural design, reinforces the conjecture-evaluation view of design. He studied strategies used in two-dimensional spatial layout problem-solving by architecture students and science students. In comparing their strategies, he found that whereas the science base students tended to search for underlying rules (analysis) and then proposed a solution which satisfied the rules (synthesis), the architecture students proceeded by trying alternative configurations (conjectures) and then testing whether they complied with the rules (evaluation). These are respectively described as problem-focused strategy, and solution focused strategy (Lera, S.G. 1981).
This approach of design understanding unveils the large overlapping that exists between the analysis, synthesis and evaluation stages of design. As against the information processing approach described in section 2.2.1, that presented design as a linear, sequential and repetitive succession of events. The cognitive approach suggest that the analysis and synthesis stages take place almost simultaneously, with design evaluation going on in parallel, and taking part in the generative process. This view of design is going to be used as a frame work for this thesis, and will be further explained in Chapter 5.

Having said that, it does not mean that design is fully understood. In fact, there is still much to explain about design :

- How undirected thinking influences design ?
- What guides the production of design conjectures ?
- How design gets evaluated ?
- etc...

These questions could be answered by different approaches to the research of design :

1- Interview designers (Darke, J. 1979).

2- Observe what designers do (Lera, S. 1982, Hodde, A. 1988).

3- Construct artificial experiments under controlled conditions (Lawson 1972).

4- Think about design.

More recently, the development of computer aided architectural design systems has encouraged the investigation of design methods. It also required the development of mathematical models, which will be described in the next chapter. This type of work could be considered as a way to improve our understanding of design, and offers new means to comprehend designers decision making processes.

5- Develop and use computer based design aides.

Although it is suggested by Lawson (1980) that it might not be a good idea to try to fully understand design;

> "Perhaps we should hope never fully to understand the way designers think, for it is exactly because the designer does not know what he will think next which makes design such a challenging and satisfying occupation..."(p 202).

Nonetheless, there are some good reasons to pursue research work for a better understanding of design mechanisms. Firstly, to improve education of design by, for example, describing design methods that have produced 'good' or 'bad' solutions. Secondly, to help designers justify their design decisions to other members of the design team, or to the clients and/or users. Thirdly, to enable the identification and location of design errors, this is to prevent mistakes being repeated. Finally, use the observations to help guide the specification of computer aides for architectural design, the ones destinated to aid the 'architect', not the 'draftsman'. In the next chapters, this last point will be further explained.

---

(1) I am deeply indebted to Pr B.R. Lawson for enlighting discussions on the subject of this section, of which some ideas were suggested by him. Needless to say, he is not responsible for my conclusion.

Chapter 3 : COMPUTER SIMULATION AND MODELLING.

3.1 COMPUTERS TECHNOLOGICAL FUNDATIONS.

> "Roughly speaking those who work in connection with the ACE *(Automatic Computing Engine)* will de divided into its masters and its servants. Its masters will plan out instruction tables for it, thinking up deeper and deeper ways of using it. Its servants will feed it with card as it calls for them. ... As time goes on the calculator itself will take over the functions both of masters and servants. The servants will be replaced by mechanical and electrical limbs and sense organs... The masters are liable to get replaced because as soon as any technique becomes at all stereotyped it becomes possible to device a system of instruction table which will enable the electronic computer to do it for itself. It may happen however that masters will refuse to do this. They may be unwilling to let their jobs be stolen from then in this way. In that case they would surround the whole of their work with mystery and make excuses, couched in well chosen gibberish, wherever any dangerous suggestions were made. I think that a reaction of this kind is a very real danger. This topic naturally leads to the question as to how far it is possible in principle for a computing machine to simulate human activities."(p 20-21).
>
> TURING, A.M. (1947)

There are different approaches to making computing machines. One of these, the analogue computer, is now outdated and no longer used. When we talk about computers here we are looking at a particular restricted class of machines - the stored program digital computer. These are the type of machines that have been commonly available since the 1950s, namely 'digital computers'.

3.1.1 Computers historical development.

The original 'concept' of the digital computer can reasonably be attributed to two cybernetics pioneers; John Von Neumann and Alan M. Turing. They both, in separate reports, agreed on the description of a device that would have to carry out four basic functions: INPUT, to take in data and instructions; MEMORY, to keep track of the data; PROCESSING, to do the actual computing; and OUTPUT, to report the answer back to the human user (Michie, D. 1980, Reid, T.R. 1985). Their work would have been impossible without the mathematical foundation provided by the English philosopher and mathematician George Boole. In the mid 19th century, Boolean algebra showed how logical problems could be expressed in symbols. These symbols could then be manipulated without destroying the sense of the original problem, provided that all the factors involved in the problem could be expressed by variables that have only two states which Boole called "true" or "false". It is, in what is considered the most influential scientific text of our time '*Principia mathematica* '(1910-1913), that Whitehead, A.N. and Russel, B. argued that logic is not only inseparable from mathematics but is also the foundation of it. This work developed also a propositional calculus in which problems could be solved in terms of a series of statements that are either 'true' or 'false'.

The construction of the first general purpose computing machine is attributed to Vannevar Buch in the 1930s (Evans, C. 1981). As against the special purpose calculating machines like Pascal's '*Pascaline* ' that was exclusively mechanical -in 1644 the first significant calculating machine, followed by Charles Babbage '*Analytical Engine*' in 1833. Bush, introduced thermionic tubes or valves as a remedy to the slow speed operation, imperfect tolerance, and the difficulty of switching to solve one type of problem rather than another on the mechanical differential analyser at the Massachusetts Insitute of Technology. Working at the same Institute, Shannon (1938) demonstrated that it was possible to perform complex mathematical operations by means of relay circuits. Those had only two possible states; "on" or "off", which could completely emulate Boolean logic. It is generally accepted that the technological development of computers has evolved through successive hardware performance improvements, with the principle of binary switching remaining the same. Each of these advances has marked the start of a new computer generation.

- First generation computers:

The first general purpose machines (programmable), had their mechanical components replaced with valves in which the values were stored as voltages. The valves were large and unreliable, and consumed an enormous amount of power, which, of course, they also discharged in heat. Consequently, they had to be spaced widely apart and supplied with coolers, making them enormous in size (Evans, C. 1981). The second World War was the main stimulus to develop such machines; Alan Turing with a group British and Polish mathematicians gave the Allies an invaluable step up by cracking the Germans' "Enigma" military code using a real-life variation of the abstract 'Turing Machine' (Hodges, A. 1983). Cryptography, has since been one of favourite applications of computers number crunching power.

- Second generation computers:

The invention of the transistor by three Bell laboratories engineers in 1947 promised to eliminate all the bugs of the vacuum tube. The transistor achieved amplification and rapid on-off switching by moving electronic charges along controlled paths inside a solid block of semiconductor material (Reid, T.R. 1985). Not only the transistor is smaller then the valve but much less energy consuming, this resulting in much smaller and much more reliable component.

- Third generation computers:

The improvement in the computer components in the 1960s continued with the introduction of integrated circuits, this marking the start of the third generation of computers. Then, many transistors and their connecting wiring could be built on one circuit or microschip. It not only contributed to further reduce the size of computers but also made their cost more accessible to large government bodies, as well as research and education establishments. It is during this period that high level languages like Algol and Fortran became commonly used. These, encouraged scientists to implement computer applications in various engineering fields.

During the same period, important research was done in computer interactive graphic, producing work like Sutherland's (1963) publication on the 'Sketchpad'.

- Fourth generation computers:

Very large scale integration (VLSI) circuits appeared in the 1970s. These are basically extremely small circuit boards where transistors were replaced by layers of semiconductors like silicon. This marked the start of the proliferation of microcomputer systems.

Myers, G.J. (1978) argues, that the construction of physical computing devices has advanced significantly; for example, circuits speeds and densities have increased by orders of magnitude, new storage technologies have been invented, and the microprogramming concept has been exploited. The software field has advanced tremendously; for instance, there are now better tools, methodologies, and programming languages, software applications are more and more sophisticated, new algorithms have been invented, and the construction of such programs as operating systems and

compilers are fairly well understood.
However, there have been almost no advances at the hardware/software interface, the level of a system referred to as the **computer architecture**. This is perhaps not totally true (or not any more), because we have seen very recently parallel processing machines (Transputers) coming out of research laboratories and showing great promise as the next step from Von Neumann computers architecture (Walker, P. 1985). At a more experimental level connectionist machines (Neural Networks) are becoming a realistic alternative. They were introduced to model neural electrical activity, based on concepts derived from an analogy with the human central nervous system (Denker, J.S. 1986). Projects like these, have started off the long awaited next computer generation.

- Fifth generation computers.

The challenge to revolutionise computer hardware and software was taken by Japan in April 1982, when the Institute for New Generation Computer Technology (ICOT) was founded. It was a major and impressive national plan of the Ministry of International Trade and Industry (MITI), called 'Fifth Generation Computer Systems'. Moto-oka *et al* (1982) anticipated the emergence of knowledge base systems (KBS) and Expert Systems as the key development areas of the new computer generation. In response to the Japanese challenge, Britain started an advanced information technology program, known as the 'Alvey Program' (Feigenbaum, E.A. & McCorduck, P. 1984). This one included a project directly related to the construction industry, which consisted of an Expert System for building cost management (It will be briefly described in section 8.2.1.).

Basically, the fifth generation computers function is not information processing, in the conventional sense, but 'knowledge processing'. Intelligent Knowledge-Based Systems (IKBS), are the central elements of this new computer generation. They are based on systems that draw reasoned conclusions from knowledge (inference processors). This requires a radically different concept of the hardware/software design. Based on a mutual interaction of algorithms, computer architecture and technology, it consists of an assembly containing a knowledge based processor, an inference processor and an intelligent user interface (Bishop, P. 1986).

In spite of the conceptual break with previous computer generations, it would seem that the hardware of the fifth generation computers will still be based on Very Large Scale Integration of semi-conductor components. However, a very higher degree of parallelism will distinguish the new computers architecture.

Thus, new programming languages are required to cope with the innovative hardware, and the types of processing for knowledge base management. Non-procedural languages like OCCAM, ADA, and LISP are among the solutions. As well as declarative languages like PROLOG and HOPE (Bishop, P. 1986).

The intelligent user interface of the fifth generation computers is expected to reach the level of interaction between computer and user to something approaching the intelligence inherent in inter personal communication. Techniques of natural language recognition, speech synthesis and voice recognition are some of the key instruments to achieve such computer/user interaction. The current range of fourth generation computers user interface is likely to move on to the next generation, namely Windows, Icons, Mice, Pointers and Standards menus (WIMPS).

3.1.2 Hardware versus software.

The type of computers used during the course of this research were fourth generation machines. Known as having a 'Van Neuman' type of architecture (see Figure 3.1).

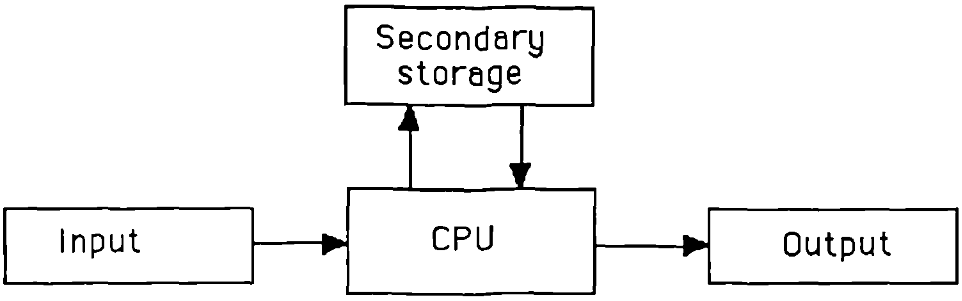


Figure 3.1 - Generalised Computer System -

Three main components constitute these machines, hardware, firmware and software:

- Hardware.

Data arrives to, and leaves the computer Central Processing Unit (CPU) through the input/output (I/O) devices. These are hardware devices to interface directly the CPU with the user. They cover a wide range of capabilities and characteristic. Terminals, printers, plotters, scanners, digitizers, voice synthetisers and voice recognition instruments are different types of I/O devices. An input device is required to provide a means to enter data and programs in the CPU, also an output device is require to receive at least some of the results of the processing.

- Firmware.

Firmware is that aspect of the operation of the system which, together with the hardware, defines the structure or shape of the computer system as seen from the user's viewpoint. It is a sort of interface between the hardware and the software of a machine, consisting of elementary sequences of instructions to the CPU to be performed frequently and rapidly. Usually, it takes the form of hardwired microprograms located in a Read Only Memory (ROM). Although, recently, the use of various semiconductor electrically erasable and reprogrammable read-only memories (EPROM) has brought together both the high speed and ready reprogramability of microprograms (Baywater, R.E.H. 1981). Firmware is often used in computer animation to speed up graphic manipulations that use standard algorithms, and needs to be performed at high speed for real time animations for example.

- Software.

Software is the name given to a program or a set of instructions for a computer. There are two broad classes of software: applications software and system software. An application program is set of computer instructions whose purpose is to carry out a specific user's application, a well known example of such program is a 'word processor'. Systems software (also called Operating System) refers to programs consisting of a set of instructions whose purpose is oriented around either efficient handling of an applications program or efficient utilisation of the computer system for the benefit of its applications-oriented to the users. Thus, systems software is of indirect, but critical, benefit to the users, whereas an application program is of direct benefit to that same user (Murray, T.J. 1985).

3.2 MATHEMATICAL MODELLING.

3.2.1 Systems theory.

> ' A building functions the way it does because its parts have certain attributes and because a given set of relationships exist among them. If these attributes and the way they are connected change, the function changes... It follows from these

> considerations that effectively to approach architecture in functional terms involves treating it as a system.'(p 14).
>
> Handler, A.B. (1970)

The 'systems theory' is the emerging contemporary view of organised complexity. It replaces the atomistic view of organised simplicity as seen by the Newtonian science. Which viewed the physical universe as a mechanistic aggregate of parts in isolable causal relations (Laszlo, E. 1972). The systems view instead, treats systems as integrated wholes of their subsidiary components. Neelamkavil, F. (1987) defines a system as :

> "...a set or assemblage of entities (elements or components) interrelated to each other and the whole so as to achieve a common goal."(p 17).

Systems can be classified as natural (solar system) or artificial (computer system). Interrelationship or interdependence is the key word here. Systematic models are intended to represent and interpret the real world, in such a way that predictions can be made about its behaviour. A system is composed of one or more sub-systems, and sub-systems consist of one or more sub-systems, and so on. We come across several systems in our daily life such as the postal system, water distribution system, tax system or educational system. The systems relationship must achieve reasonable consistency. Markus (1967) suggested that architecture could be seen as a system comprising four sub-systems - the building system, the environment system, the activity system and the objective system. The purpose of systems study is to learn, to design, to change, to preserve, and if possible control the behaviour of the system. Nevertheless, March (1974) points out potential problems in dealing with systems. He argues that when studying a system as if it were a set of elements with fairly strong relationships between them, we may be overlooking a large number of weak relationships between elements which may have more effect than a few strong ones. Systems can be studied by direct experimentation, by building prototypes, or by building mathematical/logical models. Experimentation with a prototype is often too expensive, impractical, risky, or time-consuming, and will not be dealt with in this thesis. It is interesting though to mention that scaled models suffer from the deficiencies inherent in modelling at small scale the properties of the full-size structures, and consequently suffer in accuracy if the physical laws do not give exactly the same effect as in the full-size building for example. Analogies and digital calculation demand an even more precise knowledge of the physical laws, and the external conditions to be encountered (Cowan, H.J., Gero, J.S., Ding, G.D. & Muncey, R.W. 1968). Mathematical/logical modelling is central to the specification of computer aided design systems and will be examined in the next section.

3.2.2 Mathematical models.

> "... a class of architectural problems can, under suitable conditions, be transformed into a class of mathematical problems..."(p 42).
>
> March, L. (1976)

Among symbolic models, the mathematical models are the most important and widely used category of model (see Figure 3.2).

A mathematical model is a set of mathematical and logical relations between various system elements. Being derived from '*modus* ' (in Latin a measure) the word 'model' implies a change of scale in its representation (Aris, R. 1978).

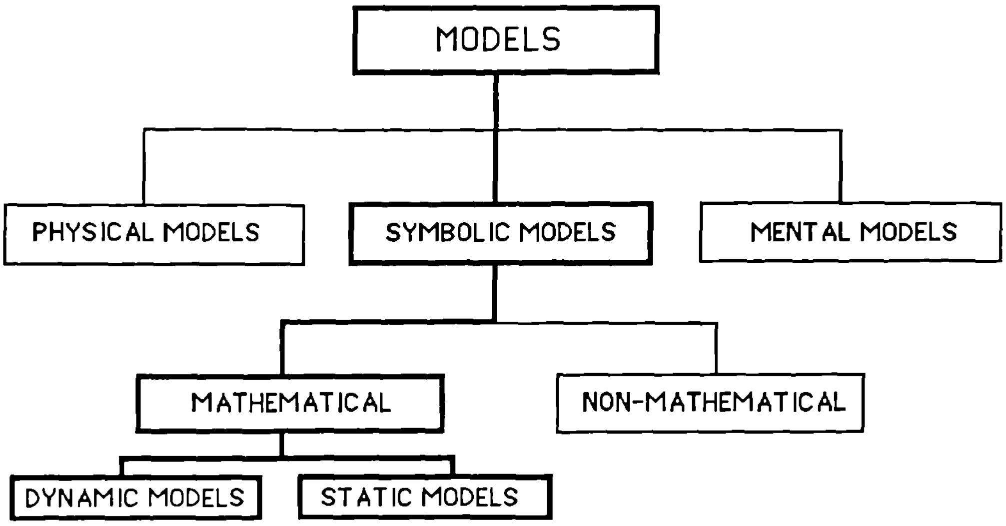


Figure 3.2 - Types of models according to Neelamkavil (1987) p32.

Unquantifiable elements like attitude, values, etc., cannot be included in mathematical models. The dynamic models are generally described by differential equations, while algebraic equations characterise static mathematical models.

Maver (1970) in his analysis develops three categories of theoretical modelling for architectural design :

1. Formal mathematical modelling.

This is defined as the modelling of the relationship between interdependent variables using mathematical expressions. Regression analysis techniques are used to establish relationships between selected design parameters, and then linear functions calculated. Maver identifies three short comings to this modelling method. First, selected relationships might not fully represent the real world situation (this point was already made in section 3.2.1), and if they do, they do not necessarily have a linear relationship. Second, there is a danger in extrapolating from past experiences statistical data. Finally, a model using the relationships of previous solutions might not promote innovative design. These arguments are not proper to mathematical modelling for architectural design, but are general to operational research methods.

2. Heuristic modelling.

Heuristic modelling in Markus' terms is to search for the 'best' alternative out of an infinite range of possible solutions. This is in fact the method used in early computer aided floor layout optimization research (Whithead, B. and Eldars, M.Z. 1964). The 'heuristics' are the rules applied to limit the area of search. Once selected, these take the form of algorithms, and according to Shaviv (1987) when implemented require too much computer time or trivialize complex architectural problems.

3. Simulation modelling.

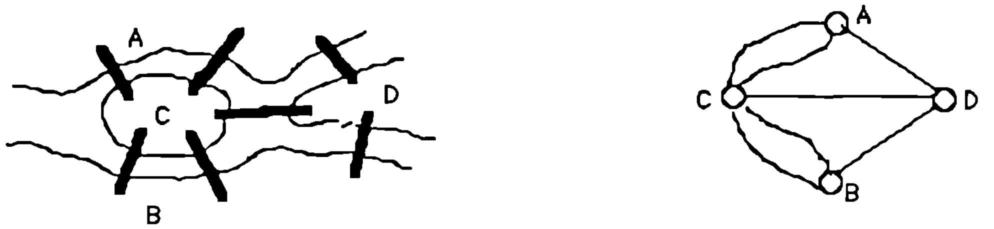
'Simulation' here is taken to mean the modelling of design solutions. By modelling a variety of solutions, performance indices can be compared and a satisfactory solution can be approached progressively, says Markus. Parametric modifications of the model with relational integrity preserved, are essential to undertake such modelling.

We shall come back to this categorisation of design modelling later in chapter 4. However, it is interesting to point out that these modelling methods (each having a different design application), if combined, would cover most stages of the design process. Therefore, it seems that the ultimate mathematical model for an integrated system should support the following requirement. First, the definition of strong, as well as weak relationships between elements. Second, allow each element to have several attributes to apply heuristics with multi-variate problems search. Finally, support both static and dynamic mathematical modelling.

3.2.3 Graph theory and connectivity.

> " If something like a theory of architecture will ever be developed, then one of its first chapters will deal with the theory of cell configurations..."(p 179)
>
> Rittel, H.(1970)

A growing interest in the theory of graphs has developed during the last two decades, particularly among applied mathematicians and engineers. This is because of its inherent simplicity, and its wide range of applications. The first formal description of its principles was made by Euler, a Swiss mathematician, in 1736 (Deo, N. 1974). He set himself the problem of walking around the city of Konigsberg, where seven bridges join different parts of it, and by starting from any point, crossing all the bridges, each only once, to arrive back at the starting point (see figure 3.3).



- Konigsberg bridge problem.

- Euler's graph of the bridge problem.

Figure 3.3 - First graph application.

Euler proved that a solution for this problem does not exist, also that a simple geometrical figure of points and lines could represent any kind of relationships. By definition, a graph consists of a set of objects called *vertices* , and another set, whose elements are called *edges* . Each edge is identified with an unordered pair of vertices. The most common representation of a graph is by means of a diagram, in which the vertices are represented as points and each edge as a line segment joining its end vertices. Often this diagram itself is referred to as the 'graph'.

There is no doubt, that the theory of graphs has useful applications in problems of architectural layout and planning (Levin, P.H. 1964). It has often been used to represent the relation of adjacency between pairs of rooms in the plan; that is to say whether rooms do or do not touch (see Figure 3.4), abut, or share some length (may be surface) of wall in common (Steadman, P 1976). By applying a special branch of mathematics, such as graph theory, to architecture, we can use existing algorithms already developed for other purposes. March and Steadman (1971) have previously used Kirchoff's law (for electricity) to solve the problem of dimensioning architectural plans. More recently, Roth and Hashimshony (1988) have used 'Max-flow Min-cut' algorithm for decomposing complex graphs, and to turn non-planar graphs into planar ones.

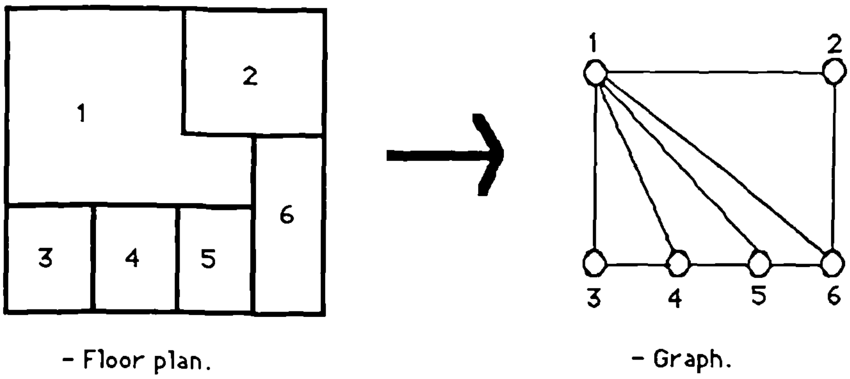


Figure 3.4 - A space adjacency planar graph.

Less frequently, graphs have been used to represent adjacencies between spaces in three dimensions (Teague, L.C. 1970).

Whereas, most work previously done, was motivated by a wish to solve design circulation problems of large building plans, using computer methods based on heuristic approaches, or exhaustive enumeration approaches. We will not be concerned here with generating optimum plan layouts, but with establishing the relationship of spaces in a computer building model, and using this data for building design evaluation.

3.3 ALGORITHMS AND DATA STRUCTURES.

" Algorithms + Data Structures = Programs "

Wirth, N. (1976)

Berztiss (1975) summarises three distinct phases in the process of solving a problem using a digital computer. Firstly, a mathematical model of the problem has to be found. Secondly, an algorithm has to be developed within the mathematical model. Finally, a computer representation has to be selected for the data on which the algorithm is to operate. The computer representation of data is also called 'data structure', and any manipulation of it operates under the control of 'algorithms'.

According to Eastman (1978), two approaches to Man-Machine collaboration were initially envisaged in developing computer based design aids. These were outlined by Coons (1963), and were based on a scenario developed from the joint development of computer graphics and time-sharing systems.

First, the automatic encoding of data from manual representational methods was envisaged. This approach relies on pattern recognition techniques and systematic analysis, the complexity of syntax drawings and design information interpretation are central to the slow progress of this method (Klinger, A., Fu, K.S. and Kunii, T.L. 1977).

The second approach relies on an essentially machine readable encoding of design information. In other words, the computer defines the syntax by which communication takes place. It is the actual computer language 'data types' that will define the abstract entities.

Ultimately, a merger of these two approaches should take place when the translation between drawings and machine readable formats improves (eg: Scanner technology).

The current development of Computer based design aids falls in the second category of Man-Machine design collaboration, of which the communication syntax is represented in the computer by the geometrical algorithms and data structures to which the next section is to be devoted (see Figure 3.5).

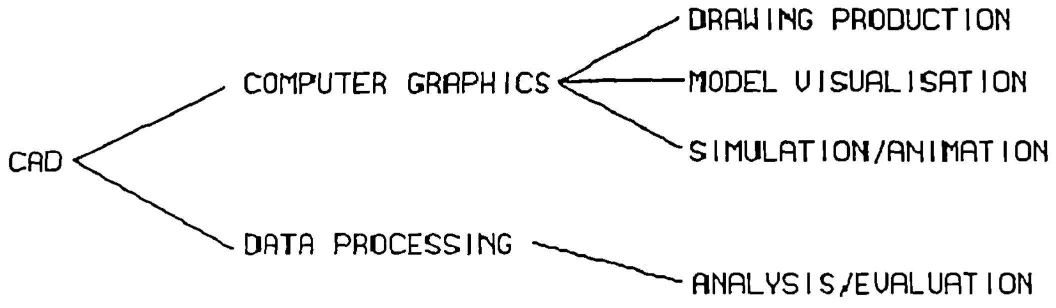


Figure 3.5 - CAD development overview.

CAD is an acronym, some times used for 'Computer Aided Design', and at others for 'Computer Aided Drawing', or even 'Computer Aided Drafting'. Unless specified, it is the first meaning that will be used during the rest of this thesis.

3.3.1 Algorithms.

The definition given to an algorithm by Kronsjo (1979) is the following:

> " A procedure consisting of a finite set of unambiguous rules which specify a finite sequence of operations that provides the solution to a problem or to a specific class of problems... "(p 1).

More briefly, the Concise Oxford Dictionary defines an algorithm as a "process or rules for (esp. machine) calculation". The word 'algorithm' it self comes from the name of a 9th century Arab mathematician 'Al-Khwarismi' who described how to perform the four arithmetic operations in the decimal number system. The notion of an algorithm has been extended quite recently, and denotes nowadays a set of rules which specify a sequence of actions to be taken for solving a problem (Marciszewski, W. 1981). Very often, an analogy is drawn between algorithms and computer programming, where a computer is made to to work through a set of procedures (i.e.: an algorithm), to achieve a desired result.

It is usually in the planning phase of writing a computer program that algorithms are used, in which case it is very common to use a flowchart as an aid in analysing what the program is to do. The flowchart is purely an aid to the programmer in visualizing the program steps : it traces out the algorithm, or the general shape of the solution of the problem. As such, it is part of the program documentation (see Figure 3.6).

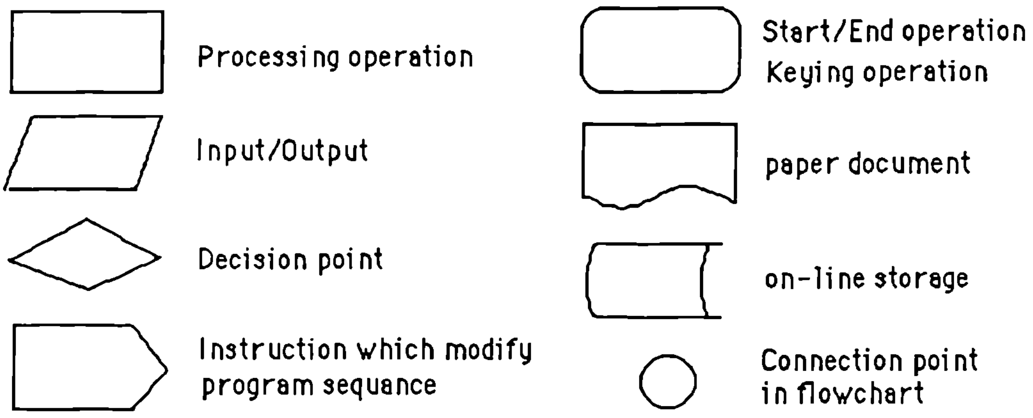


Figure 3.6 - Conventional symbols used in flowcharts.

3.3.2 Geometrical data structures.

There are at least three approaches to the design of a CAD system; the two-dimensional drafting system approach, the two-and-a-half-dimensional system, and the three-dimensional modelling system approach. The first creates and manages basic geometrical primitives. The manual sketching and drawing of lines and arcs on paper is replaced by similar operations on a Visual Display Unit (VDU) or Sketchpad. It is only two-dimensional geometry that is supported at this stage. The second creates a three dimensional model, but the views of the model are limited to a few orthogonal projections. The last approach creates an analog model of a desired three-dimensional object, and stores it as geometric data, attributes, and relationships. This approach can be used for both visualizing a real-world object and generating traditional production drawings.

- Two dimensional system data structures (2D):

Computer aided drafting systems store drawing information as a collection of images, items, or objects. These terms refer to the same elements. Unfortunately, there is no standard terminology. Each image corresponds to a primitive data structure that contains all of the information necessary to fully describe each individual image.

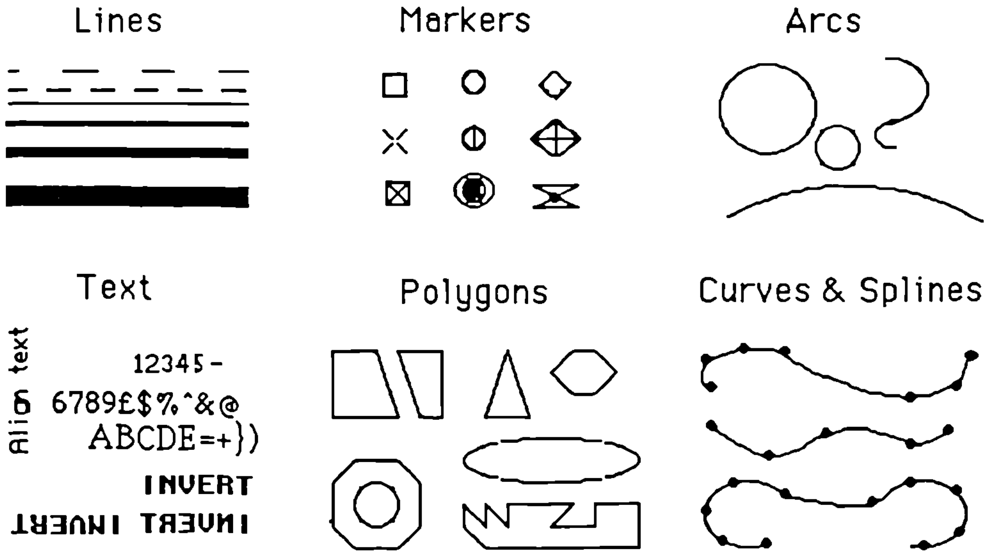
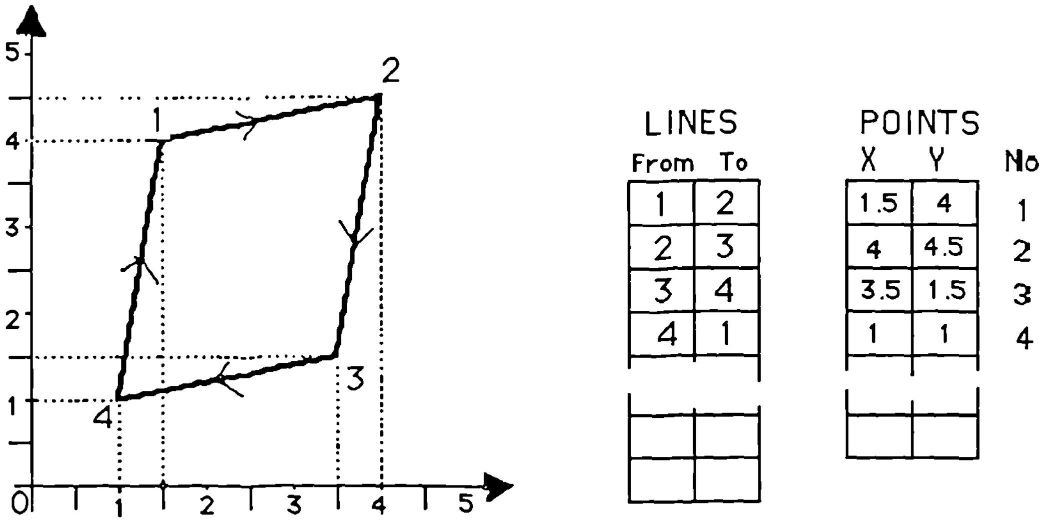


Figure 3.7 - Two dimensional graphical primitives.

The input of drawings is done through a 'graphic editor'. It is the equivalent of a word processor, but, instead of manipulating alpha-numerical information, it allows for the input and manipulation of graphical elements such as lines, circles, regular or non-regular shapes and so on. Most common drawing primitives encountered in two dimensional CAD systems are shown in Figure 3.7.

Each of these graphics primitives is defined with geometric information represented in a Cartesian system, were points are defined by x and y coordinates (see Figure 3.8).



LINES

From	To
1	2
2	3
3	4
4	1

POINTS

X	Y	No
1.5	4	1
4	4.5	2
3.5	1.5	3
1	1	4

Figure 3.8 - Graphical input of a shape and data storage.

In addition, each primitive has data attributes to describe how it is drawn or visualised. Lines and curves will have a colour, line-style, and width parameter to be used by the display and output device to render the graphics.

The Graphical Kernel System (GKS) was developed with the objective of easing the production and manipulation of pictures, in a way that does not depend on the computer or graphical device used (Hopgood, F., Duce, D., Gallop, J. & Sutcliffe, D.C. 1983) . It supports four main primitives :

(1) - *Polyline* : Draws a sequence of connected line segments.

(2) - *Polymarker* : Marks a sequence of points with the same symbol.

(3) - *Fill area* : Displays a specific area.

(4) - *Text* : Draws a string of characters.

With these geometrical primitives all kind of modelling may be performed. The usual 'Euclidean Transformations' are scaling, rotation, translation and reflection. This may be accomplished by the use of two-dimensional transformation matrices, on which a wealth of literature is now available for computer graphics software programmers (Newman, M.N. & Sproull, R.F. 1981; Bowyer, A. & Woodwark, J. 1983; Foley, J.D & Van Dam, A. 1984). To facilitate the transfer of graphical data between computers, the IGES (Internal Graphics Exchange Specification) was developed and emerged as the likely standard of moving data from one CAD system to an other (Mayer, R.J. !987).

- Two-and-a-half-dimensional modelling data structures (2.5D):

This approach is also called the 'box geometry' method. It basically consists of building a graphical description by the input of a number of related views. Each view being an orthogonal projection or section of the model. The attraction of such a method is that it is relatively simple to draw each of the views as a two-dimensional drawing. However, it is not a completely unambiguous description of a solid object. Perhaps a a not too significant limitation for some building designs, but certainly not the best way to observe and explore spatial conflicts in a multi-level and non-orthogonal design.

The data structure of two-and-a-half-dimensional modelling is a mixture of two-dimensional and three-dimensional modelling data structures.

- Three dimensional modelling data structures (3D):

First of all, we should make clear the difference between three-dimensional

surface modelling and three dimensional **solid modelling**. Both help visualise real world objects and provide extensive mathematical information for engineers and designers, but the two modelling types involve different mathematical foundations, different terminology, and present solutions to different design problems (Lyons, E. 1988).

Surface modellers represent objects as a series of surface entities, and result in models that approximate the appearance of physical objects. They provide information only about the skin of the object. On the other hand, solid modellers aim to create models that provide numerical information on real world objects. This is to be used to measure volume, determine mass, find centre of gravity, and examine all sorts of physical properties for design and engineering purposes. They provide information about both the exterior and interior structures of a solid. The difference between surface and solid modelers is in their notion of spatial relationship and location. Whereas a solid modeler can determine whether a point is inside, outside or on the surface of a solid. A surface modeller can only keep track of whether a point is on or off a surface (Lyons, E. 1988).

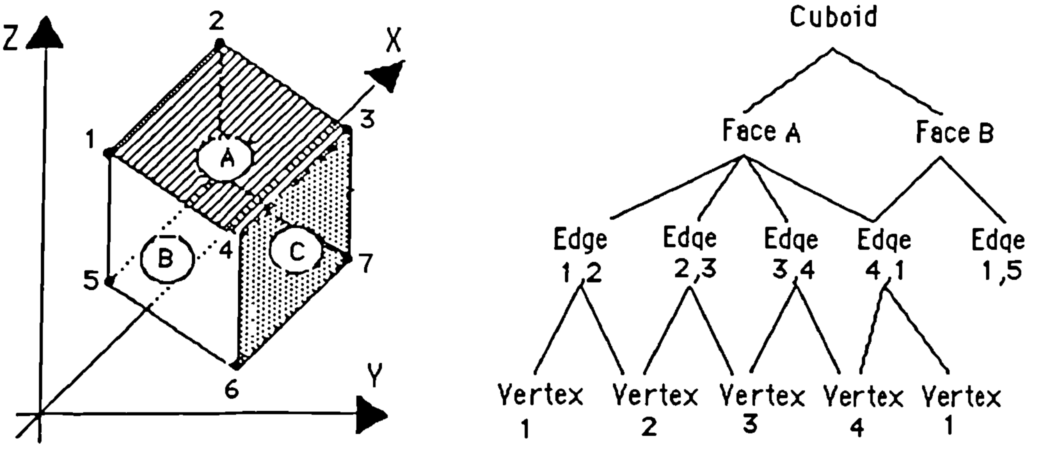


Figure 3.9 - Hierarchical model representing polyhedral computer shapes.

Architectural design CAD applications have mainly used surface modelling techniques for three dimensional building modelling (see Figure 3.9). Recently, there have been advocates of solid modelling for architectural design (Mitchell, W.J. 1986, Lyons, E. 1988), they promise to revolutionise CAD by providing a unified database for a complete building description, which could be used for automatic generation of drawings, and as a basis for various analyses. This does have the limitation to assume that the same drawings are used for both design and construction, which is not the case in architecture. Eastman, C. (1987), warns that this is inherently incompatible with architectural drawings and practice.

Taking the argument further, it can be said that architecture is as much concerned with **voids** as it is with **solids**, and the ultimate CAAD system should incorporate both void modelling as well as solid modelling. Incidentally, Yessios, C.I. (1987) suggests a computational method of 'void architectural modelling', which uses space enclosures as the primitive elements in architectural composition. It aims to address the deficiencies of solid modelling when applied to architectural design.

It might be useful at this point to explain how in principle the modelling of buildings is approached in different computer aided architectural design systems. There is first the approach that bases the construction of the computer building model on the input of 'space outlines'. The designer needs to have a clear idea of the geometry of each individual space of the building design. In a very simplified manner it means that the input consists of laying down all the spaces of a particular floor, by drawing in plan the space outline of each space. The system then asks for the height and thickness of internal and external walls, to then build a three-dimensional model of the building (see Figure 3.10). This approach was used by the ABACUS Unit of the University of Strathclyde, in the GOAL building modelling system (Sussock, H. 1982).

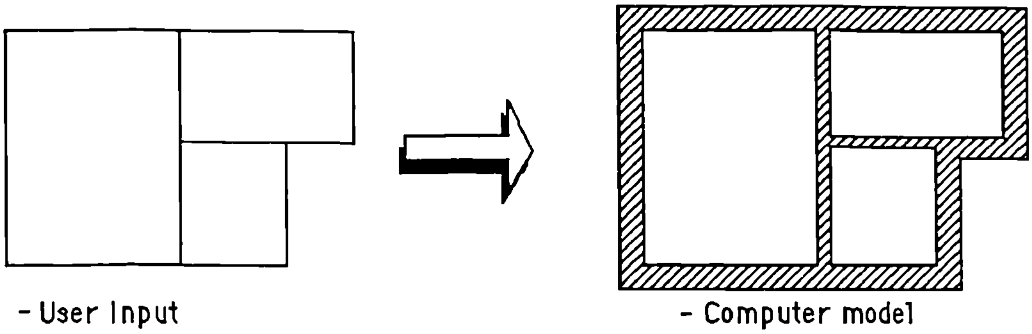


Figure 3.10 - The ABACUS approach to generate building models.

The second approach consists of defining in advance the building materials that are going to be used (i.e.: walls constructions, windows and doors), and then use those to describe the building plan. The system in this case will interpret the information to generate a three-dimensional space model (Figure 3.11 & 3.12). This method is for instance used by GABLE CAD system, which will be described in some details later in Chapter 7.

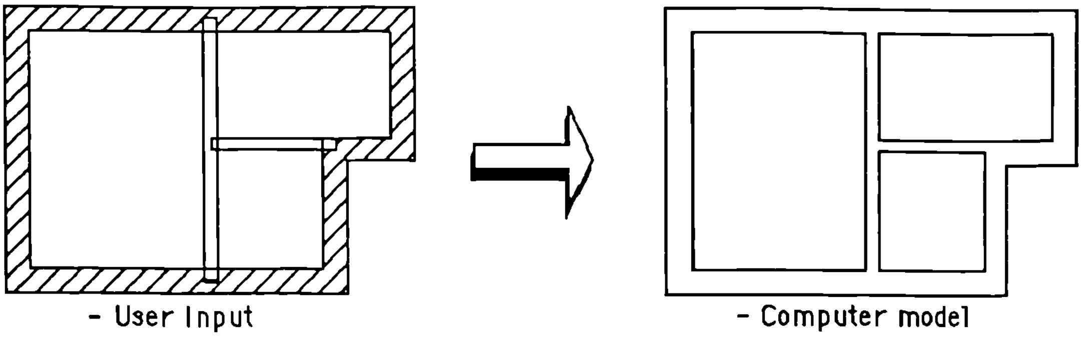


Figure 3.11 - The GABLE approach to building modelling.

The two approaches are complementary, the first one suits sketch layout design for space manipulation at an early stage of the design process. Whereas the second one is more appropriate for designs that are not so much concerned with the organisation of spaces (or rooms) in the first place, but want to produce accurate design drawings.

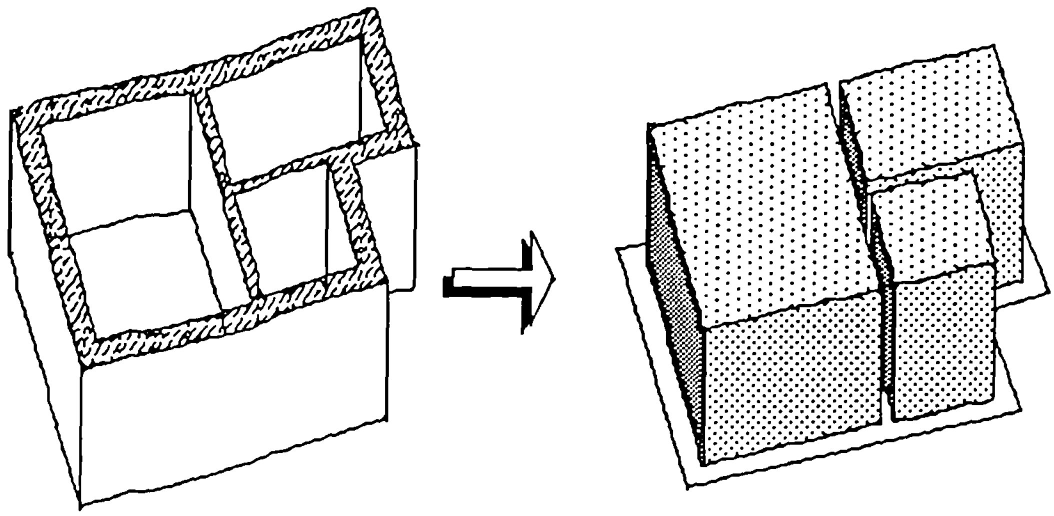


Figure 3.12 - Three-dimensional space model from a building model.

The combination of both is essential, to make CAD systems support the design process as describes in Chapter 2. In other words, to simultaneously analyse, synthesis and evaluate the design with sufficient information on the building model to make appropriate design decisions.

There is currently research going on at the University of Sheffield Department of Architecture, pursuing the aim of combining spatial modelling and elemental modelling. It intends to create a building on the system as a series of spaces, and afterwards break it up into individual building elements. However, at the time this manuscript is written, it is too early to speculate on the impact of this approach on computer aided design developments.

As far as geometrical data is concerned, the geometrical data structure of the building computer model can take only so much information, therefore specificational data on the building construction elements are stored in a standard data base. These data bases are managed by what is called a Data Base Management System, described in the next section.

3.3.3 Data Base Management Systems.

Whereas commercial CAD systems are depicted in terms of their geometric capabilities and their applications, very little is revealed about their data structures. Those are so jealously guarded, that advanced techniques of information processing such as database system theory have had very little repercussion on CAD software design.

Data Base Management Systems (DBMS) techniques have been successfully applied in commercial and business types of data processing, it is not until recently that they are being considered for CAD data processing.

Before going any further, may be the DBMSs concepts deserve to be given some explanation, and later, their relevance to the different stages of the design process will be discussed.

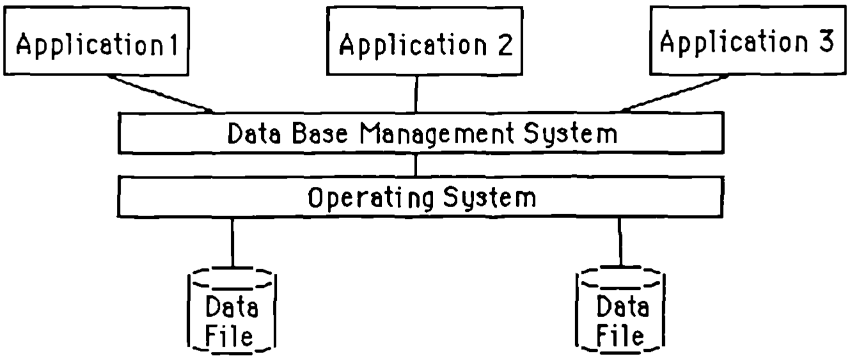


Figure 3.13 - Data Base Management System Processing.

The diagram in Figure 3.13 describes the role of the DBMS in computer systems. It consists of interfacing different applications with the host operating system and data files. It is important to differentiate between schema and sub-schema (Murray, T.J. 1985).

SCHEMA (Internal Model)	: SUB-SCHEMA (External Model)
- Interfaces with the computer technologies.	- Logical view of the data base.
- D.B.M.S.	- Query applications (Data Manipulation Language).
- Operating system access.	

It is generally accepted that the most significant characteristic of a DBMS is the separation of the **logical view** and the **physical view** of the data. The objective of this separation is to make data storage independent of the applications programs using that data.

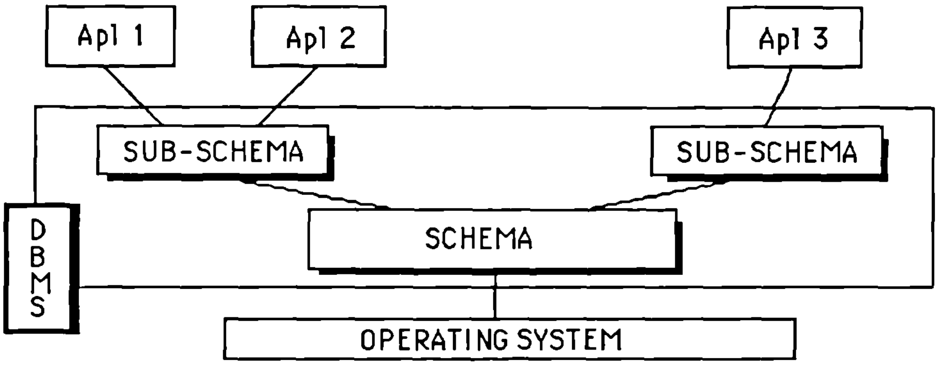
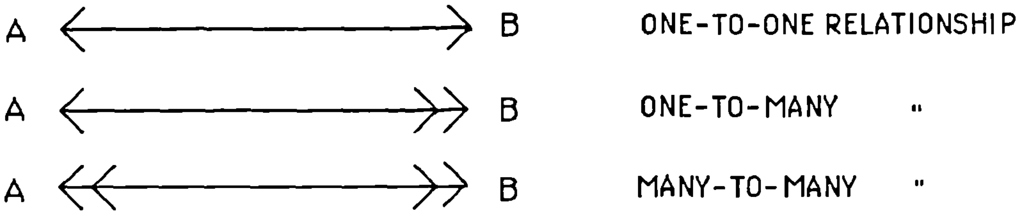


Figure 3.14 - General organisation of a relational data base system.

A schema is the logical view of the entire data base, whereas, a sub-schema is an applications program's logical view of the data base. Although only one schema will be associated with any data base, there can be multiple sub-schemas associated to the data base (see Figure 3.14). There exist different data models in DBMS; **hierarchical, networked** and **relational** models. Their differences lay in the way association between data items is performed. Basically, there are three types of data associations :



- The hierarchical data model.

A hierarchical data model represents data as a set of nested one-to-one and one-to-many relationships.

- The network data model.

The network data model represents data as a set of record types and pairwise relationships between records. Relations that involve more than two record types are not 'directly' permitted.

- The relational data model.

The relational data model uses the concept of a relation to represent what is usually called a file. A relation is viewed as a two-dimensional table.

Koriba, M. (1983), reviewing DBMS theories applied to CAD, concludes that Relational Data Base Management (RDBM) technology offers sound theoretical foundations upon which some common methodology of CAD software design may be drawn. This is emphasised by Kalay (1983), who explains that the data redundancy and extensive cross-reference linkage used by **hierarchical models** representing polyhedral shapes in computers, are unsuitable for storage and transmission over communication channels. These, problems are eliminated by the **relational model**, which allows compact storage and non-manipulative query operations through the **select**, **project**, and **join** operators (Codd, E.T. 1970, 1982 and 1985). Four data manipulation operations; (1) retrieval, (2) modification, (3) deletion and (4) insertion are usually grouped together and part of a language (data manipulation language - DML). Among conventional DMLs, it is SQL (

Structured Query Language; the acronym SQL is usually pronounced 'sequel') that has the largest acceptance in relational data base technology. This provides the user with formal means to rearrange the data so that it will conform to the needs of particular queries.

The relational model data structure represents data in a two dimensional tabular form. Much of the relational literature refers to these tables as 'relations'. Rows of a table are generally called *tuples* , also the terms 'row' or 'record' are often used. A crucial feature of relation data structure is that associations between tuple (records) are represented solely by data values in columns drawn from a common domain (Date, C.J. 1975). It is a characteristic of the relational approach that all information in the data base -both 'entities' and 'associations'- is represented in a single uniform manner, namely, in the form of tables.

Considerable work has been done on storing geometrical objects in a relation database system (Guttman, A. & Stonebraker, M. 1982, Kalay, Y. E. 1983, and Meier, A. 1986). Unfortunately, there is no room in this thesis to review the work done in this very promising field. However, a tabular data base with relational functionality, will be used in this research to store and manipulate data extracted from dedicated geometrical data files (see Chapter 7).

Chapter 4 : COMPUTER AIDED ARCHITECTURAL DESIGN.

4.1 WHY C.A.A.D. ?

> "The objective of a CAD system is to blend the designer and the computer into a problem-solving team able to attain the goals of design problems more efficiently than each one working alone."(p 61).
>
> Latombe, J-C. (1977)

4.1.1 Man-Machine collaboration in design.

Simon (1965), in his early work on computers and automation, distinguished two main areas of concern to the development of a man-machine research strategy. The first having a technological dimension, the second, an economic dimension. With each incorporating two opposite attitudes, one **radicalist**, and the other **conservative**.
For instance, according to Simon, a technological radicalist would believe that computers will be able to do anything a man can do, whereas, a technological conservative would assert that computers do only what they are programmed to do. Similarly, an economist radicalist would see in automation the start of a mass unemployment era, resulting with an organisation masterminded by machines, whereas, conservative economists will simply see in automation a continuation of the industrial revolution.
As a computer technology pioneer, Herbert Simon, identifies himself as a technological radical and economic conservative. There have been various attitudes towards the use of computers as human partners. Some, were very enthusiastic, whereas others like Weizenbaum, J. (1984 and 1985) have had a very skeptical attitude, especially concerning the subject of artificial intelligence (called also cognitive simulation or information processing psychology).

The professionals associated with the design and construction of buildings,

have certainly not been over'whelmed by the idea of working with machines. Whilst many architects were aware of the development of computers, very few were convinced that they had a direct bearing on their practice (Gero, J.S. 1983). The use of computers in technical areas associated with building design, not appearing until the late 60's early 70's.
Nevertheless, it must be said that the cost of computers at that time might have caused the caution of the design professionals. In other words, the productivity improvement promised by computers, was not sufficient to justify the high investment cost.
However, the research effort was sustained. Freeman (1987), suggests that apart from the two classical explanations given to technological change, namely; 'demand-pull' or 'technology-push'. There is a third form that could be defined as a new combination of radical innovations related both to major advances in science and technology and to organisational innovation. This is particularly the case of research in information technology and allied subjects. For instance this research work is itself aimed to apply new techniques (information technology) to help solve old problems (building design). Nonetheless, it is not until a new design tool has proved useful and cost effective, that it will replace an existing traditional method.

Anticipating the use of computers in architecture, Negroponte (1970) highlighted ways in which the machine could assist the design process , he identified three possible applications :

> 1 - The automation of current procedures, to speed up and reduce the cost of existing practices.
>
> 2 - The alteration of existing methods to fit within the specifications and construction of

the machine.

3 - The evolution of the design process represented in the machine, with man-machine mutual training, resilience and growth to be developed.

It is possible to view this categorisation in the light of existing computer applications to design.

The first approach could correspond to what is commonly known as Computer Aided Drafting, or also Computer Aided Design Drawing systems (Schilling, G.T. 1988). These are generally two-dimensional drawing production tools, and have been underlying the significant part of the increased productivity in design, attributed to computers. They can save a substantial proportion of the time spent on drawing production, especially when there is a high degree of repetition in the design work. Effectively, they consist of automating the drawing process and are fairly easily absorbed into existing design office procedures. The limitation of draughting systems is in coordinating information between different drawings. This is entirely left to the user, because drawings are held in separate two-dimensional files. It is also impossible to perform any evaluation work on such drawings, since all the information they contain is abstract graphical primitives with no relation to building elements.

The second category, would include all types of Computer Aided Design systems that support two-dimensional as well as three-dimensional modelling. These allow us to design buildings spatially, by storage and manipulation of construction elements (i.e.: walls, doors, windows, sanitary fittings, etc...). Systems that belong to this category and exhibit some building design capabilities, are called Computer Aided Building Design (C.A.B.D.) systems. The detailed information required by such systems on the building design solution, makes them difficult to use at early stages of the design process. They require that the user obeys very strict procedures, and impose the use of a pre-defined semantics and syntax that can sometimes be very complex. They also, occasionally, restrict the user to very regular forms, not allowing curved surfaces or various thickness materials. Obviously, a strong argument in their favour is the three-dimensional visualisation capability. Since they allow direct observation of spatial conflicts, while drawings do not (Eastman, C. 1975). The up-dating of the information contained in the drawings is made much easier, by the fact that they all originate from the three-dimensional model. This means any alteration of the building model would automatically be reflected in the drawings. Another substantial advantage is the prospect of performing building evaluation with such systems.

Last but not least, Negroponte suggests a third way of looking at human-machine collaboration in design. He suggests, in his book "*the Architecture machine* " (1970), a '**man-computer symbiosis**' to perform the design of buildings. This, he describes as :

> "...a machine that can follow your design methodology and at the same time assimilate your conversational idiosyncrasies..."(p 11).

He adds;

> " ... an architecture machine must understand our metaphors, must solicit information on its own, must acquire experiences, must talk to a wide variety of people, must improve over time, and be intelligent. It must recognize context, particularly changes in goals and changes in meaning brought about by changes in context.."(p 119-121).

There is no computer at the moment that can interact with men in this fashion. In fact, the above quote is very much like the description of the capabilities that one might expect to find in a human designers. This raises

the problem of comparing man with machine, as well as the questions of machine intelligence and machine learning capabilities. The nearest thing to such an ambitious vision could be what has resently emerged from Artificial Intelligence (AI) research, namely Knowledge Based Systems (KBS) and Expert Systems (mentioned earlier in section 3.1.1).
The argument in favour of such research (i.e.: in machine learning and machine intelligence) is validated by Simon (1983), saying :

> " Perhaps the deepest legitimate reason for doing machine learning research is that, in the long run for big knowledge-based systems, learning will turn out to be more efficient than programming,.."(p 36).

He adds :

> "... among the the most important kinds of learning research to carry out in AI are those that are oriented toward understanding human learning."(p 36).

As a means to understand human cognitive abilities, there is no doubt in the relevance of AI for design education, and possibly, for the development of knowledge based systems for design. Nevertheless, whatever this new development comes out with, the human control over the decision making process in design should never be given to a machine however 'intelligent' it might be. Simply because as computers are at the moment, they process information sequentially, whereas design problem solving needs to consider all aspects of a problem simultaneously (see section 2.2.3). This makes it most dangerous to rely on a computer for multi-variable problem solving.

In architectural CAD, the possibilities and problems of KBS for design have certainly been under investigation (Kalay, Y.E. 1985, Lansdown, J. 1982, Lansdown, J. & Roast, C. 1987). Some implementations were even done in logic programming languages like LISP and PROLOG to develop new CAAD applications (Swinson, P.S. 1982). These have not yet had much impact on the Computer Aided Design field, but their influence on future developments is almost certain.

The next section intends to identify the different ways in which computers are used in design, and underline the area on which the focus is going to be made.

4.1.2 Using computers in architectural design.

Cross (1977) reviewing what he calls the 'state-of-the-art' in computer aided architectural design of the 60's and early 70's period, groups the systems in four categories:

- Computer analysis.
- Computer synthesis.
- Computer evaluation.
- Integrated system (analysis + synthesis + evaluation).

This categorisation is still very visible in the large spectrum of existing CAD products, as well as in the research and development of new systems.

In a resent review and evaluation of CAD systems for the construction industry, carried out by Hamilton, I. & Winterkorn, E. (1985) for the Construction Industry Computer Association (C.I.C.A.). It has been shown that computer programs are being aimed at a large variety of subjects like : management, quantities and stock control, accounting, design graphics, structural engineering, services engineering, transport and communications. A similar review of computer systems used by the U.K. construction industry was undertaken by Gill, M. & Atkin, B. (1985), and shows that very few systems are specially 'architectural design' oriented, however, they are largely used by architects. These would cover applications such as office management project cost analysis and control, project scheduling and management, civil engineering, space planning and facilities management, computer aided drafting, as well as mathematics and statistics.

It is well known that the evaluation of CAAD systems is not a simple task. First, because it is difficult to find reliable and objective information on the suitability of a particular system for a particular use. Secondly, because the only way to know if a system is doing what it pretends to do, is to use it. The success of installing a computer system depending on the attitudes and capabilities of its operators.

Campion (1968) anticipated applications of computers in architectural design for management techniques, administrative tasks, design production and basic design. Similarly, Lawson (1985) asserts that the computer in the design office is to have four roles, in organisation and management, information presentation, solution evaluation and solution generation.

Whereas information management and information presentation computer tools (i.e.; Word Processor, Spread Sheet, Data Base Management and Computer Aided Drafting systems) have encouraged the introduction of information technology in the architectural design office, designers are still very rarely using the computer as a design partner. It is not until the computer is used for analytic and generative tasks that it becomes possible to envisage involving 'it' in the design process.

Reynolds (1980) suggests that the computer in design could be used in; an **analytical** way to check the viability of a proposed solution, and in a **generative** way, to produce a design solution.

- Generative design.

The use of computers as a means of automatically generating design solutions, has for a long time been the focus of CAAD research. The early applications consentrated on generating architectural layouts, a research area known also as 'spatial allocation problem'. Some of the first attempts were oriented towards the creation of a floor layout that minimise a certain objective function. Whitehead, B. and Eldars, M.Z. (1964) produced a program to design a single story building so as to minimize the traveling time of the occupants moving between spaces. This was based on an optimization approach, of which most algorithms require too much computer time, and trivialize complex architectural problems (see section 3.2.3).

Another approach to generative design, focused on the search for the most feasible solutions that responded to a given set of constraints without a particular objective function. The constraint satisfaction approach is limited by the fact that design criterias could not all be computationaly represented.

It was followed by an approach that uses algorithms to automatically generate 'all feasible' solutions, and then allowing the architect to choose the best alternative. Due to the complexity and size of algorithms used by this approach, many attempts to use it were limited to architectural problems composed of fewer than 16 entities (Shaviv 1987). Making little difference with manual floor plan generation, since small projects are not difficult to design.

In architecture, the design is generated in response to several requirements. Consequently, one would expect a design solution produced by a computer to comply with all these requirements, which is not the case with existing generative systems, because they only support a limited number of quantifiable design parameters. The design 'solution' generated in such conditions will need alterations and transformations eventually to fulfil all the design requirements as perceived by the human designer. A mention to some very resent work in this field will be made in section 5.11.

However, the few generative systems that have been developed so far, have put much emphasis on the production of 'a' design solution, mainly floor layouts (Armour & Buffa 1963, Whitehead & Eldars 1964). Whereas it should be the management of design constraints (i.e.; brief or programme) during the process of generating a design solution that ought to be given most attention, to help validate the produced solution.

- Analytical design.

Here analytical design stands for design evaluation. The evaluation of a design artifact is made with reference to design requirements, and by comparison to the performance of similar artifacts that have proved to have a good or bad behaviour in similar design conditions. When evaluating a design , the architect needs to know about the aims and motivations of such design, and have information or experience on the behaviour of similar buildings. To evaluate a building, five performance characteristics can be identified :

- Aesthetical.
- Spatial (Functional).
- Environmental.
- Structural.
- Financial.

Existing computer aided design tools have made buildings evaluation for aesthetical evaluation accessible to most designers. Whereas architects can now produce many kinds of views and drawings, they can still do relatively little about evaluating the structural, functional, environmental and financial performances of their buildings.

When generating models of buildings on computers, the only aspects that can be modeled are those which can be quantified, it is on these quantifiable elements that the evaluation will takes place. Generally, it consists of three operations; (1) measuring the model, (2) performing calculations on those measurements and then (3) comparing the results with targeted performances. The comparison stage of the evaluation is the most crucial, because judgemental decisions are made in the absence of non-quantifiable information. This last issue will be dealt with in the next chapter, with particular reference to financial evaluation.

The review of design methods and design methodology done in the second chapter, suggested that generative and analytical design should never be dissociated. Accordingly, there are arguments in favour of integrating generative and analytical computer aided design techniques. Shaviv (1987) believes that :

> " The combined use of computational generation and evaluation techniques will help designers find better solutions to the architectural layout problem than could be derived manually."(p 211).

He suggests that alternative design solutions should be generated automatically; according to design requirements formulated as constraints and as objective functions. Then, the generated alternative would be evaluated and appraised by means of automatic programs. Finally, the deficiencies of the proposed solution, identified by the evaluation, would be rectified through the <u>automatic or manual</u> generation of new and better design alternatives.

The above scenario is very attractive, in the sense it offers the possibility to design 'with', rather then 'on' a computer system. Although, it has the danger of placing the naive user in the situation where his/her design decisions are a series of 'reactions' and 'transformations' of what the system will have produced base on a limited number of design parameters. It is not clear wether the transformation of the model is done by the user, by the computer or by both. This is certainly a very important matter that deserve considerable attention.

In a similar line of thought, Lansdown (1986,1987) suggests the use of computer systems in design as 'prototype modification' instruments. His argument is that good designs are not created out of nothing, almost every design is a modification of some prototype that already exists in the designer's mind. He believes that :

> "The best designers are those who possess a rich supply of mental prototypes, and the skill to modify them to suit the task in hand."

According to Lansdown (1988) the computer aids currently available for designers are ill-conceived, instead of building blocks, they should provide prototypes which the designer can modify at will. He advocates the us of 'frame-grabbed computer images' as prototypes for computer aided art and design work, which he believes is a more 'productive' way of designing than starting work from a blank sheet.
This might be valid for artistic work, but probably not for architecture. Because a video image transferred to a computer will be at best a two-dimensional drawing, with no information what so ever about the functional, environmental, structural and financial characteristics of the

building (or building drawing) it was taken from. It is essential to have attached to any building model, to be used as a prototype, information on its individual components as well as contextual behaviour.

Whereas we have seen in this section that neither generative design nor analytical design methods have yet, separately, proved to offer much to CAAD users, at least in their current state. There could be something to expect from the combination of both.

For the sake of the following argument, we shall take on the idea of design being a 'prototype modification' process in Lansdown's terms. Where the building 'prototype' would be a three-dimensional building model, either computer generated or human generated. The computer aided architectural design process, would then consist of evaluating the design solutions by comparison with computer models of existing buildings (see Figure 4.1).

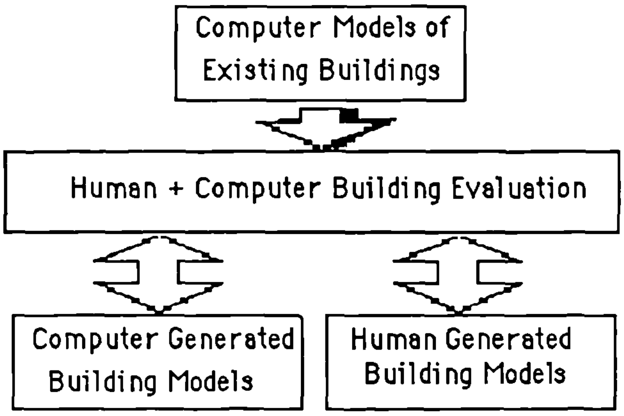


Figure 4.1 - Human/computer architectural design interaction.

Design transformations would then take place on the proposed solution by the human designer, with the computer assistance. This last argument will be taken further in chapter 5.

4.1.3 Ad-hoc, integrated and 'dis-integrated' approach.

Broadly, there are three ways to approach the development of computer based design systems. The first two were clearly articulated by Lansdown (1969), and have been largely in use. The **ad-hoc** approach, and **integrated** approach both have advantages depending on the situation in which they are used.

- Ad-hoc approach.

In the ad-hoc approach, a number of programs exist for tackling specific, limited-scope design problems. Independent design programs will each access a number of data base files of parts, components, technical data and geometrical data generated by the user. For example, thermal evaluation, day lighting or cost modelling will each require a set of the data representing the computer building model. This involves usually a tedious data input, and is consequently an error prone task. It is some times possible to transfer data from the building modelling system to the appropriate evaluation package through specific links, which tends to be a complicated task. Although, this is less likely to be a problem with new computer networking and files transfer handling facilities. This approach represents a simple way of performing very limited building modelling and evaluations.

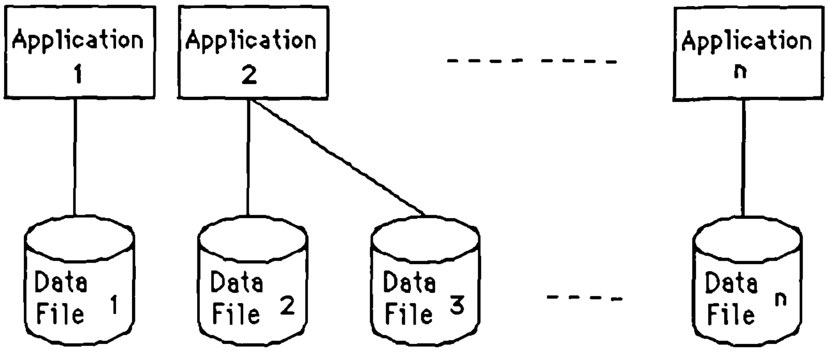


Figure 4.2 - Diagrammatic representation of the ad-hoc approach.

- Integrated approach.

The integrated approach can be envisaged as a comprehensive building modelling system. It will benefit from the integration of two dimensional and three dimensional descriptions of the model, as well as the measurement and calculation roles of the system. It means that each individual evaluation study makes virtually no demands on the designer, since the model would have initially been described using editing techniques.

The main advantage of the integrated evaluation approach is that the designer can run a whole multiplicity of evaluative jobs on the same computer building model. For example the same design solution can be checked against building industry regulations and at the same time undergo a thermal evaluation, and/or a cost evaluation.

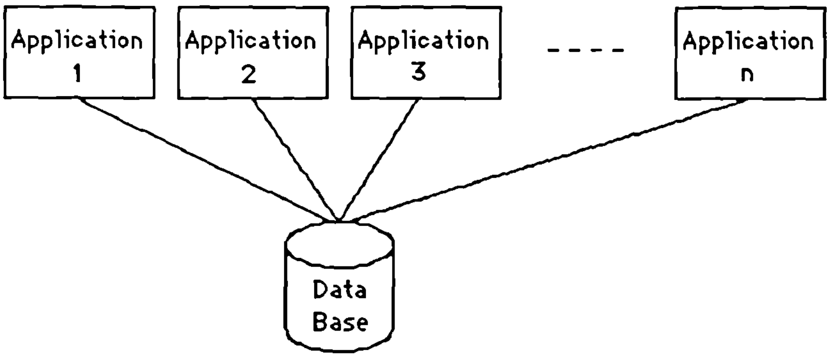


Figure 4.3 - Diagrammatic representation of the integrated approach.

- Dis-integrated approach.

More recently, Richens, P. (1983 and 1984) suggests a 'dis-integrated' modelling approach to computer building design. He argues that to integrate architects activities, it is necessary to dis-integrate their data.

It is intended to aid coordination and enforce consistency while enabling all members of the design team to work with the most recent information. This approach faces major data consistency problems, but it has advantages in giving each discipline control of its own data, and in working in just the same way when they are using separate computers.

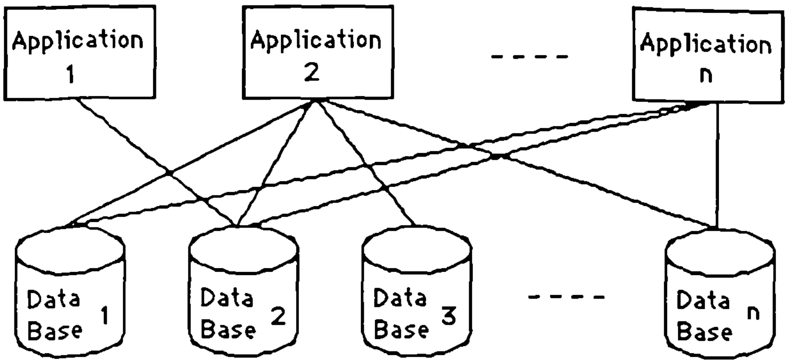


Figure 4.4 - Diagrammatic representation of the 'dis-integrated' approach.

Eastman (1978), an early advocate of INTEGRATED DESIGN DATABASES (IDD), argues that integrity and consistency problems of computer aided design can be dealt with IDDs. He believes that these will allow the definition of global goals and contextual conditions to be satisfied for a design, and later, during relatively unstructured design decision processes, the computer will automatically check decisions as they are made against the earlier stated goals, warning the user regarding inconsistencies and possibly making changes to other parts of the design so as to correct them. This last point could be a big constraint on the user of such a system. A designer must be able to describe his solution to the computer gradually as it develops, and the system must accept information in virtually any sequence and allow the designer to change any aspect of his design at any stage. However, Eastman goes on to suggest a general conceptual structure that should facilitate automatic integrity management, to which he sees an application at the analysis and synthesis stages of the computer aided design process.

Whereas the actual internal structure of a CAAD system matters very little if it is to be used for the organisation, management and information presentation of design, it is crucial for design evaluation. Any computer aided evaluation system must be able to converse with the designer during his possible highly integrated iteration of briefing, analysis, synthesis and evaluation (Lawson, B.R. 1981). This requirement has profound implications for the kinds of data structures and data organisation which such computer programs must establish. In this thesis, a combination of the integrated and dis-integrated data organisation, coupled with a sophisticated data structure are used, of which an extensive description will be given in Chapter 7.

PART B -
PROBLEM DEFINITION

Chapter 5 : DESIGN EVALUATION.

> "A design is a statement about values. Designing is inextricably bound up with evaluation".(p 1).
>
> March, L. (1976)

5.1 BUILDING DESIGN EVALUATION.

In the earliest notion of design morphology, introduced by Asimow (1962), evaluation was featured as an independent stage of design, taking place in a linear sequence of events. Markus (1967), was among the first to explore the ways in which measurement and appraisal can become part of the design method. He showed its relevance both as, a generative technique of use during the creative stages, and as an evaluative technique of use in more advanced stages of the design process. In describing his approach he suggested :

> "... appraisal is central to design for two reasons. First, because data from buildings-in-use are the means whereby what would remain a series of isolated and closed design projects becomes a single linked design activity, in which the connecting elements are a series of performance appraisals... Second, because the complete building-in-use is only one of many embodiments of a design concept, albeit the last, most detailed and precise."(p 1567).

He adds :

> "Models generated at earlier stages in the design have sufficient details for many of the performance appraisal techniques to be applicable. Thus, while appraisal presupposes earlier design activities of a generative kind, which have produced testable solutions, it itself is part of the generative process."(p 1567).

In other words, evaluation is assigned the position of an integral part of the design process, as against being the last stage in the architect's relationship with the building. The lack of emphasis upon evaluation as an integral part of design work is seen, by Gregory (1982), as due to the design techniques which are taught. These tend to concentrate upon synthesis and neglect evaluation; supposedly because synthesis is more spectacular or more mysterious, the expression of 'creativity'. It is true that in the language used about design, evaluation does not always appear.

Evaluation represents an attempt to find a value for a particular design arrived at by synthesis. Where there are multiple proposals the values need to be compared in some way. This comparison may refer to the foundations of the proposals, or attempt to examine their likely future consequences or anything else which seems likely to help in choice (Gregory, S. 1982). The actual use of evaluation and its derived results can broadly be categorised in three; (1) Evaluation before design, (2) Evaluation during design and (3) Evaluation after design.

Although our interest here is in evaluation during the design process, it is useful to describe the various uses of evaluation to clarify the problems of developing a computer aided design evaluation implementation.

It is suggested by Donald (1988), that evaluation research and its results can, (1) be fed-forward; where the results of studies of existing buildings are applied to the construction of other new buildings, (2) fed-in; whereby evaluation research is conducted for a building in the design and construction process, and (3) fed-back into an existing building designs.

The last use of evaluation belongs to the Post Occupancy Evaluation (P.O.E.) application, and will not be dealt with in this thesis. Although, the implications of CAD on such issues ought not to be undermined. Instead, we will concentrate on the feed forward and feed in uses of evaluation, respectively applied before and during design, and their interactions.

First of all, it is important to identify what type of information and knowledge is catered for, and in what form will it be fed forward or fed in. There are generally two broad sets of information which can be collected; (1) the details of a problem and its solutions, in relation to a particular building being studied, and (2) general broad type of 'ideas' which have the potential of use in future designs. It is this last type of information that could be suited for what Lansdown calls 'prototype transformation' design. However, when information has been fed forward, it may still be necessary to evaluate its adequacy within the new person-environment context (Donald 1988). A major consideration at this point is the form this information should take.

Several studies have attempted to discover how evaluation research information should be presented to the designers. Goodey and Matthew (1971) did a study on information flow in British architects' offices, and their preferences among presentation styles. Their recommendations emphasized brevity, clarity, visual illustration well coordinated with text and architectural vocabulary. This last point is very significant, and will be developed later in the thesis. More recently, Mackinder, M. and Marvin, H. (1982) looked at the interaction between the designer and information, and they report an unwillingness to consult written data, because this is seen as time consuming.

Lera *et al* (1984) argue that the nature of information presentation is not the core issue, because even if information is presented in a usable form, it will help very little, if the architect does not consult it in the first place. This means that throughout the process of designing, the choices that designers make (i.e.: their choice of consulting or not consulting information) are theirs, and are affected by their predispositions. These reflect the designers attitudes, values, and belief toward not only what is possible but about what ought to be done.

Many published accounts of the design process have emphasized the important role of value judgement in design (Daley, J. 1969, Dark, J. 1979, Lera, S. 1981). In an interesting book, Collins, P (1971), has sown the contiguities which seems implicitly shared by the philosophy of law and the philosophy of architecture, and the affinity between the judicial criteria of architecture and law. There is, unfortunately, no room in this thesis to deal with this subject fully. However, human ability to make balanced judgement remains the best instrument for design decision making. This 'talent' could be enhanced by supporting it with precise, reliable and rapidly accessible information.

There is no doubt in the fact that, communication and decision making in design evaluation are closely related design issues, and that the process of designing a building is a function of design team communication. Wallace, W.A. & Kelly, J.R. (1987) say in this respect :

> " Each member of the design team together with the client evolve the design through a longitudinal process of communication. The roles of the participants and the timing and configuration of their communication is obviously vital."(p 20).

It is not until the designer communicates, and by doing so gets informed, that he or she can make a decision concerning the value of a design solution. In the case of evaluation before design, the information consulted will describe an existing building. Whereas, when evaluating during design, the information used will describe a design model.

The information gathered by the designer will be used to **compare** and **contrast**, two essential ingredients of evaluation. For evaluation prior to design, the comparing and contrasting will involve on one hand the building, and on the other hand the level of satisfaction (or unsatisfaction) it provides to its users and to the environment. Whereas, for evaluation during design, comparisons will be made between the design model and the objectives aimed at by such design. In both cases it is specific building characteristics that will be contrasted. These are multiple, some of them are quantifiable and others are not.

Since our aim is to involve the computer in building design evaluation, the next section will differentiate between quantifiable and non-quantifiable building design characteristics.

5.1.1 Measurement and appraisal in design.

The measurement and evaluation of building performances is a full part of the design method and we shall now identify what types of building performances and which quantifiable characteristics are the most suitable for computability. In an attempt to define systematic routines and techniques for design evaluation, Markus (1967) suggested four main types of activities that ought to be undertaken; identification, finding relationships, model building, and optimization.

- 1. IDENTIFICATION.

This consists of identifying design constraints. The design constraints were clearly defined by Lawson (1971 and 1980), who differentiated between domain constraints, function constraints and generator constraints. These are three sets of relations between the variables of the architectural design problem. Organising them is a very difficult task for the designer, since he must decide what to take on trust, what to question and what to reject. Unfortunately, systematic constraints identification routines and techniques, which could help designers considerably, have been given very little attention. However, recent research developments at the M.I.T. (Groos, M.D., Ervin, S.M., Anderson, J.A. & Fleisher, A. 1987 and 1988), is showing some promise in this direction.

For the time being, the identification of constraints will be entirely left to the designer, who might use the computer to store and organise information as well as relationships between the different constraints.

- 2. FINDING RELATIONSHIPS.

To find design relationships, Markus (1967,1971) suggests a conceptual model called the Building-Environment-Activity-Objectives system. The system has five main parts :

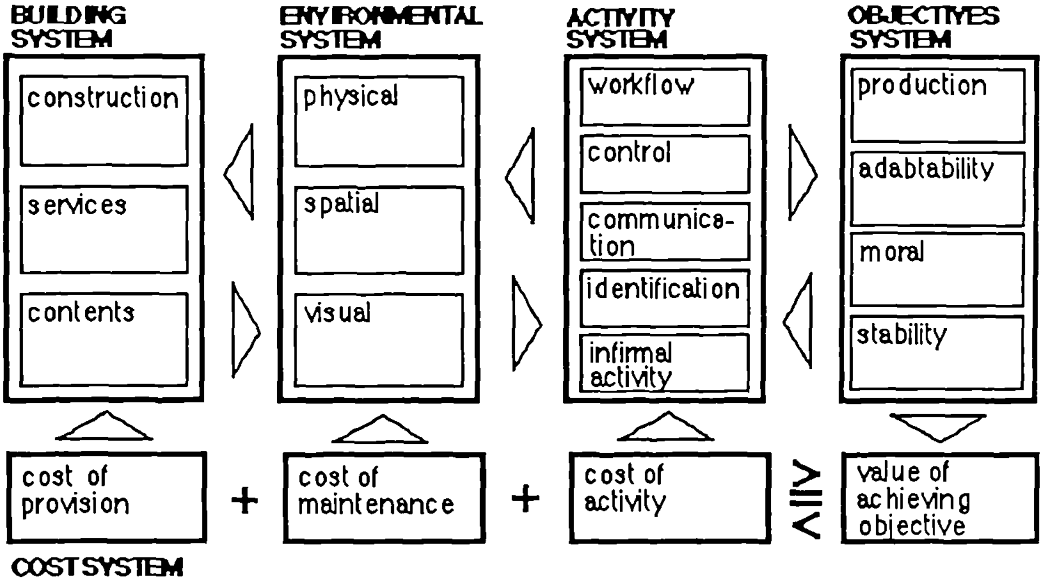


Figure 5.1 - The *Building-Environment-Activity-Objectives* system (Markus 1971).

This system is considered to consist of people and objects interacting in a complex way (see Section 3.2.1). The 'objects' are the fabric or hardware that generates the environment (natural and/or artificial). Whilst, 'people', assumed to be goal oriented, use the environment to achieve objectives of an idealised kind.

i- The building system;

The building system includes all concrete items, components, assembly and objects that are part of the building. Its complete description gives not only all the physical characteristics of these parts but also their dimensions and relationships. This information is normally described, in a very explicit manner, in a set of drawings, specifications, and bill of quantities. Although, some of the relationships between elements are sometimes implicit in the drawing representation technique and/or terminology used by the building professionals. Few architectural CAD systems do handle at the moment the information required to describe Markus's building system, and it is interesting to quote him saying.

> "Much of today's interest in automatic data processing and handling of these documents (drawings) misses the relation that the characteristics of a building must not only be able to be described unambiguously but must also be quantifiable if worth while predictive relationships are to be obtained."(p 1571).

Here, the emphasis is put on the quantifiability of building elements to predict relationships, which tend to suggest that if a building element is quantifiable, it is subsequently easy to relate it to other building elements. This assumption is based on the fact that relations are a straight-forward matter, when implicit in drawings. In the implementation of CAAD systems, this last point has proved to be a considerable challenge. However, when over come (which will be described in chapter 7). These still remains the problem of relational integrity, a matter of major importance when trying to transform a building model for interactive evaluation.

ii- The environmental system;

The environmental system has three main sub-systems, the spatial, the physical and visual. It is to insure a proper functioning for the different patterns of activity housed in the building, as well as modify the external climate to provide a controlled internal environment within which the activity system can develop. The relationship between these sub-systems is particularly complex and the elucidation of this relationship is central to the understanding of building and environmental design.

iii- The activities system;

The activities system sets up and maintain an activity pattern consisting of production, communication, informal behaviour, identification, control, work flow etc... A combination of those can some times, if identified by the designer, constitute a 'radical' constraint in Lawson (1980) terms. For which a 'radical' solution has to be found, to solve the design problem in its totality.

iv- The objectives system;

It consists of the long term aims for which the organisation exists. The objectives provide the context for all the activities and consequently for the buildings and environments. Markus (1971) point out that

> "Often an organisation's objective is in conflict with broader, social objectives (e.g. industry and pollution) or with narrow personal ones (e.g. production and friendship formation). Such micro and macro conflicts are inherent in all orgenisations and the designer had to understand them and adopt priorities. Often his own objectives will cause further conflict."(p 85).

The objectives give rise to the activities which it is necessary to implement in order to achieve those objectives. The designer's role is, partly, to reduce conflicts between the different objectives, and in doing so, he will introduce his own aims. Lera (1982), identifies the designer's aims as originating from; (1) self-imposed goals and (2) precedent knowledge. The first are part of the designer's personal values and opinions, like style and ideology. Whereas the 'precedent knowledge' consists of, on one hand stereotypes and patterns, and on the other hand information on previous buildings (i.e.: published information on designs or existing buildings). This last point has some significance to this research work, in the sense that it links the 'objectives' that are the beginning and the end of the whole system, to external previous experiences of the designer that are essential for the evaluation for design appraisal and evaluation.

v- The resources and cost system.

Each of the previous systems has an initial and/or continual cost or value. The building system costs to provide design, construction materials, labour; the environment system has costs of energy maintenance and cleaning; the activity system consumes resources in the form of wages, salaries, the materials it uses (or wastes), hence the growing need for facility management computer systems in modern services society.

If the system is not to run at a loss, the value of achieving the objectives must not exceed the combined cost of the building, environmental, and activity systems. The difficulty of quantifying values in terms of money or other units commensurate with cost, should not over shadow the need for adopting cost benefit analyses for solving problems related to the building during the design process. The way to appraise any design concept is to embody its characteristics in a testable model.

- 3. MODEL BUILDING.

Markus (1967) defines the building model as :

> "...nothing more than a representation of an abstract concept in more or less concrete form- symbol, mathematical relationships, schematic diagram, detailed design drawing, three dimensional model or a network simulation system. In any of these, design variables (design parameters) can be changed and the consequences (dependent variables) can be measured."(p 1572).

In making models of buildings, the only variables that can be modeled are those which can be quantified. Hence the importance of measurement in the appraisal of building performance. A building model can be seen as a representation of a system in which the relationship of parts are fully understood. The modelling of the building system described above, would for instance consist of representing abstract construction materials, and the relationship between them. For example, walls, floors, ceilings, windows, and doors would be independent entities, with the relation of walls 'to' ceiling and window 'to' wall explicitly defined. In the case of computer building modelling, each of the building system elements will be associated with a geometrical element, that will have the similar properties to the real world element it represents. For example a wall will be represented by a surface (usually vertical) of various thickness and height.

The measurement and appraisal of the resources put in a building system, will consist of measuring the geometrical characteristics (i.e.: length, width, area, volume, etc...) of all the building elements used to model the building on the computer. Each element having a unit cost (i.e.: cost per-square metre, or per-cubic metre, etc...) and a labour cost. The calculation of the total building system cost becomes a simple matter of adding up the elemental cost. However, the building system is not the only element of the built environment represented in the conceptual model. We have seen above that the environmental and activity systems have both an initial and/or continual cost. These ought to be taken into consideration in building modelling and evaluation. For instance, the environmental system needs to distinguish between 'in-side' and 'out-side' for thermal and visual performance evaluation. Also, it must distinguish between spaces of the building to enable the appraisal of proper functioning for the different patterns of activity. This is an essential information for the activity system evaluation.

It is clear that computer model building system, for a comprehensive representation of resources, ought to incorporate both elemental and spatial modelling. The measurement and appraisal of a building design will not only enumerate the elements of the building fabric in congruence to there constructive role, but also according to the environment and/or activity they have in common..

- 4. OPTIMISATION.

Optimisation is the last appraisal activity, it consists of selecting the 'best' design from any number of possible solutions. The points outlined here will cover only the quantifiable aspects of design variables. Imponderable factors, like aesthetics, will not be included in this analysis. Markus (1967) argues that without a single criterion, alternative solutions for parts of the design can never be combined nor the best overall solution be found. According to him, it is meaningless to ask; "Which is better, a good lift service or freedom of glare ?". But it is quite meaningful to ask; "Which of two lifts design is better, or which of two lighting installations produces less glare."

From this single standard evaluation approach, 'cost' suggests itself as the best criterion for optimisation of building design evaluation. The objective being to produce design solutions of maximum value at minimum cost. The argument could as well have been optimising comfort or satisfaction in terms of physiological or psychological response, or optimising a particular activity like circulation (see section 3.1.2). Cost is simply a convenient scale of measurement, and the considerable computation involved in calculating the performance of an outline design, makes it, by the use of fast computers, feasible to consider the implications of many design alternatives. The balancing of architectural objectives is certainly very much influenced by the prime cost of a design. However, money is not necessarily more objective than comfort, desire, sensation or satisfaction (Markus 1970). Under different conditions (eg: high inflation, poverty, economic boom, etc..), judgement of the value of a certain increment of money does change considerably.

From a methodological point of view, the practice of optimisation, using statistical decision theory for optimising behaviour, is getting largely replaced by the 'satisficing' behaviour management approach. Simon (1981) argues :

> " ... real-world business firm turns to procedures that find good enough answers to questions whose best answers are unknown. Thus normative microeconomics, by showing real-world optimization to be impossible, demonstrates that economic man is in fact a satisficer, a person who accept "good enough" alternatives, not because he prefers less to more but because he has no choice."(p 36).

In the current condition of domination of the market systems, against the state systems, it is recommended to the architect to learn from the economics man. Therefore, it is not believed that cost on its own could justify the acceptance of a proposed design solution. But that among other design constraints, the cost parameter should be well handled (i.e.: 'closely monitored with the rest of the design constraints), so it can help make the 'best' choice.

5.1.2 Inquiry and decision.

It is through the process of decisions that problems are solved and action taken. It is argued by O'Shaughnessy, J. (1972), that the steps that in such decision-making -establishing objectives, identifying alternatives, discovering consequences and finally making a choice-, are composed of the elements of description, explanation, prediction and evaluation.

- Description.

A description is regarded as a factual statement, because it is usually based on simple direct observation where no attempt is made to change what is being observed. However, observation can be erroneous. For it is selective and depends on interpretation of sensory stimuli. Such interpretation involves giving meaning to sensations by organising them through theory and concepts.

- Explanation.

Explaining the origins of a problematic situation offers guidance in deciding appropriate action. To prevent the repetition of a problem, then it needs to be explained. It seems that re-stating or re-formulating a problem can enable the recall of ideas and concepts not aroused by the original statement. Explanation seeks to find the conditions that are functionally related to events.

- Prediction.

Prediction refers to future events, as against explanation which argues about the nature of current or past events. There is an isomorphic relationship between explanation and prediction, since to be able to explain it is to be able to predict correctly. A prediction results from some hypothesis, and the validity of that prediction is related to the probability with which the hypothesis is true.

- Evaluation.

In an evaluative decision, which is what designers continuously do, the decision-maker is concerned with the decision whose immediate goal is to rank the value of a set of items according to some criterion. Its aim is to determine relative worth or contribution to objective.

In evaluating the cost of a building, the designer has to undertake the above operations several times and in various sequences. The next section will describe the main techniques devised for building cost evaluation.

5.2 BUILDING COST EVALUATION.

> " One major task facing building research workers is analysis of cost data to discover systematic cost variation with changes in design parameters. Without this, cost analysis will remain a checking tool and real *design* with costs cannot take place."(p 1571).
>
> Markus, T. (1967)

In the procedure of the implementation of a building scheme, the architect usually rely on the quantity surveyor to supply cost information to plan the various operations required for the execution of the construction. The cost information is normally based on the actual costs of previous buildings of similar type, suitably adjusted to make allowance for such factors as difference of location, site conditions, market conditions and quality of works. The costing exercise is called 'cost planning', and prepares a provisional estimate on a comparative or interpolation basis.

5.2.1 Cost planning and cost modelling.

The general construction preparation procedure requires the quantity surveyor to interact with the architect from the outline proposal stage of design. As soon as design drawings are produced, the quantity surveyor is in a position to give general guidance on cost, and evaluate the financial effect of different solutions. From this phase an outline cost plan is usually prepared, which will considerably influence the final potential building cost. At the scheme design stage, the quantity surveyor checks his approximate estimate figure and, with the aid of extensive cost information, re-appraises the initial cost plan with provisional target cost figures allocated to each element or major part of the building. Later, when the design is more detailed, comparative costs of different forms of construction, material and services layout are provided by the quantity surveyor. He will adjust the distribution of cost in the cost plan, and include running and maintenance costs if they have significant effect on the outcome. Continuous checks will ensure that the development of the design remains compatible with the cost plan.

According to Seeley, I.H. (1972 and 1984), there are two basic methods of cost planning currently in use; elemental cost planning and comparative cost planning, although in practice variations of those have been introduced.

(1) Elemental cost planning.

This method uses approximate methods, such as cost per place (i.e.: per occupant), cost per square metre or cost per gross floor area. The building in broken down in various elements of construction or constructional parts, such as walls, floors and roofs, and each element is allocated a cost based on cost analysis of previously erected buildings of similar type. The sum of the cost targets set against each element, must not exceed the total estimated cost.

(2) Comparative cost planning.

The comparative system starts from sketch plan, but does not use fixed budgets like the elemental system. Instead, a cost study is made showing the various ways in which the design may be realised and the cost of each alternative considered. The cost study is usually based on approximate quantities and constitutes an analysed estimate. In Seeley's (1972) words, comparative cost planning ;

> "...is not necessarily to show how cheaply a building can be produced but to show the spread of cost over various parts of the building and what economics are feasible."(p 136).

He adds;

> "The comparative system endeavours to show the architect the cost consequences of what he is doing and what he can do. It shows the effect of choice of design for one component of the building on other."

The hindrance in using comparative cost planning, is the difficulty of breaking down the building into parts, and the handling of that information for comparative cost analysis.

For cost advice at early design stage, '**cost modelling**' seems to be a more flexible instrument to analyse cost variations. It is often mentioned in conjunction with the use of computers, although it does not necessarily depend on them. According to Ferry, D.J. and Brandon, P.S. (1984) cost modelling may be defined as :

> ' ...the symbolic representation of a system, expressing the content of that system in terms of the factors which influence its cost.'(p 101).

Seeley (1984) gives three examples which illustrate the wide range of investigations and activities for which cost modelling can be suited. He describes Brandon, P.S. and Moore, R.G. (1983) computer programs on building appraisal. These have constructed an elemental cost analysis data base, from which information could be retrieved and adjusted for new project.

Similarly, Smith, G. (1980) has devised an approach whereby the design target parameters and cost targets are set out and compared with those contained in the budget model. This is a quick way of assessing the chosen strategy at the early stage of the design.

Finally, he refers to Mathur, K. (1982) cost modelling approach. This is seen as being very global, as building forms are seen as the single most important and strategic decision made at early stages in the design process. It determines the cost of superstructure, services and operation of the building. This approach postulates that if a suitable data base was generated which relates compactness of form, cost elements and total cost, then an early decision on form could provide a reliable cost prediction.

However, it is generally thought among quantity surveyors that the increasing use of computers has the potential for vast amounts of cost data and design parameters of previous projects to be readily accessible and used in the design and cost planning of new buildings. A system that would support the elemental modelling of building components, and would provide historical cost information for such elements, holds obviously considerable promise in the field of cost estimating and cost modelling.

5.2.2 Design costing.

Design costing represents an attempt to put costs against a building design solution. It operates at various stages of the design process, and thus requires different techniques depending on the amount of information available. Estimating methods can be associated with building modelling techniques. Beeston (1987) has suggested that they may be classified as 'descriptive model methods', 'in-place quantities model methods' and 'realistic model methods'.

Descriptive models method.

This approach is based on the idea of attaching cost to descriptive features of a design rather than its building materials. Descriptive models use formulae in which the variables describe the design and its environment by such key factors as size, shape, type of construction and location.

In-place quantities models method.

The in-place quantities method could be described as a building material quantification based approach. It usually assumes that there is a fixed relationship between design variables and the material cost. Each item of building material used in the construction is associated with a unit cost. This method requires a great deal of detail. and is therefore limited to late design stages. Increasing the detail and complexity of quantity-based methods does not necessarily produce greater overall accuracy. This approach is inadequate as a design tool since modern business methods make increasing demands for accurate financial forecasting and control at early design stages.

Realistic models method.

Realistic models are intended to improve estimating accuracy by reflecting the effects of the constructional process on the cost. For example, brickwork will be costed differently at higher levels of the building because of the overheads required to raise scaffolding and lift materials. Although apparently more accurate this technique depends upon having details of the whole construction management process, which are not usually available at early design stages.

Emphasising the importance of cost modelling to take place in the initial phases of the design process, Atkin, B. (1987) writes :

> "Is is at early design where the quantity surveyor can exercise the most influence and thereby provide a better service to clients. Once the early stages of design have passed, the design and, therefore, its cost are virtually *frozen*."(p 60).

Our aim being to allow cost modelling to take place at the early stages of the design process, it seems appropriate to adopted the descriptive model method of design costing. At this point two problem areas in the use of descriptive models can be identified. The first is the method used to produce raw data on the design variables for the model. The second being the actual performance of cost analysis as a way of exploring cost situations. The latter is not going to be dealt with in this thesis, since matters like regression methods for econometrics and cost data comprehensiveness are beyond the scope of this research. Nevertheless, an original approach is envisaged for the production of quantitative data on the design solution using an 'elemental cost technique'.

Chapter 6 : COST MODELLING IN C.A.A.D. ENVIRONMENT.

> "...CAD systems have much to offer the entire design and construction team by acting as a focal point for integrating all information on a project. As the role of the QS in management develops, so too should an understanding of the tools and techniques by which the design process might be managed more efficiently."(p 13).
>
> Atkin, B.L. (1986)

6.1 COMPUTERS IN QUANTITY SURVEYING.

It is generally accepted that, decisions taken in the early design stages strongly influence the final building cost. Our aim here is to assist the design team in producing buildings which are cost effective from the point of view of initial cost, running cost, maintenance and efficiency of use. To achieve these objectives, appropriate design tools need to be made available to designers.

In developing computer aids for building design evaluation in general or for building cost evaluation in particular, two main approaches can be envisaged; (1) the implementation of computer programs that mimic routine processes that exist in the traditional practice of the design evaluation work, or (2) the development of new methods that take advantage of the information processing power of computers and the logic that can be built in the information processed.

In the early 60's, the increasing use of machines for accounting, statistical and similar purposes, by industry and commerce, has encouraged quantity surveyors (QSs) to examine their application to quantity surveying practices. Most implementations consisted of reproducing existing techniques and methods of bill of quantities production.

The following section will describe these early implementations, and will be followed by a proposal for an integration of 'state-of-the-art' CAAD techniques to building cost modelling and evaluation.

6.1.1 Bills of Quantities automation.

Since the first developments of electronic computers, Quantity Surveyors have been interested in the use of machines for the automatic production of bills of quantities. The preparation of bills of quantities traditionally consisted (and still does in most cases) of four distinct operations ;

(1) Taking-off : The measurement of units of material and labour from architects drawings, by a standard method.

(2) Squaring : The multiplication of the dimensions of the taking-off to obtain areas, volumes and linear dimensions.

(3) Abstracting : The process of collecting like items and putting them into the a standard order, and converting them into the units to be used in the final bill.

(4) Billing : Transferring information from abstract sheets to final documents, the bill of quantities.

The last three operations are almost entirely routine processes, and consequently, lend themselves to mechanisation. One of the first to advocate the mechanization of the "working-up" process (i.e.: the last three operation), was H. M. Stafford (1957). He emphasised the need of standardized quantities for mechanized billing.

The "working-up" involves operations which can be reduced to the five following functions : calculating, sorting, classifying, summarizing and recording (Dent 1964). All these basic clerical and accountancy procedures made, in the late 50's early 60's, an increasing use of machine accounting in such fields as banking, sale analysis, pay-roll preparation, stock control, and the like.

It is reported by Lancaster Britch, A. (1963) that, in 1958, a firm of charted surveyors approached Ferranti Ltd to investigate the use of electronic computers for the production of bills of quantities. The promise given by the preliminary investigation encouraged the creation of one of the first firms in the U.K. to exploit the new technology commercially. 'Computaquants Ltd' was set up, and started offering a service to quantity surveyors, who needed only to send along taking-off sheets to obtain fully processed copies of the bill of quantity.

During the same period, more ambitious work was going on in the USA. Where an analysis system that would permit comparative assessments of cost and benefit at every design stage from schematics to working drawings was under development (Barnett, J. 1967). This research project used the 'Sketchpad' program (Sutherland 1963, Barnett 1965) as a graphical input interface to the computerized cost estimating system. What it basically tried to do was to automate the 'Taking-off' operation for preparing the bills of quantities. It is interesting to note that this research project was initiated and carried out by architects, and is described by Barnett (1967), saying :

> "The computer programs follow the way in which an architect would naturally work, providing a method of evaluating his own designs, rather than attempting to automate the design process. The more typical computer-oriented approach would have been to produce some kind of optimization program that concentrated on generating the cheapest possible building, to the exclusion of everything else."(p 166).

This project was unfortunately given little attention, and ended up being used only by the small architectural office that developed it. Possibly, because it was a head of its time, in the sense that computer technology had not yet matured, and did not enjoy the acceptance it has got now a days among the architecture professionals.

As computer technology evolved, the techniques for mechanized billing improved. For instance, punch card techniques were replaced by interactive keyboard input on screen monitors. More recently, digitizers are used to take-off measurements from hand made drawings, or less frequently, measurements are derived from CAD systems. However, the largest use of computers made by QSs is for cost estimating and cost planning. There are currently many cost estimating packages available, but they all perform cost analysis in different ways. The reason for it, is that they were developed by QSs trying to automate their own mechanised procedures of cost analysis. Lancaster Britch, A. (1963), had foreseen such situation and identified two different directions in which machine billing could develop :

(a) through individual and unrelated efforts, or

(b) through some centralised body initiating research and development.

It is obvious that the first approach was the most successful, if one may call it a success.

The architecture profession has usually turned to quantity surveyors for accurate and timely advice on the cost of construction during early design stages. The problem of bill of quantities production automation, has been addressed by both architects and quantity surveyors. Unfortunately, it some times, was perceived as a threat to the 'convenient' distribution of work among specialised professionals, especially QSs.

Whereas it is generally the quantity surveying profession that fears loosing some of its work, by the computerisation of bill of quantities production operations, there are, surprisingly enough, claims that automating measurement functions is more threatening for the architects, then the QS (Curran, J 1987).

We have no intention here to get involved in the polemic that surrounds the issue of bills of quantities automation, and its professional consequences. Our aim here is to show how it is possible to develop an automatic system of computer aided building cost modelling, which uses data directly from a CAD system and obviates the need for manual data entry. Such development has given rise to new expectations and opened new horizons for the integration of CAD and cost modelling. The next chapter will describe in some detail the work undertaken.

6.2 INTEGRATION C.A.A.D./BILLS OF QUANTITIES.

Computer based cost modelling has always held out the prospect of a good feed back on likely building cost at the early stages of design. It is an attractive prospect to all members of the building design team. However if that program can in turn be fed with data directly from a system actually being used to design the building then at least two important advantages can be obtained.

Firstly, one of the potential areas of human error has been removed since most of the data required by the cost modelling system is obtained without the need for manual entry. Since the CAD system itself reveal any error in the form of unexpected geometry in perspectives and other views, it seems unlikely that this data will contain any errors of significance in cost terms.

The second advantage of this integrated approach lies in the ability to design as opposed merely to estimate. In this regard it is important to consider not only capital cost but all the other criteria the design may have to satisfy. With an integrated CAD system it should be possible to explore design alternatives in all their aspects and to be able trade off advantages and disadvantages. The isolated cost modeller may thus present a useful tool for the quantity surveyor faced with the task of estimating the cost of a fixed design. The integrated CAD system and cost modeller may allow that quantity surveyors to become a more active member of the design team, suggesting and exploring design changes, which may immediately be modeled not just in cost terms but also be seen in the CAD system and studied for all their implications (Belhadj, T.A. and Lawson, B.R. 1989).

6.2.1 CAD/BQ system integration.

Two main approaches can be envisaged to the way quantity surveyors can exploit the inherent measurement capabilities of CAD systems. First, by using a dimensioning, scheduling and reporting approach. Which would consist of extracting relevant information from the CAD system files, and then transfer them to a database. Second, by integrating computer aided building modelling and cost modelling.

The first approach consists of creating lists of building parts from a drawing. It uses the specification and description of elements designed with the CAD system, and then creates materials schedules. This approach tends to promote links between CAD and other packages Which could either be standard database management or spread sheet software, or dedicated cost estimating and cost planning applications. A project, currently carried out at the 'Construction Computing Centre' of Napier college in Edinburgh, attempts to achieve such a link by using a commercial CAD package, where components (eg: walls, floors, etc...) are identified and automatically measured, then their dimensions are transferred, again automatically, into a spreadsheet (Curran, J. 1987). Similarly, Ashworth, G.N. & Wilson, R.A. (1988) described what they call a CAD/BQ (Bills of Quantities) integration consisting of linking a large commercial CAD system to a dedicated estimating program developed, on request, by a quantity surveying practice. These stand-alone programs are some times very good at doing their own job, but do not make full use of the modelling capabilities that the CAD system has to offer.

The second approach uses 3D models as the basis for describing the geometry of the building, with elements sufficiently defined to support later measurement and evaluation. This means that the building modelling system will require considerable non-geometrical information introduced before and during design. This is to enable the production of bills of quantities without further manual intervention. The measurements in this case are passed to a built-in module for cost analysis and evaluation. The latter will be fine-tuned to the capabilities of the computer aided building modelling system. The claim for integration in this approach, is not just for having brought together two programs in one application, but to have enabled two design tasks to take place almost simultaneously on the same set of data.

The idea that the data required by a cost modeller could be extracted from a three dimensional CAD system does not in itself seem extraordinary. However, it does presents quite difficult problems unless the CAD system itself is able to perform a certain level of 'intelligence'. To examine this idea it is easier to move towards the kind of CAD system needed in three steps.

The most basic CAD systems are really simple two dimensional drafting systems. Here the data held by the computer is entirely abstract and uncoordinated. The lengths, areas and volumes of items cannot be obtained from any one drawing of the building, and unfortunately such systems cannot relate the various drawings in order to obtain three dimensional data. At its most crude level however, a link to a cost modeller could be made in which the user points to graphical items within drawings, explains what they are and leaves the system to compute some basic geometrical data. Unfortunately this does not really show either of the advantages of integration as discussed above. The scope for human error and omission is still considerable and there is no real provision for interacting with a design. It would simply have to be redrawn and all the work done again. In any case it is unlikely that a building would be entered into a drafting system until the design was already reasonably well developed.

The second kind of CAD system we might envisage is a three dimensional modeller. Here all the major features of the building would be modeled and their geometrical data could be extracted. However unless the computer also 'knows' what each item is in building construction terms this data must still be individually explained to the computer. In such a system even the simple distinction between internal and external walls may be missing.

This then brings us to the idea of an 'intelligent' modeller specifically designed to hold data about buildings. Such a modeller must clearly 'know' what each element is (i.e.: wall, window, door, etc...), how it is constructed (i.e.: bricks, concrete, etc...), and how the elements are interconnected. It has been concluded that this leads to the separation in the system of three quite distinct types of data which were called locational, specificational, and relational (Lawson 1981). The separation of these kinds of data enables the designer to make changes which are understandable by the computer in building terms.

To illustrate these principles we shall consider a simple door. The specification data here would almost certainly describe the size and shape of the door as well as its construction, and this data will be common to all doors of that type in the building. The locational data will describe where one particular instance of the door type is, while the relational data is likely to describe the wall in which the door is an opening, the direction in which it faces and the room or rooms into which it looks. A detailed description of the data structure used will be given in section 8.1.2., but it is relevant to this discussion to note that specificational data varies in type with building elements which are categorised as one of three kinds. There are 'units' such as doors and windows, 'surfaces' such as walls and floors, and 'edges' such as eaves. These categories vary in their ability to contain geometric data within the specification. Described, for example, by its specification, the 'surface' may only contain cross sectional data leaving each instance to contain the remaining size and shape description within the locational data.

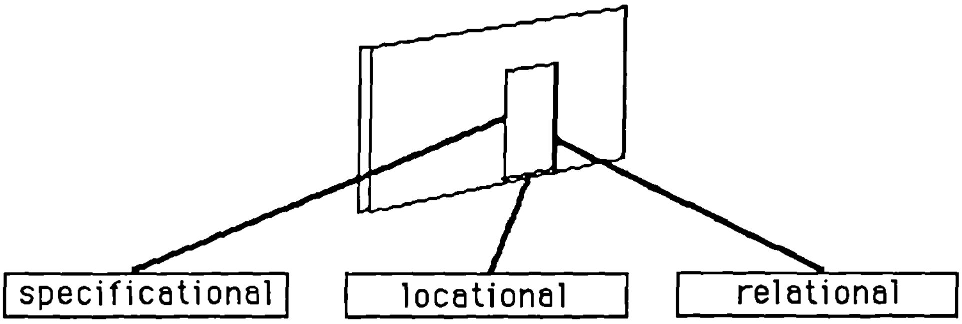


Figure 6.1 - The GABLE BMS three basic data representations.

The GABLE 4D (Graphical Aids for Building Layout and Evaluation - 4

Dimension) series building modelling system which was originally researched at Sheffield University and developed by GABLE CAD System Ltd. follows these general data structure principles. In this system, the relational data is automatically developed by the computer using a special series of routines which so analyse the pattern of building elements that a room by room model of the building is established in addition to the elemental model. This enables the designer to extract data about the building in many ways. For example data can be listed by element, by floor level, by room, by orientation, or any combination of these. So, for example, it is possible to ask for the areas of all windows of a particular type facing north on the first floor or above which look into a particular type of space.

This effect is achieved in GABLE by performing a survey of the building in which all quantities are automatically measured and, together with specificational references and relational descriptions of each item, this data is streamed into a series of data base files which are then accessible to the user (see Section 7.1.4.). The specificational data is also held in data base files of the same format so that cross reference between quantities and specifications is possible. It is these data base files which are used as the 'link' not only to the cost modeller but also to other evaluative routines such as those which provide heat load and solar gain calculation. The data in these files can be searched and reported upon by using normal data base type enquiry and report formatting techniques, thus providing very flexible scheduling facilities. There is also a provision to display this data in graphical form by creating business graphics type files in the GABLE drafting system (see section 8.3.2). If required, these schedules and diagrams can be combined with the original building plans and plotted together.

A typical sequence of events during early design stages using the GABLE system would thus be as follows. The designer would explore the initial design form, and an 'intelligent' three dimensional building model would be developed. This model in then automatically surveyed creating a database of use to many evaluative routines. In turn, this data is used by the cost modeller routine to create a 'Floor Fundamental Quantities' file in which each record describes data about a particular floor level. This is effectively the descriptive model used for costing.

This technique allows editing of the data at any stage, but most usefully for cost modelling purposes, using the 'Floor Fundamental Quantities' file. Here, experiments on such descriptive features as the floor to ceiling height, floor areas and the like can offer the 'DESIGNING TO A COST' facility describes earlier. Of course the 'COSTING FOR A DESIGN' approach is satisfied by the appropriate selection of cost rates from the historical data base of previously modeled buildings.

So far we have demonstrated an integrated approach from CAD to cost modelling. However, the techniques developed have the further potential of offering 'space based' cost modelling. Space based cost modelling aims to draw a parallel between space modelling and cost modelling. It could take place on either a descriptive model or an in-place quantities model. Basically, it consists of grouping building elements that belong to the same spacio-functional unit of the design solution, and permits the identification and location of cost in the building layout for cost estimating within the early stages of the design process.

6.2.2 Space based building measurement.

The idea behind space based cost modelling is to add an extra informative dimension to the cost of a design solution, this being the 'location of cost', or to suit the architects vocabulary the 'spatial location of cost'. The best way to illustrate this method is by comparing it with traditional design practice. If we look at the way information is processed from the client to the architect, and from the architect to the quantity surveyor, it is clear that in the first place the client expresses the brief chiefly in the form of functions with different priorities of inter-communication. From that, the architect elaborates a three dimensional model where functions are associated with spaces or rooms. Then the three dimensional model is represented in two dimensional projections (i.e.: plan, elevation or section), which the quantity surveyor uses to take off measurements. It is at this stage that an essential piece of information is lost; the quantities of materials measured from the drawing are not related to spaces and subsequently the functions to which they belong.

It is by associating building elements to spaces (or rooms), and by providing the means to maintain these relationships that it is possible to appreciate the cost implication of design variations. The integration of CAAD with 'intelligent' building modelling and cost modelling, enables us to preserve this spatial information, and pass it across to the quantity surveyor for 'intelligent' cost modelling.

This method of cost modelling is not entirely new, and has been in 'vogue' in the late 60s and early 70s, but did not have much success in the application field. Under the name of '**spatial costing**', its concept was to express the cost of a room of a certain type in terms of its floor area.

Generally, it is accepted that there is a high correlation between the cost of room finishes, fittings, ceilings and walls, and the activity for which the room is designed. Other items like infra-structure, structure, external envelope and services are considered separately. For instance, the cost of a wall attached to a room would be the cost of the finishings plus approximately 50% of the structural partition; same thing with floors and ceilings.

The main reason for the unpopularity of this method is the difficulty in dividing up the normal bill of quantities measurements of the building into the room categories for cost analysis purposes. According to Ferry & Brandon (1984) :

> "The room concept requires more complex analysis and is therefore even more expensive to use, It also lacks the simplicity of the elemental form and the clear communication of what is meant and included."(p 112).

This criticism of the spatial costing method is now less valid, because computer models, when appropriately conceived, enable us to perform complex analysis and still present it in a simple elemental form. In fact, Beeston (1987) believes that for a substantial improvement in estimating, advice to designers and cost control, it is necessary to change fundamentally the basis of calculation so that it corresponds to the way in which cost arises. Ferry & Brandon (1984) had anticipated such progress saying :

> "Computer models which symbolise a particular building and can build up rates from the basic resource costs are more likely to provide a better solution (*to the spatial costing method*) in the future."(p 112).

The results obtained by the computer implementation of this method, through its integration to an 'intelligent' CAAD system, have produced tremendous results. It offers the possibility to involve only a selected number of spaces on a particular floor level of a building, and/or a selected number of floors of a computer building model in the cost analysis. The architect and/or the quantity surveyor are able, for example, to cost circulation areas independently from office spaces, or compare the cost of two spaces located on different floor levels of a building, or examine the cost implication of designing the building in quite different forms.

However, two major deficiencies can be found in this approach. Firstly, the accuracy of spacial costing, limited by the descriptive model technique used for cost modelling, is not as accurate as we may desire. The only alternative is realistic models (see section 5.2.1), but those will require models of the influences of market forces, to which there seem to be no real answer at early design stages. Secondly, this approach can only be used once a sensible three dimensional building model exists in the CAAD system.

A full description of the work undertaken for the implementation of space based cost modelling will be done in the next chapter.

PART C -
APPLICATION DEVELOPMENT.

Chapter 7 : SOFTWARE DEVELOPMENT INTEGRATING GABLE CAD SYSTEM WITH A COST MODELLING SYSTEM.

7.1 GABLE CAD SYSTEM.

The GABLE CAD system began life as an educational and research project at Sheffield University, Department of Architecture. Its early aims centred around efforts to help students understand some of the more technical aspects of building performance using computers to demonstrate the effects of design decisions.

Since its beginnings, an extensive suite of programs was developed, incorporating the early research which aimed to simulate comprehensively building characteristics in terms of appearance, environmental performance, cost, as well as having facilities for producing conventional architectural drawings.

The philosophy behind this suite of programs was both that systems should integrate technical appraisal into the design process, and that its organisation should cause a minimum of disturbance to the designer. These basic premises made it clear that its development would have to be based on an understanding of the way architects approach design and present projects.

GABLE is intended, not only to aid solving specific design problems when they occurred, but also as a facility to maximize the benefit of traditional design methods. Inherent to this sort of approach is the need for flexibility. Although it has proved necessary to realise building performance characteristics to a certain extent, the intention has been to impose a minimum of extra discipline on the designer.

In assessing the various features and benefits of CAAD systems there is one central theme underlying the debate; how integrated these features are, since it is only through integration that the real benefits of increased productivity and better co-ordination of output are achieved. GABLE has pioneered the development of these ideas about an integrated approach to computer aided design in the building industry.

7.1.1 General system organisation.

GABLE 4D SERIES :

Gable is made up of a number of inter-related systems (groups of programs) as seen in Fig 7.1, dealing with two dimensional drafting, three dimensional modelling, data organisation and the interpretation and evaluation of building models (the 'fourth' dimension). Each system consists of a number of modules or programs.

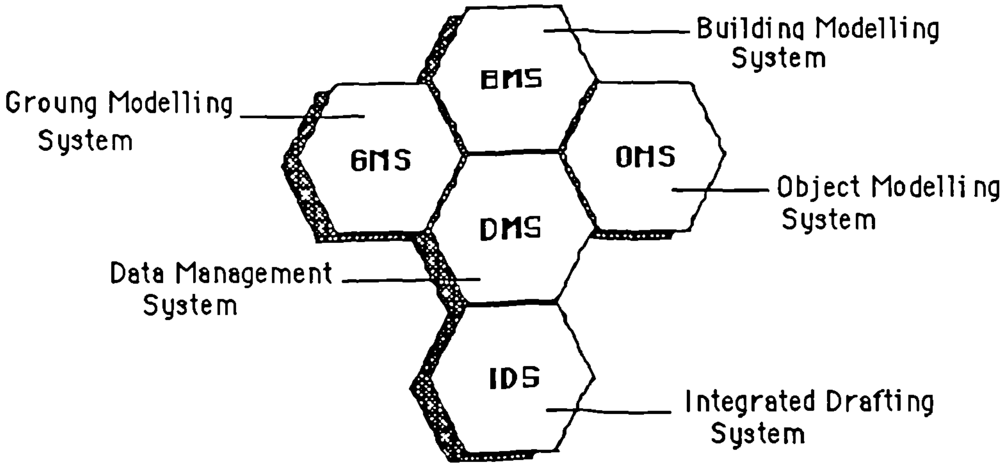


Figure 7.1 - GABLE 4D series MODULES -

IDS (Integrated Drafting System) is GABLE's two dimensional drafting system. Drawings can be created directly in IDS or derived from three dimensional models produced by other Gable system modules.

OMS (Object Modelling System) is one of the three dimensional modelling systems. Dealing with purely abstract three dimensional forms, it allows cuboids known as 'blocks' to be defined and decorated with 2D drawings, as well as full 3D models. Gable will interpret the 3D model to produce 2D IDS drawings.

BMS (Building Modelling System) extends 3D modelling to give some knowledge of building construction. Walls, floors, ceilings, roofs, doors and windows are included in the available elements, the Gable database holds specifications of these building elements, and the system will evaluate the model in terms of enclosed rooms and spaces, as well as deriving 2D drawings.

DMS (Data Management System) is a general relational data base system, which allows the user to interrogate and modify the database Gable uses to hold its knowledge of specifications and properties of elements.

GMS (Ground Modelling System) is a 3D modelling system with special features and elements for modelling land forms and sites.

Various other modules concerned with system management, plotting drawings, etc. are grouped together in a utility facility.

7.1.2 Modules description.

GABLE 4D Series is primarily a building design system (CAAD - Computer Aided Architectural Design).

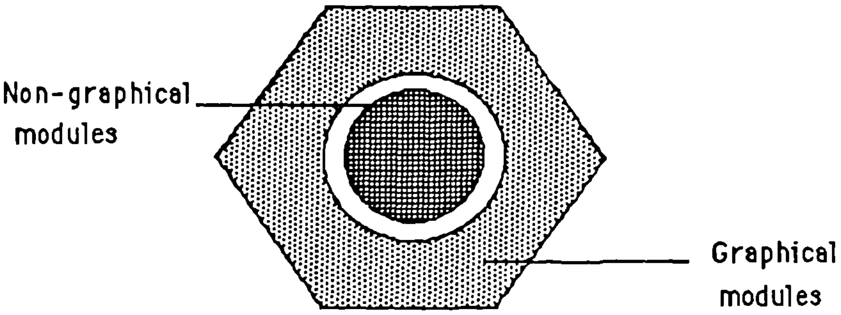


Figure 7..2.1 - Graphical and non-graphical modules.

Its modules, can first of all be grouped into two categories, (1) graphical modules, and (2) non-graphical modules (Fig 7.2). The graphical ones will require a specific hardware to support interactive graphic operations, whereas non-graphical ones could be used on alpha-numeric consoles.

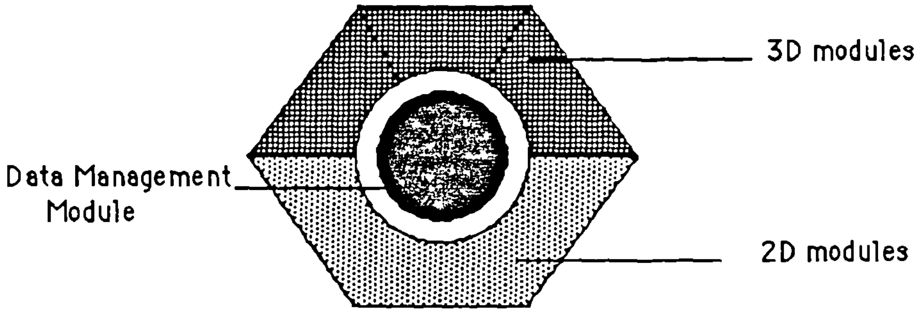


Figure 7.2.2 - 2D and 3D modules.

A further subdivision, with in graphical modules, will distinguish between (1) two dimensional, and (2) three dimensional graphic modules (Fig 7.2.2). However, GABLE allows information to be exchanged, between the two and three dimensional systems. In other words, the 3D system not only allows users to view the model, but pass these views as drawing files into the 2D system. Similarly, 2D drawing files can be passed back to the 3D system. Thus these two systems taken together can provide for the three dimensional co-ordination of drawings. For example, if an element was to move in the 3D model, it will be possible not only to see it in the new position in any 3D view but also to update the 2D drawings (eg: plans, elevations, sections or perspective).

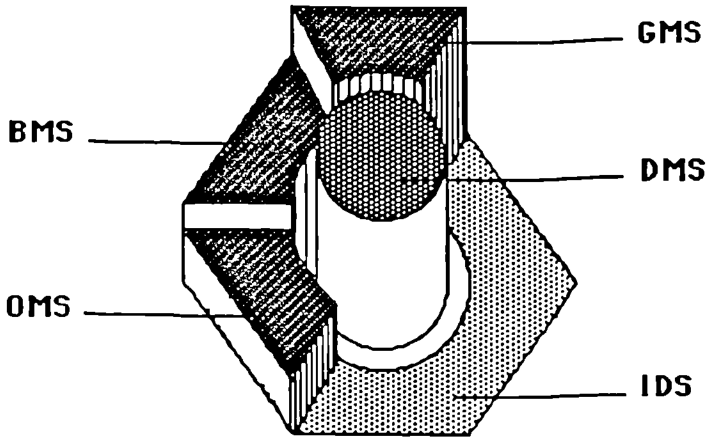


Figure 7.2.3 - GABLE 4D SERIES integration.

The 3D environment on GABLE is further sub-divided into three different modelling applications, each one addressing a particular design modelling need (Fig 7.2.3). For instance, OMS supports the modelling of objects in three dimensions, these are made-up of lines and surfaces. Similarly, GMS supports the modelling of site layouts using specific elements to suit ground modelling and surveys (i.e.: spot-heights, strings, break-lines, etc...). There are currently plans to develop and urban modelling system (UMS), which would handle urban design situations. More related to our research is BMS. It supports the modelling of buildings, which in addition to OMS elements uses building-specific elements -internal and external walls, roofs, floors, ceilings, doors and windows.

With reference to the CAAD systems differentiation made earlier in chapter 3, GABLE can be considered in part as an 'elemental modelling' based system, of which BMS is the core. Since this research is mostly related to building design and evaluation, we shall concentrate on BMS and its related applications.

7.2 Model building with GABLE BMS.

A GABLE building data model is composed of three main classes of elements : surfaces, edges and units. To enable the user to rapidly develop a model which is comprehensible to the system, all the main component parts of a real building are considered as falling into one of these categories. Any complex element must be built up from these basic units. The user adds elements on-screen, usually by means of mouse, bit-pad or puck.

It might be useful to use an example to illustrate the model building process in GABLE, and go through the sequence of events necessary to generate a building computer model. For this purpose, we shall use as a model, a small flat the author lived in during his stay in Sheffield.

7.2.1 Elements specification.

First, it is necessary to specify the building materials, or building components that are going to be used for the construction of the model. These are grouped into three ; (1) surface specification, (2) windows and doors, and (3) staircase design.

SURFACES SPECIFICATION :

This is done using Module 391 of GABLE system. It consists of defining building materials, by describing their density, thermal conductivity, heat specification, etc.. For example, a record structure of the material file will be as follow :

RECORD 1
CODE : 00
TITLE : air cavity

SPEC : 0
K VALUE : 0.11
DENSITY : 1.768
SPECIFIC HEAT : 0
VAPOUR RES :
DAMP PROOF :

RECORD 2
CODE : 01
TITLE : facing brickwork
SPEC : 1
K VALUE : 1.2
DENSITY : 160
SPECIFIC HEAT : 0
VAPOUR RES :
DAMP PROOF :

Out of these materials the user composes his surfaces. A surface can consist of one or several layers of different materials or different surfaces. For example a surface record might be as follow:

surface spec : 1
code : 001
description : facing brickwork ext.
surface type : external wall

LAY	M/S	SPEC	THICKNESS	DESCRIPTION
* 1	M	34	150 mm	facing brickwork
2	M	1	50 mm	air cavity
3	M	21	50 mm	polyurethane foam insulation
* 4	M	5	150 mm	dense concrete block
5	M	13	12 mm	dense plastework

overall thickness 362 mm
dimensioned thickness 350 mm

A surface can be used in different situations, and will be named according to the particular context in which it is used. For example, a surface might be an external wall, internal wall, floor, ceiling with room below, roof, etc... (see Appendix 10.2).

This information is stored in a database of the relational type, which uses many-to-many relationships to allow the same material to be used in different surfaces, and the same surface to use different layers of material. The information on the layering of materials in a surface is used in graphical mode to represent, in a section, the different textures or pattern for details drawing.

WINDOWS AND DOORS :

Windows and doors are specified in Module 391, and are graphically represented using OMS BLOCK elements. The latter are cuboid volumes, defined by their width, depth and height. Doors can have more than one swing, and windows can have various glazed areas as well as a various number of segments. The characteristics of all windows and doors are stored in a DMS file, whereas the geometrical descriptions and attributes (i.e.: pen, colour etc...) are stored as OMS blocks. The plan graphics may be edited and elevations or sections produced (see Fig 7.2.4).

STAIRCASE DESIGN :

It is Module 392 that is needed to specify staircases. This specification option is not particularly relevant to this work, but could be very briefly described as a means to produce outline designs of straight and circular staircases to UK Building Regulations (or any other regulation). Optionally, this module will produce a BUILDING file of the stair for use in modules 320 - FLOOR LAYOUTS.

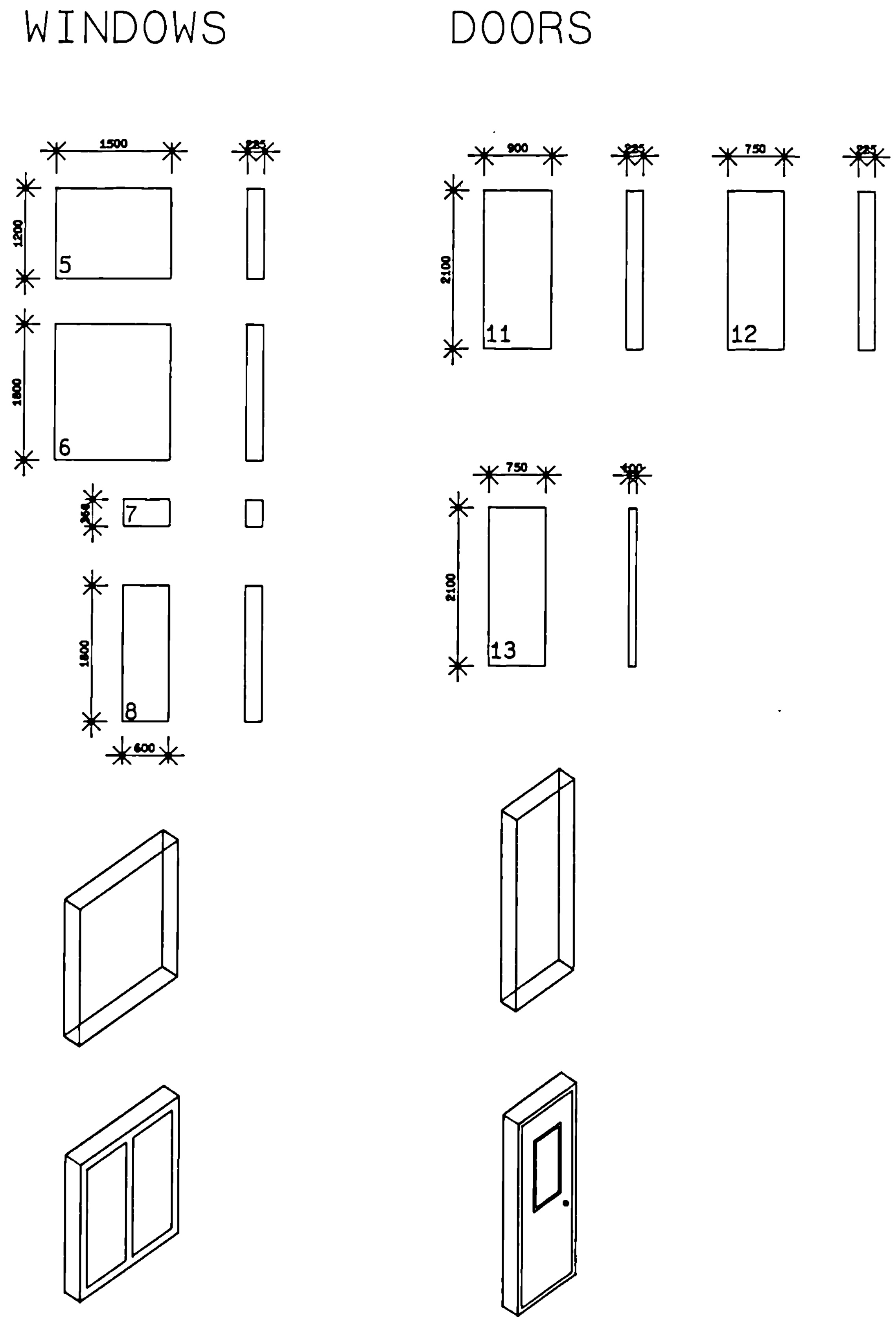


Figure 7.2.4 - Windows and doors graphical specification.

7.2.2 Building modelling.

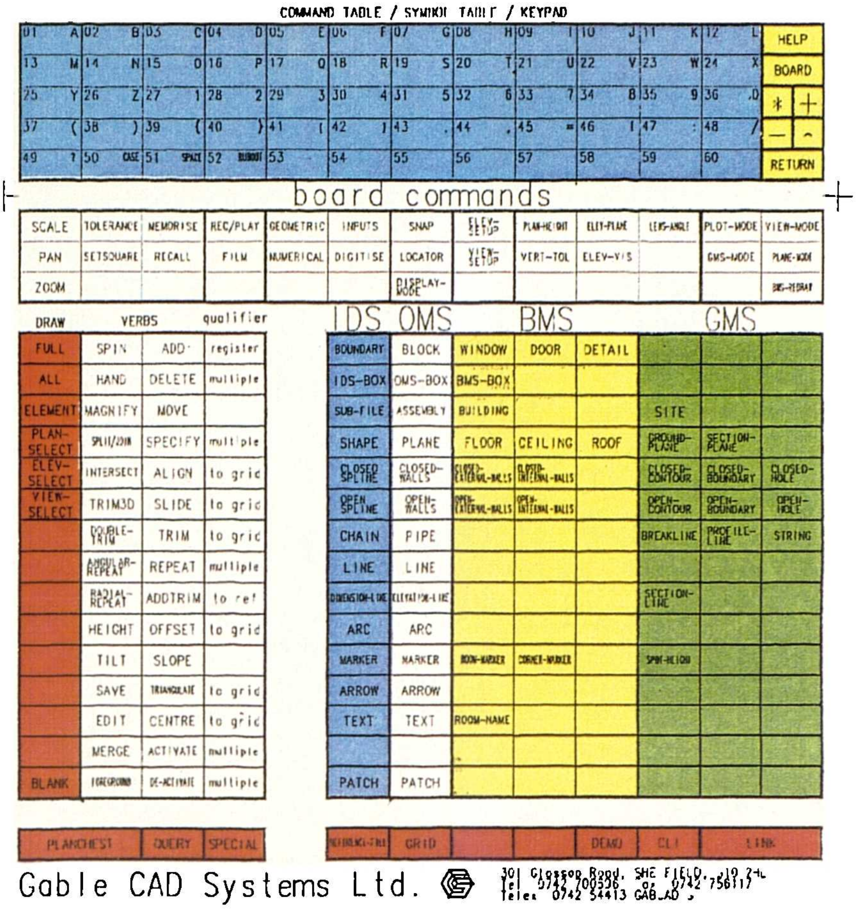
The graphical modelling of buildings on GABLE CAD system is done using Module 320 - FLOOR LAYOUT. Considered as the most comprehensive module of the systems, BMS includes the three specification modules we have described above, a graphical floor layouts module (Module 320), two interpretation modules (Modules 300 and 310), and a continuously expanding list of evaluation modules for thermal, lighting , acoustic and cost evaluation.

FLOOR LAYOUTS is the BMS graphics module. A building model may be a discrete item or may be made up of up to 100 floor levels (0-99). All OMS elements are available as well as BMS building specification elements. User-defined windows, doors and constructions are available and stairs defined using the Staircase Design module may be placed as building sub-files. The 3D building model may be viewed and edited in plan, elevation, section or 3D view (i.e.: perspective, axonometric, etc...). The construction of the building model requires to go through three steps; (1) input, (2) interpretation, and (3) output.

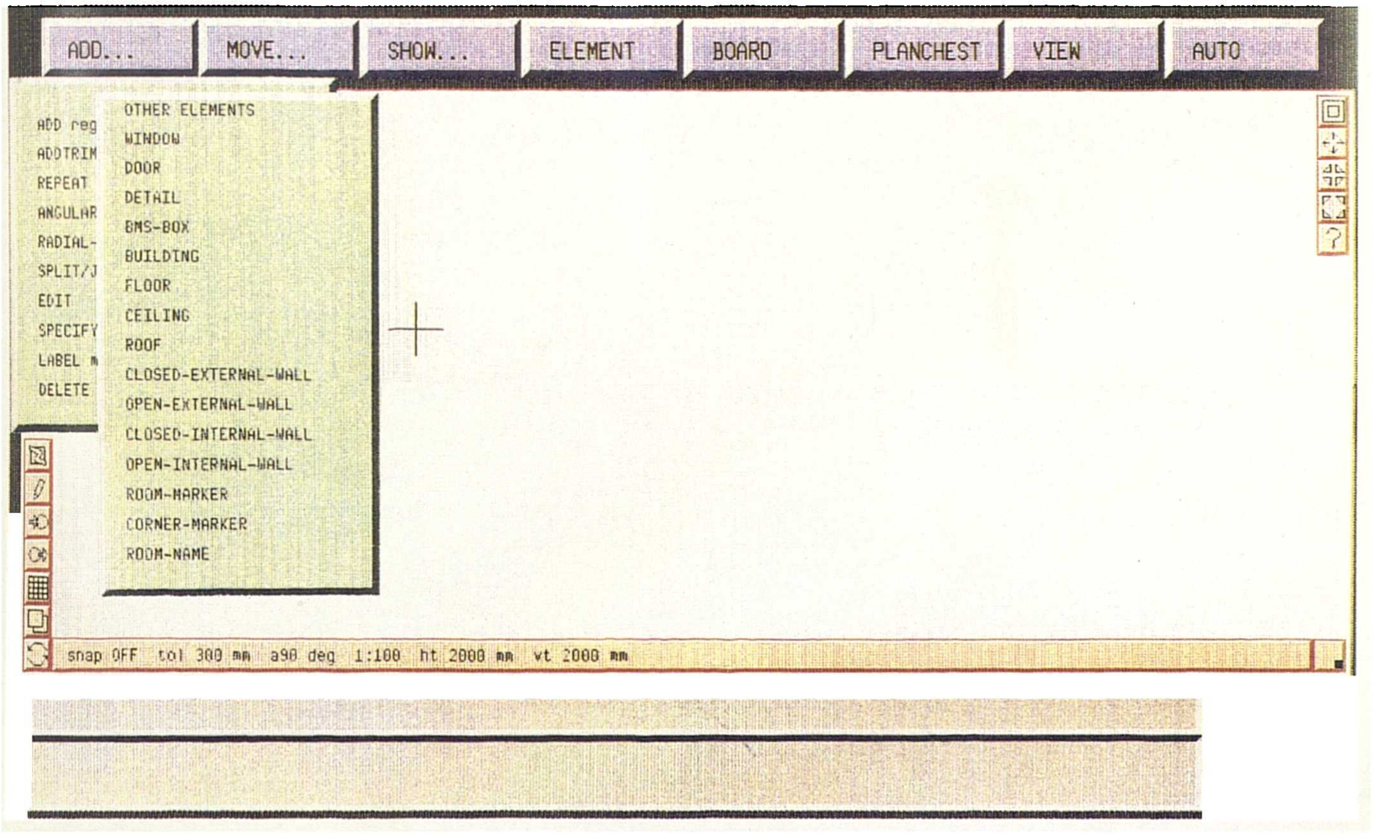
7.2.2.1 Input.

Preliminary to the graphical input, we have seen that the user has to establish the specifications of building elements such as walls, floors, doors and staircases. Respectively using Modules 390, 391, and 392. These elements can now be used to generate building layouts in Module 320. In plan, the user is allowed to ADD building elements using the GABLE graphical user interface.

At the time this manuscript is written, a GABLE user has the issue commands by pointing at commands on a bitpad menu (see Fig 7.2.5 (a)),



7.2.5 (a) - GABLE 4D-SERIES digitiser menu.



snap OFF tol 300 mm a90 deg 1:100 ht 2000 mm vt 2000 mm

7.2.5 (b) - GABLE 4D-SERIES pull-down menu user interface.

but there should be shortly a pull-down menu user interface available (see Fig 7.2.5 (b)). All the instructions to perform an operation in the graphical modules, such as creating and manipulating elements, are issued from this bit-pad COMMAND system. These commands can be identified as follow :

BOARD commands control the nature of the display and aspects of its use as an 'electronic drawing board'. For example, there are commands to change SCALE, PAN, and ZOOM drawings.

DRAW commands provide a range of ways in which the contents of the display are presented. They include views like PLAN, ELEVATION and PERSPECTIVE.

VERBS and QUALIFIERS provide a comprehensive range of functions to create and modify drawings. VERBS are self explanatory, for example ADD, DELETE, ALIGN, MOVE, etc...

ELEMENTS are nouns which vary from module to module. For example in Module 320 (BMS), they are EXTERNAL WALLS, INTERNAL WALLS, CEILINGS, FLOORS, etc... A detail description of BMS elements will be listed in the next section. To issue a full command , the user has to combine a VERB with a NOUN, like : ADD + FLOOR, or MOVE + INTERNAL WALL, etc...

The building floor layout input is simplified for the user in a number of ways. For instance, only the centre line or one edge of walls need to be drawn since the system can obtain their thickness from the surface specification file. For example, in Figure 7.2.6 the ADD + CLOSED EXTERNAL WALL, command action is shown by a closed loop of directed segments. Which, when completed will automatically take the appearance shown in Figure 7.2.7, where the entered segments get expended to the wall thickness they are supposed to represent.

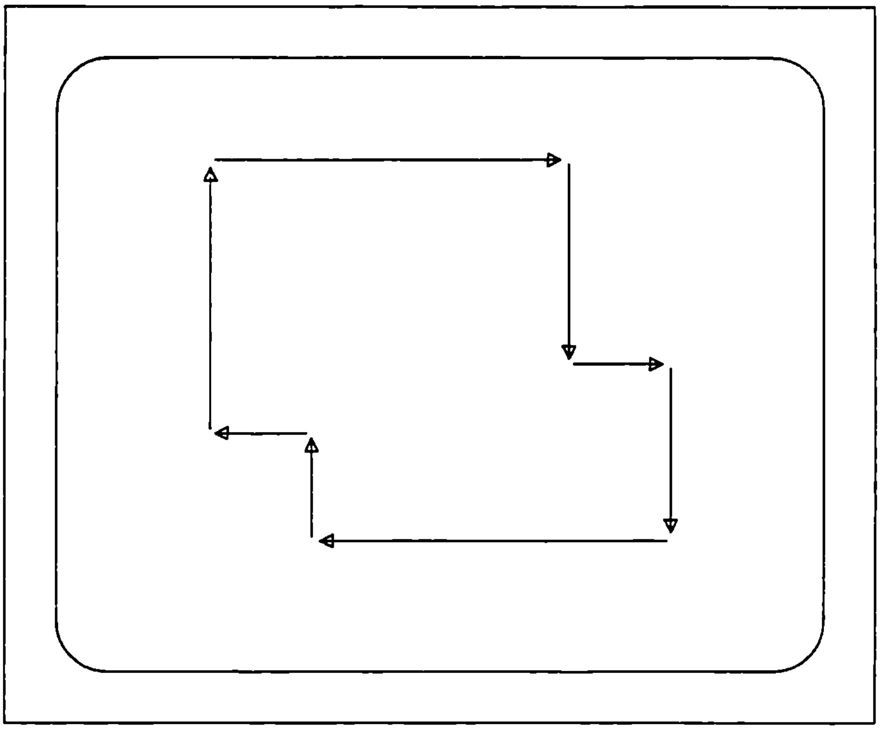


Figure 7.2.6 - The ADD + CLOSED EXTRENAL WALL command input.

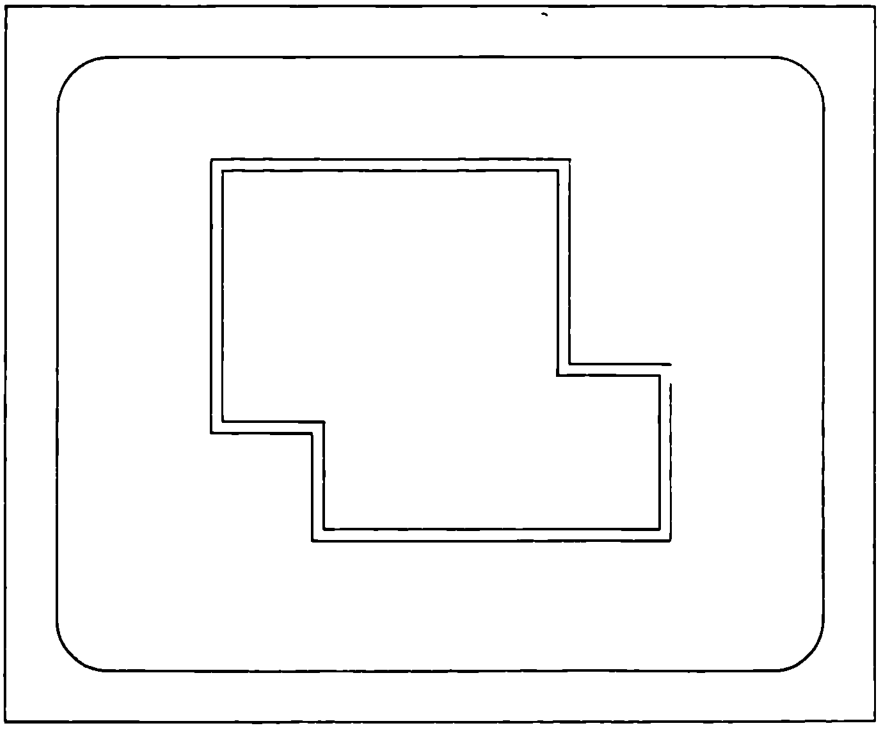
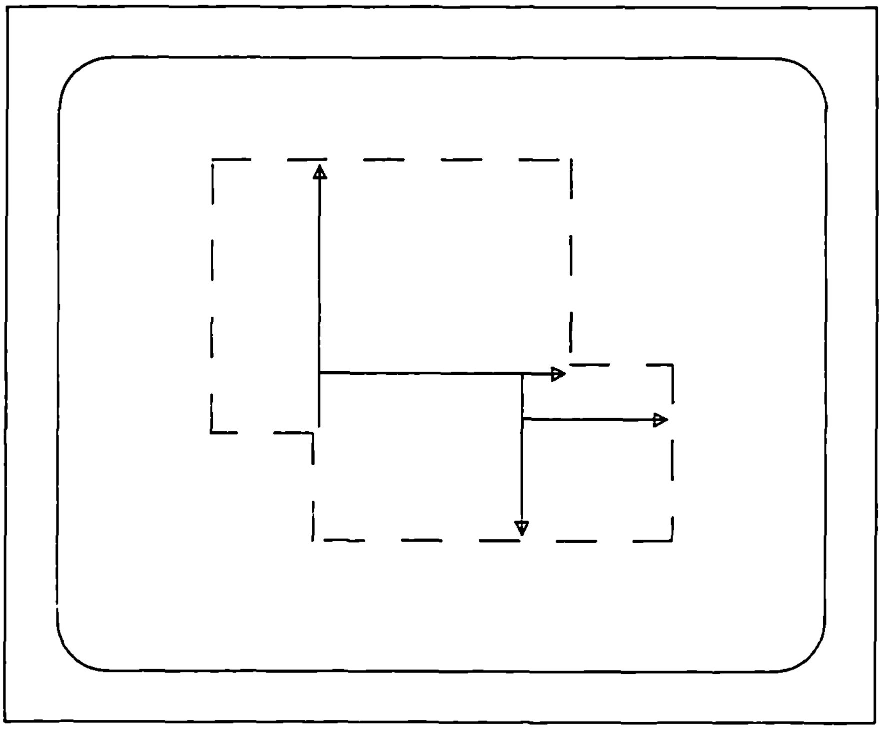
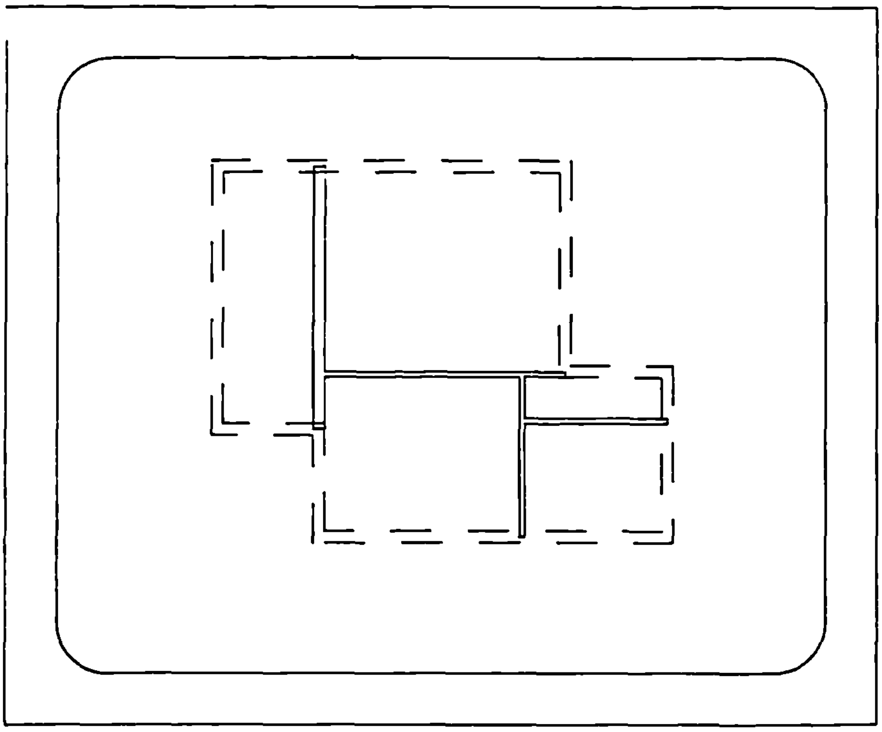


Figure 7.2.7 - CLOSED EXTERNAL WALLs as re-drawn on the screen.



7.2.8 - The ADD + OPEN INTERNAL WALL command input.



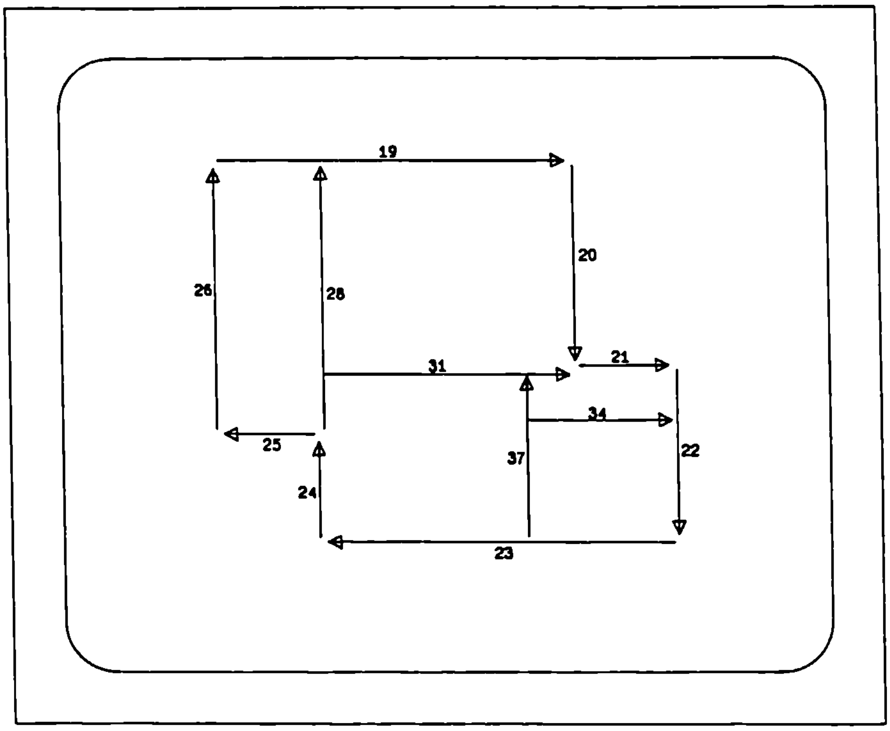
7.2.9 - OPEN INTERNAL WALLs as re-drawn on the screen.

Similarly, when the ADD + OPEN INTERNAL WALL command is issued, by pointing to both ends of each wall partition viewed in plan (see Figure 7.2.8). These get redrawn as seen in Figure 7.2.9. Each wall panel can of course have a different specification and subsequently a different thickness. Various graphical manipulations can be made on each of these building elements, until the suitable dimensioning of rooms in found. Walls can be ALIGNED or TRIMED one to the other. Each wall segment is associated to a number representing its position in an array of the BMS data structure as it will be explained in section 7.2.3 (see Figure 7.2.10). These wall numbers are used at interpretation and evaluation stages.

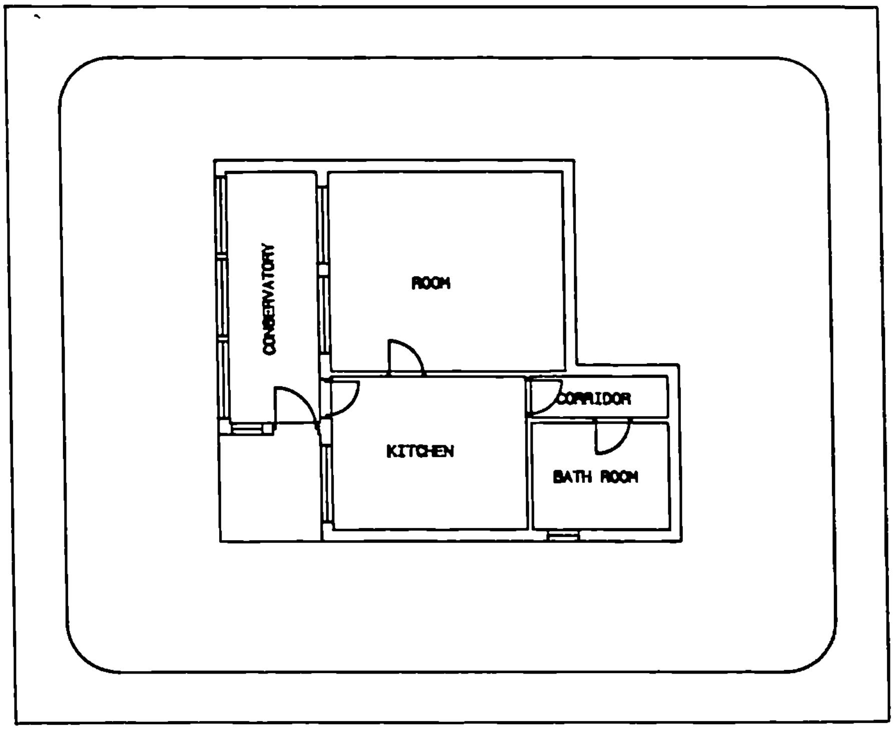
Windows and doors need only to be positioned by their centre and will automatically be aligned by the system into their walls (see Figure 7.2.11). As long as floors and ceilings are flat and horizontal they need not be drawn at all as GABLE will insert these automatically during the interpretation phase.

When storing each floor level the user must specify which level (0-99) it is to be in the building and its datum. All heights specified in that floor are relative to that datum, thus by simply altering this floor datum the whole level may be raised or lowered in the building. During interpretation GABLE will fill each space with a floor and ceiling at these heights and of the defaults specification, unless otherwise instructed.

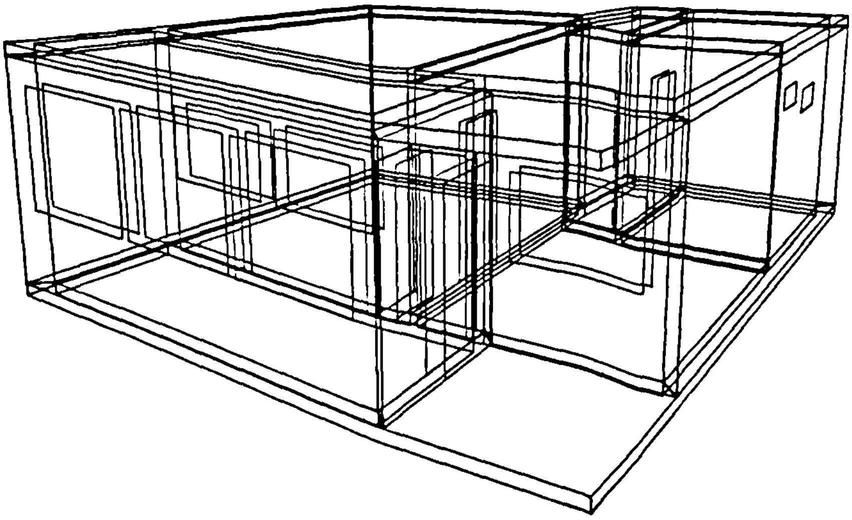
If the floor or ceiling in any room are irregular, then they must actually be drawn by the user and GABLE instructed not to create them automatically by use of room markers. A room marker element may be added in any room to instruct GABLE to vary from the default of that room. The floor and/or ceiling may thus be raised or lowered, have their specifications altered or be omitted altogether by this means.



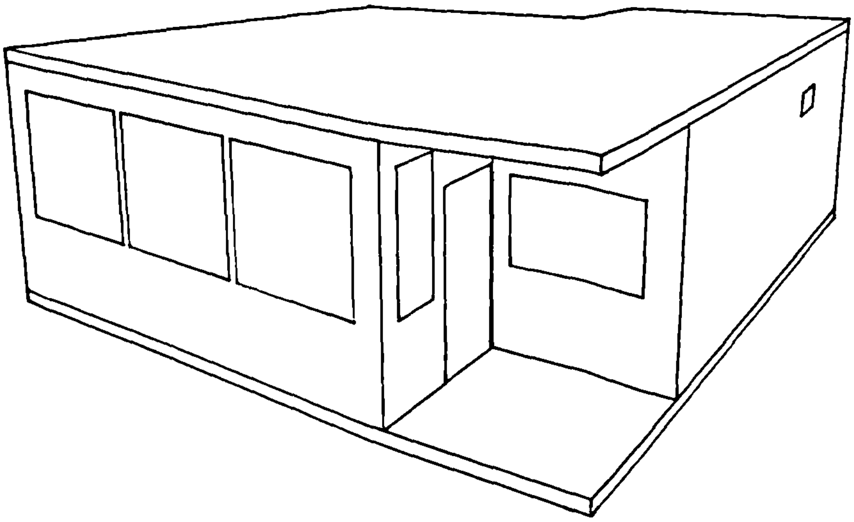
7.2.10 - Wall numbers as positioned in the data structure.



7.2.11 - Plan view of building model with windows and doors fitted.



7.2.12 (a) - Wireline external view of the building model.



7.2.12 (b) - Fully-hidden line external view of the building model.

Room markers may be positioned anywhere inside a room but no more than one marker should be used in each room. Room names may be used throughout the building but care should be taken to avoid duplicating names in rooms unless these rooms are to be considered identical (see Figure 7.2.12). Only one name should be used in any one room. After the interpretation phase it will be possible to refer to rooms by these names.
The three-dimensional model can be checked for unexpected geometry in perspective or other view, either in WIRE-LINE (see Figure 7.2.13 (a)) or in FULLY-HIDDEN LINE (see Figure 7.2.13 (b)).

7.2.2.2 Interpretation.

GABLE BMS has an automatic interpretation module known as ISAAC (Interior Space Assembly And Contents). This module uses the data prepared in the specification (i.e.: modules 390,391,and 392) and layout (i.e.: module 320) modules to assemble a meaningful model of the building as a whole (Lawson & Riley 1983). It reads this information, and constructs spaces inside the building as in Figure 7.2.13, generating a file called 'interpreted assembly'.

GABLE keeps a record of the status of each floor in the building and ISAAC will, if requested, perform all the work necessary to re-establish a complete building model no matter what changes have been made to the building. ISAAC works up the building from the lowest level to highest, performing its interpretive function. The first job at each level is to analyse the arrangement of external and internal walls. This is done in two stages, first internal space outlines are defined by going along each wall and finding intersections or abutments occurring on the right hand side of the wall.

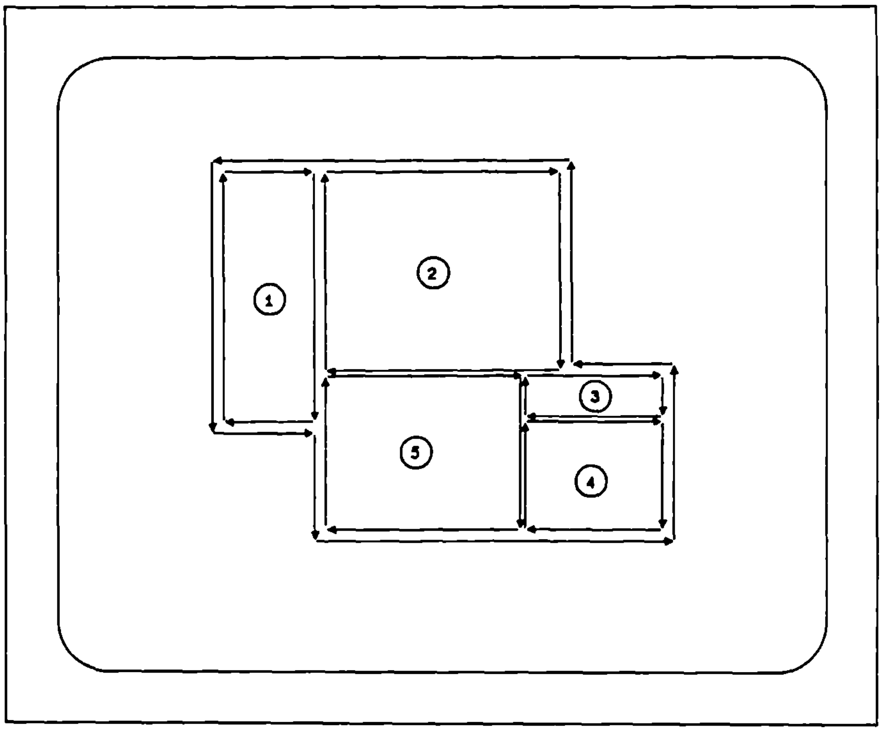


Figure 7.2.13 - Space outlines definition during model interpretation.

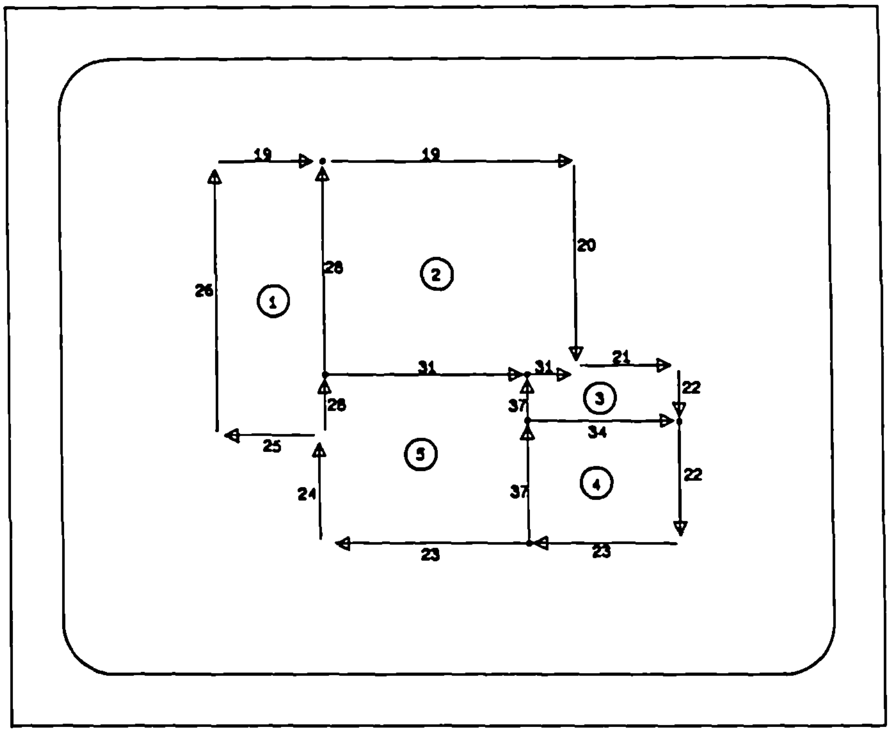


Figure 7.2.14 - Walls splitting at abuttements and intersections.

Second, the external space outlines are found for each building block or internal open space, by going along each wall and finding intersections or abutments occurring on the left hand side of the wall (see Figure 7..2.13).

Once this analysis is completed, ISAAC establishes the plan outline of each space and each loop of external surfaces at that level. It then analyses windows and doors and assigns each one to its original wall and associates it with either two rooms or one room and the exterior. At this point it is important to note that further work had to be done by the author on ISAAC for the purpose of this particular research project. It consisted essentially of splitting each single wall panel that has an other wall butting into it or intersecting it, as shown in Figure 7.2.14. The reason for this being the need to associate each single wall panel, in a floor level, to either two rooms or one room and the exterior. It allows the spatial location of walls, and not just openings (i.e. : windows and doors), as previously handled by ISAAC . This approach enables us to consider spatial modelling as against elemental modelling for building evaluation. While the user concentrates on getting the building layout consistent, in terms of building elements interaction, the system looks after the implicit arrangement of spaces that are created. This interaction between **elemental** and **spatial** description of a design solution, is considered here as an original contribution to the field of computer aided building modelling and evaluation.

What has been achieved here is the mapping of building elements relationships horizontally, by extracting the implicit arrangement of closed spaces contained in a user input of a building floor layout. Establishing the relationship of spaces in a computer building model raises the issue of spaces adjacencies, mentioned previously in section 3.2.3, where graph theory was shown to be of most relevance.

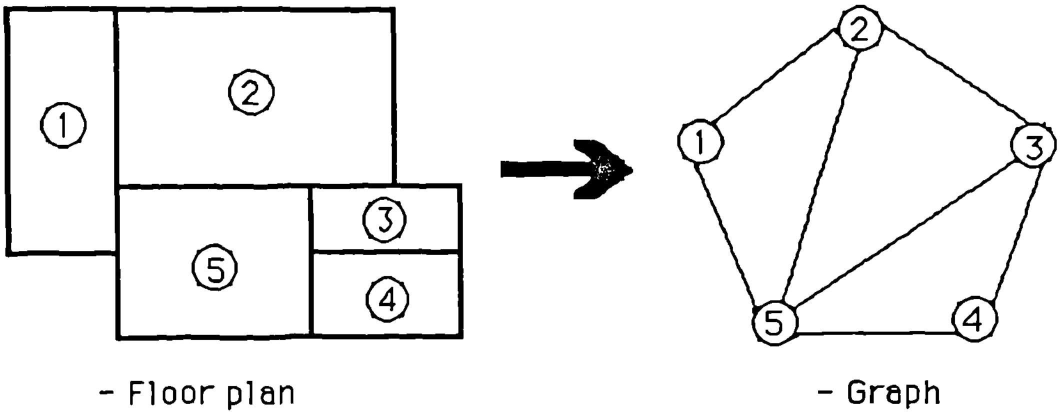


Figure 7.2.15 - Graph representation of spacial adjacencies in plan.

The interpretation of the GABLE building model described above, will establish a graph of the spatial adjacencies as shown in Figure 7.2.15. Spaces are represented by the vertices of the graph, whereas wall panels are represented by the graph edges.

However, the described enhancement can only apply to walls and not to user defined floors, ceilings and roofs. This will restrict the spacial location of surfaces to vertical ones, since user added floors, ceilings, and roofs are not associated to spaces. The interpretation of three dimensional spatial relationships of building models, is beyond the scope of this thesis. Such area of research was dealt with by Teague, L.C. (1970), who used network representation of squared rectangles to contain geometrical and topological information.

Further options in ISAAC allow the user to call for a multi-storey space model to be built. This may be started by selecting any named room at any level. That room, and all other rooms connected to it on the floor above or below by holes,

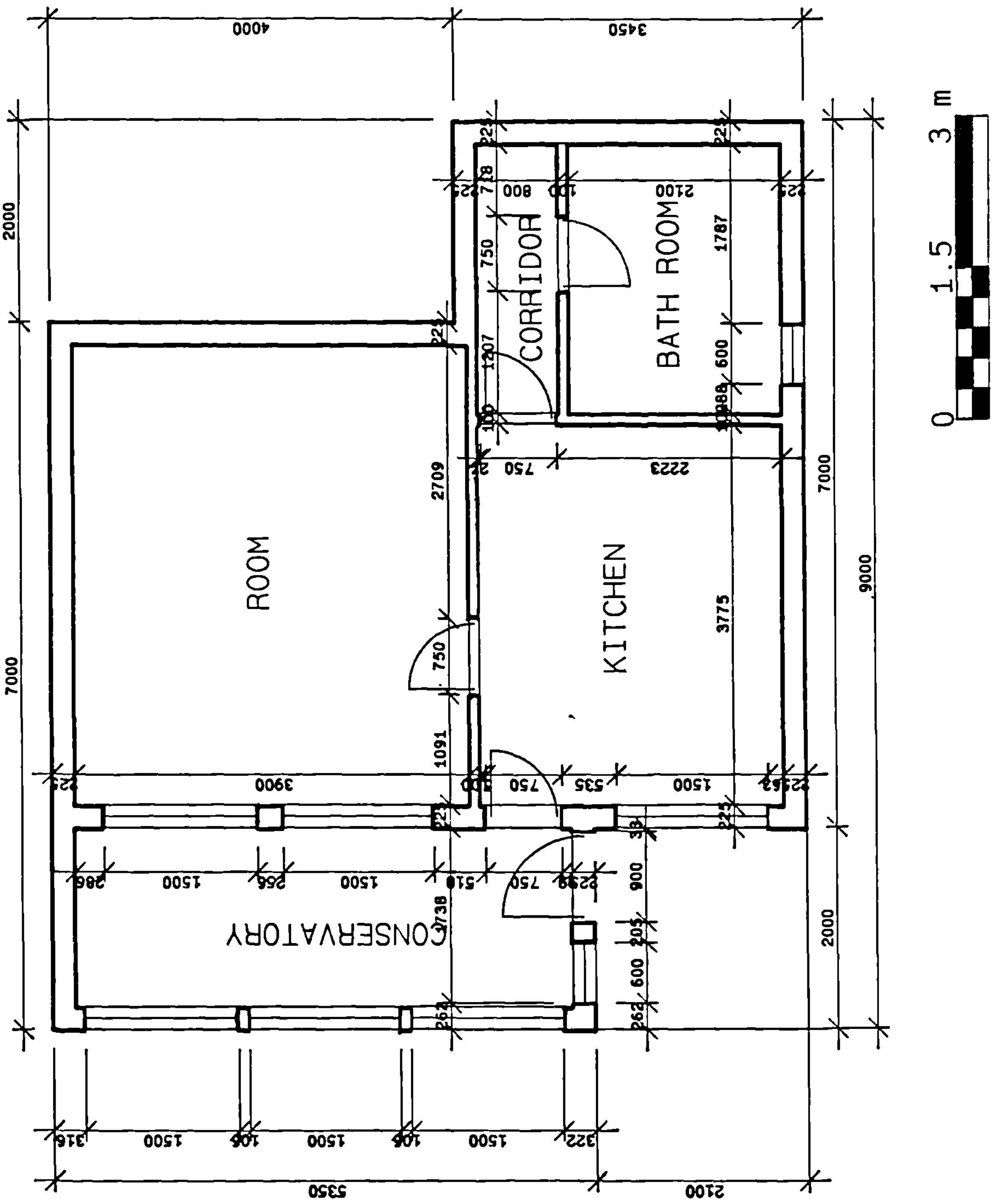


Figure 7.2.16 (a) - View in plan of building model with dimension line.

are then included in the multi-storey model. Similarly the exterior model can be built. These models may be useful for interior or exterior perspective work.

7.2.2.3 Output.

Once ISAAC has established all the relational data which shows the interconnections between elements of the building model, it is possible to obtain a wide range of output, both graphical and numerical. In our particular case, it is the latter type of output that we will be interested in.

i - Graphical Output.

Although visualisation drawings are available during the input phase, much more accurate architecturally conventional drawings can be created automatically from the interpreted building model. Floor plans will now show walls correctly meeting at their ends and corners. It will also show windows and doors in sections, in relation to their host walls. Similarly, building sections, elevations and whole range of visualisation drawings are now available. These drawings are all passed directly in GABLE IDS (Integrated Drafting System), where they may be edited, enhanced, combined, and finally plotted (see Figures 7.2.16 (a), (b) and (c)).

ii - Numerical Output

The building may now be 'surveyed' to produce a range of data base files, showing the quantities of elements in the building. These files are passed directly into the GABLE DMS (Data Management System), and are compatible with the specification files generated during the input phase, with which they may be cross referenced. DMS itself provides a comprehensive range of analysis, editing and report publishing facilities, including business graphics into the IDS. The survey output will be fully described in section 7.1.4.

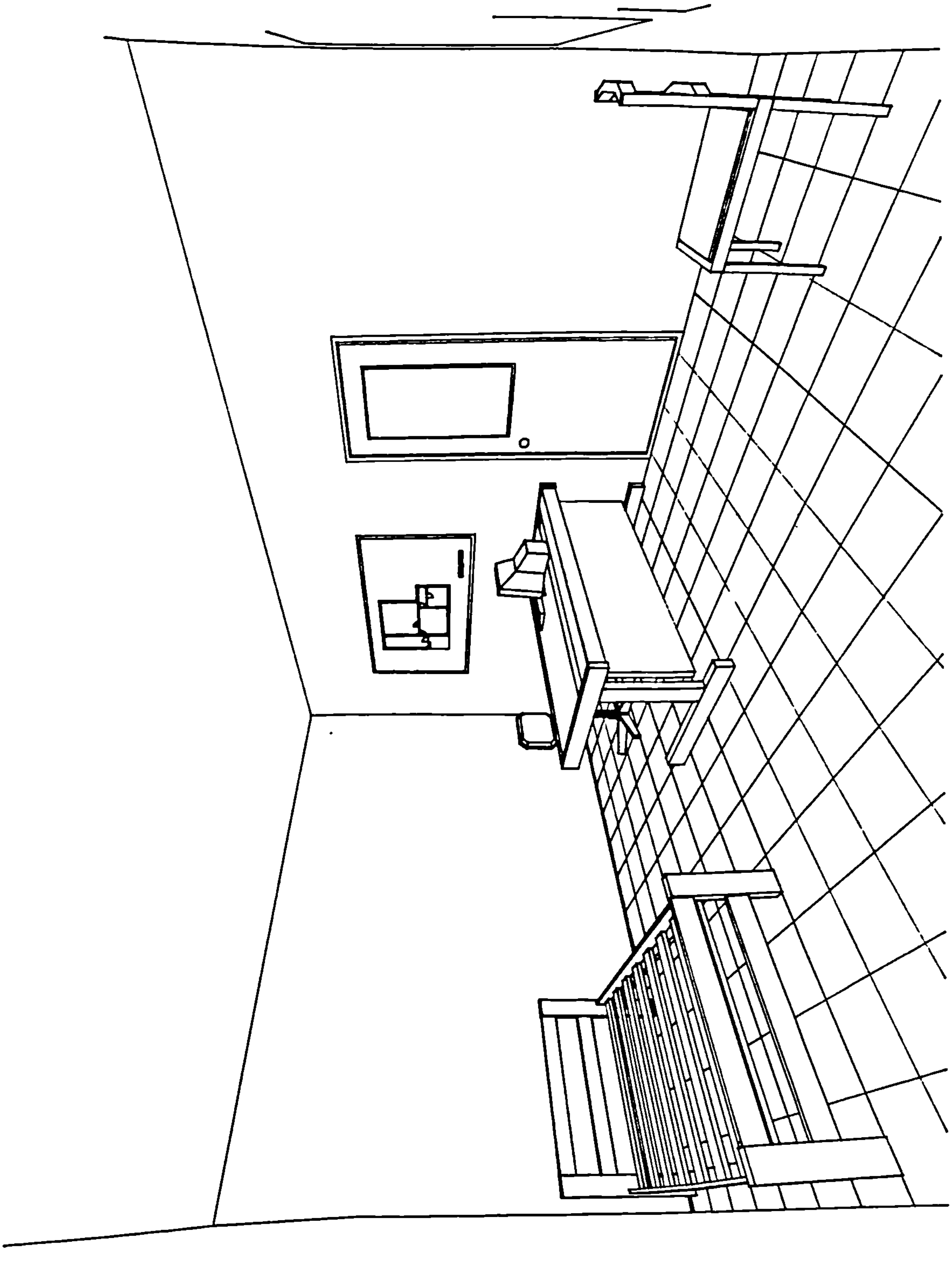


Figure 7.2.16 (b) - Internal view of building model in FULLY-HIDDEN LINE.

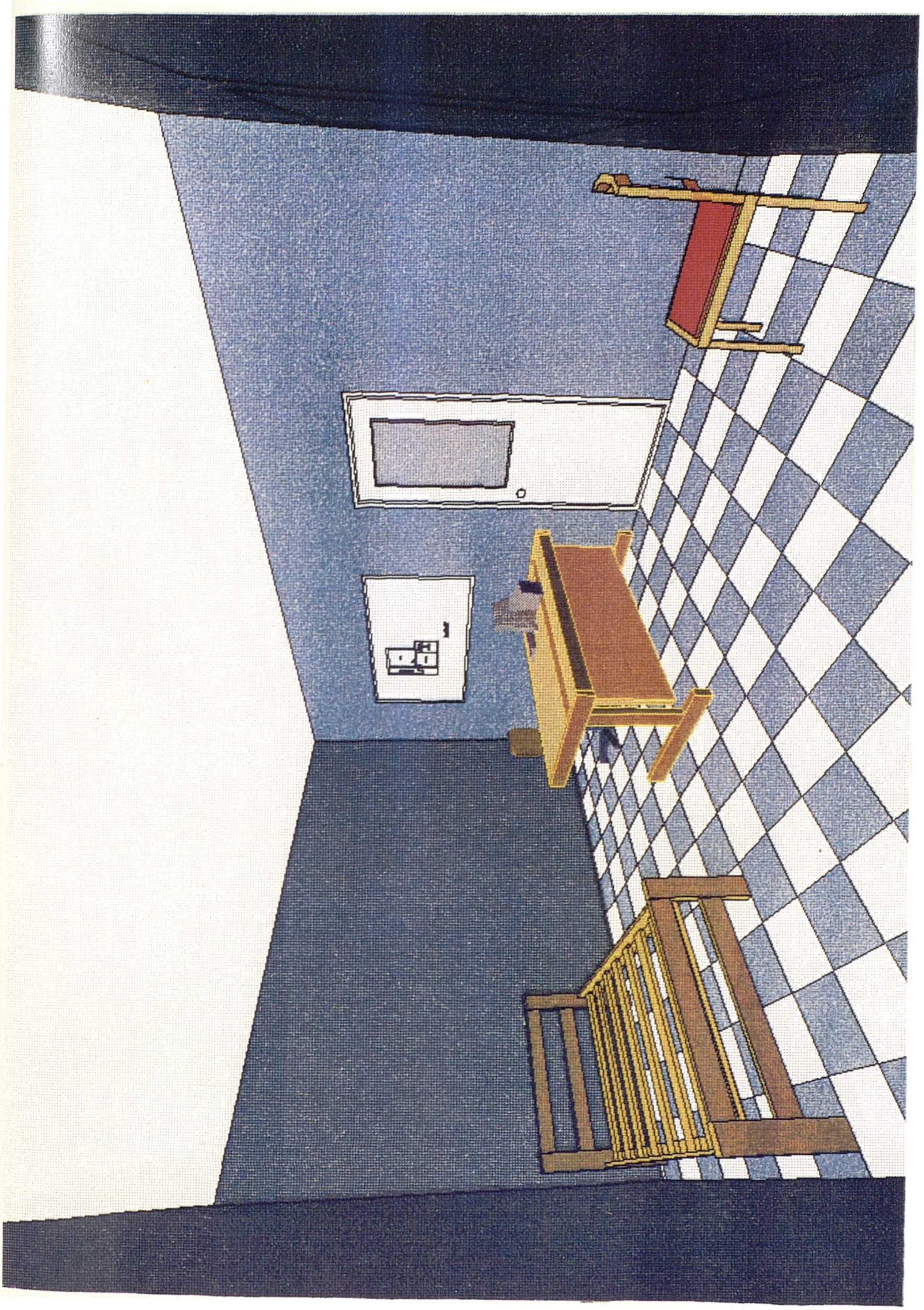


Figure 7.2.16 (c) - COLOUT FULLY SORTED screen dump out put.

7.2.3 Data Structures.

The sequence of events described above, is the outcome of a complex data manipulation, invisible to the user, taking place on a host computer. In memory, GABLE data is held in large arrays. All main geometrical data is stored in three single column real by eight (real *8) arrays PX(MP), PY(MP), and PZ(MP). Two further arrays are PS and PF6 of integers by 4 (integer *4) are respectively used to hold specificational data, and header information for each element to determine its type.

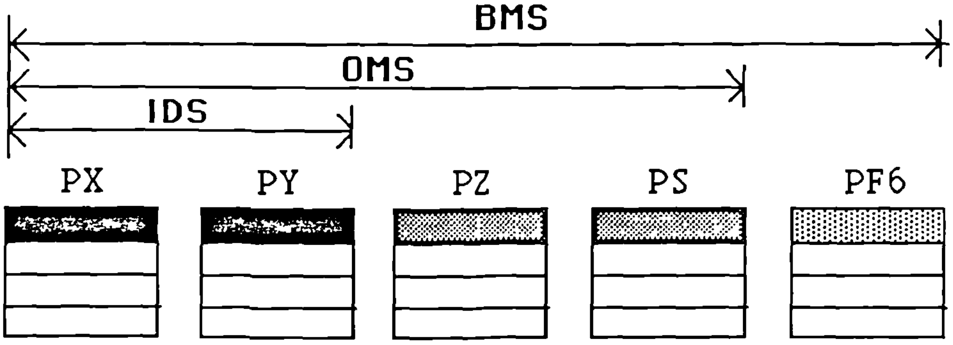


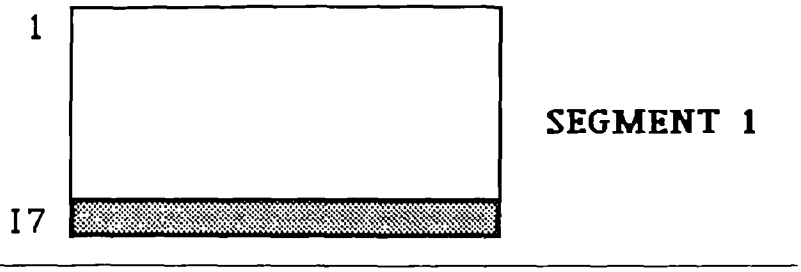
Figure 7.2.17 - GABLE arrays structure.

These arrays are divided into three main segments :

SEGMENT 1 - consists of rows 1 to I7 of the arrays, and contains header information describing the two other segments.

SEGMENT 2 - rows I7 +1 to I9 contains all linear and polygonal elements with F2 values 3 to 9 and 15.

SEGMENT 3 - rows I8 to MP (end of file) contains all points with F2 values 10 to 13.



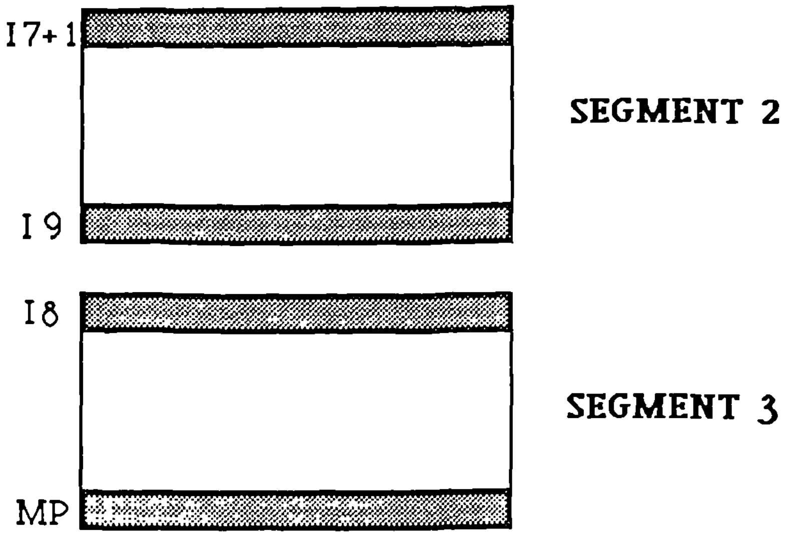


Figure 7.2.18 - GABLE data structure segments.

Each GABLE BMS element occupies one header row of PX,PY,PZ, PS and PF6 together with N further rows of data. The integer value of the header in PX always specifies the number of rows following. The integer part of PY indicates the F2 value of the element (F2 is a variable that holds a value to identify the element used, example; F2=5=closed wall, F2=10= arc, etc...)

F2	BMS	OMS	IDS
1	Window/Door	Block	Boundry
2	BMS - Box	OMS - Box	IDS - Box
3	Sub-building	Sub-assembly	Sub-file
4	Floor/Ceing/Roof	Plane	Shape
5	Closed int/ext wall	Closed wall	Closed spline
6	Open unt/ext wall	Open wall	Open spline
7		Pipe	Chain
8		Line	Line
9		Elevation-line	Dimension line
10		Arc	Arc
11	Room-Marker	Marker	Marker
12		Arrow	Arrow
13	Room name	Text	Text
15		Patch	Patch

Figure 7.2.19 - Table of elements associates to F2 values in GABLE modules.

Finally, an additional array (PP for the Patch element) keeps track of the relational data for patches, while in memory.

A very important feature of GABLE data structure, is contained in a file created by the interpretation module ISAAC (i.e. : Module 300). This file is called interpreted assembly, and contains exactly the same information as the BUILDING file, in addition of two arrays PR1 and PR2. These two arrays hold relational information

7.3 SURVEYING BUILDING MODELS.

Surveying a building model is a very significant facility available in GABLE, it is part of DMS surveying options. Module 630 : DMS - BMS surveyor, allows the user to produce schedules (example : room area, room volume, walls girth, windows and doors area, etc...) for any BMS file or building model.

Four files are required for the execution of such a survey; two DMS data files and two graphical assemblies. Out of these, Module 630 generates five DMS files named LEVEL, ROOM, ISAAC, LAYOUT and OPENING (see Figure 7.3.1).

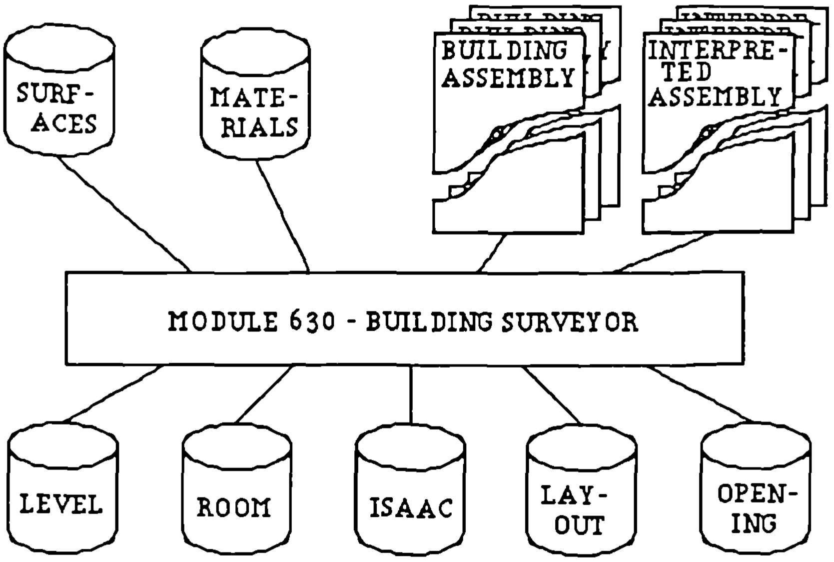


Figure 7.3.1 - Files involved in a MODULE 630 building survey.

The information contained in these files is continuously being enhanced, as the system incorporates new features or new elements. For example the extra-interpretation performed by ISAAC, described in section 7.2.2.2, has required the inclusion of two extra-fields to the survey LAYOUT file structure. More recently, a all new file has been added to the building surveyor output. The OBJECTS file will contain information on any element added as a sub-file in the building model.

7.3.1 Data production for building evaluation.

The process of building surveying is in itself fairly simple. For example, all measurements in the ROOM, ISAAC survey files are made using the data from the interpreted assembly (i.e. : generated by Module 300 - ISAAC). The measurements in the LAYOUT file are generated directly from the BMS wall data (i.e. : SURFACES and MATERIALS - DMS files), as well as both the building assembly and the interpreted assembly. The LEVEL file is created using data from the building model index file (i.e. : file describing the summery of the building model), and data in the already produced survey file ROOM. Finally, the OPENING file which consists of a list of all windows and doors in the range of levels being surveyed. It is created using data from the interpreted assembly. As mentioned earlier (see Section 6.2.1), specificational, relational and location descriptions contained in the graphical building model are all contained in these survey files. The structural contents of all the survey files will be found in appendix 10.3 .
For example, the survey of the building model generated in section 7.1.2 , will produce the DMS data reported in Figures 7.3.2 (a) to (e).

This data produced for building evaluation, has the very special characteristic to be, at the same time, a mirror image of the information contained in the original BMS building model, and represent a "reading" of the inter-relatedness of all the elements in the building model (Lawson 1985). It is also interesting to note that many building appraisal programs can make use of this data.

```
********************
LEVEL - FILE REPORT
********************

CODE  LEVEL DATUM    CEILING   DOWNSTAND NO OF       GROSS    EST
                     HEIGHT              ROOMS       AREA     VOLUME
-----------------------------------------------------------------------
  0     0     0      3000         0        5          48.43  145.29
```

Figure 7.3.2 (a) - Report of the building survey file : LEVEL.

```
********************
ROOM - FILE REPORT
********************

CODE ROOM        ROOM LEVEL ROOM   GROSS  FLOOR OTHER  EST      GROSS  NET     WIN- DO-
     NAME                   TYPE   AREA   AREA  AREA   VOLUME   GIRTH  GIRTH   DOWS ORD
---------------------------------------------------------------------------------------
1    CONSERV.    1    0     1       9.46  0      9.46  28.39    13.80  12.15    6   2
2    ROOM        2    0     1      18.08  0     18.08  54.25    17.12  16.37    2   1
3    CORRIDOR    3    0     1       2.23  0      2.23   6.69     7.17   5.67    0   2
4    BATH ROOM   4    0     1       6.48  0      6.48  19.44    10.22   9.47    1   1
5    KITCHEN     5    0     1      12.17  0     12.17  36.52    14.00  11.75    1   3
```

Figure 7.3.2 (b) - Report of the building survey file : ROOM.

```
*********************
LAYOUT - FILE REPORT
*********************

CODE  ELEMENT          SPEC   LEVEL WALL    GROSS GIRTH ROOM   ROOM  ORIENTATION
                                    NUMBER AREA         No 1   No 2
-----------------------------------------------------------------------------
1     EXTERNAL WALL    3      0     19       6.33 2.11  1      0       360.00
2     EXTERNAL WALL    3      0     19      14.99 4.99  2      0         0
3     EXTERNAL WALL    3      0     20      11.66 3.88  2      0         0
4     EXTERNAL WALL    3      0     21       5.66 1.88  3      0         0
5     EXTERNAL WALL    3      0     22       2.88  .96  3      0        90.00
6     EXTERNAL WALL    3      0     22       7.46 2.48  4      0        90.00
7     EXTERNAL WALL    3      0     23       8.85 2.95  4      0       180.00
8     EXTERNAL WALL    3      0     23      11.81 3.93  5      0       180.00
9     EXTERNAL WALL    3      0     24       6.30 2.10  5      0       270.00
10    EXTERNAL WALL    3      0     25       6.33 2.11  1      0       180.00
11    EXTERNAL WALL    3      0     26      15.71 5.23  1      0       270.00
12    INTERNAL WALL    4      0     28       3.86 1.28  1      5       270.00
13    INTERNAL WALL    4      0     28      11.85 3.95  1      2       270.00
14    INTERNAL WALL    2      0     31      11.81 3.93  2      5         0
15    INTERNAL WALL    2      0     31       3.18 1.06  2      3         0
16    INTERNAL WALL    2      0     34       8.85 2.95  3      4         0
17    INTERNAL WALL    2      0     37       7.46 2.48  4      5       270.00
18    INTERNAL WALL    2      0     37       2.70  .90  3      5       270.00
```

Figure 7.3.2 (c) - Report of the building survey file : LAYOUT.

```
**********************
OPENING - FILE REPORT
**********************

CODE  ELEMENT SPEC LEVEL WALL   GROSS GIRTH ROOM  ROOM  ORIEN-  CILL   INT/
                         NUMBER AREA        No 1  No 2  TATION  HEIGT  EXT
-----------------------------------------------------------------------------
1      WINDOW   6    0     26    2.70  6.60  1     0     270    1.00    EXT
2      WINDOW   6    0     26    2.70  6.60  1     0     270    1.00    EXT
3      WINDOW   6    0     26    2.70  6.60  1     0     270    1.00    EXT
4      WINDOW   5    0     28    1.80  5.40  1     2      90    1.00    INT
5      WINDOW   5    0     28    1.80  5.40  1     2      90    1.00    INT
6      WINDOW   8    0     25    1.08  4.80  1     0     180    1.00    EXT
7      WINDOW   7    0     23     .21  1.92  4     0     180    1.90    EXT
8      WINDOW   5    0     24    1.80  5.40  5     0     270    1.00    EXT
9      DOOR    11    0     25    1.89  6.00  1     0     180    0       EXT
10     DOOR    12    0     28    1.57  5.70  1     5      90    0       INT
11     DOOR    13    0     31    1.57  5.70  2     5     180    0       INT
12     DOOR    13    0     37    1.57  5.70  3     5     270    0       INT
13     DOOR    13    0     34    1.57  5.70  3     4     180    0       INT
```

Figure 7.3.2 (d) - Report of the building survey file : OPENING.

```
********************
ISAAC - FILE REPORT
********************

CODE ELEMENT   SPEC  LEVEL WALL   GROSS NET   GIRTH ROOM ORIEN-  INCLI- INT/
                           NUMBER AREA  AREA             TATION  NATION EXT
------------------------------------------------------------------------------
1     EXT WALL   3     0     26   15.03  6.93 5.01  1     270     90    EXTERNAL
2     EXT WALL   3     0     19    5.66  5.66 1.88  1       0     90    EXTERNAL
3     INT WALL   4     0     28   15.03  9.86 5.01  1      90     90    INTERNAL
4     EXT WALL   3     0     25    5.66  2.69 1.88  1     180     90    EXTERNAL
5     EXT WALL   3     0     19   14.32 14.32 4.77  2       0     90    EXTERNAL
6     EXT WALL   3     0     20   11.36 11.36 3.78  2      90     90    EXTERNAL
7     INT WALL   2     0     31   14.32 12.74 4.77  2     180     90    INTERNAL
8     INT WALL   4     0     28   11.36  7.76 3.78  2     270     90    INTERNAL
9     INT WALL   0     0     32   0     0     0     3       0     90    INTERNAL
10    EXT WALL   3     0     21    5.32  5.32 1.77  3       0     90    EXTERNAL
11    EXT WALL   3     0     22    2.40  2.40  .80  3      90     90    EXTERNAL
12    INT WALL   2     0     34    8.36  6.78 2.78  3     180     90    INTERNAL
13    INT WALL   2     0     37    2.40   .82  .80  3     270     90    INTERNAL
14    INT WALL   2     0     31    3.03  3.03 1.01  3       0     90    INTERNAL
15    EXT WALL   3     0     22    6.97  6.97 2.32  4      90     90    EXTERNAL
16    EXT WALL   3     0     23    8.36  8.14 2.78  4     180     90    EXTERNAL
17    INT WALL   2     0     37    6.97  6.97 2.32  4     270     90    INTERNAL
18    INT WALL   2     0     34    8.36  6.78 2.78  4       0     90    INTERNAL
19    EXT WALL   3     0     23   11.32 11.32 3.77  5     180     90    EXTERNAL
20    EXT WALL   3     0     24    5.96  4.16 1.98  5     270     90    EXTERNAL
21    INT WALL   0     0     32   0     0     0     5       0     90    INTERNAL
22    INT WALL   4     0     28    3.71  2.1  1.23  5     270     90    INTERNAL
23    INT WALL   2     0     31   11.32  9.74 3.77  5       0     90    INTERNAL
24    INT WALL   2     0     37    9.67  8.10 3.22  5      90     90    INTERNAL
```

Figure 7.3.2 (e) - Report of the building survey file : ISAAC.

Varied fields as room acoustics or sound insulation, artificial lighting or solar penetration, structural analysis or checking of building regulation can use the geometrical, specificational and relational information contained in five survey files. Cost evaluation is one of the appraisals that can make an extensive use of surveyed data, and will be described in the next section.

7.4 THE COST MODELLER.

The diagram in Figure 7.3.3 should describe GABLE features used to arrive at a building cost model.

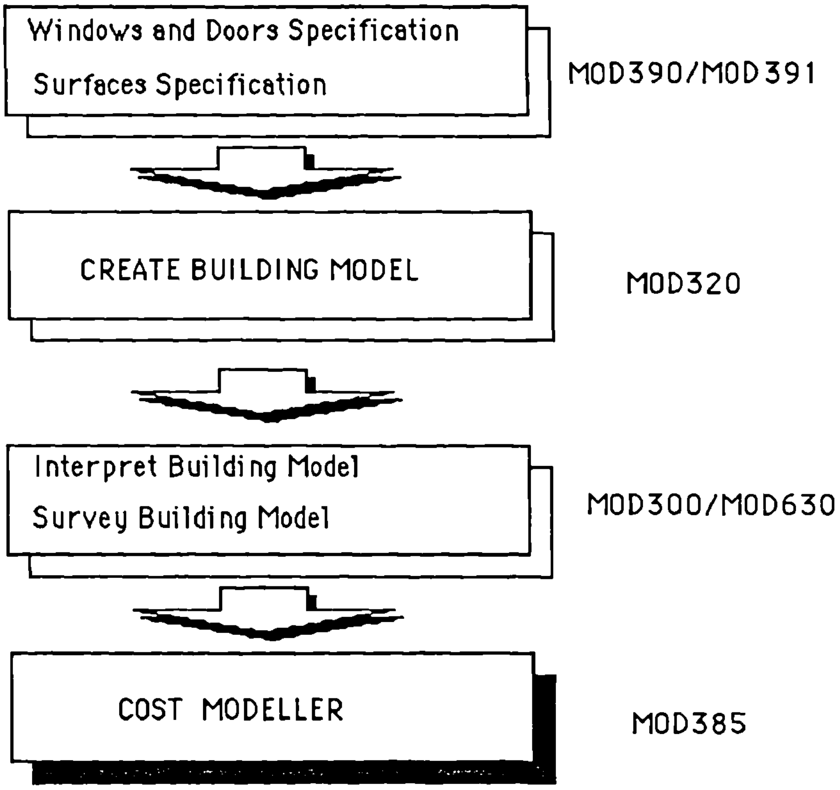


Figure 7.3.3 - GABLE modules pre-requisits for COST MODELLING.

The next sections will describing the work involved in the development GABLE - Module 385, or in other word the 'COST MODELLER'.

7.4.1 Concept description.

The development of a building cost modelling module in the GABLE CAD system was aimed at enabling practical early cost evaluation. This new module is to contribute in promoting cost data analysis to discover systematic cost variations, with changes in design parameters contained in GABLE building models.

For this purpose, it was decided to analyse a piece of existing cost modelling software, to get a better understanding of what is being done in the field.

A quantity surveying practice was approached, and an agreement was made to make available a cost modelling program that was initially developed for the own use of that practice. The software came with a user manual, but consultations were regularly exchanged to make sure that the package was used correctly.

It was soon revealed that "Bucknall Austin Plc." cost model program suited our aims, and the next step consisted of analysing it for an eventual adaptation to GABLE CAD system. It appeared from that analysis that the software had several features organised around a central cost modelling 'unit'. The diagram should in Figure 7.3.4 illustrates the articulation of its main features.

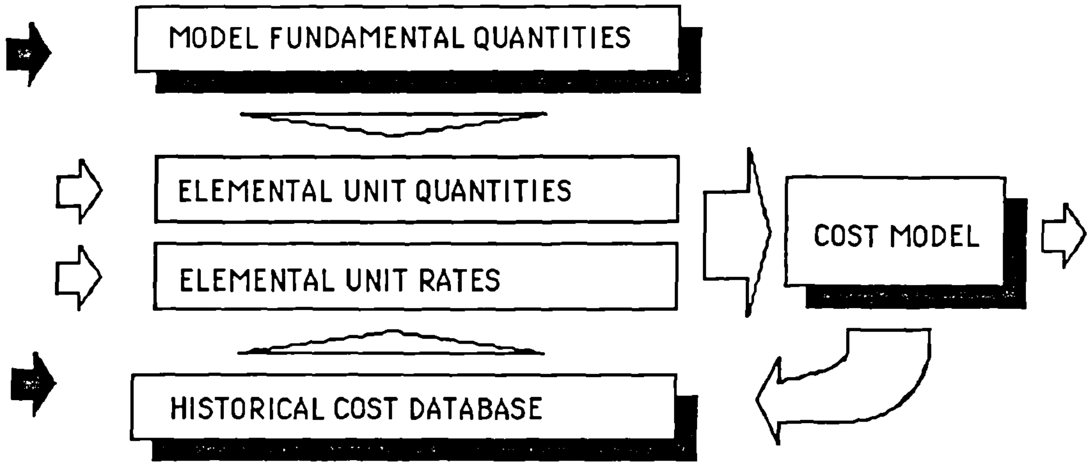


Figure 7.3.4 - Main features of 'Bucknall Austin' cost modelling system.

Bucknall Austin Plc. have based their cost consultancy strategy around ELEMENTAL COST PLANS, which are prepared for all schemes. The role of the element cost plan as a cost control, cost monitoring and cost checking tool is a fundamental part of the practice strategy (Patchell, B.R.T. 1987).

The practice recognised the need for an additional form of cost planning, else then abbreviated quantities techniques. The argument was that budget should never be based on gross floor area rates, as they do not take into account shape, form or volume. Also, abbreviated quantities cost plans require considerable details and variations in options, which require remeasurement as there is little re-use of the abbreviated quantities.
A very similar approach to cost analysis was taken in the development of Module 385. However, more emphasis was put on the measurements production process in conjunction with the CAD system. The number of items of 'fundamental' building works quantities, was dictated by the type of building elements that GABLE BMS can support.

Technically, the cost modeller attempts to make to best of GABLE's ability to provide specificational, relational and locational data with each building element of the building model. Thus, fundamental building quantities, can be located in the building (i.e. : floor level and/or space location), and can be related to each other or to the space and/or floor they belong to (i.e. : windows and doors relation to the walls they lie in, or doors and windows relation to a particular room on a specific level of the building), and can be associated to a specification of building material.

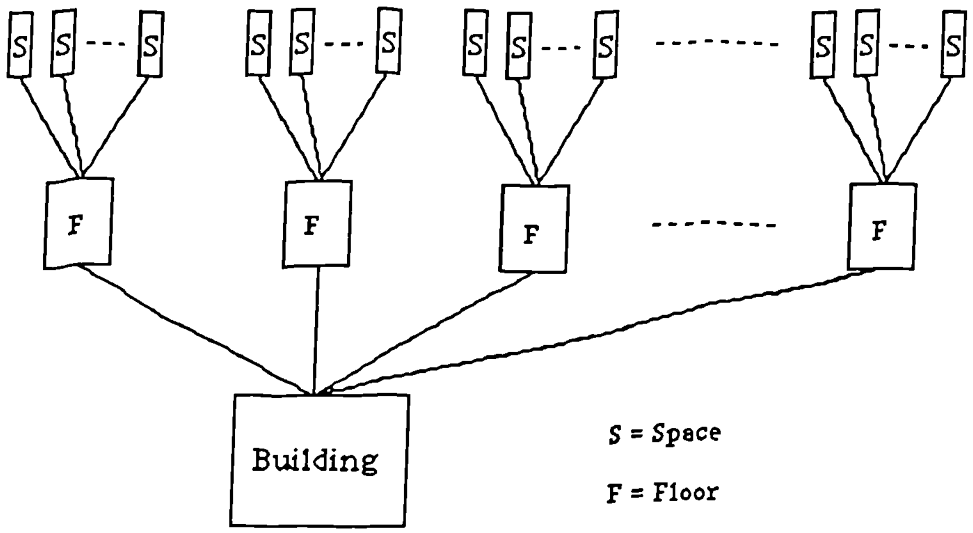


Figure 7.3.5 - Hierarchical tree of the elements relation to the building.

A typical sequence of events during a cost modelling cession on GABLE CAD system would be as follow. The designer would explore the initial design form, and an 'intelligent' three dimensional building model would be developed. This model is then automatically surveyed creating a data base of use to many evaluative routines. In turn, this data (i.e. : LEVEL, ROOM, ISAAC, LAYOUT and OPENING files) is used by the cost modeller routines to create different files.

The user is given the option to group the building measurements around spaces or floor levels, before producing a 'Building Elemental Model' (BEM). The building elemental model is the ultimate representation of the building measurements for the cost analysis, it currently uses the Ci/Sfb format but could equally well be in BCIS or any other similar standard. It contains information on different building works, and the quantities of building elements involved in the construction of each of those. A sample of a BEM file can be found in Appendix 10. To arrive to a BEM file, the user will have to

```
:GABLEJOB:TAMI:DMS:SFQ

RECORD  1
  CODE            : 0
  LEVEL           : 0
  GROSS AREA      : 9.46
  SPACE TITLE     : CONSERVATORY
  SPACE NUMBER    : 1
  SPACE TYPE      : 0
  EXT WALL BEDS   : 2.13
  INT WALL BEDS   : .56
  EXT STOREY HT   : 3000.06
  INT STOREY HT   : 2999.89
  EXT WALL AREA   : 28.39
  INT WALL AREA   : 15.04
  EXT WIND AREA   : 9.18
  INT WIND AREA   : 3.60
  EXT DOOR AREA   : 1.89
  INT DOOR AREA   : 1.57
  CEILING AREA    : 9.46
  SUSP. CEILING   : 0
  GR. SLAB AREA   : 12.15
  ROOF AREA       : 0

RECORD  2
  CODE            : 1
  LEVEL           : 0
  GROSS AREA      : 18.08
  SPACE TITLE     : ROOM
  SPACE NUMBER    : 2
  SPACE TYPE      : 0
  EXT WALL BEDS   : 2.00
  INT WALL BEDS   : .66
  EXT STOREY HT   : 3000.06
  INT STOREY HT   : 3000.06
  EXT WALL AREA   : 26.66
  INT WALL AREA   : 25.69
  EXT WIND AREA   : 0
  INT WIND AREA   : 3.60
  EXT DOOR AREA   : 0
  INT DOOR AREA   : 1.57
  CEILING AREA    : 18.08
  SUSP. CEILING   : 0
  GR. SLAB AREA   : 20.74
  ROOF AREA       : 0

RECORD  3
  CODE            : 2
  LEVEL           : 0
  GROSS AREA      : 2.23
  SPACE TITLE     : CORRIDOR
  SPACE NUMBER    : 3
  SPACE TYPE      : 0
  EXT WALL BEDS   : .64
  INT WALL BEDS   : .23
  EXT STOREY HT   : 3000.06
  INT STOREY HT   : 3000.00
  EXT WALL AREA   : 8.55
  INT WALL AREA   : 13.80
  EXT WIND AREA   : 0
  INT WIND AREA   : 0
  EXT DOOR AREA   : 0
  INT DOOR AREA   : 3.15
  CEILING AREA    : 2.23
  SUSP. CEILING   : 0
  GR. SLAB AREA   : 3.10
  ROOF AREA       : 0

RECORD  4
  CODE            : 3
  LEVEL           : 0
  GROSS AREA      : 6.48
  SPACE TITLE     : BATH ROOM
  SPACE NUMBER    : 4
  SPACE TYPE      : 0
  EXT WALL BEDS   : 1.22
  INT WALL BEDS   : .26
  EXT STOREY HT   : 3000.06
  INT STOREY HT   : 3000.00
  EXT WALL AREA   : 16.31
  INT WALL AREA   : 15.34
  EXT WIND AREA   : .22
  INT WIND AREA   : 0
  EXT DOOR AREA   : 0
  INT DOOR AREA   : 1.57
  CEILING AREA    : 6.48
  SUSP. CEILING   : 0
  GR. SLAB AREA   : 7.96
  ROOF AREA       : 0

RECORD  5
  CODE            : 4
  LEVEL           : 0
  GROSS AREA      : 12.17
  SPACE TITLE     : KITCHEN
  SPACE NUMBER    : 5
  SPACE TYPE      : 0
  EXT WALL BEDS   : 1.36
  INT WALL BEDS   : .49
  EXT STOREY HT   : 3000.06
  INT STOREY HT   : 2999.92
  EXT WALL AREA   : 18.11
  INT WALL AREA   : 24.71
  EXT WIND AREA   : 1.80
  INT WIND AREA   : 0
  EXT DOOR AREA   : 0
  INT DOOR AREA   : 4.72
  CEILING AREA    : 12.17
  SUSP. CEILING   : 0
  GR. SLAB AREA   : 14.02
  ROOF AREA       : 0
```

Figure 7.3.3 - Sample of the COST MODELLER - SFQ file.

```
:GABLEJOB:TAMI:DMS:FFQ

RECORD  1
  CODE             : 0
  LEVEL            : 0
  GROSS AREA       : 50.74
  GRND SLAB AREA   : 58.10
  LET USE AREA     : 48.43
  EXT STOREY HT    : 1534
  INT STOREY HT    : 1534
  EXT WALL AREA    : 98.02
  INT WALL AREA    : .00
  EXT WIND AREA    : 11.20
  INT WIND AREA    : 3.60
  EXT DOOR AREA    : 1.89
  INT DOOR AREA    : 6.30
  ROOF AREA        : 0
  CEILING AREA     : 0
  STRCASE FLIGHT   : 0
  LIFT FLIGHTS     : 0
  SUSPENDED CEAR   : 0
  SANITARY FITNS   : 3
  NO OF SPACES     : 5
```

Fig 7.3.4 (a) - FFQ file for the all building model.

```
:GABLEJOB:TAMI:DMS:FFQ_TEST

RECORD  1
  CODE             : 0
  LEVEL            : 0
  GROSS AREA       : 40.60
  GRND SLAB AREA   : 55.41
  LET USE AREA     : 38.96
  EXT STOREY HT    : 3000
  INT STOREY HT    : 3000
  EXT WALL AREA    : 69.63
  INT WALL AREA    : 79.54
  EXT WIND AREA    : 2.02
  INT WIND AREA    : 0
  EXT DOOR AREA    : 0
  INT DOOR AREA    : 5.51
  ROOF AREA        : 0
  CEILING AREA     : 38.96
  STRCASE FLIGHT   : 0
  LIFT FLIGHTS     : 0
  SUSPENDED CEAR   : 0
  SANITARY FITNS   : 3
  NO OF SPACES     : -4
```

Fig 7.3.4 (b) - FFQ file for the model without conservatory.

create a Floor Fundamental Quantities (FFQ) file, in which each record contains data about a particular floor level in the building (see Appendix 10.6). This could be the descriptive model used for costing.

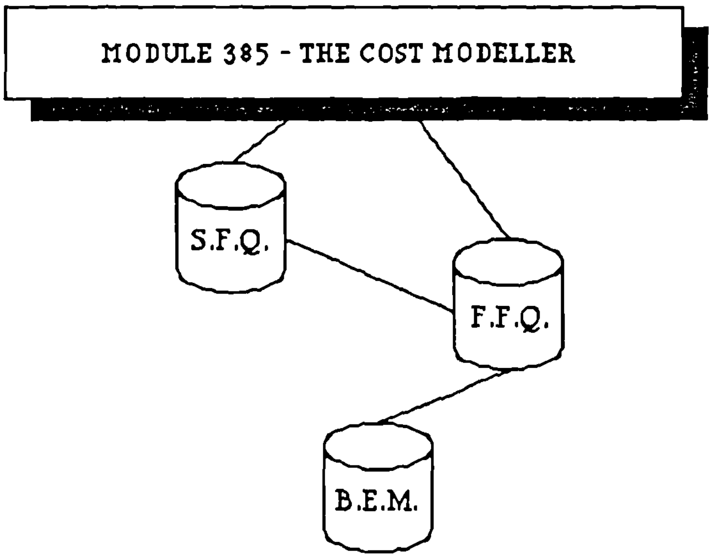


Figure 7.3.8 - The COST MODELLER file production sequence.

Since GABLE can support the modelling of buildings that have up to one hundred floors, the FFQ file will contain up to that number of records. However, the user can combine any selected number of floors to be included in the Building Elemental Model. This last option is the first move towards the 'designing to a cost' facility described earlier. Although, it should be used carefully since the interpretation of spaces three dimensional adjacencie is not provided.

7.4.2 Space based cost modelling.

Alternatively, the user can create a Building Elemental Model, out of a Space Fundamental Quantities (SFQ) file. The techniques developed in the corse of this research, have shown the potential of offering space based cost modelling. It basically consists of grouping building elements that belong to the same spatio-functional unit of the design solution. In the SFQ file each record describes an individual space rather than a complete floor level (see Appendix 10.5). By relating building elements to spaces, and by providing means to maintain these relationships, it is made possible to appreciate the cost implications of space design modelling.

At early design stage it might be useful to include only particular spaces on each floor level. In which case, quite sophisticated corrective calculations can be made by the system because of the way it retains spatial location data. If, for example, a particular space is omitted from the data, then the walls specification of its neighbouring spaces may change from internal to external, if appropriate, entirely automatically. Thus experiments can be conducted without the need to return to the three dimensional building model while still maintaining the constructional logic of the building. This has been possible with the full combination of locational, specificational and relational data which can be obtained from the 'intelligent' building modelling system described earlier in the thesis.

7.4.3 Cost analysis.

The final link in the chain involves the inclusion of cost rate data applied to the Building Elemental Model. Therefore, a data base of rates must be established from which appropriate rates may be selected. These rates are elemental unit costs, and are worked out by measuring existing buildings, or by automatically surveying previously designed ones. In either event the measurement method has to be similar to the one used by the 'cost modeller',

```
CODE  ELEMENT                      UNIT QUANTITY  PROJECT   S/ ELEMENTAL  PERCENT
                                                  RATE      /T COST

(1-)  SUBSTRUCTURE_ _ ____ _              0          0       I      0       0
(11)  ground                       m2    58.10      25.41    I   1476       6.6
(13)  floor beds                   m2    58.10       1.91    I    111        .5
(16)  retaining walls, founds      m2    58.10      89.32    I   5189      23.1
(17)  pile foundations             Nr     0          0       I      0       0
------------- ---- --  ------ - ---- ------ -------- -- --- -----         -  -
                                                                 6776      30.5

CODE  ELEMENT                      UNIT QUANTITY  PROJECT   S/ ELEMENTAL  PERCENT
                                                  RATE      /T COST

(2-)  STRUCTURE_  _                       0          0       I      0       0
(21)  external walls area          m2    81.93      46.37    I   3938      17.7
(22)  internal walls area          m2    40.98      14.73    I    604       2.7
(23)  floors                       m2     0         27.30    I      0       0
(24)  stairs                       Nr     0        659.36    I      0       0
(27)  roofs area                   m2     0         23.61    I      0       0
(28)  frames                       m2    50.74       0       I      0       0
-------------------------- --- --- -   -- -- - - -   ------ - -         --- --
                                                                 4547      20.4

CODE  ELEMENT                      UNIT QUANTITY  PROJECT   S/ ELEMENTAL  PERCENT
                                                  RATE      /T COST

(3-)  COMPLETION_________  ___            0          0       I      0       0
(31)  openings in ext. wall        m2    13.09     160.60    I   2102       9.5
(32)  openings in int. wall        m2     6.30      76.71    I    483       2.2
(33)  floor openings               m2     0          0       I      0       0
(34)  balustrading                 Nr     0        480.00    I      0       0
(35)  suspended ceilings area      m2     0          0       I      0       0
(37)  rooflights                   m2     0          0       I      0       0
--------  -  ---------------------- ---------------- -------- - - --
                                                                 2585      11.7

CODE  ELEMENT                      UNIT QUANTITY  PROJECT   S/ ELEMENTAL  PERCENT
                                                  RATE      /T COST

(4-)  FINISHES_________________           0          0       I      0       0
(41)  external walls area          m2    84.93       0       I      0       0
(42)  internal walls area          m2   166.90       3.98    I    664       3.0
(43)  floor finishes               m2    48.43      15.71    I    761       3.4
(44)  stair finishes               Nr     0        274.95    I      0       0
(45)  ceiling finishes             m2    48.43       4.91    I    239       1.1
(47)  roof finishes                m2     0         19.89    I      0       0
---------------------------------------------------- --- -------------------
                                                                 1664       7.5


CODE  ELEMENT                      UNIT QUANTITY  PROJECT   S/ ELEMENTAL  PERCENT
                                                  RATE      /T COST

(5-)  SERVICES________                    0          0       I      0       0
(51)  refuse                       m2    50.74       0       I      0       0
(52)  waste disposal               m2    50.74       6.00    I    301       1.4
(53)  hot and cold water           m2    50.74       4.69    I    238       1.1
(54)  gases supply                 m2    50.74       1.56    I     79        .4
(55)  space cooling                m2    50.74       0       I      0       0
(56)  space heating                m2    50.74      20.34    I   1032       4.6
(57)  air conditioning             m2    50.74       0       I      0       0
----------- ------------------------ -----------------------------------------
                                                                 1653       7.5
```

```
CODE  ELEMENT                      UNIT QUANTITY  PROJECT   S/ ELEMENTAL  PERCENT
                                                  RATE      /T COST

(6-)  INSTALLATION  _                     0          0       I      0       0
(62)  power                        m2    50.74      13.76    I    698       3.1
(63)  lighting                     m2    50.74       0       I      0       0
(64)  communication                m2    50.74       0       I      0       0
(66)  transport                    Nr    50.74       0       I      0       0
(68)  security                     m2    50.74       0       I      0       0
 -     --     -   --  -- -    ------ --  --     - -----    - ----------
                                                                  698       3.1

CODE  ELEMENT                      UNIT QUANTITY  PROJECT   S/ ELEMENTAL  PERCENT
                                                  RATE      /T COST

(7 )  FIXED FITTINGS                      0          0       I      0       0
(71)  circulation                  m2    50.74       0       I      0       0
(72)  general fittings             m2    50.74      16.94    I    861       3.9
(73)  culinary                     m2    50.74       0       I      0       0
(74)  sanitary,hygiene             Nr     3.00      87.13    I    261       1.2
(75)  cleaning,maintenance         m2    50.74       0       I      0       0
(76)  storage,screening            m2    50.71       0       I      0       0
(77)  special activity             m2    50.74       0       I      0       0
 - ---  --  ----              - - - -- - -- --  ---    ---------- --  --
                                                                 1121       5.1

CODE  ELEMENT                      UNIT QUANTITY  PROJECT   S/ ELEMENTAL  PERCENT
                                                  RATE      /T COST

(8-)  LOOSE FITTINGS_                     0          0       I      0       0
(81)  circulation                  m2    50.71       0       I      0       0
(82)  general fittings             m2    50.74       0       I      0       0
(83)  culinary                     m2    50.74       0       I      0       0
(84)  sanitary,hygiene             m2    50.74       0       I      0       0
(85)  cleaning,maintenance         m2    50.74       0       I      0       0
(86)  storage,screening            m2    50.74       0       I      0       0
(87)  special activity             m2    50.74       0       I      0       0
- ------- --------   ---- ----- ----- ---------  - --- ------------ ---- --
                                                                    0       0

CODE  ELEMENT                      UNIT QUANTITY  PROJECT   S/ ELEMENTAL  PERCENT
                                                  RATE      /T COST

(9-)  SITEWORKS_______                    0          0       I      0       0
(91)  ground preparations          m2    58.10       0       I      0       0
(92)  minor structures             m2    58.10       0       I      0       0
(93)  enclosures                   m2    58.10       0       I      0       0
(94)  surface treatment            m2    58.10      36.54    I   2123       9.6
(95)  drainage                     m2    50.74      13.48    I    684       3.1
(96)  electric services            m2    50.74       6.97    I    354       1.6
(97)  fittings                     m2    50.74       0       I      0       0
(98)  special                      m2    50.74       0       I      0       0
-----------------------------------------------------------------------------
                                                                 3161      14.3

CODE  ELEMENT                      UNIT QUANTITY  PROJECT   S/ ELEMENTAL  PERCENT
                                                  RATE      /T COST

(10-) PRELIMINARIES_ __ __  _ __          0          0       I      0       0
(101) preliminaries                m2     0          0       I      0       0
(102) contingencies                m2     0          0       I      0       0
(103) price and design             m2     0          0       I      0       0
-----------------------------------------------------------------------------
                                                                    0       0

BUILDING ELEMENTAL COST MODEL

-----------------------------------------------------------------------------
TOTAL =                                                         22200     100
```

Figure 7.4.1 (a) - BEM file report for the all building model.

```
CODE  ELEMENT                      UNIT QUANTITY  PROJECT  S/ ELEMENTAL  PERCENT
                                                  RATE     /T COST

(1-)  SUBSTRUCTURE ______________          0        0       I     0         0
(11)  ground                       m2     55.41    25.41    I   1408       7.9
(13)  floor beds                   m2     55.41     1.91    I    106        .6
(16)  retaining walls, founds      m2     55.41    89.32    I   4949      27.7
(17)  pile foundations             Nr      0        0       I     0         0
--------------------------------------------------------------------------------
                                                                6463      36.2

CODE  ELEMENT                      UNIT QUANTITY  PROJECT  S/ ELEMENTAL  PERCENT
                                                  RATE     /T COST

(2-)  STRUCTURE _________________          0        0       I     0         0
(21)  external walls area          m2     67.61    46.37    I   3135      17.5
(22)  internal walls area          m2     31.26    14.73    I    505       2.8
(23)  floors                       m2      0       22.30    I     0         0
(24)  stairs                       Nr      0      659.36    I     0         0
(27)  roofs area                   m2      0       72.61    I     0         0
(28)  frames                       m2     40.60     0       I     0         0
--------------------------------------------------------------------------------
                                                                3640      20.3

CODE  ELEMENT                      UNIT QUANTITY  PROJECT  S/ ELEMENTAL  PERCENT
                                                  RATE     /T COST

(3-)  COMPLETION ________________          0        0       I     0         0
(31)  openings in ext. walls       m2      2.02   160.60    I    324       1.8
(32)  openings in int. walls       m2      6.17    76.71    I    473       2.4
(33)  floor openings               m2      0        0       I     0         0
(34)  balustrading                 Nr      0      480.00    I     0         0
(35)  suspended ceilings area      m2      0        0       I     0         0
(37)  rooflights                   m2      0        0       I     0         0
--------------------------------------------------------------------------------
                                                                 747       4.2

CODE  ELEMENT                      UNIT QUANTITY  PROJECT  S/ ELEMENTAL  PERCENT
                                                  RATE     /T COST

(4-)  FINISHES __________________          0        0       I     0         0
(41)  external walls area          m2     67.61     0       I     0         0
(42)  internal walls area          m2    136.13     3.98    I    542       3.0
(43)  floor finishes               m2     38.96    15.71    I    612       3.4
(44)  stair finishes               Nr      0      274.95    I     0         0
(45)  ceiling finishes             m2     38.96     4.94    I    192       1.1
(47)  roof finishes                m2      0       19.89    I     0         0
--------------------------------------------------------------------------------
                                                                1346       7.5

CODE  ELEMENT                      UNIT QUANTITY  PROJECT  S/ ELEMENTAL  PERCENT
                                                  RATE     /T COST

(5-)  SERVICES __________________          0        0       I     0         0
(51)  refuse                       m2     40.60     0       I     0         0
(52)  waste disposal               m2     40.60     6.00    I    244       1.4
(53)  hot and cold water           m2     40.60     4.69    I    190       1.1
(54)  gases supply                 m2     40.60     1.56    I     63        .4
(55)  space cooling                m2     40.60     0       I     0         0
(56)  space heating                m2     40.60    20.34    I    826       4.6
(57)  air conditioning             m2     40.60     0       I     0         0
--------------------------------------------------------------------------------
                                                                1323       7.5

CODE  ELEMENT                      UNIT QUANTITY  PROJECT  S/ ELEMENTAL  PERCENT
                                                  RATE     /T COST

(6-)  INSTALLATION                         0        0       I     0         0
(61)  power                        m2     40.60    13.76    I    559       3.1
(63)  lighting                     m2     40.60     0       I     0         0
(64)  communication                m2     40.60     0       I     0         0
(66)  transport                    Nr     40.60     0       I     0         0
(68)  security                     m2     40.60     0       I     0         0
--------------------------------------------------------------------------------
                                                                 559       3.1

CODE  ELEMENT                      UNIT QUANTITY  PROJECT  S/ ELEMENTAL  PERCENT
                                                  RATE     /T COST

(7-)  FIXED FITTINGS                       0        0       I     0         0
(71)  circulation                  m2     40.60     0       I     0         0
(72)  general fittings             m2     40.60    16.94    I    688       3.8
(73)  culinary                     m2     40.60     0       I     0         0
(74)  sanitary,hygiene             Nr      3.00    87.13    I    261       1.5
(75)  cleaning,maintenance         m2     40.60     0       I     0         0
(76)  storage,& screening          m2     40.60     0       I     0         0
(77)  special activity             m2     40.60     0       I     0         0
--------------------------------------------------------------------------------
                                                                 949       5.3

CODE  ELEMENT                      UNIT QUANTITY  PROJECT  S/ ELEMENTAL  PERCENT
                                                  RATE     /T COST

(8-)  LOOSE FITTINGS                       0        0       I     0         0
(81)  circulation                  m2     40.60     0       I     0         0
(82)  general fittings             m2     40.60     0       I     0         0
(83)  culinary                     m2     40.60     0       I     0         0
(84)  sanitary,hygiene             m2     40.60     0       I     0         0
(85)  cleaning,maintenance         m2     40.60     0       I     0         0
(86)  storage,screening            m2     40.60     0       I     0         0
(87)  special activity             m2     40.60     0       I     0         0
--------------------------------------------------------------------------------
                                                                   0         0

CODE  ELEMENT                      UNIT QUANTITY  PROJECT  S/ ELEMENTAL  PERCENT
                                                  RATE     /T COST

(9-)  SITEWORKS _________________          0        0       I     0         0
(91)  ground preparationa          m2     55.41     0       I     0         0
(92)  minor structures             m2     55.41     0       I     0         0
(93)  enclosures                   m2     55.41     0       I     0         0
(94)  surface treatment            m2     55.41    36.54    I   2025      11.3
(95)  drainage                     m2     40.60    13.48    I    547       3.1
(96)  electric services            m2     40.60     6.97    I    283       1.6
(97)  fittings                     m2     40.60     0       I     0         0
(98)  special                      m2     40.60     0       I     0         0
--------------------------------------------------------------------------------
                                                                2855      16

CODE  ELEMENT                      UNIT QUANTITY  PROJECT  S/ ELEMENTAL  PERCENT
                                                  RATE     /T COST

(10-) PRELIMINARIES _____________          0        0       I     0         0
(101) preliminaries                m2      0        0       I     0         0
(102) contingencies                m2      0        0       I     0         0
(103) price and design             m2      0        0       I     0         0
--------------------------------------------------------------------------------
                                                                   0         0

BUILDING ELEMENTAL COST MODEL

--------------------------------------------------------------------------------
                                                               17882       100
```

Figure 7.4.1 (b) - BEM file report for the building without conservatory.

and of compatible format (i.e.: Ci/Sfb or BCIS in the context of this research). At this stage all projects in the cost data bank are categorised by building types, and given a general description.

For example, a project can belong to the category of 'residential buildings', and described according to its size, shape, type of construction, installations, site abnormals, location and tender type. All of this data is held in standard data base files with the same format as all quantities and specification data. This enables us to use standard data base enquiry , selection and cross referencing techniques to find the most suitable project from which to select the cost data. It then simply remains for the cost modelling system to extract the appropriate rates from this project and apply them to the currently modeled one. The Building Elemental Quantities file will see its quantities multiplied by the imported rates, to arrive at a final cost model for the building.

In fact, the multiplication operation in this case is made more sophisticated by attaching a price adjustment factor to the rate after it is imported from the cost data base. In addition to allowing for such standard items as price rises or regional variation, this also allows for further modelling possibilities from within the Building Elemental Model file. Operations such as adjusting the rate of a particular building element are thus made possible simply by increasing or decreasing the price adjustment factor for that item, without altering the original imported rate.
It is suggested that the cost data bank should be organised in such a way, that any imported cost data is adjusted before use in a cost analysis.

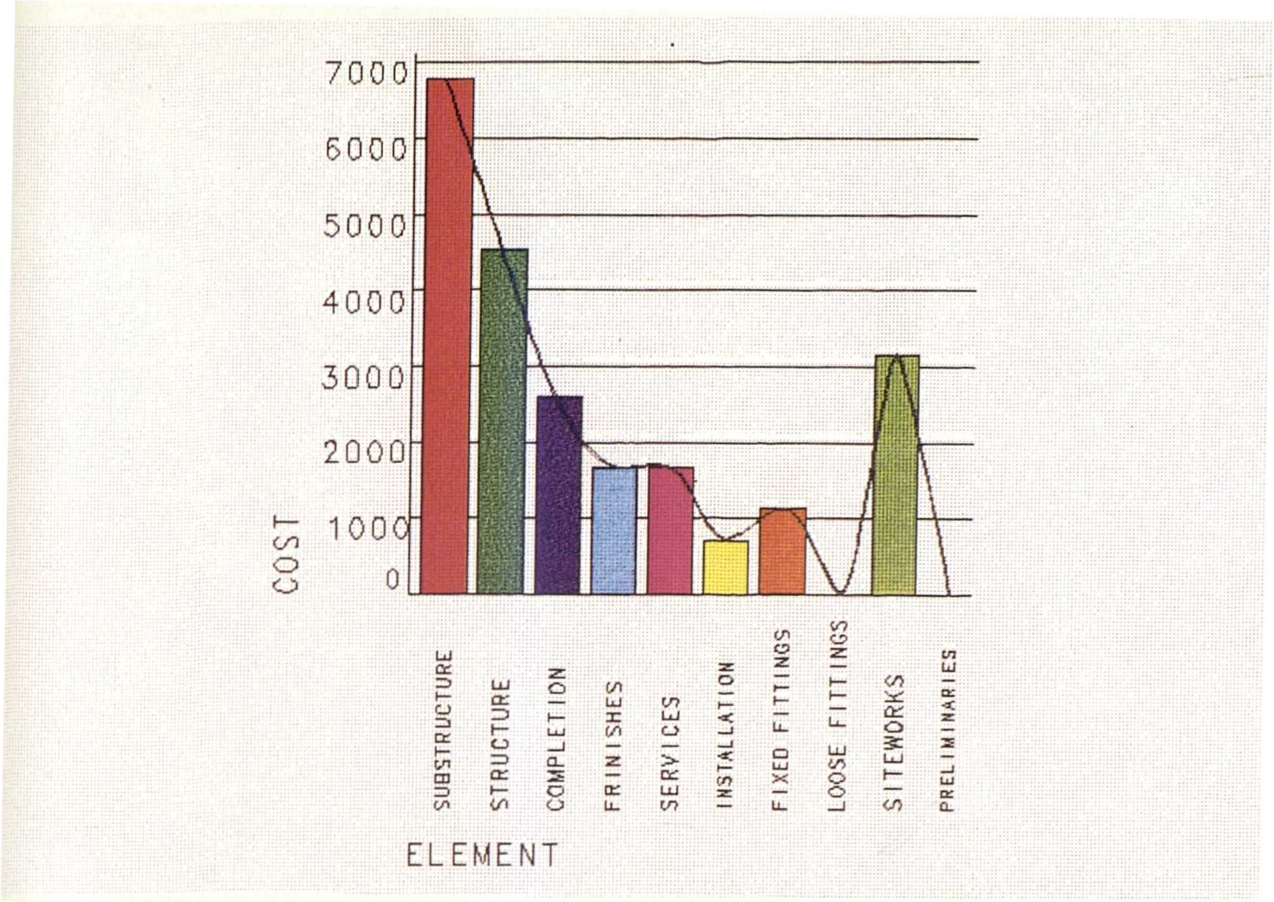


Figure 7.4.2 (a) - Bar-chart of cost analysis for the all building model.

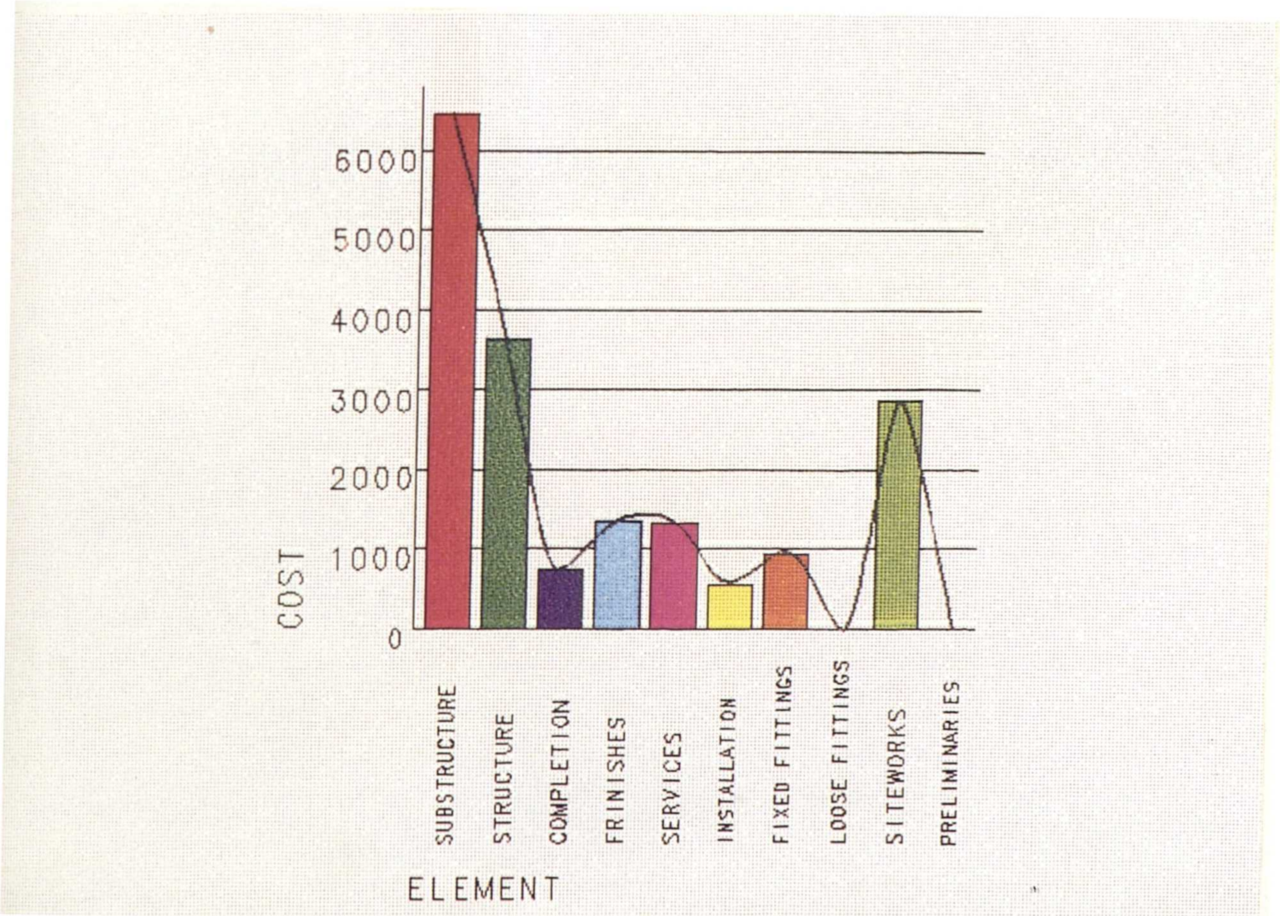


Figure 7.4.2 (b) - Bar-char for cost analysis for the building without conservatory.

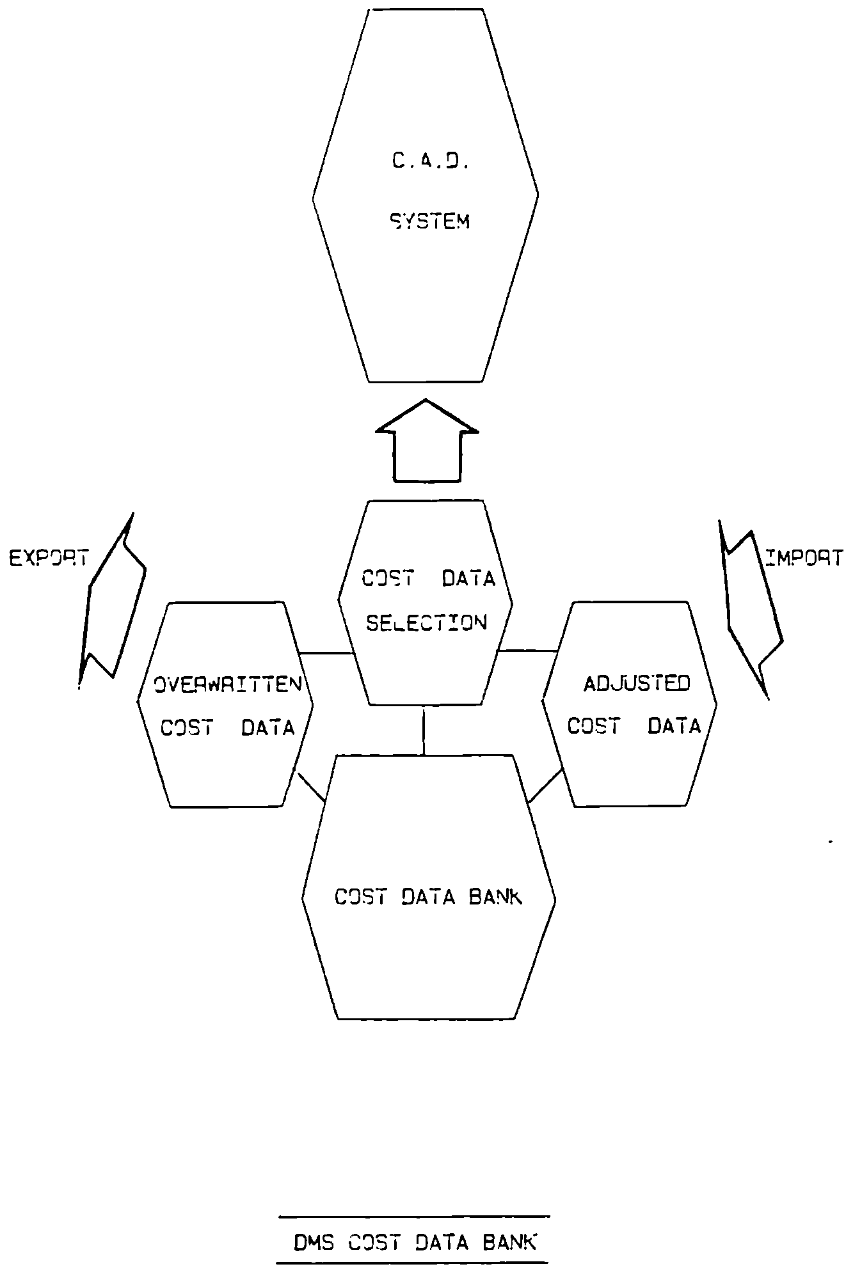


Figure 7.4.3 - Cost data bank interaction with the CAAD system.

This is to control the distortion of rates generated using a particular method of measurement, and the applied to an other measurement method. The diagram shown in Figure 7.4.3, describes the cost data bank interaction with the CAD system and the cost information supplier.

The sequence of events described above can be repeated for a different combination of spaces (or rooms) contained in the building model. For example, a Floor Fundamental Quantities file can be created out of a selected number of rooms. In Figure 7.3.4 (b), the FFQ file represents the building model quantities without the 'conservatory'. From that file a cost analysis is performed to estimate the cost variations due the change in the design (i.e. the omission of one space). It is important to note that the geometrical model has not been altered, instead, the survey data will change according to the new design layout situation.
A Building Elemental Model file is then created, and rates imported from the cost data bank. A report of the new BEM file is shown in figure 7.4.1 (b). If it is the same rates as the previous cost model that are imported, which is the case in this particular experiment, it might be useful to compare the two BEM files. In other words compare the cost distribution in the cost model representing all the spaces contained in the building model, with the cost model representing the building model without the 'conservatory'. To ease such a comparison, histograms have been produced of each BEM file, and are shown in Figures 7.4.2 (a) and 7.4.2 (b). It can be seen on these diagrams, that it is the completion cost that has the most dramatically been cut down. For the obvious reason that the conservatory with its large windows, bears a large proportion of the completion cost.

This approach to cost modelling has made 'cost location' a realistic option during design costing. Nevertheless, there are two major deficiencies in it. Firstly the accuracy of descriptive model techniques is not as high as may be desired, however there seems to be no real answer to this at early design stage. Secondly, this cost modelling system can be used only once a sensible three dimensional building model exists in the CAAD system.

Behind this work lies an attempt to increase our capability to predict building cost whilst the building design is still fluid enough to allow for changes which might be suggested by cost modelling, and to be incorporated in the design as smoothly as possible.

PART D -
DISCUSSION AND CONCLUSION.

Chapter 8 : DESIGN OFFICE AUTOMATION AND C.A.A.D. INTEGRATION.

8.1 LINKING GRAPHICS, WORDS AND NUMBERS.

The roles of the computer in the design office have been described in section 4.1.2., and its use as a generative and/or analytical tool was discussed. This research work on computer aided building design and evaluation has highlighted the importance of relating graphical information to numerical and textual information.

In the particular case of cost modelling, we have seen that by locating building elements spatially and relating them to each other, it is possible to estimate and model consistently the cost of a design solution. The design solution is modeled graphically, with the visualisation of the CAD system helping to reveal any error in the form of unexpected geometry. The implicit logic of relationship between the building elements, built in the computer model, is used to monitor the modelling of textual and numerical information attached to the model. Words and numbers are either attributes describing intrinsic properties of building elements, or indicating in what context they were used. The storage of the graphical information and the numerical/textual information is shared between a graphical database, and a standard general purpose database (in our case of the relational type). This arrangement is very promising for media integration in design office automation.

8.1.1 Design office automation.

The separation between drawing and, say, specification writing, is largely the result of using two different media, graphical and textual/numerical. The isolation of these tasks is not always desirable. We have seen that, when designing, different kinds of information are handled. Lists of materials, such as windows, doors, hardware and finishes, are as important to design as drawings; they are often selected by consulting manufacturer's catalogues, which in turn use drawings to give dimensions and describe their products. Design specifications are written by adapting this existing information to a specific project. For example, to estimate the cost of a construction, specific quantity information has to be extracted from drawings, and price information from a costing database, and then combine the two.

In designing and documenting a single decision or detail, it is often drawn, listed, scheduled, engineered, specified and costed in different steps. In doing this, there can be a tremendous amount of redundant work involved in handling a single building element.

Computers can be useful in bridging some of the gaps imposed by different media. In this respect, Crosley (1988) suggested that, first, computers can enable the transfer of information from one medium to another, eliminating the need to recreate it. Second, some computers can do more than one thing at once, a process called 'multi-tasking'. By combining these two capabilities, things can be done that are impossible to envisage using traditional methods. The beauty of such a thing is that decisions can be made while designing, based on information accessed very rapidly, which could be modified later, then reported in textual or numerical form.

When talking about design office automation and the linking of graphical, textual, and numerical information, we ought to distinguish between processing information during conceptual design, and presenting that same information after the design is completed.

It has been show in section 4.1.1, that the automation of the design process as such is not suitable. First, because design decisions are largely concern with value judgement, and second, because the machines used for design automation are at the moment processing information sequentially, whereas design rely on forms of parallel (or lateral) thinking and multi-variable problem solving. However, computers can be very useful during early conceptual design stages, if they provide a suitable environment for problem re-structuring, and allow the overlapping of the different design stage (i.e. : analysis, syntheses and evaluation). It is to the latter issue, that this research work has aimed to contribute, and while doing so, it has been revealed that it is essential to be able to link graphics, words and numbers, to achieve any improvement of the computer's role in conceptual design.

For the successful processing of information among various members of a multi-disciplinary design team, during the conceptual stages of design, it is essential to allow the design information to be exchanged in its various forms (i.e.: Textual, numerical, and graphical (or pictorial)). More importantly, the implicit relational information contained in the two and three dimensional computer models, must be consistently kept whenever an information exchange occurs. This has been demonstrated earlier, and is considered as key element of progress in the field of design office automation.

On the other hand, the quality of design information presentation rely also considerably on the ability of a computer system to handle different media. Integrated software applications, which have been available for some time, are programs that include word processors, database managers and spread sheets. They are supposed to offer the ideal solution for data transfer between different applications. Whereas, in the architectural and engineering office, most non-graphical tasks can be done with the help of integrated packages. These become of little help when it is required to handle graphical information.

A solution to this problem is offered by what is called *Hypermedia.* Before describing a hypermedia environment, it is better to first introduce the *Hypertext* concept, which is at the source of this new technology. Ted Nelson (1967), one of the pioneers of hypertext, defined it as :

> "a combination of natural language text with the computer's capacity for interactive branching, or dynamic display... of a nonlinear text...which cannot be printed conveniently on a conventional page."

More recently, Conklin, J. (1987) in a comprehensive survey of the subject writes :

> "The concept of hypertext is quite simple: Windows on the screen are associated with objects in a database, and links are provided between these objects, both graphically (as labeled tokens) and in the database (as pointers)"(p. 17).

Hypertext could be described as being, in its original form, a tool that provides access to a large text or document, with mechanisms to make links between any two pieces of information in the system. However, with todays computer technology, a logical extension to Hypertext has produced Hypermedia. Which instead of linking just text, users can link to other media, such as graphics, video, spreadsheets, animations and voice. In short, users of hypermedia systems can link together information of any media type provided by the current technology.

There has been attempts to make use of a Hypertext system for CAD applications (Delisle, N. and Schwartz, M. 1986), but so far, the techniques of this new concept seem to be restricted to word processing and data base management applications.

It would certainly be interesting to see CAAD benefit from Hypermedia technology developments. Nevertheless, there are already information technology products available, based on similar techniques, that can be profitable for designers. Some of them are described next.

8.1.2 Multi-media CAAD applications.

A variety of applications can be developed based on the interaction of graphical and non graphical information in a CAD environment. Some have recently emerged from such combinations, like **Facility Management** and **Geographical Information Systems.**

The aim of the CAD system in a facilities management package is to enable an office manager to 'see', by looking into a database, a drawing of a particular floor or of a whole building and to be able to manipulate items around the office. The flexibility of the database is essential, with the ability to manage reports as well as make structured complex enquiries (Lawrence, A. 1988). Very much similar to what we have been doing to enable cost modelling to take place at early design stage, but in this situation it is post-occupancy evaluation and building maintenance that it is aimed at.
For example, facilities management applications will enable text data to be built up around items in the drawing. Typically this data will include code numbers, costs and date of purchase, description, dimensions, maintenance schedule, special problems and so on. Location information should be automatically handled by the CAD system. By using this information a facilities manager is able to 'see' what is going on in the building. It is possible to pick out items from the 2D or 3D model and check on their price,

their age, their maintenance schedule, their history, and so forth. It is also possible to take an item from one room to another and all the textual information relating to that item will go with it. Facilities management at present is still in its infancy, but has potentially lot of promise. In the long term, as networking technology advances, it might become the main building block of the future building control system.

Another application that emerges from the two way link between CAD systems and database management, is what is now called Geographical Information Systems (GIS). This allows the contents of databases to be displayed against a map. A reason to make such applications so powerful is that so much everyday data has a spatial element, that is, it can be related to a specific location on the Earth. This includes, for example, data on the distribution of natural resources, the location of buildings, the incidence of pollutants, the infrastructure of utility and transport services and even the population's health, wealth and voting habits (Sweet, P. 1989).
The techniques used here are not radically new, but the refinements in relational database technology and the increasing availability of digital data (eg. satellite digital pictures) have all combined to take GIS out of the research institutions. GIS offer important savings in the labour required to keep the maps up to date and reduce chances of error (Bennion, F. 1989). Although, this type of application do not appear to have direct relevance to architectural design. It has the substance to support architectural design decision, by allowing rapid access to urban and regional planning information.

8.2 FROM DESIGN TO MANAGEMENT.

The last decade has seen a rapid change in the roles of some architects, quantity surveyors and builders. As George, B.W. (1988) put's it :

> "Any architect, if he now wishes, may build; at the same time builders are offering design services and quantity surveyors are some times seen as a threat to architects and builders by offering project management services" (p. 4).

This new situation, however exaggerated, did not occur instantly, and it is believed that several reasons have helped bringing it about. For example, the industry/profession fragmentation due to the fluctuating workload caused by the receding economic situation, is believed to have increased competitive pressures on adjacent professions (McGee, J. 1988). However, it is not the intention to investigate the causes of this new condition, but to help manage it adequately.

The role of information technology in accomplishing this professions overlapping can be consequential. In fact, the subject of this thesis is an example of the new technology potential to bring together the efforts of the architect and the quantity surveyor at the earliest stages of conceptual design. Yet, this work would be of limited impact on the whole building process, if the construction is conducted without a formalised project management system. It is obvious that there is no point is conducting a design in a fraction of its ordinary time, when the construction process is lasting as long as it usually does. Hence the importance for information technology to extend its support to project and construction management.

8.2.1 Managing the whole design and construction environment.

Amid the existing applications for project and construction management, two general categories of computer aids can be distinguished. First, systems for strategic planning of a construction prior to formal design, and second, systems for the monitoring and controlling of the construction operations. Both benefit from the development of formalised computer based project management techniques, which mainly rely on the 'Programme Evaluation and Review Technique' (PERT) and the Critical Path Method (CPM). This methods had their infancy in the early 1950's and considerably progressed along side the computer technology.

Advocating the application of CAD to construction management, Atkin, B. (1985) argues :

> "In practice, tacking account of the ... construction factors would be difficult if the processes were of an entirely manual nature. However, the use of powerful computer-aided design systems, incorporating database management system, may provide part of the answer"(p. 146).

In fact Atkin (1986), at the Department of Construction Management at the University of Reading has conducted a project into the use of computer aided design systems as tools of management. Among his recommendations, he suggests the use of inferencing mechanisms in conjunction to CAD databases to analyse the construction process based on a computer building model.

As fas as construction management prior to detailed design is concerned, a concrete solution has been put forward, based on the latest information technology techniques. This is a construction project planning Expert

System (ES), resulting from the unique construction industry involvement in the Alvey program (see section 3.1.1). Elsie (it is the name of the package) supports *The Strategic Planning of Construction Projects* , and consists of four modules representing different types of ES problems (Brandon, P.S. 1988) :-

(1) Financial budget - to determine the cost of the structure proposed for a given standard of building. In this module the user has to answer approximately twenty five questions on the designed building (the system considers only office buildings), from these the software gathers an indication of quality, form and size which allows it to 'derive' a specification for the building.

(2) Procurement - this module determines the suitable type of contractual arrangement between the client, designers and the building team. Its suggestion of procurement method is defined by five major strategies in common use in the UK construction industry (i.e.: Conventional, Two-stage tendering, Management contracting, Construction management, and Design and build).

(3) Time - forecasting time is a key feature in building construction development strategy. Consequently, this module makes an assessment of the time required to construct the building, and tries to optimise the overall development time.

(4) Development appraisal - finally, this module gathers up information used in the other modules together with inputs by the user to discover the viability and profitability of the scheme.

In a series of articles reviewing Elsie, Ashworth (1988) reports that :

> "... the approach adopted *(i.e.. expert systems)* has led to a simpler linkage of the four modules than could

> have been achieved with conventional programs."(p 51).

Justifying the initial 'avant gard' picture given to the RICS (Royal Institution of Chartered Surveyors) involvement with the Alvey program, marking the move of the profession into so-called fifth generation computers (see Building, 15 November 1985, p 97). It is believed, for example, that using ES makes it easier to produce more transparent and provide better explanation to the system decision making procedure in the case of the Financial budget module.

The performance prediction of an architectural product designed on a computer system, will use algorithmic and knowledge-based computer programs to support rational design decision making (Schmitt, G. 1987). The system described above is based on knowlege-based programming techniques, whereas the cost modelling system presented in this thesis in based on algorithmic programming techniques. The combination of the two techniques, as well as the interaction between graphical and non-graphical design representations, is a forcible development in information technology application to architecture and construction. Some issues of such development are looked at in the next section.

8.2.2 Main issues in current C.A.A.D. development.

Throughout the development of computer aided architectural design systems, various socio-professional and technological issues were raised. Some of them were perceived as 'problems', and were adequately addressed, whereas others still remain unclear and require further investigation. These could be grouped in two main categories, social issues and technological issues.

- **Social issues** :

(1) Changing the structure of professions.

What may be the most significant implication of CAD introduction in the design office, is the impact it potentially has on re-shaping professional relationships and structures. We have seen earlier, in Chapter 2, that the complexity of modern design problems have separated designing from making, and that professionalisation has reinforced the remoteness of architects from builders. CAAD, and most information technology tools for that matter, offer the possibility to dismantle the existing strong professional barriers that have been discouraging communication between the various members of the design and construction team (McGee, J. 1988). There are, in CAAD, opportunities for designers to reconsider the way information travels between the members of the design team.

(2) Computer Aided Participatory Design.

The other major social implication that computers, as a new technology, have on our lives is their effectiveness in improving public involvement in design. Computers as education and communication tools have the ability to supplement and support participation such as workshops and public meetings. This practice is called Computer Aided Participatory Design (C.A.P.D.), it aims to improve the traditional participatory design methods in quality and frequency. In the first place, the emphasis of CAPD is on preparing both the public and professionals to participate more fully in the design by having equal access to information. With later, the possibility to provide the same public with three-dimensional images of the consequences of environmental decisions (Quayle, M. 1987).

(3) The myth of Information Technology.

What is considered as the myth in this information technology era, is the notion that we are entering a radically different stage of human history. Masuda, Y. (1985), in his vision of 'Computopia', argues that the post-industrial society will be an information making and consuming society, where the production of information values and not material values will be the driving force. This might be true, but there is no sign that such change will have any effect on the economic, political and cultural state of order (or disorder) the world is living in.

Hamelink, C.J. (1986), agues this point by writing :

> "It (*information revolution*) expects that the application of computer-steered technologies will effectively terminate a social structure which is characterized by an endless struggle between winners and losers, between rulers and ruled."(pp 7-8).

In fact, information technology retains the tendency where an innovation or invention which ought to increase the independence of the many, tends to become an instrument of even closer control by the few. What generally tends to happen is that a technical discovery leads to cheaper cost. The saving is used to make greater profits, which are then instantly reinvested in the battle to seize an even greater share of the market (Ascherson, N. 1987). There is relatively little we architects can do about this, but it is important to be aware of the potentials of any radical change in social order during this period of 'paradigm' shift.

- Technical issues :

(1) Computers parallel processing capabilities.

One of the crucial points in computer development is the hardware performance. We have seen earlier, in section 3.1.1, that computer architecture was at the breach of a radically new implementation, using parallel processing machines. This will almost certainly be of significant influence on future developments in CAAD. It could be either by allowing the production and processing of high resolution images at very high speed, or also by enabling the exploration of radically new implementations, closer to human parallel information processing, using the parallelism of these machines.

(2) Computer user interface.

Most CAD system developers are now taking advantage of the latest developments in personal computing and human interface research by including icon driven, direct manipulation menus on Work Stations. This user interface has promoted a direct manipulation approach to the process of drawing on CAD systems, which can be as natural and as comfortable as manual drafting.

(3) CAD data exchange in the building industry.

In a forecast made on the UK construction industry by the National Economic Development Office (NEDO) CAD working party (Building, 14 April 1989, pp 88-89). It is expected that by the middle of the next decade, most of the large construction firms will use CAD on the majority of their projects, and it is expected that their will be need to exchange data (graphical data essentially) on a large proportion of these projects.

Exchanging data between different CAD systems can be in effect a considerable burden to CAD users. The absence of a standard file exchange format, is certainly causing a considerable confusion among CAD users and software developers. Some attempts to arrive at a 'universally' accepted graphics file formats were made (see section 3.3.2), but with relatively little success. It is hoped that in the future a solution is found to this problem, as it bears a large responsibility on CAD's acceptance in architectural practice and education.

Chapter 9 : CONCLUSION.

The initial research proposal of this project was aimed at studying the 'combination of the analytical and generative ways of using computer-aided design, and examining the influence of such implements on the design process'. There was no problematic as such, but rather a general interest in the way two main approaches to the use of computers in architectural design could co-exist, and a concern for the impact this might have on the traditional practice of design.

From the first readings of the Computer Aided Architectural Design (CAAD) literature (Mitchell, W.J. (1977), Cross, N. (1977), Raynolds, R.A. (1980) and Paterson, J. (1980)), it appeared that the undertaking of such a task required some background knowledge of design theory and design methodology, as well as a fair understanding of the concepts behind the computer technology.

During the course of acquiring such knowledge, it became clear that there were many potentials of CAAD that were yet to be exploited. It was observed that, at a time when the increasing availability of computer aided design tools made building model visualisation for aesthetic examination accessible to most designers, the possibilities of computer aided building evaluation were still very limited.

The research environment at the GABLE Research Unit, and particularly the GABLE 4D-SERIES CAAD system governed the succeeding developments of this research. After a period of time spent at getting familiar with the system, along with in depth investigation of the CAAD subject matters. It became clear that the emphasis was going to be put on computer analytical design as against generative design. Among the building performance characteristics looked at, cost evaluation and cost modelling were used as examples. The GABLE CAAD system provided the support for the integration experiment of building modelling and cost modelling. Software specification and development necessary to implement a new facility was conducted, and then integrated into the existing CAAD system.

This research project has capitalised on the capabilities of the GABLE CAAD system to develop a Cost Modelling application. The system's data structure and its ability to interpret building models (i.e.: 'infer' the implicit spatial arrangements), were seen as GABLE strengths and were considerably drawn on.

This thesis demonstrates the potential of integrating CAD and cost modelling by producing a working software solution using a descriptive model approach to cost modelling. With this technique, it has been possible to construct a three dimensional design model, and by relying on existing cost data databases, relating to the type of projects, cost out alternative designs before deciding on the final solution. The production of a building elemental model can include all described features of the geometrical model (i.e.: located in any part of the building), or can be limited to certain selected areas of the model. This offers the possibility of involving only a selected number of spaces on a floor level, and/or a selected number of floors in a building for cost modelling. This approach makes 'cost location' a very realistic option during design costing. Thus the architect will be able for example to cost the circulation areas independently from office spaces, or compare the cost of two spaces located on different floors levels of a building, or to examine the cost implications of designing the building in quite different forms, a facility called 'space based building cost modelling'. The argument in favour of such approach to design modelling and evaluation in a CAAD environment could be taken further and extended to any other type of building evaluation.

What can be considered as an original contribution to the field of computer aided building modelling and evaluation is the interaction between **elemental** and **spatial** description of a design solution. With this approach, construction elements like walls, roofs, windows and doors are modeled independently and subsequently associated with the spaces to which they belong. This has a significant impact on cost modelling and more generally building evaluation.

However, it is conceded that there are two major deficiencies in this approach. Firstly, the accuracy of a descriptive model technique is not as high as may be described, although there seems to be no real answer to this at early design stages. Secondly, this system can only be used once a sensible three dimensional building model exists in the CAAD system. With regard to this last point, work is currently going at Sheffield University which may give rise to automatically or semi-automatically generated building models. Here, a brief is used together with expert knowledge of previous projects to hypothesis a building model. Such a model may be quite worthless as a piece of architecture but may suffice to feed cost modelling and other evaluations such as heat load calculations at briefing design.

Looking beyond the realm of this thesis, Chapter 8 has briefly examined the prospects and difficulties that CAD might encounter in its future developments. The Hypermedia concept was introduced and it possible

application in CAAD described. It is strongly felt that a 'multimedia' presentation facility assisting the designer to visualise and communicate his design solution, is a commodity to expect available in the not too far future. At the time this manuscript is written, new systems with design assistance and generation capabilities are under development at leading CAAD research centres throughout the world (Pohl, J. 1989). In those, the development of expert systems applications holds a privileged research position. As an emerging software development technology, Expert Systems and Knowledge Based Systems promise to encapsulate the knowledge of building construction experts (or any other expert for that matter), and make it available to those that are less experienced.
It is hoped that this two research vectors (i.e.: Hypermedia technology and Knowledge engineering technology) come rapidly to fruition and more importantly get combined to provide designers with what could be considered as the next generation design tools. Unfortunately, the novelty of these techniques and the lack of detailed information about current applications makes it difficult to forecast their impact on architectural practice and education.

In conclusion, it can be said that design is certainly evolving from an intuitive, unstructured activity, to present attempts to explicitly define and compute certain aspects of design knowledge. This thesis has put the emphasis on design evaluation, and made a modest contribution in bringing closer the computer and the design in performing design evaluation. It seems that it is inevitable that further externalisation of the design process will occur, and that the use of computers in design will expand. A prospect of which the out come is unknown, but will certainly depend on our ability and desire to externelise design, as well as our ability and desire to computerise design.

TABLE OF FIGURES

TABLE OF FIGURES

BIBLIOGRAPHY

BIBLIOGRAPHY:

- Akin, O. (1986) *Psychology of Architectural design,* Pion Ltd, London.

- Alexander, C. (1964) *Notes on the synthesis of form,* Harvard University Press, Cambridge.

- Archer, B.L. (1965) "Systematic methods for designers", Design Council, London.

- Aris, R. (1978) *Mathematical modelling techniques,* Pitman, London.

- Armour, G.C. and Buffa, E.S. (1963) "A heuristic algorithm and simulation approach to relative location of facilities", Management Science 9(2): Jan, pp 294-309.

- Ascherson, N. (1987) "Inventions never quite set us free", The Observer, 13 December, p 7.

- Ashworth, G.N. & Wilson, R.A. (1988) "The integration of computer aided design and quantity surveying systems", Paper presented at the*Quantity Surveyors and Computers* Conference, The Barbican Centre, London, 21st April.

- Asimow, M. (1962) *Introduction to design,* Prentice-Hall, Englewood Cliffs, N. J.

- Atkin, B.L. (1985) "Computer graphics in construction management", in *Information systems in construction management,* Barton, P. (Ed), Mitchell's professional Library.

- Atkin, B.L. (1986) "Time for CAD", Chartered Quantity Surveyor, Vol 8, No 2, p 13.

- Atkin, B.L. (1987) *CAD Techniques : Opportunities for chartered quantity surveyors,* Published by the Royal Intitution of Chartered Surveyors.

- Atkinson, R.C., Atkinson, E.R. and Hilgard, E.R. (1953), *Introduction to psychology,* Harcourt Brace Jovanovich Publishers, New York.

- Baglivo, J.A. and Graver, J.E. (1983) *Incidence and symmetry in design and architecture,* Cambridge University Press, Combridge.

- Barnett, J. (1965) " Computer-Aided Design and automated working drawings", Architectural Record, October, pp 85-97.

- Barnett, J. (1967) "Computerised cost estimating", Architectural Recors, January, pp 163-166.

- Beeston, D. (1987) "A future for cost modelling", in *Building cost modelling and computers,* by Brandon, P. (Ed), Spon.

- Belhadj, T.A. and Lawson, B.R. (1989) "Computer Aided Building Evaluation - A cost modelling experiment", paper submited to the "Science et Technology au Service de l'Architecture" Conference, to be held in Paris in Dec 1989.

- Bennion, F. (1989) "Geographic information systems", The Computer Bulletin, Vol 1, Part 5, June, pp 13-16.

- Berlyne, D.E. (1965) *Structure and direction of thinking ,* John Wiley, New York

- Berztiss, A.T. (1975) *DATA STRUCTURES theory and practice ,* Academic Press. London.

- Best, D. (1985) *Feeling and reason in the arts ,* George Allen & Unwin, London.

- Bijl, A. (1987) " An approach to design theory" in *Design theory for CAD* , Proceeding of IFIP conference (Tokyo), Yoshikawa, H & Warman, E. A. (Eds), North-Holland.

- Bishop, P. (1986) *FIFTH GENERATION COMPUTERS, concepts, implications and uses,* Ellis Horwood Limited.

- Bowyer, A. and Woodwark, J. (1983) *A programmer's geometry,* Butterworths, London.

- Brandon, P.S. and Moore, R.G. (1983) *Microcomputers in building appraisal,* GRANADA, London.

- Brandon, P.S. (1988) "An expert system fot the strategic planning of construction projects", Paper presented at the*Quantity Surveyors and Computers* Conference, The Barbican Centre, London, 21st April.

- Britch, A.L. (1963) "Quantity surveying by electronic computers", The Architects Journal, 27 March, pp 669-674.

- Broadbent, G (1973) *Design in Architecture,* John Wiley, New York

- Broadbent, G (1979) "The development of design methods - a review ",Design Method and Theories,Vol 13, No 1, pp41-45.

- Campion, D. (1968) *Computers in architectural design,* Elservier Publishing Company, London.

- Codd, E.F. (1970) "A relational model of data for large shared data banks", Communications of the ACM, Vol 13, No 6, June 70, pp 377-387.

- Collins, P. (1971) *Architectural Judgement* , Faber & Faber, London.

- Conklin, J. (1987) "Hypertext: An introduction and survey", IEEE Computer Magazine, September, pp 17-41.

- Coons, S.A. (1963) "An outline of the requirements for a computer aided design system", in Proceeding of joint computer conference, ACM, New York.

- Cowan,H.J.,Gero,J.S.,Ding,G.D. & Muncey,R.W. (1968) *Models in Architecture,* Elsevier Publishing Company, London.

- Crosley, M. (1988) "Linking, graphics and numbers", Architectural and Engineering Systems, January, pp 29-34.

- Cross, N. (1977) *The automated architect,* Pion Ltd., London.

- Cross, N. (1985) "Styles of learning, designing and computing", Design Studies, Vol 6, No 3, (july).

- Curran, J. (1987) " The QS and the CAD system", Charted Quantity Surveyor, Vol 10, No2, pp 11.

- Daley, J. (1969) "A philosophical critique of behaviourism in architectural design", in Broadbent, G. and Ward, A. *Design methods in architecture,* Architectural Association, Paper No 4, London.

- Darke, J. (1979) "The primary generator and the design process", Design Studies, Vol 1, No 1, pp 36-44.

- Date, C.J. (1975) *An introduction to data base systems,* Addison-Wesley Publishing Company, Reading, Massachusetts.

- Date, C.J. (1983) *Database : A primer,* Addison-Wesley Publishing Company, Reading, Massachusetts.

- Delisle, N. and Schwartz, M. (1986) "Neptune: A Hypertext System for CAD applications", Proc. of AMC SIGMOD International Conference on Management of Data. Washington, D.C., May 28-30, pp 132-143.

- Denker, J.S. (Ed) (1986) "Neural networks for computing", AIP Conference proceeding 151, American Institut of Physics.

- Dent, C. (1964) *Quantity surveying by computer,* Oxford University Press, London.

- Design Methods Group (DMG) Reference sheet No1, June 1978.

- Deo, N. (1974) *Graph theory with applications to engineering and computer science,* Englewood Cliffs (N.J.), London, Prentice Hall.

- Donald, I. (1988) " Place evaluation", in "*Environmental policy assessment and communication*", Canter,D.,Krampen,M. & Stea,D. (Eds), Avebury.

- Duncker, K (1945) "On problem solving" Psychological Monographs, Edited by Dashiell, J.F.,Vol 58 No5, The American Psychology Association.

- Eastman, C.M. (1975) "The use of computers instead of drawings in building design", American Institute of Architects Journal, March, pp 46-50.

- Eastman, C.M. (1978) "The representation of design problems and maintenance of their structure", in *Artificial intelligence and pattern recognition in computer aided design* , Latombe, J. (Ed), North-Holland Publishing Company.

- Eastman, C.M. (1987) "Fundamental problems in the development of computer-based architectural design models", in *Computability of desing,* Kalay, Y.E. (Ed), Wiley Interscience, John Wiley & Sons, New York.

- Evans, C. (1981) *THE MAKING OF THE MICRO :A history of the computer* , Victor Gollancz Ltd, London.

- Feigenbaum, E.A. & McCorduck, P.(1984) *THE FIFTH GENERATION : artificial intelligence and Japan's computer challenge to the world,* Pan Books, London.

- Ferry, D.J. and Brandon, P.S. (1984) *Cost planning of buildings,* GRANADA, London.

- Foley, J.D. and Van Dam, A. (1984) *Fundamentals of interactive computer graphics,* Addison Wesley.

- Foque, R. (1982) "Beyond design methods-Arguments for a practical design theory." in Changing Design Edited by Evans, E.,Powell, J.A., and Talbot, R.J., John Wiley & Sons, Chichester.

- Freeman, (1987) 'The case for technological determinism". in *Information technology : social issues* „ Finnegan, R., Salaman, G. and Thompson, K. (Eds), The Open University, Hodder and Stoughton.

- Garner, R.W. (1962) *Uncertainty and structure as psychological concepts,* John Wiley, New York.

- George, B.W. (1988) "Project Mangement and Microcomputers", Paper presented at the*Quantity Surveyors and Computers* Conference, The Barbican Centre, London, 21st April.

- Gerardin, L. (1968) *Bionics,* World University Library, London.

- Gero, J.S. (1983) "Computer-Aided Architectural Design - past, present and future", Architectural Science Review, Vol 26, No 1, pp 2-5.

- Gill, M. & Atkin, B. (1985) "Survey of CAD systems used by the UK construction industry, Building Magazin, 28 June to 26 July.

- Gilhooly, K.J. (1982) *Thinking:Directed, Undirected and Creative,* Academic Press, London.

- Goodey, K.J. & Mattew, K. (1971) "Architects and Information", Research paper No 1, Institut of Advanced Architectural Studies, University of York.

- Gregory, S. (1982) "Evaluation", Design Studies, Vol 3, No 3, july, pp 147-152.

- Groos, M.D., Ervin, S.M., Anderson, J.A. & Fleisher, A. (1987) "Designing with constraints", in *Computability of desing,* Kalay, Y.E. (Ed), Wiley Interscience, John Wiley & Sons, New York.

- Groos, M.D., Ervin, S.M., Anderson, J.A. & Fleisher, A. (1988) "Constraints : Knowledge representation in design", Design Studies, Vol 9,No 3, July , pp 133-143.

- Guttman, A. and Stonebraker, M. (1982), "Using a RDMS for CAD data", Database Engineering, No 6, pp 21-26.

- Halprin, L. (1969) *RSVO cycles: creative processes in the human environment,* Braziller, New York.

- Hamilton. I, & Winterkorn, E. (1985) *CAD sytems evaluation for construction,* Construction Industry Computing Association, Cambridge.

- Handler, A.B. (1970) *System approach to architecture,* American Elsevier Publishing company, Inc.

- Hassan Fathy, (1973) *Architecture for the poor,* University of Chicago Press, Chicago.

- Heath, T. (1984) *Method in Architecture,* John Wiley, New York.

- Hillier, B. and Leaman, A. (1972) "Knowledge and design", in Mitchell, W.J. (Ed), Environmental Design: Research and Practice, Proceedings of the EDRA/ar8 Conference, U.C.L.A., pp 29.3.1-29.3.14.

- Hillier, B., Musgrove, J. & O'Sullivan, P. (1972), "Knowledge and design", in Mitchell, W.J. (Ed), EDRA 3/8 Conference Proceedings, pp 29.3.1-29.3.14.

- Hodges, A. (1983) *The Enigma of intelligence,* Burnett Books, London.

- Hodde, A. (1988) "Rencontre avec des architectes", Techniques & Architecture, No 380, pp 112-114.

- Hopgood, F., Duce, D., Gallop, J. & Sutcliffe, D.C. (1983) *Introduction to the Graphical Kernel System (GKS),* Academic Press.

- Jones, J.C. (1970) *Design Methods: seeds of human futures,* John Wiley, New York

- Kalay, Y.E. (1983) "A relational database for non-manipulative representation of solid objects", Computer Aided Design, Vol 15, No 5, pp 271-276.

- Kalay, Y.E. (1985) "Redefining the role of computers in architecture : fromm drafting/modelling tools to knowledge-base design assistant", Computer Aided Design, Vol 17, No 7, pp319-328.

- Klinger, A., Fu, K.S. and Kunii, T.L. (Eds) (1977) *Data structures, computer graphics, ans pattern recognition,* Academic Press, Inc.

- Koestler, A. (1964) *The Act of Creation,* Hutchinson, London.

- Kohler, W. (1947) *Gestalt psychology : An introduction to new concepts in modern psychology,* Liveright Publishing Corporation, New York.

- Koriba, M. (1983) "Database systems: their applications to CAD software design", Computer Aided Design, Vol 15, No 5, pp 277-287.

- Kroll, L. (1986) The Architrecture of Complexity, Translated by Jones, P.B., B.T. Batsford Ltd, London.

- Kronsjo, L. (1979) *Algorithms, their complexity and efficiency,* Wiley, Chichester

- Lansdown, J. (1969) "Computer-aided building design - the next step", RIBA Journal, April, pp 138-140.

- Lansdown, J. (1982) *EXPERT SYSTEMS : Their impact on the construction industry,* RIBA conference Fund.

- Lansdown, J. (1986) "Do CAD systems fulfill designer's needs ?", Computer Aided Design, Vol 18, No 6, July/August, pp 299-300.

- Lansdown, J. (1987) "A theory of computer aided design, A possible approach", in Graphics for designers, Earnshaw, R.A. (Ed), BCS Conference.

- Lansdown, J. (1988) "Prototype modification : creative computer aid", The Architects Journal, 21 December, pp 52-57.

- Lansdown, J. & Roast, C. (1987) "The possibilities and problems of knowledge-based systems for design", Environmental & Planning B, Vol 14, pp 255-266.

- Laszlo, E (1972) *The system view of the world,* Basil Blackwell, Oxford.

- Latombe, J-C. (1977) "Artificial intelligence in computer-sided design: the 'TOPIC' system", in *CAD systems* by Allan, J.J. (Ed), North-Holland Publishing Company, Amsterdam.

- Lawrence, A. (1988) "A model office", CADCAM International Magazine, August, pp 13-16.

- Lawson, B.R. (1970) "Open and closed ended problem solving in architectural design", in Honikman, B. (Ed), AP70: Proceeding of the architectural psychology conference at Kingston Polytechnic, pp 87-90.

- Lawson, B.R. (1971) "Problem structure displays in computer aided architectural design", Ergonomics paper, Applied psychology department, University of Aston, Birmingham.

- Lawson, B.R. (1972) "Problem solving in architectural design", Ph.D. thesis, University of Aston, Birmingham, UK.

- Lawson, B.R. (1979) "Cognitive strategies in architectural design", Ergonomics, Vol 22, No1, pp 59-68.

- Lawson, B. R. (1980) *How designers think,* Architectural Press, London.

- Lawson, B.R. (1981) "The current and future roles of the computer in the building design office", in Proceeding of the CIB W-65, Third Symposium of Organisation and Management of Construction, Dublin, pp A. 2.91-A.2.103.

- Lawson, B.R. (1985) *Computer aided design for architects,* Unpublished manuscript, University of Sheffield, Dept. of Architecture.

- Lawson, B.R. and Riley, J.P. (1982) "ISAAC a technique for the automation of spaces from drawn building floor plans", Proceeding CAD 82 conference, Butterworth, pp 663-666.

- Lera, S. (1981) "Empirical and theoritical studies of design judgement : a review", Design Studies, Vol 2, No 1, pp 19-26.

- Lera, S. (1982) "At the point of decision", Building, 28 May, pp 47-48.

- Lera, S., Cooper, I., & Powell, J.A. (1984) "Information and designers", Design Studies, Vol 5, No 2, pp 113-120.

- Levin, P.H. (1964) "The use of graphs to decide the optimum layout of buildings", The Architects Journal, 7 Oct, pp 809-815.

- Lyons, E. (1988) "Solids and surfaces" ,Computer Graphics World, June 1988, pp 36-39.

- McGee, J. (1988) "The changing structure of the surveying profession", Public lecture in estate management, 2 February, Tempelton College, Oxford.

- Mackinder, M. & Marvin, H. (1982) "Design decision-making in architectural practice", Research paper No 19, Institute of Advanced Architectural Studies, University of York.

- Mainstone, R. (1977) "Brunelleschi's dome" Architectural Review No 967(Sept), pp 157-166.

- March, L. (1974) "Quantifying the environment" in *Models, Evaluations & Information Systems for Planners,* Perraton,J. and Baxter, R. (Eds), LUBFS Conference Proceeding No 1, MTP Construction.

- March, L. (1976) *The architecture of form,* Cambridge University Press, Cambridge, UK.

- March, L. and Steadman, P. (1971)*The geometry of environment* , Methuen & Co Ltd.

- Marciszewski, W. (Ed.) (1981) *Dictionary of Logic,* Martinus Nijhoff Publisher, The Hague.

- Markus, T.A. (1967) "The role of building performance measurement and appraisal in desing", The Architects' Journal, 20, Dec 67, pp 1567-1573.

- Markus, T.A. (1970) "Optimization by evaluation in the appraisal of builsings", Building, 21 August and 18 September.

- Markus, T.A. (1971) "A doughnut model of the environment and its design", in *Design participation,* Cross, N. (Ed), Academy Press.

- Mathur, K. (1982) "Resources and performance optimisation", Chartered Quantity Surveyor, June.

- Maver, T.W. (1970) "A theory of architectural design in which the role of the computer is identified", Building Science,Vol 4, pp 199-207.

- Mayer, R.J. (1987) IGES, An answer to the problems of CAD database exchange, Byte Magazine, Vol12, No 7, pp 209-214.

- Meier, A. (1986) 'Applying relational database techniques to solid modelling", Computer Aided Design, Vl 18, No 6, July/August, pp 319-326.

- Michie, D. (1980) "Turing and the origins of the computer", New Scientist, 21/2/80, pp 580-583.

- Mitchell, W.J. (1977) *Computer-Aided Architectural Design,* Van Nostrand Reinhold Company, New York.

- Mitchell, W.J. (1986) "Solid modelling and volumetric composition in architecture", Design Computing, Vol 1, No 2, pp 123-136.

- Moto-oka *et al* (1982) *Fifth generation computer systems* , Proceeding of the International Conference on Fifth Generation Computer Systems (JIPDEC), Tokyo, Japan, North-Holland Amsterdam.

- Murray, T.J. (1985) *Computer based information systems,* Richard D Irwin, Inc., Homewood, Illinois.

- Myers, G.J. (1978) *Advances in computer architecture,* John Wiley & Sons, New York.

- Nasar, J.L. (1980) "Third generation design methods" Design Methods and Theories,Vol 14,No 2,pp 90-92.

- Neelamkavil, F. (1987) *Computer simulation and modelling,* John Wiley & Sons.

- Nelson, T. H. "Getting It Out of Our System", in *Infromation Retrieval: A Critical Review,* by Schechter, G. (Ed), Thompson Books, Washington, D.C.

- Negroponte, N. (1970) *The architecture machine,* The MIT Press, Cambridge, Massachusetts.

- Newell, A., Shaw and Simon, H.A. (1958) "Elements of a theory of human problem solving." Psychological Review, Vol 65, No 3.

- Newell, A. and Simon, H.A. (1972) *Human Problem Solving,* Prentice-Hall, Englewood Cliffs, N. J.

- Newman, M.N. and Sproull, R.F. (1981) *Principles of interactive computer graphics,* International Student Edition.

- Ornstein, R. (1987) *Multimind : A new way of looking at human behaviour,* Macmillan, London.

- O'Shaughnessy, J. (1972) *Inquiry and Decision,* George Allen & Unwin Ltd, London.

- Patchell, B.R.T. (1987) "The implementation of a cost modelling theory", in *Building cost modelling and computers,* by Brandon, P.S. (Ed), E. & F.N. Spon, London.

- Paterson, J. (1980) *Architecture and the microprocessor,* John Wiley & Sons, New York.

- Perkins, P.N. (1981) *The mind's best work,* Harvard University Press, Cambridge, Massachusetts.

- Pfefferkorn, C.E.(1975) "The Design Problem Solver:A system for designing equipment or furniture layouts" in *Spatial synthesis in computer aided design* , by Easteman, C.M. (Ed), Applied Science Publisher, London.

- Pfortmiller, L. (1987) "Data structures in CAD software", Byte Magazin, Vol 12, No 7, pp 177-184.

- Pohl, J. (1989) "Artificial Intelligence in the CAD system environment", Architectural & Engineering Systems, June, pp 48-50.

- Prager, F.D. and Scaglia, G. (1970) *Brunelleschi : Studies of his technology and inventions,* The M.I.T. Press, Cambridge.

- Quayle, M. (1987) "Computer-Aided Participation Design : SITSEE", The Journal of Architectural and Planning Research, 4:4:, pp 335-342.

- Reid, T.R. (1985) *Microchip The story of a revolution and the men who made it,* Collins, London.

- Reynolds, A. (1980) *Computer methods for architects* , Butterworths, London.

- Richens, P. (1983) "Dis-integrated models for building design", in Proceedings of PArC 83, pp 121-129, Oneline Conference Ltd.

- Richens, P. (1984) "What a system of the future might look like...", pp 41-47, Proceeding of the W78 Colloquium orgranised by the International Council for Building Research Studies (CIB), 5th - 7th June, Watford, UK

- Rittel, H.W.J. (1970) "Comments : Theories of cell configurations", in *Emerging methods in environmental design and planning* , Moore, T. (Ed), The MIT Press, Cambridge, Massachusetts.

- Rittel, H.W.J. (1972) "Son of Rittelthink", Interview with Portzen, J-P. and Grant, D., in DMG Occasional paper No 1, pp 5-10.

- Roth, J. & Hashimshony, R. (1988) " Algorithms in graph theory and their use for solving problems in architectural design", Computer Aided Design, Vol 20, No 7, pp 373-381.

- Rowe, P.G. (1987) *Design Thinking*, The M.I.T. Press, Cambridge.

- Schilling, G.T. (1988) "Structuring CADD layering", Architecture, American Institut of Architects, January, pp 102-132.

- Schmitt, G. (1987) "Expert systems in design abstraction and evaluation", in *Computability of design*, Kalay, Y.E. (Ed), John Wiley & Sons.

- Seeley, I.H. (1972) *Building Economics*, Macmillan, London.

- Seeley, I.H. (1984) *Quantity surveying practice*, Macmillan, London.

- Shannon, C.E. (1938) "A symbolic analysis of relay and switching circuits", AIEE Transactions, Vol 57, pp 722.

- Shannon, C.E. and Weaver, W. (1949) *The mathematical theory of communication*, Univ. of Illinois Press, Urbana.

- Shaviv, E. (1987) "Generative and evaluative CAAD tools for spatial allocation problems", in *Computability of design*, Kalay, Y.E. (Ed), John Wiley & Sons.

- Simon, H.A. (1965) *The shape of automation for man and management*, Harper & Row, Publishers, New York.

- Simon, H.A (1981) *The science of the artificial*, The MIT Press, Cambridge, Massachusetts.

- Simon, H.A. (1983) " Why should machines learn ?", in *Machine learning*, by Michalski, R., Corbnell, J. Mitchell, T. (Eds), Tiago publishing company, Palo Alto, California.

- Smith, G. (1980) "Cost planning the design process", Chartered Quantity Surveyor, August.

- Stafford, H.M. (1957) "Standardized quantities and mechanized billing", The Architects Jounal, 18 April, pp 576-580.

- Steadman, P. (1976) "Graph-theoritic representation of architectural arrangement", in *The architecture of form*, by March, L. (Ed), Camridge University Press, Cambridge.

- Sudbo, B. (1988) "Reports Zurich '87", The European Association for Architectural Education 10th International Forum on Architectural Education and the Information Explosion, Zurich 5-7 Nov 97, News Sheet 21, March 1988, pp 1-3.

- Sussock, H. (1982) "GOAL : General Outline Appraisal Layouts", Occasional paper 62, ABACUS, University of Strathclyde, Glasgow.

- Sutherland, I.E. (1963) "Sketchpad: A man-machine graphical communication system", Proceeding of the 1963 Spring Joint Computer Conference, Baltimor, Maryland : Spartan Books.

- Sweet, P. (1989) "Mapping out a future", Database/Graphics, Computing Magazin, British Computer Society, 23 March, pp 26-27.

- Swinson, P.S.G. (1982) "Logic programming:a computing tool for the architect of the future", Computer Aided Design, Vol 14, No2, pp 98-104.

- Teague, L.C. (1970) "Network models of configurations of rectangular parallelepiped", in *Emerging methods in environmental design and planning* , Moore, G.(Ed), MIT Press, Cambridge, Massachusetts.

- Tovey, M. (1986) "Thinking styles and modelling systems",' Design Studies, Vol 7, No 1, pp 20-30.

- Turing, A.M. (1947) "The automatic computing engine", Lecture delivered at the London Mathematical Society, 20 Feb 1947, Typescripts in Turing Collection, Kings College Library, Cambridge.

- Viollet-le-Duc (1863) *Discources in Architecture,* Trans., B. Bucknall, Boston.

- Walker, P. (1985) " The Transputer", Byte Magazine, Vol 10, No 2, pp 95-122.

- Wallace, W.A. & Kelly, J.R. (1987) " Decision taking in building design, CIB Proceeding of the fourth international symposium on building economics (Session B), Copenhagen.

- Watson, J.B.(1913) "Psychology as the behaviourists view it." Psychilogical Review 20. pp158-177.

- Weizenbaum, J. (1984) *Computer power and human reason : From jugement to calculation,* Pelican Books.

- Weizenbaum, J. (1985) "The myths of artificial intelligence", in *The information technology revolution,* Forester, T. (Ed), pp 84-94.

- Whithead, B. and Eldars, M.Z. (1964) "An approach to the Optimum Layout of Single-Story Building", Architects' Journal, 17 june, pp 1373-1379.

- Whithead, A.N. and Russel, B. (1910-1913) *Principia mathematica* , Cambridge University Press.

- Wirth, N. (1976) *Algorithms + Data Structures = Programs,* Prentice-Hall, Inc.

- Yessios, C.I. (1987) "The computability of void architectural modelling", in *Computability of desing,* Kalay, Y.E. (Ed), Wiley Interscience, John Wiley & Sons, New York.

- Zevi, B. (1964) "History as a method of teaching architecture" in *The history, Theory and Criticism of Architecture,* Whiffen editor, The M.I.T. Press, Cambridge.