SIMULATING INFORMATION FLOW TO ASSIST BUILDING DESIGN MANAGEMENT

By

TAREK MOHAMED HASSAN, BSc., MSc.

A Doctoral Thesis submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy of the Loughborough University

April 1996

TABLE OF CONTENTS

ABSTRACT

The design of modern buildings has become an increasingly complex activity. This is because of greater demands by Clients in terms of performance, quality, economy and time. These demands coupled with the complex iterative nature of design have resulted in increasing challenges in building design and in the management of the design process.

The design process is information driven. Initial research by the writer showed that the main difficulties encountered during the management of the design process are information related. Information transfer and communication issues have been identified as key factors in the successful management of the process. It was concluded that current planning techniques are ill-suited for planning, monitoring and controlling building design because they neither accommodate the iterative nature of design nor permit the choice of alternatives. This research sought to develop better tools to aid design managers in improving the management of the process. Although all phases of the design process were examined, the main focus of this research was the Conceptual and Schematic design stages.

To investigate these stages a generic data flow model was developed using the structured analysis diagramming technique of Data Flow Diagrams. The model was based on data from preliminary case studies and was validated by interviews with construction industry professionals.

Industry feedback showed that improved management of the design process should not only include better techniques for planning and scheduling but also allow design managers to investigate the iterations between design tasks and predict the effects of different scenarios. Matrix partitioning techniques were used to identify loops of iterative design tasks in the data flow model. A Discrete Event Simulation Model was developed to predict the effects of different scenarios. This model was based on data from the Data Flow Model and the identified iterative design loops. In addition, dynamic factors input by the user such as the durations and resources of the design tasks allowed the examination of the effects of different scenarios of information related criteria. These criteria were identified from industry survey and interviews. The simulation model was rigorously tested and validated through subsequent case studies and review by industry practitioners.

The thesis concludes that the use of Data Flow Modelling in conjunction with Matrix Analysis and Discrete Event Simulation techniques provides a powerful tool for assessing the impact of change within the design process and could form the basis for managing and planning multi-disciplinary design work.

ACKNOWLEDGEMENTS

I should like to express my sincere thanks and gratitude to Dr Andrew Baldwin, my first supervisor, for his invaluable guidance, support, encouragement and advice without which this thesis would not have been completed.

My thanks are also due to Dr Simon Austin, my second supervisor, for his experienced suggestions and constructive comments throughout the research. I should also like to extend my gratitude to Professor Ronald McCaffer, the director of this research, for his continuous support and encouragement.

The author also wishes to express his thanks and gratitude to the professional staff of the following companies for the valuable information they provided and for their time in discussing my research, completing questionnaires, attending demonstrations and responding to interviews:

- Ove Arup and Partners; Birmingham offices
- Ove Arup and partners; Nottingham offices
- AMEC Design and Management
- Kyle Stewart
- John Laing
- Fisons

Particular thanks are due to Mr David Storer, design manager - Ove Arup and Partners (Birmingham offices) and Mr David Hammond, senior architect and design leader - AMEC Design and Management for the valuable information they provided and the continuous feedback they offered.

To all these people and many more, I will always remain grateful.

LIST OF CHAPTERS

Page

PART I: INTRODUCTION

CHAPTER 1: INTRODUCTION

PART II - REVIEW OF LITERATURE & CURRENT PRACTICE

CHAPTER 2:THE BUILDING DESIGN PROCESS AND ITS MANAGEMENT

CHAPTER 3:MODELLING THE DESIGN PROCESS

CHAPTER 5: A GENERIC MODEL FOR THE CONCEPTUAL AND SCHEMATIC DESIGN STAGES

CHAPTER 6:THE SIMULATION MODEL

CHAPTER 7:EVALUATION OF THE DEVELOPED TOOLS

CHAPTER 8:IMPROVING THE MANAGEMENT OF THE DESIGN PROCESS

PART IV-CONCLUSIONS AND RECOMMENDATIONS

CHAPTER 9: CONCLUSIONS AND RECOMMENDATIONS

LIST OF APPENDICES

LIST OF FIGURES

LIST OF TABLES

PART I

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND TO THE RESEARCH

In the current extremely competitive construction market, designers and contractors must respond swiftly and efficiently to Clients' requirements and provide a building within the agreed standards, and satisfying the cost and time constraints. Efficient management of the design process is imperative to ensure that the Client's requirements have been met before starting construction. Design changes, and/or interference in the construction process resulting from late construction information is costly and timely. Lack of design management results in insufficient information for completing detailed designs or instances of conflicting construction details. We all need faster, more accurate data (McGee 1992).

The construction industry has increasingly come to recognise the need for more effective information transfer between different participating organisations and internally, among the personnel of these organisations. (Gray et al 1994, Newton 1995, Austin et al 1993,1994,1995, Ndekugri et al 1988).

The importance of improved design management is now widely recognised. A report by NEDC showed that more than 50% of problems on building sites were related to poor design information (NEDC 1987). These problems were often found to be more significant than those attributed to poor workmanship and site management. With the costs in Europe of rectifying building failures running at 12-15% of total construction expenditure (Cornick 1991), the rewards for improving management of design information are very great. This has been confirmed by Glavan and Tucker (1991) who have shown how many minor design-related problems significantly affect construction performance.

In a handbook for the successful management of design, Gray et al (1994) identified ten steps to good design management. These steps reflect the importance of information transfer and communications issues for successful design management with respect to:

- timing of information transfer;
- quality of information exchanged;
- identification and understanding other participants information needs; and

- means of information transfer.

A report by NEDC (1990) entitled "Information Transfer In Building" showed that it is important to recognise that there is a hierarchy in information transfer and that there must be a point at which it can be controlled. Many problems which occur within the construction industry can be traced to either:

- lack of information transfer;
- late information transfer; or
- unresolved conflict through lack of information transfer management.

A recent study of the investigation of the decision-making processes of professional designers on engineering by Manyanga (1993) has shown:

- there is no consensus model as to how the process is conducted but there is general agreement that the process is information driven;

- the decision-making process is dependent on the information the designer has at the time that the decision is made. Lack of information leads to uncertainty forcing the designers to make tentative decisions for future confirmation or to introduce flexibility to the design which raises the project costs; and

- an information package which includes all the information required by designers can be identified. This information should be included in the client's brief, otherwise facilities should be provided that allow the designers to obtain it.

Many attempts have been made to model the design process. Early models were either descriptive or prescriptive showing the different stages of design and emphasising its iterative nature. Examples are models of French (1985), Pahl and Beitz (1988), and the VDI model produced in Germany (Cross 1991). Other models such as the RIBA (Royal Institute of Building Architects) plan of work (1973) are aimed at producing a framework of stages describing the different design and managerial tasks. However, none of these models address in detail the information transfer and communication issues which have been identified by different researchers as the key factors to the successful management of the design process. It was not until the late 1980s that structured analysis diagramming techniques, first developed for systems analysis purposes, were used to model the design process and to show the information exchange within the process. A well established technique in this respect is Data Flow Modelling which has been used by different researchers in the area of design and construction management. Examples of such research are Newton (1995), Hanby (1993), Gharib (1991), and Fisher (1990, 1992).

Improved management of the design process should not only include improved techniques for planning and scheduling but also allow design managers to investigate different scenarios within the design process (Baldwin et al 1995, Austin et al 1995). The early stages of this research confirmed that bar charts and critical path networks, the basis of the majority of project planning systems, are unsuitable for planning, monitoring and controlling the building design process because they neither accommodate the iterative nature of design nor permit the choice of alternatives. Design managers are therefore in need of more sophisticated tools and techniques to both co-ordinate design across different design disciplines and to effectively plan and manage the design process. These tools must aid design managers in planning design by taking into consideration its iterative nature and foreseeing the effects of change that affects communications and information transfer issues during the design process.

Computer based simulation offers significant potential for such sophisticated tools and is already an accepted technique for improving construction productivity. Many researchers have made use of computer based simulation techniques. Examples of research in this area are the work undertaken by Halpin (1977-1992) who developed a simulation software to simulate cyclic construction operations and Dawood (1991) who developed a computer based capacity planning system for precast concrete production. However, a literature survey showed no application of the simulation techniques in the area of the management of the design process.

A realisation by the writer that more efficient design management could be possible by modelling and improving the flow of information between all the parties concerned with the building and that simulation could assist in the development of tools to assist management led to the hypothesis and hence aim and objectives of the research.

1.2 RESEARCH HYPOTHESIS

The hypothesis of this research is that "existing planning techniques are unsuitable for the management of the design process. Techniques based on a combination of Data Flow Diagrams, Matrix Analysis and Discrete Event Simulation will improve the management of the Conceptual, Schematic and Detailed design phases". This hypothesis will be tested through the aim and objectives.

1.3 THE AIM AND OBJECTIVES OF THE RESEARCH

The aim of this research was to study, model and simulate the information flow during the building design process to allow analysis of the effects of typical events and hence improve the management of the whole process.

This aim was divided into the following objectives:

1. To study the nature of the design process in general and the building design process in particular.

2. To examine current practice for planning and managing the building design process.

3. To identify the main problems in design management.

4. To investigate existing models for the design process

5. To model the information flow between the different participants within the building design process.

6. To identify typical events and information related problems.

7. To develop a computer based simulation tool to predict the effects of the identified events and problems and produce design schedules based on these predictions.

8. To assess the benefits that the developed tools offer to improve the management of the design process.

The main emphasis of this research relates to the Conceptual and Schematic stages of design. However, it was recognised that the tools and techniques developed by the writer are applicable throughout the whole design process and therefore reference is also made to the detailed design stage.

1.4 RESEARCH METHODOLOGY

To meet the research objectives, the following research tasks were undertaken:

1. A comprehensive literature survey was undertaken to review the related text books, professional journals and publications concerning the nature of design, design management and its problems, information management, current planning techniques, concurrent engineering and modelling the design process.

2. The literature survey was supported by interviews held with construction industry professionals to identify both the current practice for managing the design process and the main problems in design management.

3. A literature review was undertaken to investigate the feasibility of applying structured analysis diagramming techniques to model the design process. This included a review of the different categories of structured analysis diagramming techniques.

4. Two case studies were undertaken to form a basis of a Generic Data Flow Model for the Conceptual and Schematic design stages

5. A Generic Data Flow Model for the Conceptual and Schematic design stages was developed. The model was constructed using a proprietary CASE (Computer Aided Software Engineering) tool and was based on data from the preliminary case studies. The model was validated by interviews with construction industry professionals.

6. Matrix partitioning techniques for the Design Structure Matrix were used to identify loops of iterative design tasks.

7. A survey and subsequent interviews were undertaken with design professionals to determine the main features required to be incorporated into the simulation model and to acquire feedback from the industry on the developed Generic data flow model.

8. A literature review was undertaken to determine the most suitable simulation technique to use in the research. This included a review of different simulation techniques regarding the phases of computer simulation, different considerations for simulation modelling and simulation applications in the field of construction

management. The literature review revealed a gap in the application of simulation techniques to the design process.

9. A discrete event simulation model was developed to simulate typical events and information related criteria that occur during the design process. The model was based on data from the Data Flow Model and the Design Structure Matrix in addition to dynamic factors input by the user such as the durations and resources of design tasks. The simulation model was rigorously tested and validated through subsequent case studies and review by industry practitioners.

10. A case study was undertaken to evaluate the developed tools and carry out further validation.

11. The benefits that the developed tools offer to improve the management of the design process were demonstrated through practical examples and feedback from design professionals. This included the extension of these applications to encompass the detailed design stage.

Feedback from the construction industry, verification and validation of the developed tools by industry professionals was an on-going process throughout the course of this research. Some 50 professionals from the construction industry were involved in the research on some 60 occasions in the form of survey, interviews, seminars, demonstrations and meetings.

1.5 ACHIEVEMENTS OF THE RESEARCH

The research set out to investigate the management of the design process and the development of sophisticated computer based tools to aid design managers. The main achievements of the research are summarised as follows:

1. The identification of the main problems in design management and the key factors for successful design management.

2. The identification of the deficiencies in current planning techniques when applied to design management.

3. The development of a Generic Data Flow Model for the Conceptual and Schematic stages of design.

4. The identification of the main parameters relating to information flow in design and their investigation which require assessment under different scenarios.

5. The development of a Discrete Event Simulation Model of the building design process.

6. A demonstration of the contribution of the developed tools to the management of the design process.

The identification of the main problems of design management confirms the importance of information as a key factor in the successful management of the design process. The identification of the deficiencies in current planning techniques when applied to the management of the design process confirms the need for new sophisticated tools and techniques to aid design managers. By the development of a Generic model for the Conceptual and Schematic stages of design a basis for such tools has been established. The Discrete Event Simulation Model for the design process provides a tool which may be used to investigate the main problem areas of design management and predict the effects of the associated problems. The feedback from designers confirmed both the viability and the usefulness of the developed tools to aid design management.

1.6 A GUIDE TO THE THESIS

The thesis consists of three parts which are divided into nine chapters. A schematic guide to the thesis is illustrated in Figure 1.1. A brief summary of each chapter is presented below :

PART 1 INTRODUCTION

Chapter 1 Introduction

This chapter explains the background to the research, the aim and objectives and the research hypothesis. The work undertaken to achieve the objectives, main achievements and the guide to the research are also presented.

PART II REVIEW OF LITERATURE AND CURRENT PRACTICE

Chapter 2 The building design process and its management

This chapter reviews the relevant literature regarding the nature of design, the management of the design process, planning techniques, information management and previous work undertaken by researchers in the subject area. The current practice for design management and the problems encountered during managing design are also presented.

Chapter 3 Modelling the design process

This chapter reviews the existing models of the design process, different categories of structured analysis diagramming techniques and different techniques for simulation modelling. The use of structured analysis diagramming techniques to model the design process and the different options of simulation tools for the research are also investigated. The chapter concludes with the research hypothesis.

PART III EXPERIMENTAL WORK

Chapter 4 Research method

This chapter explains the research methodology that was adopted to meet the research objectives. Justifications for the techniques used within the research are also included.

Chapter 5 A Generic model for the Conceptual and Schematic design stages

This chapter describes the development and validation of a Generic Data Flow Model for the Conceptual and Schematic stages of design. Two preliminary case studies are also presented.

Chapter 6 The simulation model

This chapter explains the development and verification of a Discrete Event Simulation model for the design process. A description of the simulation model and its operation is given.

Chapter 7 Evaluation of the developed tools

This chapter describes a detailed case study which was undertaken to evaluate and validate the developed tools.

Chapter 8 Improving the management of the design process

This chapter presents practical examples to demonstrate the benefits that the developed tools offer to improve the management of the design process. Feedback from the industry on the developed tools is also described.

PART IV CONCLUSIONS AND RECOMMENDATIONS

Chapter 9 Conclusions and recommendations

The main conclusions of the research and recommendations for further research are presented in this chapter.

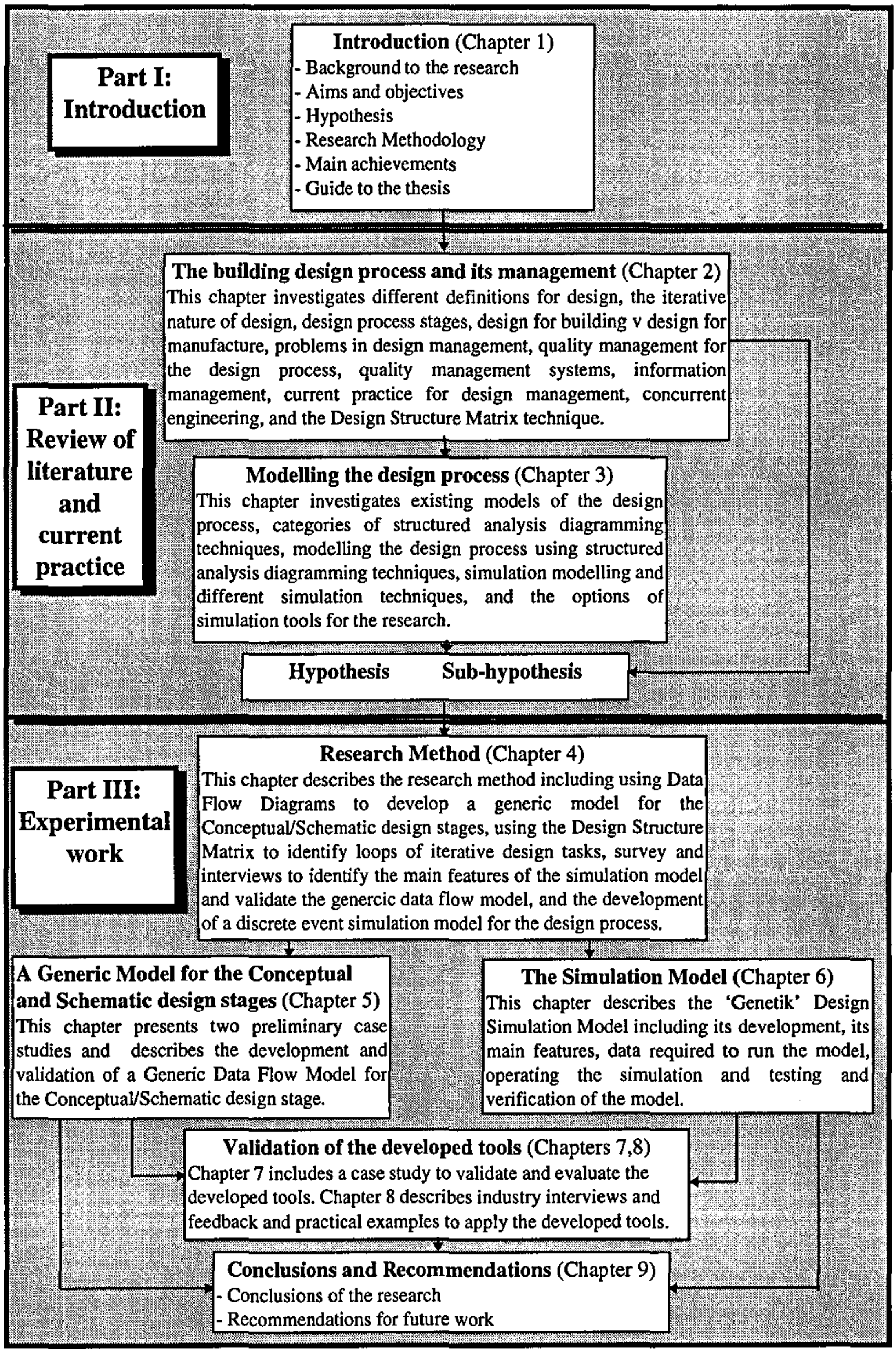


Figure 1.1 Guide to the thesis

PART II

REVIEW OF LITERATURE AND CURRENT PRACTICE

CHAPTER 2

THE BUILDING DESIGN PROCESS AND ITS MANAGEMENT

2.1 THE DESIGN PROCESS

2.1.1. Definitions for Design

Due to the very broad scope of the word 'Design', its definitions in references vary according to its area of usage. These range from meanings in language and academic dictionaries to definitions produced by researchers in the subject area. The Oxford English Dictionary (1989) provides nine meanings for 'design' as a noun and sixteen meanings for 'design' as a verb. Within the scope of this research, the most applicable of these for 'design' as a noun is:

"A plan or scheme conceived in the mind and intended for subsequent execution; the preliminary conception of an idea that is to be carried into effect by action; a project".

The most applicable meaning for 'design' as a verb is:

"To form a plan or scheme of; to conceive and arrange in the mind, to originate mentally, plan out, contrive".

Although these meanings show the 'output' of design, they do not encompass any pre-requisite inputs for 'design' nor interfaces of 'design'. This critique is also valid for the definition provided by the Academic Press Dictionary (1991) which introduced technical terms to its definition. It defines 'design' as "a scheme for the construction and ornamentation of a building, composed of plans, elevations, renderings and other drawings". However, this definition lacks other outputs for design such as specifications and justification as described by Addis (1990).

Researchers have defined design in different ways influenced by their specific area of research. Within the context of mechanical design, Culley et al (1992) defined design as the use of scientific principles, technical information and imagination in the definition of a mechanical structure, machine or system to perform pre-specified functions with maximum economy and efficiency. Cross (1989) describes design through defining the design problem. Design begins with a need that has not been satisfied because of certain obstacles or gaps. The finding of means to overcome these obstacles or gaps constitutes the design problem. Design problems usually have a set goal, some constraints within which this goal has to be achieved and some criteria by which a successful solution might be recognised (Cross 1984). Pahl and

Beitz (1988) defined designing, based on German references, as "the intellectual attempt to meet certain demands in the best possible way". Engineering design that impinges on nearly every sphere of human life, relies on the discoveries and laws of science and creates the conditions for applying these laws to the manufacture of useful products.

In an attempt to cover different perspectives of design, Pahl and Beitz (1988) described design in different respects including psychological, systematic and organisational. In psychological respects, they defined design as a creative activity that calls for a sound grounding in mathematics, physics, chemistry, mechanics, thermodynamics, hydrodynamics electrical engineering, production engineering, materials technology and design theory, together with practical knowledge and experience in specialist fields. Initiative, resolution, economic insight, tenacity, optimism, sociability and teamwork are qualities that will stand all designers in good stead and are indispensable to those in responsible positions.

In systematic respects, Pahl and Beitz (1988) defined designing as the optimisation of given objectives within partly conflicting constraints. Requirements change with time, so that a particular solution can only be optimised in a particular set of circumstances. In organisational respects, they described design as playing an essential part in the manufacture and processing of raw materials and products. It calls for close collaborations with workers in many other spheres. Thus, to collect all the information he/she needs, the designer must establish close links with salesmen, buyers, cost accountants, estimators, planners, production engineers, materials specialists, research workers, test engineers and standards engineers. A good flow of information and regular exchange of experience are essential and must be encouraged by proper organisation and personal example.

Although in their definitions Pahl and Beitz attempted to cover different aspects of design as an 'activity', the writer finds it more appropriate to describe design as a 'process' composed of different 'activities' or 'tasks' which reflect the different design aspects.

Neville (1988) defined design as a process which maps an explicit set of requirements in to a description of a physically realisable artefact which would satisfy these requirements plus implicit requirements imposed by the domain and/or the environment.

The writer concurs defining design as a 'process'. However, the definition provided by Neville defines design in general terms and could be used as a basis of developing definitions for specific areas of design. A similar approach was adopted by Gupta and Murthy (1980) who defined designing as "to suggest or outline ways to put together manmade things, or to suggest modifications in manmade things to satisfy optimally (under the given constraints) some specified human needs."

Hence, from the previously mentioned definitions, the writer regards 'design' at a contextual level as a process that requires certain inputs to produce a set of agreed upon outputs. Therefore, the writer's definition for building design is:
"A process which maps an explicit set of Client's and end users' requirements to produce, based on knowledge and experience, a set of documents that describe and justify a project which would satisfy these requirements plus other statutory and implicit requirements imposed by the domain and/or the environment"

2.1.2 The Iterative Nature of Design

Gupta and Murthy (1980) described the nature of the design process as consisting of three phases:

- Explorative phase

  This phase starts with a description of the need (brief). The aim of the design is to get as much understanding of the problem as possible.

- Transformation phase

  This is the creative phase wherein the designer summons all his experience, innovative capabilities, insights and genius to think up plausible schemes for achieving the desired result.

- Convergence phase

  In this phase the designer attempts to eliminate the unworkable and not-so-good solutions thrown up in the creative search for ideas and he attempts to converge on to the best solution (under the given conditions for the problem).

These three phases are represented in Figure 2.1.

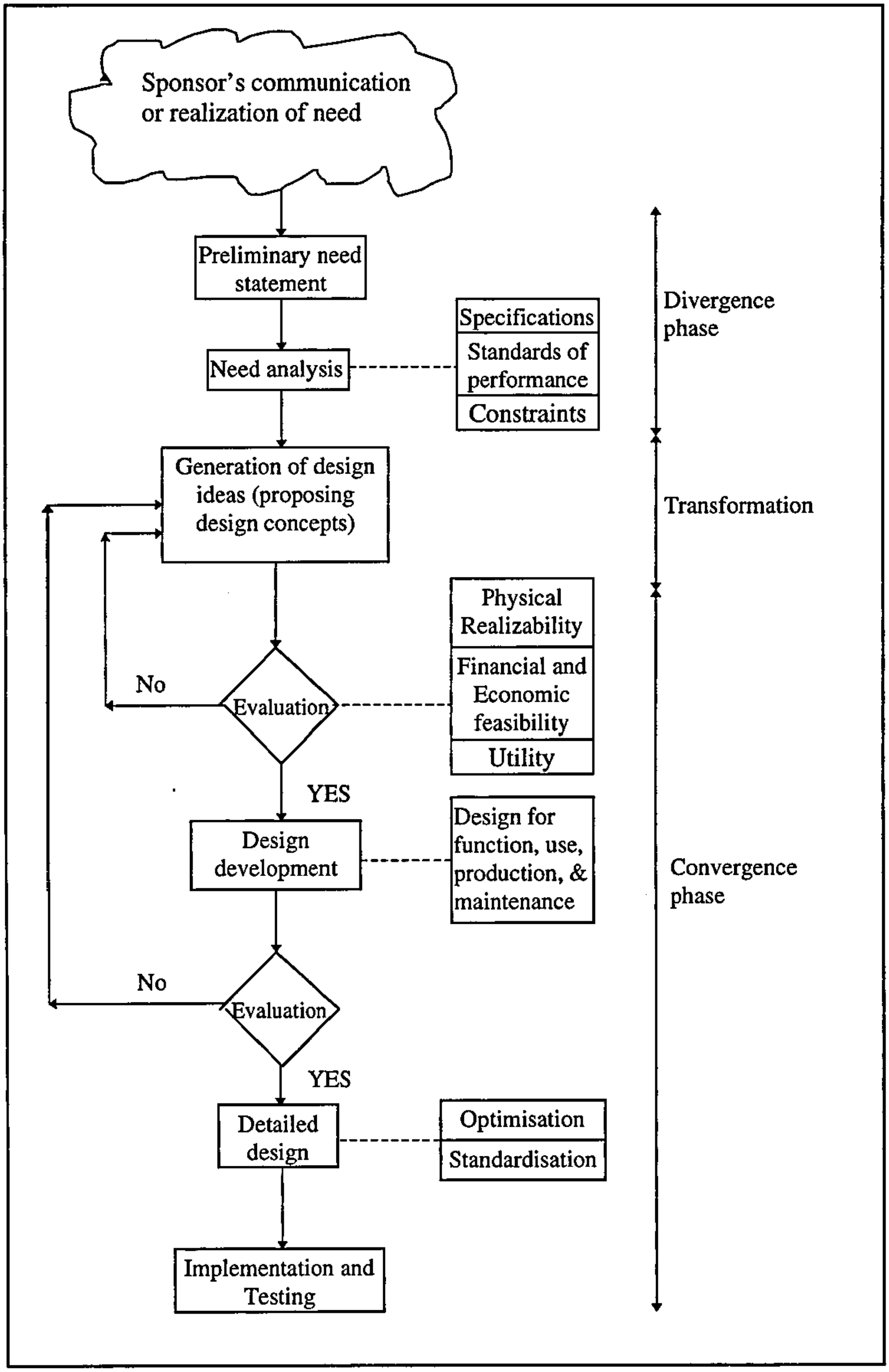


Figure 2.1 Morphology of Design (adapted from Gupta and Murthy 1980)

From Figure 2.1 it can be seen that the design process is of an iterative nature. An iterative mathematical procedure is one in which an approximate solution to a problem is initially guessed and then fed into a formula which reveals a more accurate solution. The improved solution is then put through the same procedure to reveal an even better solution and the process is continued until a solution of the required accuracy is achieved. The overriding principal is that the error decreases with every successive solution.

A systematic design/re-design procedure must inevitably form a similar pattern to such mathematical processes. This is due to the fact that there is no 'one solution only' to any design problem and that any design problem is full of ambiguities at its early stages. However, according to the previously mentioned definitions for 'design', any solution should satisfy all the pre-defined requirements and lie within the boundaries of the given constraints. The iterative design procedure makes the realistic assumption that even the best design concepts may have to be modified for improvement at various stages in their development. With complex components, the modified versions may need further improvement until the ideal solution is achieved. An efficient iterative process will ensure that each successive modification is less involved than the previous one (Hawkes and Abinett 1985). This iterative feature was confirmed by Cornick (1991) who indicated that traditional and current theoretical models of how building designers process their thoughts suggest the possibilities shown in Figure 2.2.

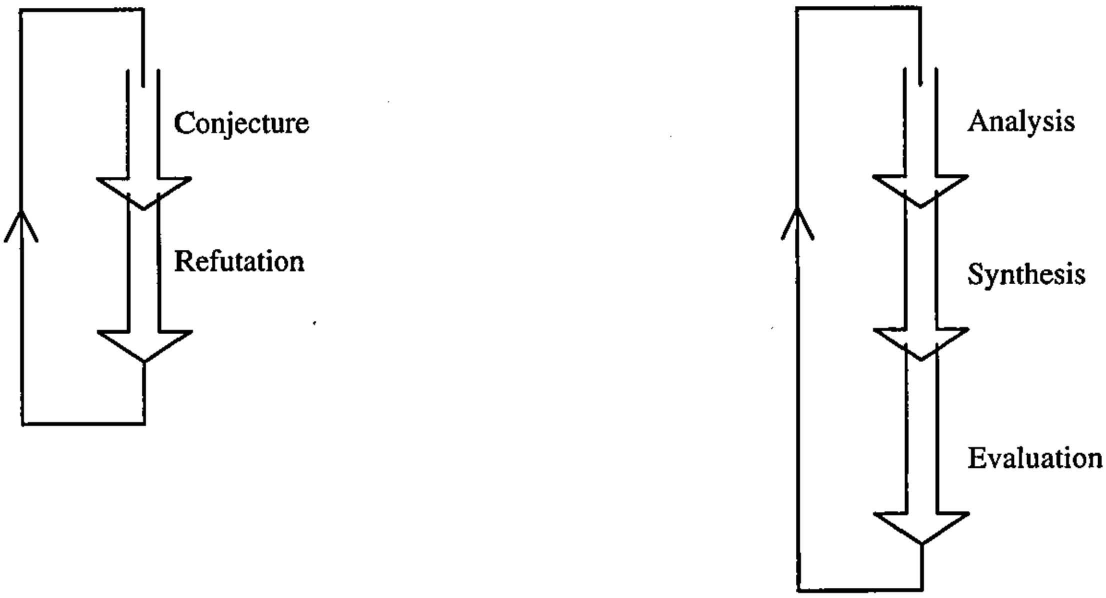


Figure 2.2 Cornick's possibilities of how building designers process their thoughts (adapted from Cornick 1991)

This iterative nature of the design process makes it complex and difficult to manage without the support of aiding tools. In straightforward design situations the tools for the management of the design process are simple. As the complexity of design increases, managers require more sophisticated tools. Cornick (1991) and NEDC (1987) showed that problems caused in modern buildings are more likely to be due to deficiencies in managing communications during the design process than to merely technological factors. Therefore, these tools must aid design managers in planning design, taking into consideration its iterative nature, and foreseeing the effect of changing different parameters that affect communications and information transfers during the design process.

The importance of such tools is also stressed by Newton (1995) who emphasised that manipulating information flows through successive stages of the design phase is the key to successful design management. It is the need for increasing sophistication in design management tools that formed the impetus for the application of simulation techniques and the development of the simulation model and the work within this research.

2.1.3 Stages of the Design Process

Although the incremental stages in the design process have been represented in numerous forms, there is no consensus among researchers on the terminology of these different stages. The main building block of each stage representing the natural evolvement of design was identified by Evans et al (1982) as Analysis→Synthesis→ Evaluation. These phases were endorsed by Jergeas (1989) and by Cornick (1991) who showed a feedback loop from evaluation to analysis to illustrate the iterative nature of design. (See Figure 2.2). Jones (1981) defined the phases of design in the sequence a design problem is solved: Divergence→Transformation→Convergence. However, the writer argues that the above mentioned phases represent the 'horizontal' dimension only of the design stages which is highly dependent on the designer's ways of thinking and hence is difficult to formulate. The 'vertical' dimension of the design stages which show the progress in a design project from concept to detail is associated with contractual and organisational aspects and hence attracted different researchers to formulate and stipulate the design tasks to be undertaken in each stage. Venegas (1987) provides a summary for the different terminologies used to define the progressive stages for design. Edel (1967) describes these stages as Initiation→ Exploration→Concept Formulation→Preliminary Design→Detailed Design→ Modifications. Beakley and Chilton (1974) define these stages as Feasibility Study→

Preliminary Design→Detailed Design. Ahuja (1984) divides the design process into Conceptual, Preliminary and Detailed design phases and gives a list of typical activities for each stage. Cornick (1991) defines the different stages of design as the briefing Phase, the designing phase-scheme, the designing phase-detail and the specifying phase.

The most well recognised model for the different stages of a construction project in the UK is the RIBA plan of work (RIBA 1973) which divides construction projects into twelve well defined stages from inception to completion and feedback. The three main stages of design formalised by the RIBA are:

- Stage C: Outline Proposals
- Stage D: Scheme Design
- Stage E: Detail Design

These stages overlap with the earlier stage B (feasibility) and the later stage F (production information).

Due to the popularity of the RIBA plan of work and the familiarity of most of the construction industry professionals with its different stages, the writer decided within the context of this research to consider the design process as consisting of the three RIBA main stages : C, D, and E. However, it was noted in practical terms, design professionals are more comfortable with using the term 'Concept Design' in lieu of 'Outline Proposals' for stage C.

2.1.4 Design for Building v Design for Manufacture

A literature search in the area of design revealed that most of the literature was related to manufactured product design. However, the nature of building design is not fundamentally different from the nature of manufactured product design in being an iterative procedure. Dias (1993) identified the difference between a building and a mechanical engineering product in terms of information requirements. This is shown in table 2.1.

Cornick (1991) indicated that the design of buildings as a process is fundamentally no different from any other artefact which has technological implications.

Usmani and Winch (1993) classified different writers with regards to their views to the design process in construction and manufacturing as being 'integrators' and 'separators'. Integrators are those who believe that the design process although unique in itself, is not affected by the product or process, while a limited number of separators propose that design processes are dependant on their product or processes. They argue that by defining projects as a flow of information through time, a commonality of approach between all schools of thought can be developed. This is because although the information content of a project is dependant upon the product, the flow of information throughout a project has certain impartial characteristics in common in construction as well as in the manufacturing industries.

	Building	**Product**
Space/Solid ratio	high	low
Detail	less important	more important
Shape data	topology	geometry
Fabrication	singular	multiple
Re-use	form	copy
Communication	inter-organisational	intra-organisational

Table 2.1 The difference between a building and a mechanical engineering product as identified by Dias (1993)

This view coincides with the recent move of researchers towards considering construction as a manufacturing process; the building being the manufactured product. Examples are Fisher (1993) who argued that applying Knowledge Based Engineering (KBE) to building design will enable regular clients of the construction industry to specify identified key standard components, that they have developed separately with specialist manufacturers, to be incorporated into their building. He showed also that a building is analogous to a manufactured product in that it must work internally as a 'system' and must represent to the owner good value for money.

Another example is Huovilla et al (1994) who linked the fast track (overlapping of design and construction activities) approach to construction projects with concurrent engineering (integrating product development with its manufacturing process) applied in the manufacturing industry. They concluded that although the two approaches are different, several methods and techniques originating from concurrent engineering have been implemented in fast tracking projects and fast tracking has thus started to integrate into concurrent engineering. (Concurrent engineering is described in more detail in section 2.2.6.)

Since the design process is information driven for both building design and product design, the writer advocates the concept of defining design projects as a flow of information through time. This is a compromise between the two schools of thought of the integrators and separators and was one of the reasons for adopting information flow modelling techniques in this research for modelling the design process. This is explained in more detail in chapters 3 and 4.

2.2 THE MANAGEMENT OF THE DESIGN PROCESS

2.2.1 Problems in Design Management

Due to the complexity of the design process, its iterative nature, and the various constraints imposed on it, design projects are often difficult to manage. A survey was carried out (Topalian 1979) to generate data on the difficulties encountered when managing design projects. This survey was conducted on 242 managers/clients from the UK and Canada who were asked to indicate agreement or disagreement with 28 statements on difficulties perceived in managing design projects. These statements were elicited from managers and designers when discussing design. The survey showed that 9 out of the 28 reasons for the difficulty in managing design projects are related to amount and/or timing of information transfer.

Bennett et al (1988) in their report "Building Britain 2001" showed that the traditional pattern of fragmented design practices is being replaced increasingly by multi-disciplinary practices which encourage and ease information transfer between professions but have the disadvantage that the communication is often informal and not documented. This makes the management of the multi-disciplinary design projects more difficult as it requires immense co-ordination to ensure all parties are constantly aware of the every-changing status of the project in an attempt to eliminate design errors and limit design changes.

One of the factors that increase the complexity of managing design is the nature of the design problem and its solution (Lawson 1980, Price 1995). This complexity was confirmed by the York Institute of Advanced Architectural Studies (Ahuja 1994) which identified the following conflicts as inherent in design work:

- Inherent complexity of design
- Uncontrollable delays due to information form clients, site acquisition, cost cutting or statutory approvals which result in difficult resource planning

- Fragmentation of design work due to involvement in multiple projects at different stages
- Shortage of time

Price (1995) summarised the problems encountered by contractors as a result of design deficiencies based on the findings of Jergeas et al (1990) and Moxley (1993). As a response to the views expressed by contractors, Price (1995) summarised also the designers views of the causes of major design problems. It was found that apart from the technical problems related with the designers' experience and expertise the main problems related with design are information related. This shows that successful information management is a fundamental contributor to eliminating design problems.

The importance of managing design information was also emphasised in a report by NEDC (1987) which showed that many problems on building sites were related to inadequacies in design information. These problems were often found to be more significant than those attributed to poor workmanship and site management. This report was a motive for a study undertaken by Coles (1987) and sponsored by the RIBA to investigate the factors affecting the design management practice in the building industry. The findings of the study showed that the most significant factors which interfere with the smooth production of technically competent designs and information for construction are:

- poor briefing and communications
- inadequacies in the technical knowledge of designers
- a lack of confidence in pre-planning for design work

The last factor highlights the inadequacies of the current planning techniques used in planning design and confirms the need for more sophisticated tools to manage the design process.

In an article on design management in building, November 1993 (Builder 1993), the difficulties arising in the management of the design process due to information transfer problems were also highlighted through interviews with construction management experts. "It is the failure in the supply of information that really has to be addressed. Drawings fail to appear at the right place, at the right time or decisions are made too late" (C Gray, Builder 1993). "Good design management involves allocating the right amount of time and manpower to ensure that drawings are produced on time. It must ensure that information is consistent, that it contains no unresolved detail and that the design meets the Client's requirements in terms of quality and cost" (Winch, Builder 1993). "Design management does not produce drawings or come up with ideas any quicker. But it does allow design changes to be tracked as they arise and stops problems occurring when it's too late" (Mackenzie-Carmichael, Builder 1993).

In a handbook for the successful management of design, "A handbook of building design" produced by University of Reading (Gray et al, 1994), ten steps to good design management were identified. Six steps emphasised information transfer and communication issues and showed that these issues represent the key to successful design management with respect to:

(i) Timing of information transfer
(ii) Quality of information exchanged
(iii) Identifying and understanding other participants' information needs
(iv) Means of information transfer

This is confirmed by the results of a survey supported by subsequent interviews conducted by the writer on professional staff within three major construction organisations in the UK namely Ove Arup and Partners, AMEC Design and Management and Kyle Stewart. A survey document was issued to a total of twenty construction professionals with different disciplinary backgrounds and managerial responsibilities. Twelve of these construction professionals were subsequently interviewed. One of the objectives of the study was to identify the main difficulties encountered by design managers during the Conceptual and Schematic stages of design. However, throughout the course of the research the writer found that these difficulties were applicable, with variable significance, on <u>all</u> the stages of design or had a significant impact on the rest of the design stages. These difficulties fall into four broad categories:

<u>Client related difficulties</u>
(i) Frequent changes with lack of appreciation of the impact of changes
(ii) Client communicating only what they think is important
(iii) Decision making by the Client
(iv) Loose brief
(v) Establishing a relationship with the client
(vi) Fulfilling Clients' actual requirements

Project related difficulties

(i) Time scale

(ii) Identifying project objectives

(iii) Allocating appropriate resources

Planning difficulties

The order of design tasks is determined in a very broad and global way. Frozen layouts were considered as a very important milestone where all design disciplines can proceed on its basis.

Information management and communication difficulties

(i) There is no formal way to judge the quality of information as it is highly dependent on the sender and recipient of information

(ii) Problems resulting from missing design information

(iii) Ensuring that all parties are aware of each others activities and requirements

(iv) Co-ordination of all design disciplinary information

(v) Communication problems among team members are summarised as follows:

- conflicts due to different personalities and human behaviour issues;
- lack of appreciation of the effects of changes across disciplines;
- unavailability of some team members during meetings due to work in other projects;
- geographical distances between team members;
- lack of awareness of some disciplines for other disciplines' problems leading to thinking that others are asking irrelevant questions;
- designers of each discipline do not know what other disciplines are expecting them to provide;
- speed of this design stage can prevent team members of becoming adequately familiar with each other or with the Client;
- lack of experience for some disciplines to advise other disciplines without carrying out the actual design;
- passing information between disciplines;
- agree at which stage will the design development be frozen; and
- engineers pressurising Architects to provide scheme drawings quickly to enable them to start their design.

(vi) 'Gate keeping' or withholding of information either intentionally or non intentionally as a result of the above mentioned communication difficulties. This is backed by Guevara and Boyer (1981) and Roberts and O'Reilly (1974) who identified 'gate keeping' of information as one of the communication problems within the construction industry. Further details about the survey and interviews are included in Appendix II of this thesis.

The survey and interviews showed also that the management of the Conceptual and Schematic stages of design is more complex than managing the detailed design stage. This is because the Conceptual/Schematic design stages represent the 'front end' for the detailed design stage and problems in managing the early stages will affect the whole design process. The decisions made at the early design stages have a major influence on the overall project costs while the cost of change is minimal. The majority of the communication problems occur during the early stages of design (Hunter, 1993). During one of the interviews undertaken by the writer, a design manager described the detailed design stage as a 'production stage' where the design criteria have been established and every design input leads to 'tangible' output(s). However, during the course of the research, the writer found that the information related difficulties and the planning difficulties are valid for all the stages of design. This is explained in more detail in sections 2.2.5 and 8.11.

To summarise the results of the extensive literature search, survey and interviews undertaken by the writer, the problems in design management are categorised into problems due to the inherent nature of design (such as the iterative nature of design), problems due to technical aspects of design (such as lack of technical knowledge for designers), Client related problems (such as frequent changes with lack of appreciation of the impact of changes), problems due to difficulties in managing information (such as the problem of missing information) and problems due to difficulties in planning design (such as inadequacy of existing planning techniques). Of these categories, the last two have been shown to be of great significance to the successful management of the design process. Therefore this research has concentrated on these categories of problems.

2.2.2 Quality Management for the Design Process

The need for efficient management of the design process coupled with the emergence of total quality management principles had dictated the requirement to apply these principles to managing the design process. It is not the intention here to define the different 'quality' related terms: 'quality assurance', 'quality control' and 'quality management'. However, the writer is presenting the relative meaning of these expressions as follows (Cornick 1991):

- The aim is 'quality', which is defined as conformance to requirements.
- The method is 'management', which allows for improvements so that non-conformance to requirements can be corrected.
- The result is 'assurance' by demonstration that conformance to requirements has occurred.
- The mechanism is 'control', which ensures that improvement and assurance can always occur.

Therefore, the requirement of a quality management of any process (design) is that its system of control can ensure that conformance to requirements can be assured. This assurance must be demonstrated in formal procedures which can capture any non-conformance to the requirements. The non-conformance can hence be corrected through management for improvement

In this research, the importance of the information transfer aspects to the successful management of the design process highlighted in section 2.2.1 is linked to the concepts of 'quality management' in two ways:

(i) As a part of the quality assurance procedures of design organisations, the information requirements and outputs for every design task should be identified and used as a 'checklist' to which the exchanged information should conform. It is the need to identify these information requirements which instigated the adoption of data flow modelling techniques to model the design process as explained in chapters 3, 4 and 5.

(ii) It is important to 'assure' and 'control' the quality of information exchanged during the design process. Although some researchers attempted to establish measures for information quality, there have been no consensus over such measures. Marchand (1990) identified eight dimensions as a framework for analysing the quality of information : actual value, features, reliability, relevance, meaning over time, validity, aesthetics and perceived value. Ronen and Spiegler (1991) outlined various dimensions of information identified by Ahituv and Neuman (1986) as accuracy, timeliness, detail and scope. Schwuchow (1990) pointed out that although information quality is an important aspect, it is difficult to find an overall measure for it. Wagner (1990) showed that measuring information quality is subjective, situation dependent and varies over time.

Hegedus (1990) suggested that the quality of information can be measured by satisfaction of users.

The views of Wagner (1990) and Hegedus (1990) were confirmed by the results of the survey and subsequent interviews undertaken by the writer. These results showed that there is no formal way to judge the quality of design information. The measure of good quality is if the information provided is enough to proceed to a next stage in design. Design information is considered of poor quality if the information is insufficient or unsatisfactory for the recipient. (Details of the survey and interviews are included in Appendix II.) This recognition of the importance of the information quality aspect has been reflected in the simulation model developed by the writer which includes the simulation of information quality as one of the features of the simulation model. This is explained in more detail in chapter 6.

Introducing quality management concepts to the model of how building designers process their thoughts has been illustrated by Cornick (1991). The result is shown in Figure 2.3 which is an adaptation of Figure 2.2.

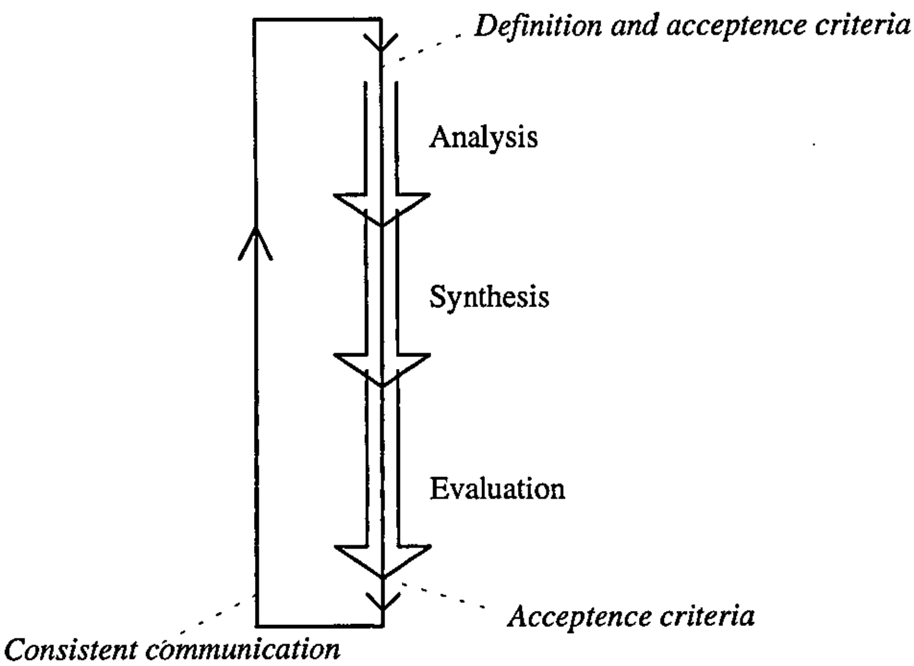


Figure 2.3 Including quality management concepts to Cornick's model (adapted from Cornick 1991)

2.2.3 Quality management systems

The recognition of the importance of 'quality' aspects to the design process resulted in a need to apply formal standards of quality management systems to the process. These standards include the BS 5750, EN 29000 and ISO 9000 on the British, European and International level. Although these standards and their structure are different, the basis for them and their contents are exactly the same (Cornick 1991). The different sections laid out in these standards describe the specifications and requirements of quality management systems, and the guidance for implementation of such systems for both design and production processes.

At the time of writing this thesis, a new Quality Standard for design management systems - the BS 7000 - was under development. The BS 7000 comprises initially of four parts, one of which is dedicated to managing design in construction (BS 7000 Part 4 1994). It is not within the scope of this research to discuss the different standards for quality systems, however, there is a particular interest in sections 3.9 and 3.10 of Part 4 of the BS 7000, entitled Communications and Management information respectively. These sections highlight the importance of having a communications policy which ensures that those concerned in design are informed about everything that may affect what they are doing without being inundated with irrelevant information. They also show that accurate and timely information is essential to enable managers to perform their duties effectively and that clear instructions should be issued to cover the following:

- what information is required, by whom and for what purpose;
- who will generate the information and maintain it;
- how it will be sorted and distributed;
- how frequently it is issued, if distributed regularly; and
- what actions should be taken on receipt of the information.

The previously mentioned 'quality systems' and 'standards' provide only guide lines to the 'quality management' of the design process. However, the emphasis on the communication and information transfer issues highlighted in these standards confirms the need for further research to identify the specific information requirements and communication routes and problems throughout the design process and assess the impact of different related criteria on the whole process. This may be supported by Usmani and Winch (1993) who regard the management of projects as the management of the information that is produced, evaluated and transferred.

2.2.4 Information Management

The importance of information management as a key factor in the successful management of a construction project has been increasingly recognised by construction industry researchers. Poor co-ordination between design and project information may lead to major communication breakdowns on construction sites and result in serious financial implications for contractors (Stephenson and Naylor, 1993). Section 2.2.1 shows the importance of managing information transfer during the design process to achieve successful design management. Stephenson and Naylor (1993) developed a prototype system to communicate, monitor and control design information during the production phase of a construction project. The construction industry, being of a heterogeneous nature, requires different companies, consultants and individuals to combine, discuss and exchange information at many levels. Each party has its own information system plus commitment to other systems and the arrangement is further complicated by the 'time status' of information (Price 1995). With the continuous growth of computer use and the increasing transfer of information between computer systems, it is inevitable that there will be dissimilar systems. A report by NEDC "Information transfer in building" (NEDC, 1990) showed the necessity of achieving a common understanding of the capabilities of the systems involved and the terminology employed.

The efforts undertaken by researchers to achieve such common understanding are summarised below:

Use of coding systems

Use of coding systems for drawings is recommended by NEDO (1987, 1990) and Latham (1994). The code is aimed at every drawing which is produced for use on site (Price 1995). It is supplementary to BS1192 Part I, 1984 and is applicable to any form of contract. However, to produce an effective set of drawings decisions about production and co-ordination must be made according to the circumstances of every new project.

Extending the application of coding systems to specification and bills of quantities followed a government sponsored initiative to look into the problems of inefficient, conflicting or incorrect project information. This led to the formation of the co-ordinating committee for project information CPI (1987) which developed a code based on work sections called "Common Arrangement".

Allocating status to information

Timely review and approval of information represents an important factor for a project success (Tiong, 1990). Computerised systems such as Database Management Systems (DBMS) may be used to facilitate document monitoring and control. Documents should incorporate all the information relevant to their issue including: originator, production date, recipients, status, revision issue and date and action needed.

This coincides with the views of industry practitioners outlined through interviews undertaken by the writer during the course of this research. Design managers suggested that adding status to any issued information will assist the information recipient to judge the quality of this information. Results of these interviews, which were preceded by an initial survey are included in Appendix II of this theses.

Standardising Computer Applications to facilitate data exchange

The main aim of standardisation of computer applications and achieving common understanding among the different systems of the project participants is to maintain fast efficient way of data exchange between different systems. A typical example is the requirement to exchange data among different CAD (Computer Aided Drafting) systems. The BS1192 Part 5 "Guide for Structuring of Computer Graphics Information" aims to "give guidance and recommendations on the production of graphical information needed to provide communication with accuracy, clarity, economy and consistency of presentation between all concerned with the construction industry including architects, civil engineers, contractors, landscape architects, services engineers, site operators, structural engineers and surveyors". It is a standard aimed at users and managers of CAD systems and not producers (NEDC 1990). The main recommendations of the standard are concerned with providing an understanding of the commonalties between CAD systems, increasing the efficiency of use of CAD, organising the transfer of CAD data between several offices and structuring data for archiving to help future retrieval. BS1192 recognises the following format for data exchange: Initial Graphics Exchange Specification (IGES), the Product Data Exchange Specification (PDES) and the more recent Standard for Exchange of Product Data (STEP). The International Standard for STEP is ISO 10303 and applies to all products including buildings (Price 1995). Although STEP is utilised within some organisations, it is still undergoing significant development and only limited understanding of the technique exists in the industry (Griffin et al 1994). The importance of the previously mentioned standards and systems for data exchange and information transfer are acknowledged by the writer, however, it is not intended within this research to discuss further details of these standards and/or systems.

An effective way of exchanging data among different CAD systems is the agreement on 'layers' (Day and Faulkner 1988). Most CAD systems use layers, each being used to hold information of a different type such as architectural data, structural data and services data. Information transferred from one system to another will be placed on the same layer in the receiving system as it was on the sending system. Hence, it must be ensured that information is transferred to an empty layer or to one on which the overwriting of information is acceptable. It is therefore possible to establish a layer convention which allows each user access to the same layers within their own systems but which keeps a block of layers for information transfer (NEDC 1990). Data is generally transferred by DXF, a proprietary format originally devised by Autodesk (Autocad user group and Autodesk, 1991). This convention is currently adopted in industry on a large scale with copies distributed to over 10000 companies fully supported by CICA with the objectives of rationalising information transfer and creating a common user environment. This is in addition to other standard layering techniques based on BS1192 part 5 such as those developed by Ove Arup (CICA 1994).

The literature surveyed by the writer showed that significant amount of research has been undertaken recently in the area of managing information exchanged between different parties of a construction project. Such research focuses on the stage of production of contract documents. However, only limited research has been undertaken on the management of information exchanged during the earlier stages of design. This exchange usually takes place within the design organisation and between the design organisation and the Client's organisation. This work is mainly represented in the production of information matrices showing information transfer interfaces and requirements during the course of a project such as those described by Coleman (1992) and NEDC (1990). The absence of standard forms for information exchange at early design stages usually results in each organisation producing its own forms and including them as a part of their own quality assurance procedures. Examples of such forms are those produced by AMEC Design and Management to record and manage the information exchanged during the design process of a particular project. These forms include:

Project Design Notes (PDN)

PDNs are used to clarify, request or confirm project information to the client. They are used to set out the philosophy upon which design will progress and to confirm the client's acceptance of that philosophy. Each PDN is to include a unique number and be booked into the computer by the Group Administrator and be accompanied by a Document Issue Note.

Document Issue Notes (DIN)

DINs accompanying all documents are issued externally by AMEC. They are to include a unique number and be booked into the computer by the Group Administrator.

Contact Report

All telephone calls or informal discussions between the client or his agent on technical, commercial or construction matters to be recorded on a Contact Report and distributed internally and to the client.

Design Change Control Forms (DCC Form)

DCC forms record any change to the agreed design. The form can be originated by any member of the design team but must be agreed with the Design Leader prior to formal issue to the design team.

Design Variation Orders (VO Forms)

If a design change as recorded on Design Change Control Form constitutes a change to the scope of the agreed works, then the Design Leader will inform the Project Cost Estimator on a VO Form. The Cost Estimator will be asked to place an estimation of cost against the variations. The Design Leader will then inform the Client using the VO form. All VO forms are to be given a unique number and entered on to the computer system by the group administrator.

Examples of the above mentioned forms produced by AMEC Design and Management are included in Appendix I of this thesis.

From the literature survey, the writer deduced that the lack of research work in managing information transfer during the design stages prior to the production of tender documents is a main contributor to the deficiency in standardisation techniques for exchanging information at this stage. This gap in the literature was one of the reasons which led the writer to explore the information transfer in more detail at the

Conceptual and Schematic stages of design and instigated the idea of producing a Generic Model for Information flow at these stages. This is explained in more detail in Part III of this thesis.

2.2.5 Current Practice in Design Management

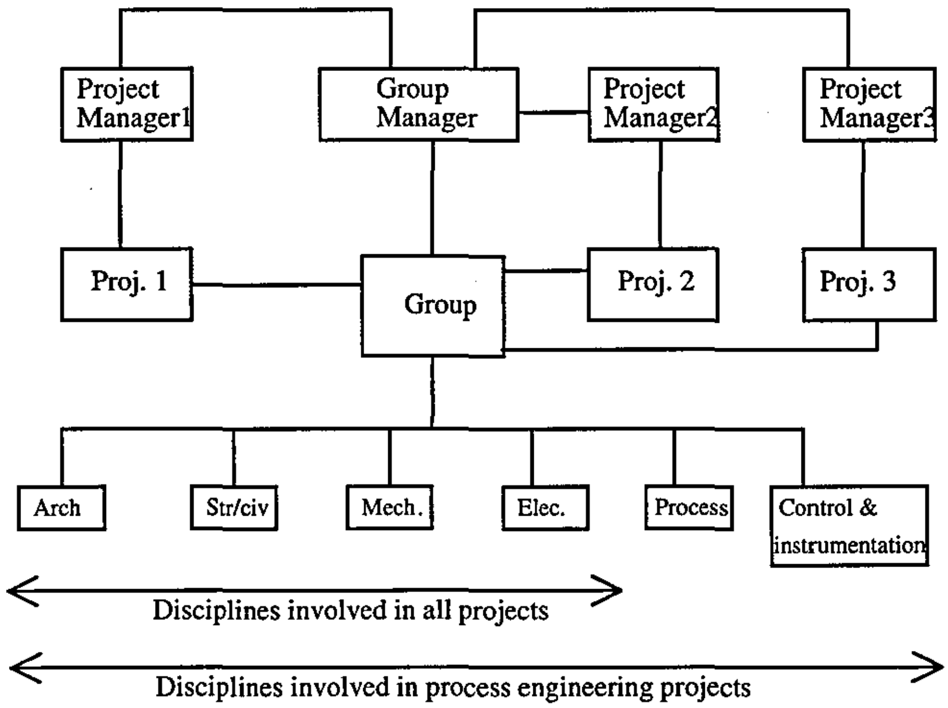
The current practice for managing and planning the building design process was investigated by the writer through interviewing design leaders from Ove Arup and Partners, AMEC Design and Management and John Laing. Due to the pressures exerted on design organisations by the Client, the design process is planned 'backwards' based on the date for tender or the date for operating the project according to the type of contract. A Master Programme in the form of a Bar Chart is produced by the Project Manager which includes 'global activities' such as Scheme Design and Detail Design together with milestones for key dates. The master programme is distributed to design team leaders of different disciplines and each team leader plans his design discipline within the frame of the Master Programme. This is achieved by identifying the number of drawings to be produced and calculating the man hours required for each design team to ultimately produce these drawings. The calculated man hours represent the basis for calculating the design resources to be allocated and the design costs that will be incurred. This information is passed to the planning engineer who will transform it into a series of design activities and consequently produce a schedule of these design activities. This schedule is usually updated weekly according to the design progress. Another technique that is also used is the Information Release Schedule (IRS). This technique is mainly used in the case of Design and Build procurement strategy. IRSs represent checklists of information and drawings that are required to be submitted at certain dates. There is a separate IRS for every subcontractor package. An example of IRS from a project undertaken by John Laing is included in Appendix I.

A technique known as Early Warning Systems (EWS) is used by some Design and Build Contractors to carry out a closer monitoring and following up of design activities. EWS is a mini-programme for every element of design work and covers working drawings, fabrication, etc. Every bar in the design programme is decomposed into every design element resulting in separate several bars representing decomposed design activities in more detail.

The way design teams are structured differs among different organisations. The graphical representations illustrated in Figures 2.4 and 2.5 represent two different design team structures.

Resources are managed in such a way that continuity of work is maintained for each design team member. This requires that a designer may be working in more than one project simultaneously. Separate reports may be produced for design work to be undertaken by each category of the design personnel (e.g. senior engineer, graduate engineer, technician etc.). Where sophisticated computerised systems are used, a central programme for the whole design organisation is used for controlling the cost of each job by entering data of different resources categories and the jobs to which they are allocated.

Within the frame of the Quality Assurance procedures of each organisation a quality plan is produced for every project by its design leader. The quality plan mainly includes the names of the project team members representing different parties, the procedure for communication between different participants, the procedure for circulation of information, and procedures for issuing design documents.



Note: - Every group is placed in the same floor within the organisation
- Each discipline is headed by a principal engineer

Figure 2.4 Design team structure #1

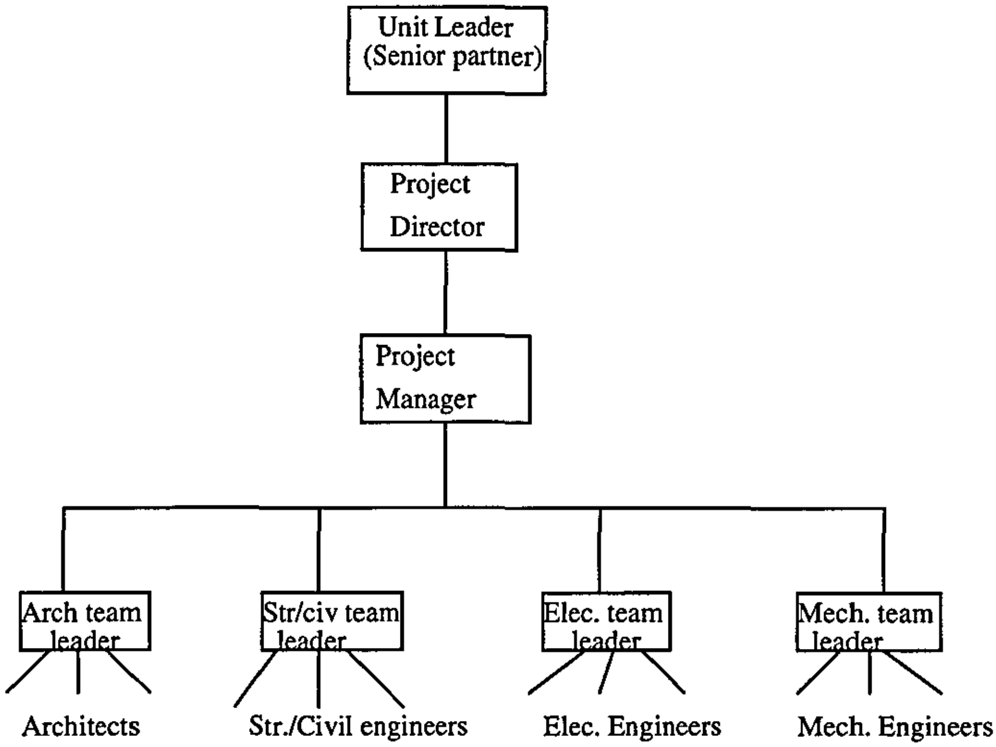


Figure 2.5 Design team structure #2

The rest of this section will discuss in more detail the different planning techniques and representations that are used to convert design activities into a design programme and the shortcomings of the existing planning techniques.

Planning is the creative and demanding mental activity of working out what has to be done, how, by when, by whom and with what i.e. doing the job in mind (Neale and Neale 1989). It involves envisaging how the job will be done, in what order and with what resources, so reducing the project to a number of manageable activities.
Early decisions has to be made regarding the project's duration. It may be either imposed by external consideration of the time available and the plan has to be made to meet this requirement or it is built up from a detailed analysis of each activity to be done and the resources available. This requires estimating the time required for each specific activity. Potential difficulties are foreseen to plan to overcome them and risks are anticipated so that their effects can be minimised.

Planning also involves scheduling resources to enable optimum use to be made of the available and most economic resources for each project and, taking all projects together, for the organisation as a whole (Neale and Neale 1989, Harris and McCaffer 1989, Barrie and Paulson 1991).

Planning techniques assist in the analysis of the plan, organising the information and have a crucial effect on the way in which the plan is communicated to others (Neale and Neale 1989). The literature reviewed by the writer showed that in spite of the iterative nature and complexity of the design process, planning techniques used by practitioners for the management of the design process do not differ from those applied in other areas of project management. The most commonly used are bar charts, network analysis (Critical Path Method) and PERT (Program Evaluation and Review Technique).

(i) Bar Charts

The simplest scheduling tool is the bar or Gantt chart. It is simple in concept, easy to construct and easy to understand. It represents the most widely used planning technique. This was confirmed through discussions undertaken by the writer with design professionals about the current practice in managing the building design process. The activities are listed in the vertical direction and elapsed time is recorded horizontally. A bar chart shows clearly the date by which each activity should start and finish but it does not show clearly the relationship between activities. (Dieter 1983, Neale and Neale 1989).

To overcome this shortcoming, a refinement has been introduced to bar charts where the planner links the horizontal time bars with vertical lines (links) to indicate the activities' logic producing a linked bar chart (Neale and Neale 1989).

A bar chart as a planning technique is suitable for simple projects, but for more complex projects more sophisticated tools are required which allow analysing the interrelationships of different activities. Nevertheless, sophisticated techniques, such as network analysis, still use bar charts as a communication tool for the results of the analysis due to its familiarity and ease of understanding.

It is the popularity of the bar chart formats which made the writer decide to use it as one of the formats for displaying the results of the simulation model. This is explained in more detail in chapter 6.

(ii) Network Analysis (Critical Path Method CPM)

Network analysis was developed in the US in the late 1950s by E. I. DuPont Co. to meet its construction project management needs (Jewell 1986). It is a general term for a graphical planning technique which shows the project as a network of its activities linked together to show their interrelationships and sequence of execution (Neale and Neale 1989). By estimating durations for the different activities, the diagram can be analysed numerically to determine the estimated project duration. This analysis also distinguishes between those activities whose timely execution is critical to the earliest completion of the project, and those which may be delayed for a specific time without delaying the project completion. Details of this technique may be found in Dieter (1983), Harris and McCaffer (1989), Barrie and Paulson (1992), and Neale and Neale (1989).

One of the powerful features of network analysis is that the logic diagram, activity durations and resources required may be considered separately although ultimately they are all interrelated. The advantage is that the planner may consider one of these components of the plan at a time rather than all at once. However, network analysis is suitable for planning deterministic activities which are either sequential or parallel such as construction activities. It is ill suited to plan activities with an iterative nature such as the design activities because it does not allow feedback loops or any iterative procedures. Therefore to apply network analysis to plan design activities, they should not contain any iterative loops or iterative loops should be 'unwrapped'.

This approach of 'unwrapping' iterative loops of design activities was used by the writer as the basis for one of the options to simulate the design process as will be explained in chapter 3.

(iii) PERT

The program evaluation and review technique (PERT) was developed by the US Navy in 1958 to assist the management of the Polaris missile project (O'Brien 1972). This technique uses the same basis as CPM but instead of using just the most likely time estimate for activity durations, it uses a probabilistic estimate of time for completion of an activity. Three time estimates are made for each activity: optimistic, pessimistic and most likely time estimate. The time estimates are assumed to follow a beta frequency. By calculating the expected time for each activity and its standard deviation which describes its scatter, the standard deviation along a path in the PERT network is calculated. Knowing the variance for each activity permits the calculation of the probability that a certain scheduled event will be completed on schedule and the probability that the project end date or key stages within the project will be completed on or before the scheduled dates. However, although PERT is a refinement to the CPM, it still does not allow feedback loops and cannot be used to plan activities of iterative nature as those involved during the design process. (Dieter 1978, Harris and McCaffer 1989, Barrie and Paulson 1992).

2.2.6 Concurrent Engineering

Concurrent Engineering (CE) is a technique mainly adopted in manufacturing engineering in the development of products by integrating design with other tasks such as the planning of manufacturing, quality and marketing (Belson 1994, Kusiak 1994). It principally aims at reducing the duration of engineering time, increasing the value of the product and reducing the costs, (Huovilla et al 1994). This is achieved by reducing the share of those activities which do not directly contribute to the conversion of requirements to the final design and by assuring that value is added by those activities contributing to this conversion. Concurrent Engineering is defined comprehensively in an Institute for Defence Analysis report as "a systematic approach to the integrated, concurrent design of products and their related processes including manufacture and support. This approach is intended to cause the developers, from the onset, to consider all elements of the product life cycle from conception through disposal, including quality, cost, schedule and user requirements" (Winner, 1988).
Belson (1994) describes the main characteristics of Concurrent Engineering as follows:

(i) Co-operation of multi-disciplinary teams while they simultaneously complete the development of a new product. Such parallel completion of tasks should be executed quicker than when doing the tasks sequentially, however, to achieve this, a number of technologies and tools is required.

(ii) The use of sophisticated electronic tools for drawings' production such as CAD and electronic communication of design data.

(iii) Application of rules to facilitate manufacture, assembly and inspection of the manufactured parts. These rules are known as Design for Manufacture (DFM), Design for Assembly (DFA) and Design for Inspection (DFI) respectively.

(iv) Provision of convenient, adequate meeting spaces equipped with all required facilities to maximise the efficiency of groups' interaction.

(v) Changes in the organisational structure from the typical pyramid structure of the manufacturing organisations to the multi-disciplinary approach to Concurrent Engineering.

(vi) The simultaneous nature of CE permits quality, from the customer's view point to be designed into the product from the start. The concept of the voice of the customer and techniques to incorporate customer interests to product features are an important directive to the CE team. One of these techniques which organise such matters in a structured way is known as Quality Function Deployment (QFD) (Zairi, 1994). QFD develops matrices that start with customers interests and then relate them to product attributes which in turn are related to product parts and processes. The design of the matrix is undertaken in a graphic way in order to focus attention on the important relationships and interrelationships.

(vii) Continuous assessment of the cost impact of every decision taken and on alternatives has to be considered.

(viii) Capturing lessons learned from design mistakes which led to manufacturing problems to avoid repeating the same mistakes. This can be achieved through combining knowledge based systems with the CAD systems of the organisation.

(ix) Recognition for all teams and employees participation.

The characteristics of CE has attracted some researchers recently to apply some concepts of CE to the construction industry. This coincides with the recent move towards considering construction as a manufacturing process.
Huovilla et al (1994) linked fast tracking approach to construction projects with CE. Both approaches aim at a shorter project duration through overlapping of the design and manufacturing processes. They have emerged as an alternative for the sequential approach of project realisation. However, Huovilla et al (1994) showed that there are major differences between the two approaches. In fast tracking, where design and construction activities are overlapped, uncertainty is increased in comparison to conventional sequential method. Consequently, often the total construction costs increase and the value of the end product decreases. Therefore, other criteria are sacrificed for that of speed. This is opposed to the CE approach where uncertainty reduction is a major feature and that improvement regarding all major objectives are pursued simultaneously. They concluded that although the two approaches are

different, several methods and techniques originating from CE have been implemented in fast tracking projects and fast tracking has thus started to integrate into concurrent engineering.

One of the techniques used in Concurrent Engineering which has attracted construction industry researchers is the Design Structure Matrix (DSM). The DSM is a square matrix of design tasks where cells indicate the dependency of one task upon the other. This technique is described in more detail in section 2.2.7. The DSM has been applied in the manufacturing industry mainly by Steward (1981, 1991), Eppinger and Eppinger et al (1990, 1991), McCord and Eppinger (1993), Pimmler and Eppinger (1994), Smith and Eppinger (1995) and Kusiak et al (1992) who used matrix manipulation algorithms to re-order the matrix in order to:

(i) Achieve optimum ordering of design tasks

(ii) Identify blocks of iterative design tasks

(iii) Maximise efficiency of design resources by designing project teams according to the requirement for each block of coupled task.

(iv) Identify tasks where project teams should be integrated

(v) Identify the design tasks which represent the 'controlling features' which account for the bulk of the time taken in the iteration process.

(vi) Decompose design problems into groups which are governed by the same set of constraints.

Recent applications of the DSM as a management tool in the construction industry research is mainly represented in the work undertaken by Huovilla et al (1995) and Newton (1995). Huovilla et al (1995) applied this technique on a case study of a building design project. They showed that the majority of the problems encountered during the design process were connected with the tasks within the iterative blocks. They envisaged that the DSM may be used in construction for planning and management of design, fast tracking analysis and visualising the effects of change initiated by the Client.

Newton (1995) applied the DSM to order design tasks during the detailed design stage of a building. This work is described in more detail in section 3.2.4.

Although techniques used in CE such as the DSM have been recently applied by construction industry researchers as a management tool, there are other CE techniques that offer potential benefits to the construction industry but have not been fully exploited. An example is the Quality Function Deployment (QFD) technique which is a method of designing and optimising the process of developing new products based on customer needs (Zairi, 1994). This technique may be applied in incorporating the elements of the client's brief of a construction project in the design of this project. One of the hurdles that hinder the full application of CE concepts to construction is that CE requires decomposing the product under consideration into components or parts and managing the engineering of these components. This could be more easily applied to a manufactured product than a construction project which is usually decomposed into disciplines (architectural, structural, services, etc.). Another hurdle is that it is possible to produce prototypes for a manufactured product (or its components), and hence optimise the process of developing the product with regards to cost, time and customer needs. This may be undertaken through assessing different alternatives and design decisions for the product under consideration. It is not yet possible to produce a prototype for a building during its design although continued advances in information technology may, through virtual reality, make this commonplace in the future.

This was one of the reasons which led the writer to explore simulation tools which allow the experimentation of different design scenarios and the assessment of different design decisions and ultimately led to the production of a simulation model for the design process. This is described in more detail in chapters 3 and 6.

2.2.7 The Design Structure Matrix (DSM)

The Design Structure Matrix has been developed initially by Steward in the early 1980s (Steward 1981) and has continued to develop by Eppinger and his research team at MIT (Eppinger and Eppinger et al 1990, 1991, 1993, Gebala and Eppinger 1991, Krishnan et al 1993, McCord and Eppinger, 1993, Pimmler and Eppinger, 1994, Smith and Eppinger, 1995).

Steward (1981) has shown that critical path schedules cannot deal with design problems, as most of the design activities are interdependent and require iteration processes. This forms the basis of the technique he has developed to improve design planning

In his technique, Steward analysed the problem by developing a precedence table showing the different design activities together with their predecessors. He then arranged these activities in a square precedence matrix with marks in the matrix showing relationships between them. This was called the Design Structure Matrix. A mark in row i column j means that i has the predecessor j. If the variables of the matrix could be re-ordered so that all marks are either on or below the diagonal (lower triangular), then variables could be determined one at a time. But since any typical engineering design contains "circuits", (or loops) it is not possible to make such ordering. However, by the "partitioning" process, variables can be re-ordered so as to confine the marks in the matrix to be either below the diagonal or within square blocks on the diagonal. Blocks should be smallest possible such that all variables occurring in a circuit will be found in the same block. Partitioning could be done manually in case of small processes, but for large processes a computer program called TERABL (recently called PSM for Windows version) has been written to do this. Figures 2.6 and 2.7 show a matrix before and after partitioning.

Since the variables within the block are interdependent, they cannot be determined one at a time without making estimates in order to break the circuit and begin the iteration. After completing the first iteration of design activities within the same block, a design review is made to determine the validity of the estimate or otherwise another iteration can be made. The marks above the diagonal show where estimates are required to start an iteration, and the objective is to obtain an ordering so that marks above the diagonal represent reasonable estimates. This could be done by "tearing" which is to choose a set of marks representing where estimates might be made so that if they were removed from the block and the variables in the block were re-ordered by partitioning, no marks would appear above the diagonal and hence no additional estimates are to be made.

Tearing can be done by assigning levels to marks that are required to be torn. Marks with higher levels (where good estimates could be made or where poor estimates are not sensitive) are to be torn first, then the matrix is re-ordered by partitioning. If further estimates are required to break all circuits, then the next higher level numbers are to be torn. This could be done by the computer, and shunt diagrams may be used to decide which variable is to be torn in order to break the circuit.

After partitioning and tearing a final matrix is achieved that may be used as a basis for planning the engineering work. Marks above the diagonal show where estimates must be made and which variables are required to proceed with the determination of each variable. Design reviews should be undertaken at the end of each block to see whether design confirms these estimates or not.

PSM Version b.30	1 :	2 :	3 :	4 :	5 :	6 :	7 :	8 :	9 :	10	11	12	13	14	15	16
1 : Passenger capacity spec.	1 :															
2 : Size-aeordynamics	X	2 :	X				X				X	X				
3 : Motor spec & wt.		X	3 :	X		X	X				X					
4 : Total weight	X	X	X	4 :			X				X	X				
5 : Stored energy requirement			X		5 :	X		X	X	X			X			
6 : Battery type - energy density						6 :										
7 : Battery size & weight					X	X	7 :									
8 : Cursing speed spec.								8 :								
9 : Speed & accel. vrs power	X	X		X					9 :			X				
10 : Acceleration spec.										10						
11 : Speed & acceleration								X	X	X	11					
12 : Structure & suspension	X	X	X	X			X				X	12				
13 : Range spec.													13			
14 : Cost--		X	X	X		X	X					X		14		
15 : Demand vrs. cost	X	X						X		X		X	X		15	
16 : Profit														X	X	16

Figure 2.6 Design Structure Matrix for the design of an electric car before partitioning (Source: Steward's PSM software)

PSM Version b.30	13	10	8 :	6 :	1 :	2 :	3 :	4 :	5 :	7 :	9 :	11	12	15	14	16
13 : Range spec.	13															
10 : Acceleration spec.		10														
8 : Cursing speed spec.			8 :													
6 : Battery type - energy density				6 :												
1 : Passenger capacity spec.					1 :											
2 : Size-aeordynamics					X	2 :	X			X		X	X			
3 : Motor spec & wt.				X		X	3 :	X		X		X				
4 : Total weight					X	X	X	4 :		X		X	X			
5 : Stored energy requirement	X	X	X	X			X		5 :		X					
7 : Battery size & weight				X					X	7 :						
9 : Speed & accel. vrs power					X	X		X			9 :		X			
11 : Speed & acceleration		X	X								X	11				
12 : Structure & suspension					X	X	X	X		X		X	12			
15 : Demand vrs. cost	X	X	X		X	X							X	15		
14 : Cost--				X		X	X	X		X			X		14	
16 : Profit														X	X	16

Figure 2.7 Design Structure Matrix for the design of an electric car after partitioning (Source: Steward's PSM software)

Once estimates are made of how many blocks are to be iterated and how long tasks are to take in each iteration, a critical path schedule can be developed. Hence, the Design Structure System does not replace critical path but provides a preliminary analysis before developing a critical path. It highlights which variables affect each other.

A computer program called Analysis Of Structure And Propagation Of Engineering Consequences Throughout (ASPECT) has been written to help trace these effects, retrieve names of responsible engineers for these variables, documents in which the variable is specified, the estimate of task durations and then the information is used to develop a schedule for implementing the change. (Steward 1981)

Steward's Design Structure Matrix provides a powerful tool to achieve the optimum order of design tasks, identify iterative design loops, and plan design based on required number of iterations. This technique has been used by the writer whilst investigating the different approaches to simulating the design process. Further details are provided in chapters 3 and 4.

Eppinger (1991) developed Steward's technique of the Design Structure Matrix with the objective of using it as a modelling tool for managing concurrent engineering for design and manufacture.

He classified the relationships between any tasks in the design process into three possible models:

(i) Dependent tasks (Series)
(ii) Independent tasks (Parallel)
(iii) Interdependent tasks (Coupled)

Managing the first two types is relatively straightforward. The management of the third type (the coupled tasks) require more design time and many iterations of information transfer. In fact, the coupled tasks model is more realistic for simultaneous engineering where the information transfer is essential and iteration is typical.

Considering Steward's Design Structure Matrix after partitioning into blocks, the design tasks could be classified into the above-mentioned three types (i.e. series, parallel, and coupled). If one task is dependent on another, then they are in series while if one task is independent on the other, then they can be carried out in parallel. Tasks in blocks will be treated as coupled tasks which must be solved simultaneously and require iteration. This is illustrated in Figure 2.8. Eppinger (1991) suggested different strategies to analyse the partitioned matrix in order to obtain a lower triangular matrix.

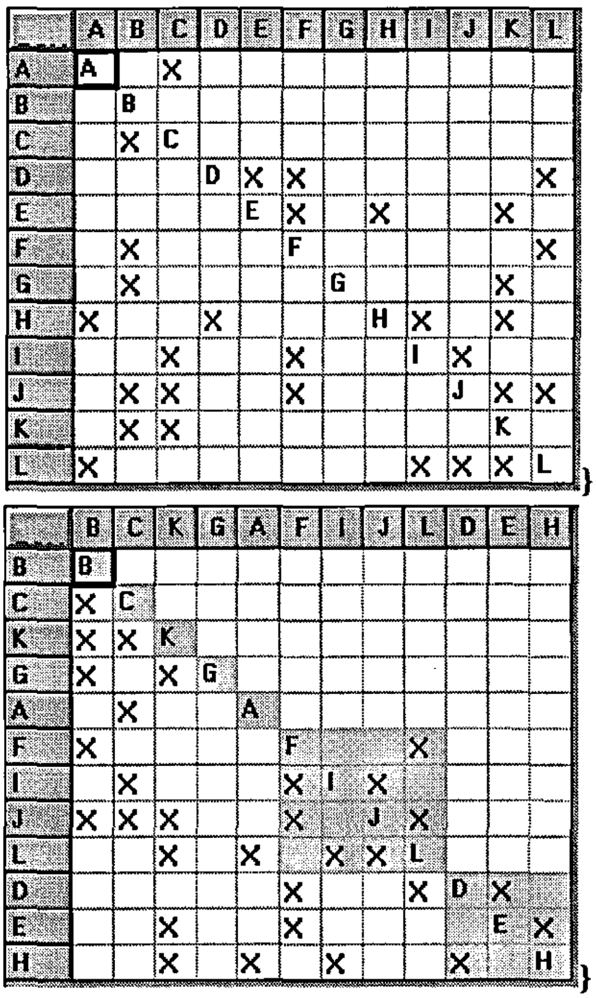


Figure 2.8 Example of a Design Structure Matrix adapted from Eppinger using Steward's TERABL program

In further research, Smith and Eppinger (1995) introduced to the Design Structure Matrix numerical measures to reflect the degree of inter-dependence between tasks. Tasks in the matrix are re-arranged to minimise the importance of the elements above the diagonal. The off diagonal values represent the strength of dependence of the task on each of the other tasks. Eppinger identified two approaches to determine the strength of dependence. In the first approach the numerical value indicates the probability that one additional iteration will be necessary if the interdependent tasks are performed in the specified order. Each of the dependencies is assigned one such probability and all potential orderings of the interdependent tasks are investigated in order to identify the ordering which minimises the probability of many iterations. The second approach does not rely on a stochastic description of the design process. The numerical value is a measure of the portion of information produced during the first iteration which will need to be changed during the second iteration. Mathematical models were developed to identify the key tasks that influence iteration. However, these mathematical models are company specific (since they have been developed specifically for a major car manufacturer) and hence could not be generalised.

Generalised models of design iteration were produced by Nukala, Eppinger and Whitney (1994) using signal flow graphs techniques. This method is more suitable for modelling design iteration of highly repetitive manufactured products in factories where conditions of manufacturing are unchangeable and hence is not suitable to apply for modelling building design iteration.

For further applications of the DSM, McCord and Eppinger (1993) used the DSM to design project teams and identify when co-ordination between and integration of these teams is most essential. The DSM is re-configured to identify blocks of coupled tasks and a separate team is assigned to carry out the design tasks for this block. Each block represents the tasks involved in the design of a component of the product under consideration. For example if the product under consideration is a 'Computer', then the blocks would represent the tasks involved in the design of the drive system, main board, screen and packaging. They suggested that overlapping of design teams (i.e. overlapping blocks of coupled design tasks) will maximise such integration.

Although the above mentioned approach provides a comprehensive analysis to the Design Structure Matrix (DSM) based on dependencies between different design tasks, it is mainly oriented towards design for manufacture. It does not show details of the dependencies or how those dependencies have been identified. Therefore, in the work undertaken by the writer, it was decided to use the DSM based on information dependencies between different design tasks. However, there was a need first to identify these design tasks and their information requirements. This is explained in more detail in chapter 4.

A similar technique to the DSM has been developed by Rogers (1989). In his research, Rogers (1989) developed a computer-based tool including a knowledge-based system for the multi-level decomposition of design tasks with the minimum feedback loops. The modules of a design system are partitioned into circuits which represent sub-systems where each module is simultaneously dependent on all the other modules within the same circuit. Within each circuit, there exists feedback links representing iterations, while between the circuits there are feedforward links indicating that there is no iterations among them. In this case, circuits can be ordered in a multi-level format.

The user divides the design system into elements and defines the relationship among these elements. The different types of elements are Design Variables (DV), Constraint Functions (G), Behaviour Variables (BV) and Objective Function (OB). The relationships are of the type:

$$DV = f(Gi)$$
$$G = f(DVi)$$
$$OB = f(DVi)$$
$$BV = f(DVi)$$

The module or task is the one that represents the function (f) and each module has got its input and output. For example, if DV1 = f(G1, G2), then this module has got the output of G1 and G2 and an input of DV1. The user also assigns a weight and duration for each module.

Planning and ordering of tasks is carried out using a knowledge base shell (CLIPS) taking its facts from the input data previously mentioned. This is the main difference between the work of Rogers (1989) and Steward's previous work. While Steward implements the grouping of tasks into circuits with matrix manipulations, work by Rogers follows the same steps but replaces matrix manipulation for grouping by applying rules contained in a knowledge base. This procedure is more flexible and allows new rules to be added. The program checks the output of each module against the input requirements of other modules. If the output of the module is contained in the input list of at least one other module, then that module contributes to the solution of the problem. The user has the choice to order modules according to input/output requirements or by parallel requirements according to the rules given to the knowledge base. Scheduling according to input/output requirements will re-order modules and circuits based on their couplings. Ordering within a circuit is done based on the weight assigned to the modules giving priority to highest weights.

The results of ordering modules are shown in an NxN matrix format. This is illustrated in Figure 2.9 which shows a simple design problem comprising eleven modules after being ordered. Three circuits (a,c,b), (h,f,e,g), and (j,i) are identified.

Another main variation with Steward is that while Steward shows ordering of tasks in a partitioned square matrix having the tasks shown in rows and columns, Roger's work displays modules on the diagonal of a square matrix. A horizontal line from a module indicates an output and a vertical line to a module indicates an input. Contrary to Steward, an intersection under the diagonal represents a feedback, while that above the diagonal represents a feedforward. This is illustrated in Figure 2.9. After ordering and grouping of modules, a multi-level decomposition of tasks can be displayed without any feedback links among the circuits. The only iterations are contained within the circuits. Times of executing circuits can be calculated as well in case of sequential or parallel execution.

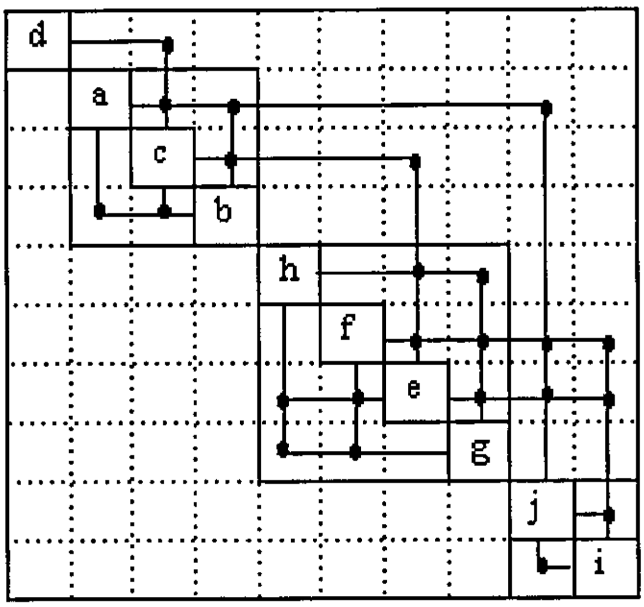


Figure 2.9 An illustration for NxN display of modules, circuits and links after scheduling using Roger's approach

The program can also display a dependency rectangular matrix showing relationships between constraints and independent design variables. Building the dependency matrix after the planning and scheduling functions reveals dependency patterns that may prove advantageous when developing multilevel optimisation algorithms.

Rogers' work was based on previous work undertaken by Sobieski (1982) for NASA for the optimisation of design problems. Sobieski decomposed a main design problem into sub-problems and each sub-problem is managed by minimising its constraints violations. A linear extrapolation for each sub-problem towards the main problem is formed and the system is optimised for its objective function and constraints. Sobieski did not show how to decompose a design system into sub-systems. This was the area Rogers developed with the objective of multi-level decomposition for the design tasks.

The approach adopted by Rogers concentrates on technical aspects of design problems and sub-problems rather than managerial aspects with the objective of grouping design tasks on the organisational level. However, it was considered as one of the options for the simulation tools to be used by the writer in this research. These options are described in chapter 3.

2.3 SUMMARY AND CONCLUSIONS

This chapter has reviewed relevant literature and research on the design process with regard to the nature of design, different design process stages, the management of the process and the problems encountered, and the current practice and techniques used in planning and managing design. The review undertaken by the writer has not been limited to the building design process, but also encompassed the manufacturing design process. This reflects the recent trend towards considering construction as a manufacturing process.

There is no consensus among researchers and practitioners with respect to the different stages of the design process. (The RIBA plan of work will be used within this research as a guideline for the different design stages due to its popularity and the familiarity of most of the construction industry professionals with its different stages.)

Design is, by its nature, an iterative process. This iterative nature makes it complex and difficult to manage. Current planning techniques such as network analysis and PERT are suitable for planning deterministic activities which are either sequential or parallel such as construction activities. They are ill-suited to plan activities with an iterative nature such as the design activities because they neither allow feedback loops nor any iterative procedures.

Few attempts have been made to apply concurrent engineering techniques as used in the manufacturing industry to the construction industry. One potential technique for design management considered appropriate to construction is the use of matrix analysis to achieve the optimum order for design tasks and highlight which tasks should be carried out in an iterative fashion. These techniques have been used by the writer to identify loops of iterative design tasks while developing the Simulation Model described in part III of this thesis.

The design process is information driven. The main difficulties encountered during the management of the design process are predominantly information related. Information transfer and communication issues have been identified by different researchers as the key factors to the successful management of the design process. However, the literature survey undertaken by the writer showed little research work focused on managing the information exchanged during the earlier stages of design prior to the production of contract documents. For this reason, this research focused on the Conceptual and Schematic stages of design.

A review of the current practice for design management undertaken by the writer showed that in complex multi-disciplinary design situations, design managers lack suitable tools to aid them in managing the process. These tools must aid design managers in planning design, taking into consideration its iterative nature, and foreseeing the effects of changing different parameters that affect information transfer and communications during the design process. However, there is a need first to identify the information flows exchanged during the design process. It is the need for increasing sophistication in design management tools that formed the driving force for the development of models for the information flow and the application of simulation techniques.

CHAPTER 3

MODELLING THE DESIGN PROCESS

3.1 EXISTING MODELS OF THE DESIGN PROCESS

Evans et al (1982) provide a summary of the efforts made by some design professionals to understand and model the processes behind the design disciplines. Until the mid 1950's, designers tended to focus solely on creating good designs and not on understanding the processes behind design to make them more efficient and effective. The process of design was considered intuitive and there was apparently no need to understand "how designers designed". It was a common idea that design was a trait which a person may or may not have, (Evans et al 1982, Venegas 1987). This view changed in the late 1950's. There was an increased pressure to design more efficiently as industries developed quickly. It was then that experts from other disciplines, such as operations research and ergonomics, began to study and model design as a process in order to improve its efficiency.

Many attempts have been made to model the design process. Some of these models simply describe the sequences of activities that typically occur in designing, other methods attempt to prescribe a better or more appropriate pattern of activities. Both these types of models are described below.

3.1.1 Descriptive Models

Descriptive models of the design process usually emphasise the importance of generating a solution concept early in the process thus reflecting the 'solution focused' nature of design thinking. The initial solution 'conjecture' is then subjected to analysis, evaluation, refinement and development. Sometimes the analysis and evaluation show up fundamental flaws in the initial conjecture and it has to be abandoned, a new concept generated and the cycle started again.

An example of descriptive model is French's model (French 1985) of the design process shown in Figure 3.1. The circles represent stages reached, or outputs, and the rectangles represent activities, or work in progress.

According to French, the process begins with an initial statement of a 'need'. The design activities that follow are:

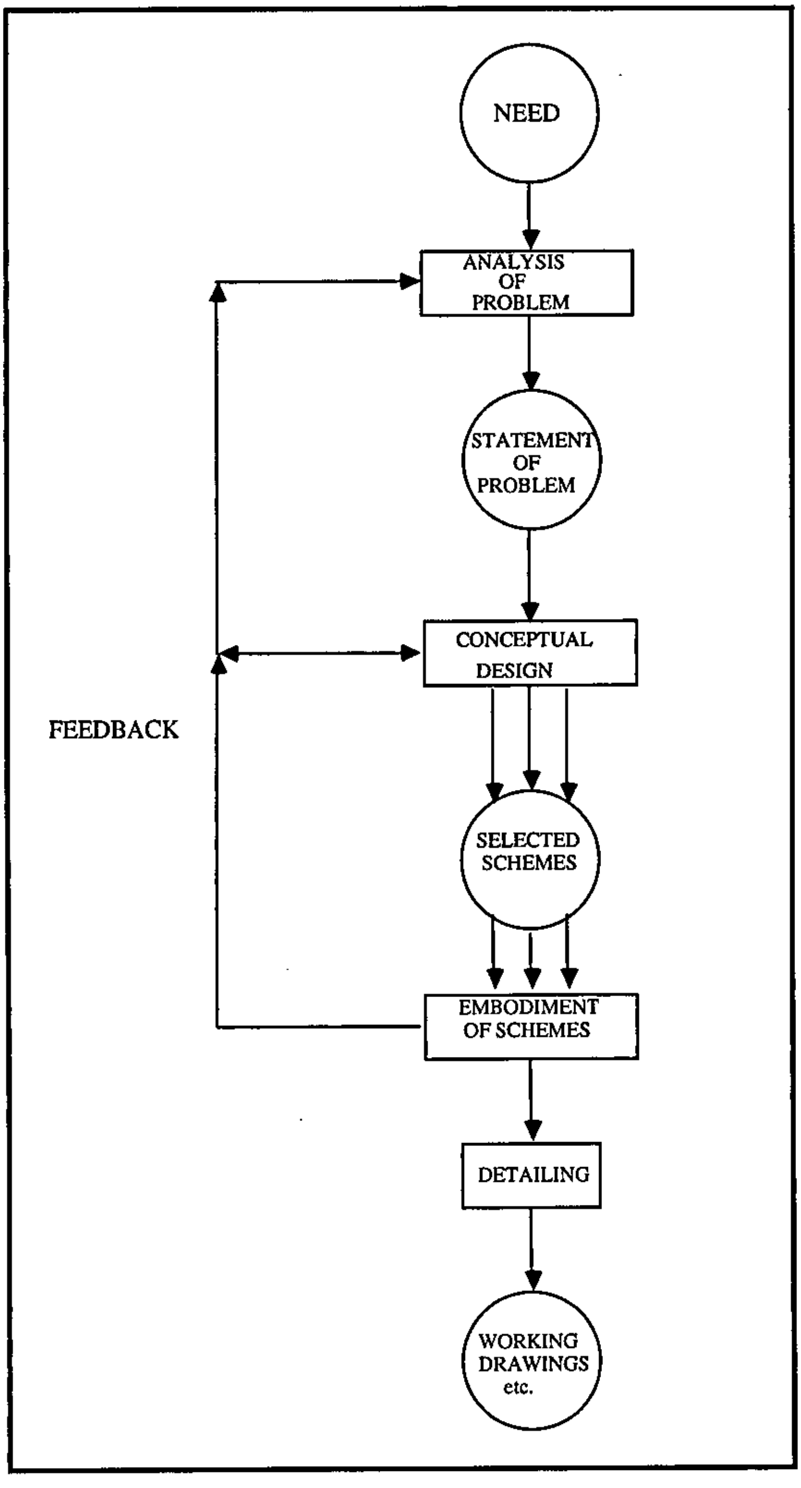


Figure 3.1 French's Model of the Design Process

(i) Analysis of the problem.

(ii) Conceptual design: In this phase, broad solutions to the design problem are generated. It is the phase where engineering science, practical knowledge, production methods and commercial aspects need to be brought together and where the most important decisions are taken.

(iii) Embodiment of schemes: In this phase the schemes are 'worked up' in greater detail and a final choice is made between different alternative schemes. The end product is usually a set of general arrangement drawings. There are feedback loops from this phase to the conceptual design stage.

(iv) Detailing: This is the last phase in which a very large number of small but essential points remains to be decided. The work should be of very high quality, otherwise delay and expense will be incurred.

These activities are typical of conventional engineering design. However, French assumes that after the 'embodiment of scheme' stage, the design is frozen and there is no feedback loops at the detailing stage. This may not always be the case in practice.

3.1.2 Prescriptive Models

Prescriptive models are concerned with trying to persuade or encourage designers to adopt improved ways of working (Cross 1991). They usually offer a more algorithmic systematic procedure to follow, and are often regarded as providing a particular design methodology. Examples of prescriptive models are Pahl and Beitz's model (Pahl and Beitz 1988) shown in Figure 3.2 and the VDI model (Verein Deutscher Ingenieure) produced in Germany (Cross 1991) shown in Figure 3.3.

Pahl and Beitz also based their model on four stages of design as French. For each stage, they identified a prescriptive list of task that have to be undertaken during that particular stage as shown in Figure 3.2.

Although Pahl and Beitz's model shows the iterative feedback loops between all the design stages and provides a check list of design tasks to be carried out during each stage, those tasks are broad and general. The model does not show also the information requirements which are necessary to perform a certain task or proceed to the next design stage.

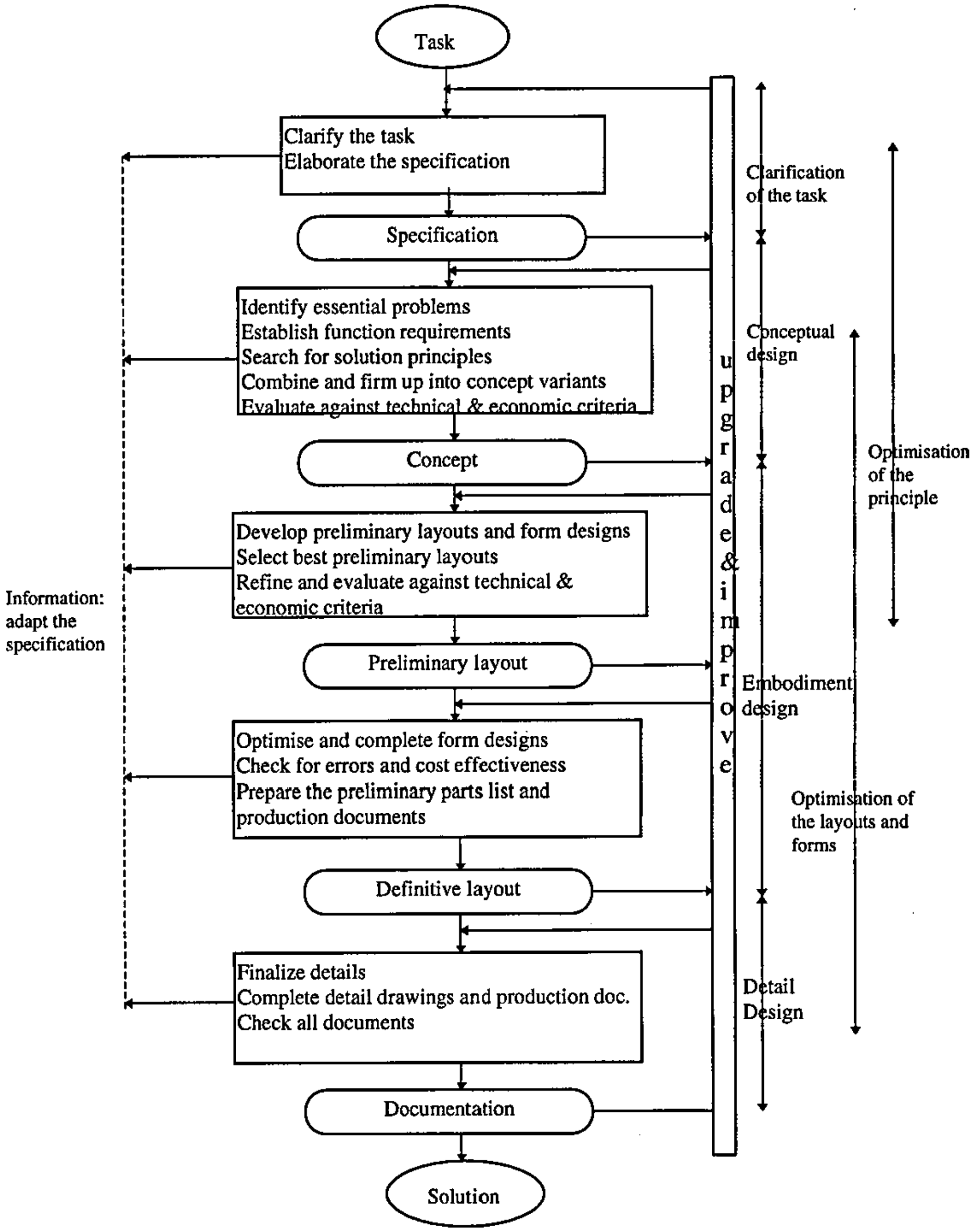


Figure 3.2 Pahl and Beitz's model of the design process (adapted from Cross 1991)

The VDI model (Cross 1991) shown in Figure 3.3 suggests a systematic approach in which 'The design process, as part of product creation, is subdivided into general working stages making the design approach transparent, rational and independent of a specific branch of industry'. The structure of this general approach is based on seven stages each with a particular output. In the VDI Guideline, it is emphasised that several solution variants should be analysed and evaluated at each sage and that there is more detail in each stage than is shown in the diagram (Cross 1991). There is also a warning that the stages shown in the approach do not necessarily follow rigidly one after the other. They are often carried out iteratively, returning to preceding ones, thus achieving a step-by-step optimisation.

The VDI Guideline follows a general systematic procedure of first analysing and understanding the problem as much as possible, then breaking this into sub-problems finding suitable sub-solutions and combining those into an overall solution. This kind of approach has been criticised because it is based on a problem focused rather than a solution focused approach. It therefore runs counter to the designer's traditional ways of thinking (Cross 1991).

In addition, the writer believes the model has failings with regard to its final stages which jump from 'complete overall layout' producing 'definitive layout' to 'prepare production and operating instructions' producing the 'product documents' without showing any intermediate stages. For this reason, the writer argues that this model may not be applied to the construction industry although the model Guideline claims that it is independent of a specific branch of industry.

The most commonly recognised and accepted prescriptive model for a building project in the UK is the RIBA plan of work (RIBA, 1973). It is a framework of stages describing all the design work and management tasks in a project programme from inception to completion. For each stage, the plan defines the purpose of work and decisions to be reached, the tasks to be done, the people directly involved and the different functions of these personnel. The three main stages of design formalised by the RIBA and their objectives are summarised in Table 3.1.

Although providing details of each stage of design, the RIBA plan of work does not show their information requirements. This fact is highlighted by Lawson (1980) who maintains that while the RIBA plan of work is a useful design management tool, it merely defines what is to be done rather than how. In an attempt to incorporate information flow for the different design stages outlined in the RIBA plan of work, Jergeas (1989) developed a design process model in the form of a lengthy flow chart. He emphasised the importance of incorporating 'constructability' principles to reduce construction costs. However, the information flow shown in his model was in a global form for the whole design stage. Information requirements for <u>each</u> design task were not shown and the flow chart does not reflect the iterative nature of design.

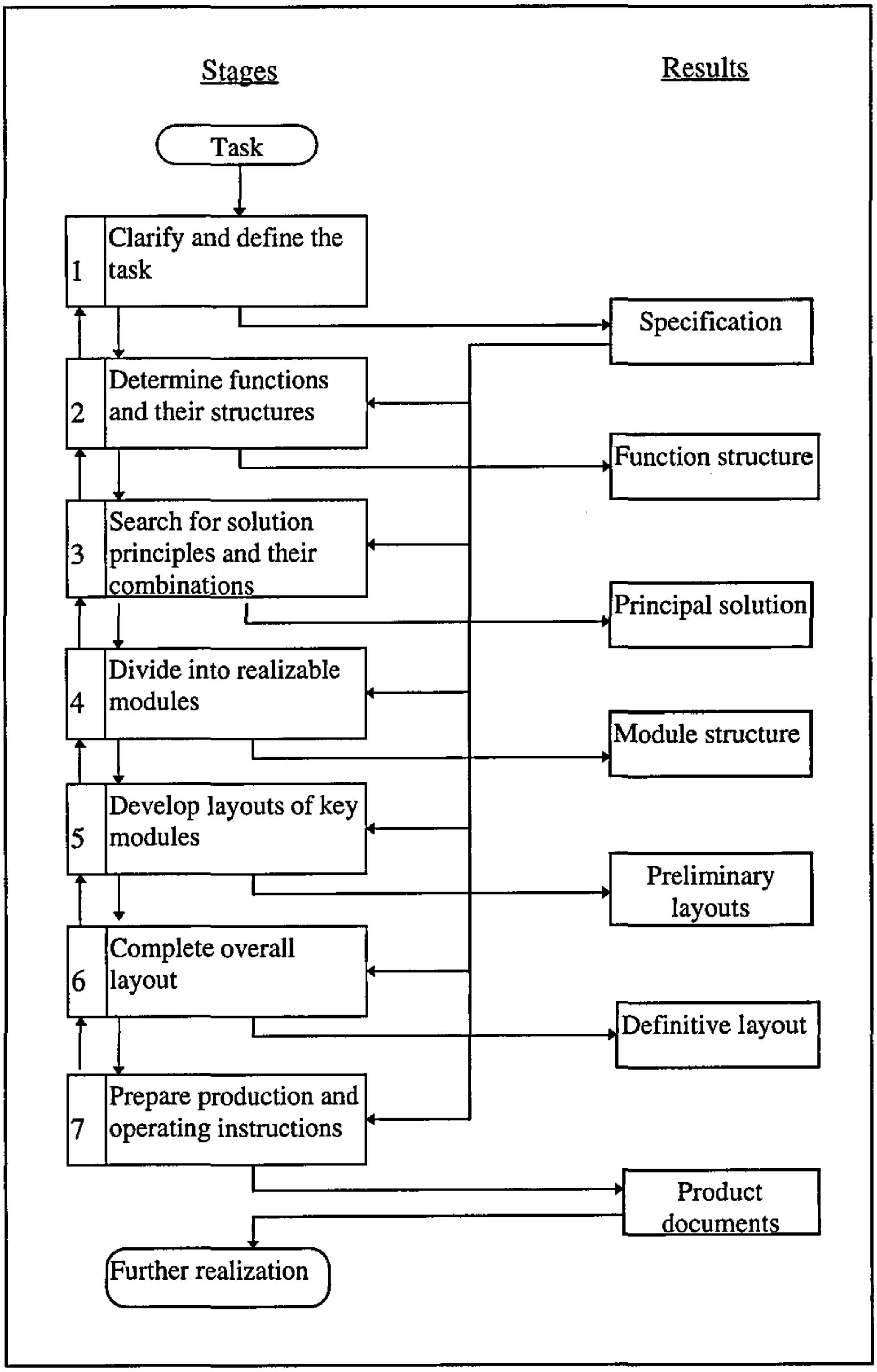


Figure 3.3 The VDI 2221 model of the design process (adapted from Cross 1989)

Stage	Purpose of work and Decisions to be reached
C: Outline Proposals	To determine general approach to layout, design and construction in order to obtain authoritative approval of the Client on the outline proposals and accompanying report
D: Scheme Design	To complete the brief and decide on particular proposals including planning arrangement appearance, constructional method, outline specification and cost and to obtain all approvals
E: Detail Design	To obtain final decision on every matter related to design, specification, construction and cost

Table 3.1 The three main stages of design as formalised by the RIBA plan of work

3.1.3 Other Models

Frost (1992) developed a converging two branched model for the early innovative part of the design process. In this model, problem nodes are identified in one branch and potential solution nodes were identified in the other. It is at the convergence of these two branches that synthesis occurs.

Dias and Blockley (1994) integrated product and process models for design through the definition of generic units called 'roles'. A process model role has been defined by Platt and Blockley (1993) as a collection of responsibilities. In order to exercise those responsibilities, objectives will be negotiated between roles which are then translated into tasks to fulfil functions. A product model role is simply a collection of functions. This is because product models only describe artefacts with action-reaction capabilities while process models describe human activity with action, reaction and intentionality. Therefore a product role is a special restricted case of a process role. The generic nature of the roles should ensure sufficient generality to support any type of subsequent decomposition. Relationships between entities should be declared in reciprocal fashion. Two of the most useful relationship types identified by Dias and Blockley were generalisation/specialisation and aggregation/decomposition. These principles were applied to product and process models for design. In a product model, the building role could be decomposed into system and subsystem roles, which in turn are aggregation of element roles. There could be different systems such as architectural, structural and services, each of which contain different roles at different hierarchical level. Relationship between roles may belong to different aggregation hierarchies and some roles could be shared by different systems (Dias 1993). In a process model, the project managerial role can be recursively decomposed into managerial, professional and technical roles. Individual roles can be aggregated into groups and subgroups. For the design process, these groups would correspond to the systems in a product. These include architectural, structural and services groups. The collection of all the group roles constitutes the project team.

This approach described by Dias and Blockley is an attempt to combine a product model of a building with a process model for design. They have presented the main modelling concepts without showing details of the model itself. Although they described two approaches for implementing the model which rely mainly on object oriented programming, the writer believes that such a model is more theoretical than practical and would be difficult to implement or apply in the real design world.

Powell and Newland (1993) introduced human and psychological factors in modelling the behaviour of designers during information exchange at the design process. They showed that when an information system is successfully matched to peoples' preferred ways of learning, it results in better communication and deeper learning. They based their research on models developed by Pepper (1942) and Kolb (1976) to model peoples' world views.

This review of design process models shows that these models either describe (or prescribe) the process in terms of its different stages or in terms of the thinking ways and behaviour of designers. The models of French and Pahl and Beitz concentrate on classifying the different stages of design as conceptual, embodiment and detail design showing the iterative loops between these stage while the VDI model suggests seven problem focused systematic stages with guidelines emphasising that these stages are often carried out iteratively. Therefore the main commonalty amongst these models is demonstrating the iterative nature of design.

The models of Frost, Powell and Newland, concentrate on human and behavioural aspects of designers. They address the different thinking ways of designers and the factors that influence it. However, these models describe only different designers' attitude towards learning design problems and exchanging information with other design team members, but it cannot prescribe or impose a specific thinking method for designers. The model of Dias and Blockley tries to integrate the functional roles of a building with the responsibility roles of a process model for design. They have also addressed the behavioural issues of designers in showing that the complex interactions between individual responsibility roles within a group of designers can give different patterns according to the behaviour of individuals within the groups which is difficult to predict in practice.

The above mentioned modelling examples show that although there has been many attempts to model the design process, there is still no consensus among authors which reflect the complex nature of the design process. Although these examples show that design has been analysed at a tactical level, they do not address the information flow issues. It was not until the late 1980's when structured analysis diagramming techniques developed for systems analysis purposes were used to model both design and construction processes and to show the information exchange within these processes. These techniques offer new opportunities for modelling the design process. This approach is explained in more detail in the following section.

3.2 THE USE OF STRUCTURED ANALYSIS DIAGRAMMING TECHNIQUES TO MODEL THE DESIGN PROCESS

3.2.1 Introduction

Structured techniques evolved from a coding methodology (structured programming) to techniques including analysis, design and testing methodologies as well as project management concepts and documentation tools (Martin and McClure 1985). Structured techniques were introduced as a step towards changing software building to be an engineering discipline which could be automated . They were introduced by the academic community in the late 1960's and became popular in industry in the early 1970's after being adopted by IBM on major projects. By the late 1970's, structured techniques had evolved into a set of techniques to include the whole software life cycle addressing both technical and management issues. Experience in the development of systems had shown that analysis is a critical step in developing software systems and programs because it affects all the development steps that follow. Analysis is not an easy task (Marca and McGowan 1988, Martin and McClure 1985). It is difficult because of communication problems, changing system requirements and inadequate estimating techniques. Structured analysis proposes to solve these difficulties by providing a systematic approach to performing analysis and by producing a new improved system specification and hence it concentrates on clear, concise communication (Martin and McClure, 1985). Structured analysis is based on the following concepts:

- Top down hierarchical organisation
- Divide and conquer: it is the concept of solving different problems by dividing a problem into a set of smaller independent problems that are easier to understand and solve. It is a powerful and essential tool in dealing with complexity.
- Graphical communication and documentation tools.

Demarco (1978) explained that the major difference between classical analysis and structured analysis is a new system specification that is much more rigorous and much more user-friendly than the gigantic, impossible to read, narrative specification produced by classical analysis methods. Several different categories of structured analysis diagramming techniques exist. These are now reviewed.

3.2.2 Categories of Structured Analysis Diagramming Techniques

Martin and McClure (1985) identified the use of structured diagrams in four main areas:

(i) Overview systems analysis: An overall model of an organisation and its systems may be drawn. Processes are decomposed hierarchically and overall flow of data and processes are modelled.

(ii) Program architecture: The overall architecture of a program or set of programs is drawn showing the separate modules.

(iii) Program detail: The detailed logic within one program module is drawn.

(iv) Data structure: An overall structure of the data is drawn. Database models and file representation are drawn using different diagramming techniques.

As the scope of this research was to use structured analysis diagramming techniques to model the construction design process, and not software development, only the techniques applied for overview system analysis were considered. These techniques are examined in the following sections.

3.2.2.1 <u>Data Flow Diagrams (DFD)</u>

Demarco (1978) defined a data flow diagram as a network representation of a system. The system may be automated, manual or mixed. The data flow diagram portrays the system in terms of its component pieces with all interfaces among the components indicated. The most significant characteristics of DFDs are that they are graphical, partitioned, multidimensional, emphasise flow of data and de-emphasise flow of control.

There are two similar versions of data flow diagrams: Gane and Sarson; and Yourdon and Demarco. The main difference between both versions lies in the symbolic diagramming conventions, as Gane and Sarson's version adopts more sophisticated symbols to build a data flow diagram which are mainly oriented towards software building. An example for this sophistication is having different symbols for flow of materials and flow of data because it is important for software building to distinguish between computer data and non computer data. Since the scope of this research is to apply structured analysis diagramming techniques to the design process and not to software building, only Demarco/Yourdon's version will be considered.

Elements of a Data Flow Diagram

The elements of a data flow diagram as specified by Demarco are shown in Figure 3.4

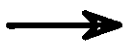
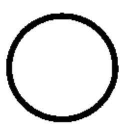
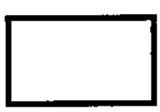
	DATA FLOW	A pipeline through which packets of information of known composition flow.
	DATA STORE	A temporary repository of data (also known as files)
	PROCESS	A transformation of incoming data flow(s) into outgoing data flow(s)
	SOURCE OR SINK	A person or organisation lying outside the context of a system that is a net originator or receiver of system data

Figure 3.4 Elements of a Data Flow Diagram

A simple example of a data flow diagram is shown in Figure 3.5.

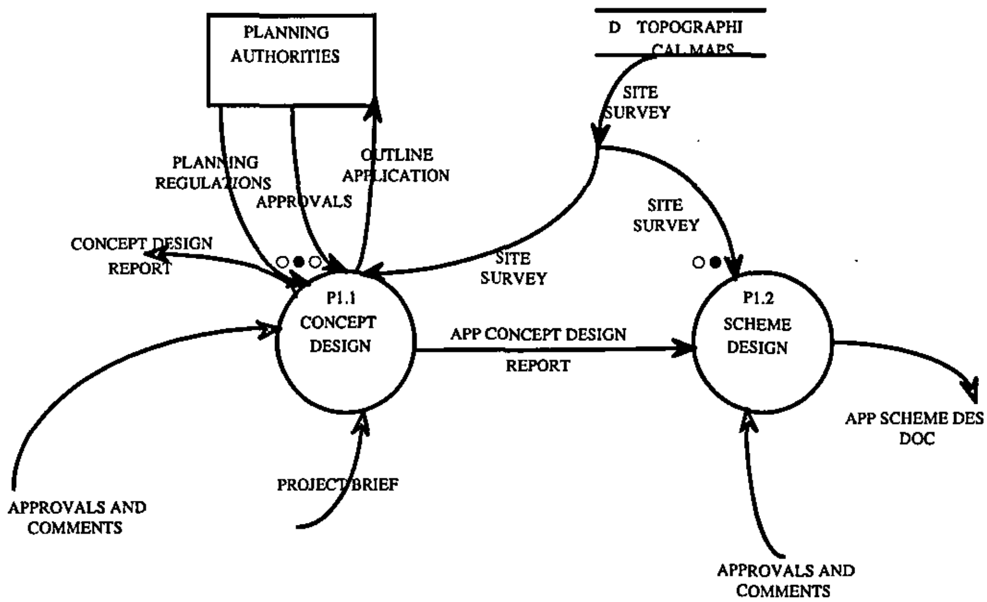


Figure 3.5 An example of a data flow diagram

Levelled Data Flow Diagrams

A DFD is a tool for top down analysis. When a system is too large for its DFD to be shown on a single diagram, the system is partitioned into sub-systems. The top level of a levelled set of DFDs is called the Context Diagram and the bottom level is composed of a set of unpartitioned processes called the functional primitives. When a process is further decomposed into lower levelled processes, the main process is called the 'parent' and each of the decomposed processes called a 'child'. All data flows shown entering a child diagram must be represented on the parent by the same data flow into the associated bubble. Outputs from the child diagram must be the same as outputs from the associated parent bubble. This is known as balancing of DFDs.

Process Specification and Data Dictionary

When a DFD is produced during structured analysis, a process specification and a data dictionary are also produced to give additional system information. A process specification is developed for every functional primitive in the lowest level DFD. It defines how data flows in and out of the process and the transformations the data undergo. The data dictionary contains definitions of all data in the DFD. Data flows and data stores are described in terms of their constituent data elements. The data dictionary can also include physical information about the data such as data storage devices and data access methods.

Data Flow Diagrams by Ward and Mellor

Ward and Mellor (1985) introduced some refinements to Demarco's methodology for data flow diagrams. These were mainly oriented towards structured developments for real-time systems. The main refinements were dividing data flows into time continuous and time discrete, introducing control flows and control processes and defining the behaviour of a control process using state transition diagrams. An example of a state transition diagram is illustrated in Figure 3.6.

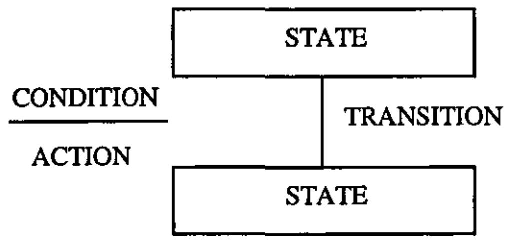


Figure 3.6 Example of a state transition diagram

The refinements introduced by Ward and Mellor are mainly suitable for application in real time systems. However, modelling other processes, like the design process, can benefit from some of these refinements such as the 'control flows' which can represent the 'approvals and comments' of the design activities.

3.2.2.2 Functional Decomposition

Functional decomposition is used in most structured design and analysis. A high level function is decomposed into a tree structure of lower level function. Functional decomposition can be applied to structures of organisations, programs, files and reports. It applies to functions rather than data, however, similar diagrams are sometimes drawn for the decomposition of both data and functions. An example of functional decomposition is shown in Figure 3.7.

3.2.2.3 Structure Charts

A structure chart is a form of functional decomposition. Martin and McClure define a structure chart as "a tree or hierarchical diagram that defines the overall architecture of a program by showing the program modules and their interrelationships". Along with DFDs, structure charts constitute a very common structured design methodology.

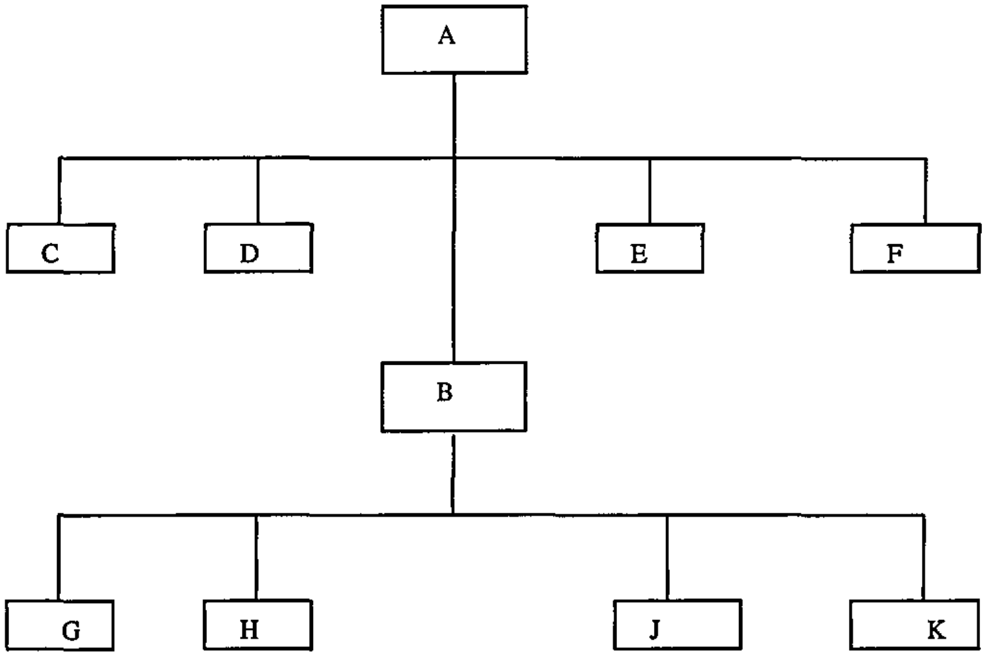
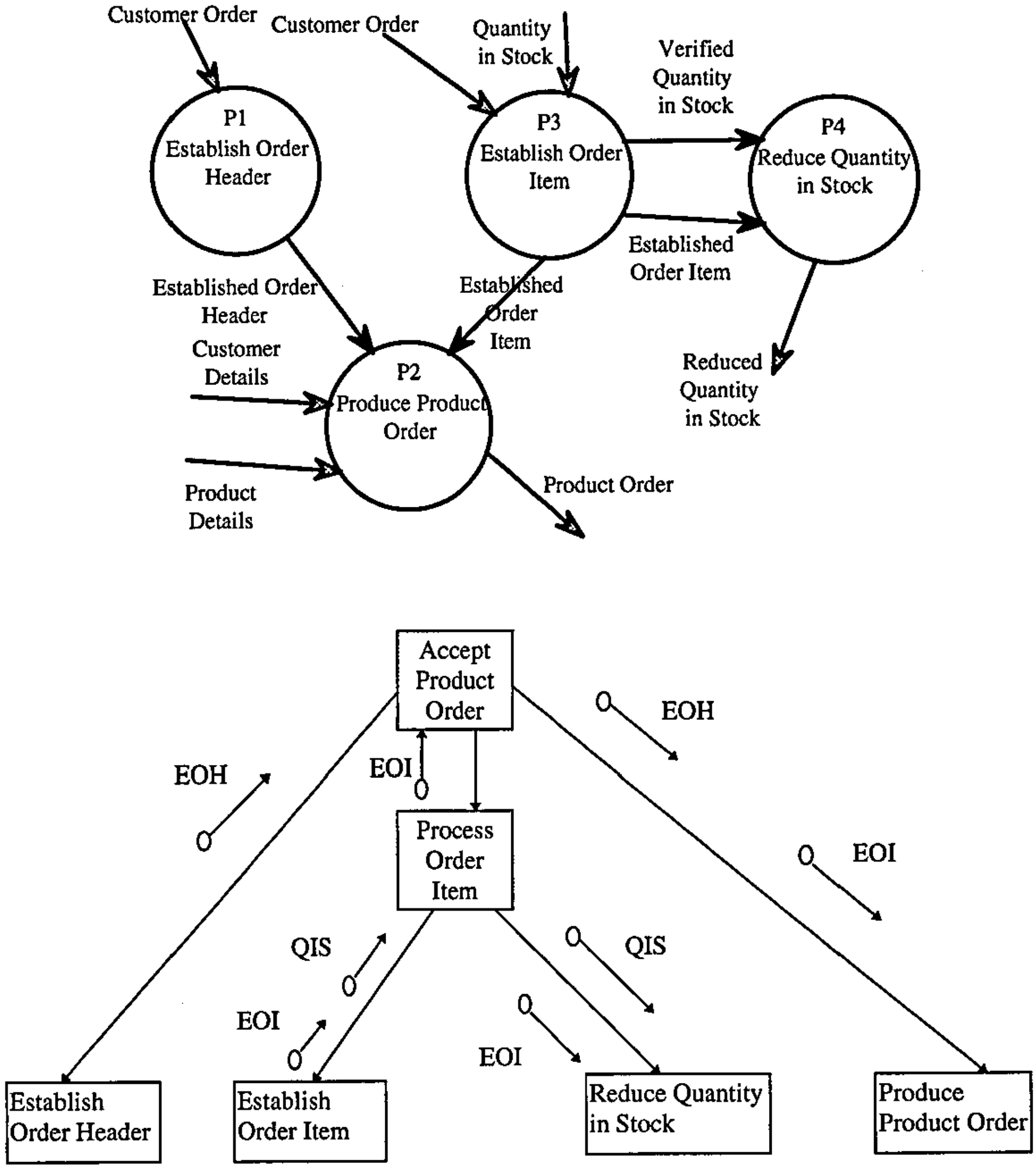


Figure 3.7 Example of Functional Decomposition

An example of a structure chart is shown in Figure 3.8.
However, structure charts do not describe the input and output data for each process. They are more suitable for showing hierarchy of program modules. They become complicated when data and control variables are written on them.

3.2.2.4 HIPO Diagrams (Hierarchical Input, Process Output)

A HIPO diagram is a diagramming technique using a set of diagrams to show the input, output and functions of a system or program. Like a structure chart, they show what a system does rather than how. There are three basic types of HIPO diagrams: visual table of contents, overview diagrams and detail diagrams. The purpose of the visual table of contents is to show the overall functional components of a system or program. Overview and detail HIPO diagrams consist of three parts: an input box, a process and an output box. They are similar to a data flow diagram in that they show the flow of data through processes. However they are more difficult to draw than data flow diagrams and they are limited to defining procedural components. They are more suitable for small systems as they become difficult to read when there are several process steps or input/output data items to show. An example of HIPO diagrams is shown in Figures 3.9 and 3.10.



Key: EOH = Established Order Header
EOI = Established Order Item
QIS = Quantity in Stock

Figure 3.8 A Data Flow Diagram converted into a Structure Chart (adapted from Gharib 1991)

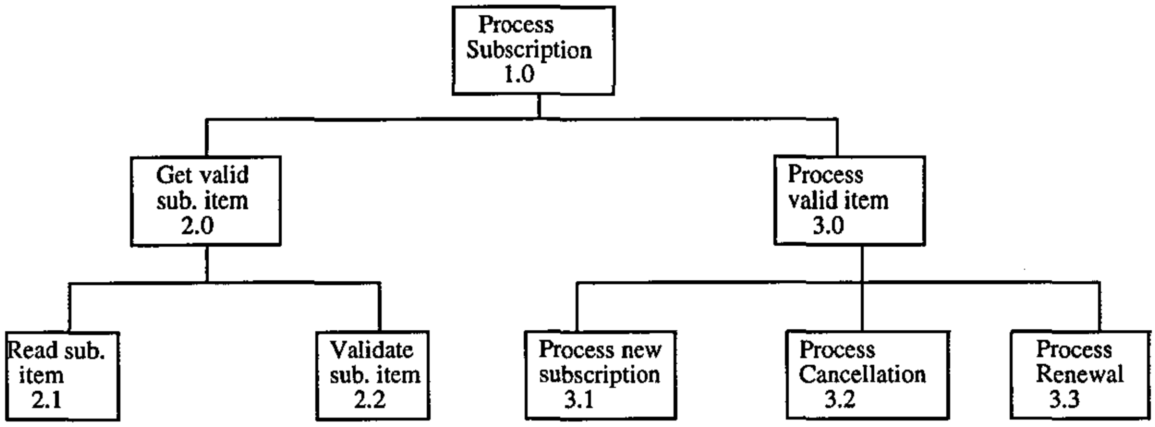


Figure 3.9 The visual table of contents in the highest level HIPO diagram (adapted from Martin and McClure 1985)

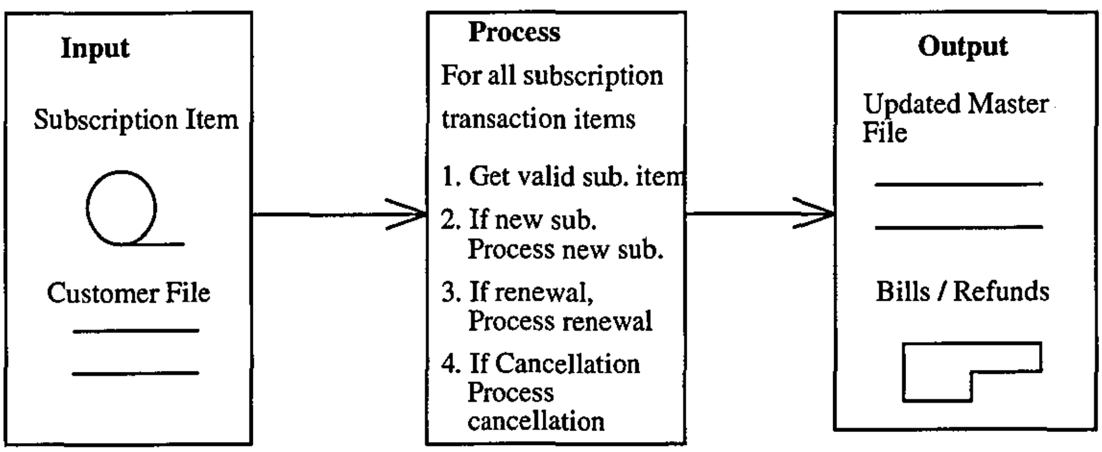


Figure 3.10 An overview HIPO diagram (adapted from Martin and McClure)

3.2.2.5 Warnier-Orr Diagrams

A Warnier-Orr diagram represents graphically the hierarchical structure of a program, a system or a data structure. It draws it horizontally across the page with nested brackets instead of down the page with blocks as shown in Figure 3.11. Like HIPO diagrams, when Warnier-Orr diagrams are used at a low level, they become large and difficult to read (Martin and McClure 1985). They have the advantages that they are easy to learn and use and that they offer one technique for both high level and detail design and for both procedure and data structure design (Gharib 1991). However, Warnier-Orr Diagrams have limitations for software building. They do not show conditional logic as well as other detail-level diagramming techniques do and they are not database oriented. Also, they do not show data flows. Their main use is for the structured design of computer programs as it is simple to transfer a Warnier-Orr diagram into structured program code because of its Begin-End structure format.

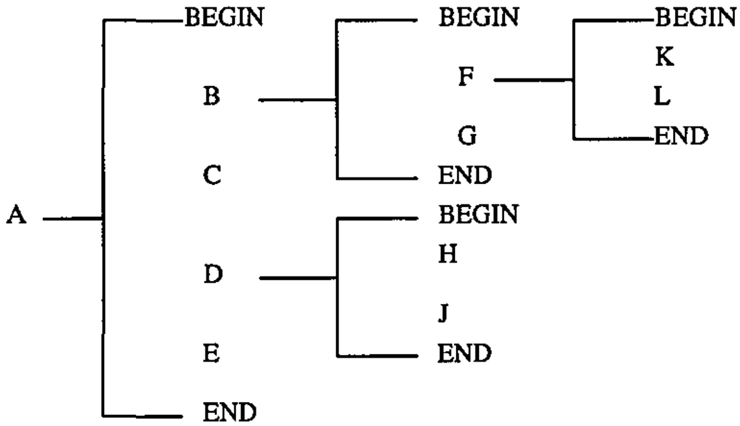


Figure 3.11 An example of a Warnier-Orr Diagram

3.2.2.6 Action Diagrams

Brackets are the basic building blocks of action diagrams. Inside the bracket is a sequence of operations entered from top to bottom. Inside a bracket there may be other nested brackets, the nesting shows the hierarchical structure of a program. Figure 3.12 shows the representation of a hierarchical structure with brackets. Data entering the process are written at the top right corner of the block and data leaving are written at the bottom right corner. However, data flow movements within a system are not shown with the same fluidity apparent in data flow models. (Gharib 1991).

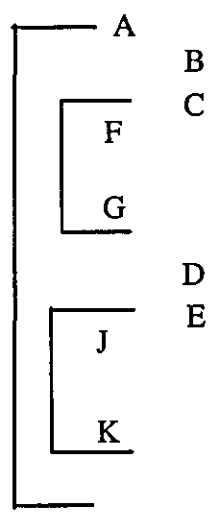


Figure 3.12 An example of an Action Diagram

3.2.2.7 Decision Trees

Decision trees were defined by Martin and McClure (1985) as a model of a discrete function in which the value of a variable is determined; based on this value some action is taken. Once a decision tree is executed, one path will be followed depending on the variable being tested. This path on the tree begins with the "root" and ends with a "leaf". An example of a decision tree is shown in Figure 3.13. Decision trees do not show data flows and hence, they are not suitable for modelling data and information flow.

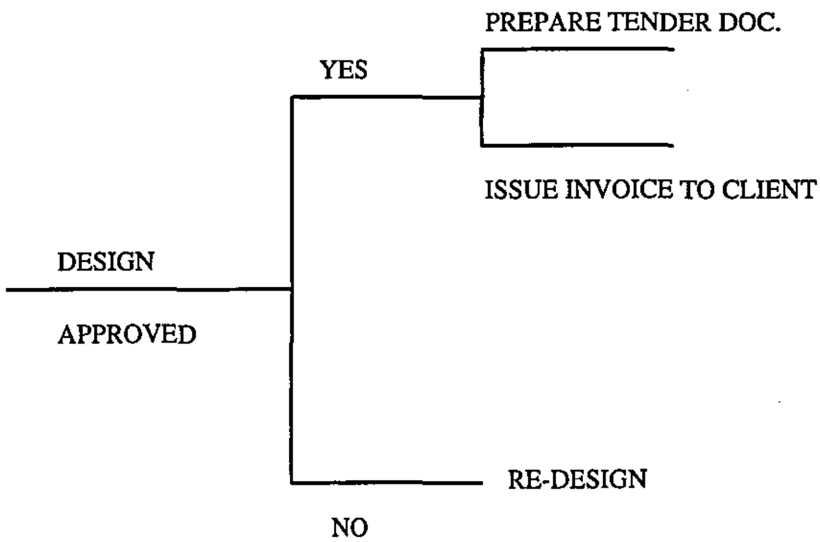


Figure 3.13 An example of a Decision Tree

3.2.2.8 HOS Charts

HOS (Higher-Order Software) is a rigorous form of functional decomposition. These forms of decomposition are precisely defined with mathematical rules at each step and thus are provably correct. The decomposition continues until blocks are reached from which executable program code can be generated. HOS is based on binary tree structures. An example of a binary tree structure is shown in Figure 3.14.

An HOS tree chart shows the decomposition of broad function of a system into sub-functions. The broadest overview is at the root of the tree and the leaves representing the primitive functions. HOS is a mathematically based tool suitable for a professional systems analyst. Although they are capable of modelling data flow, they are complex and not user-friendly (Gharib 1991). This view is supported by the writer.

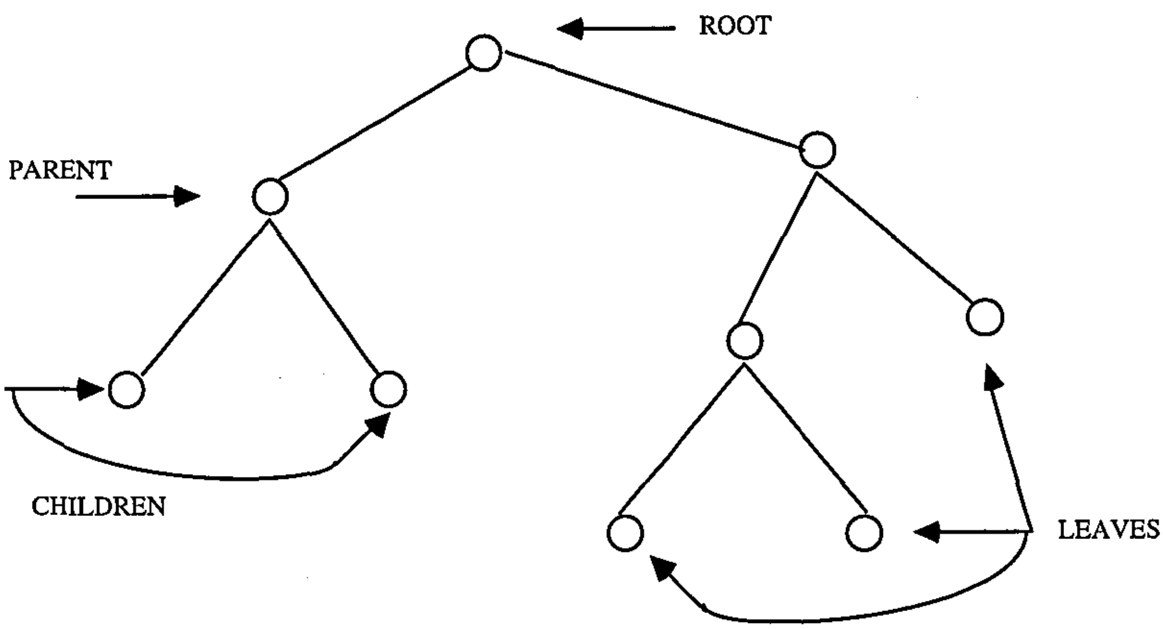


Figure 3.14 An example of an HOS Chart

3.2.3 IDEF0 Technique

The Integrated Computer Aided Manufacture Definition Method (IDEF0) is an automated graphical adaptation of Structured Analysis and Design Technique (SADT) aimed at standardising contractor communications and services. It has been developed by the U.S. Air Force in the early 1970s to standardise manufacturing process descriptions across many different aerospace contractors. (Plaria et al 1995, Colquhoun et al 1993, Eppinger 1992, Marca and McGowan 1988, Ross 1977). The basic building block of IDEF0 is a box representing an activity. Each activity is defined as an act which under control, transfers input into output using a mechanism. Hence, inputs flow into the box from the left, outputs flow out to the right, constraints flow into the top of the box and the mechanism, or those responsible for the activity, flow into the bottom side. This is illustrated in Figure 3.15.

IDEF0 models consist of a hierarchy of related diagrams. Each diagram is based on a diagonal row of boxes (activities) presented in node number order that represent the subject under scrutiny. The activities are connected by a network of arrows representing the inputs, outputs, controls and mechanisms. An example of an IDEF0 diagram is illustrated in Figure 3.16.

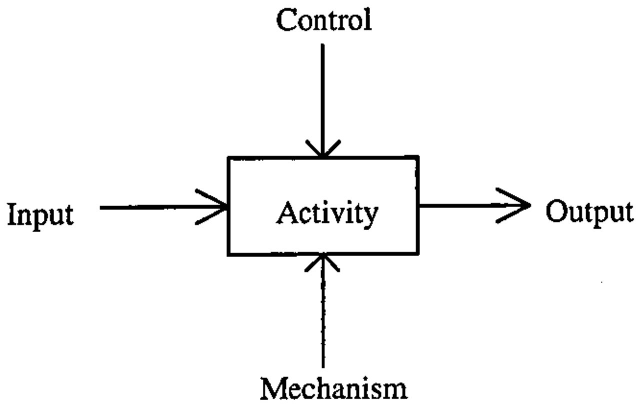


Figure 3.15 The building block of IDEF0 diagrams

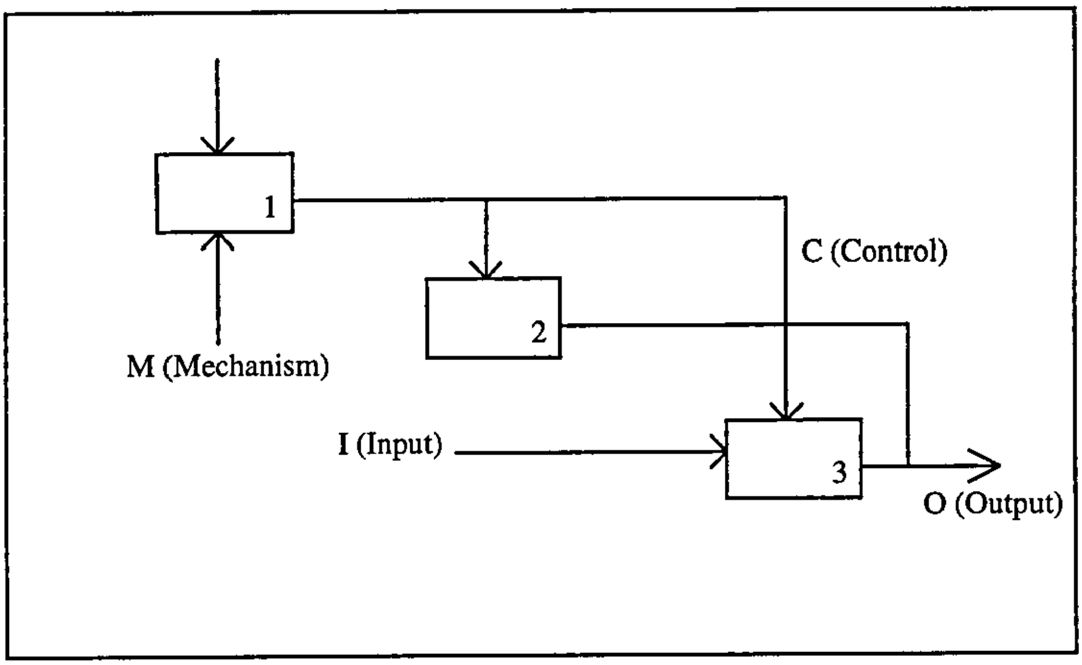


Figure 3.16 An example of an IDEF0 diagram

Supporting the basic principles of the technique is the IDEF0 forms and procedures guide (Colquhoun et al 1993, Ross et al 1980). These procedures provide a structured means of controlling, documenting and validating the model building process.

IDEF0 techniques share with DFDs the property of being a hierarchical top down approach. Each building block may be a component of another higher level block and may itself be decomposable into more component blocks. As with DFDs, balancing and consistency rules should be maintained between parent and children diagrams. However, the components of IDEF0 are different from those of DFDs. Whilst the DFD is composed of processes, data flows, data sources or sinks, and data files each represented by a different symbol, the IDEF0 is composed of an activity box and four arrows representing inputs, outputs, control and mechanisms. In highly sophisticated IDEF0 diagrams it is difficult to recognise the representation of each arrow, while in DFDs it is easier to recognise the representation of a symbol. This makes DFDs easier to read and understand more than the IDEF0 diagrams. This view of the writer is backed up by a comparison between IDEF0 and DFDs undertaken by Yadev et al (1988) and reviewed by Colquhoun et al (1993). The comparison showed that although the graphical presentation of the two techniques differs, IDEF0 having more rigorous set of rules, the concepts and model building process of DFDs are analogous to those of IDEF0. The basic difference between the two techniques being in the specific data analysis focus of DFDs. Although Yadev et al (1988) concluded that their comparison failed to establish which technique produces better results, they propose that the DFD technique is easier to learn and use. The writer concurs with this view. A similar comparison was undertaken by Maji (1988). He showed that the basic idea of IDEF0 and DFDs are very similar but that in the DFD, source/destination of data is shown, whereas in IDEF0 it is difficult to understand this aspect of the model. A similar conclusion was achieved by Mandel (1990). Moreover, the writer finds the nature of the DFDs symbols being processes, data flows, data sources or sinks and data stores makes them more suitable to model the building design process than the controls and mechanisms represented in the IDEF0, which are more suitable to model a manufacturing process.

3.2.4 Modelling the Design Process Using Structured Analysis Diagramming Techniques

Section 2.2 stressed the need for improved communications and a better understanding for the information exchanged during the design process. This need coupled with the data and/or process oriented modelling features offered by the structured analysis diagramming techniques has attracted researchers during the last decade to use these techniques in modelling the design process.

Sanvido and Norton (1994) developed an Integrated Design Process Model (IDPM) which combines the basic design activities that produce the physical design with other strategic activities on the corporate level. The model is represented in IDEF0 format. They argue that "by properly integrating the design process, a company can minimise design liability and exposure for a company". The model was constructed to represent the building design practice in the American building industry.

Apart from the writer's reservations on using the IDEF0 techniques to model the building design process previously highlighted in section 3.2.3, the model provides a balanced representation between technical design activities and corporate activities.

Venegas (1987) modelled the early phases of the design process to aid in the field of design construction integration research. The primary purpose of the model was to improve the constructability input in early design. The model highlights the means for integration and identifies the requirements to implement design/construction integration during the initial design phases. Venegas (1987) described the integrated design process and identified how the roles of owners, designers and constructors change under an integrated approach. He developed his own diagramming technique to build the model. This technique is a hierarchical top down approach and shares some characteristics with DFDs in that it contains processes, inputs, outputs and data flows although the symbolic representation is different. Venegas added more sophistication by introducing additional symbols for milestones, specific actions and decision points. He classified processes into principal processes and special processes. Although the writer recognises the benefits that the model may offer by improving the constructability input in early design, the excessive sophistication rendered the model complicated and difficult to read.

Newton (1995) developed a 'Design Process Model' which maps the design tasks and information flows involved in the detailed design of a building. The model consists of a basic framework to which smaller discrete sub-models are added. This framework is represented by a hierarchy of blocks. Different design alternatives or options are covered by different sub-models. Data flow diagrams were used to construct the sub-models to define the processes or tasks undertaken by each discipline and the information transfers between them (Newton, 1995). Using Steward's Design Structure Matrix (Steward 1981) previously explained in section 2.2.7, the functional primitive tasks of the DFDs were analysed using matrix manipulation techniques for partitioning and tearing. Dependencies between different tasks were determined by information requirements dictated by the DFDs taking into consideration the strength of dependencies.. The objective was to determine, based on information requirements, the optimum order for design tasks.

Newton did not adhere strictly to the rules of DFDs methodology. Although the hierarchy of blocks in the Design Process Model is represented by sets of decomposed

DFDs, balancing is not always fulfilled due to grouping information flows (which were named ICDI - Issued and Checked Design Information) which represent different information flows when used in different parts of the model (i.e. ICDI for load calculations is different from ICDI for steel work design). The details of information flows are shown only in the lowest level of the hierarchy (where the functional primitive tasks are represented). The sources of information at the lowest levels are represented by an alpha-numeric letter corresponding to documents produced by other tasks in the same discipline or in other disciplines in the 'horizontal dimension of the model'. This represents another deviation from the DFD methodology.

In research carried out by Hanby (1993), data flow diagrams were used to model the design process of a Heating Ventilation and Air Conditioning, HVAC, system. An individual model for each process is created and written to output required information such as the cost or time taken. The process models are linked by the data flows. Processes are represented as inputs and output contained locally within each process model. A knowledge-based system (CLIPS) is used to execute the processes where data flows are represented as facts and the individual process models written as rules. Once the input data becomes available, usually as an input from an upstream model, the model will execute generating information which is reported to the supervising program such as cost and time taken, and also output facts which will in turn enable downstream models to execute. Algorithms have been written to trace the effect of changes during the design of the HVAC system (Hanby et al 1993, Hedges et al 1993).

In this approach, information flowing between different processes in the Data Flow Model is one directional. There are neither iterative procedures nor interdependent tasks. There is also no interaction with processes in other disciplines. This approach is more suitable for modelling the design of mechanical systems than for modelling the building design process which involves iterative design tasks.

3.2.5 Other Applications of Structured Analysis Diagramming Techniques in the Field of Construction Management

The use of structured analysis diagramming techniques has not been limited to modelling at the design phase only, but has been extended to encompass construction and contractors operations. The suitability of the technique was confirmed by Gharib (1991) who had undertaken a preliminary study to model information flow in a design and build environment using data flow diagrams. This was supported by Fisher (1990, 1992) who used DFDs to build a data flow model for a construction company. He concluded that structured analysis techniques could have very considerable benefits when assisting the industry and its associated professions redesigning its products, processes and procedures so as to harness the maximum benefit from the technology. He recommended that there should be greater emphasis on the middle levels of DFDs which allow the development and validation of the current system and that simpler descriptions and specifications should be used in the data dictionary.

3.2.6 The Modelling Technique Used in the Research to Model the Building Design Process

The literature review about the design process in chapter 2 emphasised the importance of communications and information transfer issues as key factors for successful design management. This was the reason for the writer directing his attention to structured analysis diagramming techniques for modelling the design process where information flows between processes could be modelled. After a thorough examination of the previously mentioned structured analysis diagramming techniques, data flow diagrams were considered the most suitable technique for modelling the information transfer during the design process. This was primarily because data flow diagrams are a valid, proven, well established technique in the field of building design and construction research. More details about this decision are provided in chapter 4.

3.3 SIMULATION MODELLING

3.3.1 Introduction

Simulation is a dynamic process in which a model, a representation of a real system provides a basis for experimentation. This model may be a scale model, a physical model, or a set of mathematical equations and logical relationships (Paul and Balmer 1993). The experimentation process is used to iterate systematically towards an acceptable solution by repeatedly observing the performance of the model for different specific sets of conditions. An appropriate result is then selected from the set of outcomes that is obtained. This process thus allows different policies to be tested without being entangled with the real system (Pilcher and Flood 1984).

Therefore, the most advantageous aspect of simulation is the capability it offers for experimenting different scenarios on a representation for a real system (a model), but not on the system itself. The model is used as a vehicle to experiment on a trial and error basis to demonstrate the more likely effects of different policies. Hence, those that produce the best results in the model would be implemented in the real system. This important characteristic of simulation has been reflected in definitions provided by different researchers.

Pidd (1992) defined simulation as the process where the analyst builds a model of the system of interest, writes computer programs which embody the model and uses a computer to initiate the system's behaviour when subject to a variety of operating policies. Thus the most desirable policy may be selected.

Naylor (1966) defined computer simulation as a numerical technique for conducting experiments on a digital computer, which involves certain types of mathematical and logical models that describe the behaviour of a business or economic system (or some component thereof) over extended periods of real time.

The definitions provided by Pidd and Naylor are oriented towards 'computer simulation'. However, for simple systems simulation can be undertaken manually. More generalised definitions for simulation have been provided by Mize and Cox (1968) and Shannon (1975).

Mize and Cox (1968) defined simulation as the process of conducting experiments on a model of a system in lieu of either direct experimentation with the system itself or direct analytical solution of some problem associated with the system.

Shannon (1975) defined simulation as the process of designing a model of a real system and conducting experiments with this model for the purpose of either understanding the behaviour of the system or of evaluating various strategies for the operation of the system.

Shannon's definition is more comprehensive than that of Mize and Cox because it highlights not only the characteristic of conducting experiments on a model of a real system, but also shows the benefits that could be drawn from simulation in the decision making process through evaluating, before hand, various strategies for operating the system and thus decreasing the inherent risk. However, Shannon referred to the designing of the model but did not refer to its construction which is fundamental to the simulation process.

Therefore the writer's definition for simulation is " The process of designing and constructing a model of a real system and conducting experiments with the model for the purpose of understanding the behaviour of the system and evaluating the various outcomes to assist in the decision making process and decrease the inherent risk"
This definition includes the design and construction aspects of simulation modelling together with the important aspect of experimentation.

Pidd (1992) identified four advantages of simulation against real experimentation:

(i) Cost: Although simulation can be time-consuming and therefore expensive in terms of skilled manpower, real experimentation is also expensive especially if something goes wrong.

(ii) Time: Once a computer model is developed, it is possible to simulate weeks, months or years in a few seconds of computer time. Therefore, a whole range of policies may be properly compared.

(iii) Replication: Real world does not allow precise replication of an experiment. Simulations are precisely repeatable.

(iv) Safety: One of the objectives of simulation is to estimate the effect of extreme conditions, and to do this in real life may be dangerous or even illegal.

It is the first three advantages for simulation presented by Pidd and, concurred by the writer, which led to the writer's decision to pursue simulation techniques in order to analyse and experiment with the different criteria which influence the information transfer during the design process.

3.3.2 Phases of Computer Simulation

Simulation involves the setting up of a model of the system under study, in which all relevant components are defined, and the way in which they change through time and affect each other are exactly specified. This model is then set in motion and its behaviour is observed. It is allowed to run for a certain time and a comparison is held between the values taken by variables in the model and the values taken by corresponding variables in the real system (Paul and Balmer 1993). If there is close correspondence, then the model is considered a good representation of reality. Therefore, the model provides a potentially powerful tool to conduct controlled experiments by systematically changing different parameters and re-running the model.

There are three phases for computer simulation (Pidd 1992).

(i) Modelling

(ii) Programming the simulation model

(iii) Experimentation

3.3.2.1 Modelling

A model is essential in computer simulation to mimic a real system by unfolding the model through time. In order to be useful, the model should be valid. There are two types of validity:

(i) Black box validity: This ignores the detailed internal workings of the model and is concerned only with the predictive power of the model. Black box validity has been undertaken by the writer while testing the simulation model developed within this research. The output of the simulation model in the form of a schedule for design tasks has shown a logic sequence of carrying out these tasks in consistency with the data flow model (e.g. no task can start before receiving its necessary information). More details of this aspect are provided in chapter 6.

(ii) White box validity: This ensures that the components of the model represent known behaviour and/or any valid theory which exists. This type of validity is mainly related with random procedures which may follow different probability distributions.

3.3.2.2 Programming the Simulation Model

Whatever the choice of programming technique for the simulation model, there is a growing tendency for a highly disciplined and structured approach to be taken to the programming. Different programming techniques will be discussed later in this chapter.

3.3.2.3 Experimentation

The final phase in any simulation project is to carry out different experiments on the model in order to observe the behaviour of the system represented by the model under different operating strategies. The experiments must be planned so that the various factors which may influence the results can be disentangled. Therefore, the experimenter can determine the effect of the different factors on the system being simulated.

The aforementioned three phases for simulation are difficult to separate in practice, particularly the modelling and the programming phases. It is not practically possible to program without an adequate model and experimentation is impossible without having a working program; nevertheless, some overlap will occur.

3.3.3 Different Considerations for Simulation Modelling

Three different aspects are to be considered in building any simulation model (Pidd 1992):

(i) Time Handling

(ii) Stochastic or Deterministic Durations

(iii) Discrete or Continuous Change

3.3.3.1 <u>Time Handling</u>

A big advantage in simulation is to control the speed at which the experiment proceeds. There are two approaches :

Time Slicing

This is the simplest way to control the flow of time in a simulation as time is moved forward in equal intervals. However, a decision should be taken about the length of the time slice before the simulation starts. If a wrong decision is taken, or certain events within the system occur in an unequal interval of time, then wasteful and unnecessary checking of the state of the model and consequently longer run time will occur.

Next Event Technique

It is often preferable to use variable time increments as many systems include slack periods of different lengths. In this case, the model will be updated and examined only when an event (a change of state) is due. This approach is called the next event technique. Its main advantage is that the time increment adjusts itself automatically to periods of high and low activity, hence avoiding wasteful and unnecessary checking of the model state, and that it makes clear when significant events have occurred in the simulation. However, more information should be held for controlling the simulation, (Pidd 1992).

The next event technique has been used by the writer for time handling while developing the Design Simulation Model within this research. This is because design tasks vary in durations and using time slicing will result in inefficient running of the simulation model. The next event technique assures that the model state is checked only whenever an event occurs, such as completing a design task. (More details of the simulation model are provided in chapter 6.)

3.3.3.2 Stochastic or Deterministic Simulation

Deterministic Simulation

When a system is clearly understood, it is possible to predict precisely what will happen. Therefore a deterministic system is one whose behaviour is completely predictable. An example of such a deterministic system is a cycle of operations in an automatic machine.

Stochastic Simulation

A stochastic system is one whose behaviour cannot be completely predicted. However, it might be known how likely certain events will occur. In this case, random sampling and probability distributions will be used to predict durations of certain activities or frequencies of occurrence of certain events. A very popular stochastic simulation method based on random sampling is called Monte Carlo Simulation. An example of the use of stochastic simulation is to simulate the times for the break down of a machine.

The simulation model developed by the writer for the design process has the capability of running in either deterministic or stochastic mode according to the user's choice. The deterministic mode may be used by the design manager to schedule design tasks according to the specified durations and constraints. On the other hand, the stochastic mode may be used to assess the likelihood of completing a design project at a certain time by using random durations for design activities. Further details are provided in chapter 6.

3.3.3.3 Discrete or Continuous Change

Discrete Change

In discrete event simulation (simulation with discrete changes), the variables in interest are only those pointing to a change in the state of the system. Discrete event simulation will be discussed in more detail later in this chapter.

Continuous Change

In continuous simulation, models allow continuous change when variables are continuously changing their value as the simulation proceeds. These changes could be represented by differential equations which, theoretically, allow variables to be computed at any period of time. Typical examples are economists modelling behaviour of economic systems through differential equations or engineers simulating equipment they design.

Since this research deals with modelling design information flows which are released at discrete points of time (e.g. whenever a design task has been completed), it was decided to focus only on discrete event simulation.

3.3.4 Simulation Applications in the Field of Construction Management

The existence of simulation as a powerful management tool has been always tempting for the construction industry to be applied in different managerial areas, particularly those involved with cyclic or repetitive operations. Most of the simulation models used depend for their time element on random sampling from frequency distributions compiled from the range of lapsed durations for the different elements of work in a cyclical operation representing the activity of a resource.

Pilcher and Flood (1984) developed a discrete event stochastic simulation model for construction operations Their objective was to determine the most economic resource combinations needed to undertake construction activities. The model was applied on a simple excavation operation and on a sensitivity analysis of a concrete mixing and distribution system.

Woolery, and Crandall (1983) developed a stochastic network simulation model for construction scheduling. It consists of dependent and independent random variables, (like weather delays, legal delays and environmental delays) and is based on Monte Carlo Simulation. Data for each network activity consist of a time distribution for the activity under optimal conditions and a series of time distributions for different problems that may lengthen the completion time of the activity. Dependencies between network activities were considered and modelled. They showed also that time or seasonal dependencies for a network activity may be modelled.

Ahuja and Nandakumar (1985) developed a simulation model to forecast a construction project completion time based on simulating expected occurrence of uncertainty variables. From information collected for progress update of the tactical plan and by simulating the project environment, the combined impact of the uncertainty variables could be predicted for every progress period based on Monte Carlo simulation. Including the combined impact in the duration, estimate of each activity will generate a new activity duration distribution. Consequently, the probability of achieving the original project completion time and of completing the project at any other time is computed.

Carr (1979) developed a simulation model for uncertainty determination (MUD) to simulate construction project durations. The simulation is performed on two stages: First is a Monte Carlo sampling of all random variables independent on the time of the year the activities are performed containing most uncertainties except weather. Second step is to include the weather uncertainty. The actual duration samples were used as an input to CPM and every single simulation output was subject to statistical processing to calculate activity mean time and standard deviations.

Halpin (1977,1992) developed a simulation model CYCLONE (CYCLIC Operations NEtwork) based on building networks of active and idle states to represent cyclic construction processes.

Kalk (1978) introduced some refinements to the early versions of CYCLONE, in terms of the computer program structure, and called it INSIGHT. He used it with Paulson et al (1981) as a module in a more sophisticated computer program. Other modules of the program included Data Capturing using time lapse photographic documentation of operations, statistical analysis of data, and interactive graphics.

CYCLONE has been mainly developed to model and simulate repetitive cyclic construction operations like earth moving and concrete mixing and pouring. It is not suitable to model and simulate iterative inter-dependent activities such as design activities. This has been confirmed during a discussion between the writer and the software developer.

Dawood (1991) developed a computer based capacity planning system for precast concrete production with the objective of improving the efficiency of the production process. As with other simulation applications within the field of construction management, the model developed by Dawood has been based on activities of cyclic nature which are involved in the production line of a precast concrete factory.

In a research carried out by Laurikka (1993) to assess the suitability of applying simulation tools to the construction industry, he concluded that these tools are suitable for building construction production planning in limited problem areas only. They have been developed to satisfy manufacturing industry's need, but they are usually inappropriate in planning construction site operations due to the difference in nature between the two industries. He identified the most applicable area in which simulation could be applied within the construction industry to be when simulating operations and actions which are cyclic in nature. However, the writer argues that simulation techniques have other potential applications within the construction industry if the suitable technique is being selected. This is being demonstrated throughout this research during the evaluation of the different options for developing a simulation model for the design process and the subsequent development of that model.

To summarise, the areas in which simulation techniques have been used in construction management research are:

(i) Planning of the effective use of construction resources, especially for cyclic operations like earth moving and concrete mixing and pouring.

(ii) Scheduling of construction activities considering random uncertainty variables.

(iii) Forecasting construction duration and construction project completion time considering expected occurrence of different uncertainty variables.

(iv) Modelling and simulating repetitive cyclic construction operations for the purpose of improving these operations in terms of time, cost, output and productivity.

The literature search undertaken by the writer has shown that there has been no application of the simulation techniques in the area of management of the design process and all the applications reviewed have been only for the management of construction. This provided the writer with the momentum to pursue the application of simulation techniques to the management of the design process.

3.3.5 Simulation and Structured Analysis Techniques

The literature reviewed showed that there has been very little work done in the area of combining DFDs and simulation (ESD 1989, Ward and Mellor 1985, Warren et al 1992). The writer believes that this is due to the fact that DFDs are static and show information flows only but do not include any time element, while simulation deals with changes in systems as time elapses.

The main area where this combination has existed was in the development of real time embedded systems like computerised systems in spaceships or aeroplanes. This is called Real Time Systems Analysis. Simulation extensions were added to data flow diagrams in sophisticated, expensive, work station based software. A simulator checks the model's functional and timing behaviour under various conditions and different scenarios could be created to exercise the real time embedded system specifications. This approach is mainly oriented towards modelling and simulating software and hardware and is not suitable for simulating the design process.

Warren et al (1992) developed a work station based prototype system that automatically produces stochastic discrete event based simulation models from data flow diagrams. DFDs are augmented with dynamic attributes and simulation results are produced directly from a CASE tool data dictionary. This is achieved by associating a simulation run parameter input window with every process. This approach provides a systematic way to convert DFDs from their static state to a dynamic state through introducing time elements to each process in the DFD and hence allows simulating the data flow model. However, it does not allow for allocating any attributes to information links between the different processes which may reflect different parameters that affect the information exchange. Therefore, the writer finds this approach unsuitable to simulate information related criteria during the design process.

Mujtaba (1994) developed a simulation model for the order-to-ship (OTS) process to simulate the activities that occur between the receipt of orders and the shipment of products within a factory. A graphical model of the OTS process has been built using a process modelling technique named Hierarchical Process Modelling (HPM) which has been developed within the same research. HPM is a derivation of IDEF0 technique described in section 3.2.3 and borrows DFD modelling technique to model the context or environment of different blocks comprising the IDEF0 diagram. The HPM has been converted into a discrete event simulation model using object oriented programming simulation language. This required building another graphical model for the OTS process as a pre-requisite to writing the object oriented simulation model. The simulation model has been written in object oriented programming language because it involves interactions of many entities of different types such as products, parts, shipment, order, vendor, factory, customer, etc.

The similarity between this approach and the approach used by the writer within this research is in the concept of transforming a structured analysis diagram into a simulation model through writing simulation routines. It was not necessary for the writer to use object oriented simulation programming due to the limited number of classes of entity involved (design tasks, resources) and hence the conventional approach has been found more appropriate. This facilitated the transformation process as there was no need to build another graphical model for the simulation purposes. The data flow model for the design process, developed by the writer, has been also used as the graphical model required to build the simulation model (at the level of the FPTs). The writer believes that the approach adopted by Mujtaba is a complicated, expensive approach, (the research team consisted mainly of Hewlett-Packard employees who had under their disposition various hardware and software), that requires building two different graphical models for the same process in addition to a computer model for the simulation purposes. These tasks would appear to outweigh any advantages of using the object oriented approach.

3.3.6 Options of Simulation Tools for the Research

3.3.6.1 Traditional Simulation Approaches

As previously mentioned, only discrete event simulation will be investigated for this research.

The concept of discrete event simulation was described briefly in section 3.3.3.3. In discrete event simulation a system is usually constituted of objects and operations in which these objects engage (Pidd 1992). Terminology associated with objects of the system could be:

- Entities: Elements of the system being simulated and can be individually identified and processed. They could be either permanent or temporary. Examples of entities used by the writer in developing the simulation model are different design tasks and different resources.

- Classes: Entities of the same type are grouped in classes.

- Attributes: Each entity may have one or more attributes that give more information about the entity. Examples of attributes used by the writer in developing the simulation model are values given to information links which represent information quality, missing information, etc.

- Sets: They represent change of states of entities during the simulation. Example of these sets in the developed simulation model is the change in the state of tasks from started to completed.

Terminology associated with operations of the system could be:

- Event: An instant of time at which a significant state change occurs in the system. An example is the 'event' that a design task starts.

- Activity: Operations and procedures initiated at each event such as calling the necessary resources to carry out a design task

- Process: When a sequence of events are grouped in a chronological order in which they will occur, this is called a process.

- Simulation Clock: The point reached by simulated time in a simulation.

There are several approaches to discrete event simulation:

Writing Simulation Programs in a General Purpose Language
A simulation programme could be written in Fortran, Pascal or any general purpose language. It needs strong programming skills and it is not recommended except in certain circumstances because it is like 're-inventing the wheel' due to the existence of simulation programming languages.

Simulation Programming Languages
These are languages whose problem orientation is towards the specifics of simulation programming. Examples are Simscript, Simula, Modsim, Ecsl and others. They require writing simulation programmes in their own syntax and hence they require the user to learn the simulation programming language.

They follow one of the following approaches:

(i) The Event Approach
A set of event routines each of which describes the operations in which entities engage when the system changes state. It involves execution only of possible events and hence it runs faster than activity based, however, it is more difficult to write.

(ii) The Activity Approach
This concentrates on the interactions of the various classes of entity, rather than on mapping out the possible operations which might follow from a state of change as in the event approach. It is easier to write, but it treats each activity as independent which leads to run time inefficiency.

(iii) The Process Interaction Approach
The whole life cycle of an entity is taken as the basic logical building block of a simulation model, the progress of an entity being stopped temporarily by either unconditional or conditional delays.

(iv) The Three-Phase Approach
A combination of the simplicity of the activity approach with the efficient execution of the event approach (Pidd 1992). It is based on two types of events:

B-events (Bound or Book-keeping events): They are the events which are executed directly whenever their scheduled time is reached.

C-events (Conditional or Co-operative events): They are the events which their execution depends on the co-operation of different classes of entity or on the satisfaction of specific conditions within the simulation.

The Three Phase Approach to discrete event simulation has been used by the writer to develop the simulation model within this research. In addition to the aforementioned advantages, the writer found the nature of the B-events and C-events associated with this approach suitable for building the simulation model. Further details about the selection of the simulation technique for this research are discussed later in this chapter and in chapter 4.

Simulation Data Driven Packages

They are mainly based on Activity Cycle Diagrams, which is a way of modelling the interactions of entities and are particularly useful for systems with a strong queuing structure. The main elements of an activity cycle diagram are entities, queues and activities. The diagram shows the life history of each class of entity and displays graphically their interactions. Each class of entity is considered to have a life cycle which consists of a series of states and the entities move from state to state as their life proceeds. The states could be active state involving the co-operation of different classes of entity or idle state which involves no co-operation between different classes of entity and it is a state where the entity is in queue. Probability distributions need to be established for activity durations in case of stochastic simulation. Simulation data driven packages are more suitable for simulating typical repetitive cycles such as customers arriving and queuing or car arrivals at a car park. This is different from the nature of the design process and hence this approach was not pursued.

Simulation Modelling Environment

This type of software is a hybrid of simulation programming languages and simulation data driven packages. It is a programming environment that includes its own language and constructs that may be used in simulation, and can be used as a core for a simulation software. This offers the user the possibility of programming those unique portions of the system that cannot be adequately modelled by the data driven packages built in constructs.

3.3.6.2 Knowledge Based Systems

Computer programs using Artificial Intelligence techniques to assist users in solving difficult problems involving knowledge, heuristics and decision making are called knowledge based systems (KBS) or expert systems. (Adeli 1988, Allwood, 1989). A KBS is an 'intelligent' interactive computer program that can play the role of a human expert by using heuristic knowledge or rules of thumb. The heuristics are usually accumulated by a human expert over a number of years. Using heuristics, a knowledge based system can make educated guesses, recognise promising approaches and avoid blind search; and consequently it can narrow down the search process in a solution space (Adeli, 1988). The components of a knowledge based system are represented in Figure 3.17. More details of KBS may be found in Adeli (1988), Allwood (1989), and Harmon, Maus and Morrissey (1988).

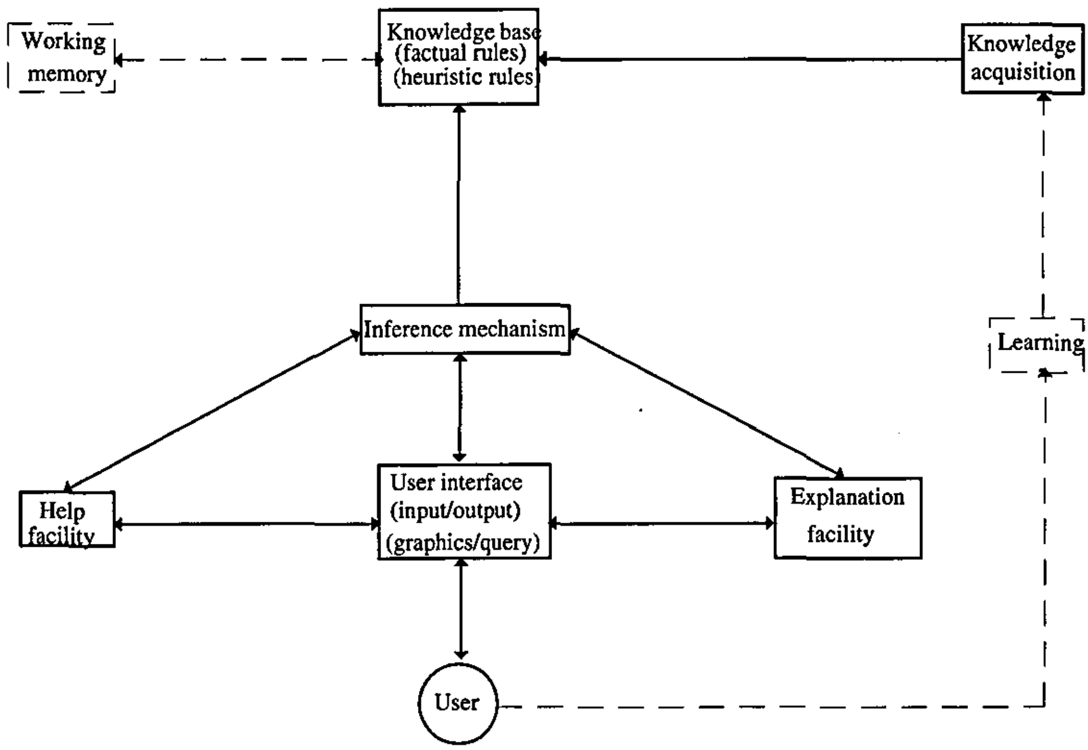


Figure 3.17 The components of a Knowledge Based System

Touran (1990) has attempted to combine KBS with simulation systems by using the KBS as a front end to a simulation model. The KBS interfaces between the user and the simulation model and assists the user to make the appropriate decisions necessary for the data input to the simulation. A prototype model has been developed which interfaced CYCLONE simulation software with an expert system shell (an expert system shell is the expert system without the knowledge base, consisting of the inference mechanism and working memory and other development facilities). The simulation model has been built for a cyclic earth moving process and the role of the KBS was to assist the user to select the appropriate earth moving equipment. The decision for the equipment used has an impact on different parameters within the simulation model.

The writer argues that this approach is more appropriate when a sophisticated selection process or a difficult decision is a pre-requisite to running the simulation model. However, when other types of rules are required to be incorporated in a simulation model (such as information availability, or resources availability in the case of executing design tasks), advantages have to be taken of the simulation languages or environment where these rules may be incorporated in a simulation model. This will lead to one simulation model which is more simple than attempting to interface a simulation model with a KBS.

From the definitions for simulation mentioned earlier in this chapter, the purpose of simulation is to experiment the response of a system to different policies. Knowledge based systems, based on rules and facts, can be used as a simulation tool when simulating information flows in a data flow diagram. Availability of data flows can be considered as rules to execute processes. Two similar approaches have been used by Rogers (1989) and Hanby (1993) and are described in section 2.2.7 of this thesis. However, from the review of KBS, the writer finds such systems in their typical forms are more suitable (within the context of this research) for simulating the effects of missing information or changes in design information through mapping all tasks that have not received their necessary information or have been affected by any changes. They are ill suited to simulate dynamic events which are associated with a simulation clock that advances according to the durations of these events. This is because KBS are mainly structured to assist in the decision making process based on combinations of different rules. Conventional discrete event simulation techniques are more suitable for carrying out the simulation of dynamic events. Nevertheless, rules and facts could be incorporated in a discrete event simulation model by using the capabilities of the simulation programming languages or simulation environment previously described in section 3.3.6.1.

3.3.6.3 Network Analysis

This approach, developed by the writer, combines network analysis with DFDs and the Design Structure Matrix (DSM) to produce a simulation model to simulate the order of design tasks and tasks durations on 'what if' basis and to evaluate the impact of design changes upon construction. Each of these three techniques have been described separately in detail in sections 2.2.5, 2.2.7 and 3.2.2 of this thesis. A proposed prototype model illustrating this approach is shown in Figure 3.18.

The DFDs representing the design process are used to identify the information flows necessary to complete the design. The functional primitive tasks of the data flow model are arranged in the DSM and matrix analysis techniques are used to optimise the order of the design tasks.

After achieving the optimum order for the design tasks the user may review these tasks and construct the precedence network. The network is drawn based on the order identified from the matrix. Where appropriate, design tasks may be collected together and represented by a single activity.

The process will not however eliminate all the cycles within the design work. Where a circuit of design work still exists, the design manager must estimate how many 'cycles' of design should be undertaken. The circuits of design may then be "unwrapped" to provide a precedence diagram without circuits.

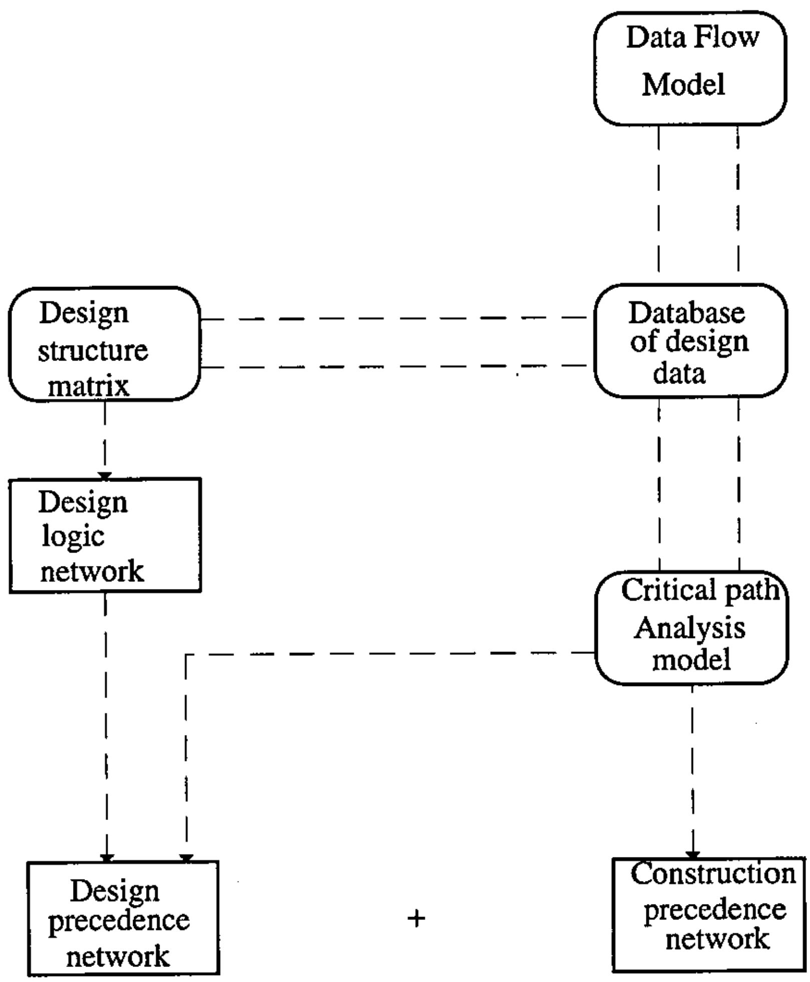


Figure 3.18 An overview of the simulation approach using critical path analysis

This process of "unwrapping" is described by Steward (1981). Figure 3.19 shows a matrix of ordered tasks. This may be re-drawn as a precedence diagram. Figures 3.20 and 3.21 show this taking place in two stages. Figure 3.20 shows the diagram with circuits included. Figure 3.21 shows the diagram re-drawn assuming two iterations of design within the outer cycle, i.e. preliminary and final design. For the inner design cycle it has been assumed that two iterations will take place during the preliminary design and one in the final design.

Where the precedence diagram shows a design task being repeated several times, consideration must be given to the assumed duration of the design activities at each stage of the cycle. When a design task is repeated several times the time taken to repeat the work on successive cycles may be expected to reduce. The design manager should use previous experience to determine suitable durations. Alternatively an algorithm such as that provided by Steward (1981) may be used. This algorithm provides a method of assessing the duration of each iteration of the task which assumes a set up time for the first iteration then a percentage reduction in the design time for each subsequent iteration.

PSM	1 :	2 :	3 :	4 :	5 :	6 :	7 :	8 :	9 :	10
1 :	1 :									
2 :	X	2 :								
3 :	X		3 :							
4 :			X	4 :					X	
5 :				X	5 :					
6 :					X	6 :				
7 :						X	7 :			
8 :						X		8 :		
9 :								X	9 :	
10 :									X	10

Figure 3.19 The matrix of ordered design tasks

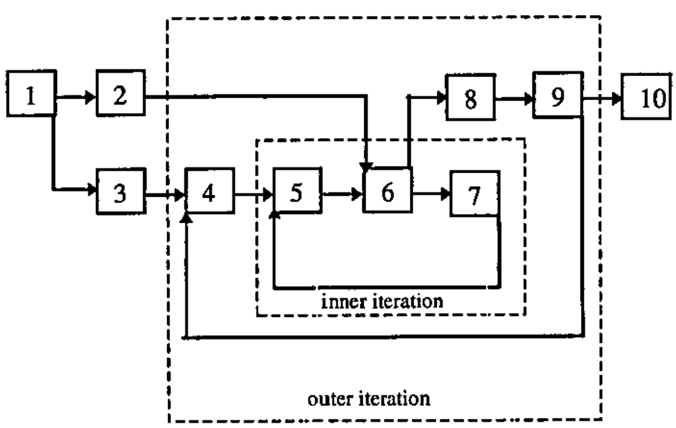


Figure 3.20 A precedence graph developed from the matrix (adapted from Steward, 1981)

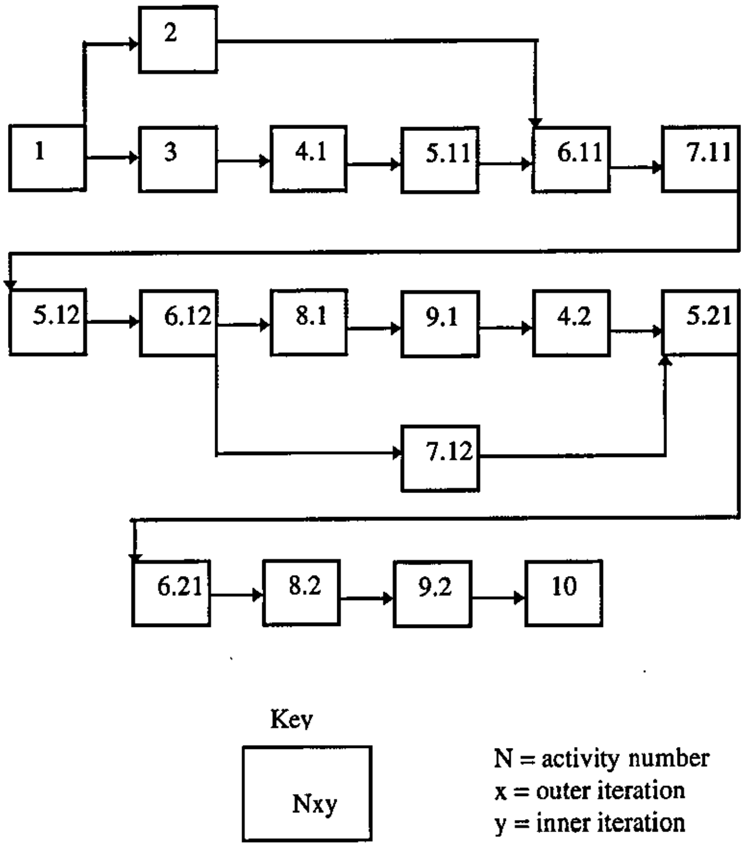


Figure 3.21 A precedence diagram adapted from Figures 3.19 and 3.20

The Application of this Simulation Approach

This model permits the simulation of the following common events during design and construction:

- changes in the design tasks;
- the availability of design information;
- changes in procurement strategy;
- changes in construction method;
- the delivery of design information for construction; and
- changes in the availability of design staff.

The tasks that need to be undertaken within the design process may change due to decisions relating to the type of building product or the design work involved. When designers change their designs, a different information flow may be required. Requests from the contractor or changes in the procurement strategy for the construction work may demand the release of certain packages of information before the design team would normally prefer to complete the design. The simulation model is able to reflect such changes by: removing/changing existing links; establishing new links within the Design Process Model; establishing new functional primitive tasks; repeating the matrix modelling process; and producing a revised precedence diagram. Similarly, should the contractor decide to make changes in the construction method, the model will allow a review of the design tasks involved and the time required to complete the work.

Having established an agreed programme for the design work, the design manager may effectively monitor and control the production of the design deliverables. The production of the network allows the application of established techniques based on critical path planning techniques. In addition the design manager will have access to structured data flow diagrams to assist in monitoring the arrival and dispatch of information within the design process. By linking the precedence network for design to the precedence network for construction, the full impact of design changes may be evaluated. An example of this approach is provided by Baldwin et al (1994).

After developing the prototype for this type of simulation, and while investigating other alternatives for the simulation technique to be used within this research, the writer decided not to pursue this approach to transform the prototype model to a full scale model. This is because this approach does not allow the user to monitor instantaneously the changes in different design tasks and/or resources as the simulation time elapses, nor does it allow the interaction with the model. Additionally, this approach involved the laborious task of constructing two graphical models: the Data Flow Model and the design/construction network after unwrapping iterative design loops. Investigations undertaken by the writer for different alternatives for the simulation technique to be applied for this research has shown that this approach, although feasible, does not represent the most appropriate way forward. This is discussed in more detail in Chapter 4.

3.3.7 The Simulation Technique used for this research

After thorough investigations of the different options for techniques to simulate the design process, it is hypothesised that the use of the Three Phase Approach to discrete event simulation using a simulation modelling environment provides the most suitable simulation technique for this research. Discrete event simulation allows the user to interact with the model and to instantaneously observe the changes that occur in the model as the simulation clock advances. The type of events involved in the three phase approach have been found suitable to represent events that occur during the design process. The simulation modelling environment provides flexibility in the modelling aspects, with the possibility of incorporating conditional rules to execute the required actions while running the model. This facility is not found in simulation data driven packages. However, a considerable programming effort is involved. Further details about this decision are provided in Chapter 4.

3.4 SUMMARY AND CONCLUSIONS

This chapter has reviewed relevant literature and research on different approaches to modelling the design process and on different simulation modelling techniques. The objective of this review was to select the most appropriate modelling approach and simulation technique to the design process.

Early models were either descriptive or prescriptive showing the different stages of design and emphasising its iterative nature. Some models addressed the different ways of thinking and learning styles of designers and the factors that influence them. However, none of these models addressed in detail the information transfer and communication issues which have been identified by different researchers as the key factors to the successful management of the design process. It was not until the late 1980s when structured diagramming techniques developed for systems analysis purposes were used to model the design process and to show the information exchange within the process.

Nine methodologies for structured analysis diagramming techniques have been reviewed to assess their suitability for modelling the design process. The advantages and disadvantages of each have been highlighted. The review has concluded with the hypothesis that the data flow modelling technique is the most suitable to use in this research for modelling information transfer during the design process. It is a valid, proven, well established technique in the field of building design and construction research. Therefore, the writer decided to adopt the data flow modelling technique to model the building design process.

Simulation as a powerful management tool has been used by the construction industry in different managerial areas, particularly those involved with cyclic or repetitive operations. Typical applications of simulation techniques in construction management research have been the areas of planning of the effective use of construction resources, (especially for cyclic operations), scheduling of construction activities considering random uncertainty variables, forecasting construction duration and construction project completion and modelling and simulating repetitive cyclic construction operations for improvement in terms of time, cost, output and productivity. However, the writer argues that simulation techniques, within the construction industry, should not be limited to cyclic operations. They have further potential applications if a suitable technique is selected.

The literature survey undertaken by the writer revealed a gap in the application of simulation techniques to the design process. This gap has provided the writer with the momentum to pursue the application of such techniques to design management.

Few attempts have been made to combine simulation techniques with structured analysis techniques. The main area where this combination has existed was real time systems analysis. This approach has been found to be mainly oriented towards modelling software and hardware and is not suitable for simulating the design process. Other attempts included the development of an object oriented simulation model based on an IDEF0 model and a second object oriented graphical model for a factory operation. This approach has been found complicated and expensive and is not suitable to simulate the design process because it does not involve as many classes of entity as those involved in the factory operation.

The options of simulation tools to be used for this research were conventional discrete event simulation, a knowledge based systems approach and a network analysis approach.

Different options for conventional discrete event simulation have been reviewed. These included writing simulation programmes in a general purpose language or simulation programming languages, simulation modelling environment and simulation data driven packages. Different approaches for simulation programming have been also reviewed.

An overview for knowledge based systems has been provided highlighting their potential use as a simulation technique based on information from data flow models. Attempts of combining simulation techniques with knowledge based systems have been also reviewed.

A proposed prototype model, developed by the writer, combining data flow diagrams, the design structure matrix and network analysis to produce a simulation model for the design process has been presented. The approach, although feasible, is not considered the most appropriate way forward for the research.

After thorough investigation of the different options of simulation techniques to simulate the design process, the writer has hypothesised that the use of discrete event simulation technique using a simulation modelling environment which supports the three phase approach provides the most appropriate simulation technique for this research. Discrete event simulation allows the user to observe instantly the changes that occur in the simulated events involved in the three phase approach. The types of these events have been found analogous to the events that occur during the design process. More flexibility in the modelling aspects is provided when using a simulation modelling environment than with simulation data driven packages.

The chapter concludes with the hypothesis that techniques based upon data flow diagrams, matrix analysis and discrete event simulation will improve the management of the design process as they will provide design managers with sophisticated tools to aid them in managing the process.

PART III

EXPERIMENTAL WORK

CHAPTER 4

RESEARCH METHOD

4.1 OVERVIEW

This chapter describes the research methodology adopted throughout the course of this research. After setting the research aim and objectives, a seminar attended by representatives of the collaborating companies was held. This included Ove Arup and Partners (Nottingham Branch), Ove Arup and Partners (Birmingham Branch), AMEC Design and Management and Hawker Siddeley Power Engineering. The purpose of the seminar was to present the research objectives, the proposed methodology and the contribution that this research would provide to improve the management of the design process. The writer was encouraged by the response of the participants who confirmed the benefits that would be drawn from the research which reflected the need of industry practitioners for sophisticated tools to improve the management of the design process. This was backed up by initial interviews undertaken by the writer with design managers from the collaborating companies and also the John Laing construction organisation all of whom emphasised the need to investigate the current practice for design management. (The results of these interviews are presented in section 2.2.5.)

The main focus of this research was the Conceptual and Schematic design stages. The literature survey undertaken by the writer revealed that research in the area of managing information exchanged during the early stages of design prior to the production of drawings and tender documents was very limited. However, as indicated in Chapter 1, all phases of the design process were reviewed. Stages C and D of the RIBA plan of work were considered as 'guidelines' during the development of the models produced.

Feedback and collaboration from the industry was an on going process throughout the development, verification and validation of the models and case studies produced during the course of this research. Tables 4.1 and 4.2 summarise the role of the industry professionals during the different stages of the research.

Project Objectives	Arup Nottingham Case Study	Arup Birmingham Case Study	Management of the Design Process	The Generic DFD	Generic DFD +Comm. Problems+ Sim features	AMEC Case Study	The Simulation Model	The Simulation Model Contd.
Seminar at LUT Attendees:	P. Geeson Assoc. Dir. Arup Nott.	B. Clifford Assoc. Dir. Arup Bir.	G. Marshall PM Arup Nott.	D. Storer * PM Arup Bir.	Survey + Interviews (see attached list)	D. Hammond PM AMEC	D. Storer ** PM Arup Bir.	W. ** Rochester Senior Proces eng. AMEC
P. McGee AMEC	G. Marshall PM	D. Storer PM	D. Storer PM Arup Bir.	T. Atkinson ** PM Laing			D. Hammond *** PM AMEC	
M. Murray Director AMEC	S. Cliffe Str. Eng.	M. Whild Architect	T. Atkinson PM Laing	D. Smith ** Client rep. Fisons			M. Murray ** Director AMEC	
P. Geeson Assoc. Dir. Arup Nott.			D. Smith Client rep. Fisons	D. Hammond* Design leader AMEC			P. Waskett ** Mech. eng. AMEC	M. ** Butterworth Architect AMEC
J. Perks Hawker Siddely			A. Newton Civil Eng. AMEC	Survey ** (see table 4.2)			P. Geeson ** Assoc. Dir. Arup Nott.	
Seminar at AMEC:				Seminar at ** AMEC:			I. Hedges ** Mech. eng. AMEC	K. ** Armstrong Civ. eng AMEC
19 attendees AMEC, Arup , CIBSE				19 attendees AMEC, Arup , CIBSE			A. Newton *** Civ/str. eng. AMEC	

Table 4.1 Industry Involvement Throughout the Course of the Research

* Provided data to <u>build</u> the model and was involved in verifying the model

** Involved in validating the model

*** Provided data to <u>run</u> the model and was involved in validating the model

Name	Organisation	Position	Background
B. Clifford	Arup	Assoc. Director	Design
C. Evans	Arup	Associate	Design, Civil
Ms. M. Whild	Arup	Arch. sub-consult.	Arch., design manag.
D. Webley	AMEC	Group Manager	Mech., Design of process eng. projects
D. Starr	AMEC	Project Manager	Design, Design&Build, labs
M. Murphy	AMEC	Principal Arch.	Arch., build.design, refurb.,labor.
O. Vickery	AMEC	Principal Mech. engineer	Building services, design leader
J. French	AMEC	Principal Architect	Arch., Pharmac. projects
A. Robertson	Kyle Stewart	Associate	Public health eng., design manager, pharmac. projects
J. Dixon	Kyle Stewart	Principal Arch. Assistant	Arch. design, site manager
J. Cunliffe	Kyle Stewart	Assoc., Project design manager	Civil & structures
D. Carlisle	Kyle Stewart	Director	Civil engineer,design manager

Table 4.2 List of interviewees

A generic model for the Conceptual and Schematic stages of design was developed using the structured analysis diagramming technique of Data Flow Diagrams. The data used to build the model was based on historical review of design projects and observations of live projects. The developed model was verified by review of the industry professionals who were involved in these projects. Validation of the model was carried out by exposure to other industry representatives and discussions relating to their projects and their experience. Further validation was undertaken by exposure to new projects.

The developed Data Flow Model was analysed at the level of the functional primitive tasks using matrix manipulation techniques of the Design Structure Matrix. The objective of this analysis was to identify loops of iterative design tasks.

A simulation model for the design process was developed using the Three Phase Approach for Discrete Event Simulation by means of a simulation modelling environment. Data from the Data Flow Model and the Design Structure Matrix represented the 'front end' to the developed simulation model. The features of the simulation model were identified from industry survey and interviews. The model was designed to accept data from data flow diagrams after identifying loops of iterative tasks (if any). The simulation model was initially produced at a prototype level which was verified continuously using test data. The model was then extended with enhancements to functionality. Verification of the full scale model was undertaken by running sample data from data flow diagrams of the design process. Other data from a model of the estimating and tendering process in another research were also used to verify the simulation model. The developed simulation model was validated by use on an on-going design project 'shadowed' by the writer. However, the writer was unable to apply the model throughout the full design period due to the project circumstances and time scales for the research programme. Therefore validation was undertaken by industry feedback through demonstrations and discussions held by the writer followed by completing a 'feedback document' by each industry representative.

The following sections describe the development of the Generic Model for the Conceptual and Schematic stages of design, the technique used to identify loops of iterative design tasks, and the development of the simulation model for the design process.

4.2 USING DATA FLOW DIAGRAMS TO DEVELOP A GENERIC DATA FLOW MODEL FOR THE CONCEPTUAL / SCHEMATIC DESIGN STAGES

4.2.1 Justification of using Data Flow Diagrams

The literature review of the design process described in Chapters 2 and 3 emphasises the importance of communications and information transfer issues as key factors for successful design management. This was the reason for the writer directing his attention to structured analysis diagramming techniques where information flows between processes could be modelled. After a thorough examination of the structured analysis diagramming techniques described in chapter 3, the writer decided to use

Data Flow Diagrams for modelling the information transfer during the design process for the following reasons:

(i) They are a specialised tool for modelling data flows among systems.

(ii) They are user-friendly and easy to read, learn and use.

(iii) They may be used to model the flow of data from the context level to the detail level of the primitive functions of a system.

(iv) Levelled data flow diagrams allow managers to restrict their reading to the top few levels and still get the overall picture, while designers reading from the abstract to the detailed levels narrowing in on particular areas of interest.

(v) DFDs are produced and read in conjunction with the data dictionary and processes micro-specifications for more human comprehension. Data flow structures or elements and descriptions of processes which could not be shown on the diagram are cross-referenced with the data dictionary.

(vi) DFDs components: processes, data flows, data stores and data sources or sinks are suitable to be used in modelling information transfer during the building design process. Processes may represent different design tasks, data flows may represent information exchange, data stores may represent standards, specifications or any design files and data sources or sinks may represent external entities or organisations such as Client, contractor, local authorities, etc.

(vii) The data flow modelling technique is a valid, proven, well established technique in the field of building design and construction research. This is confirmed through previous work by Newton (1995), Hanby (1993), Fisher (1992), and Gharib (1991) described in chapter 3.

(viii) The existence of CASE (Computer Aided Software Engineering) tools have overcome problems of balancing rules and facilitated drawing DFDs and using them as a convenient modelling tool.

(ix) Each level of a levelled data flow diagram can be conveniently restricted to a standard A4 sheet size of paper which facilitates the distribution and review of the model by industry representatives.

4.2.2 The software used to build the Data Flow Models

One of the reasons which encouraged the writer to adopt data flow modelling techniques was the existence of CASE tools. CASE tools (Computer Aided Software Engineering) are a new breed of graphics-oriented micro-computer-based software tools. A CASE tool is defined as any software tool that provides automated assistance for software development, maintenance or project management activities (Byte, April 1989). CASE tools for structured systems analysis and design vary from a simple PC run software supporting structured methodologies like Demarco or Gane/Sarson to sophisticated work station based software used to model real time embedded systems. The basic features of CASE tools are:

(i) Checking balancing and consistency between parent and children diagrams and throughout the whole diagram with the facility of adding consistent flows from parent to child automatically.

(ii) Creating diagrams and inputting data easily.

(iii) Decomposition of the diagrams to the required number of levels.

(iv) Grouping of data.

(v) Zooming facilities. (i.e. the ability to view the diagram as a whole or to enlarge and analyse specific areas of the diagrams)

(vi) Carrying out any changes in the diagram updates the whole model automatically.

(vii) Well structured comprehensive data dictionary which can be updated automatically when updating any component in the diagram.

(viii) Comprehensive reporting facilities with the possibility of customising any report.

The capabilities of CASE tools have even been extended to generate code in other programming languages like C, COBOL, dbase, Paradox and others.

However, from the software review conducted by the writer, contacts with software companies and contacts with experts in the field of simulation and structured analysis, it was concluded that it was unlikely that a commercial software product would fulfil the research requirements for both the data flow modelling and simulation aspects. Therefore it was decided to carry out the analysis in four stages:

(i) Construct data flow diagrams using a CASE tool.

(ii) Extract information from the data flow model to be input for the simulation.

(iii) Extract information from the data flow model to be arranged in a matrix format for the purpose of identifying loops of iterative design tasks.

(iv) Transfer the revised data into a simulation model.

At the early stages of this research the SELECT CASE tool was available on the University campus. It was used to build and analyse the data flow diagrams for the first Case Study. It proved to be easy to use and had the basic features of a CASE tool. However, it was found to be unsuitable for large models and reports from the data dictionary could not be customised. Therefore it was decided to acquire the System Architect CASE tool which creates a separate encyclopaedia for each developed model and has fulfilled most of the requirements. System Architect had been used to construct the data flow models for the rest of the research.

4.2.3 The development and the validation process

Two case studies were undertaken by the writer to form a basis of a Generic Data Flow Model for the Conceptual and Schematic Design Stages. The first case study was based on historical data from a design project and was undertaken in collaboration with Ove Arup and Partners; Nottingham Branch. The second case study was based on data from a live project where the writer acted as an observer/recorder during design meetings and was undertaken in collaboration with Ove Arup and Partners; Birmingham Branch. The data flow models produced within the two case studies were verified through interviews held with the design engineers, architects and design managers involved with the studied projects. (A list of the interviewees is included within Table 4.2.) This resulted in some refinements and suggestions which have been incorporated into the models. The two case studies provided the writer with the basis and experience to produce a Generic Data Flow Model for the Conceptual and Schematic Design Stages.

The production of the Generic Model was an iterative process. The initial version of the model was produced based on the two case studies. This followed validation of the model through interviews with industry professionals within Ove Arup and Partners, AMEC Design and Management, John Laing and Fisons. This resulted in some refinements and suggestions which were incorporated into the model. The writer also presented the model together with the overall research objectives during a seminar held within AMEC Design and Management and attended by senior managers from AMEC, Ove Arup and Partners and CIBSE. Feedback from the attendees reflected the need of industry professionals for a better understanding of the Conceptual and Schematic stages of design and the lack of sophisticated tools for design management. This reaction confirmed the contribution that this research would provide to the industry. Additionally, a survey followed by subsequent interviews were conducted over twenty construction professionals within three major construction organisations namely Kyle Stewart, AMEC Design and Management and Ove Arup and Partners. The objectives of the survey and interviews were to validate the developed model and to identify the features required for the proposed simulation model. The results showed that the model being for the Conceptual and Schematic Design Stages is independent of the procurement strategy. It showed also that the Schematic design stage is more difficult to manage than the Detailed design stage which highlights the value of the model. More details about the results are found in section 4.4 and in Appendix II.

The developed Generic Data Flow Model was also validated against the actual design process during a case study undertaken by the writer in collaboration with AMEC Design and Management as designers and Loughborough University of Technology as the Client. The data collected included notes and observations recorded by the writer when 'shadowing' the meetings held between AMEC and the different Client committees and user groups in addition to minutes of meetings. The information exchanged was recorded and then categorised within different headings and allocated to at least one of the information flows on the Generic Model. Additionally, further validation of the model was undertaken through interviews conducted by the writer with the design leader. This confirmed that the model represented the design process subject to minor adjustments due to the special nature of the project. Details of this case study are included in Chapter 7.

4.3 USING THE DESIGN STRUCTURE MATRIX TO IDENTIFY LOOPS OF ITERATIVE DESIGN TASKS

After the development of a data flow model for the Conceptual/Schematic design stages, the next step was to identify the loops of iterative design tasks; one of the characteristic features of design. This is done at the level of the functional primitive tasks (FPTs) of the data flow model. The Design Structure Matrix (DSM) originally developed by Steward (1981, 1991) and furtherly developed by Eppinger and his research team at MIT (Eppinger and Eppinger et al 1990, 1991, McCord and Eppinger, 1993, Pimmler and Eppinger, 1994, Smith and Eppinger, 1995) has been used to perform this function. The functional primitive tasks of the data flow model are arranged in a square matrix where marks in the cells represent the information dependencies between different design tasks. This technique and its applications have been described in sections 2.2.6 and 2.2.7. Matrix analysis techniques were applied to re-order and 'partition' the matrix and identify the loops of iterative design tasks. A software program provided by Steward based on matrix manipulation algorithms was used to analyse the Design Structure Matrix. Data from the Data Flow Model and the loops identified from the DSM constitute the 'front end' of the developed Simulation Model.

4.4 SURVEY AND INTERVIEWS TO IDENTIFY THE MAIN FEATURES OF THE SIMULATION MODEL AND TO VALIDATE THE GENERIC DATA FLOW MODEL

A survey supported by subsequent interviews has been conducted by the writer on professional staff within three major construction organisations in the UK; namely Ove Arup and Partners, AMEC Design and Management, and Kyle Stewart. A survey document was issued to a total of twenty construction professionals with different disciplinary backgrounds and managerial responsibilities, twelve of whom were subsequently interviewed. (A list of the interviewees is provided in Table 4.2.) A copy of the survey document together with the full results are included in Appendix II. The main aims of the study was to identify the main features required to be incorporated in the simulation model and to acquire feedback from the industry professionals on the developed Generic Data Flow Model. The objectives of the study were:

(i) to identify the most appropriate design stage to simulate;

(ii) to assess the impact of the procurement strategy on the Generic Data Flow Model;

(iii) to identify the difficulties encountering design managers during the Conceptual/Schematic design stages;

(iv) to investigate measures for information quality;

(v) to identify communication problems during the Conceptual/Schematic design stages;

(vi) to investigate different means for information exchange;

(vii) to assess the potential benefits that could be drawn from using DFDs as a managerial tool; and

(viii) to identify the most important design tasks and information flows during the Conceptual/Schematic design stages represented in the Generic Data Flow Model.

The full results of the study are included in Appendix II. The main conclusions that were drawn are:

(i) The management of the Conceptual/Schematic design stages is more difficult than the detailed design stage. However, further investigation is required to assess the benefits of simulating each of these stages. (This investigation has been carried out during the case study described in Chapter 7.)

(ii) There is no fundamental impact of the procurement strategy on the produced Generic Data Flow Model for the Conceptual/Schematic design stages. Therefore, the model is valid for all types of procurement.

(iii) Difficulty in performing design tasks and/or obtaining design information and the importance of some information vary according to the background and discipline of the designer or the design manager involved.

(iv) Required elements of the design brief should be elicited by design staff in a structured way. The data dictionary of data flow models may be used to identify these elements according to the nature of each project.

(v) Communication problems experienced during the design process may lead to 'gate keeping' (withholding) of information among participants intentionally or non intentionally.

(vi) There is no formal way to judge the quality of information exchanged. The measure of information quality varies according to the sender and the recipient of information. Missing information or information of insufficient quality from the recipients point of view are supplemented by assumptions.

(vii) Electronic information exchange provides fast effective data exchange. Such means of data exchange have not yet been fully exploited by the construction industry.

(viii) Allocating appropriate resources and efficient resource utilisation is directly proportional to the efficient management of the design process.

(ix) Data flow models provide a useful effective tool which may be used to improve communications during the design process, and hence improve the management of the process. These models assist in identifying information requirements for different design tasks, and in identifying other designers' problems. They may be used in the training of engineers and architects.

4.5 IMPLICATIONS ON THE SIMULATION WORK

One of the objectives of the survey and interviews was to identify typical events that occur during the design process which will be beneficial for the design manager to simulate. By introducing durations and resources to the functional primitive tasks of the Generic Data Flow Model, data from the model may be manipulated to construct a simulation model. Running the simulation under different criteria allows the design manager to assess the impact of different events on the whole design process. The typical events identified from the conclusions drawn from the survey and interviews which represent the main features of the simulation model are:

(i) The variation of the quality of information exchanged between different design tasks.

(ii) Performing a design task based on assumed data inputs.

(iii) Changes in design information

(iv) The problem of missing information

(v) Releasing the information from different design tasks in packages.

(vi) 'Gate keeping' of information among design team members.

(vii) Resources allocation and assessment of their utilisation throughout the whole design process.

Feedback from industry professionals showed that although the above mentioned events are valid for all the stages of design, the importance of their associated problems may vary at the conceptual/schematic stages as opposed to the detailed design stage. This is described in more detail in Chapter 8.

A full description of the features, components and operation of the Design Simulation Model is included in Chapter 6.

4.6 USING THE THREE PHASE APPROACH FOR DISCRETE EVENT SIMULATION TO DEVELOP A SIMULATION MODEL FOR THE DESIGN PROCESS BY MEANS OF THE SIMULATION MODELLING ENVIRONMENT GENETIK

4.6.1 The justification for this approach

After the development of the Generic Data Flow Model, the identification of the loops of iterative design tasks using the Design Structure Matrix, and the identification of the typical events required to be simulated during the design process; the next step was to use this data to develop a simulation model. After a thorough investigation of the different simulation techniques for simulating the design process, (as presented in section 3.3), the writer decided to use the *Three Phase Approach* to *discrete event* *simulation* using a *simulation modelling environment.* This was primarily for the following reasons:

(i) *Discrete event simulation* provides a 'transparent' approach to simulate the design process under different information related criteria. It allows the user to monitor instantaneously the changes in different design tasks and/or resources as the simulation clock advances. It allows also interaction with the model. (Discrete Event Simulation techniques are described in section 3.3.)

(ii) *The Three Phase Approach* to discrete event simulation has been used to develop the simulation model. This is due to the suitability of the B-events and the C-events associated with this approach for building a simulation model of the design process. The C-events would represent the events which start design tasks on fulfilment of different conditions like completion of predecessor tasks, availability of necessary information, resources availability, etc. The B-events would represent events associated with the completion of scheduled design tasks with the specified durations such as the change in the state of the task from on-going to completed or change in the state of resources from busy to idle or vice-versa. (Details of the Design Simulation Model developed by the writer are found in Chapter 6.)

(iii) A *simulation modelling environment*, named 'Genetik' which supports the Three Phase Approach to discrete event simulation has been used to construct the simulation model. This decision has been made because there was no 'off the shelf' data driven simulation packages which could build simulation models directly from data flow diagrams. Therefore there was a need to use a simulation environment which provided the necessary flexibility required for the modelling aspect and allowed incorporating certain rules within the structure of the model. There was the trade-off of considerable programming effort to write routines in its own syntax. However, the programming effort is less than that would have been involved if the simulation model has been constructed using simulation programming languages or general purpose languages.

4.6.2 The Development and the Validation Process

A decision was made by the writer to build the Simulation Model on an incremental basis, starting with a basic model followed by expanding that model to incorporate the required features which were identified from the conclusions of the survey and interviews described in section 4.4. This decision was made to allow testing the model after each step in its development and hence facilitate debugging any errors.

The simulation model is based on information from the Generic Data Flow Model. Therefore it was decided to start building the simulation model based on a prototype data flow model which includes all the features and components of a full scale model. This procedure allowed appending different features to the Simulation model as the research progressed and testing the model with sample data on incremental basis. After rigorous and thorough testing of the model on the prototype level, the full scale Simulation Model was constructed. Sample data were input into the Genetik Simulation Model to test and verify the routines developed and the operation of the model. This was undertaken on an incremental basis with manual checks of the output to ensure the accuracy and robustness of the model.

The simulation model was designed to be independent of specific data. The data required should be captured from a data flow model and a design structure matrix and input in the relevant data entry tables of the simulation model. This is irrespective of a particular data flow model or a design structure matrix. Hence, the model is generic and may be used to simulate any process represented by a data flow model with minor adjustments. Therefore, although the Genetik Simulation Model has been developed to simulate the Conceptual/Schematic stages of the design process, the way the model has been set up allows extending its application to the detailed design and construction stages with minor adjustments. The feasibility of extending the application of the Simulation Model to the detailed design stage was confirmed through feedback from industry professionals. This is explained in more detail in Chapter 8. Examples for using the simulation model to simulate typical events during the detailed design stage (in addition to the Conceptual/Schematic design stages) are also included in Chapter 8.

Data collected during the case study undertaken by the writer in collaboration with AMEC Design and Management, (designers), and Loughborough University, (the Client), during the Conceptual Design Stage of the University Engineering Complex have been used to run the simulation model. Forms have been produced by the writer and given to the design leader (to issue them in turn to different members of the design team) to record the information exchanged during performing different design tasks and the action taken upon receipt of information to assess the quality of information. The design leader was asked to provide information about the durations and resources of the functional primitive tasks identified in the data flow model. This is explained in more detail in Chapter 7.

Data from the case studies have been used to set up examples illustrating the different features of the Simulation Model and to demonstrate the areas where the developed Generic Data Flow Model and Simulation Model may be used to improve the management of the design process. This is described in detail in Chapter 8.

The Simulation Model was validated through demonstrations held by the writer to industry professionals within Ove Arup and Partners, Nottingham branch; Ove Arup and Partners, Birmingham branch and AMEC Design and Management. The objective of the demonstrations was to acquire feedback on the contribution that the application of the simulation model will offer to improve the management of the design process. A total of 10 demonstrations have been undertaken and a feedback document was issued to each attendee at the end of every demonstration. A copy of the feedback document is included in Appendix X. The responses showed the importance of the problems experienced during managing the design process and which are reflected in the features of the developed simulation model. They showed also the suitability of the developed tools to provide the solution to these problems and that the application of these tools will help to improve design management. More details are provided in Chapters 6 and 8.

The following chapters describe the experimental work undertaken by the writer. Chapter 5 describes the Generic Data Flow Model for the Conceptual/Schematic design stages, its development process, the two case studies that preceded the model development and the validation of the developed models. The developed Simulation Model, the operation of the model and testing and verification of the model is described in chapter 6. Chapter 7 describes a case study undertaken by the writer to evaluate and validate the developed tools. Examples based on data from the case studies, the benefits the developed tools provide to improve the management of the design process, feedback from the industry and the feasibility of extending such tools for application at the detailed design stage are described fully in chapter 8.

CHAPTER 5

A GENERIC MODEL FOR THE CONCEPTUAL AND SCHEMATIC DESIGN STAGES

5.1 INTRODUCTION

This chapter describes the development of a Generic Model for the Conceptual and Schematic stages of design, built using Data Flow Diagrams, with a proprietary CASE (Computer Aided Software Engineering) tool. Two case studies were undertaken by the writer to form the basis of the Generic Data Flow Model. The first case study was based on historical data from a design project and was undertaken in collaboration with Ove Arup and Partners; Nottingham Branch. The second case study was based on data from a live project where the writer acted as an observer/recorder during design meetings and was undertaken in collaboration with Ove Arup and Partners; Birmingham Branch. The data flow models produced within the two case studies were verified through interviews held with the design engineers, architects and design managers involved with the studied projects. This resulted in some refinements and suggestions which have been incorporated into the models. The two case studies provided the writer with the basis and experience to produce a Generic Data Flow Model for the Conceptual and Schematic design stages.

The production of the Generic Model was an iterative process where continuous validation was sought through feedback and interviews with industry professionals. The developed Model was also validated against the actual design process during a case study undertaken by the writer in collaboration with AMEC Design and Management as designers and Loughborough University of Technology as the Client. The validation of the model was an on going process until the production of the final version of the model.

The two preliminary case studies and the development and validation of the Generic Data Flow Model are described in this chapter.

5.2 CASE STUDY 1: A DATA FLOW MODEL FOR A POWER PLANT DESIGN IN COLLABORATION WITH OVE ARUP & PARTNERS; NOTTINGHAM BRANCH

5.2.1 Background to the Project

Corby Power Station is a 350 MW power station, owned and operated by Corby Power Limited which is a joint venture between East Midlands Electricity, Hawker Siddeley Power Engineering and ESB. For construction, Corby Power Limited have appointed Ewbank Preece as their Engineer. Ove Arup and Partners (Nottingham offices) as Consulting Engineers with their sub-consultant Bartlett Gray and Partners as Architect were first commissioned to prepare the environmental statement, planning application and later they were responsible for all stages of civil engineering and building works design. Hawker Siddeley Power Engineering was the turnkey contractor for the project and later on, they employed Kier Construction as the design and build contractor and since that time, Ove Arup and Partners were performing all civil engineering and building works design for Kier Construction. A separate reclamation contract was completed by Balfour Beatty Civil Engineers with Ove Arup and Partners as the Engineer. Electro-mechanical contractors were appointed directly by Hawker Siddeley Power Engineering.

A schematic diagram showing the role of different parties is shown in Figure 5.1.

5.2.2 Aims and Objectives of the Case Study

The main aim of this case study was to investigate the use of data flow diagrams as a method for modelling the information flows during the early stages of the design process.

The objectives of the study were:

(i) To establish a working relationship with Arup's Nottingham staff.

Since this case study was the first one in the course of the research, it was essential to establish a working relationship with Arup Nottingham office through producing a data flow model from the information they provided, and validating the model through discussions with their professional staff and monitoring a future similar project.

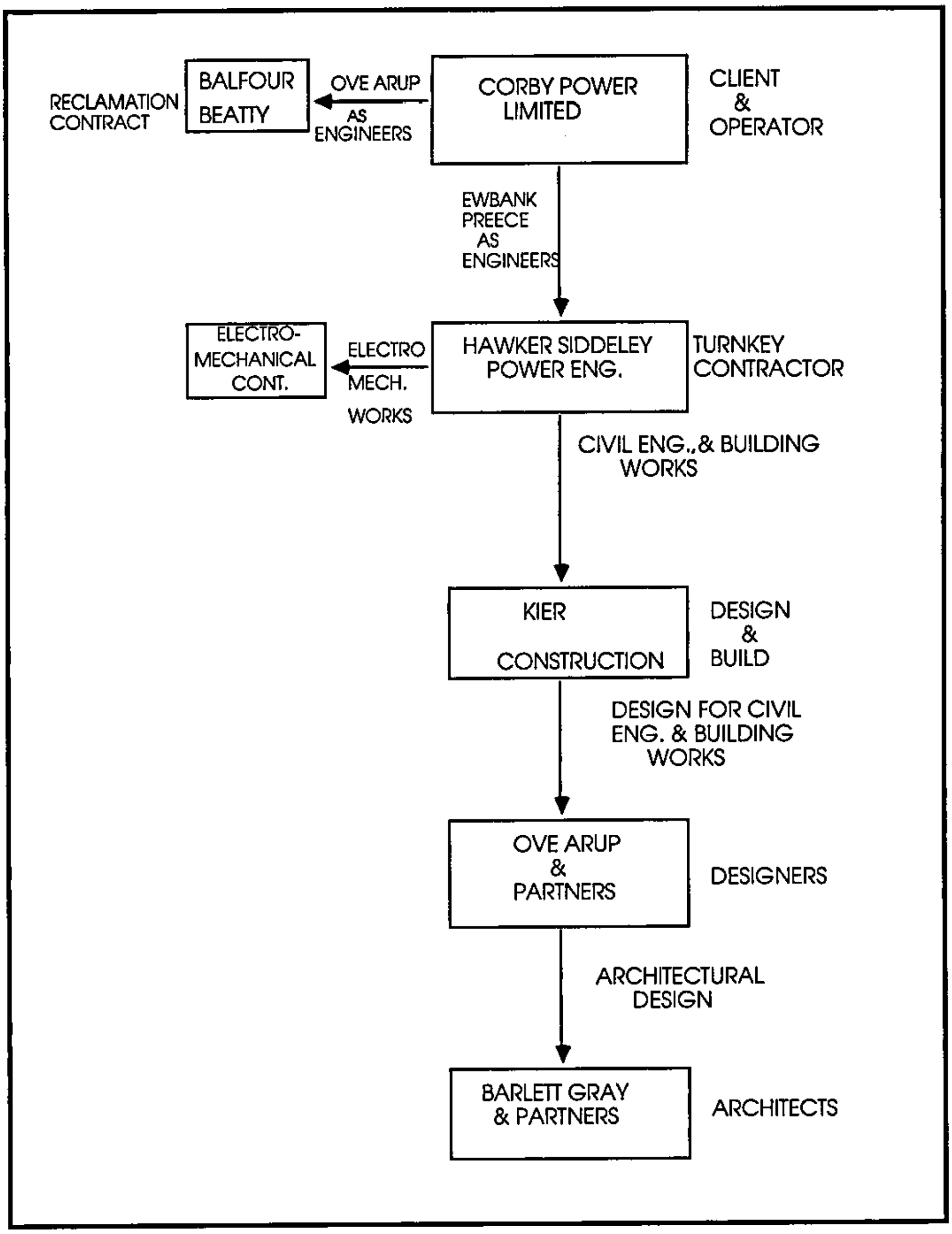


Figure 5.1 Different parties involved during the design of Corby Power Station

(ii) To assess the use of historical data to build data flow models

It was essential to decide on the suitability of the use of historical data to construct data flow models or should live ongoing projects be monitored.

(iii) To assess the usefulness of project documentation in identifying the information required to build data flow diagrams

(iv) To assess a suitable approach for decomposition of higher level design tasks
It was essential to assess and validate the most suitable approach to decompose higher level design tasks. These could be either decomposed to project elements or to project disciplines.

5.2.3 Sources of Information

At the time Ove Arup and Partners, as one of the collaborating companies to the research, provided access to information about the design process, the design and construction works had already finished and the plant was about to start operation. Therefore, the source of information was from historical documents which had to be borrowed from their office, photocopied and returned back.

The information collected covered only the works carried out by Ove Arup. There was no formal project brief produced by the Client, but there was a 'Scope of Work Document' produced by Arups which was a form of proposal for design works that reflect the Client's requirements.

The documents collected included:

- Geotechnical Report
- Environmental Statement
- Scope of Work Document
- Progress Reports
- Minutes of Meetings
- Programmes
- Quality Records
- Team Responsibilities
- History of events effecting the design programme of turbine hall and water treatment annexe which were prepared for the purposes of a claim
- Preliminary Design Document

Some information needed to be acquired from members of the design team, but due to the fact that it was a historic project, most of the design team were allocated to other projects in other branches. The only remaining members were the project manager for a certain period of the project and the structural engineer, who were interviewed. As the architectural works were sub-consulted, there was little information relating to architectural design works. There was also no information about electro-mechanical design works as these were carried out by other contractors appointed directly by the turnkey design and build contractor.

The historical information also did not show exactly the information flow or which disciplines were suspended or awaiting information from other disciplines.

5.2.4 The Data Flow Model

Ove Arup's scope of work included some 40 elements in the design between functional plant buildings, auxiliary buildings like administration buildings and workshops, structural supports for machines and civil and external works. Due to the constraints on the information available as previously discussed, it was decided to decompose the model in the 'path' of the plant buildings, and further decompose it in the 'path' of Functional Plant Buildings where Arup were the leading designers as the architectural input for these buildings was minimum. (For architectural input to the auxiliary buildings the sub-consultant architect was the leading designer). The data flow model was constructed, and demonstrated to Ove Arup who verified the model after requesting some refinements. These were incorporated in the model. Arup also indicated that the model should be suitable for application to any industrial project.

The data flow model, including Arup's suggestions together with a sample report from the data dictionary are presented in Figure 5.2.

5.2.5 Conclusions Drawn From the Study

From the constructed data flow model, and based on observations of the writer supported by statements from Arup's professional staff, the following was concluded:

(i) Data flow diagrams are a useful technique for information transfer representation and can be easily understood. All the personnel at Arup's Nottingham office who were involved in commenting on the data flow diagrams had no prior instruction in the use of the technique. Within a short period of time however, they were conversant with the principles to the extent of being able to comment in a meaningful way.

(ii) Building data flow models from historical projects do not always reveal how and when the information was transferred and do not show which design tasks were suspended waiting for information from other tasks. Monitoring live ongoing projects is essential.

(iii) Although constructed for a power plant, the model was considered by the reviewers to be a generic representation that could be applied to other industrial projects in order to identify the information requirements for different design tasks.

(iv) Although the contractual situation relating to this project was contorted, the professional staff involved confirmed that such situations were not unusual for design and build contracts.

(v) As a result of this contractual situation and due to the organisational structure shown in Figure 5.1, it may be noticed in diagram 'O-Power Plant Design' that there is not direct information transfer between process '3': 'Civil, Structural and Arch. Design' and process '4': 'Electro-mechanical Design'. Arup's professional staff pointed out that this information link was in some instances necessary to proceed with their design without delay. This was the case especially in the design of the functional plant buildings where provision had to be made for equipment and machines.

(vi) It may be noticed from the 'Context Diagram' that 'Time and Cost Brief' information from the 'Turnkey design and build Contractor' to the 'Power Plant Design' process is of great importance and was shown on the Context Diagram, at the suggestion of the Arup staff. (This should not be confused with the 'Project Brief' information from the 'Client and Operator).

(vii) It was the task of the design team to acquire the statutory approvals for the Client from the statutory bodies and pass this information to the Client as shown in the 'Context Diagram' and diagram O - 'Power Plant Design'.

(viii) It was suggested by Arup to include a separate process 'Commercial assessment' which collects cost control information from different disciplines and passes the 'cost information' to the technical co-ordination as shown on diagram O - 'Power Plant Design'.

(ix) The dotted information flows of 'Approvals and Comments' and 'Statutory approvals' shown on the model represent control data flows which control the proceeding of the associated process as suggested by Ward and Mellor (1985)

(x) Arup preferred the decomposition of the model to the lowest level in terms of the project <u>elements</u> and not <u>disciplines</u> because this was the way they planned, followed up and cost controlled large projects. They preferred including the 'Architectural design of Plant Buildings' process in the lowest level as part of the decomposed 'Design of Functional Plant Buildings' process as shown on diagram 3.1.2 - 'Design of Functional Plant Buildings'.

(xi) Some tasks like 'load calculations' and 'design checks' were not included as separate processes in the lowest levels, as they were considered by Arup as parts of the different components of the decomposed design process.

(xii) From the collected documents, the most useful was the 'history of events affecting the design programme of turbine hall and water treatment annexe'. This document was prepared for the purpose of a claim for abortive work, and it recorded in a tabular and graphical form, types and dates for information received, information issued and information requested. Another useful document was the Scope of Works document which identified the information and assumptions on which Arup based their design. The least useful documents were the programmes which did not show any information transfer.

(xiii) The writer found that data flow diagrams are a simple and quick way of modelling information flows. However, as the model is decomposed to lower levels and due to balancing rules, it was difficult to model ongoing processes like the 'project management'. It was assumed that this task exists within all the levels of the model.

(xiv) As only designers and engineers will be dealing with the model, the process decomposition, data definitions and processes mini-specs should be at a reasonable level of comprehension. It is not expected to define some basic data like 'settlement tolerances' or 'columns layout'.

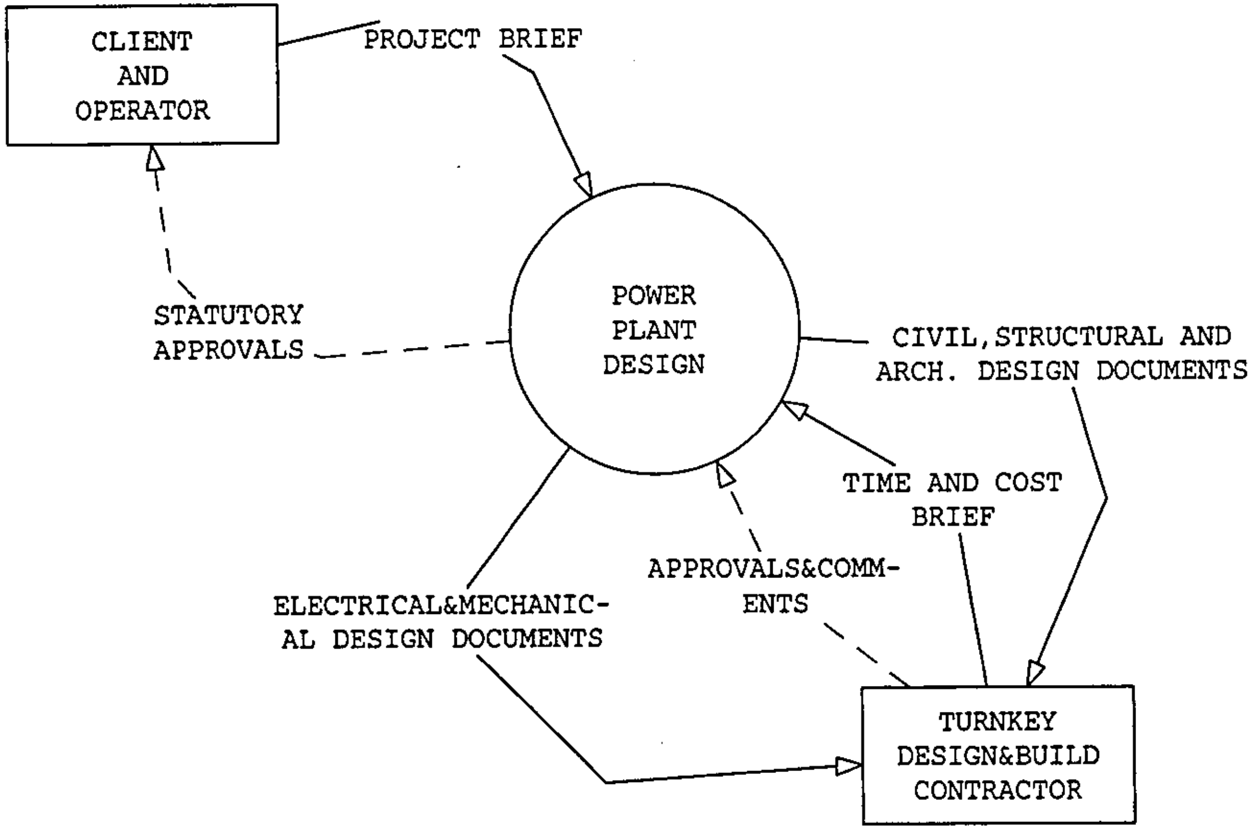
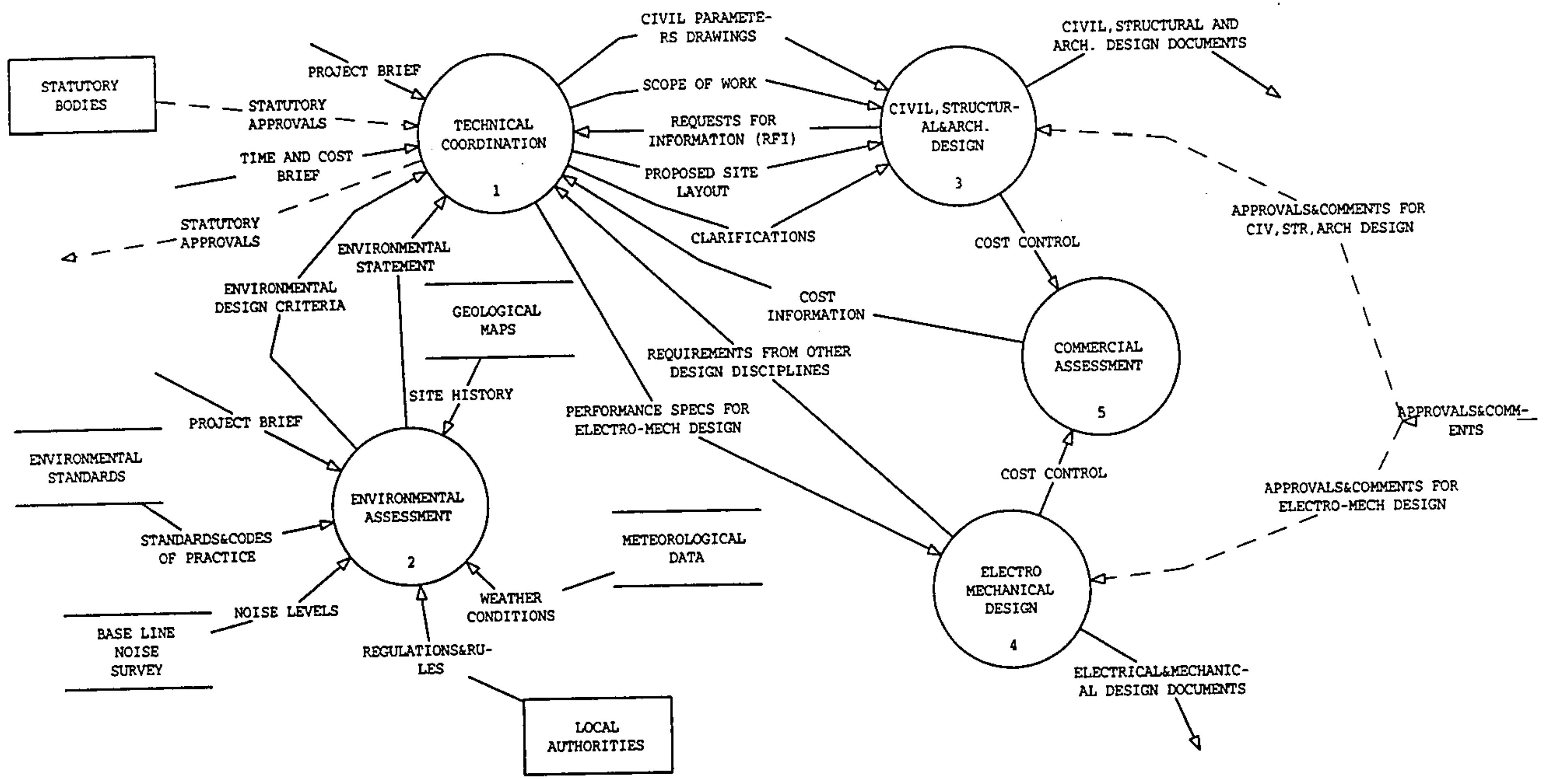
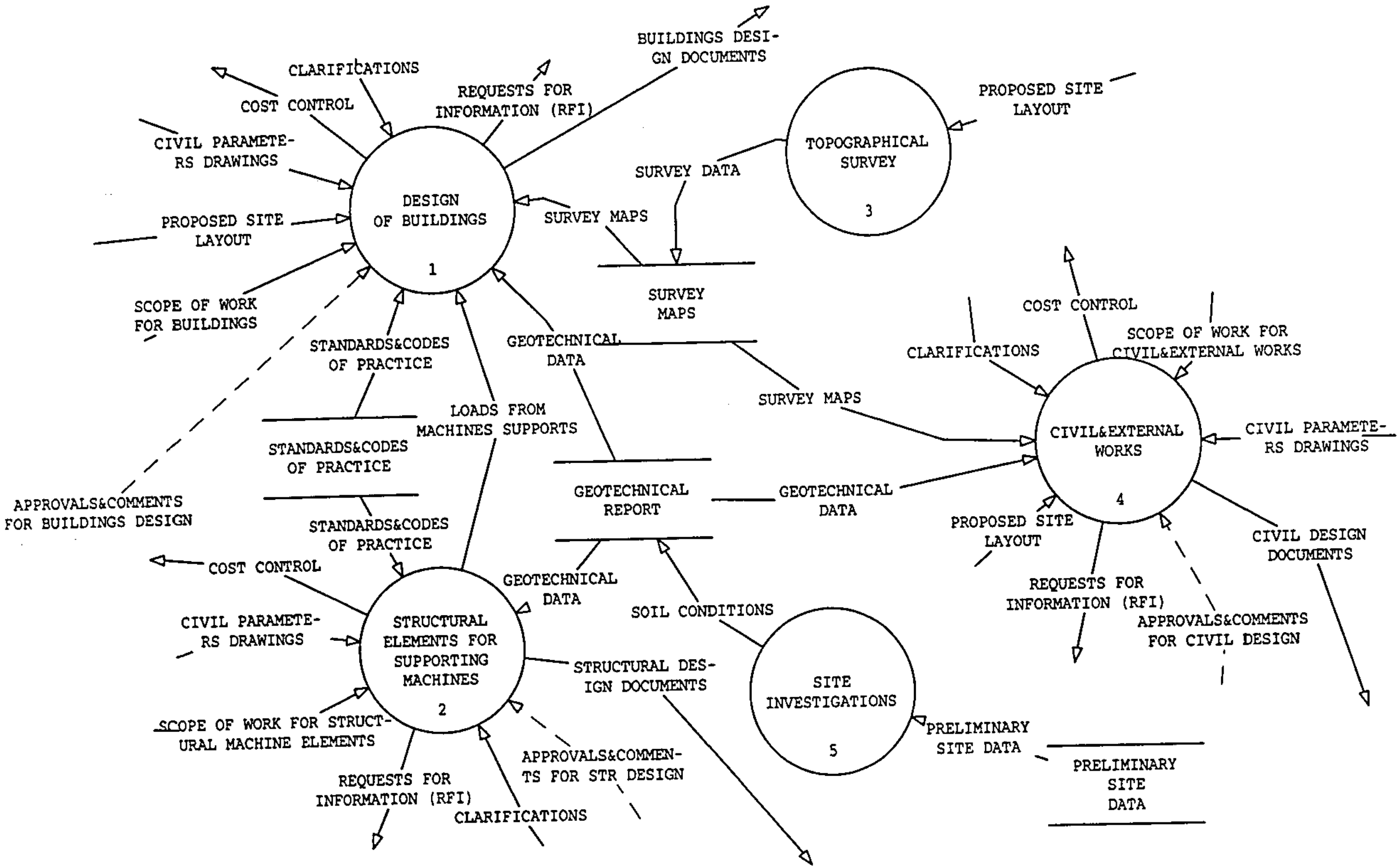


Figure 5.2 Data Flow Mod-
el for Power Plant Design

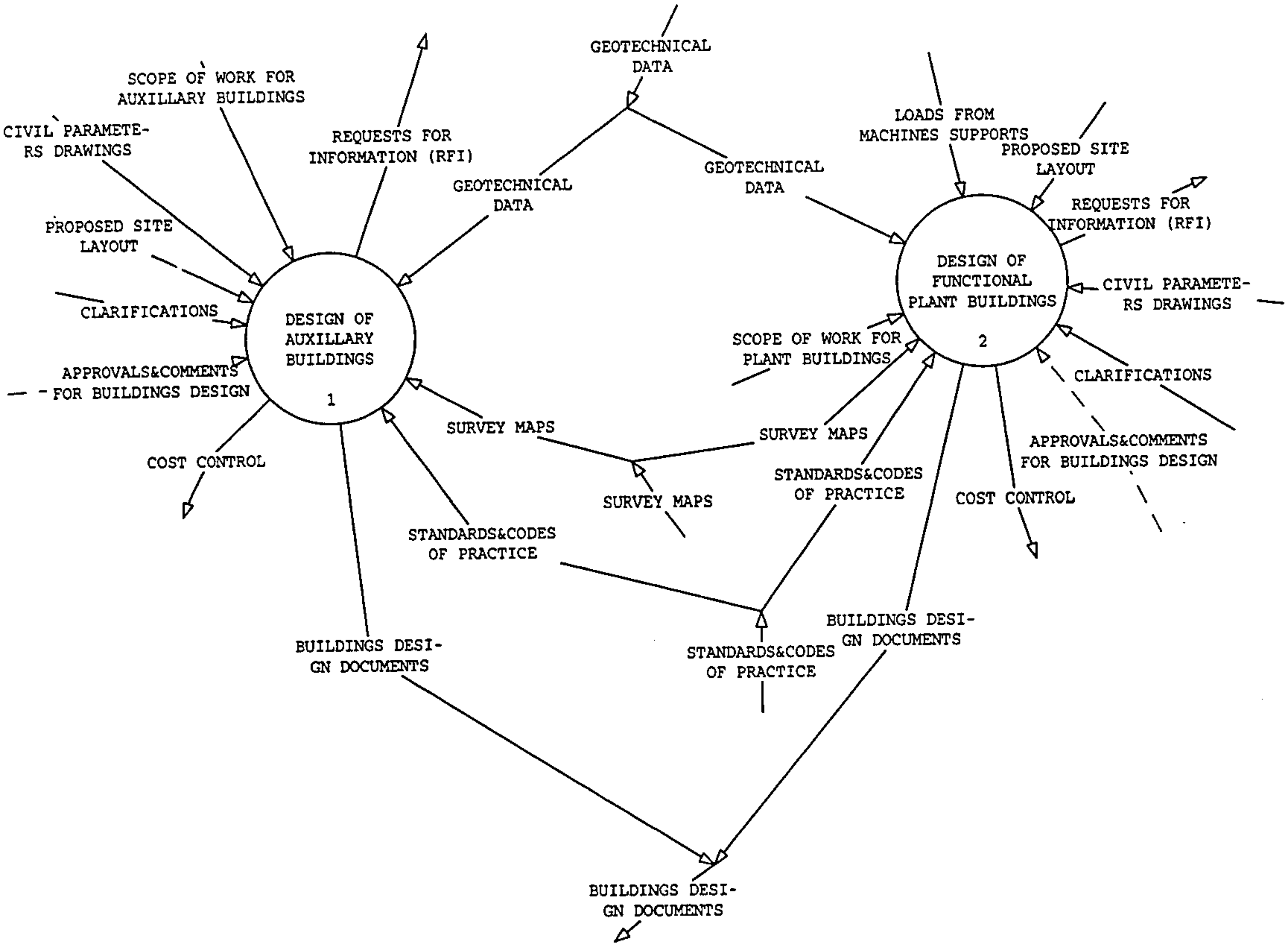
Context-Diagram - POWER PLANT DESIGN



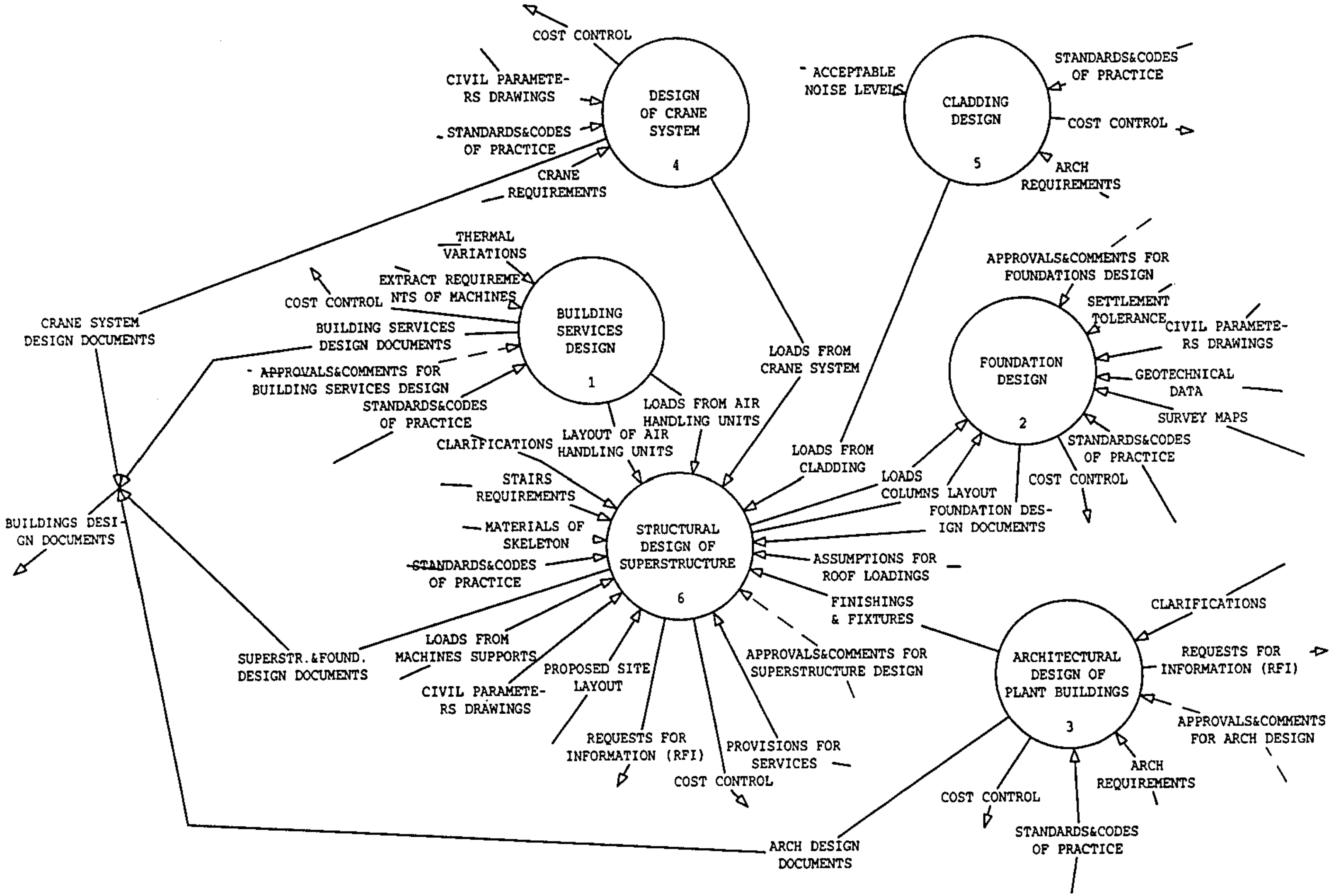
0 - POWER PLANT DESIGN



3 - CIVIL, STRUCTURAL&ARCH. DESIGN



3.1 - DESIGN OF BUILDINGS



3.1.2 - DESIGN OF FUNCTIONAL PLANT BUILDINGS

3.1.2.2 - FOUNDATION DESIGN

```
@IN  = APPROVALS&COMMENTS FOR FOUNDATIONS DESIGN
@IN  = CIVIL PARAMETERS DRAWINGS
@IN  = COLUMNS LAYOUT
@IN  = GEOTECHNICAL DATA
@IN  = LOADS
@IN  = SETTLEMENT TOLERANCE
@IN  = STANDARDS&CODES OF PRACTICE
@IN  = SURVEY MAPS
@OUT = FOUNDATION DESIGN DOCUMENTS
@OUT = COST CONTROL

@PSPEC FOUNDATION DESIGN
  Consider
GEOTECHNICAL DATA
Different types of LOADS
SETTLEMENT TOLERANCE
SURVEY MAPS
CIVIL PARAMETERS DRAWINGS
COLUMNS LAYOUT
LOADS from adjacent buildings
STANDARDS&CODES OF PRACTICE
  Decide a suitable type of foundation
  Case of pile foundations
Decide type of piles (friction piles, end bearing piles,..)
Calculate number of piles for every pile group
Calculate depth of piles
Design pile caps
Design ground beams
  Consider foundation alternatives
  Carryout a cost estimate and produce COST CONTROL data
  Issue FOUNDATIONS CALCULATIONS AND DRAWINGS for design checks
  Wait for
FEEDBACK
  Carry out necessary corrections
  Issue FOUNDATION DESIGN DOCUMENTS
  Wait for
APPROVALS&COMMENTS FOR FOUNDATIONS DESIGN
  Carry out necessary amendments
```

5.3 CASE STUDY 2: A DATA FLOW MODEL FOR A FACTORY DESIGN IN COLLABORATION WITH OVE ARUP & PARTNERS; BIRMINGHAM BRANCH

5.3.1 Background to the Project

Johnsons Controls is an American manufacturer for car seats covers supplying car seat covers to leading car manufacturers. They have branches in different countries of the world, the European branches being directed through their largest branch in Belgium. Ove Arup and Partners (Birmingham offices) used to be Project Managers and Designers for a series of Johnsons Controls factories in Europe. These projects followed a traditional procurement strategy. The design and construction of these projects used to be done with a very compact time schedule, as the Client usually gives Arup the date for the first production of the factory which coincides with their first supply commitment and all works have to be scheduled backwards. This was due to the fact that the Client did not want to buy land and freeze an asset for a long time before starting to supply products for customers. Due to timely negotiations with the local authorities and some strategic tactics from the Client, Arup had to proceed with the design work without confirmation of the site location. The site had to be assumed as one from the different options for the Client. During the design period, the actual site location was confirmed.

The architectural design works were sub-consulted, however, Arup's relationship with the architect was more as an 'in-house' architect than main consultant/sub-consultant relationship. As a way of fast tracking the project, it was decided that the steel work frame design should terminate early in order to be issued in a separate tender while other design works were not yet completed. This project was chosen for the following reasons:

(i) It was a live project which facilitated monitoring of the information transfer.

(ii) The period of the design works was relatively short, and

(iii) It was one of a series of almost similar projects which allows full validation of the model on a similar project at a later stage.

5.3.2 Aims and Objectives of the Project Study

The aims and objectives of the project study were:

(i) To assess the use and importance of live on-going projects to build data flow models.
A previous study by the writer concerning building data flow models to model the information flows during the design process for a historical project showed that historical data do not always reveal how and when the information was transferred and do not show which design tasks were suspended waiting for information from other tasks. Therefore it was important to monitor a live on-going project to assess the use and importance of live data to build data flow models.

(ii) To test and validate the constructed model on a similar project.
Since that project was one of a series of almost similar projects that Arup's Birmingham offices undertook the design works for the Client, it was a good opportunity to construct a data flow model and validate it on a similar project that starts afterwards.

(iii) To assess the use of data flow models in aiding and improving the management of the design process.
As one of the objectives was to validate the model on a future similar project, it would be of great importance to assess the areas where data flow models could aid and improve the management of the design process.

(iv) To investigate the observer/recorder technique as a method for information elicitation to build data flow models.

5.3.3 Sources of Information

Access to information occurred after Arup had submitted their scheme report, but before they started the developed design. Arup do not follow the RIBA plan of work or any other systematic model to break down the design stages. Their definitions for different design stages and the tasks involved in each stage differ according to the type of project, its complexity and the agreement undertaken with the Client on the deliverables. In this project, the design stages were classified as scheme design, developed design and detailed design.

Prior to this case study, it was decided, with the agreement of the project manager, that the writer would attend the weekly design meetings and act as an observer/ recorder. This would provide direct experience of all the discussions, issues raised, information required, and information issued by designers representing the different disciplines. No other interaction with members of the design team was proposed. This decision was taken in order not to cause any disruption to the design team, as the design programme was very tight. It was decided also for the same reason, not to conduct any interviews with the design team during the design phase. However, these interviews might be conducted, if necessary, after terminating the design. As the period of the pre-tender design stage was only four weeks, four design meetings were only attended.

Only the first meeting was attended by <u>all</u> the design team members. At the other meetings, some of the design team, mainly those responsible for the mechanical, electrical and drainage design, were busy with their design and/or their presence in the meetings was not absolutely essential. However, the architect was present in all the meetings as the architectural design works had a leading role in this project. The client was not present in any of the meetings. All meetings with the client were held with the project manager only.

There was little information available about the details of information transfer among the design team in the period between the weekly design meetings.

In addition to observed and recorded information from the weekly design meetings, other sources of information were:

- the scheme design report;
- the programmes for the design and construction work; and
- some faxed instructions by the client.

Due to the good relationship between the client and Arup, most of the client's requests or changes were agreed verbally or over the telephone and were not documented at any stage of the design process.

As with a previous study in collaboration with Arup Nottingham offices, there was no formal project brief produced by the client, however, the scheme report produced by Arup was considered as a 'benchmark' or 'checklist' for the main issues that reflected the client's requirements.

5.3.4 The Data Flow Model

As previously mentioned, access to weekly design meetings was provided after producing the scheme report and before starting the developed design. However, an initial meeting was held between the writer and the project manager during the scheme design. During that meeting, the project manager explained the background to the project and the ongoing scheme design process. There was a minimum interaction between the design team members during the scheme design as there were many uncertainties from the client side especially those concerning the site location and the negotiations with the local authorities, as mentioned in section (5.3.1) , which if it had failed, the whole project could have been aborted. Hence, Arup were cautious in risking overheads for the project at this stage.

Therefore, the scheme design stage was modelled, based on information from the project manager and from the scheme report. The model was decomposed to four levels to encompass the scheme and developed design stages, but was not further decomposed to show details of tender drawings production. The data flow model is presented in Figure 5.3. Samples from the data dictionary are included in Appendix III.

Diagrams 1.1 Scheme Design, 1.1.1 Scheme Building Design, 1.1.1.1 Scheme Arch Design and 1.1.1.2 Scheme Str Design were constructed based on information from the project manager, and from the scheme report.

Since the developed design was monitored as an ongoing process, the model was built gradually according to the writer's observations during the design meetings.

Diagram 1.2 Developed Design, the source 'Building Control Officer', process 1.2.6 'Internal Roads Design' and the associated data flows were not included until the second week. As the Client introduced late changes to the production and office area layouts, the data flow 'Final production and office area requirements' was added also after the second week. The need for some information for the 'Architectural Design' process from the 'Structural Design' like the 'bearing walls location' and 'level of brick walls' was not realised before the second week when they were added to the model.

By the third week, the nature of information flows became more detailed. For example 'manholes locations' and 'louvers locations' required for the 'Architectural Design' .

At the fourth week, there was a need to co-ordinate the specifications produced by different disciplines, especially in the areas where they overlapped, and consequently the process 'Specs Co-ordination' was added.

The decomposed diagrams 1.2.1 'Structural Design' and 1.2.3 'Architectural Design', the processes 'Footpaths Design', 'Ancillary Buildings Design', 'Landscaping Design' and their associated data flows were added after the second week.

After the final design meeting has been held, the constructed data flow model has been verified through interviews with professional staff involved within the case study. The interviews showed that the model produced represented the information flows between the design tasks during the design process. Minor refinements were suggested which were incorporated into the model. It was pointed out that having an 'In House' Architect for this project simplified the model. It would have been more complex if the Architect was a separate party sub-consultant.

5.3.5 Conclusions drawn from the Study

From the constructed data flow model, and based on observations of the writer, the following was concluded:

(i) Data flow diagrams are useful in identifying information which, although often appearing of low importance to the design staff, prove to be of great importance to other members of the design team. A typical example is the data flow 'Gutters Details' flowing from 'Architectural Design' to 'Structural Design' in diagram 1.2 'Developed Design'. This information although considered trivial to the architect in the early stage of developed design, is of great importance for the steel works design to proceed which was required to terminate in an early stage to issue an early steel work tender.

(ii) Diagram 1.2 'Developed Design' shows that there was special consideration for information required from the source 'Building Control Officer'. This information was the 'Building Regulations' required by the 'Architectural Design' process and 'Smoke Detection and Ventilation Requirements' required by the services design process. These regulations are described in the data dictionary. It was essential to have this information complete for the building regulation submission to be issued at the same date of tender issue. In this case, all requirements of the building control officer would have been fulfilled in advance.

(iii) Referring to process 1.2.7 'Specs Co-ordination' in diagram 1.2 'Developed Design', there was special emphasis during the design process on the co-ordination of specifications between different design disciplines. In Arup's previous projects, lack of information on the specifications produced by certain disciplines led to conflict in specifications produced by other disciplines. A typical example was in specifying the mechanical louvers where architectural and mechanical design disciplines were involved and in specifying footpaths materials where architectural and structural design disciplines were involved.

(iv) The data flow 'Final production and office areas requirements' from the source 'Client and operator' to the 'Architectural design process' appears in diagram 1.2 'Developed Design', as it is always assumed that the client introduces changes to these layouts, especially when they also operate the project. It is essential at the developed design stage that the client issues their final requirements for layouts and that they are 'frozen' at this stage so as not to affect the major design disciplines.

(v) Due to the fact that the scheme design proceeded based on an assumed site location, the input for process 1.1.1.2.1 'Scheme Foundations Design' was 'Assumed Ground Conditions' and consequently the output was 'Assumed Foundations' as shown in diagram 1.1.1.2 'Scheme Structural Design'.

(vi) Where information was required for a design task but was unavailable before tender, the design task proceeded and the relevant details were included later as an addendum to the tender. An example of this is the architectural roof sections required for process 1.2.1.1 'Steel Frame Design' as shown in diagram 1.2.1 'Structural Design'.

(vii) Diagram 1.2 'Developed Design' shows the importance of the role of the 'Architectural Design' for the project. It represents the 'centre' of information between the major design disciplines.

(viii) It was observed during the construction of the model that the main design tasks and information flows appeared in the early stages of constructing the model. The nature of the tasks and information flows added later to the model were related to details in the design process which appear in lower decomposed levels as previously described in section 5.3.4.

(ix) This study showed that constructing data flow models from live projects and attending design meetings as observer/recorder is more informative than in cases of historical projects.

(x) There is a need to be able to apply simulation techniques to data flow models to assess the impact of different factors that affect the information transfer and the timing of receipt of information. This appears particularly significant in cases of information required early in the design process such as in 'gutter details' previously mentioned in item (i), or in information required from the building control officer which affects considerations to be taken in architectural, structural and services design disciplines. This information affects the issue of building regulations submission which in the project described needed to be issued on the same date as the tender issue.

(xi) Although throughout the duration of this research a similar project was not available to monitor and verify the developed model, feedback from the design staff at Arup has indicated that the model will be of use in future projects as a tool to aid in the management of the design process.

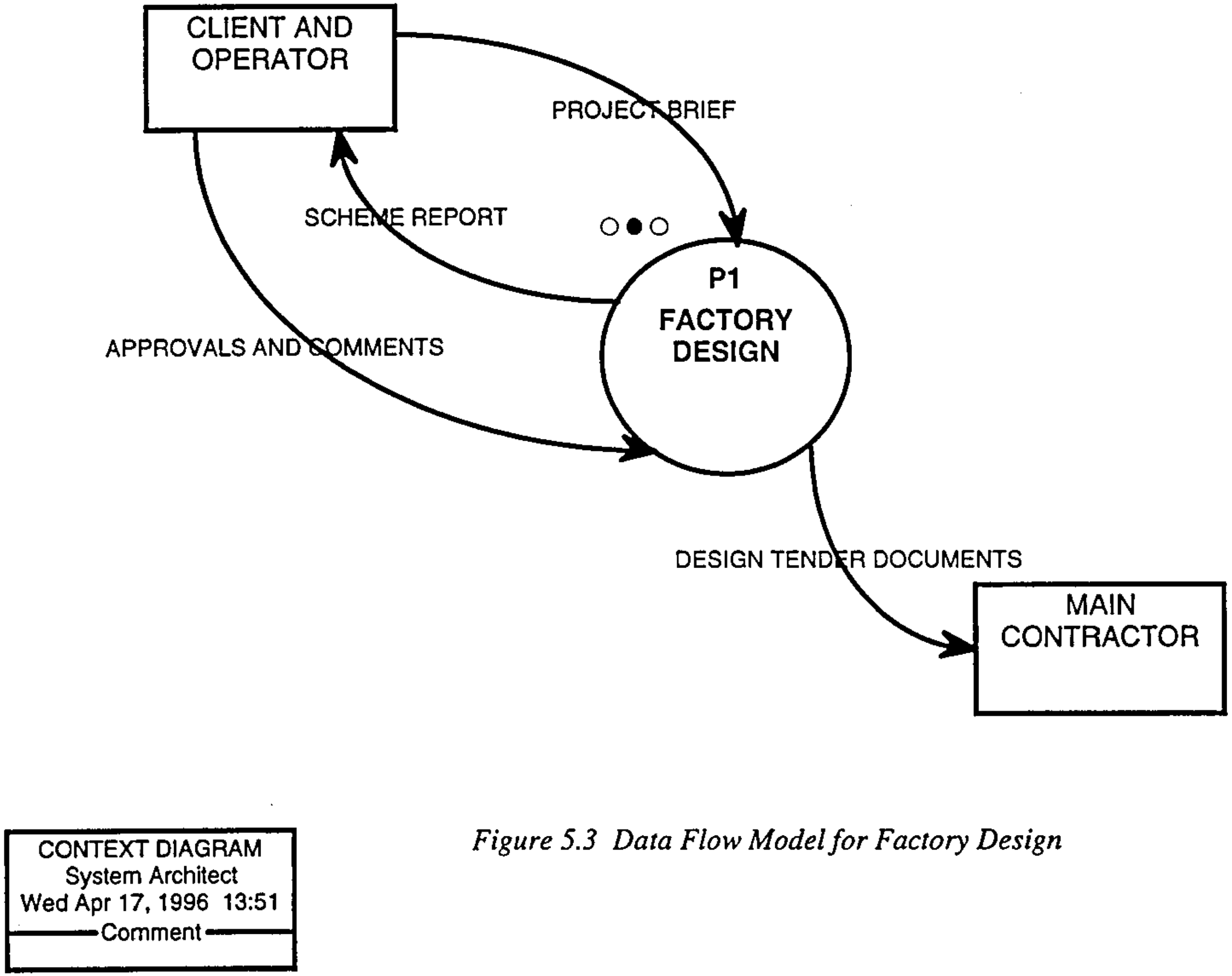
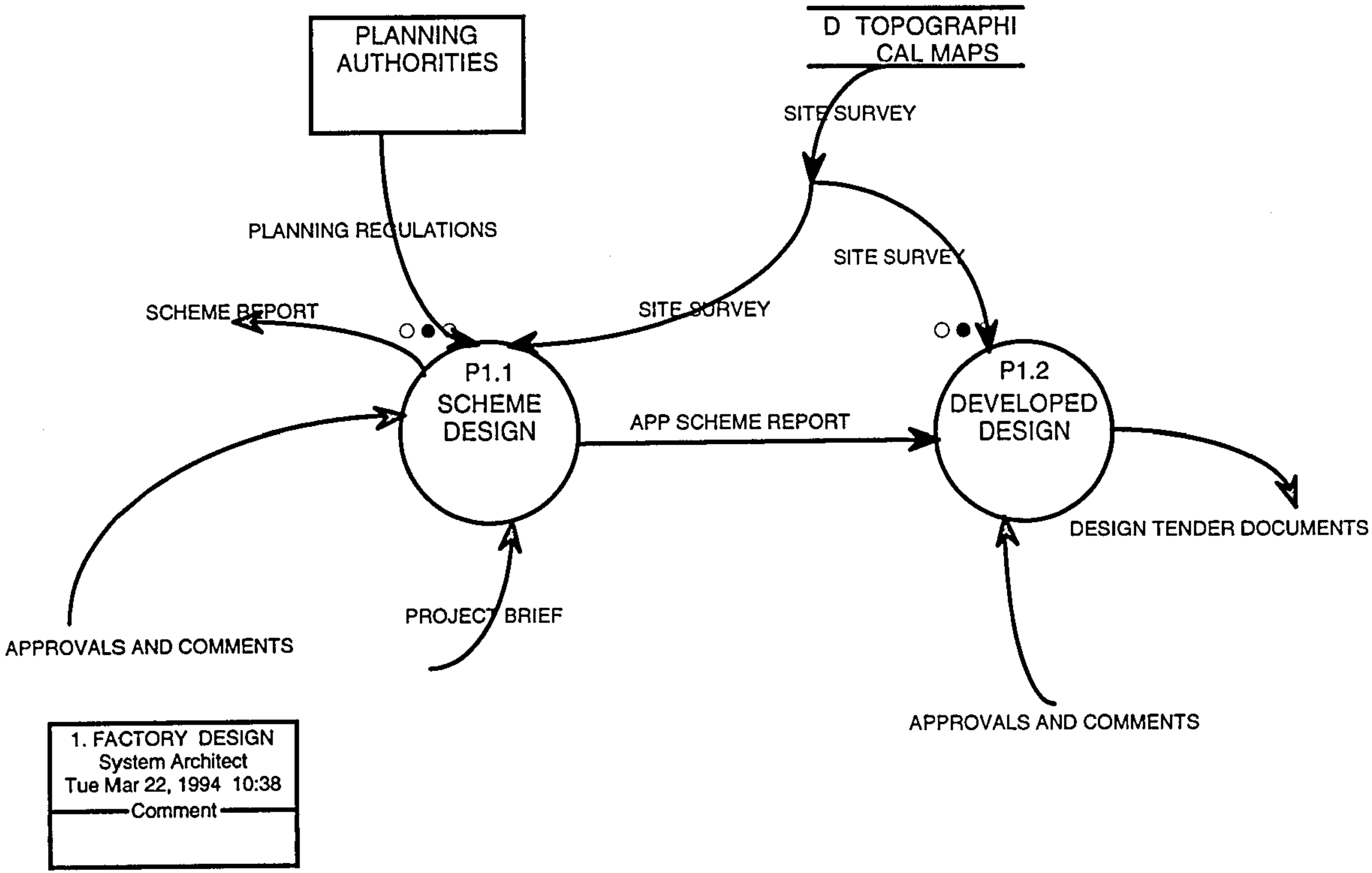
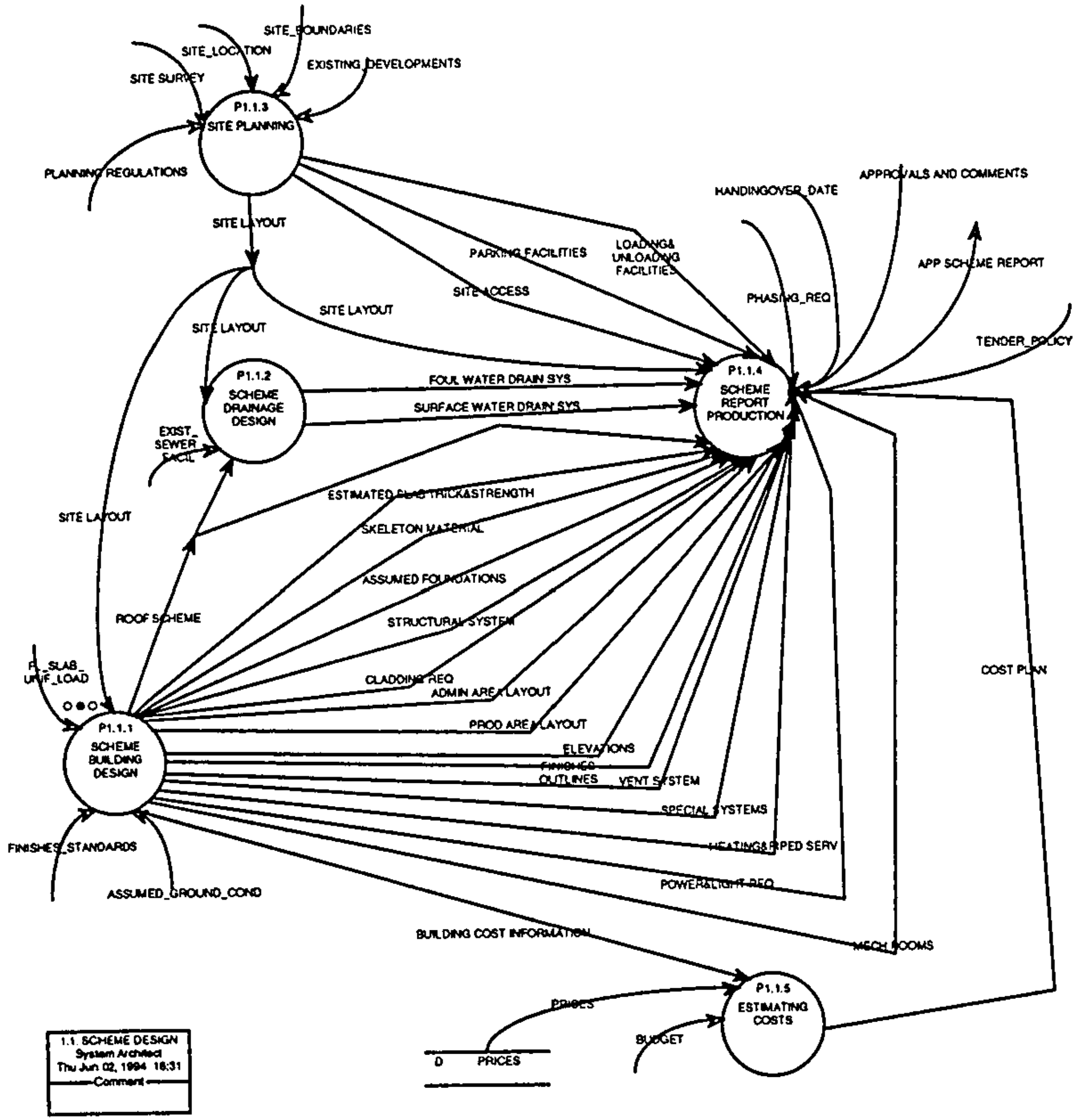
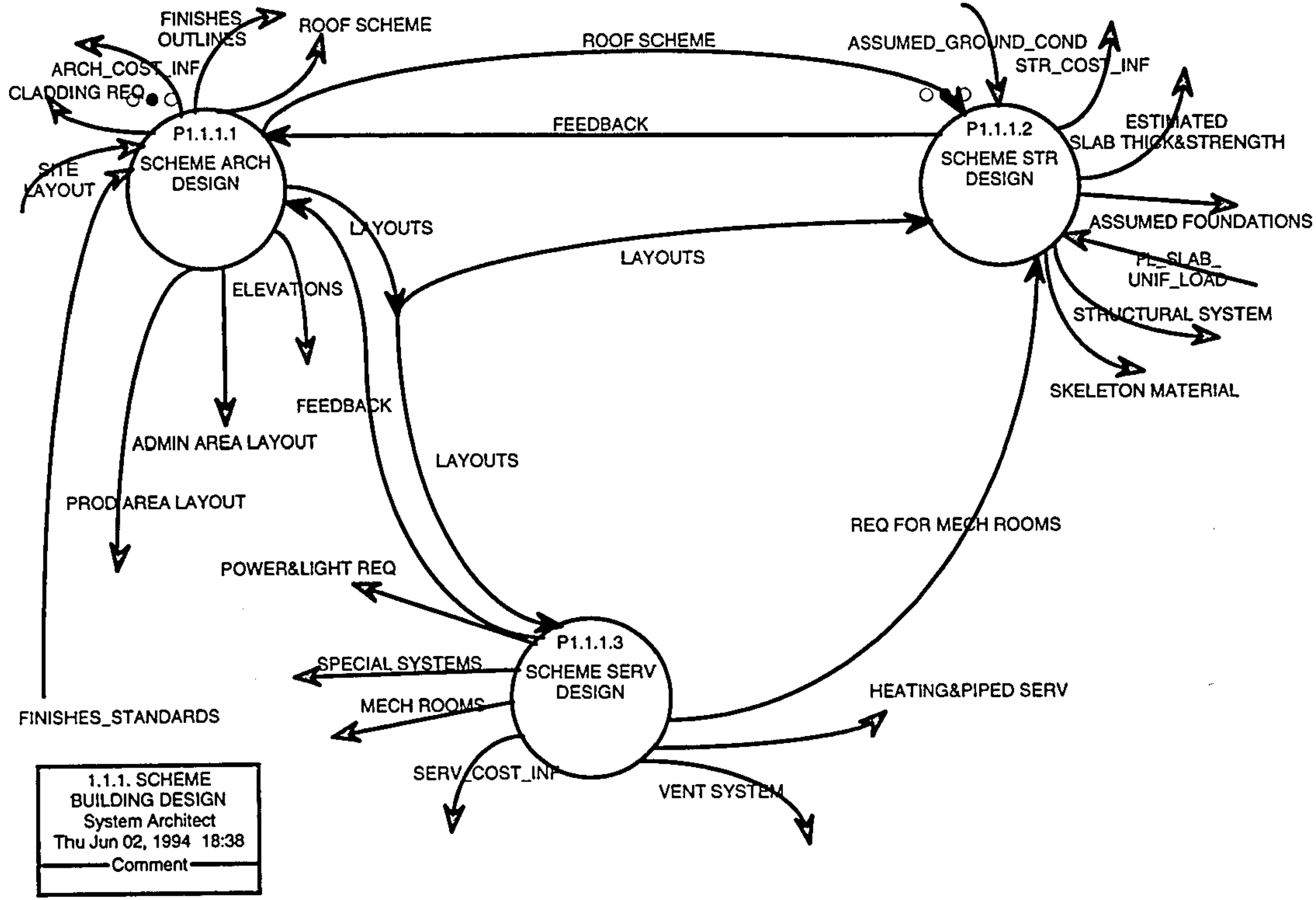


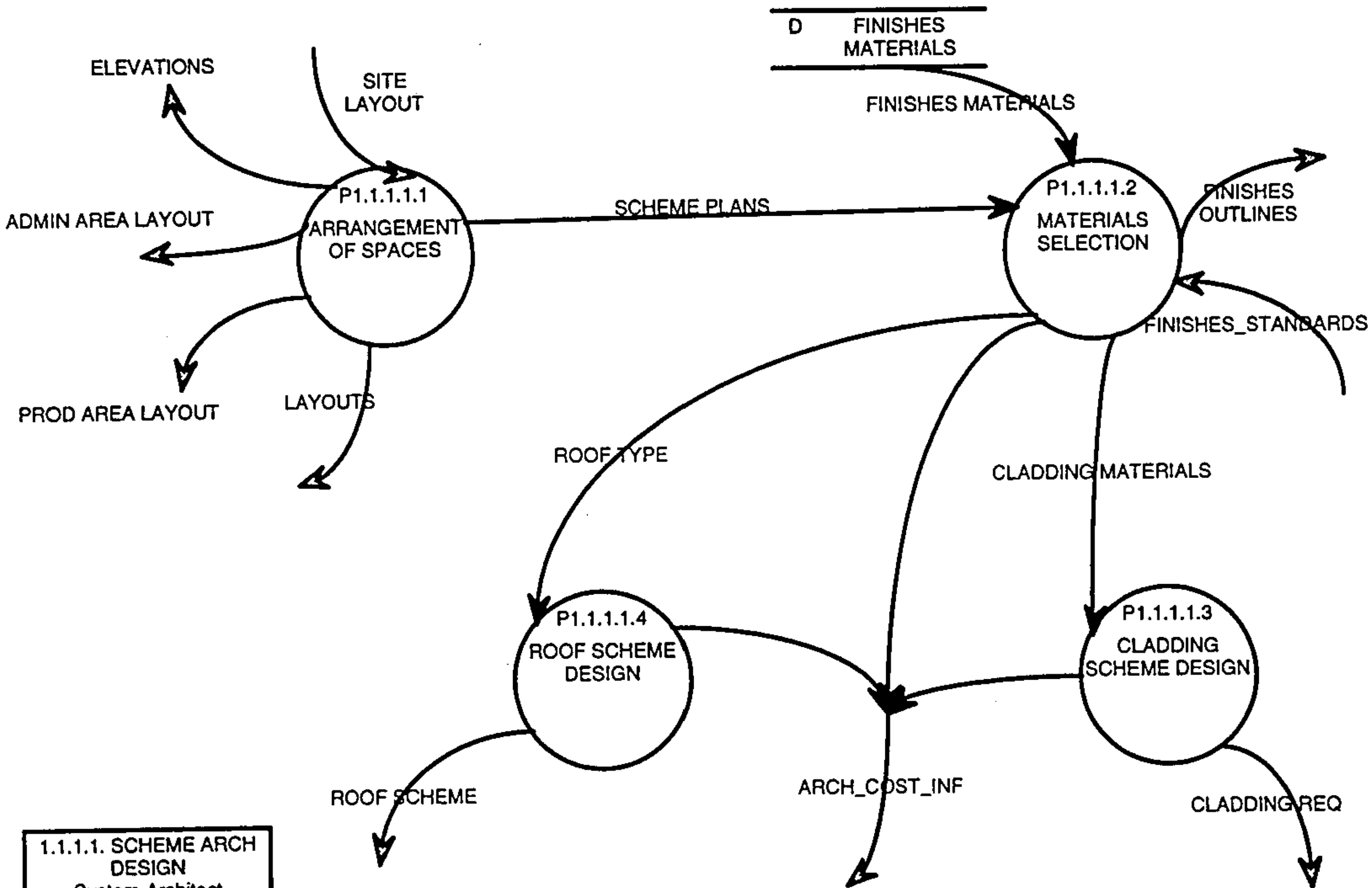
Figure 5.3 Data Flow Model for Factory Design


PLANNING AUTHORITIES
D TOPOGRAPHICAL MAPS
SITE SURVEY
PLANNING REGULATIONS
SITE SURVEY
SCHEME REPORT
SITE SURVEY
P1.1 SCHEME DESIGN
P1.2 DEVELOPED DESIGN
APP SCHEME REPORT
DESIGN TENDER DOCUMENTS
PROJECT BRIEF
APPROVALS AND COMMENTS
APPROVALS AND COMMENTS
1. FACTORY DESIGN
System Architect
Tue Mar 22, 1994 10:38
Comment

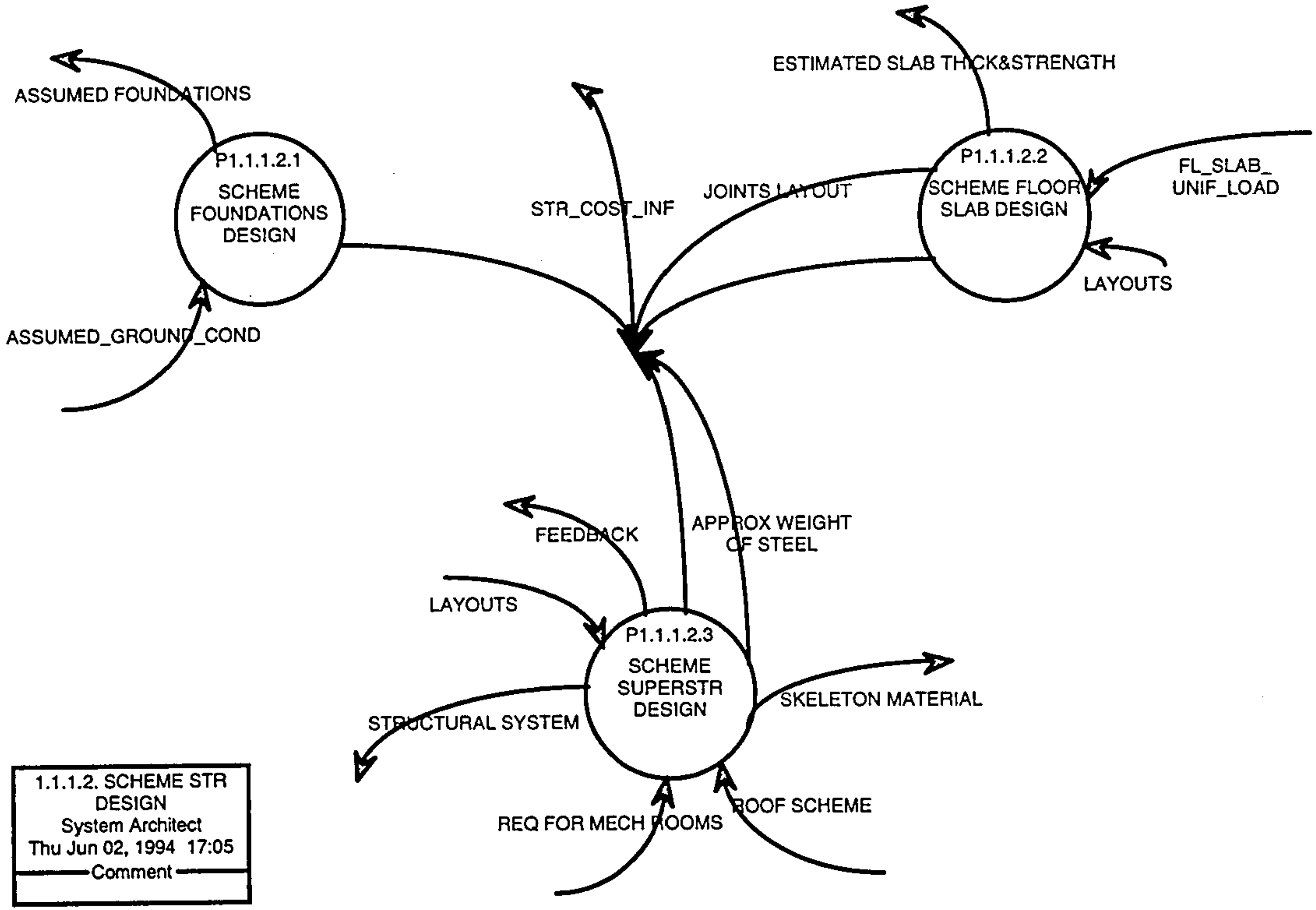


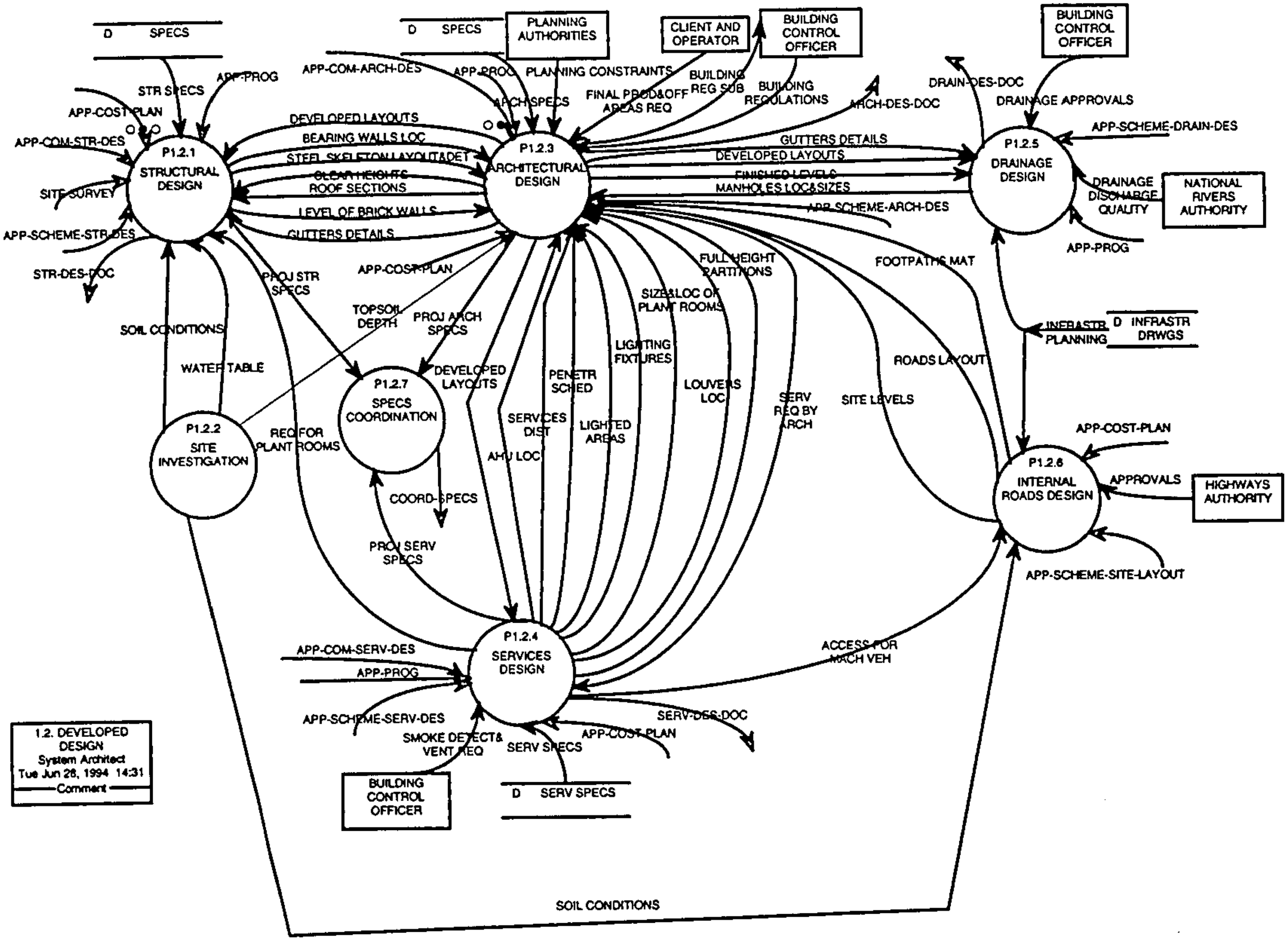
SITE_BOUNDARIES
SITE_LOCATION
EXISTING DEVELOPMENTS
SITE SURVEY
P1.1.3
SITE PLANNING
PLANNING REGULATIONS
SITE LAYOUT
HANDINGOVER_DATE
APPROVALS AND COMMENTS
PARKING FACILITIES
LOADING&
UNLOADING
FACILITIES
APP SCHEME REPORT
SITE ACCESS
PHASING_REQ
SITE LAYOUT
SITE LAYOUT
TENDER_POLICY
P1.1.2
SCHEME
DRAINAGE
DESIGN
FOUL WATER DRAIN SYS
SURFACE WATER DRAIN SYS
P1.1.4
SCHEME
REPORT
PRODUCTION
EXIST_
SEWER_
FACIL
ESTIMATED SLAB THICK&STRENGTH
SITE LAYOUT
SKELETON MATERIAL
ASSUMED FOUNDATIONS
ROOF SCHEME
STRUCTURAL SYSTEM
COST PLAN
FL_SLAB_
UNIF_LOAD
CLADDING REQ
ADMIN AREA LAYOUT
P1.1.1
SCHEME
BUILDING
DESIGN
PROD AREA LAYOUT
ELEVATIONS
FINISHES
OUTLINES
VENT SYSTEM
SPECIAL SYSTEMS
FINISHES_STANDARDS
HEATING&PIPED SERV
ASSUMED_GROUND_COND
POWER&LIGHT REQ
BUILDING COST INFORMATION
MECH ROOMS
P1.1.5
ESTIMATING
COSTS
PRICES
BUDGET
D PRICES
1.1 SCHEME DESIGN
System Architect
Thu Jun 02, 1994 18:31
Comment

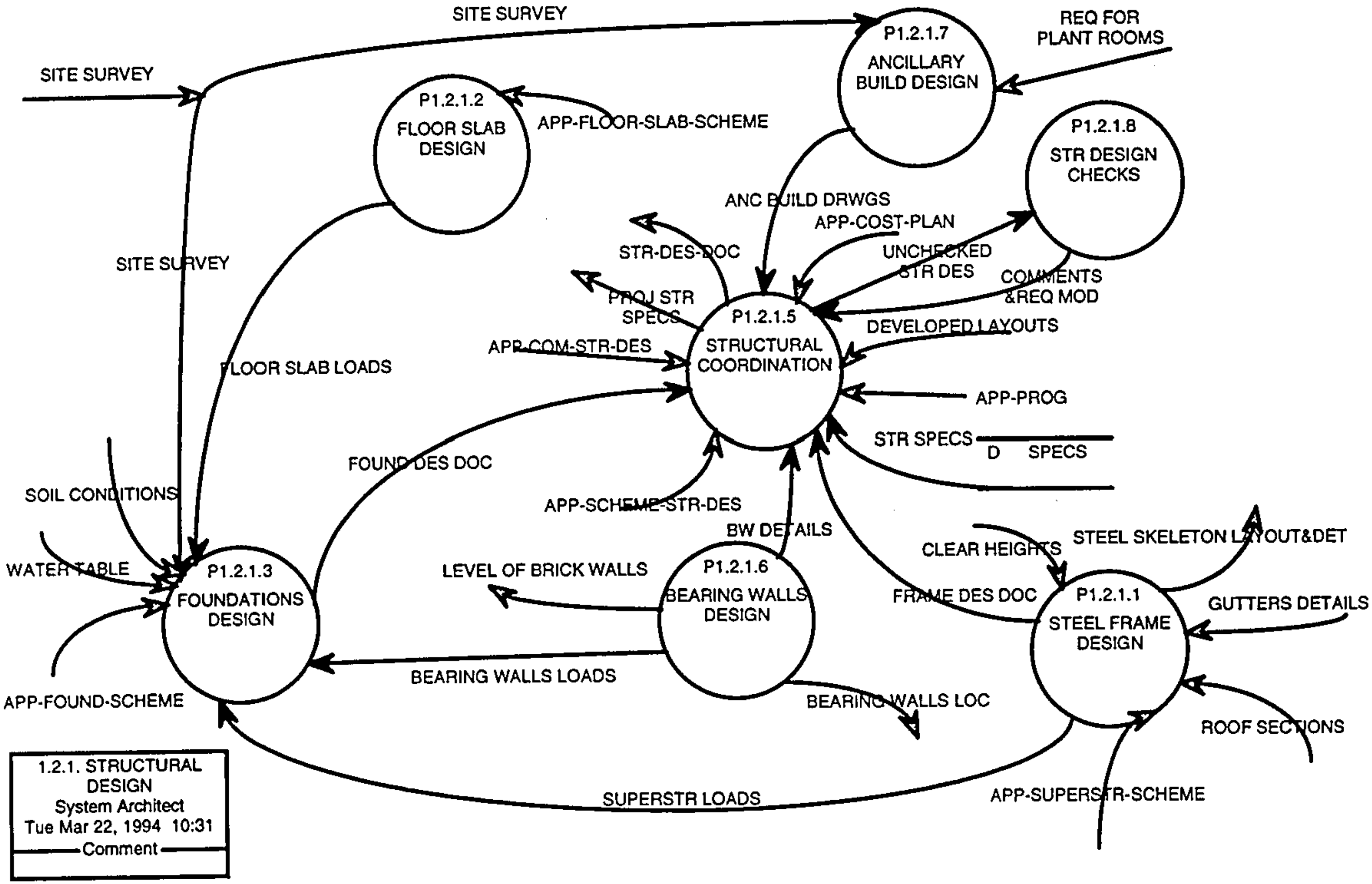

P1.1.1.1
SCHEME ARCH DESIGN
P1.1.1.2
SCHEME STR DESIGN
P1.1.1.3
SCHEME SERV DESIGN
FINISHES OUTLINES
ROOF SCHEME
ARCH_COST_INF
CLADDING REQ
SITE LAYOUT
ROOF SCHEME
FEEDBACK
ASSUMED_GROUND_COND
STR_COST_INF
ESTIMATED SLAB THICK&STRENGTH
ASSUMED FOUNDATIONS
PL_SLAB_ UNIF_LOAD
STRUCTURAL SYSTEM
SKELETON MATERIAL
LAYOUTS
LAYOUTS
ELEVATIONS
FEEDBACK
ADMIN AREA LAYOUT
LAYOUTS
PROD AREA LAYOUT
REQ FOR MECH ROOMS
POWER&LIGHT REQ
SPECIAL SYSTEMS
MECH ROOMS
FINISHES_STANDARDS
HEATING&PIPED SERV
SERV_COST_INF
VENT SYSTEM
1.1.1. SCHEME BUILDING DESIGN
System Architect
Thu Jun 02, 1994 18:38
Comment



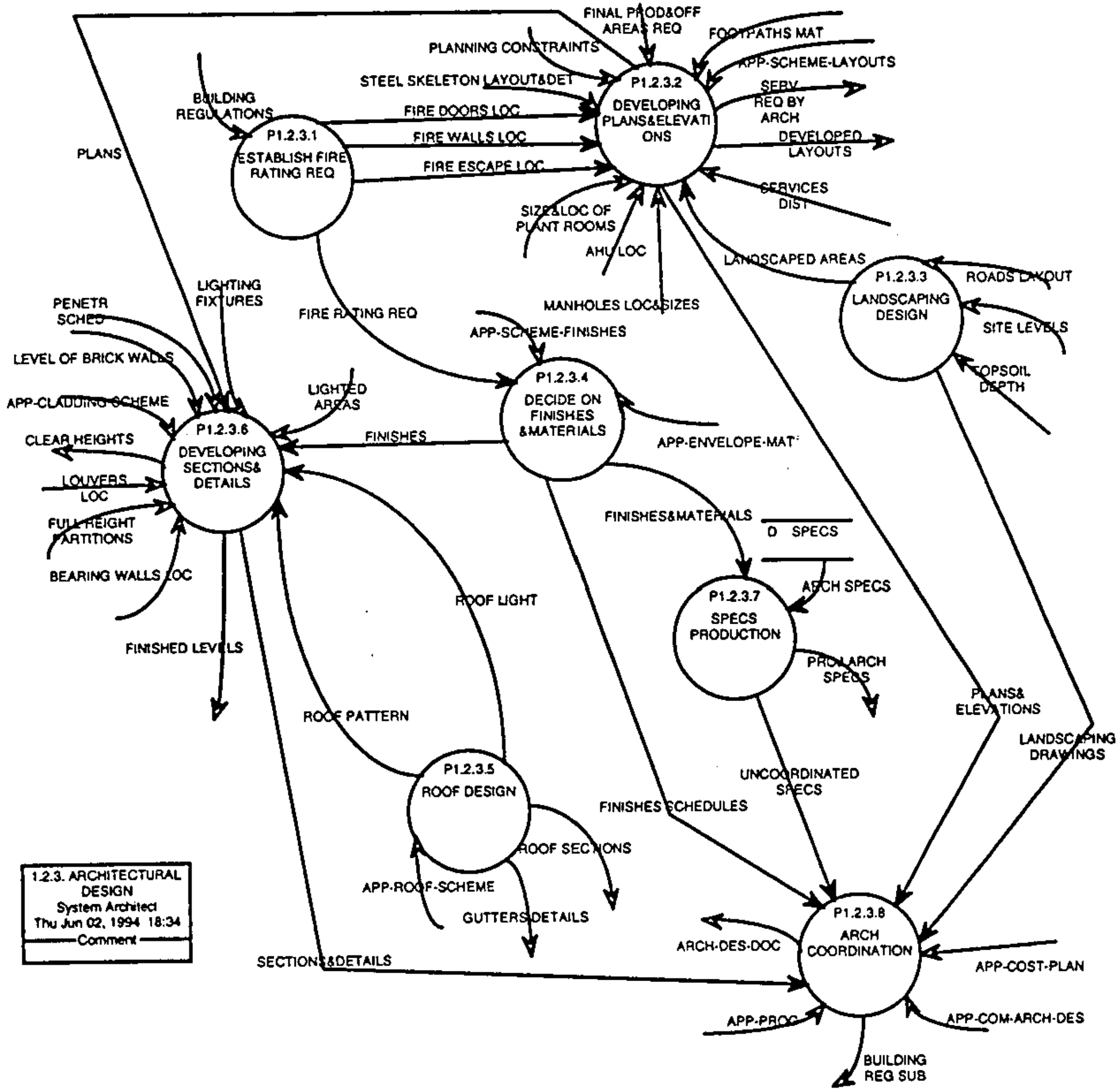
1.1.1.1. SCHEME ARCH DESIGN
System Architect
Tue Mar 22, 1994 10:25
Comment


ASSUMED FOUNDATIONS
P1.1.1.2.1
SCHEME FOUNDATIONS DESIGN
ASSUMED_GROUND_COND
STR_COST_INF
JOINTS LAYOUT
ESTIMATED SLAB THICK&STRENGTH
P1.1.1.2.2
SCHEME FLOOR SLAB DESIGN
FL_SLAB_ UNIF_LOAD
LAYOUTS
FEEDBACK
APPROX WEIGHT OF STEEL
LAYOUTS
P1.1.1.2.3
SCHEME SUPERSTR DESIGN
SKELETON MATERIAL
STRUCTURAL SYSTEM
REQ FOR MECH ROOMS
ROOF SCHEME
1.1.1.2. SCHEME STR DESIGN
System Architect
Thu Jun 02, 1994 17:05
Comment


P1.2.1
STRUCTURAL DESIGN
P1.2.2
SITE INVESTIGATION
P1.2.3
ARCHITECTURAL DESIGN
P1.2.4
SERVICES DESIGN
P1.2.5
DRAINAGE DESIGN
P1.2.6
INTERNAL ROADS DESIGN
P1.2.7
SPECS COORDINATION
D SPECS
D SPECS
D INFRASTR DRWGS
D SERV SPECS
PLANNING AUTHORITIES
CLIENT AND OPERATOR
BUILDING CONTROL OFFICER
BUILDING CONTROL OFFICER
BUILDING CONTROL OFFICER
NATIONAL RIVERS AUTHORITY
HIGHWAYS AUTHORITY
STR SPECS
APP-PROG
APP-COST-PLAN
APP-COM-STR-DES
SITE SURVEY
APP-SCHEME-STR-DES
STR-DES-DOC
SOIL CONDITIONS
WATER TABLE
APP-COM-ARCH-DES
APP-PROG
ARCH SPECS
PLANNING CONSTRAINTS
FINAL PROD&OFF AREAS REQ
BUILDING REG SUB
BUILDING REGULATIONS
ARCH-DES-DOC
DEVELOPED LAYOUTS
BEARING WALLS LOC
STEEL SKELETON LAYOUT&DET
CLEAR HEIGHTS
ROOF SECTIONS
LEVEL OF BRICK WALLS
GUTTERS DETAILS
GUTTERS DETAILS
DEVELOPED LAYOUTS
FINISHED LEVELS
MANHOLES LOC&SIZES
APP-SCHEME-ARCH-DES
DRAIN-DES-DOC
DRAINAGE APPROVALS
APP-SCHEME-DRAIN-DES
DRAINAGE DISCHARGE QUALITY
APP-PROG
FOOTPATHS MAT
INFRASTR PLANNING
ROADS LAYOUT
SITE LEVELS
APP-COST-PLAN
APPROVALS
APP-SCHEME-SITE-LAYOUT
PROJ STR SPECS
APP-COST-PLAN
TOPSOIL DEPTH
PROJ ARCH SPECS
DEVELOPED LAYOUTS
REQ FOR PLANT ROOMS
COORD-SPECS
PROJ SERV SPECS
FULL HEIGHT PARTITIONS
SIZE&LOC OF PLANT ROOMS
LIGHTING FIXTURES
PENETR SCHED
SERVICES DIST
LIGHTED AREAS
LOUVERS LOC
SERV REQ BY ARCH
AHU LOC
ACCESS FOR MACH VEH
APP-COM-SERV-DES
APP-PROG
APP-SCHEME-SERV-DES
SMOKE DETECT& VENT REQ
SERV SPECS
APP-COST-PLAN
SERV-DES-DOC
SOIL CONDITIONS
1.2. DEVELOPED DESIGN
System Architect
Tue Jun 28, 1994 14:31
Comment



SITE SURVEY
SITE SURVEY
P1.2.1.7
ANCILLARY BUILD DESIGN
REQ FOR PLANT ROOMS
P1.2.1.2
FLOOR SLAB DESIGN
APP-FLOOR-SLAB-SCHEME
P1.2.1.8
STR DESIGN CHECKS
ANC BUILD DRWGS
APP-COST-PLAN
SITE SURVEY
STR-DES-DOC
UNCHECKED STR DES
COMMENTS &REQ MOD
PROJ STR SPECS
P1.2.1.5
STRUCTURAL COORDINATION
DEVELOPED LAYOUTS
APP-COM-STR-DES
FLOOR SLAB LOADS
APP-PROG
STR SPECS
D SPECS
FOUND DES DOC
SOIL CONDITIONS
APP-SCHEME-STR-DES
BW DETAILS
CLEAR HEIGHTS
STEEL SKELETON LAYOUT&DET
WATER TABLE
P1.2.1.3
FOUNDATIONS DESIGN
LEVEL OF BRICK WALLS
P1.2.1.6
BEARING WALLS DESIGN
FRAME DES DOC
P1.2.1.1
STEEL FRAME DESIGN
GUTTERS DETAILS
BEARING WALLS LOADS
APP-FOUND-SCHEME
BEARING WALLS LOC
ROOF SECTIONS
SUPERSTR LOADS
APP-SUPERSTR-SCHEME
1.2.1. STRUCTURAL DESIGN
System Architect
Tue Mar 22, 1994 10:31
Comment



P1.2.3.1
ESTABLISH FIRE RATING REQ
P1.2.3.2
DEVELOPING PLANS&ELEVATIONS
P1.2.3.3
LANDSCAPING DESIGN
P1.2.3.4
DECIDE ON FINISHES &MATERIALS
P1.2.3.5
ROOF DESIGN
P1.2.3.6
DEVELOPING SECTIONS& DETAILS
P1.2.3.7
SPECS PRODUCTION
P1.2.3.8
ARCH COORDINATION
D SPECS
FINAL PROD&OFF AREAS REQ
PLANNING CONSTRAINTS
STEEL SKELETON LAYOUT&DET
FOOTPATHS MAT
APP-SCHEME-LAYOUTS
SERV REQ BY ARCH
DEVELOPED LAYOUTS
SERVICES DIST
BUILDING REGULATIONS
FIRE DOORS LOC
FIRE WALLS LOC
FIRE ESCAPE LOC
SIZE&LOC OF PLANT ROOMS
AHU LOC
MANHOLES LOC&SIZES
LANDSCAPED AREAS
ROADS LAYOUT
SITE LEVELS
TOPSOIL DEPTH
PLANS
LIGHTING FIXTURES
FIRE RATING REQ
APP-SCHEME-FINISHES
PENETR SCHED
LEVEL OF BRICK WALLS
APP-CLADDING-SCHEME
LIGHTED AREAS
CLEAR HEIGHTS
LOUVERS LOC
FULL HEIGHT PARTITIONS
BEARING WALLS LOC
FINISHES
APP-ENVELOPE-MAT
FINISHES&MATERIALS
ARCH SPECS
PRO ARCH SPECS
ROOF LIGHT
FINISHED LEVELS
ROOF PATTERN
PLANS& ELEVATIONS
LANDSCAPING DRAWINGS
UNCOORDINATED SPECS
FINISHES SCHEDULES
ROOF SECTIONS
APP-ROOF-SCHEME
GUTTERS DETAILS
ARCH-DES-DOC
APP-COST-PLAN
SECTIONS&DETAILS
APP-PROC
APP-COM-ARCH-DES
BUILDING REG SUB
1.2.3. ARCHITECTURAL DESIGN
System Architect
Thu Jun 02, 1994 18:34
Comment

5.4 THE GENERIC DATA FLOW MODEL FOR CONCEPTUAL AND SCHEMATIC DESIGN

5.4.1 The Development and Validation of the Model

The production of the Generic Model was an iterative process. The initial version of the model was produced on the basis of two preliminary case studies. This was followed by validation of the model through interviews with industry professionals within Ove Arup and Partners, AMEC Design and Management, John Laing and Fisons. This resulted in some refinements and suggestions which were incorporated into the model. A few tasks and information requirements were regarded by some interviewees as part of later design stages. This is partly due to the natural overlap between the different stages of design and partly due to the lack of consensus among researchers and professionals in the construction industry about the tasks that comprise every design stage as explained in Chapter 2. However, to maintain the generic nature of the model, it was decided to retain these elements to be used at the discretion of every user organisation. The writer has also presented the model together with the overall research objectives during a seminar held within AMEC Design and Management and attended by senior managers from AMEC, Ove Arup and Partners and CIBSE. Feedback from the attendees reflected the need of industry professionals for a better understanding of the Conceptual and Schematic stages of design and the lack of sophisticated tools for design management. This reaction confirmed the contribution that this research would provide to the industry. Additionally, a survey followed by subsequent interviews were conducted over twenty construction professionals within three major construction organisations namely Kyle Stewart, AMEC Design and Management and Ove Arup and Partners (refer to tables 4.2 and 4.1 in chapter 4 for a list of the interviewees and the role of the industry involvement). One of the objectives of the survey and interviews was to validate the developed model and to identify important aspects of the design process. The results showed the following:

(i) The model being for the Conceptual and Schematic design stages is independent of the procurement strategy.

(ii) The Schematic design stage is more difficult to manage than the Detailed design stage. This highlights the value of the developed model.

(iii) Due to the fact that all the interviewees were of different backgrounds there was no real pattern in the difficult information sources identified nor the difficult design tasks identified.

(iv) For the difficult design tasks identified by the interviewees, the difficulty in obtaining the information requirements for these tasks varied, but the importance of these information did not. The interviewees identified all technical information as important (ranked as >=5 on a scale of 1-7, 7 being most important) for the design task to proceed. (Information like the approved program and approved cost plan was seen by some as less important)

(v) The difficulties with external information sources where approvals and regulations were necessary, (e.g. different authorities, insurers) were due to the fact that these sources are involved after a substantial part of the design has been already completed, and hence any input may require re-design and other implications on other design tasks. Also there was frequently a difficulty in interpretation of regulations and the time taken by these sources to take decisions or provide approvals.

(vi) Difficulty in obtaining information and the importance of some information were seen from different perspectives according to the background and discipline of the interviewee. A piece of information may be considered as important or difficult by one designer or manager but not have the same importance by another.

(vii) Difficulties in communications or acquiring information are more prevalent when dealing with external sources. Information required from sources or disciplines within the same organisation is easier to obtain as it is more difficult to control external sources.

(viii) There is no formal way to judge the quality of information exchanged. The measure of information quality varies according to the sender and the recipient of information. Missing information or information of insufficient quality from the recipients point of view are normally supplemented by assumptions on the part of the recipient.

(ix) Data flow models provide a useful effective tool which may be used to improve communications during the design process, and hence improve the management of the process. These models assist in identifying information requirements for different design tasks, and in identifying other designers' problems. They may be used in the training of engineers and architects.

The developed Generic Data Flow Model was also validated against the actual design process during a case study undertaken by the writer in collaboration with AMEC Design and Management as designers and Loughborough University of Technology as the Client. The data collected included notes and observations recorded by the writer when 'shadowing' the meetings held between AMEC and the different Client committees and user groups in addition to minutes of meetings. The information exchanged was recorded and then categorised within different headings and allocated to at least one of the information flows on the Generic Model. This confirmed that the model represented the design process subject to minor adjustments due to the special nature of the project. Details about this case study are included in Chapter 7.

5.4.2 The Generic Data Flow Model

The final version of the Generic Data Flow Model for the Conceptual and Schematic stages of design is presented in Figure 5.5. The model is decomposed into five levels and contains some 50 functional primitive tasks. The hierarchy and structure of the model is illustrated in Figure 5.4. Two reports from the data dictionary produced by the software tool are included in Appendix VI. The report 'All definitions' shows the breakdown of the defined data flows into data elements and the necessary definitions for some functional primitive tasks. The report 'Data flow diagram symbol list' lists all the processes, data flows, information sources and data stores within the model. Other reports such as listing processes inputs and outputs may also be produced. It is noted that it is not the intention in this research to provide complete definitions for design processes and data flows as it is considered that the model will be used by designers who are familiar with design expressions. Additionally, to maintain the generic nature of the model, specific definitions for the scope of design tasks and contents of information flow may be specified in more detail (if necessary) by the user organisations to meet their specific practice and/or particular nature of design works.

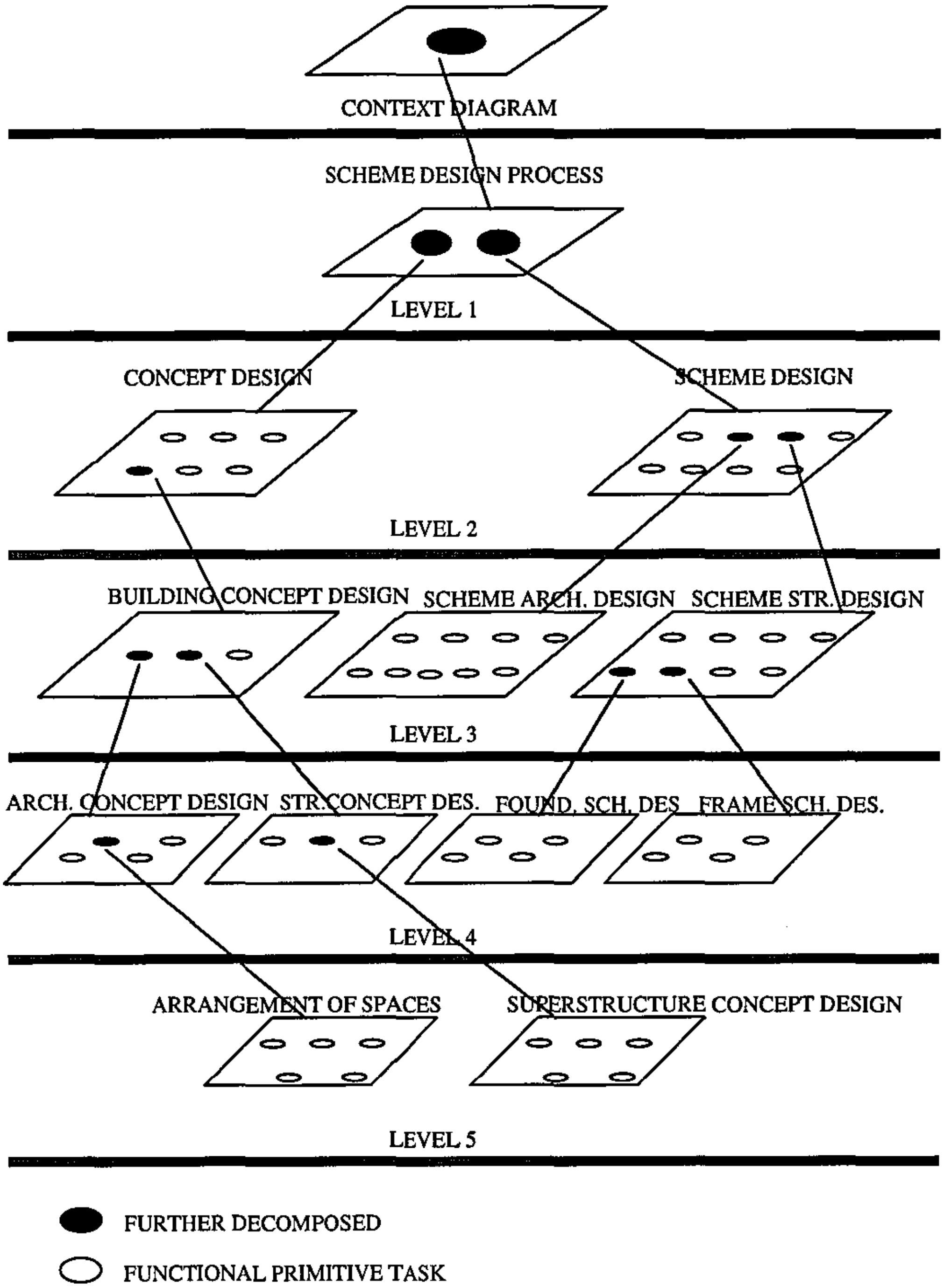


Figure 5.4 The hierarchy and structure of the Generic Data Flow Model

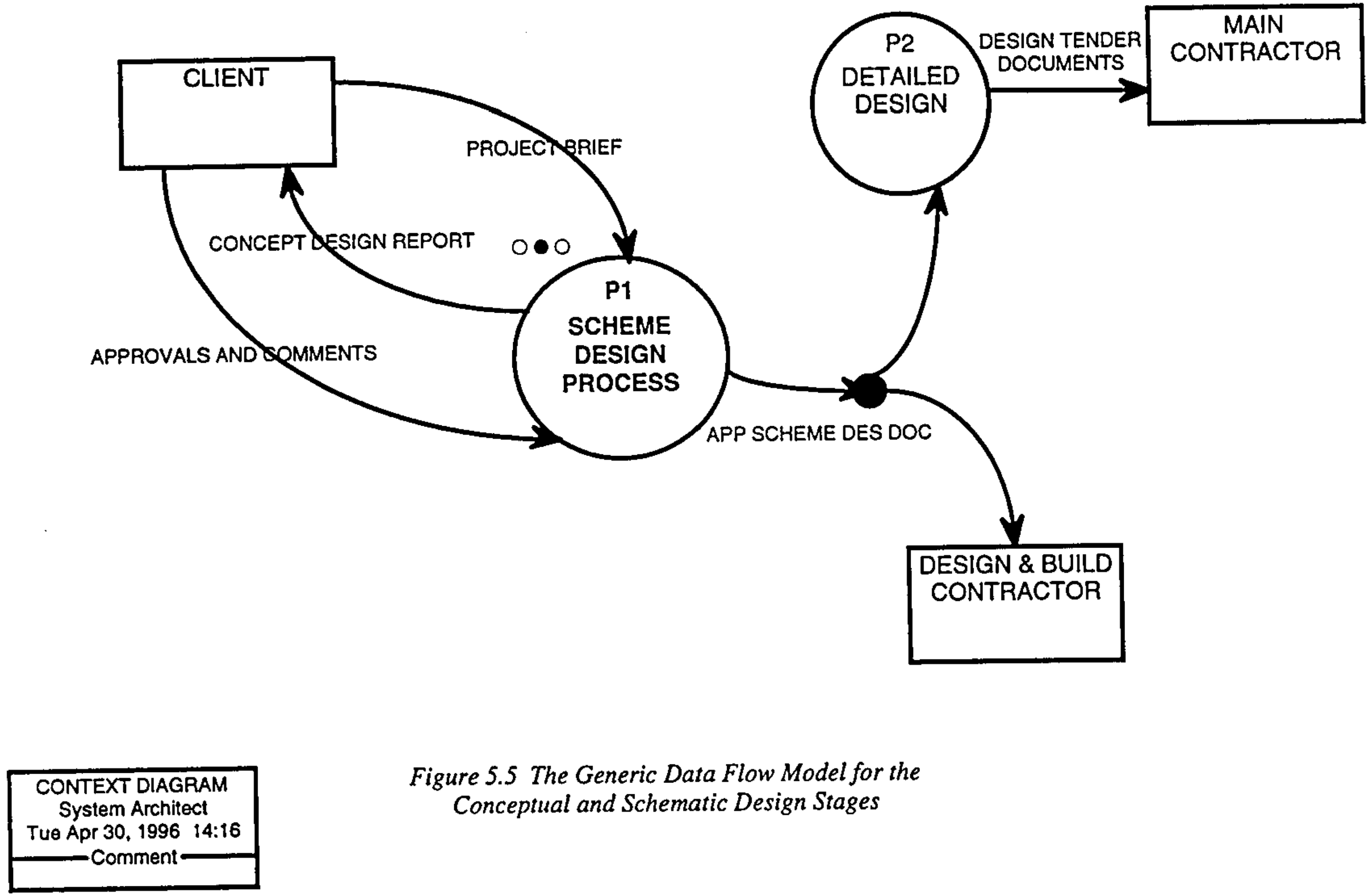
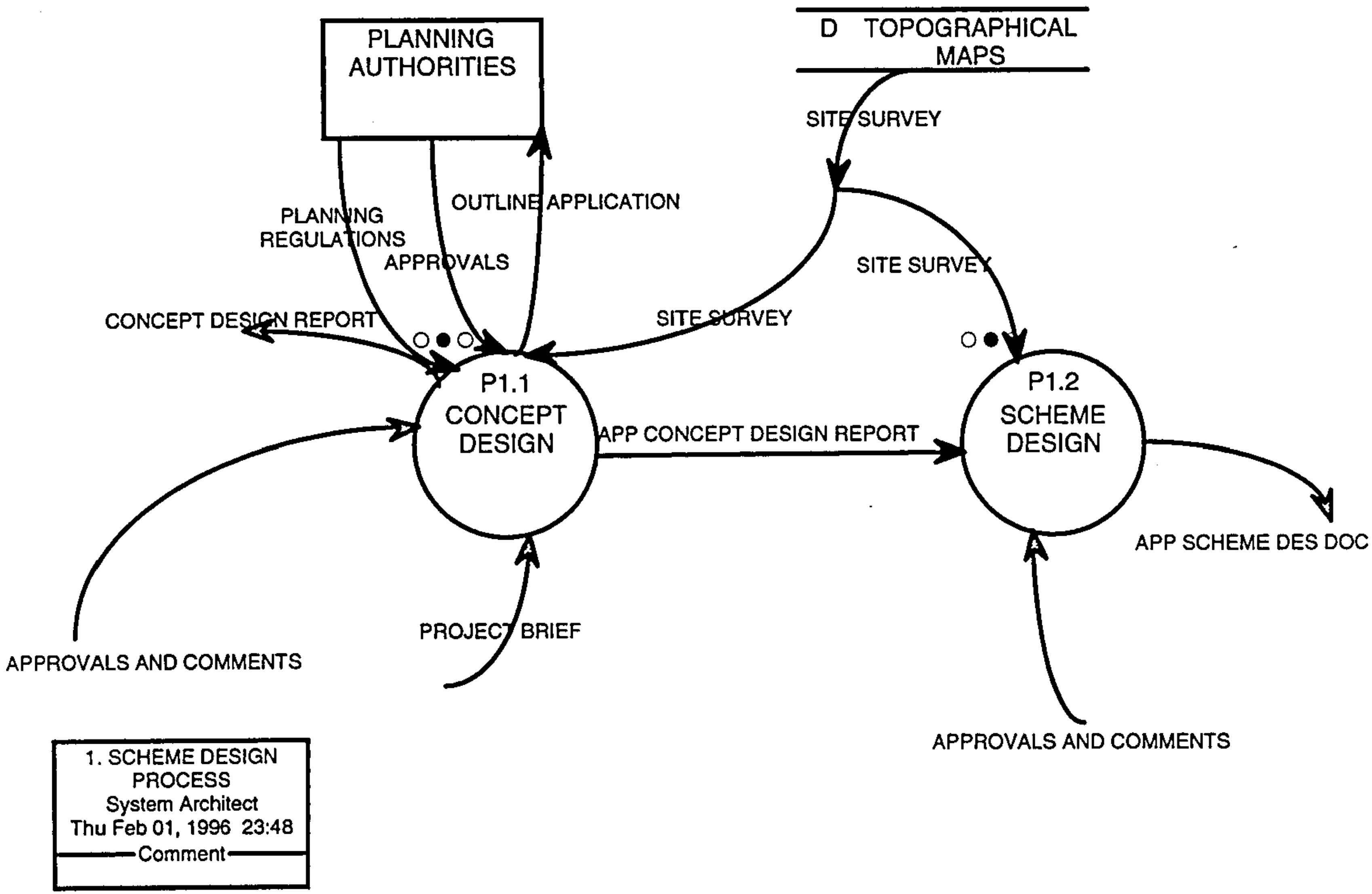
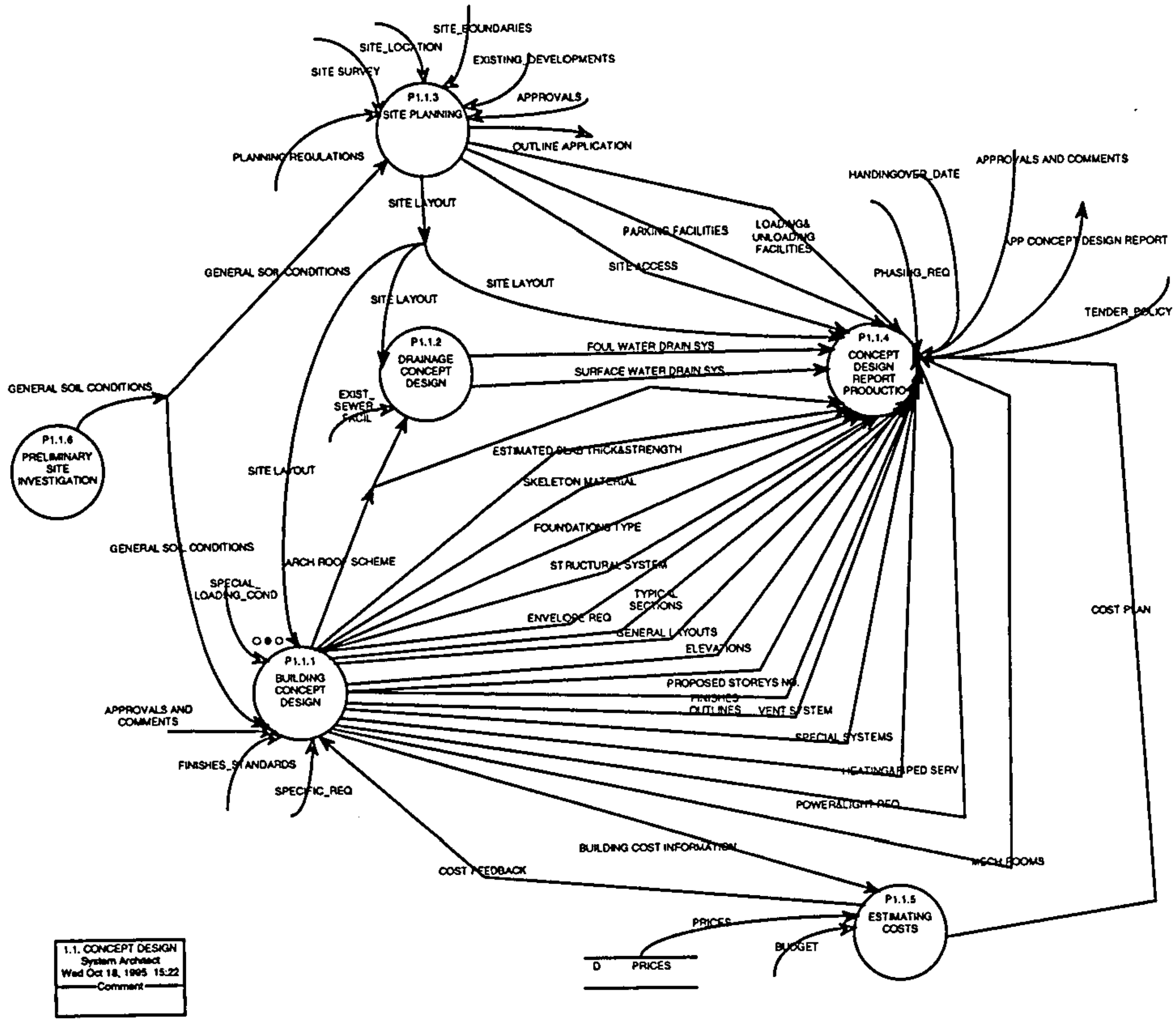
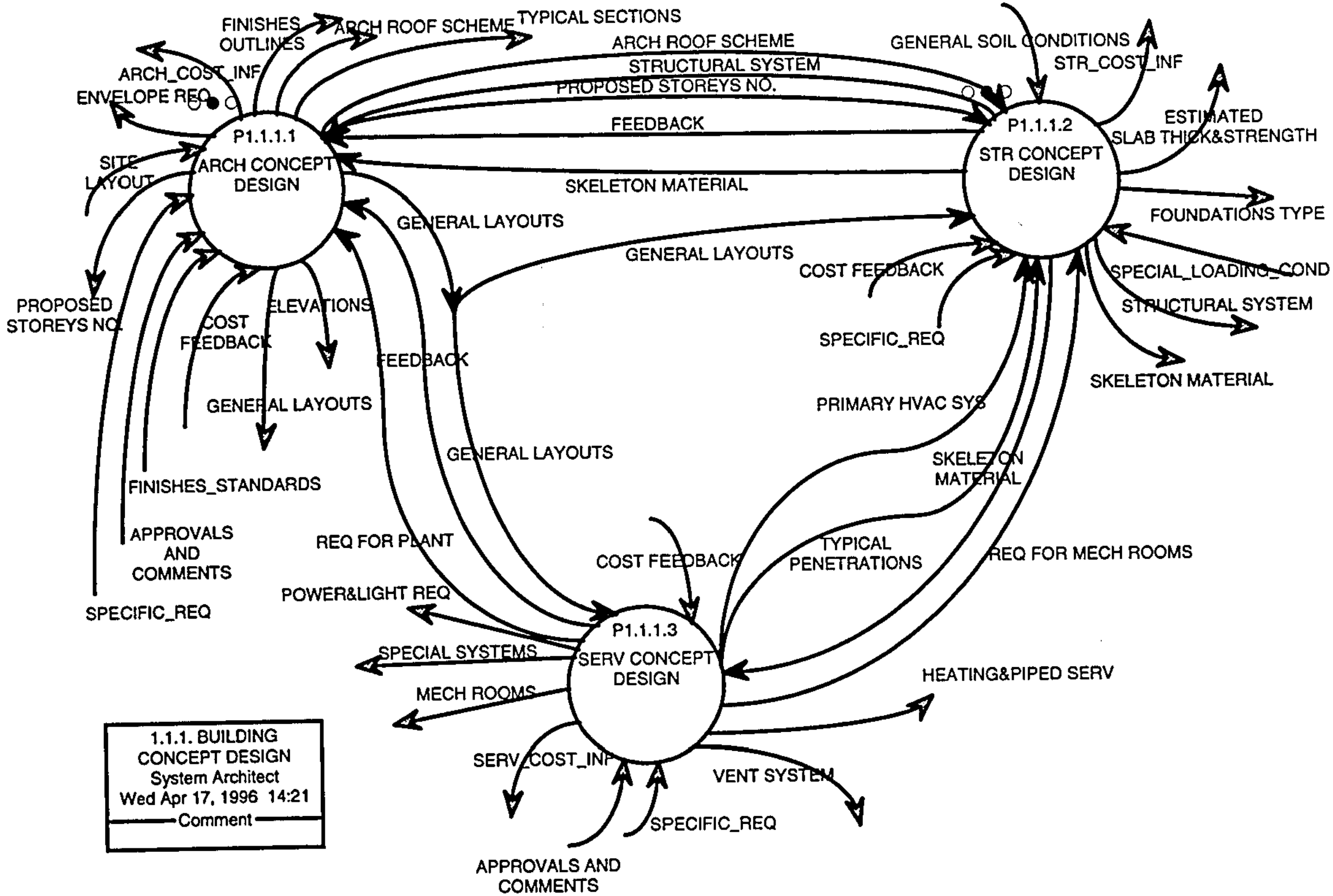


Figure 5.5 The Generic Data Flow Model for the Conceptual and Schematic Design Stages

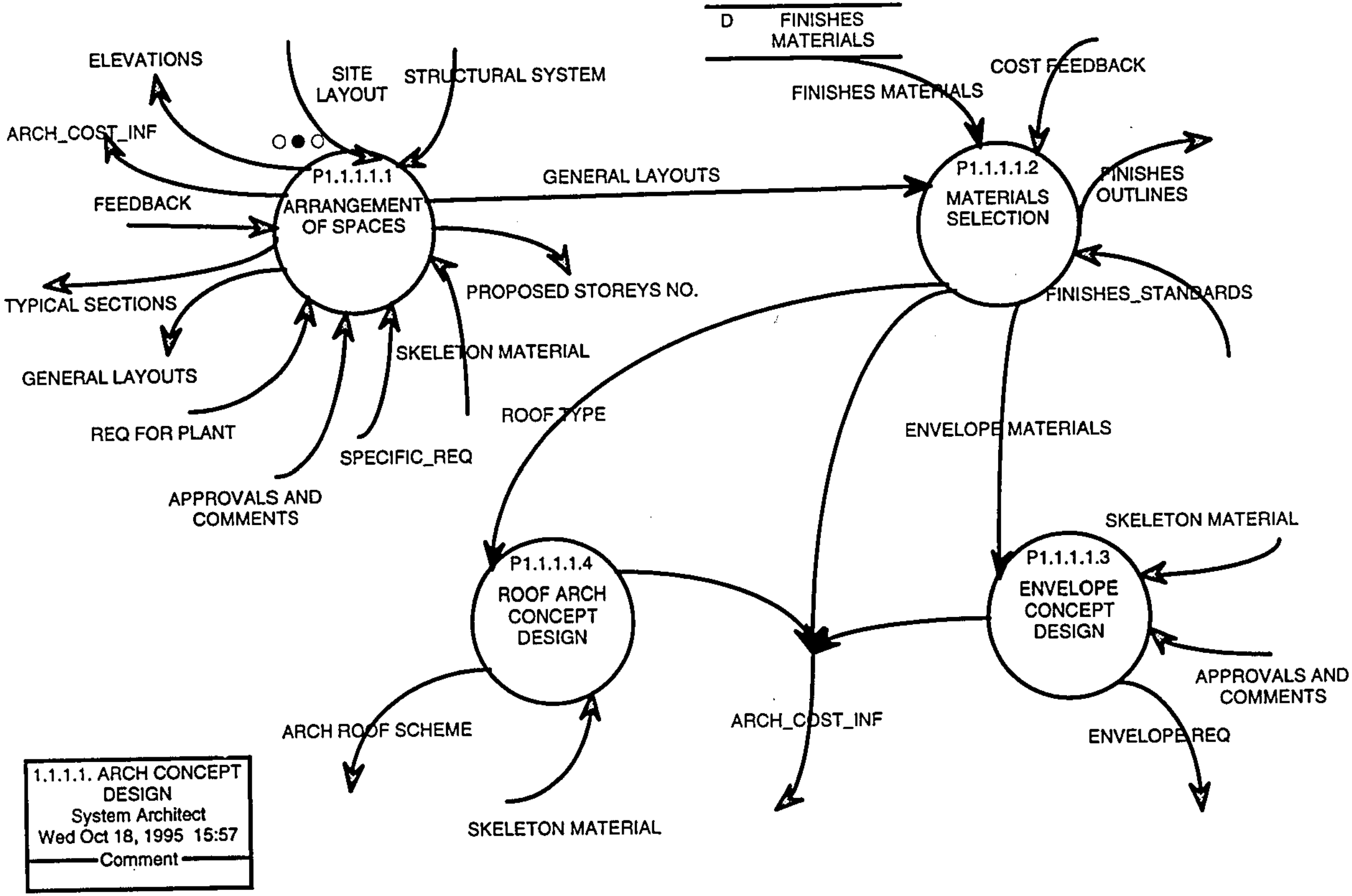

PLANNING AUTHORITIES
D TOPOGRAPHICAL MAPS
SITE SURVEY
OUTLINE APPLICATION
PLANNING REGULATIONS
APPROVALS
SITE SURVEY
SITE SURVEY
CONCEPT DESIGN REPORT
P1.1 CONCEPT DESIGN
APP CONCEPT DESIGN REPORT
P1.2 SCHEME DESIGN
APP SCHEME DES DOC
PROJECT BRIEF
APPROVALS AND COMMENTS
APPROVALS AND COMMENTS
1. SCHEME DESIGN PROCESS
System Architect
Thu Feb 01, 1996 23:48
Comment



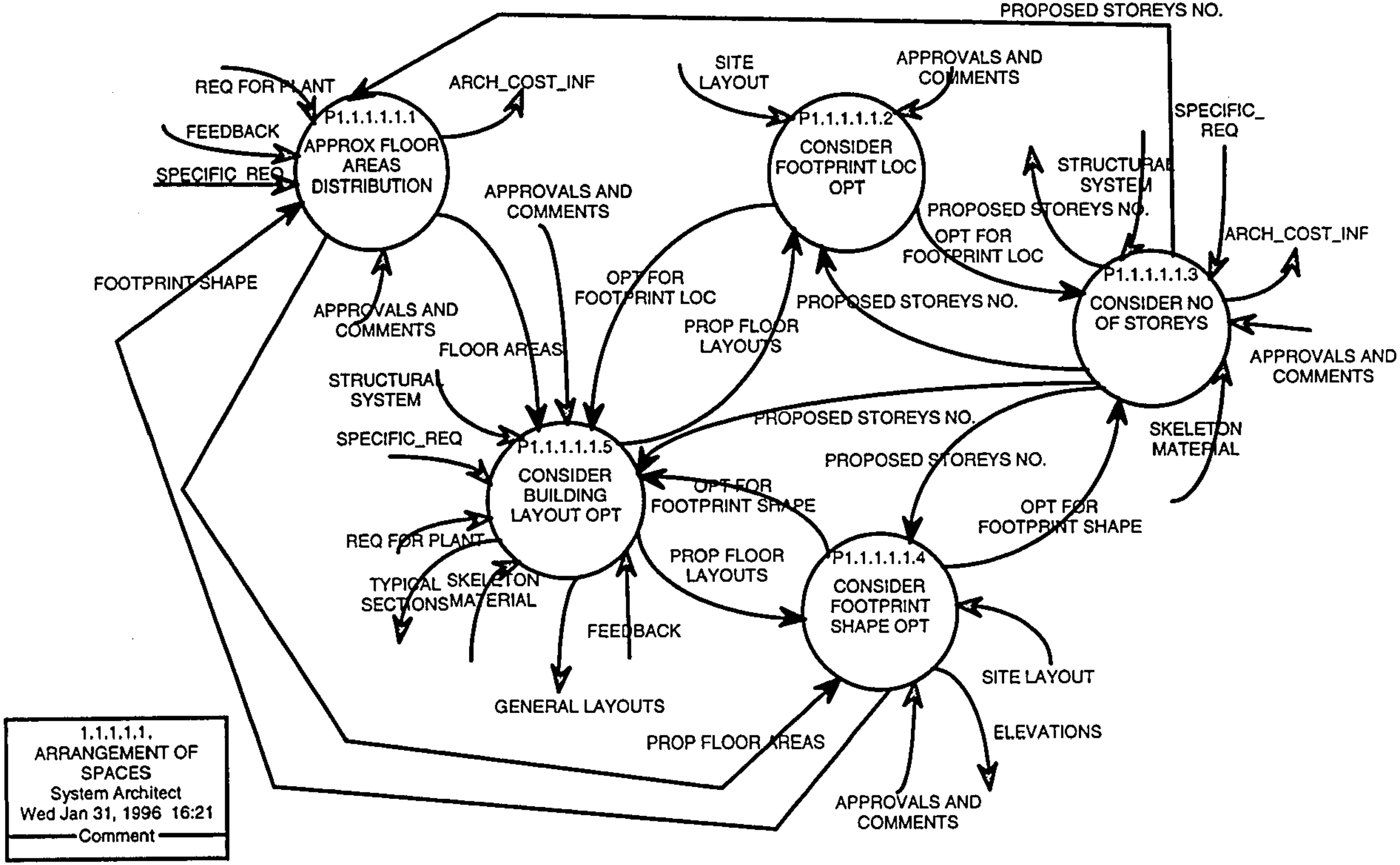
SITE_BOUNDARIES
SITE_LOCATION
EXISTING_DEVELOPMENTS
SITE SURVEY
APPROVALS
P1.1.3
SITE PLANNING
OUTLINE APPLICATION
PLANNING REGULATIONS
SITE LAYOUT
GENERAL SOIL CONDITIONS
PARKING FACILITIES
LOADING& UNLOADING FACILITIES
SITE ACCESS
SITE LAYOUT
SITE LAYOUT
HANDINGOVER_DATE
APPROVALS AND COMMENTS
APP CONCEPT DESIGN REPORT
PHASING_REQ
TENDER_POLICY
P1.1.2
DRAINAGE CONCEPT DESIGN
FOUL WATER DRAIN SYS
SURFACE WATER DRAIN SYS
P1.1.4
CONCEPT DESIGN REPORT PRODUCTION
GENERAL SOIL CONDITIONS
EXIST_ SEWER_ FACIL
P1.1.6
PRELIMINARY SITE INVESTIGATION
ESTIMATED SLAB THICK&STRENGTH
SITE LAYOUT
SKELETON MATERIAL
FOUNDATIONS TYPE
GENERAL SOIL CONDITIONS
ARCH ROOF SCHEME
STRUCTURAL SYSTEM
SPECIAL_ LOADING_COND
TYPICAL SECTIONS
ENVELOPE REQ
GENERAL LAYOUTS
ELEVATIONS
COST PLAN
P1.1.1
BUILDING CONCEPT DESIGN
PROPOSED STOREYS NO.
FINISHES OUTLINES
VENT SYSTEM
APPROVALS AND COMMENTS
SPECIAL SYSTEMS
HEATING&PIPED SERV
FINISHES_STANDARDS
SPECIFIC_REQ
POWER&LIGHT REQ
BUILDING COST INFORMATION
MECH ROOMS
COST FEEDBACK
P1.1.5
ESTIMATING COSTS
PRICES
BUDGET
D PRICES
1.1. CONCEPT DESIGN
System Architect
Wed Oct 18, 1995 15:22
Comment

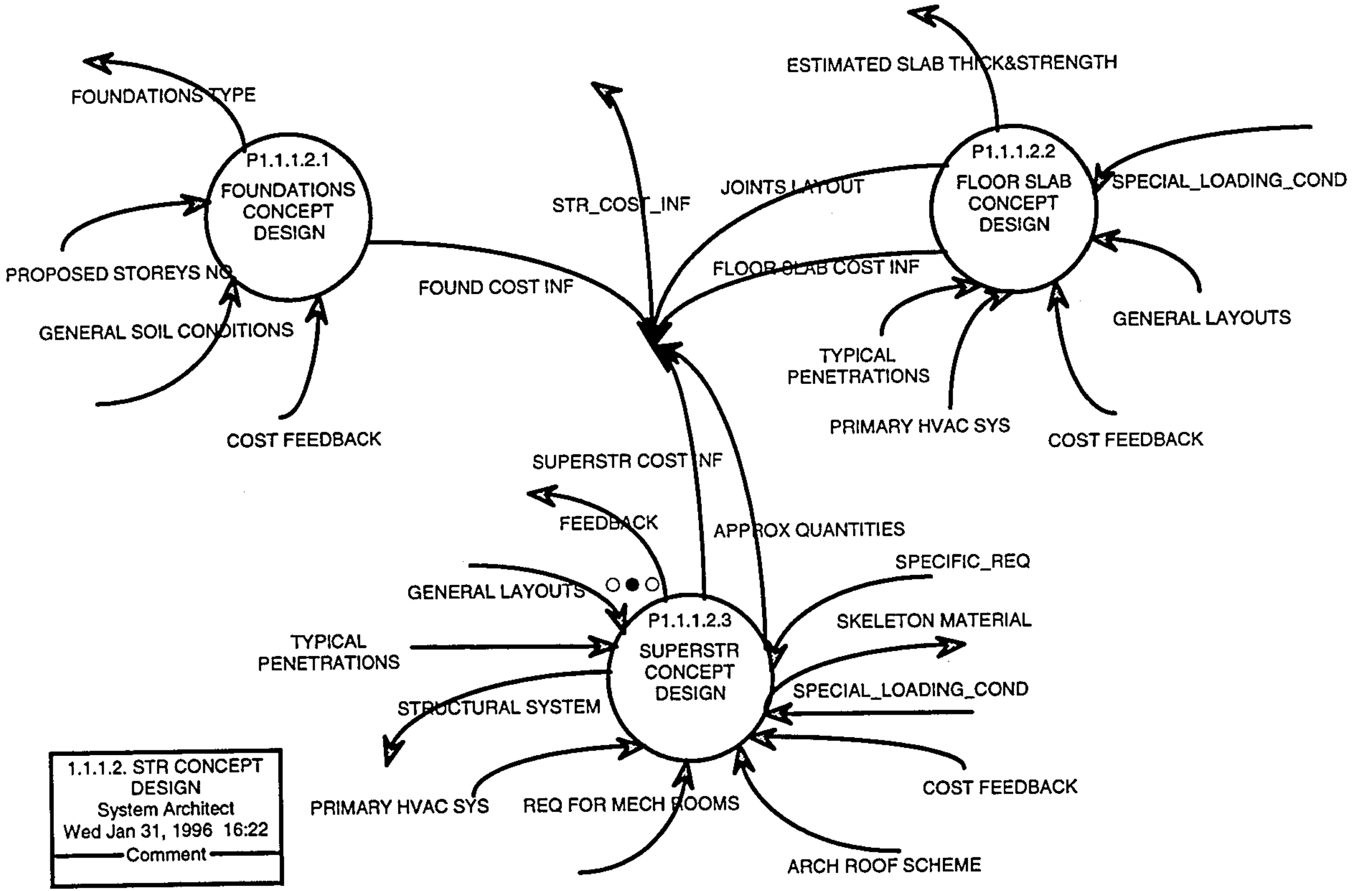


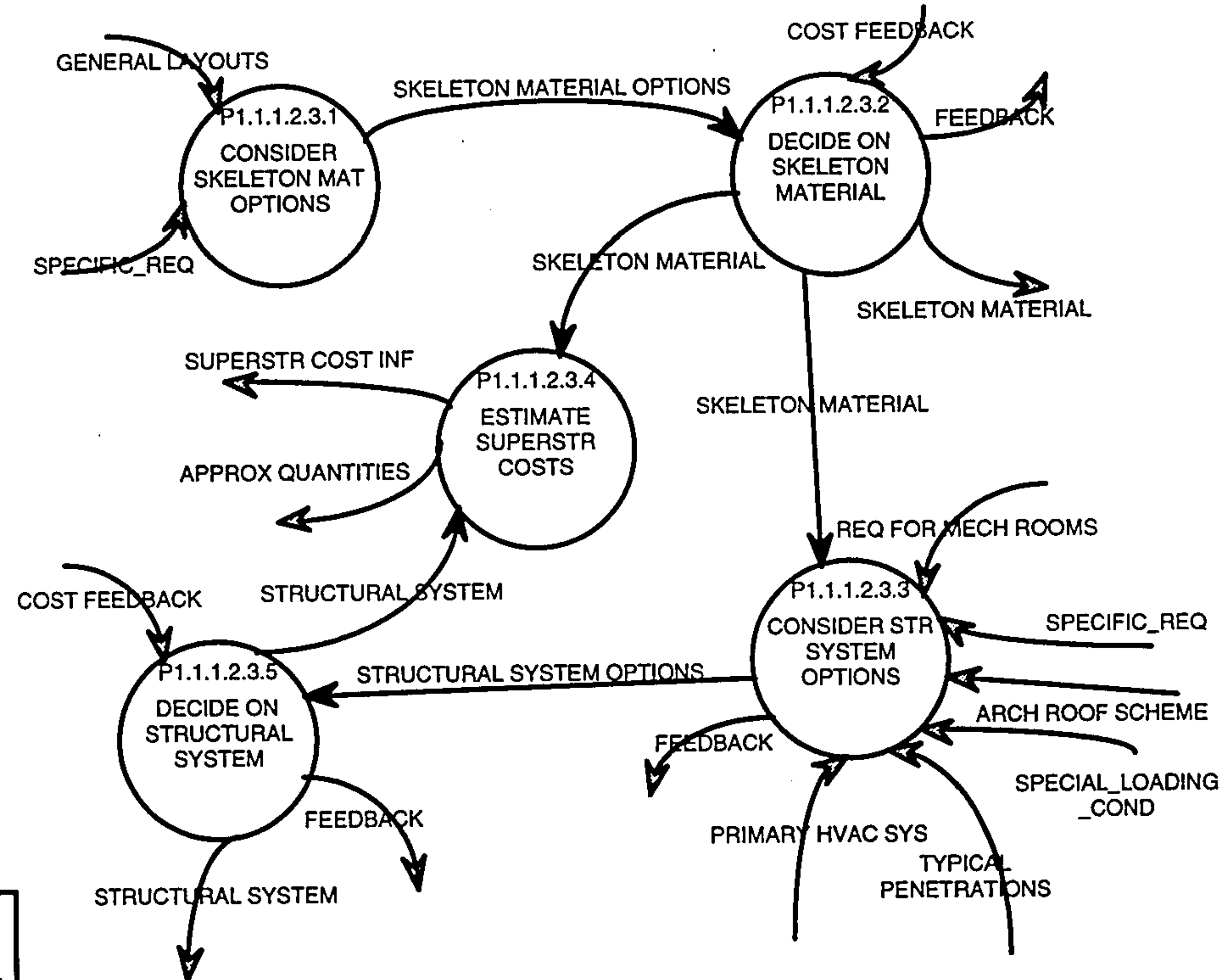
1.1.1. BUILDING CONCEPT DESIGN
System Architect
Wed Apr 17, 1996 14:21
Comment



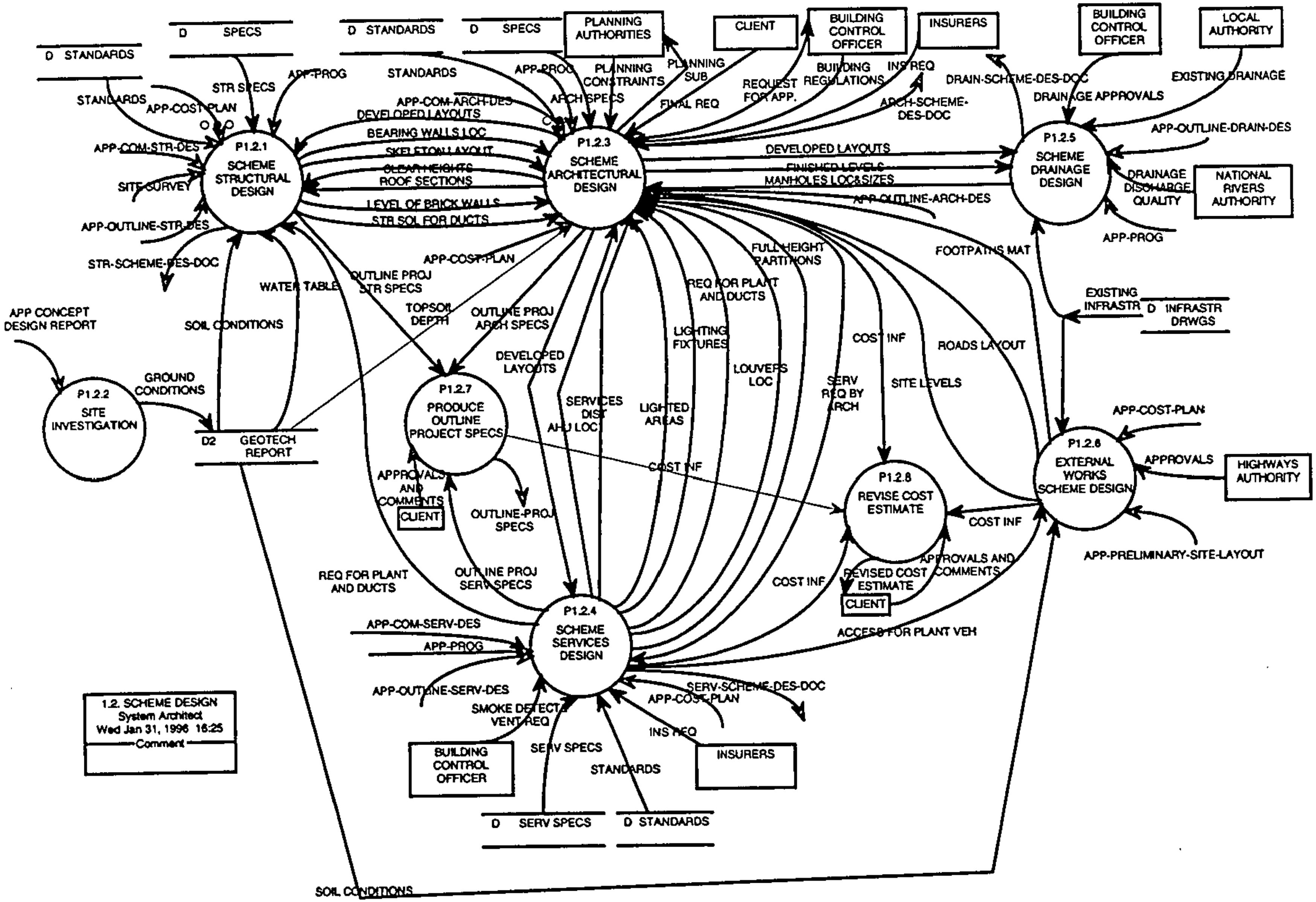
D FINISHES MATERIALS
ELEVATIONS
SITE LAYOUT
STRUCTURAL SYSTEM
ARCH_COST_INF
FEEDBACK
P1.1.1.1.1 ARRANGEMENT OF SPACES
GENERAL LAYOUTS
TYPICAL SECTIONS
GENERAL LAYOUTS
PROPOSED STOREYS NO.
SKELETON MATERIAL
REQ FOR PLANT
SPECIFIC_REQ
APPROVALS AND COMMENTS
FINISHES MATERIALS
COST FEEDBACK
P1.1.1.1.2 MATERIALS SELECTION
FINISHES OUTLINES
FINISHES_STANDARDS
ROOF TYPE
ENVELOPE MATERIALS
SKELETON MATERIAL
P1.1.1.1.4 ROOF ARCH CONCEPT DESIGN
P1.1.1.1.3 ENVELOPE CONCEPT DESIGN
APPROVALS AND COMMENTS
ARCH ROOF SCHEME
ARCH_COST_INF
ENVELOPE REQ
SKELETON MATERIAL
1.1.1.1. ARCH CONCEPT DESIGN
System Architect
Wed Oct 18, 1995 15:57
Comment


PROPOSED STOREYS NO.
REQ FOR PLANT
ARCH_COST_INF
SITE LAYOUT
APPROVALS AND COMMENTS
FEEDBACK
P1.1.1.1.1.1
APPROX FLOOR AREAS DISTRIBUTION
P1.1.1.1.1.2
CONSIDER FOOTPRINT LOC OPT
SPECIFIC_REQ
SPECIFIC_ REQ
STRUCTURAL SYSTEM
APPROVALS AND COMMENTS
PROPOSED STOREYS NO.
OPT FOR FOOTPRINT LOC
ARCH_COST_INF
FOOTPRINT SHAPE
OPT FOR FOOTPRINT LOC
P1.1.1.1.1.3
CONSIDER NO OF STOREYS
APPROVALS AND COMMENTS
PROPOSED STOREYS NO.
PROP FLOOR LAYOUTS
FLOOR AREAS
APPROVALS AND COMMENTS
STRUCTURAL SYSTEM
PROPOSED STOREYS NO.
SKELETON MATERIAL
SPECIFIC_REQ
P1.1.1.1.1.5
CONSIDER BUILDING LAYOUT OPT
PROPOSED STOREYS NO.
OPT FOR FOOTPRINT SHAPE
OPT FOR FOOTPRINT SHAPE
REQ FOR PLANT
PROP FLOOR LAYOUTS
P1.1.1.1.1.4
CONSIDER FOOTPRINT SHAPE OPT
TYPICAL SECTIONS
SKELETON MATERIAL
FEEDBACK
SITE LAYOUT
GENERAL LAYOUTS
ELEVATIONS
PROP FLOOR AREAS
APPROVALS AND COMMENTS
1.1.1.1.1.
ARRANGEMENT OF SPACES
System Architect
Wed Jan 31, 1996 16:21
Comment

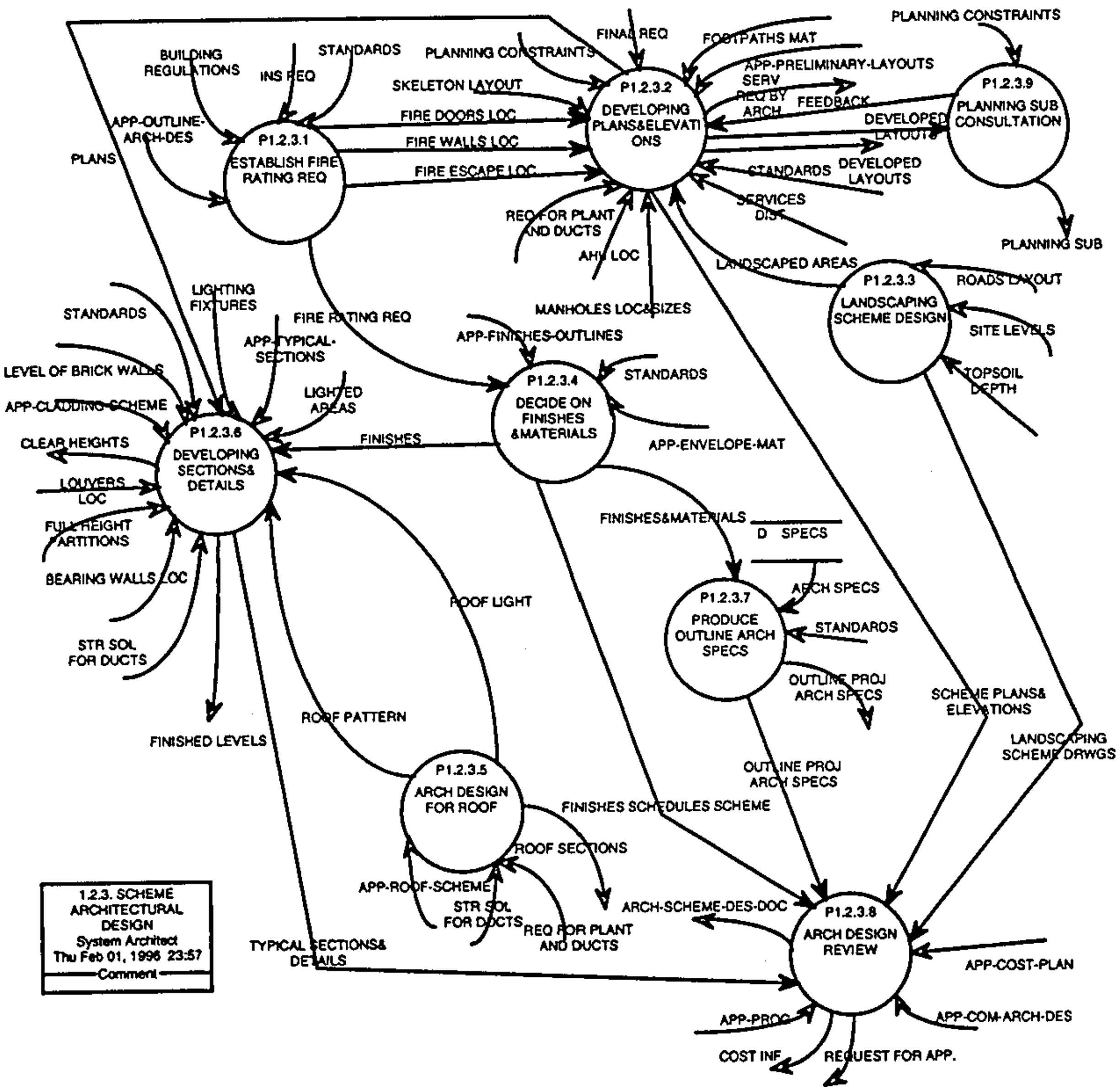

FOUNDATIONS TYPE
P1.1.1.2.1
FOUNDATIONS CONCEPT DESIGN
PROPOSED STOREYS NO
GENERAL SOIL CONDITIONS
COST FEEDBACK
FOUND COST INF
STR_COST_INF
JOINTS LAYOUT
FLOOR SLAB COST INF
ESTIMATED SLAB THICK&STRENGTH
P1.1.1.2.2
FLOOR SLAB CONCEPT DESIGN
SPECIAL_LOADING_COND
GENERAL LAYOUTS
TYPICAL PENETRATIONS
PRIMARY HVAC SYS
COST FEEDBACK
SUPERSTR COST INF
FEEDBACK
APPROX QUANTITIES
GENERAL LAYOUTS
SPECIFIC_REQ
SKELETON MATERIAL
TYPICAL PENETRATIONS
P1.1.1.2.3
SUPERSTR CONCEPT DESIGN
SPECIAL_LOADING_COND
STRUCTURAL SYSTEM
PRIMARY HVAC SYS
REQ FOR MECH ROOMS
COST FEEDBACK
ARCH ROOF SCHEME
1.1.1.2. STR CONCEPT DESIGN
System Architect
Wed Jan 31, 1996 16:22
Comment

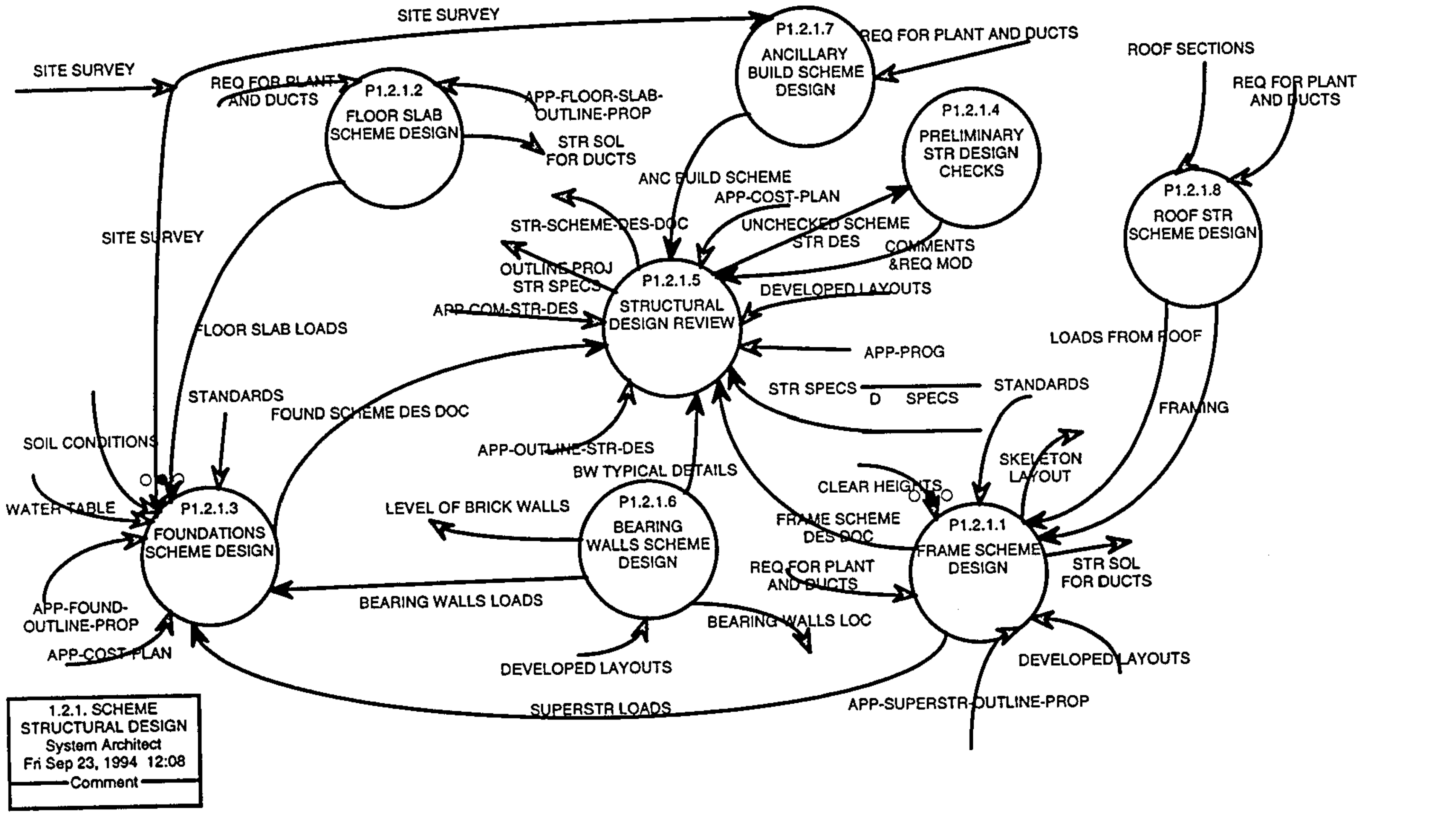


1.1.1.2.3. SUPERSTR CONCEPT DESIGN
System Architect
Wed May 24, 1995 10:30
Comment

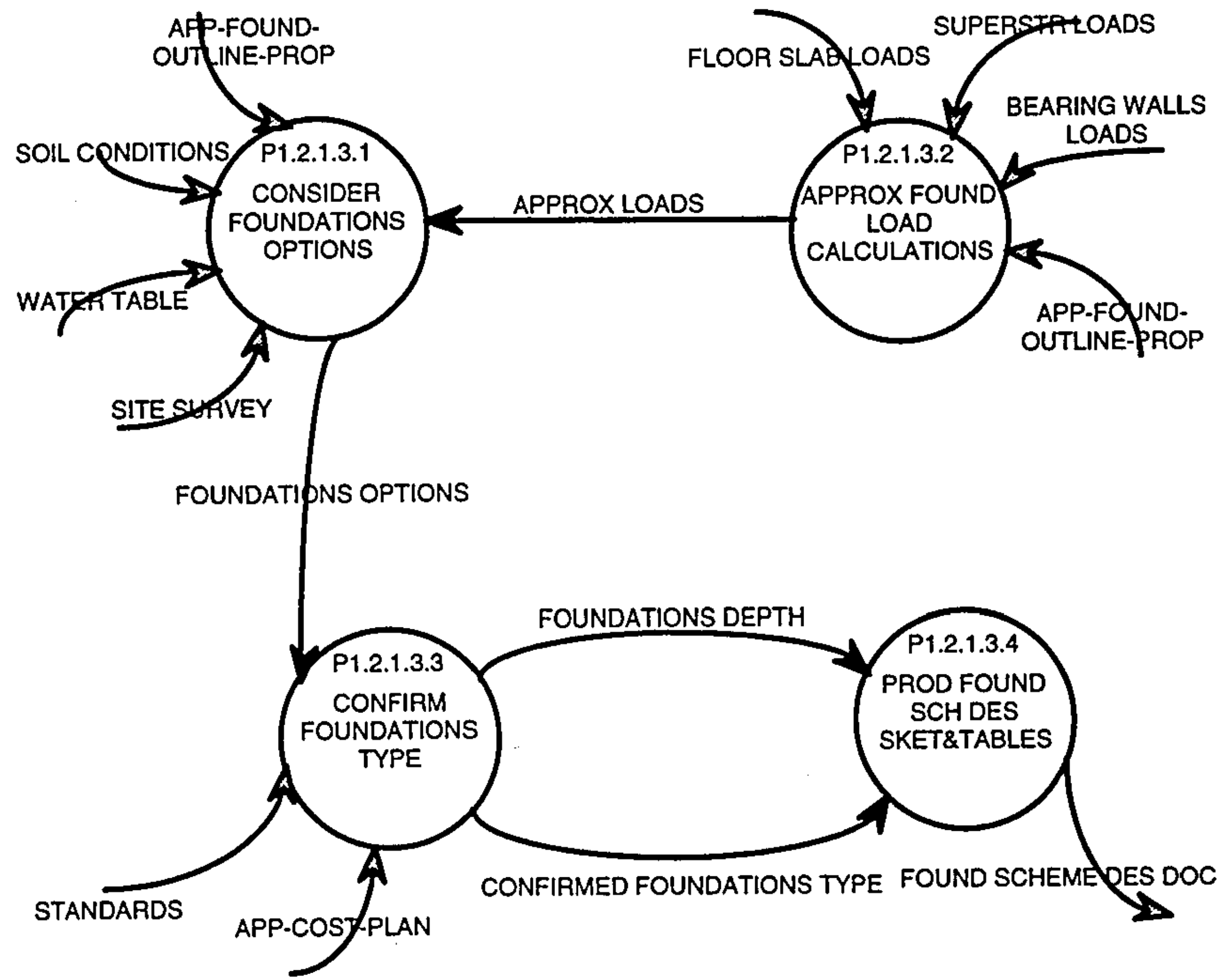


P1.2.1
SCHEME STRUCTURAL DESIGN
P1.2.2
SITE INVESTIGATION
P1.2.3
SCHEME ARCHITECTURAL DESIGN
P1.2.4
SCHEME SERVICES DESIGN
P1.2.5
SCHEME DRAINAGE DESIGN
P1.2.6
EXTERNAL WORKS SCHEME DESIGN
P1.2.7
PRODUCE OUTLINE PROJECT SPECS
P1.2.8
REVISE COST ESTIMATE
D STANDARDS
D SPECS
D STANDARDS
D SPECS
PLANNING AUTHORITIES
CLIENT
BUILDING CONTROL OFFICER
INSURERS
BUILDING CONTROL OFFICER
LOCAL AUTHORITY
NATIONAL RIVERS AUTHORITY
D INFRASTR DRWGS
HIGHWAYS AUTHORITY
D2 GEOTECH REPORT
CLIENT
CLIENT
BUILDING CONTROL OFFICER
INSURERS
D SERV SPECS
D STANDARDS
STANDARDS
APP-COST-PLAN
STR SPECS
APP-PROG
STANDARDS
APP-COM-STR-DES
SITE SURVEY
APP-OUTLINE-STR-DES
STR-SCHEME-DES-DOC
APP-COM-ARCH-DES
DEVELOPED LAYOUTS
BEARING WALLS LOC
SKELETON LAYOUT
CLEAR HEIGHTS
ROOF SECTIONS
LEVEL OF BRICK WALLS
STR SOL FOR DUCTS
APP-PROG
PLANNING CONSTRAINTS
PLANNING SUB
ARCH SPECS
FINAL REQ
REQUEST FOR APP.
BUILDING REGULATIONS
INS REQ
ARCH-SCHEME-DES-DOC
DRAIN-SCHEME-DES-DOC
DRAINAGE APPROVALS
EXISTING DRAINAGE
APP-OUTLINE-DRAIN-DES
DEVELOPED LAYOUTS
FINISHED LEVELS
MANHOLES LOC&SIZES
APP-OUTLINE-ARCH-DES
DRAINAGE DISCHARGE QUALITY
APP-PROG
FOOTPATHS MAT
EXISTING INFRASTR
APP CONCEPT DESIGN REPORT
GROUND CONDITIONS
SOIL CONDITIONS
WATER TABLE
OUTLINE PROJ STR SPECS
APP-COST-PLAN
TOPSOIL DEPTH
OUTLINE PROJ ARCH SPECS
DEVELOPED LAYOUTS
FULL HEIGHT PARTITIONS
REQ FOR PLANT AND DUCTS
LIGHTING FIXTURES
LOUVERS LOC
COST INF
ROADS LAYOUT
SERV REQ BY ARCH
SITE LEVELS
SERVICES DIST
AHU LOC
LIGHTED AREAS
APP-COST-PLAN
APPROVALS
APPROVALS AND COMMENTS
OUTLINE-PROJ SPECS
COST INF
COST INF
APPROVALS AND COMMENTS
REVISED COST ESTIMATE
APP-PRELIMINARY-SITE-LAYOUT
REQ FOR PLANT AND DUCTS
OUTLINE PROJ SERV SPECS
COST INF
ACCESS FOR PLANT VEH
APP-COM-SERV-DES
APP-PROG
APP-OUTLINE-SERV-DES
SMOKE DETECT VENT REQ
SERV SPECS
STANDARDS
APP-COST-PLAN
SERV-SCHEME-DES-DOC
INS REQ
SOIL CONDITIONS
1.2. SCHEME DESIGN
System Architect
Wed Jan 31, 1996 16:25
Comment


P1.2.3.1
ESTABLISH FIRE RATING REQ
P1.2.3.2
DEVELOPING PLANS&ELEVATIONS
P1.2.3.3
LANDSCAPING SCHEME DESIGN
P1.2.3.4
DECIDE ON FINISHES &MATERIALS
P1.2.3.5
ARCH DESIGN FOR ROOF
P1.2.3.6
DEVELOPING SECTIONS& DETAILS
P1.2.3.7
PRODUCE OUTLINE ARCH SPECS
P1.2.3.8
ARCH DESIGN REVIEW
P1.2.3.9
PLANNING SUB CONSULTATION
BUILDING REGULATIONS
INS REQ
STANDARDS
APP-OUTLINE-ARCH-DES
PLANS
PLANNING CONSTRAINTS
FINAL REQ
FOOTPATHS MAT
APP-PRELIMINARY-LAYOUTS
SERV REQ BY ARCH
FEEDBACK
SKELETON LAYOUT
FIRE DOORS LOC
FIRE WALLS LOC
FIRE ESCAPE LOC
DEVELOPED LAYOUTS
SERVICES DIST
REQ FOR PLANT AND DUCTS
AHU LOC
MANHOLES LOC&SIZES
LANDSCAPED AREAS
PLANNING SUB
ROADS LAYOUT
SITE LEVELS
TOPSOIL DEPTH
LIGHTING FIXTURES
FIRE RATING REQ
APP-TYPICAL-SECTIONS
LIGHTED AREAS
APP-FINISHES-OUTLINES
APP-ENVELOPE-MAT
LEVEL OF BRICK WALLS
APP-CLADDING-SCHEME
CLEAR HEIGHTS
LOUVERS LOC
FULL HEIGHT PARTITIONS
BEARING WALLS LOC
STR SOL FOR DUCTS
FINISHED LEVELS
FINISHES
FINISHES&MATERIALS
D SPECS
ARCH SPECS
OUTLINE PROJ ARCH SPECS
ROOF LIGHT
ROOF PATTERN
SCHEME PLANS& ELEVATIONS
LANDSCAPING SCHEME DRWGS
FINISHES SCHEDULES SCHEME
ROOF SECTIONS
APP-ROOF-SCHEME
ARCH-SCHEME-DES-DOC
TYPICAL SECTIONS& DETAILS
APP-COST-PLAN
APP-PROC
APP-COM-ARCH-DES
COST INF
REQUEST FOR APP.
1.2.3. SCHEME ARCHITECTURAL DESIGN
System Architect
Thu Feb 01, 1996 23:57
Comment

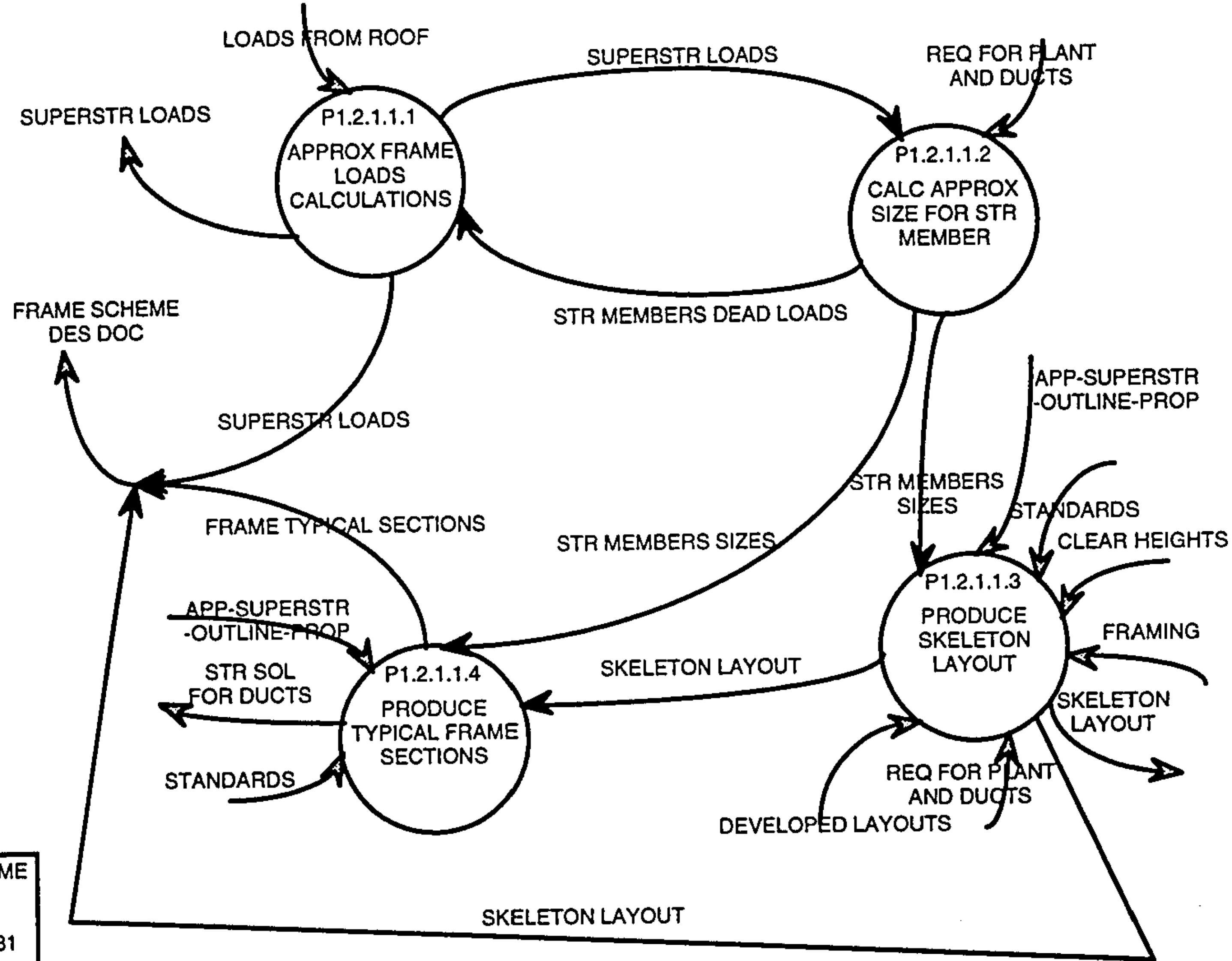


SITE SURVEY
SITE SURVEY
P1.2.1.7
ANCILLARY BUILD SCHEME DESIGN
REQ FOR PLANT AND DUCTS
ROOF SECTIONS
REQ FOR PLANT AND DUCTS
REQ FOR PLANT AND DUCTS
P1.2.1.2
FLOOR SLAB SCHEME DESIGN
APP-FLOOR-SLAB-OUTLINE-PROP
STR SOL FOR DUCTS
P1.2.1.4
PRELIMINARY STR DESIGN CHECKS
ANC BUILD SCHEME
APP-COST-PLAN
UNCHECKED SCHEME STR DES
COMMENTS &REQ MOD
P1.2.1.8
ROOF STR SCHEME DESIGN
SITE SURVEY
STR-SCHEME-DES-DOC
OUTLINE PROJ STR SPECS
P1.2.1.5
STRUCTURAL DESIGN REVIEW
DEVELOPED LAYOUTS
APP-COM-STR-DES
FLOOR SLAB LOADS
APP-PROG
LOADS FROM ROOF
STR SPECS
D SPECS
STANDARDS
STANDARDS
FOUND SCHEME DES DOC
FRAMING
SOIL CONDITIONS
APP-OUTLINE-STR-DES
SKELETON LAYOUT
BW TYPICAL DETAILS
CLEAR HEIGHTS
WATER TABLE
P1.2.1.3
FOUNDATIONS SCHEME DESIGN
LEVEL OF BRICK WALLS
P1.2.1.6
BEARING WALLS SCHEME DESIGN
FRAME SCHEME DES DOC
P1.2.1.1
FRAME SCHEME DESIGN
STR SOL FOR DUCTS
REQ FOR PLANT AND DUCTS
APP-FOUND-OUTLINE-PROP
BEARING WALLS LOADS
BEARING WALLS LOC
APP-COST-PLAN
DEVELOPED LAYOUTS
DEVELOPED LAYOUTS
SUPERSTR LOADS
APP-SUPERSTR-OUTLINE-PROP
1.2.1. SCHEME STRUCTURAL DESIGN
System Architect
Fri Sep 23, 1994 12:08
Comment



1.2.1.3. FOUNDATIONS
SCHEME DESIGN
System Architect
Wed Jan 31, 1996 16:29

Comment



1.2.1.1. FRAME SCHEME DESIGN
System Architect
Fri Sep 23, 1994 14:31
Comment

CHAPTER 6

THE SIMULATION MODEL

6.1 INTRODUCTION

This chapter describes the development of a Discrete Event Simulation Model to aid design managers in the management of the design process. The model simulates the information exchange between functional primitive design tasks during the design process and allows the user to assess the impact of changes on other design activities. Relevant change scenarios were identified from the analysis of the survey and interviews presented in section 4.4. The objective of developing the simulation model was thus to transform the developed Generic Data Flow Model from its static state to a dynamic state in order to study the impact of changes such as:

(i) Starting a design task at an earlier time based on assumed information.

(ii) 'Gate keeping' or withholding design information among design team members.

(iii) Changes in design information

(iv) Missing information

(v) The variation of the quality of information exchanged between different design tasks.

(vi) Releasing the information from different design tasks in packages or phases.

(vii) Allocating different resources to each design task and the assessment of their utilisation throughout the whole design process.

The chapter describes the production of the simulation model, its components and the data required to build it. It describes also the testing and verification process and presents sample results which reflect the main features of the model.

6.2 BACKGROUND TO THE PRODUCTION OF THE SIMULATION MODEL

The Simulation Model was developed on an incremental basis, starting with a basic model followed by expansion to incorporate the required features. This approach not only allowed testing of the model after each step in its development to facilitate debugging but also gave some flexibility to building in the required features of the model as they were identified by the ongoing dialogue with designers.

The simulation model is based on information from the Data Flow Model. Therefore the initial simulation model was based on a prototype data flow model which includes all the features and components of a full scale model (Figures 6.1 and 6.2). After thorough testing of the model at the prototype level, the full scale Simulation Model was constructed.

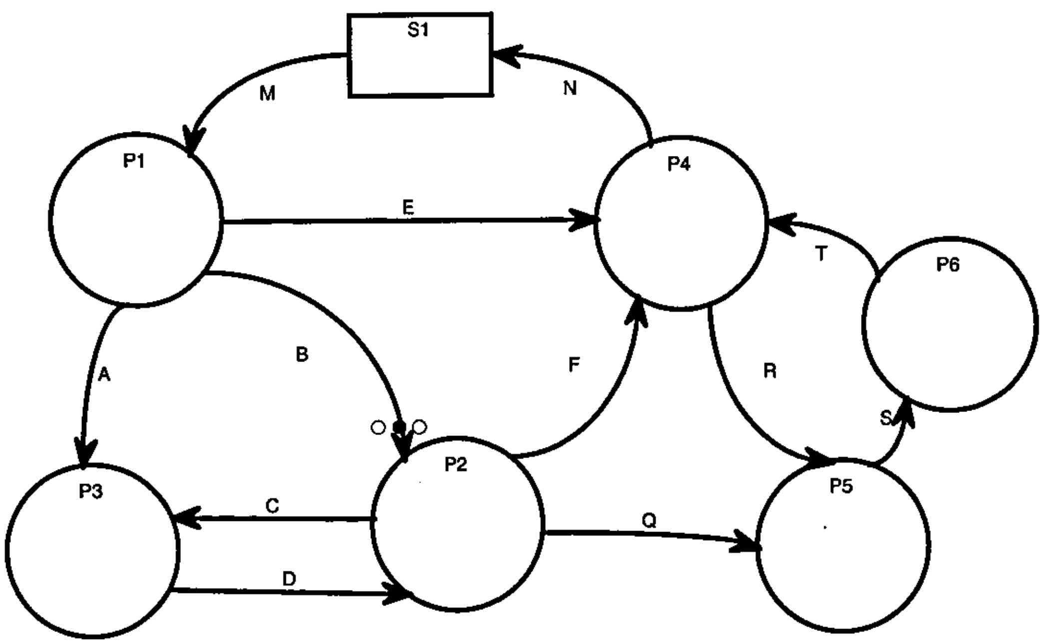


Figure 6.1 Level 0 of the prototype data flow model

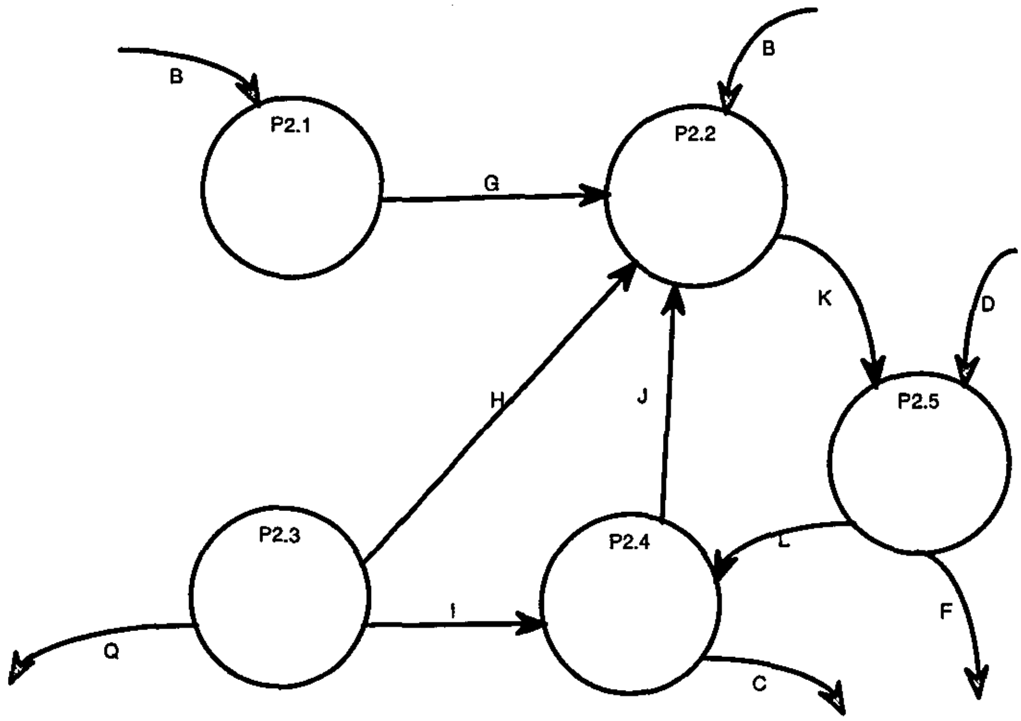


Figure 6.2 Level 1 of the prototype data flow model (process P2 is decomposed into sub-processes)

6.3 DATA REQUIRED TO RUN THE SIMULATION MODEL

Since the building design process is of an iterative nature, it was necessary after constructing the data flow model to identify the loops of iterative design tasks at the level of the functional primitive tasks (functional primitive tasks are tasks at the lowest level of the data flow diagrams). The identification of these loops was carried out using matrix analysis techniques. In this technique, the functional primitive tasks (FPTs) of the data flow diagrams (DFDs) are arranged in a square matrix. Each FPT is represented by an identically labelled row and column. Within each cell in the matrix, a mark in row *i* column *j* represents an information dependency for row *i* from column *j*. These information dependencies had been identified from the data flow model. Software written by Steward based on his TERABL program (DVS for the version running under DOS and PSM for the version running under Windows) was used to 'partition' the matrix into diagonal blocks, each block representing a group of design tasks which fall in the same iterative loop. For example, if task (A) requires information from task (B), (B) requires information from (C), and (C) requires information from (A), then tasks (A), (B), and (C) fall within the same iterative loop.

Durations and resources were assigned to each design task identified in the data flow model. The data flow model and the matrix partitioning represent the 'front end' to the Discrete Event Simulation Model as illustrated in Figure 6.3.

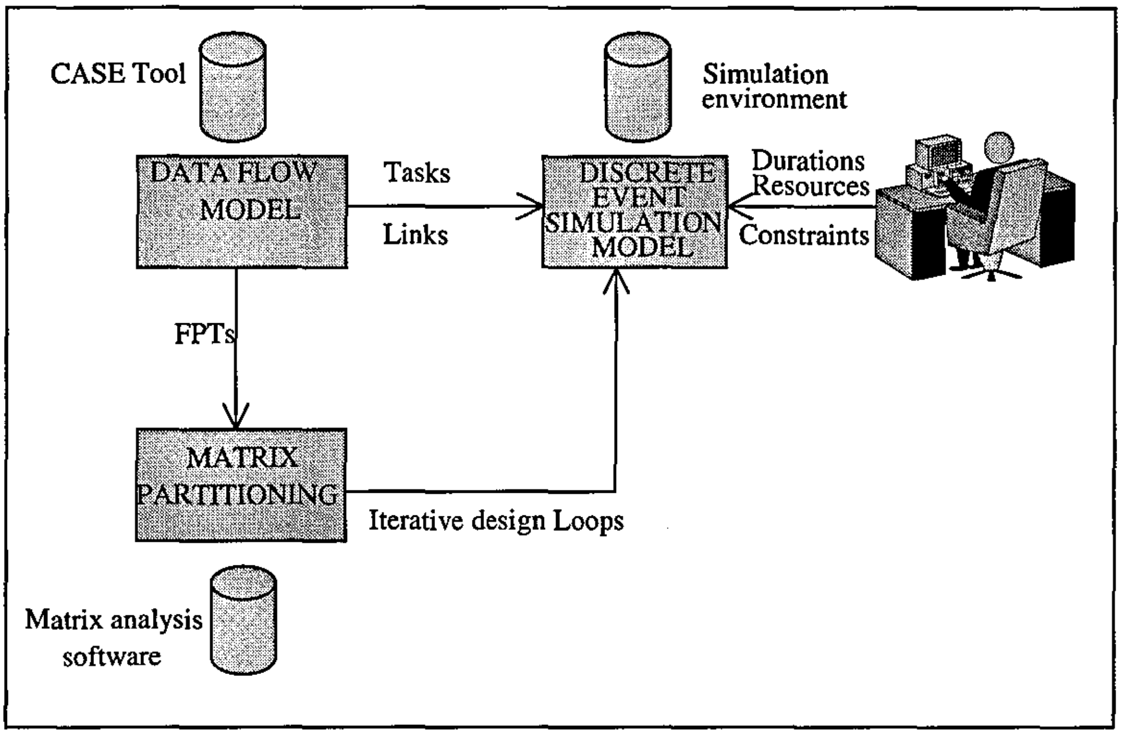


Figure 6.3 Data required to run the Simulation Model

6.4 FEATURES OF THE SIMULATION MODEL.

The data required to run the simulation model are based upon the information links between the design tasks identified in the Data Flow Model together with the grouping of design tasks within the iterative loops identified from the matrix analysis and the duration and resources requirements for every design task identified by the user. This is shown diagrammatically in Figure 6.3. The simulation model has been constructed using a simulation environment called 'Genetik' which is a hybrid of simulation languages and simulation data driven packages. Details of the selection of this simulation technique are included in section 4.6. The main features of the Simulation Model developed by the writer are:

(i) It is able to run in either deterministic or stochastic mode. The user chooses the required mode from a pop-down menu, and the model automatically calls the appropriate routines according to the user's choice.

(ii) The quality of design information is simulated by allocating attributes to information links between different design processes and between design processes and external sources of information (e.g. Client, local authorities, etc.). Before running the simulation, the user enters the value of a global integer representing the cut-off value for the information quality. If, during the simulation, the value of any attribute is less than the cut-off value, the associated design tasks are not performed and an appropriate message appears on the screen.

(iii) Links between different tasks may be switched 'on' or 'off'. If a switch is set as 'off' a design task may proceed based on estimated data inputs. Subsequent tasks are tagged 'conditional'. When the design task receives the finalised information, a second iteration is carried out on the conditional design tasks, with a reduced duration based on a percentage (chosen by the user) of the first iteration.

(iv) The model allows for phased release of information. This introduces an important refinement whereby a task with several outputs releases information in a pre-defined order (defined in a data dictionary), allowing some dependent tasks to start earlier.

(v) The model allows the simulation of 'gate keeping' of information, a communication problem associated with individuals who retain design data instead of making it available to the design team. Information links in the model have 'gates' which may be opened or closed. When a gate is closed, a variable is assigned to links representing the time lapse between finishing the task and releasing the information.

(vi) The model allows the simulation of resources deployment and utilisation (up to eleven types of resources) throughout the design process. This includes architects, designers, engineers, managers and draftsmen for each discipline. Before running the simulation, the user identifies the resource levels for each resource type allocated for the project together with the resources requirement for every design task. If, during the simulation, a task receives its requisite information but the required resources are not available, an appropriate message will appear on the screen and the commencement of this task will be deferred until other task(s) are terminated and release their resources. This may result in increasing the total project duration and the design manager has to assess the trade-off between increasing the number of allocated resources or delaying the design completion date.

(vii) For tasks which are within iterative loops, the model maps the tasks in the loop to find a design task that has received information inputs from tasks outside the loop (after carrying out checks for 'switches', 'gates', quality and phased release of information), and can therefore start the first iteration of the loop. There is also the option of the user nominating the design task that initiates the loop. The resources required for all the design tasks comprising the loop should be available before starting this loop. The time taken to complete the first iteration may be taken, according to the user's selection from a pop down menu, as the duration of the task that initiates the loop or the longest duration of a task within the loop (i.e. the rest of the iterative tasks within the same loop are assumed to be interacting concurrently). This represents the rapid, complex, interactive period of the schematic design stage. The second iteration of a loop can start when all remaining tasks in the loop fulfil all conditions and the available resources are adequate. The duration of the second iteration is specified by the user.

(viii) The results of running the simulation may be displayed in five forms:

(a) A bar chart showing the start and end of every task and of every iteration for every iterative task with different colours representing different iterations.

(b) Icons showing the change of state of every task as the simulation clock advances e.g. state I - ready to start, state II - end of first iteration.

(c) A table for the design tasks showing the start and end of every task and of every iteration for every iterative task.

(d) Icons representing the change of state for every resource from "busy" to "idle" or vice versa as the simulation clock advances.

(e) Histograms for every resource type showing the utilisation of this resource throughout the whole design process.

6.5 THE GENETIK DESIGN SIMULATION MODEL

6.5.1 Design of the Model

The simulation model has been constructed following the Three Phase Approach to Discrete Event Simulation because of its flexibility and suitability to represent design activities and simulate data from a data flow model. This approach includes two different types of events:

- B-events (Bound or Book-keeping events): These events occur directly by the simulation executive whenever their scheduled time is reached.

- C-events (Conditional or Co-operative events): The occurrence of these events depend on the co-operation of different classes of entity or on the satisfaction of specific conditions within the simulation.

The possibility of using 'Genetik' to identify the loops of iterative design tasks has been investigated. It was simple to make the model differentiate between non iterative tasks and iterative tasks. However, to allow the model to distinguish the number of iterative loops and the tasks within each loop would necessitate programming routines with a logic similar to that of the Design Matrix software. Whilst there was no evidence that programs written in the Genetik modelling language (with or without additional routines written in a suitable programming language) were incapable of being produced to perform this task, Genetik was not considered the most suitable language to write these routines. Moreover, it was considered that, within the time available, it would be better to focus on other issues and utilise the available software.

The simulation model was designed to be independent of specific data. The data required may be captured from a data flow model and a design structure matrix and input in the relevant data entry tables of the simulation model. This is irrespective of a particular data flow model or a design structure matrix. There are no limitations for the number of tasks, information links, and/or resources to be simulated as the developed routines relate to variables representing these numbers. These variables would change according to the relevant number used in different sets of data. The Action routine which has been developed to display a bar chart of tasks was designed to include a scroll bar which relates to a variable representing the number of tasks. It is created automatically if the number of tasks are greater than the physical size of the screen. The size of the thumb mark which appears in the scroll bar is also determined by the routine according to the number of tasks. Hence, the model is generic and may be used to simulate any process represented by a data flow model with minor adjustments. Therefore, although the Genetik Simulation Model has been developed to simulate the Conceptual/Schematic stages of the design process, the way the model has been set up allows extending its application to the detailed design and construction stages with minor adjustments. Chapter 8 includes examples of the model being used for the simulation of detailed design. The model has also been used successfully to simulate the information flow within the estimating and tendering process in another research project.

6.5.2 Description of the Genetik Design Simulation model

The simulation modelling environment 'Genetik' was used to construct the model. Genetik is a hybrid of simulation languages and simulation data driven packages. This feature is attributed to a vocabulary of control statements that Genetik includes. A Genetik model is constructed out of a number of building blocks known as Units or Modules. The basic structure and functionality of these units are already defined within Genetik, leaving the model builder the job of defining the detail in each unit. When the model runs, these modules are combined according to the specified logic. Each building block has its own editor which is used to edit or amend data or which prompts the model builder to perform certain actions.

The main modules used in building the Genetik Simulation Model fall into four broad categories:

- Modules relating to Data
- Modules relating to Logic
- Modules relating to Pictures
- Modules relating to Interactions

6.5.2.1 Modules relating to Data: TABLES, VARIABLES, ENTITIES, LISTS

There are two ways of storing data in the Genetik Simulation Model. One way is by using a module called TABLE. Within the model, different tables are used to hold information about design tasks based on data from the Data Flow Model (such as tasks names and dependencies) and the Matrix Partitioning (such as loops of iterative design tasks), in addition to other data such as resource requirements, durations and constraints imposed by the user. Other tables are used to capture information about the change of state for every resource type as the simulation runs to be used in plotting histograms of resource utilisation. Examples of a table containing data about durations and resources and another table containing information about links between different design tasks in the Generic Data Flow Model are included in Appendix V.

ENTITIES and LISTS are special types of tables. The Data in LISTS should be members of pre-defined ENTITIES. The data in ENTITIES include the description and colour of every entity and is used when displaying entities on the screen. Within the context of this model, Entities represent different types of resources and Lists represent a list for every resource type in each of the "busy" state and the "idle" state. These lists are filled automatically as the simulation runs.

Some data items cannot usefully be expressed by a Table since they are just individual values. This data is stored in a Module called VARIABLE. The reason this term is used is that the stored value may need to be changed when the model is run. VARIABLES may be GLOBAL or LOCAL. A global variable is one whose value is accessible whenever it is used within the model (i.e. within any module) whereas the value held in a local variable can only be accessed within the module where it was given the value. The value of a local variable can then be changed within one unit without fear that this change will affect other units. Variables can also be classified by the type of data that they contain. The different types may be Integer values, Real values, Text values and Table Row Pointers (variables representing rows in tables). A more sophisticated variable type is an Action variable, which is a variable that executes certain tasks whenever it is called by other modules.

6.5.2.2 Modules relating to Logic

These modules consist of a sequence of Statements or Routines developed by the writer in the Genetik programming language. When the model is run the computer works through the Statements, one by one, executing each of them in turn. One line may change the value in a TABLE or VARIABLE, another may call other module(s) or may display data on the screen.

Modules relating to Logic are:

- ACTION UNITS
- C-EVENTS
- B-EVENTS
- UTILITIES

ACTION UNITS

The most important module relating to Logic is the ACTION UNIT. Action Units drive the model and are the means by which the model builder controls the flow of events. An essential ACTION UNIT required for the model to run is STARTUP. Whenever the command RUN is given to the programme, Action Unit STARTUP starts executing the logic. Examples of other Action Units developed by the writer are actions to initialise the simulation, execute the simulation, run in either stochastic or deterministic mode, select the duration of the first iteration in loops (the duration of the task that initiates the loop or the longest task duration within the loop), allow interaction with the model by editing tasks durations and resource requirements through the results menu, trace conditional design tasks (tasks performed based on assumed information), draw bar charts for the results, produce tabular reports, draw icons for design tasks, display the change of each resource state as the simulation runs, draw histograms for each resource utilisation and an action unit that holds a list of C-Events that will occur. Some actions can be invoked through the results menu bar created by the writer, others are called by other modules within the model.

An example of an Action Unit to draw a bar chart of the results is included in Appendix V.

C-EVENTS

C-EVENTS (Conditional or Co-operative events) are events whose occurrence depend on either the co-operation of different classes of entity (like design tasks) or on the satisfaction of specific conditions within the simulation. Examples of C-Events developed by the writer are C-Events that start design tasks, either conditionally or unconditionally, on satisfaction of all conditions and availability of resources, schedule time lapsed between completing a design task and releasing information required by other team members (gate-keeping of information) and phase the release of information from a task to a subsequent task(s). There are different C-Events for running the simulation in either deterministic or stochastic mode and for different types of design tasks (loop tasks or non loop design tasks). For every C-Event an associated B-Event is scheduled with a certain duration.

An example of a C-Event to resume the first iteration of a conditional loop task (i.e. a loop task that commenced based on assumed information) on receipt of all requisite information and availability of all resources is included in Appendix V.

B-EVENTS

B-EVENTS (Bound or Book-keeping events) are events which occur directly by the simulation executive whenever their scheduled time is reached i.e. on satisfaction of

all conditions for the associated C-Event. At any point in time when the simulation is running all the forthcoming B-Events are held in an ordered list, called the B-Event list. This list is in the form of a temporary table in the TABLES module. The B-Events list gives the names of all forthcoming B-Events and the time at which they will occur in the future; they are ordered in time sequence. The duration of each scheduled B-Event is specified by the user in the associated C-Event. For example, the duration of the B-Event associated with starting a design task is the duration required to perform this design task. Within the context of this model, B-Events are mainly related to changing the state of design tasks (e.g. changing the state from started to finished), changing the state of each resource from "busy" to "idle" or vice versa, and calculating the start and end times of each design task.
An example of a B-Event to complete design tasks is included in Appendix V.

UTILITIES

UTILITIES are modules which when called return certain parameters based on other given parameters. In addition to over 350 Genetik built-in utilities, other utilities have been produced by the writer in Genetik code to perform certain functions. Examples of utilities developed by the writer are utilities to check the dependency of any design task on other task(s), to test the information quality condition on every link between different design tasks and between external sources of information and design tasks, to check the availability of required resources for every design activity, to transfer different types of resources from the "idle" LIST to the "busy" LIST or vice versa, and to cross reference values between different tables.
An example of a Utility to check the dependency of any design task is included in Appendix V.

6.5.2.3 <u>Modules relating to PICTURES</u>

These modules are:

- PICTURES
- SCREENS
- WINDOWS
- ICONS

PICTURES

A PICTURE can be imagined as a large piece of paper. It is so large that for all intents and purposes it is infinite, bounded only at the left and bottom edges. It provides a surface on which objects can be drawn and written.

SCREENS

A SCREEN is the means by which PICTURES are presented. It is precisely the same size as the computer screen, and thus the computer can display one and only one SCREEN at any one time. Like a cinema screen, a Genetik screen is not a surface for drawing on but an area for displaying or projecting PICTURES.

WINDOWS

WINDOWS are the mechanisms which link SCREENS and PICTURES. A WINDOW is 'positioned' onto a screen and 'looks at' a specified part of a PICTURE. WINDOWS can vary in size but because they must be part of a screen they cannot exceed the size of a screen. By using WINDOWS, parts of more than one picture can be shown on one screen, and the part of each picture required to be shown can be specified.

ICONS

An ICON is considered as a 'sub-picture' which can be written to a PICTURE. An ICON has an origin which is used to locate it when it is written to a PICTURE.

Within the context of this model, fourteen SCREENS have been defined by the writer. The first SCREEN comprises a WINDOW containing the results menu (developed also by the writer), a second WINDOW displaying the current simulation time, and a third blank WINDOW where pre-defined ICONS of design tasks are displayed when the model runs.

The second SCREEN comprises a blank WINDOW where a bar chart representing a schedule of the design tasks is displayed whenever the appropriate action is invoked from the results menu.

The third SCREEN comprises a WINDOW displaying the current simulation time and a WINDOW displaying twenty two lists for the eleven resource types, each in either a "busy" state or an "idle" state.

The remaining eleven screens each represent a histogram for the resource utilisation of every resource type.

6.5.2.4 <u>Modules relating to INTERACTIONS</u>

Interactions between the user and the simulation model are very important as they allow the user to select a particular action from several options or to edit or change data or parameters to assess the impact of different scenarios. Interactions can be either through Function Keys or Menus. An FKEYMODE module allows the model builder to define certain Actions that will be invoked when the user activates pre-defined function keys. A MENU module is a list of options each of which must have an ACTION UNIT associated with it. An alternative way of defining ACTION UNITS associated with menu options is by defining a list of ACTIONS in the TABLES module for each menu and sub-menu option. The menu options created by the writer allow the user to initialise the simulation, execute the simulation in either deterministic or stochastic mode, edit design tasks and resource requirements, select duration for first iteration of loop tasks, display the results in either icon format, bar chart format, resource occupation format, histogram format, or tabular format and finally exit from the results menu. The model must wait for an option to be selected and no further options can be selected until the menu is invoked again.

The Simulation Model developed by the writer is composed of some 140 modules. Details of each module are included in Appendix IV.

6.5.3 Operating the simulation

Whenever the simulation is running, all the forthcoming B-EVENTS are held in an ordered list, called the B-Event list. This list gives the names of all forthcoming B-Events and the time at which they will happen in the future. The B-Events are ordered in time sequence.

The simulation proceeds as follows:

- STEP 1: Simulation time is advanced to the next B-Event, the relevant event is picked up and this entry is deleted from the B-Event list

- STEP 2: The computer diverts to the relevant EVENT module.

- STEP 3: The logic in the EVENT module is written to ensure that the appropriate changes are made to the states of certain entities (the design tasks). Additionally a new B-Event may be scheduled into the B-Event list.

- STEP 4: At this stage, since various entities (design tasks) have changed their state, it may be possible for a C-Event to occur, i.e. the required co-operating entities (other design tasks) may all be in the required states. Thus the computer invokes each C-Event module, in turn, and the appropriate tests are made to check if the relevant entities are in the relevant states for that C-Event to occur. If they are, that C-Event is executed; this involves further changes to the states of certain entities. If not, control is returned to STEP 1, and the next event in the B-Events list is activated. If any C-Event module can be activated (because the conditions are fulfilled), all the other C-Event modules must be re-invoked since states will have changed within one C-Event which may make it possible to activate other C-Events.

The steps in running the simulation model are illustrated in the flow chart shown in Figure 6.4.

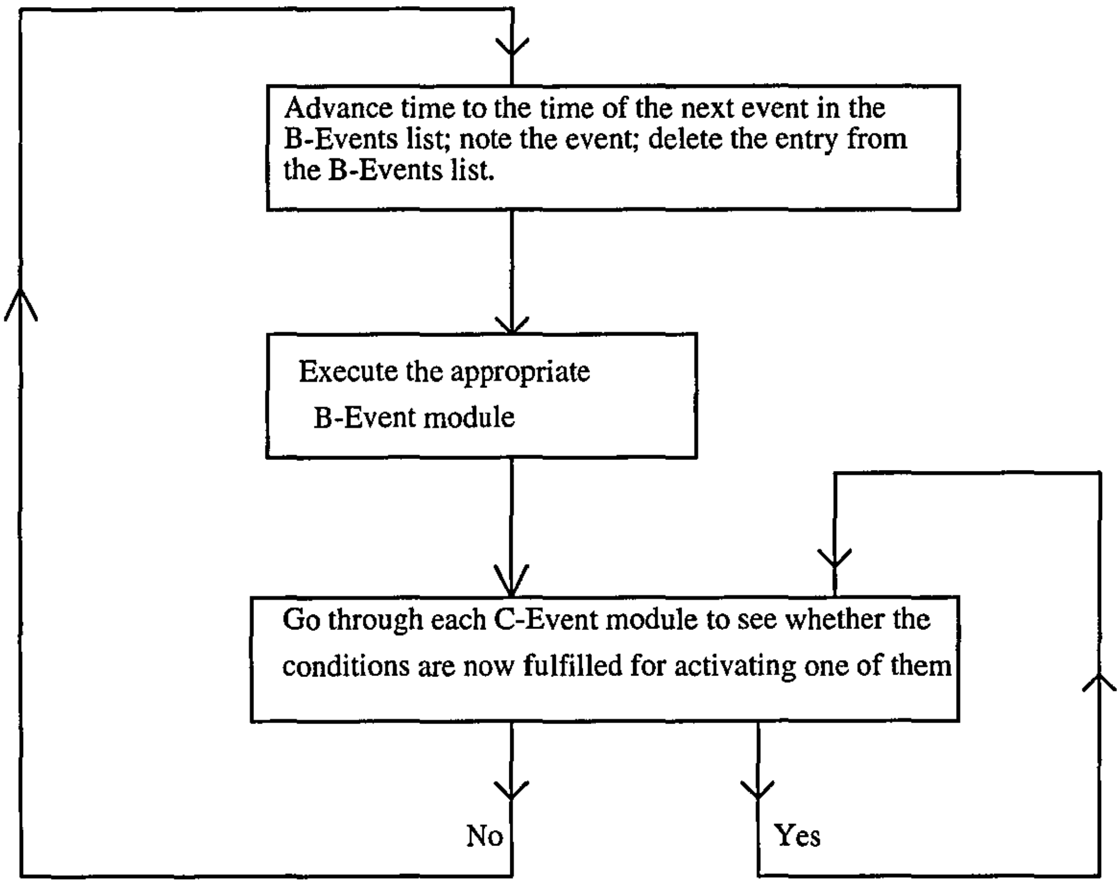


Figure 6.4 Flow chart showing the steps in running the Simulation Model

6.6 VERIFICATION OF THE SIMULATION MODEL

Sample data from an early version of the Generic Data Flow Model was input into the Genetik Simulation Model to test and verify the routines developed and the operation of the model. This was undertaken on an incremental basis with manual checks of the output to ensure the accuracy and robustness of the model.

The iterative design loops have been identified using the TERABL software for matrix analysis developed by Steward.

After identifying the loops and assigning durations and resources to each functional primitive task, data from the data flow model and the matrix analysis have been entered in the appropriate tables in the Genetik Simulation Model and the model was set to run.

Figure 6.5 shows samples from the results produced by the model in bar chart format. The task numbers represent the same task numbers as those in Generic Data Flow Model and the matrix analysis. The numbers on the bars represent the start and finish time for every task or for every iteration of a task. The tasks with single red bars represent the non iterative design tasks. Tasks with blue and red colours represent iterative design tasks in loops; the blue bar representing the first iteration and the red bar representing the second iteration. The letter 'C' after the finish time indicates that the task has been completed or completed its first iteration on 'conditional' basis. This is due to starting the task based on assumed information or on information from other task(s) which finished on 'conditional' basis.

Figure 6.6 shows the results produced by the model in icon format. The numbers underneath each icon represent the state of each task at any particular simulation time (different states are explained within Appendix IV of the thesis). The letter 'C' appearing below any task indicates that the task is running (or going to run) on conditional basis.

Figure 6.7 shows a simulation run that has stopped due to insufficient quality of information. The message displayed shows the task which could not proceed and the task providing the information.

Figure 6.8 shows how resources are simulated. For each discipline, there is a list for each type of resource in its idle and busy states. As the simulation runs, the state and number of each type of resource at any particular time is displayed. Any task which cannot start due to insufficient resources is also displayed. Figure 6.9 shows a sample for a resource utilisation histogram produced by the model for civil/structural designers.

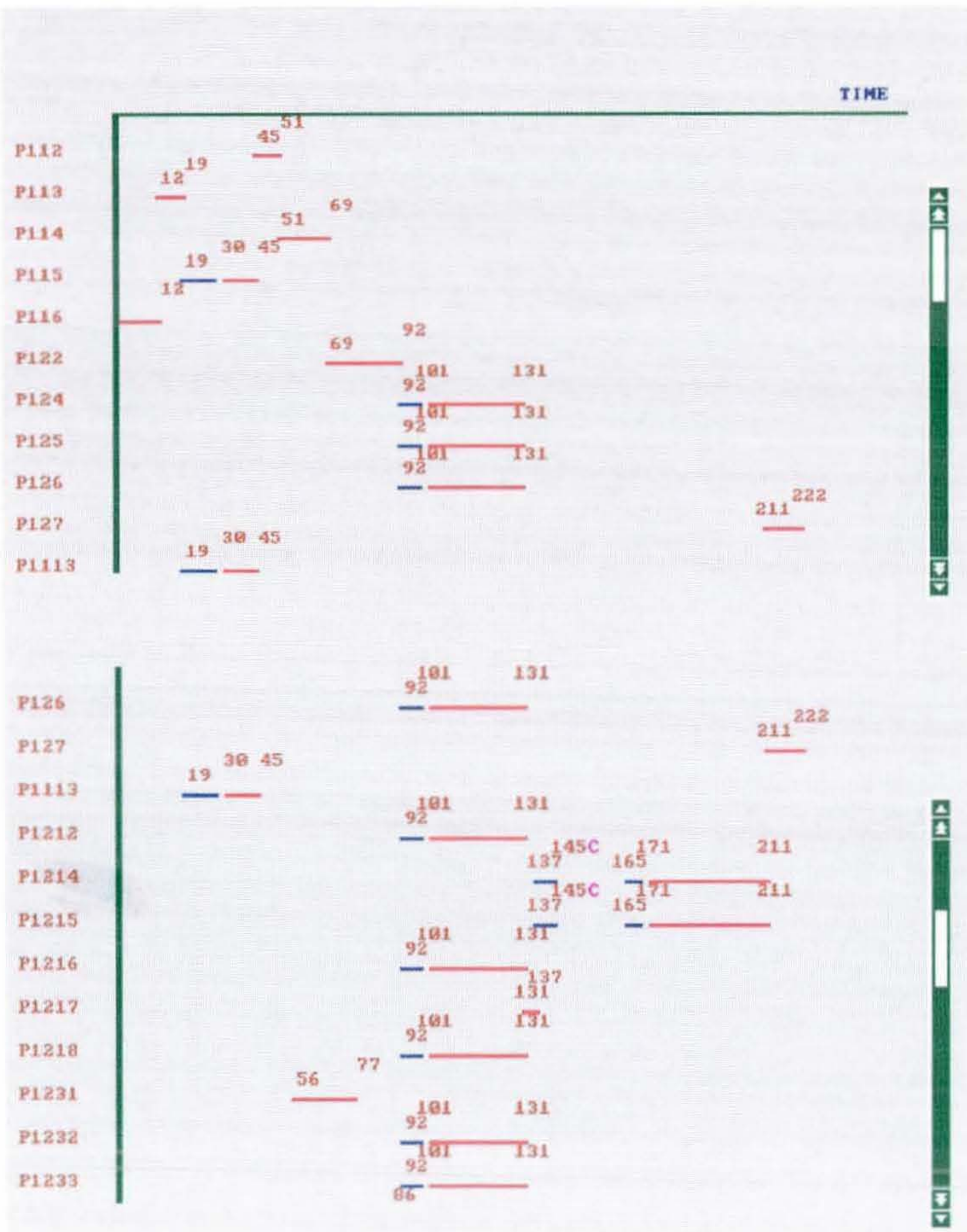


Figure 6.5 Output from the Simulation Model in bar chart format

Figure 6.6 Output from the Simulation Model in icon format

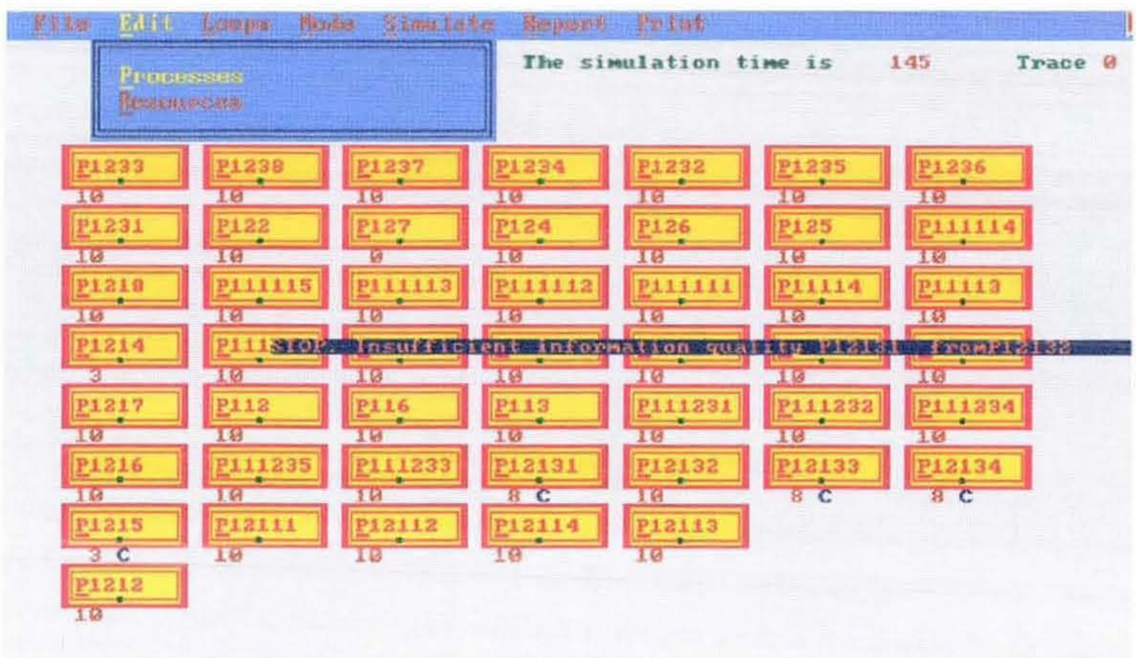


Figure 6.7 Output from the Simulation Model when simulating information quality

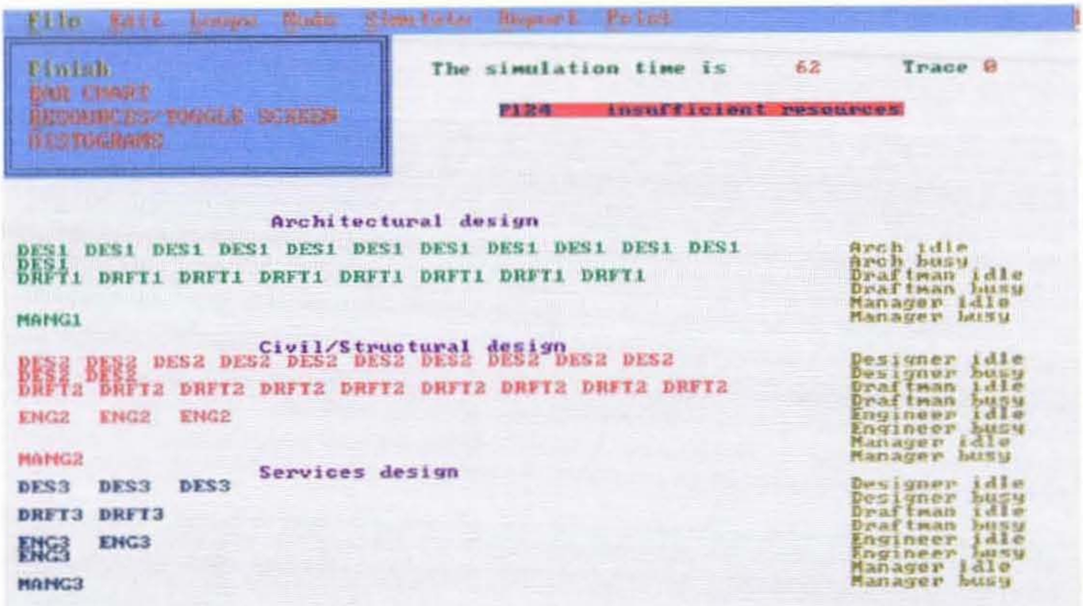


Figure 6.8 Output from the Simulation Model when simulating resources

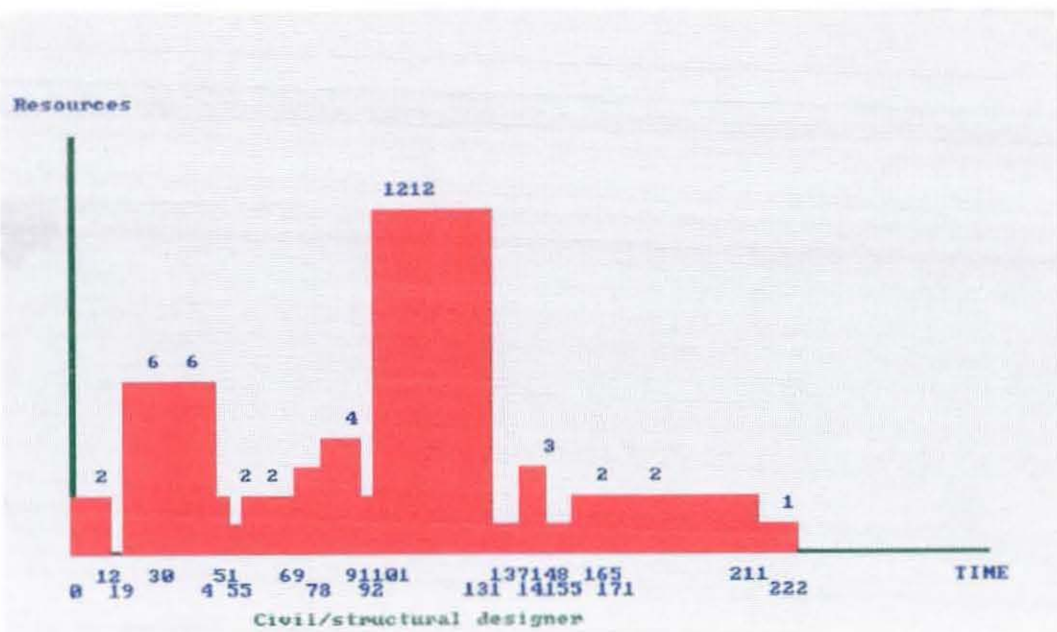


Figure 6.9 Output from the Simulation Model in the form of a resource utilisation histogram

The simulation model has been also used successfully to simulate the information flow within the estimating and tendering process in another research project. The results were validated against other commercial simulation products which confirmed the accuracy and robustness of the model.

The Simulation Model was validated through demonstrations held by the writer to industry professionals within Ove Arup and Partners, Nottingham branch; Ove Arup and Partners, Birmingham branch and AMEC Design and Management. The objective of the demonstrations was to acquire feedback on the contribution that the application of the simulation model will offer to improve the management of the design process across the different stages of design. This is explained in more detail in Chapter 8.

6.7 THE OPERATION OF THE MODEL

The model operates in the following manner:

(i) After identifying the functional primitive tasks of the Data Flow Model and the iterative design loops in the Design Structure Matrix the user will select the TABLES module of the main 'Genetik' menu to input the data which is necessary for running the model in the relevant tables. This includes the following sets of data:

  (a) Data about the design tasks such as the tasks numbers, durations, whether being iterative or non iterative and the co-ordinates of their icons which will appear in the results screen. Data about tasks durations may also be input through the 'Edit' sub-menu of the results menu to allow interaction with the model.

  (b) Data about the information sources such as the Client, building control officer, etc.

  (c) Data about the information dependencies between the different design tasks or between a design task and an information 'source'. The table which captures this data includes also information about the attributes of these links which require experimentation under different scenarios. The different attributes represent the quality of information exchanged, the status of the 'switch' on the information link (which represents the case of assuming the information), the status of the 'gate' on the information link (which represents the case of 'gate keeping' of information) and the percentage of completion required for a task to release the requisite information (which represents the phased release of information).

(d) Data about the loops of iterative design tasks including the tasks comprising each loop.

(e) Data about the resource requirements for each design task including the resource type and number. This data may also be input through the 'Edit' sub-menu of the results menu to allow interaction with the model.

(f) Data about the number of different types of resources on the corporate level.

It must be noted that in the case of simulating data from the Generic Data Flow Model the user will input only data about tasks durations and resources in addition to any necessary amendments in the other tables due to adjustments in the Data Flow Model for a particular company or project.

(ii) The user will select 'Run' from the main menu of Genetik. The main results menu will appear on the screen. From the 'Loops' sub-menu the user will select the strategy for operating the first iteration of iterative design tasks. From the 'Mode' menu the user will select the simulation mode to be either deterministic or stochastic. From the 'Simulate' sub-menu the user will select "Initialise" to set the values of the different variables to their initial values. If the user requires to advance the simulation clock manually to monitor the change in state of the design tasks they will select 'Trace' from the 'Simulate' sub-menu. The user will then select 'Simulate' to run the simulation.

(iii) The default screen for running the simulation is the screen representing the design tasks in icon format showing the change in state of the different tasks as the simulation clock advances. If the user requires the results as a bar chart schedule for the design tasks, they will select 'Bar chart' from the 'File' sub-menu after running the simulation. If the user requires to monitor the change in state of different resources as the simulation runs they will select 'Resources/Toggle screen' (before running the simulation) to toggle the screen with the resources screen which displays the different types of resources and their state. If the user requires histograms of resource utilisation they will select 'Histograms' from the 'File' sub-menu. The user may also obtain the results in a tabular format if they select 'Processes' from the 'Report' sub-menu.

(iv) The user may alter the values of the global integer variables of the model such as the scale of the produced charts and the global cut off value of the quality attribute for the information exchanged. This is done by selecting 'INTG VAR' from the Genetik main menu and altering the required variables.

(v) If the user requires to investigate the effects of changing tasks durations and/or resources they will use the 'Edit' sub-menu and attempt different combinations. Investigations for scenarios of information related criteria such as assuming information, missing information, information quality, 'gate keeping' of information and phased release of information will be undertaken through changing the relevant attributes in the table which includes data about the information dependencies. Re-running the simulation under different scenarios will present to the user the effects of the introduced changes.

The next two chapters describe through a case study and practical examples the validation of the simulation model and the application of the developed tools to improve the management of the design process.

CHAPTER 7

EVALUATION OF THE DEVELOPED TOOLS

7.1 INTRODUCTION

In order to evaluate the benefits and the validity of the Data Flow Model and the Design Process Simulation Model, a case study was conducted on the New Engineering Complex Project at Loughborough University of Technology (LUT).

Although the models produced by the writer encompassed the Conceptual and Schematic design stages, the writer's involvement in this project commenced at the briefing stage in order to have a better understanding to the background of the project and because of the overlap between the Briefing/Feasibility and Conceptual/ Schematic stages.

The specific objectives of the study were:

(i) to validate the Generic Data flow Model for the Conceptual/Schematic design stages as the design process evolved

(ii) to collect data for the durations and resources required for every functional primitive task in the data flow model to be used in running the simulation

(iii) to assess the quality of information exchanged between the different participants in the design process

(iv) to assess the impact of the Client's involvement on the developed models

7.2 BACKGROUND TO THE PROJECT

Due to the gradual and continuous expansion in the engineering school at LUT, the existing planning of the campus does not accommodate all the engineering departments at one site. Two departments are located at the centre of the campus while the remaining five departments are located at the west side. A need was recognised to combine all the engineering departments at the west side of the campus with the objective of creating an integrated engineering school with shared facilities. This involved the requirement to relocate two departments and refurbish some areas in the remaining departments.

The client's representation in this project was in the form of three committees, each with different responsibilities: financial, operational and strategical. Each of these committees had direct contact with the design organisation AMEC Design and Management. Additionally, there was representation for the end users in the form of committees representing both the departments which would be relocated and those which would be refurbished. The estates department liaised with all the aforementioned parties and with AMEC, and appointed a project manager who was involved from the first project meeting. Therefore, presentations needed to be given to a total of six committees, each committee consisting of 3-6 individuals. This is represented diagrammatically in Figure 7.1.

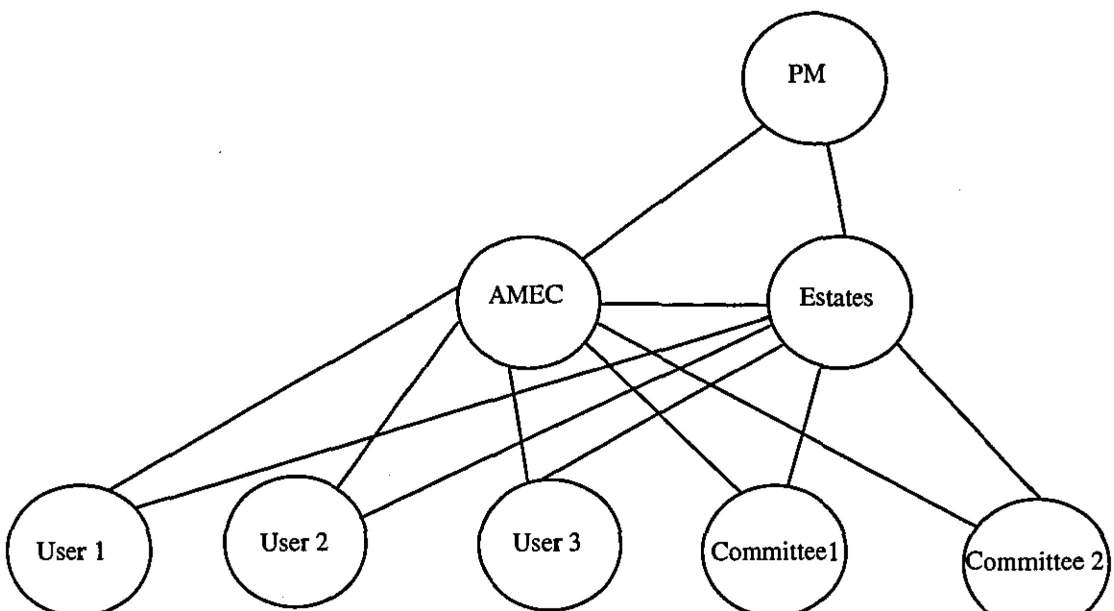


Figure 7.1 Relationships between the different parties involved

7.3 METHODOLOGY

The following methodology was proposed to meet the objectives of the case study:

(i) To validate the Generic DFD for the Conceptual/Schematic design stages it was decided, with the agreement of the project director and the client's representative, that the writer would 'shadow' the design meetings with the different Client's and end user's committees, and with the design team members. This would provide direct experience to observe/record all the discussions, issues raised and design information exchanged; categorise the information within different headings; and check that the information could be allocated to at least one of the information flows on the diagrams. A review of the DFD would be undertaken with the design leader after the Conceptual design. These data would be supplemented by interviews with the design staff involved and the Client's representative. It was agreed also that the project leader would provide the writer with the minutes of meetings and reports issued by AMEC as the design process evolved.

(ii) To collect data for the durations and resources to run the simulation model, it was decided to discuss with the design leader the durations and resources required at various points throughout the Conceptual and Schematic design stages.

(iii) To assess the quality of information exchanged, it was decided to observe the actions taken by the members of the design team on receipt of information. This would be achieved by issuing forms to the design team members indicating the details of the information received, the source of information, and the action taken upon receipt of information. This action would be an indication for the quality of information provided. The designer would be prompted to select one of the following actions:

(a) Information included directly into the design.
(b) No action due to irrelevant information received:
Information received but not required to proceed with this task
(c) No action due to inaccurate information received:
Example: sketches with missing dimensions
(d) No action due to receipt of invalid information:
Example: late information about loads received after a design modification has been agreed to include additional storeys which renders the issued loading information as invalid
(e) Request for clarification
(f) A redesign required
(g) Other actions

The issues of information relevance, accuracy and validity are the measures for information quality proposed by Marchand (1990) and Ronen and Spiegler (1991). Other proposed measures such as information aesthetics and perceived value were not included because they were not of primary interest to the research.

The choice of project presented both advantages and disadvantages.
The advantage of considering this project as a case study was that the writer, being a researcher at LUT, was familiar with the campus facilities and had already met some of the project participants. However, the writer was aware neither to let this advantage prejudice his work during this case study nor to include his own bias when drawing any conclusions.

7.4 PROJECT BRIEFING

The original briefing document issued by LUT included only a summary of the spaces requirements for the new areas to be built and the areas to be refurbished. An additional document was produced by one of the departments that was moving their location. This document included the department's prime objectives, population, specific spatial requirements and the perceptions for the new building. The main objectives and requirements of the Client and the end users were elicited by AMEC personnel through a series of value management sessions. A separate session was held with each of the two "moving" departments and with each of the Client's committees. The "non moving" departments were divided into two groups and a separate session was conducted with each group. This made a total of six value management sessions with each session attended by 5-7 participants. The value management sessions were conducted following the SMART (simple multi-attribute rating technique) described by Green (1992) in an occasional paper produced by the CIOB. Each session was managed by a specialist from AMEC (known as facilitator) and, according to the writer's observations and recorded notes, conducted in the following manner:

(i) The group had to identify the mission statement for the project from their own point of view. Each member of the group has identified their own perception of the project's mission statement and the facilitator then formulated these mission statements into one mission statement.

(ii) Members of each group identified the primary objectives and the enabling objectives of the project from their own perspective and a value tree was sketched. (The enabling objectives represent the break down of the primary objectives.)

(iii) Each member was asked to add a weighting allowance to every primary and enabling objective. This weighting allowance could be on any arbitrary scale chosen by each participant as the interest was only the relative importance of each objective.

(iv) An overall weighed score was calculated for each objective to identify the key objectives.
(An example of the weighting allowances and the overall weighed score for the prime objectives of one of the moving departments is shown in Appendix VII. An example of a weighed value tree is appended also in Appendix VII.)

(v) A brainstorming session was held at the end of each value management session to invite any ideas.

Each of the value management sessions was conducted to a similar agenda. An agenda for a typical value management session is included in Appendix VII.

By carrying out these sessions with the different users groups, AMEC personnel were able to identify the users' prioritised requirements/objectives and their perceptions for the new development. The sessions were held in a relaxed informal atmosphere conducive to the full participation of all the attendees and the generation of comments and ideas.

In addition to the value management sessions, AMEC has issued to each user group Room Data Sheets (RDS) for each room in order to elicit information about specific user requirements. The RDS included basic information completed by AMEC (such as room name, functional description) leaving the end user to complete any additional information and/or specific requirements (such as any critical dimensions). An example of RDS is included in Appendix VIII.

7.4.1 Data collected during the value management sessions

The data collected during the value management sessions were in the form of notes recording the writer's observations during these sessions. Since the aim of value management is to identify the user's objectives and perceptions to the project prior to the Conceptual design stage, the information elicited by AMEC from the user's groups has been considered as a part of the 'project brief' information flow in the Generic Data Flow Model. The following section describes the writer's observations

7.4.2 Observations and comments on information elicitation during Value Management sessions

(i) The process occasionally led the participants to become focused on certain areas, leaving some gaps in their analysis or forgetting some issues. A typical example was the focus on identifying objectives and requirements for the newly built areas without raising the issue of rationalising the existing facilities.

(ii) A member of one of the user groups commented that scores of importance should not be prejudiced by the ease or difficulty of achieving a certain objective. An example was the objective of 'maintaining departmental identity' which already existed and hence was not regarded as requiring a high score although it was of high importance.

(iii) Conducting separate value management sessions for parties with different interests led to conflicts in some objectives and consequently led to confusion especially in the requirements for the shared facilities.

(iv) During informal discussions held between the writer and members of the user groups who participated in the value management sessions, the writer noticed that the technique was regarded by the user groups as an efficient way to build team work and establish a working relationship between end users and the design team.

(v) The project manager appointed by LUT had no active role during the value management sessions. The sessions were conducted and managed by AMEC personnel.

7.5 THE DEVELOPMENT OF THE CONCEPTUAL DESIGN (OUTLINE PROPOSALS)

The series of value management sessions were held over a period of five weeks with a further two weeks to prepare a feasibility study report to proceed with the Conceptual Design. A presentation was held by AMEC to representatives of all the committees. At this presentation all the work that had been undertaken to date was presented. This included the value management trees, a site plan, sketches for different floors, models showing impressions of the project and a summary of areas to be built and areas to be refurbished.

The summary of areas presented showed an excess of more than 50% from the areas required by the brief. This was attributed to AMEC being focused on appending areas as required by different committees without enough consideration to rationalise the existing developments. Additionally, there was confusion as to whether the areas shown in the brief were net areas (room areas excluding circulations, stairs, etc.) or gross areas (room areas including circulations, stairs etc.). This required a design review for spaces rationalisation, especially for the shared facilities. This was achieved through a series of meetings held over a period of four weeks between AMEC's architect and the Client's representative who liaised with all departments. This resulted in the production of three alternatives for the conceptual design, each alternative representing a solution to the obstruction of existing infrastructure to the new developments.

Although there was a general approval by the representatives of all committees on the concept during the early presentation held by AMEC, a sudden complete disapproval of the concept by one of the departments arose at this stage. This resulted in a fundamental re-design to satisfy the requirements of this department and consequently a delay of three weeks for AMEC to prepare an alternative site plan. A sketch of the new site plan was issued to the Client's representative who liaised with the different departments to seek their approval. All the end user groups were in favour of the new concept and AMEC was instructed to proceed with the Conceptual design based on the new alternative. This required further meetings to be held between AMEC and each individual department to develop the concept and discuss spatial and functional requirements on the produced plans. These meetings took place over a seven week period after which the Conceptual design report was produced and presented to the Client together with a cost estimate for the project. The project was then halted by the Client who stated the need to revise their internal budget for the work.

The analysis of the data collected and conclusions drawn are explained in the following sections.

7.6 DATA COLLECTED AND DIFFICULTIES ENCOUNTERED DURING DATA COLLECTION

The data collected up to this stage included notes and observations recorded during 'shadowing' the design meetings held between AMEC and the different Client committees and user groups in addition to minutes of meetings. All the participants in the meetings welcomed the writer's presence when they knew the purpose of his attendance. This attitude reflected the participant's recognition of the need for improved design management. However, the writer was asked by AMEC not to attend meetings with the Client which included financial details. This was understandable. Since the design resources involved up to this stage were mainly the project leader (a senior architect) and another architect, there have been no internal design meetings held within AMEC.

A summary of the meetings shadowed by the writer and the information exchanged during these meetings is provided in Table 7.1.

These data were also analysed in the form of a bar chart showing dates of different meetings together with the key issues and problems raised. This is shown in Figure 7.2.

Table 7.1. Summary of information exchanged during the Conceptual design stage

Date	Participants	Issues raised	Summary of information exchanged through verbal discussions (formally minuted by AMEC)	Summary of information exchanged in paper format
17-5-95	AMEC, estates, PM (3 participants)	- Rationalise areas to meet brief req. - Options for relocating/ combining some areas	- Estates to AMEC Possibility of relocating/ combining some areas e.g. labs, workshops, lecture theatres (SF) - AMEC required from estates to: – Analyse occupancy and utilisation of existing lecture theatres (SF) – Seek confirmation from other departments for the possibility of relocating/ combining some areas (SF)	Global space analysis for moving departments presented by AMEC (SF)
25-5-95	AMEC, estates, capital steering committee, PM, LUT landscape specialist (4 participants)	- Space allocations - Revised spaces - Rationalisation of spaces - Obstruction of existing infrastructure to new developments	- Client to AMEC – Confirmation for areas required for labs and outlining areas which could be relocated/ shared (SF) – Analysis for utilisation of existing lecture theatres (SF) – University wild life strategy (T) – Areas the Client requires to conserve e.g. wild life, brook (T) - AMEC to Client – Impact of the new developments on existing infrastructure (O) - Estates required from AMEC: To study options for relocating and incorporating the existing boiler house (O) - AMEC required from Client: University space standards for office space and researchers	- Presented by AMEC to Client: Detailed space analysis supported by preliminary sketches (SF) - Presented by Client to AMEC: Analysis for utilisation of existing lecture theatres (SF)

12-6-95	AMEC, Estates, PM (3 participants)	- Rationalising existing infrastructure	- Estates to AMEC – Loads on existing infrastructure (O) – Area required for each researcher (SF) - AMEC to Estates – Three options for incorporating/ relocating boiler house (O) - Estates required from AMEC: – A design alternative incorporating a central flexible research area (SF)	
13-6-95	AMEC, moving dept.1 (4 participants)	Concept design does not meet the objectives of the dept.	- Moving dept.1 to AMEC – The concept design does not reflect the requirements of the dept. especially in being 'not visible' and no accommodation for future expansion.(T,F) – Walking through the department to show AMEC existing facilities (T,S,F)	
13-6-95	AMEC, estates, (2 participants)	Requirements of moving dept.1	- Estates to AMEC Postpone developing conceptual design (T) - Estates required from AMEC: Produce a sketch showing a new site plan incorporating the requirements of moving dept.1	
27-6-95				AMEC sent new alternative to Client (S,F,T,O)
4-7-95				Client sent instructions to AMEC to develop new alternative

10-7-95	AMEC, estates, PM (3 participants)	Discussing new design alternative	- Estates to AMEC – Approval in principle for the new alternative – Space requirements for some labs (S,F,O) - AMEC to ESTATES – Revised dates – Abortive work to be claimed by AMEC - Estates required from AMEC: – Develop the new concept and focus on shared facilities to maximise integration (T) – Arrange meetings with moving departments - AMEC required from estates: Information about existing services (O)	

21-7-95	AMEC, estates, moving dept.2, PM, (5 participants)	Discuss new concept	- Moving dept.2 to AMEC – General approval in principle – Walking through the dept. to show AMEC existing facilities (S,F,O) – Comments on general layouts from the dept. (S,F,O,A) – Refusal to relocate a lab. in another dept. which shares the same lab due to technical reasons (O) – Requirements for parking facilities (S,F) - AMEC to estates Suggestion to incorporate the above mentioned lab. in the design through negotiations to occupy an area in one of the existing depts. (F,O) - AMEC required from estates: Arrange for meeting with staff (end users) of dept.2	- Presented by AMEC: - Preliminary general layouts for each floor - Site plan

25-7-95	AMEC, estates, moving dept.1, PM, planning advisor (LUT), (5 participants)	Discuss new concept	- Moving dept.1 to AMEC – General approval in principle – Comments on general layouts from the dept. (S,F,O,A) – Areas allocated for researchers (instructed by estates) are inadequate and conflicts with the same areas required by the brief (S,F) - Estates required from AMEC: – Cost estimate (C) – Study re-use of existing space (T) – More focus on shared facilities to maximise integration (T) AMEC required from Estates: Arrange for meeting with staff (end users) of moving dept. 1	- Presented by AMEC: - Preliminary general layouts for each floor - Site plan

1-8-95	AMEC, estates, project steering committee, PM, planning advisor (7 participants)	Discuss new concept with project steering committee	- Client to AMEC Comments on general layouts (S,F,O,A) - AMEC to CLIENT Outcome of meeting with planning officer (A) - AMEC requested from Client: – Confirmation for areas to be allocated for researchers in each dept. (S,F) – Confirmation for occupying an area from an existing dept. to be used as a lab in the current concept (AMEC will assume this area available) (S,F,O) – Clarification for some conflicting area requirements (AMEC will proceed based on assumptions and suggest contingency alternatives (S,F) – Engineering workshops strategy - Client requested from AMEC Outline cost estimate	AMEC presented to Client: - General layouts - Revised detailed space analysis - Revised programme

16-8-95	AMEC, estates (3 participants)	Current design proposals, project cost, outstanding brief items.	- Client to AMEC – Lab could not be housed in existing dept. (S,F,O) – Engineering workshop strategy (S,F,T,O) – Confirmation for research areas (S,F) – Requirements for car park and cycle park spaces (S,F) - AMEC to Client – Envelope materials (A) – Quality plan - Client requested from AMEC: – Items included in and excluded from the cost estimate (C) – Revised cost estimate based on new information received (C)	AMEC presented to Client outline cost estimate

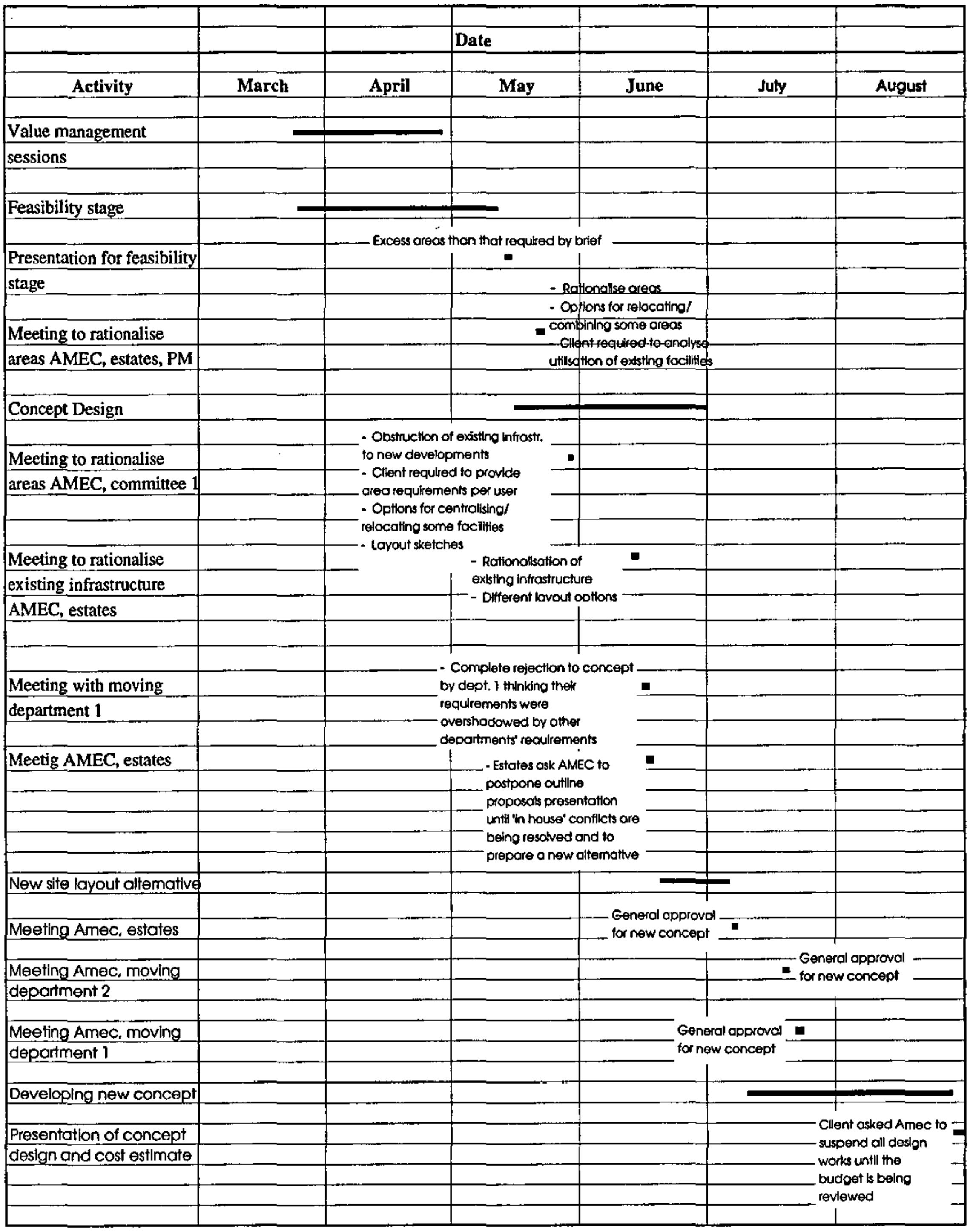


Figure 7.2 Analysis of the data collected in the form of a bar chart

The information exchanged during the design meetings at the Conceptual design stage fell into five categories:

(i) Information related to spatial and functional requirements (SF).
(ii) Information related to strategical requirements (T).
(iii) Information related to operational requirements (O).
(iv) Information related to the project aesthetics (A)
(v) Cost information (C)

The category of information exchanged is bracketed in Table 7.1.

These categories were chosen by the writer based on the observation of the nature of information exchanged during the design meetings held with the Client and the requests for information by AMEC to the Client and vice versa. They represent the issues which are usually under development during the Conceptual design stage. Moreover, classifying the information into these categories facilitated allocating the observed information to the relevant information flows on the DFDs, and hence led to a more structured way for the validation process of the Generic DFD. This is explained in more detail in sections 7.7 and 7.8.

The main difficulty in the data collection during the Conceptual design stage was to collect data about durations of the functional primitive tasks of this stage which are necessary to run the simulation model. The design leader was unable to allocate durations for discrete tasks due to the following reasons:

(i) At the Conceptual design stage, architects usually 'think' at more than one task simultaneously. Examples are footprint locations, footprint shape and building layouts.

(ii) At the Conceptual design stage, a substantial amount of time is consumed during 'thinking'. This may be either inside or outside the design office and the thinking time may vary according to the designer's experience and background and according to the project's particular circumstances.

(iii) The time taken to complete a task is usually discontinuous. There is a substantial amount of 'waiting' time for the Client to make decisions or to collect certain information. This may vary depending on the type of Client and/or the type of project.

(iv) During the Conceptual design stage, a considerable amount of time is spent in meetings with the Client and the user groups and cannot be directly attributed to any one task.

In order to overcome this problem, the writer requested permission to have access to the time sheets of the designers who participated in the design at this stage. Due to the previously mentioned circumstances which led to AMEC carrying out a substantial re-design (as a result of the 'sudden change of mind' of one of the user groups), the writer has decided to consider only the period during which the new concept has been developed. By cross referencing the time sheets with the project leader's diary, design hours have been apportioned among different tasks reflected in the diary and rationalised among the functional primitive tasks of the data flow model. Knowing the number of designers involved, the duration for each task has been estimated.

Data about resources have been collected from the project leader who identified the resources at this stage as follows:

(i) The project leader (senior architect).
(ii) Another architect
(iii) An architect involved occasionally
(iv) Occasional involvement of a mechanical engineer, an electrical engineer and a structural engineer to decide on the relevant strategies and provide feedback to the architect.

The data about the estimated durations and the resources have been used as the data input to run the simulation model. This is explained in section 7.10. Examples of simulating typical events that occur during the design process using this data are included in chapter 8.

7.7 OBSERVATIONS RECORDED DURING THE CLIENT'S MEETINGS

One of the objectives of this case study was to assess the impact of the Client's involvement on the developed models, the writer has recorded the following observations during the design meetings held with the Client:

(i) There was no impact of the Client's involvement in this case study on the developed generic data flow model as the Client's role was merely a 'source' of information as represented in the model. There were no actual tasks undertaken by the Client apart from 'seeking information'. The information provided by the Client fell into two categories:

(a) Spatial, functional, strategical and operational requirements which should have been part of the original brief and/or showed conflicts between the brief and the specific requirements of each user group. This information is grouped in the Generic DFD as 'specific requirements' which is part of the project brief provided by the source 'Client'

(b) Approvals and comments.
This information appeared in the Generic DFD as 'approvals and comments'.

(ii) It is important to resolve areas of conflicts in interest between different parties of the end user 'in house' before the design organisation start the process of brief elicitation. This would prevent confusion of the designers and the time wasted when they try to achieve conflicting objectives. It might avoid claims for abortive work and time consumed by the design organisation in trying to resolve these conflicts.

(iii) The Client and/or end users should be represented by one committee representing different interests. This committee should be responsible for rationalising objectives of different parties and liaise with the design organisation in this respect. An alternative model to Figure 7.1 showing the links between different parties participating in the project is suggested in Figure 7.3.

(iv) For Clients of large projects, it is important to produce a 'design guide lines' handbook for new projects and/or any further developments to existing projects. This would represent a reference for the designers during making design decisions. A typical example in this case study was to decide on working areas required by each researcher.

(v) A study should be undertaken by the Client for the occupancy and utilisation of existing facilities of a premises before further developments is being undertaken. This would result in time savings especially in the early stages of design.

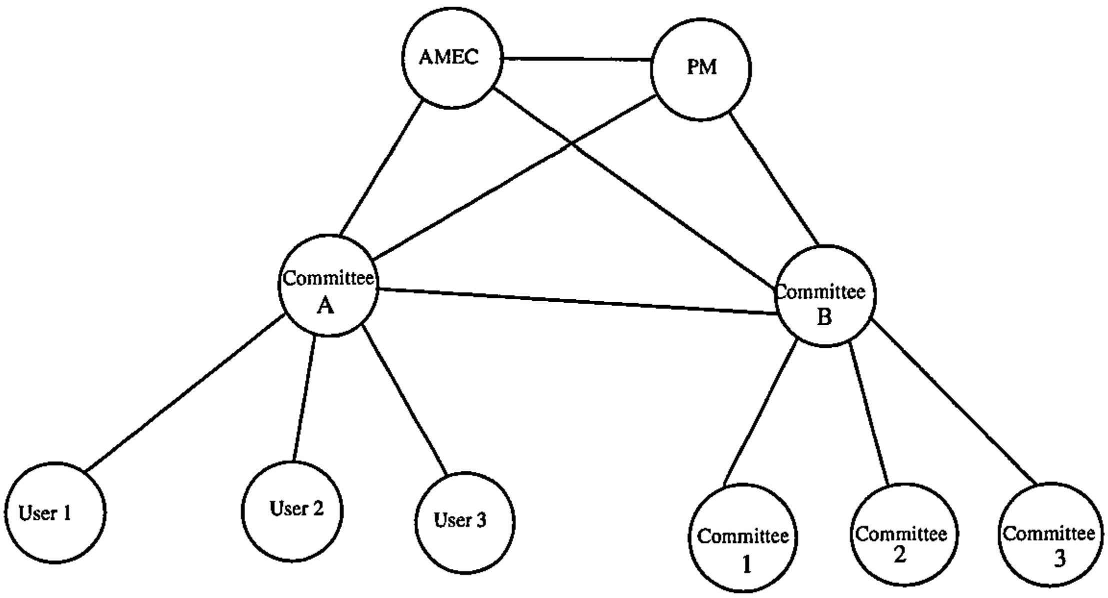


Figure 7.3 A proposed model for the links between different parties participating in the project

(vi) A similar study should be undertaken by the Client to analyse loads on existing infrastructure. In addition to time savings, this would result in massive savings in costs which might be incurred in overdesign.

(vii) It is important to seek formal approval to the design concept before proceeding to next stage(s). Implied approvals are not enough and are misleading for the designers. This is because the designers would be proceeding with the design on the basis that the concept had been approved while the Client/end users might be still in the process of reviewing it.

(viii) 'In house' politics among different parties of large clients' organisations have a big impact on the design progress and the key decisions taken during the design process.

(ix) There was neither contribution nor influence from the project manager on any of the information exchanged.

7.8 APPLICATION OF THE GENERIC DATA FLOW MODEL TO THE PROJECT

The Generic Data Flow Model decomposed to the functional primitive tasks for the Conceptual design stage was validated against the actual process through comparing the data collected by the writer with the produced model in addition to interviewing the project leader. The information provided from the Client to AMEC has been analysed and explained in section 7.7 (i). The information provided by AMEC to the Client has been allocated to at least one of the information flows on the diagram. These information fell into two categories:

(i) The conversion of the Client's spatial, functional, strategical and operational requirements into space analysis and general layouts.

(ii) Cost information

The information exchanged within the design office has been validated through interviews with the design leader. These interviews showed that the model represents the actual process with minor adjustments due to the special nature of the project being an extension to existing developments. The project leader pointed out the following:

(i) The Generic Model was a clear representation for the different tasks and information flows that occur during the Conceptual design stage and could be used as a 'check list' for information requirements for different design tasks.

(ii) Although the model represents the actual process, some of the tasks are carried out rapidly without clear realisation of these individual tasks, which emphasises the value of the model as a management tool. Examples of these tasks are in diagram 1.1.1.1.1 Arrangement of spaces.

(iii) The following elements of the model are not applicable at the Conceptual Design Stage in this particular project. This is because this project is of specific nature being an extension to existing developments and being a non industrial project.

(a) The process 'Preliminary site investigation':

There is no need to conduct a preliminary site investigation at this stage as the site is already known and other developments already exist on the site, hence the site information is already available.

(b) The information flow 'Typical penetrations':
Information about typical penetrations required by the structural engineer from the services engineer is not required at this stage. This information would be required in case of industrial projects.

(iv) The writer has observed during 'shadowing' the meetings that there has been no 'cost feedback' from the process 'estimating costs' to any of the design disciplines at this stage. Although the design team was aware of the Client's budget, the design has proceeded to fulfil the different committees requirements and the cost has been estimated based on unit cost/m^2. A more accurate cost estimate had been made at the end of the Conceptual design.

(v) The project leader pointed out that although it would be beneficial to carry out 'drainage concept design' at the Conceptual design stage, the practice in AMEC is not to start the drainage design before the Scheme design commences.

(vi) There has been a particular emphasis in this project to adopt an environmental strategy which aims to maximise environmental conservation. This has resulted in adding the process 'establish environmental strategy'.

(vii) Although the processes related with 'fire fighting' appeared in the Generic Model at the Scheme design stage, it was necessary in this project to 'establish the fire strategy' at the Conceptual stage in order to ensure that the anticipated extensions to the existing buildings would not undermine the fire strategy for those buildings (e.g. access, egress, routes for the fire brigade, etc.)

(viii) The design discipline which initiates the decision for the 'skeleton material' differs according to the type of project. For industrial projects such as factories this decision is usually initiated by the structural discipline. For projects which are driven by architects such as the LUT Engineering Complex this decision is usually initiated by the architect.

The Data Flow Model and a sample report from the data dictionary after incorporating AMEC adjustments are included in Appendix IX.

7.9 ASSESSMENT OF THE QUALITY OF INFORMATION DURING THE CONCEPTUAL DESIGN STAGE

Before starting this case study, it had been decided to assess the quality of information exchanged through the actions taken by the designers on receipt of information. This would be achieved through issuing forms to be filled by the designers as explained in section 7.3. These forms had been given to the design leader in order to be issued to different designers. However, by the end of this stage, the design leader had still not issued these forms. The reason was that the main designers involved at this stage had been the design leader himself and another architect, and the main source of information was the Client. Most of the information has been transferred through the design meetings held between the Client and the design leader and not in a tangible way (i.e. not in the form of drawings, memos, etc.) which rendered the forms difficult to complete.

Discussions with the design leader showed that any information issued by the Client at this stage would be useful for the architect, as the main concern for the architect (at this stage) is to complete the missing information in the Client's brief in order to enable him to produce the general layouts and floor area distribution. Therefore, all information issued from the Client to the project leader would 'fit the purpose' and be incorporated into the design and hence considered to be of satisfactory quality at this point in time even though this information may be incorrect or may be changed by the Client later.

The same argument is applicable for the quality of information transferred within and among other design disciplines as it is most important at this stage to have information to start the design with. Since this information would be incorporated into the Conceptual design, it would 'fit the purpose' and hence considered of satisfactory quality.

Therefore, it has been decided to study the variation in design information quality at later design stages where there is more concern about the details and specifics of the information.

7.10 APPLICATION OF THE DESIGN PROCESS SIMULATION MODEL TO THE CONCEPTUAL DESIGN STAGE

Data from the Data Flow Model were input in the Design Structure Matrix (DSM) to identify the iterative design loops. Figure 7.4 shows the identified iterative loop.

Matrix View

PSM Version b.30	9 :	1 :	2 :	3 :	6 :	10	11	12	13	14	15	16	4 :	5 :	7 :	8 :
9 : 1.1.3 Site planning	9 :															
1 : 1.1.1.1.3 Envelope concept design		1 :				X			X		X					
2 : 1.1.1.1.4 Roof arch concept design		X	2 :													
3 : 1.1.1.2.3 Superstr. concept design		X	X	3 :	X				X			X				
6 : 1.1.1.3 Services concept design					6 :				X			X				
10 : 1.1.1.1.1.4 Consd ftprnt shape opt	X					10	X		X	X	X					
11 : 1.1.1.1.1.3 Consider no. of storeys		X		X		X	11	X								
12 : 1.1.1.1.1.2 Consdr ftprnt locations	X						X	12	X		X					
13 : 1.1.1.1.1.5 Cosdr bldg. layout opt		X		X	X	X	X	X	13	X	X	X				
14 : 1.1.1.1.1.1 Floor areas distribution					X	X	X			14	X					
15 : 1.1.1.1.5 Establish fire strategy	X								X		15					
16 : 1.1.7 Establish env. strategy									X			16				
4 : 1.1.1.2.2 Floor slab concept design					X				X				4 :			
5 : 1.1.1.2.1 Foundation concept des.							X							5 :		
7 : 1.1.5 Estimating costs		X	X	X	X		X			X			X	X	7 :	
8 : 1.1.4 Concept design report prod.	X	X	X	X	X	X	X		X			X	X	X	X	8 :

Figure 7.4 Iterative design loop identified by DSM

The functional primitive tasks and their links, data about iterative design loops identified from the DSM, and data about resources and durations (which have been already estimated from the time sheets as previously explained in section 7.6) were input in the simulation model. The following assumptions have been made:

(i) The time sheets have revealed that the duration of each task was discontinuous due to 'waiting time' and 'thinking time' at the Conceptual design stage as previously explained. Since these aspects could not be practically simulated, an assumption has been made that the duration is continuous.

(ii) For the same reason, resources allocation among different tasks had to be rationalised especially for iterative design tasks.

(iii) The time spent in carrying out administrative tasks such as 'writing minutes of meeting' has been ignored.

The programme produced by the simulation model showed a total duration of five weeks for the Conceptual design as opposed to an actual duration of eight weeks. This discrepancy is attributed to the above mentioned assumptions in addition to the assumptions made in estimating the tasks durations as explained in section 7.6.

Since more than one design task is considered by designers simultaneously at the Conceptual design stage, it would be unrealistic to simulate these tasks in a discrete manner. Moreover, design resources involved are minimal and there is no need to produce resource utilisation histograms at this stage. Additionally, the available information at this stage would be incorporated in the design to start with and hence there is no need to simulate information quality.

Therefore, it was concluded that there will be no substantial benefits from running the simulation model to produce design schedules only for the Conceptual design stage and a decision has been taken to focus attention on the simulation of the later design stages.

7.11 SOURCES OF INFORMATION FOR THE SCHEMATIC DESIGN STAGE

It was decided that the writer would continue 'shadowing' the design meetings held between AMEC staff and the Client's representatives in addition to the internal design meetings within AMEC organisation during the Scheme Design Stage when more designers from other disciplines are involved. Additionally, the writer has prepared tabular forms to be completed by all participants in the design on weekly basis. The objective of these forms was to elicit data about information received and information provided by each design task represented by a list of functional primitive tasks of the Data Flow Model at the schematic design stage. The quality of information will be decided based on actions taken by the designers on receipt of information as previously explained in section 7.3.

A model of the forms was sent to a representative from AMEC to test their suitability for data collection. Minor modifications to the tables layout were suggested. Additionally it was suggested that it would be appropriate to leave the designers to list the tasks they undertake each week and rationalise these with the functional primitive tasks of the Data Flow Model in lieu of being prompted to complete information about already listed design tasks.

However, it was decided to discard this comment and to prompt the designers to fill the forms with the relevant data for each listed design task as there was the possibility of having design tasks listed by designers in a very broad and general manner.

A meeting has been conducted between the writer and the project leader to validate the Generic Data Flow Model at the schematic stage against the AMEC practice. The model was found to represent the actual process usually undertaken by AMEC with minor adjustments. These adjustments were mainly related to information flows which would usually be exchanged at this stage in an industrial project but not in a project like the LUT Engineering Complex. Examples of these information flows are 'Louvers locations' and 'Air handling units locations'. Within AMEC practice, a process of 'producing definition brief' is carried out after finalising the concept design. The definition brief document is issued to the designers at the commencement of the scheme design stage. This document represents the developed design brief and includes the brief information elicited during the concept design. This process has been added to the model. However, the project leader could not comment on details of structural design tasks. This is partly because he is an architect and not a structural engineer, and partly because he was interested only in the interface of the structural design tasks with the architectural design which drives this project. He was not interested in details of how the foundations are being designed but he was interested in the output of some structural tasks like 'Produce frame typical sections' and 'produce skeleton layout'. The reason of his interest was that these tasks represent important decisions for the aesthetics of the building .The data flow model after incorporating AMEC adjustments is included in Appendix IX.

Additionally, the writer has asked the project leader to give estimation for the durations of the FPTs for the Schematic design stage in order to run the simulation model before commencing this stage and use the output as a basis to manage the scheme design. The schedule produced by the simulation model for the conceptual and schematic stages of design is shown in Figure 7.5.

After reviewing the budget, the Client decided to suspend the following design stages until further notice. However, the writer has used the data collected during the Conceptual design stage and the data estimated by the design manager for the Schematic design stage to develop examples of using the developed tools to improve the management of the design process. These examples represent typical events that occur during the Conceptual and Schematic stages of design. This is explained in more detail in chapter 8.

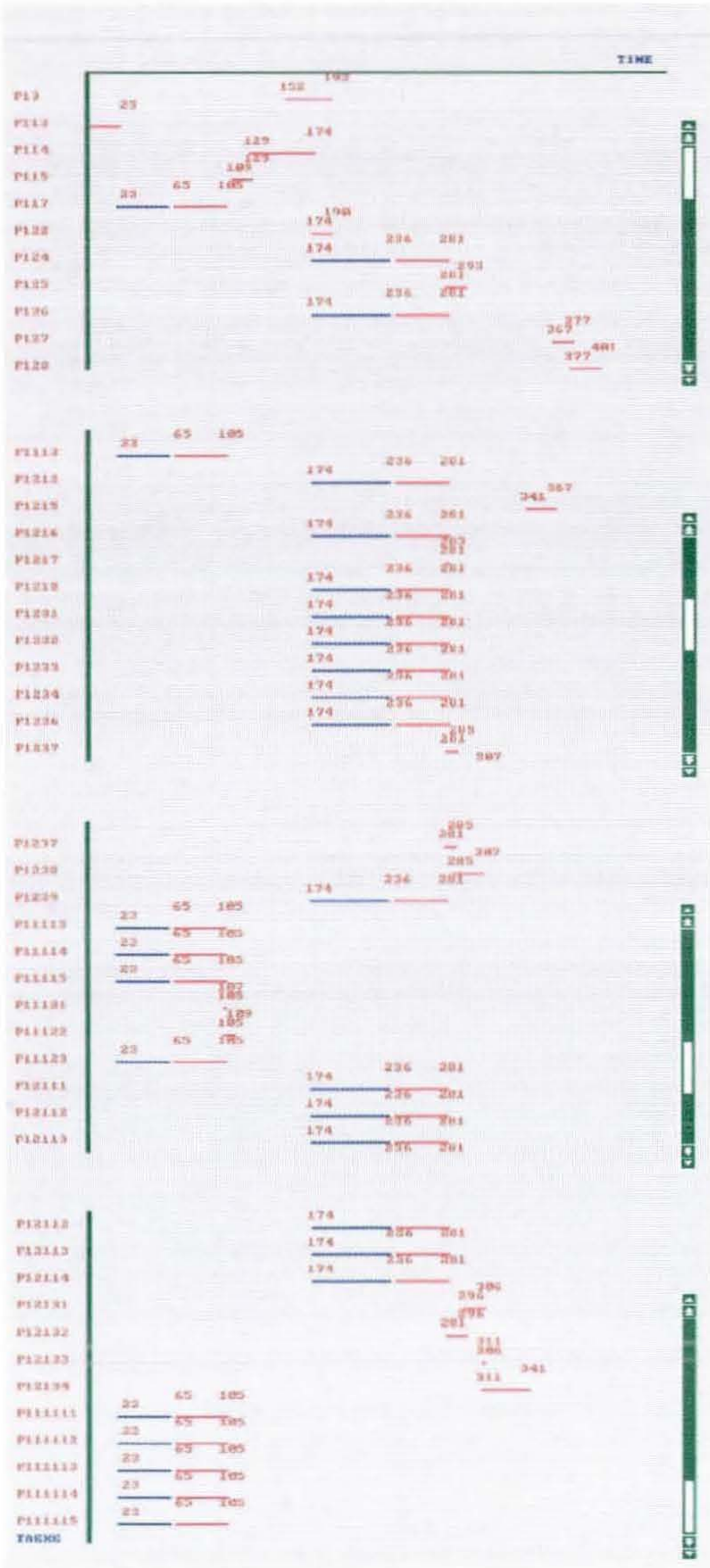


Figure 7.5 Output from the Simulation Model showing the Conceptual and Schematic design schedule

7.12 CONCLUSIONS DRAWN FROM THE LUT ENGINEERING COMPLEX CASE STUDY

The following conclusions are drawn from the LUT Engineering Complex case study:

(i) The Generic Data Flow Model for the Conceptual and Schematic design stages may be applied to any project with minor adjustments to fit the specific nature of each project. The value of the model is primarily that of a checklist to aid design management in identifying design tasks and their relevant information requirements.

(ii) The information exchanged during the design meetings at the Conceptual design stage falls into five categories:

(a) Information related to spatial and functional requirements.

(b) Information related to strategical requirements.

(c) Information related to operational requirements.

(d) Information related to the project aesthetics

(e) Cost information

This information was allocated to at least one of the information flows in the Generic Data Flow Model.

(iii) There was no impact of the Client's involvement in this case study on the developed generic data flow model as the Client's role was merely a 'source' of information as represented in the model. There were no actual tasks undertaken by the Client apart from 'seeking information'.

(iv) There are no substantial benefits from running the simulation model to produce design schedules only for the Conceptual design stage. This is due to the following reasons:

(a) At the Conceptual design stage, architects usually 'think' of more than one task simultaneously. This aspect is practically impossible to simulate and allocating durations to these tasks would be unrealistic.

(b) At the Conceptual design stage, a substantial amount of time is spent 'thinking' around the design problem. This may vary according to the designer's experience and background and according to the project's particular circumstances.

(c) There is a substantial amount of 'waiting' time from the Client to make decisions or collect certain information. This may vary depending on the type of Client an/or type of project and hence, simulation would give unrealistic results.

(d) The resources involved at this stage are minimal and hence there are no substantial benefits from simulating resource engagement or utilisation.

(v) Since the information exchanged at the Conceptual design stage is used to start the design, there is no need to assess the information quality. This is because the main concern for the designers is primarily to acquire the design information. It is at the later stages when refining the design that benefits are achieved through assessing design information quality.

(vi) Value management is a quick and efficient way to elicit users' requirements and objectives in a prioritised structured manner. The technique also builds team work between end users and the design team. The identified objectives may then be used as a 'check list' by the designers at each design stage. However, this process may occasionally lead the participants to become focused on certain areas and consequently miss more fundamental issues.

(vii) The Client and/or end users should be represented by one committee representing different interests. This committee should be responsible for rationalising objectives of different parties and liaise with the design organisation in this respect.

(viii) 'In house' politics among different parties of large clients' organisations have a big impact on the design progress and the key decisions taken during the design process.

(ix) Studies should be undertaken by the Client to establish the occupancy and utilisation of existing facilities and to analyse loads on existing infrastructure before further developments are considered.

(x) The case study confirmed that the methodology for data collection was appropriate for the Conceptual design stage. The study confirmed the need to categorise the information under different headings to facilitate the validation process for the DFDs. Therefore it was concluded that data collection during the Conceptual Design Stage is difficult to approach in a rigid way. Should a similar case study be conducted at the Schematic design stage, the same methodology would be adopted with regards to validating the DFDs but would differ with regards to running the simulation model. Prior to starting the scheme design, the design manager would be asked to comment on the Generic data flow model to carry out necessary adjustments for the project under consideration. Data concerning the durations and resources would also be estimated by the design manager before starting the process in lieu of collecting historical data. The simulation model would run based on the anticipated durations and resources. The results from running the simulation would be compared against the actual process as it progresses and, if possible, provide a contribution to the management of the process.

Due to the LUT project circumstances and the time scales for the research programme, the writer was unable to use the simulation model throughout the rest of the design period of the project. Therefore validation of the simulation model continued by industry feedback through demonstrations and discussions held by the writer followed by completing a 'feedback document' by each industry representative. Additionally, the data collected within this case study were used to produce practical examples to demonstrate the benefits that the developed tools offer to improve the management of the design process. This is described in detail in the next chapter.

CHAPTER 8

IMPROVING THE MANAGEMENT OF THE DESIGN PROCESS

8.1 INTRODUCTION

This chapter describes the application of the developed tools to improve the management of the design process. Practical examples are presented using data from the LUT case study (described in chapter 7) in addition to other data. The simulation model is used to investigate different scenarios of typical events that occur during the design process and the impact of changes on other design activities and on the project duration and resources. The chapter also describes the validation of the simulation model by industry feedback through demonstrations and discussions held by the writer followed by completing a 'feedback document' by each industry representative. Extending the developed tools for application at the Detailed design stage is also investigated.

8.2 OVERVIEW OF THE APPLICATION OF THE MODELS PRODUCED

When using the models the design manager will review the generic data flow model and identify the necessary adjustments to reflect the particular project under consideration. The Data Flow Model will be used by the design manager throughout the different design stages as a monitoring tool to ensure the completeness of the information requirements for the different design tasks. The DFDs will also assist the design manager to plan for the production of information and control design by ensuring that the required design information is included. The graphical representation of the DFDs will be used as a check list for the design tasks and their different information dependencies or interdependencies. Reports produced from the data dictionary will be used in several ways. Reports for processes inputs/outputs will be used as check lists for the information inputs necessary for each design task to proceed and for the information outputs that each design task should produce. Reports for data definitions will assist in identifying the different data elements that constitute the different information flows. These reports will be circulated among different members of the design team to achieve a mutual realisation and appreciation for other team members' information requirements. It is important that the design manager co-ordinate information with the design planner (if this is a different person) to break down the design tasks into tasks corresponding to those identified by the DFDs. This will allow the design manager to apply the developed tools easily and successfully.

The matrix modelling for the FPTs of the DFDs will assist the design manager to identify loops of iterative design tasks. These tasks will normally be multi-disciplinary and the design manager will be aware of these tasks that would be undertaken simultaneously in an iterative fashion and will require careful co-ordination. Knowing these tasks will assist the design manager in the interface management of different design disciplines. It will also assist the design manager in the selection of resources as designers performing tasks which fall within the same iterative loop should work in proximity to facilitate the communication process and increase the design efficiency. For example, if an architectural design task is carried out iteratively with a mechanical design task, then it would clearly be advantageous if the designers of these tasks are able to work in the same location. If this is not possible then closer co-operation through new technologies should be encouraged. Resources for these tasks also should be rationalised by the design manager knowing that they would be undertaken not only simultaneously but also iteratively so that tasks of the same discipline would be performed, where possible, by the same designer.

The design manager will be required to allocate to each functional primitive task the duration and the resources required to perform these tasks. The simulation model would then be run. If the project completion time forecast from the simulation does not fit the completion dead line, the design manager would attempt different scenarios on 'what if' basis for the durations and resources until the target completion time is achieved. The simulation model will produce a programme for the design process on this basis in addition to histograms representing different resource utilisation. The design manager will also be able to assess the impact of late changes introduced to the design information, especially from the Client's side, which was one of the problems identified from the survey and interviews. This will be undertaken through different scenarios for the attributes of the information links in the simulation model and running the model to identify the design tasks that would be affected. The design manager may then demonstrate to the Client the impact of the introduced changes on the whole process and that an increase in the design duration and/or cost is possible.

8.3 THE PROBLEM OF MISSING INFORMATION

The design manager will monitor the information requirements for each design task as the design process evolves using the DFDs. If at a certain point in time he/she finds out that information required to proceed with a certain task is missing, the design manager can use the simulation model to investigate how the lack of information impacts on other design activities. This will be done in the following manner:

- The missing information is equivalent to an information link of zero quality. The design manager will run the simulation model while setting the quality attribute on the information link representing the dependency of this task on the task or source that should provide that information to zero.

- Assess the state of each task at the end of the simulation run i.e. which tasks are completed, which did not start, which have completed the first iteration etc. and hence assess the implications of this missing information.

- Show the provider of information the implications of the missing information to gain an awareness for the problem

The design manager will also assess the impact of working when information is unavailable under two alternatives:

- Waiting for the information to arrive
- Making assumptions

If the design manager decides to wait for the information to arrive, he/she will assign a closed gate on this information link and will allocate a duration representing the delay for this information to arrive. He/she will run the simulation model and assess the impact of this delay.

Alternatively, the design manager would make assumptions and proceed with the design. (Assessing the impact of assumed information is described in the next section).

This feature of handling the problem of missing information is useful in all the design stages. The following example is taken from one of the case studies.

<u>Example</u>

This example is based on actual data collected during the LUT Engineering Complex case study at the Conceptual/Schematic design stages and described in Chapter 7. The following assumptions were made in this and following examples:

- There is more design input for the tasks related with the foundations design and hence longer durations. This represents a typical situation for an unknown site.

- Resources are available at all times. This assumption will maintain consistent comparisons between different scenarios.

Assuming that the site investigation process is delayed or the Client is negotiating different site options and hence precise information about the site and soil conditions is not available.

From the data flow model and the design structure matrix the design manager will be able to identify the information flows produced by the task 'site investigation' and the design tasks which are dependent on these information flows. These tasks are:

- Consider foundations options
- External works scheme design
- Landscaping scheme design
- Scheme drainage design

The design manager will initially run the simulation model without any constraints to obtain a scenario of the design project against which other scenarios would be compared. The total design time for the Conceptual/Schematic design stage is predicted to be 401 design hours (50 working days) as shown in Figure 8.1.

To assess the impact of missing information, the design manager will set the quality attribute on the information link between 'site investigation' and each of the dependent four tasks to zero. Running the simulation will present for the design manager the impact of the missing information on the remaining design tasks. This is illustrated in Figure 8.2. One iteration only for the iterative loops of design tasks at the Schematic design stage would be undertaken because two tasks within the loop are dependent on the task 'site investigation' and hence the requisite information for these tasks to proceed is incomplete. Therefore, the state of the tasks within this loop is "4" which represents completing the first iteration. Additionally, nine design tasks could not commence as a result of the missing information (in addition to the directly dependent design tasks). These tasks are:

- Approximate foundations loads calculations
- Decide on foundations type
- Produce foundations scheme design documents
- Revise cost estimate
- Structural design review
- Ancillary buildings scheme design
- Outline architectural specs. production
- Architectural design review

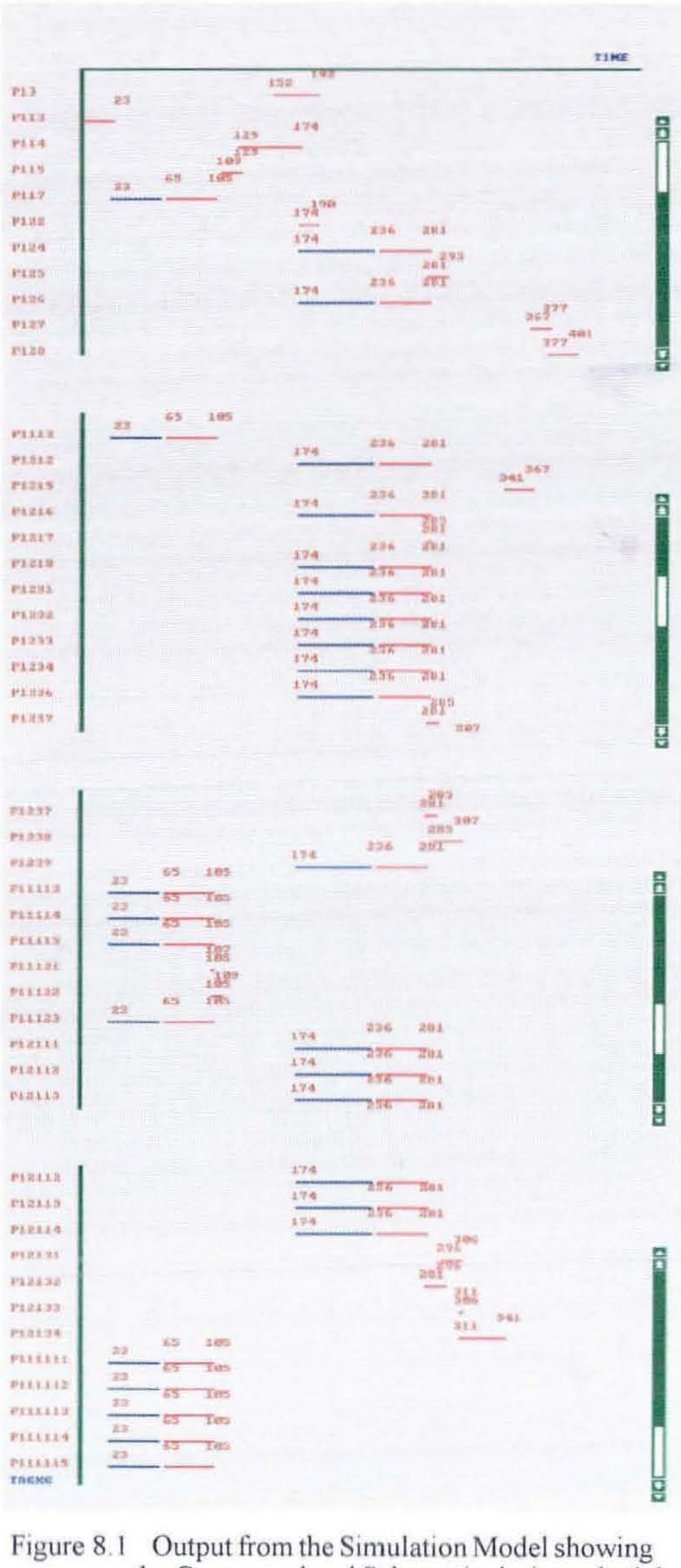


Figure 8.1 Output from the Simulation Model showing the Conceptual and Schematic design schedule

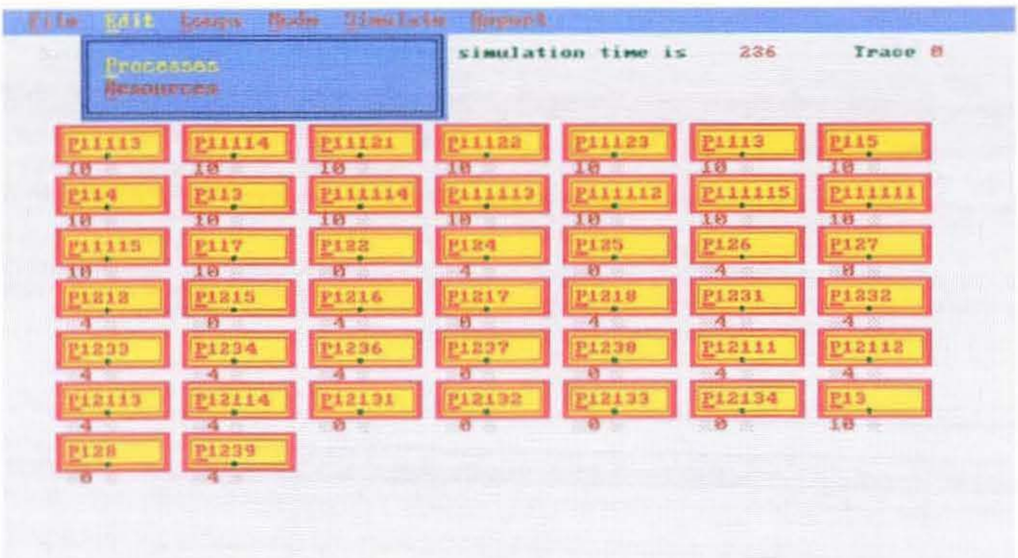


Figure 8.2 The impact of missing information

- Outline specs. production

The design manager can hence demonstrate the implications of the missing information to the site investigation team if they are responsible for the delay or to the Client if they are responsible for the delay due to negotiations for different site options.

Waiting for the missing information to be available

If the design manager decides to wait for the information to be available and anticipates a delay of two working weeks (80 hours), then they will assign 'closed gates' on the information links provided by the task 'site investigation' with time lapse of 80 hours. The design manager can then obtain a schedule of the revised dates for starting and completing the design tasks. The results are shown in Figure 8.3. The total time for the Conceptual and Schematic design stages will be delayed by 34 working hours (or 4 working days). This is a result of a delay in commencing the second iteration of the iterative design tasks due to the problem of missing information from the site investigation process. The second iteration will start at simulation time 190+80 = 270. The delay of 34 working hours represents the difference between 270 and the end of the finish time of the first iteration (236); (270-236 = 34)

8.4 ASSESSING THE IMPACT OF ASSUMED INFORMATION

Results of the interviews with designers and design managers showed that missing information is supplemented by assumptions from designers whilst confirmation on these assumptions is sought from the source. Any deviation from these assumptions may require re-design and hence subsequent tasks will be affected.

If, while monitoring the DFDs as the design process evolves, the design manager decided to proceed with a certain design task before receiving all the necessary information based on assumptions (e.g. assuming the loads to proceed with the foundations design) they will do the following:

- Run the simulation model while setting the switch representing the information link to 'off'. This means that the dependent design task would not wait for the required information and would proceed based on assumed information.

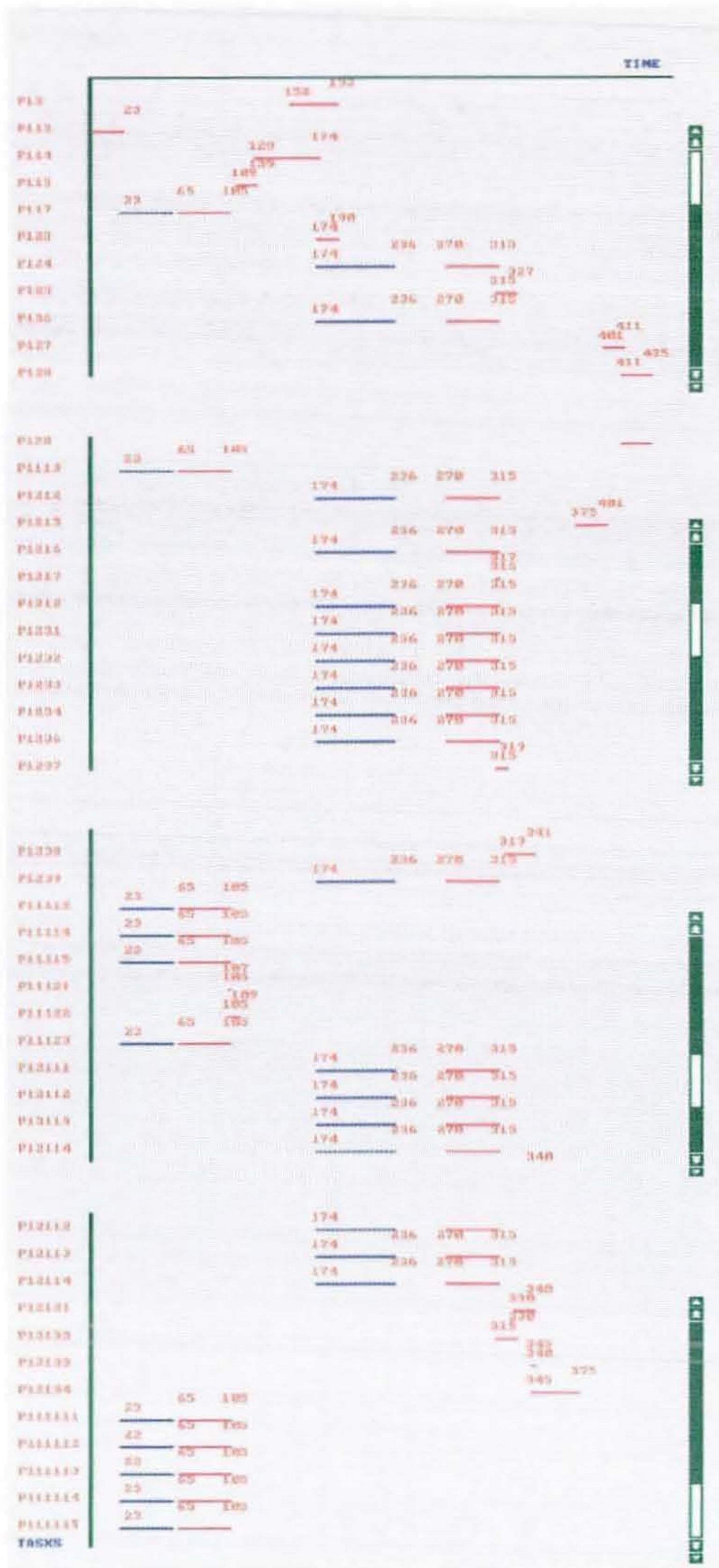


Figure 8.3 The impact of waiting for the missing information on the design schedule

- The design manager will observe (while the simulation is running) the 'conditional' tasks which would be performed initially based on assumed information either directly (the assumed information is used by the task) or indirectly (the task received information from another task which has been performed based on assumed information).

- The simulation model will produce a revised programme showing the timing of starting and finishing the affected design tasks based on assumed information, the timing of receiving the requisite information, and the timing of starting and finishing those tasks after receiving their requisite information with reduced durations. (The reduction factor should be determined by the design manager based on his experience before running the simulation.)

- The design manager will inform the relevant designers of the tasks that should be performed based on assumptions and that they should allow for these assumptions in their design and be aware that a re-design is possible. Should there be any deviations from these assumptions, the design manager will be aware of the affected design tasks.

Example

Considering the previous example in section 8.3, assume the design manager is required to proceed with the design of the foundations at an earlier time in order to release information to the contractor. The task 'consider foundations options' will be initially performed based on assuming the following information:

- Soil conditions
- Information from the definition brief about the site
- Structural loads

To represent these assumptions, the design manager will refer to the simulation model and set the 'switches' on these information links to 'off'. This means that the designers will proceed with the foundations design based on assumed information until they receive the precise information. When the precise information is available, a second iteration for the affected tasks will take place with a duration which is reduced by an arbitrary factor based on the design manager's experience; 80% in this example.

Running the simulation model will reveal that the following tasks will be carried out initially based on assumed information:

- Consider foundations options
- Decide on foundations type
- Produce foundations scheme design documents
- Structural design review
- Outline specs. production
- Revise cost estimate

The design manager will inform the relevant designers of these tasks that their designs should be performed based on assumptions and that they should allow for these assumptions in their design and be aware that a re-design is possible. The simulation model will produce a revised programme showing the timing of starting and finishing the affected design tasks based on assumed information, the timing of receiving the precise information and the timing of starting and finishing these tasks after receiving the requisite information with reduced duration. This is shown in Figure 8.4. The overall design duration was reduced from 401 hours to 380 hours (a saving of 21 design hours) as a result of starting the aforementioned design tasks based on assumed information.

8.5 BETTER UTILISATION OF DESIGN RESOURCES AND A REDUCED PROJECT DURATION THROUGH THE PHASED RELEASE OF INFORMATION

If running the simulation model revealed non levelled resource histograms and/or late design completion time, the design manager can assess the benefits of releasing the information from different design tasks in phases. When the information is released in phases, dependent tasks do not wait until the task providing the information is completed i.e. there will be an overlap in the tasks performance. The design manager will use their experience to decide on the degree of overlap i.e. the percentage of completion of the task providing the information after which the task can release the information.

By running the simulation using different scenarios of phasing information, the design manager can achieve a smoothed resource utilisation and/or a reduced project duration. This feature would be more beneficial at the late stages of design when the design tasks are more defined and there are several types and a considerable number of design resources.

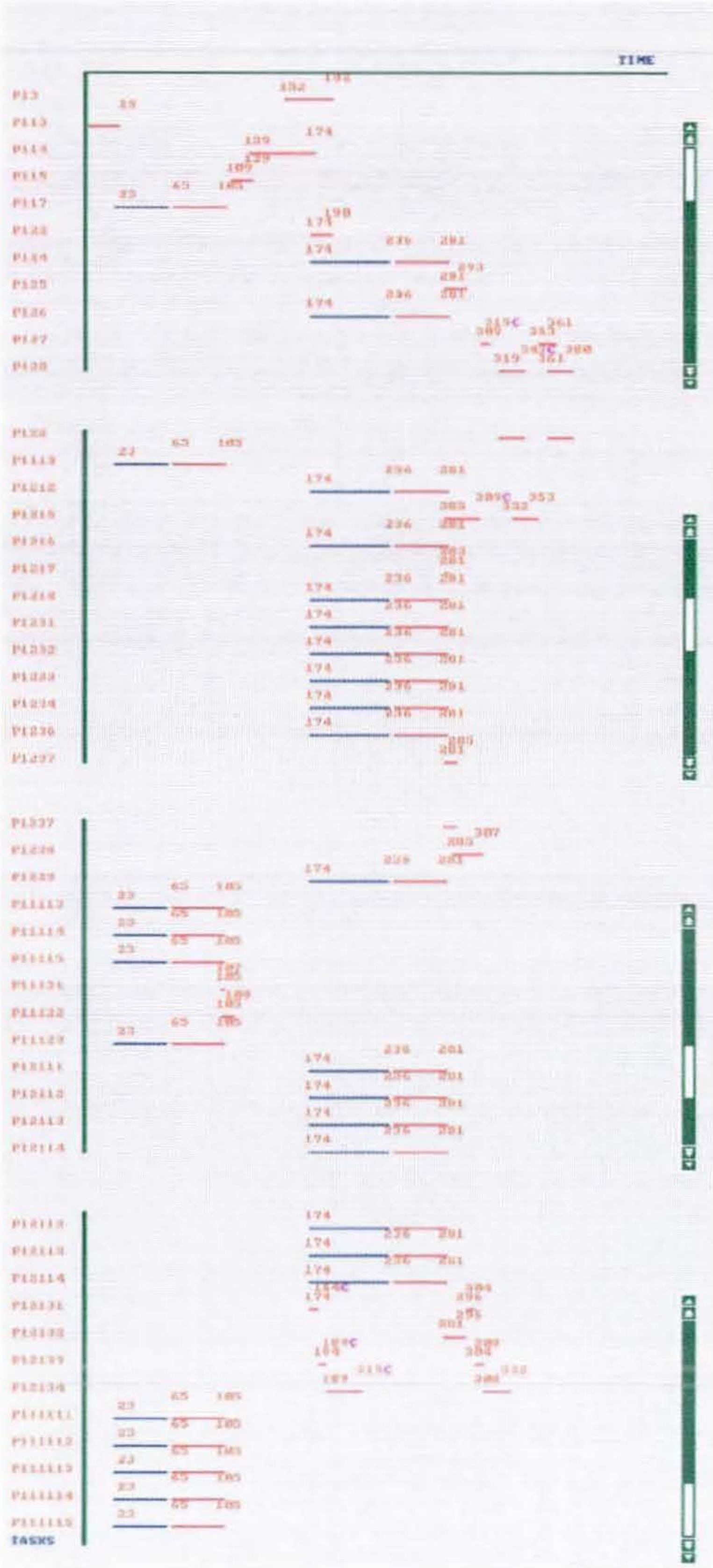


Figure 8.4 The impact of assumed information on the design schedule

Example: Reduced project duration through phased release of information:

Assume that the Client in the previous example is exerting pressures on the design manager to reduce the design duration for the project. The design manager decides to assess the impact of releasing information from some design tasks in phases on the design duration i.e. the dependant tasks will not wait for these tasks to be finished completely. The design manager will phase the release of information from the tasks 'Concept Design Report Production' and 'Produce Definition Brief in the manner represented in Table 8.1.

Running the simulation will present for the design manager a revised schedule for the design tasks as a result of releasing the information in phases. The total duration for the Conceptual / Schematic design stages has been reduced from 401 hours to 374 hours. Therefore, a saving of 27 hours (3.5 working days) is achieved. This is illustrated in Figure 8.5.

Dependant Task	**Percentage completion required from the task 'Concept design report production'**	**Percentage completion required from the task 'Produce Definition Brief'**
Site Investigation	30%	
Scheme Drainage Design	40%	50%
External works scheme design	80%	70%
Floor slab scheme design	40%	
Establishing fire rating requirements	30%	40%
Decide on finishing materials	40%	60%
Consider foundations options	20%	40%

Table 8.1 Phased release of the design information

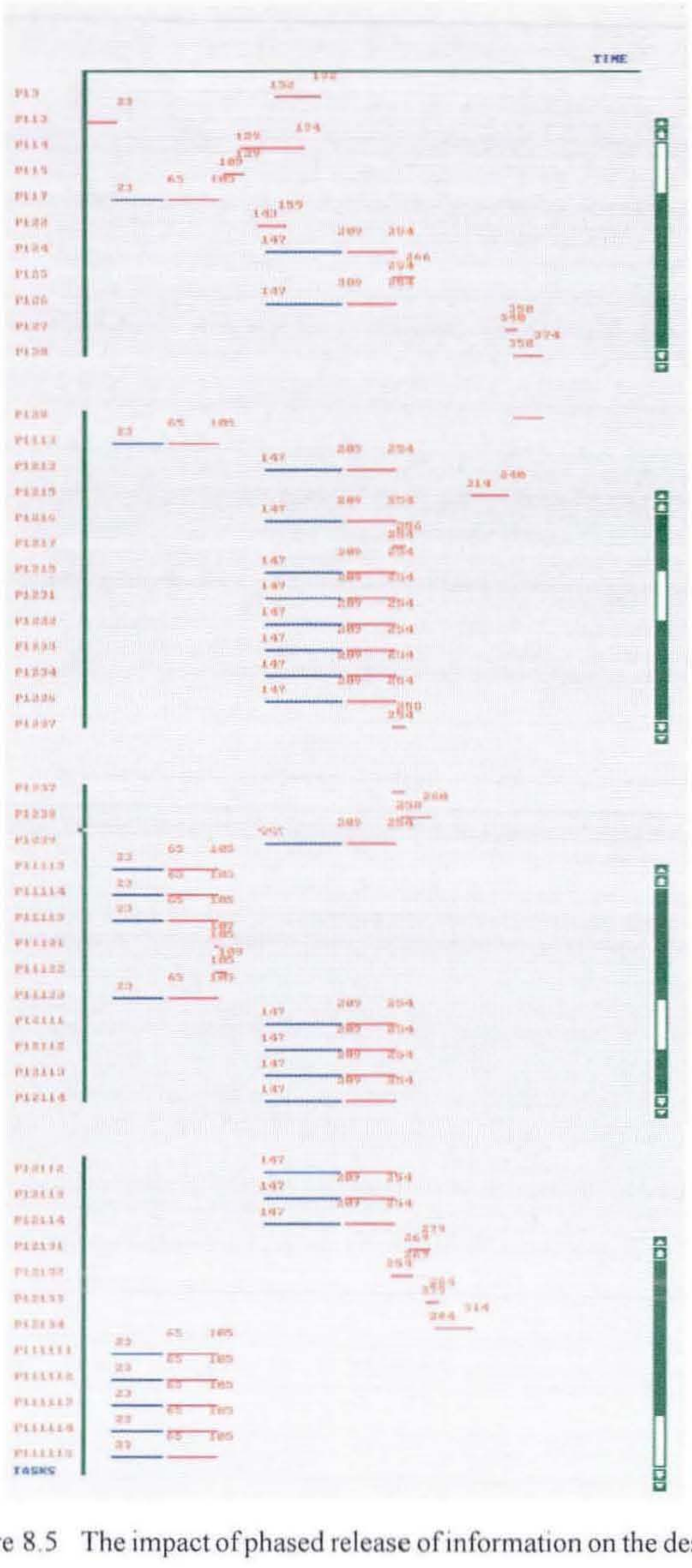


Figure 8.5 The impact of phased release of information on the design schedule

8.6 ALLOWING FOR 'GATE KEEPING' OF INFORMATION

One of the results of the industry interviews and survey was that communication problems during the design process result in 'gate keeping' of information either intentionally or non-intentionally. Knowing the design team members, their locations (i.e. working within the same building or in different cities), problems in similar projects, and other specific project related circumstances the design manager can anticipate the information flows which could be withheld by designers and can estimate the duration that this information would be withheld. By running the simulation under different scenarios of 'gate keeping', the design manager can identify the 'critical' information links where if information is withheld, the project completion time would be delayed. These information links would represent the 'bottle necks' for the project information flow and should be given particular attention by the design manager. If the design manager judges that these 'bottle necks' are inevitable, he/she should allow for such 'gate keeping' of information in the design programme.

Example

Considering the same example as that described in section 8.3. Assume that the design manager decides to assess the impact of different scenarios of withholding information. The design manager will refer to the simulation model and run it initially without any constraints to obtain a scenario of the design project against which other scenarios would be compared. The total design time for the Conceptual/Schematic design stage is predicted to be 401 design hours (50 working days).

If the design manager wants to assess the impact of withholding information from the source 'Building Control Officer' to the task 'Establish fire rating requirements' and anticipates a delay of one week (40 working hours) until this information is released, then they will assign a closed gate with time lapse of 40 hours on the relevant information link. Running the simulation will present the impact of withholding the information as illustrated in Figure 8.6. The total duration for the Conceptual / Schematic design stage is delayed by one week (40 hours) due to the delay of starting the second iteration of the loop. Therefore the information provided by the Building Control Officer is considered 'critical' to the project completion time.

Another scenario may be that the design manager decides to assess the impact of withholding cost information provided by the services engineer to the quantity

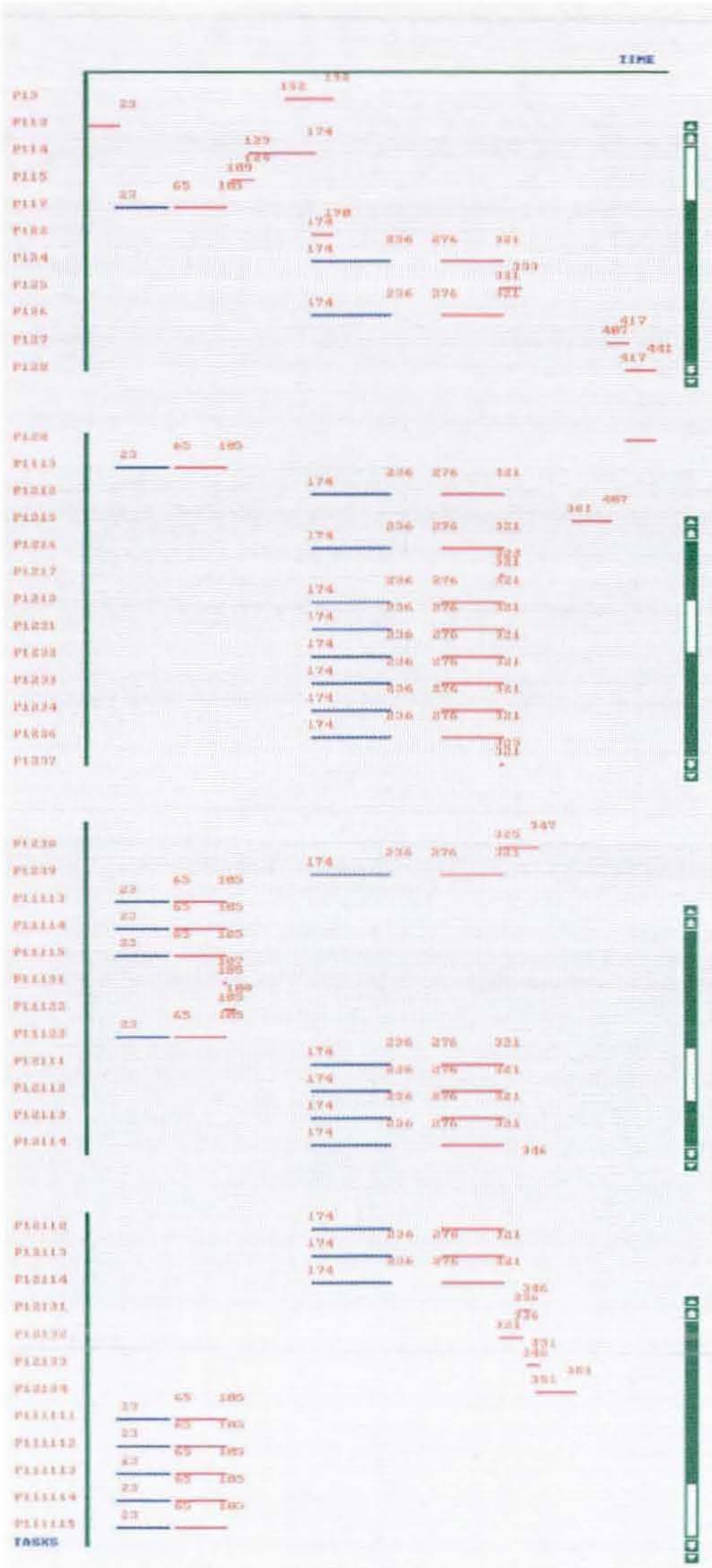


Figure 8.6 The impact of 'gate keeping' of information on the design schedule

surveyor which is necessary to carry out the task 'Revise Cost Estimate' and anticipates a delay of one week (40 hours). The design manager will then assign a closed gate with time lapse of 40 hours on the information link between the tasks 'Services Scheme Design' and 'Revised Cost Estimate'.

Running the simulation will reveal that in spite of the 'gate keeping' of the information, the duration for the Conceptual / Schematic design stage has not been affected (401 hours). This is because in order to complete the task 'Revise Cost Estimate', other information is required from other design tasks. This is illustrated in Figure 8.7. Therefore, the cost information that should be provided by the services engineer to the quantity surveyor is not critical to the project completion time.

8.7 ASSESSING THE IMPACT OF UNCERTAINTIES AND CARRYING OUT RISK ANALYSIS

During any design project it is inevitable that unforeseen events result in changes to the durations of certain design tasks. One of the features of the simulation model is the capability of random sampling for the design tasks durations with the assumption that these durations follow a normal distribution having a mean equals to the duration estimated by the design manager. The design manager will use the simulation model to assess the impact of the unforeseen uncertainties and carry out a risk analysis in the following manner:

(i) The design manager will run the simulation in the stochastic mode for several times while assuming a certain standard deviation for the tasks durations.

(ii) Record the design completion time for every run.

(iii) Assuming that these records follow a normal distribution, calculate the mean and standard deviation for the design completion time.

(iv) Calculate the probability of completing the design in a period of less than a required duration, say 250 hours.

(v) If this probability is < 90% (or any other required probability), the design manager will assume another standard deviation for the tasks durations and repeat steps (i) to (iv). If this probability >= 90%, he/she will proceed to step (vi).

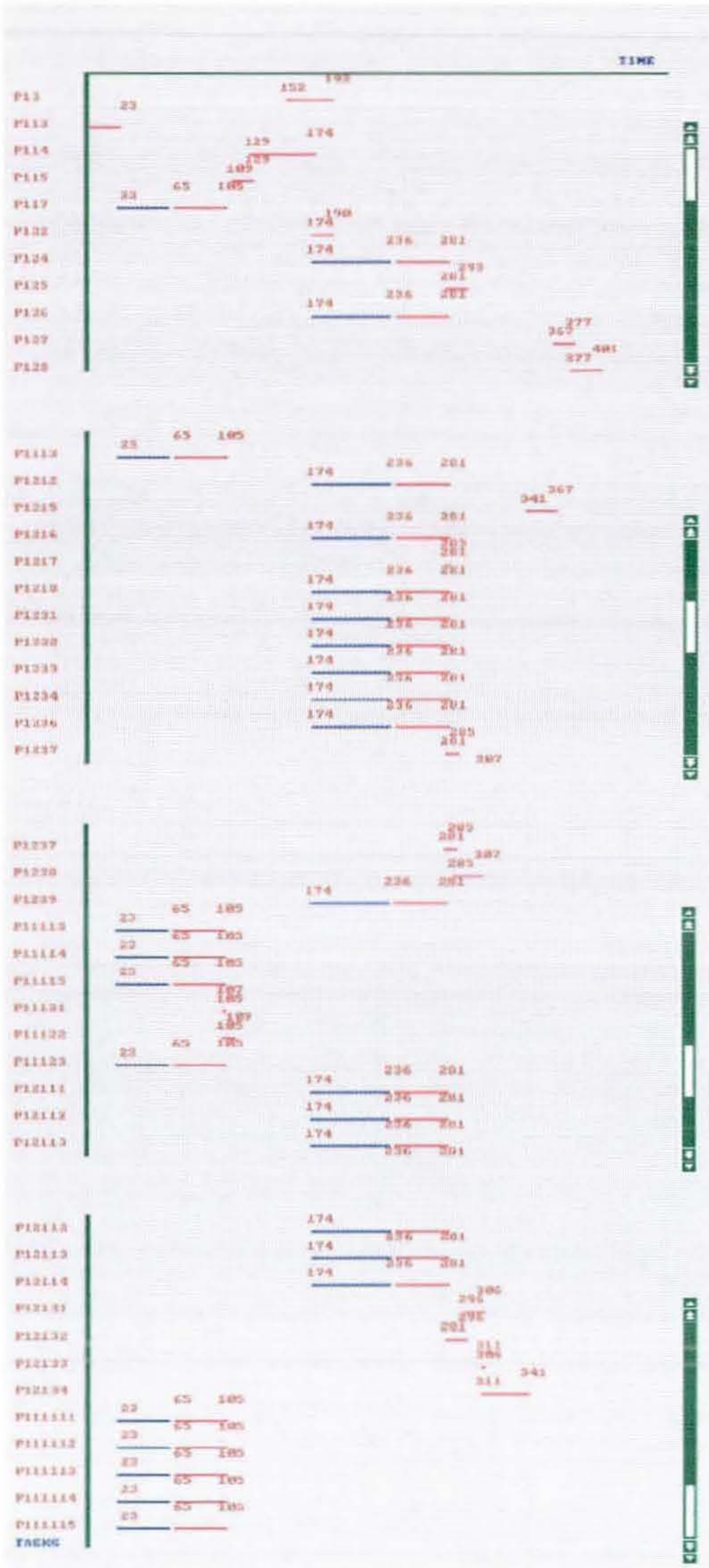


Figure 8.7 Another scenario for 'gate keeping' of information

(vi) knowing the standard deviation and mean for each task duration, the design manager will calculate for each task the 'safe' duration which they should be 95% confident (or any other level of confidence) that the task will be carried out in less than or equal to the 'safe' duration. They should make every endeavour not to exceed this duration.

Example

The stochastic running of the simulation model showed that a standard deviation of 2 is required for the tasks durations in order to have a probability of 90% of completing the design project in less than or equal 250 hours.

From statistics tables for normal distribution, the value of Z for a level of confidence 95% is equal to 1.65.

The safe duration of a task of mean = 15 will be calculated using the formula:

$$Z = \frac{X - \overline{X}}{\sigma}$$

where

$\overline{x} = mean$

$\sigma = standard\ deviation$

$$1.65 = \frac{x - 15}{2}$$

$$x - 15 = 3.3$$

$$\therefore x \approx 18 \text{hours}$$

Therefore the design manager may be 95% confident that the task will be completed in less than 18 hours.

This feature would be more beneficial in the later stages of design when the design tasks are discrete and more defined and when any delays may have serious impact on the tender and/or construction stage. However, to demonstrate this feature, data from the Conceptual/Schematic design stages of the LUT Engineering Complex case study is used in the following example.

Example:

Considering the example based on data collected during the LUT Engineering Complex case study, the design manager decides to calculate the probability of completing the Conceptual/Schematic design in five weeks (400 hours). The design manager will refer to the simulation model and select from the 'Mode' menu the Stochastic Mode. The simulation model will be run for several times, say 25 times, while setting the standard deviation for the task durations to 2 and the design manager will record the completion time for each run. The following records represent the results from running the simulation stochastically for 25 times:

387, 401, 409, 393, 410, 396, 386, 399, 399, 399, 381, 381, 395, 388, 388, 397, 389, 392, 397, 397, 393, 392, 395, 399, 391.

Assuming that these records follow a normal distribution, therefore:

$\bar{x} = 394$ hours

$\sigma = 7.01$

Where $\bar{x}$ = mean and σ = standard deviation

$$\therefore Z = \frac{x - \bar{x}}{\sigma}$$

$$\therefore Z = \frac{400 - 394}{7.01} = 0.86$$

From the statistics tables for normal distribution, the probability for completing the Conceptual/Schematic design stages in less than 400 hours (5 weeks) is 80.5%. If the design manager requires higher probability, they will repeat the previous steps while changing the standard deviation for the task durations until the required probability is achieved. It is assumed in this example that the design manager is satisfied with 80% probability. Moreover, the design manager is aware that the Client requires starting work on site as soon as possible and hence releasing information related with the foundation design is crucial. From the Data Flow Model and the Design Structure Matrix, the design manager will identify the task 'site investigation' as the task that controls the activities related with the foundations design. Therefore, it is required to calculate the 'safe' duration for 'site investigation' which he/she should be 95% confident (or any other level of confidence decided by the design manager) that the task should be carried out in a time less than or equal to the 'safe duration'. The task 'site investigation' was assigned a duration of 16 hours to run the simulation model (this duration is relatively short due to the fact that this particular case study is related with the extension of an existing development and hence the site is already known).

Therefore, this duration represents the mean for the duration distribution of this task. The standard deviation is equal to 2 as previously shown. From the statistics tables for normal distribution, the value of Z for a level of confidence 95% is equal to 1.65. Therefore, the safe duration will be calculated using the formula:

$$Z = \frac{x - \bar{x}}{\sigma}$$

where

$\bar{x}$ = mean

σ = standard deviation

$$\therefore 1.65 = \frac{x - 16}{2}$$

$$\therefore x \cong 20 \text{ hours}$$

Hence, the design manager may be 95% confident that the site investigation will be completed in less than 20 hours.

8.8 HANDLING THE PROBLEM OF ITERATION

One of the characteristics of design identified in this research is its iterative nature. The matrix modelling for the design tasks will assist the design manager in identifying the loops of iterative design tasks. The design manager will be aware that these tasks would be undertaken in parallel in an iterative fashion. These tasks would normally be multi-disciplinary and hence the knowledge of these tasks will assist the design manager in the management of the interfaces between different design disciplines. Knowing these tasks also will assist the design manager in the selection of resources as the designers performing tasks which fall within the same iterative loop should work in proximity to facilitate the communication process and increase the design efficiency. For example, if an architectural design task will be carried out iteratively with a mechanical design tasks, then both the designers of these tasks should work in the same office and not in different branches of the design organisation. Resources for these tasks should also be rationalised by the design manager knowing that these tasks should be undertaken not only simultaneously but also iteratively such that tasks of the same discipline would be performed, where possible, by the same designer. The design manager will use their experience to decide on the number of iterations for each iterative loop. The simulation model can handle up to two iterations but there is the possibility of increasing the number of iterations. Running the simulation model will allow the design manager to investigate the following:

- Different number of iterations
- Different durations for each iteration
- Nominate different tasks which would initiate the loop
- Apply tearing techniques to the matrix modelling to reduce the size of the loops and re-run the simulation to assess the impact on the remaining design tasks, on the design completion time, and on the resources utilisation.

It should be noted that these techniques will not eliminate iteration in carrying out some design tasks as the aim is not to change the nature of design. They will however provide the design manager with a tool to assist in the identification of iterative tasks and allow for iteration in the planning of design work. This will be achieved by considering different scenarios which will improve the management of the process.

Example

This example is based on data from a real design problem of a plant room described by Newton (1995) at the detailed design stage. A plan and a section of the plant room are shown in Figure 8.8. Design resources were included in this example. The hierarchy of the data flow model developed for the detailed design stage of the plant room is shown in Figure 8.9. Sample data flow diagrams are shown in Figures 8.10 and 8.11.

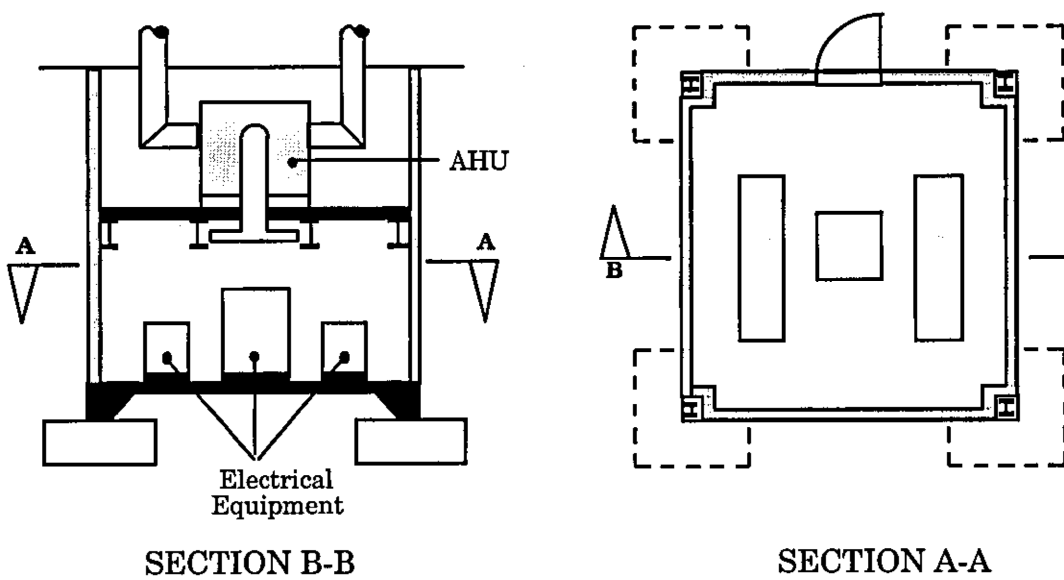


Figure 8.8 Plant Room Scheme Drawings

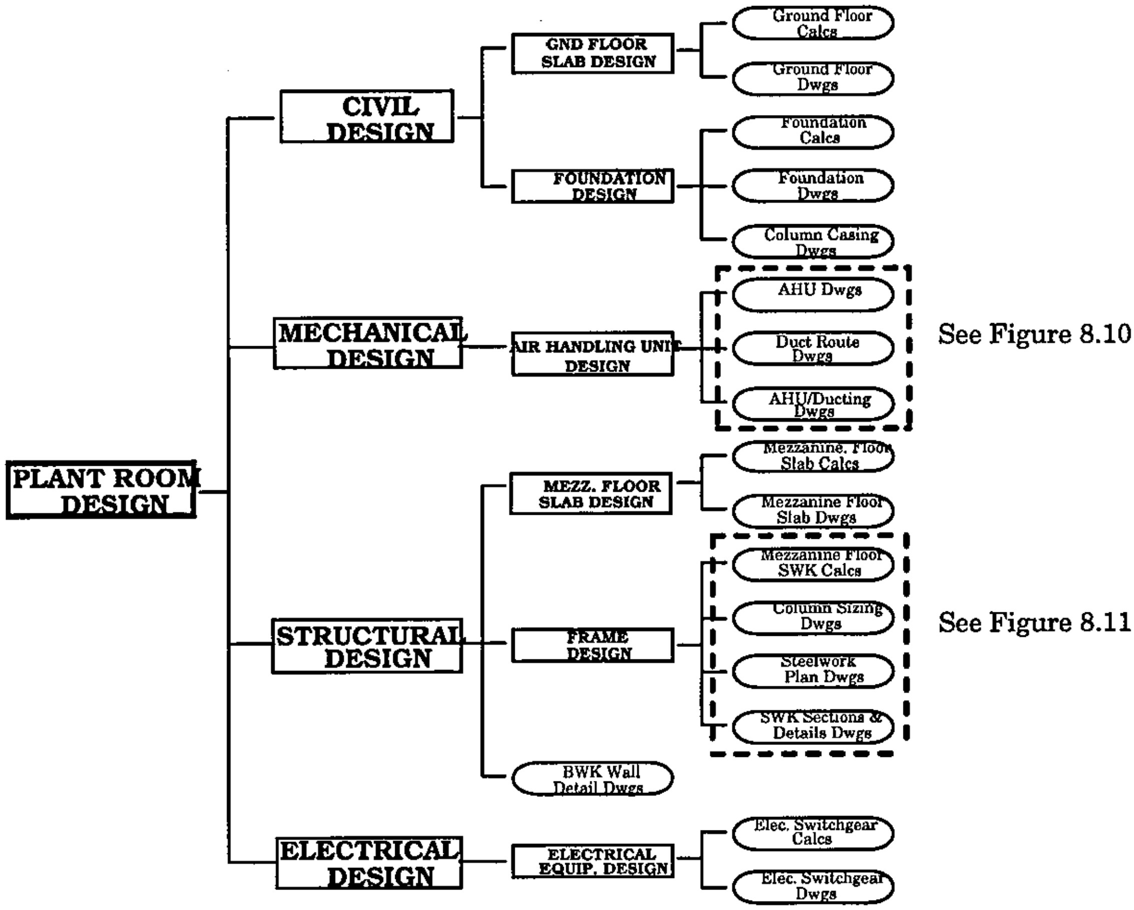


Figure 8.9 Plant Room Model Hierarchy

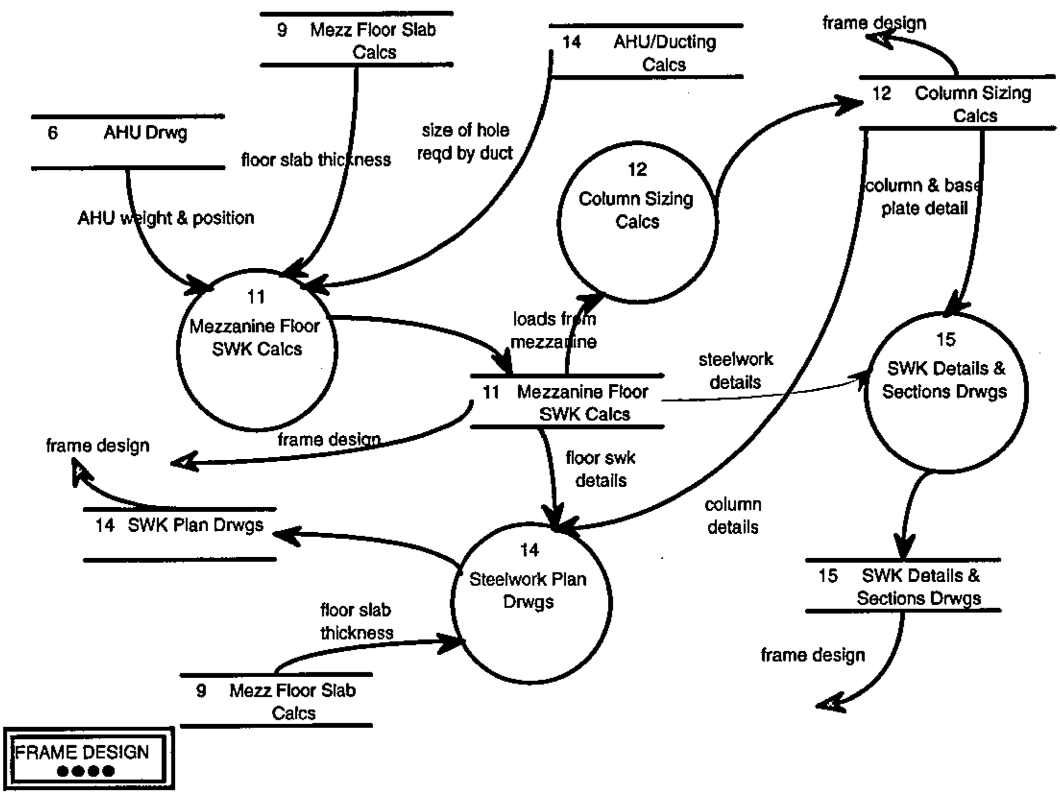


Figure 8.10 Data Flow Diagram for 'Frame Design' (adapted from Newton 1995)

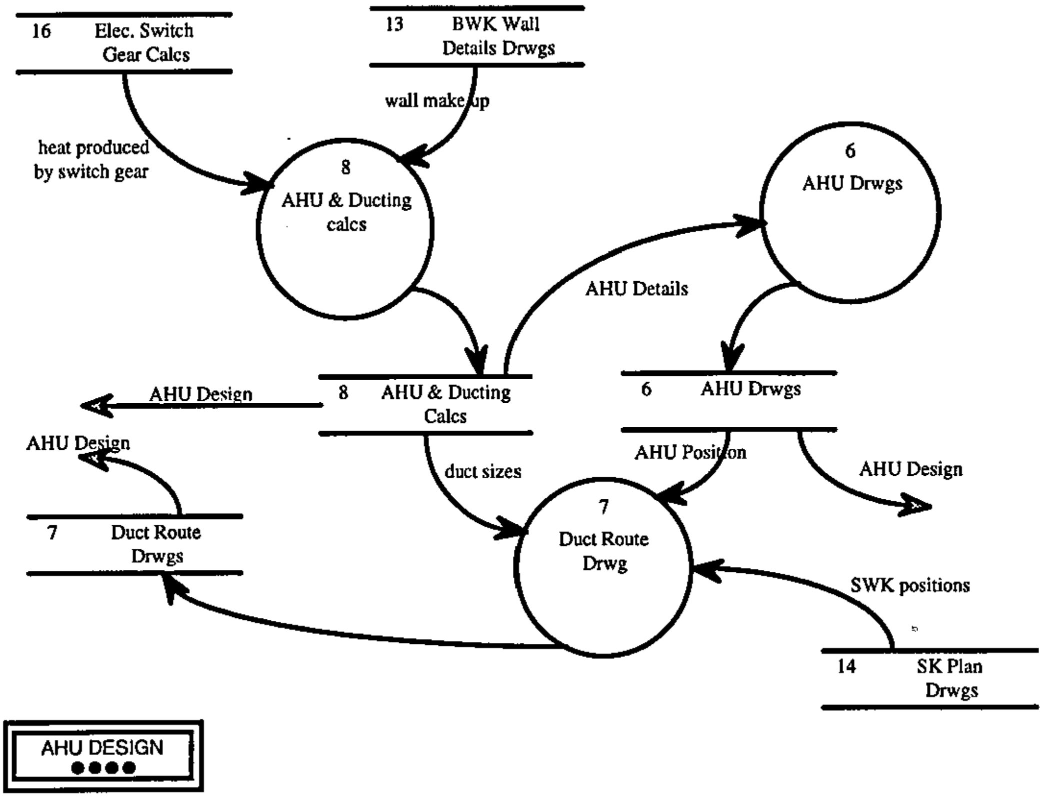


Figure 8.11 Data Flow Diagram for 'AHU Design' (adapted from Newton 1995)

The Design Structure Matrix showing the tasks dependencies and inter-dependencies is shown in Figure 8.12. Partitioning the matrix revealed an iterative loop of design tasks containing the following tasks:

- Air handling unit drawings
- Air handling unit / ducting calculations
- Mezzanine floor slab calculations
- Mezzanine floor steelwork calculations
- Column sizing calculations
- Brick work wall details drawings
- Electrical switchgear calculations

The partitioned DSM showing the loop is shown in Figure 8.13. Hence the design manager will be aware that these tasks would be undertaken in parallel in an iterative fashion. These tasks are multi-disciplinary including mechanical, structural and electrical disciplines. Therefore designers from these disciplines should work in proximity to facilitate the communication process and increase the design efficiency.

PSM Version b.30	1 :	2 :	3 :	4 :	5 :	6 :	7 :	8 :	9 :	10	11	12	13	14	15	16	17
1 : Ground floor calcs	1 :												X			X	
2 : Ground floor drgs	X	2 :			X								X				X
3 : Foundations Calcs	X		3 :									X					
4 : Foundation Drgs			X	4 :											X		
5 : Column casing drgs				X	5 :								X		X		
6 : Air handling unit drgs						6 :		X									
7 : Duct route drgs						X	7 :	X						X			
8 : AHU/Ducting calcs								8 :					X			X	
9 : Mezzanine floor slab calcs.								X	9 :		X						
10 : Mezzanine floor slab drgs						X	X		X	10				X			
11 : Mezz. floor steel work calcs						X		X	X		11						
12 : Column sizing calcs											X	12					
13 : Brickwork wall details drgs								X				X	13			X	
14 : Steelwork plan drawings											X	X		14			
15 : Steelwork sections											X	X		X	15		
16 : Electrical switchgear calcs								X								16	
17 : Electrical switchgear drgs																X	17

Figure 8.12 The Design Structure Matrix for the detailed design of a plant room before partitioning

PSM Version b.30	6 :	8 :	9 :	11	12	13	16	17	14	15	7 :	10	1 :	3 :	4 :	5 :	2 :
6 : Air handling unit drgs	6 :	X															
8 : AHU/Ducting calcs		8 :				X	X										
9 : Mezzanine floor slab calcs.		X	9 :	X													
11 : Mezz. floor steel work calcs	X	X	X	11													
12 : Column sizing calcs				X	12												
13 : Brickwork wall details drgs		X			X	13	X										
16 : Electrical switchgear calcs		X					16										
17 : Electrical switchgear drgs							X	17									
14 : Steelwork plan drawings				X	X				14								
15 : Steelwork sections				X	X				X	15							
7 : Duct route drgs	X	X							X		7 :						
10 : Mezzanine floor slab drgs	X		X						X		X	10					
1 : Ground floor calcs						X	X						1 :				
3 : Foundations Calcs					X								X	3 :			
4 : Foundation Drgs										X				X	4 :		
5 : Column casing drgs						X				X					X	5 :	
2 : Ground floor drgs						X		X					X			X	2 :

Figure 8.13 The Design Structure Matrix after partitioning

Data from the data flow model and the design structure matrix were used to run the simulation model. (Data about durations and resources available for this project were provided by Newton.) Running the simulation will provide the design manager with a schedule of the design tasks in a bar chart format with a total design time of 65 hours. The produced schedule coincides with the network produced by Newton (1995). This is shown in Figures 8.14 and 8.15 which confirms the robustness of the simulation model. The writer considered that all iterative design tasks within the same loop are performed concurrently in an iterative fashion as there should be continuous exchange of information during performing these tasks. Whilst the network produced by Newton shows only the logic of performing the design tasks but does not have a time scale, the simulation model developed by the writer encompasses such time scale which is reflected in the schedule produced after running the model. The simulation model also allows the design manager to investigate different alternatives on 'what if' basis.

Applying tearing techniques to the matrix modelling and re-running the simulation

If the design manager decides to apply tearing techniques (described in section 2.2.7) to the DSM then they should assess the relative importance of each information in the iterative loop by consulting the relevant tasks and information flows represented in the data flow model. This will result in identifying the least essential dependencies between design tasks. These dependencies are represented by the letter O in Figure 8.16.

For example. task 11, the mezzanine floor steel work calcs. was dependent on three pieces of information : Air handling unit weight and position, from task 6; floor slab thickness, from task 9; and size of hole required by duct from task 8. From Figure 8.16 the design manager judged that information from tasks 8 and 9 are not absolutely necessary to perform task 11 whereas information from task 6 was deemed essential. These decisions, based on engineering judgement, suggest that the floor slab thickness (task 9) could be predicted with a fair degree of accuracy and the size of the holes required by the duct (task 8) would not affect the steel work spacing. Further possible 'tears' were identified and are also shown in Figure 8.16. Re-partitioning the matrix such that only information identified as 'tearable' appears above the diagonal will result in the matrix shown in Figure 8.17.

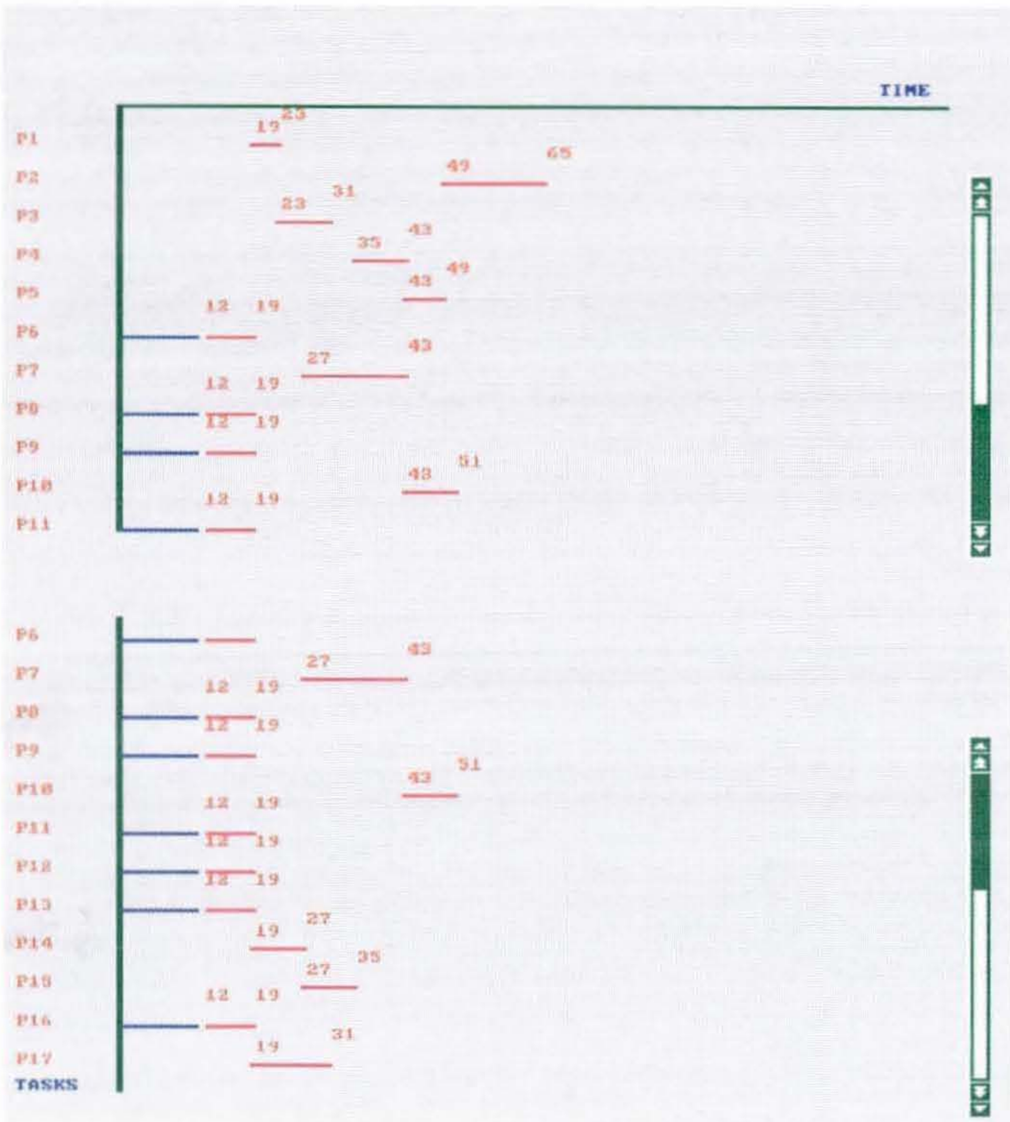


Figure 8.14 The Detailed design schedule of a plant room produced by the Simulation Model

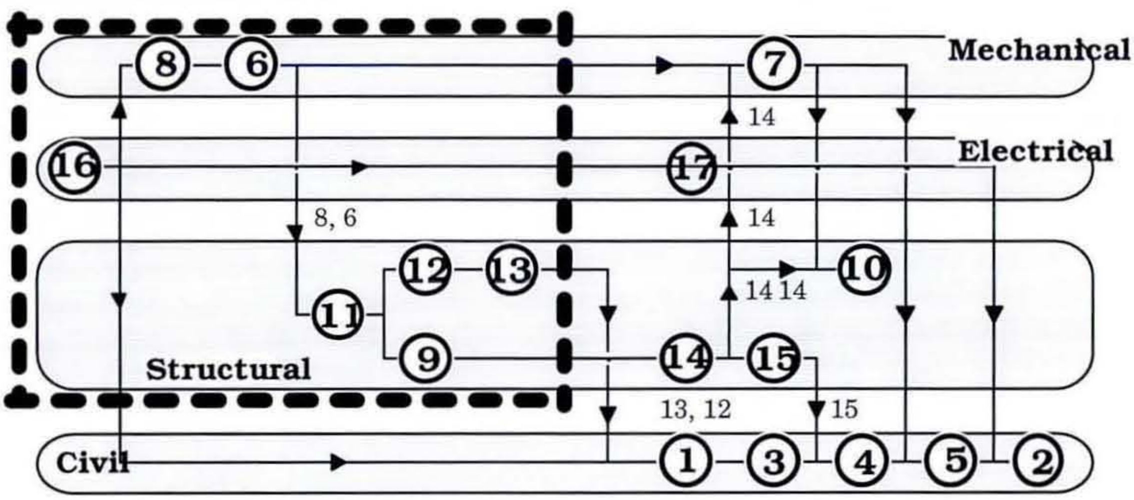


Figure 8.15 Network developed from DSM with iteration (adapted from Newton 1995)

PSM Version b.30	6 :	8 :	9 :	11	12	13	16	17	14	15	7 :	10	1 :	3 :	4 :	5 :	2 :
6 : Air handling unit drgs	6 :	X															
8 : AHU/Ducting calcs		8 :				O	X										
9 : Mezzanine floor slab calcs.		O	9 :	X													
11 : Mezz. floor steel work calcs	X	O	O	11													
12 : Column sizing calcs				X	12												
13 : Brickwork wall details drgs		X			X	13	X										
16 : Electrical switchgear calcs		O					16										
17 : Electrical switchgear drgs							X	17									
14 : Steelwork plan drawings				X	X				14								
15 : Steelwork sections				X	X				X	15							
7 : Duct route drgs	X	X							X		7 :						
10 : Mezzanine floor slab drgs	X		X						X		X	10					
1 : Ground floor calcs						X	X						1 :				
3 : Foundations Calcs					X								X	3 :			
4 : Foundation Drgs										X				X	4 :		
5 : Column casing drgs						X				X					X	5 :	
2 : Ground floor drgs						X		X					X			X	2 :

Figure 8.16 The Design Structure Matrix showing the least essential dependencies represented by 'O'

PSM Version b.30	16	17	8 :	6 :	11	12	14	15	9 :	7 :	10	13	1 :	3 :	4 :	5 :	2 :
16 : Electrical switchgear calcs	16		O														
17 : Electrical switchgear drgs	X	17															
8 : AHU/Ducting calcs	X		8 :									O					
6 : Air handling unit drgs			X	6 :													
11 : Mezz. floor steel work calcs			O	X	11				O								
12 : Column sizing calcs					X	12											
14 : Steelwork plan drawings					X	X	14										
15 : Steelwork sections					X	X	X	15									
9 : Mezzanine floor slab calcs.			O		X				9 :								
7 : Duct route drgs			X	X			X			7 :							
10 : Mezzanine floor slab drgs				X			X		X	X	10						
13 : Brickwork wall details drgs	X		X			X						13					
1 : Ground floor calcs	X											X	1 :				
3 : Foundations Calcs						X							X	3 :			
4 : Foundation Drgs								X						X	4 :		
5 : Column casing drgs								X				X			X	5 :	
2 : Ground floor drgs		X										X	X			X	2 :

Figure 8.17 The Design Structure Matrix after 'tearing'

Figure 8.17 shows that all the information above the diagonal could be estimated. If the design manager decides that such information does not need validation, therefore iterative design loops would disappear and the whole design process would be a set of tasks that could be performed either in series or in parallel. Running the simulation model based on the revised data from the DSM results in the schedule shown in Figure 8.18. The produced schedule coincides with the network produced by Newton (1995) based on matrix 'tears' and illustrated in Figure 8.19. This further confirms the robustness and the accuracy of the results produced by the simulation model.

8.9 IMPROVING RESOURCES MANAGEMENT

One of the conclusions from the industry interviews and surveys was that the allocation of appropriate resources and efficient resource utilisation is an essential requirement for the efficient management of the design process. The design manager may use the simulation model to investigate different combinations of resources prior to starting the design and study the utilisation histograms produced to identify the resources requirements at different times during the design programme. This will help to achieve a levelled resource utilisation. The design manager may also, whilst running the simulation, identify the design tasks which could have commenced at

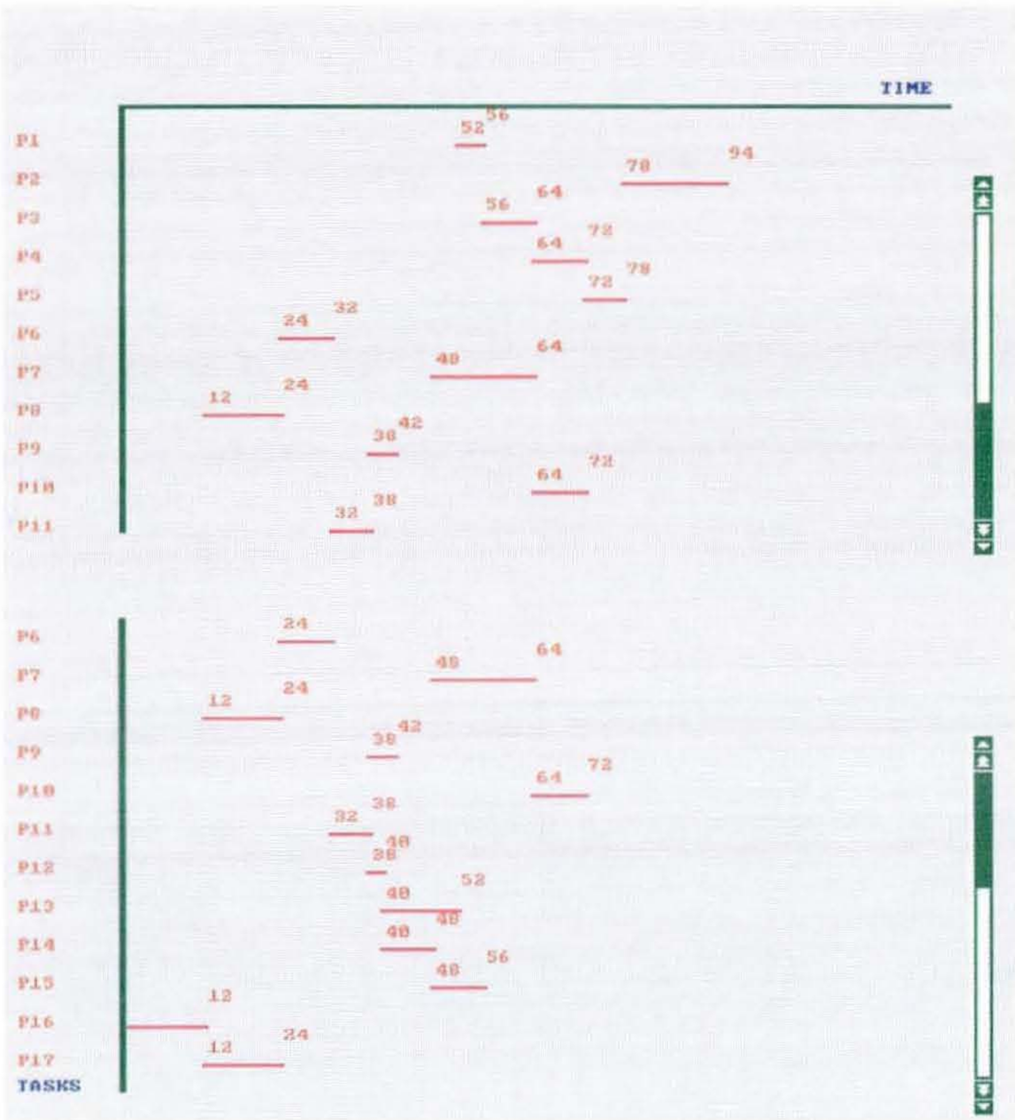


Figure 8.18 The Detailed design schedule of the plant room after tearing the iterative design loop

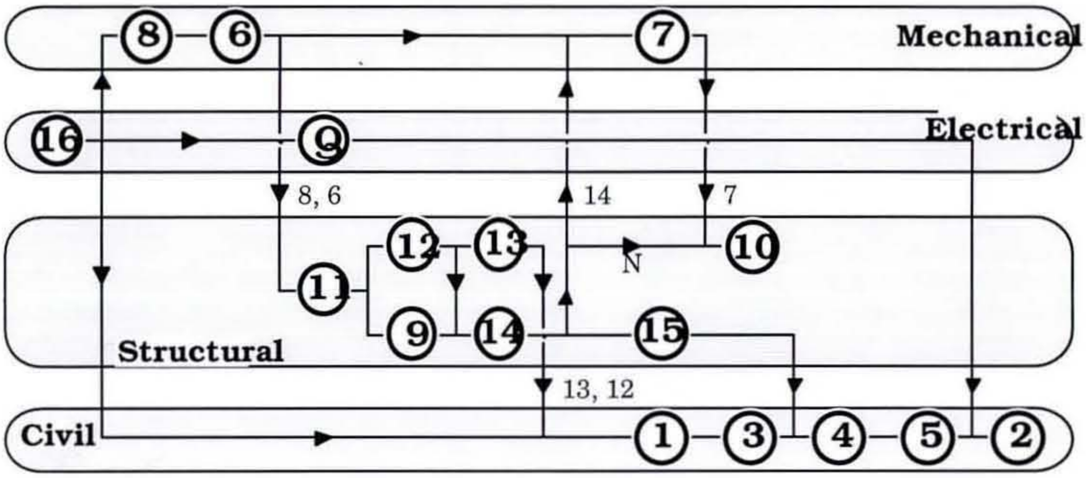


Figure 8.19 Network developed from DSM without iteration (adapted from Newton 1995)

earlier times but were unable to do so due to lack of resources. He/she would assess the trade-offs of increasing the number of resources to start certain tasks at earlier times.

Such an analysis is comparable with that undertaken by project managers using existing critical path software packages which, through their inability to accommodate iterative cycles of work, are inappropriate for design management.
This feature of the model is more beneficial at the detailed design stage when there are several types and considerable number of design resources.

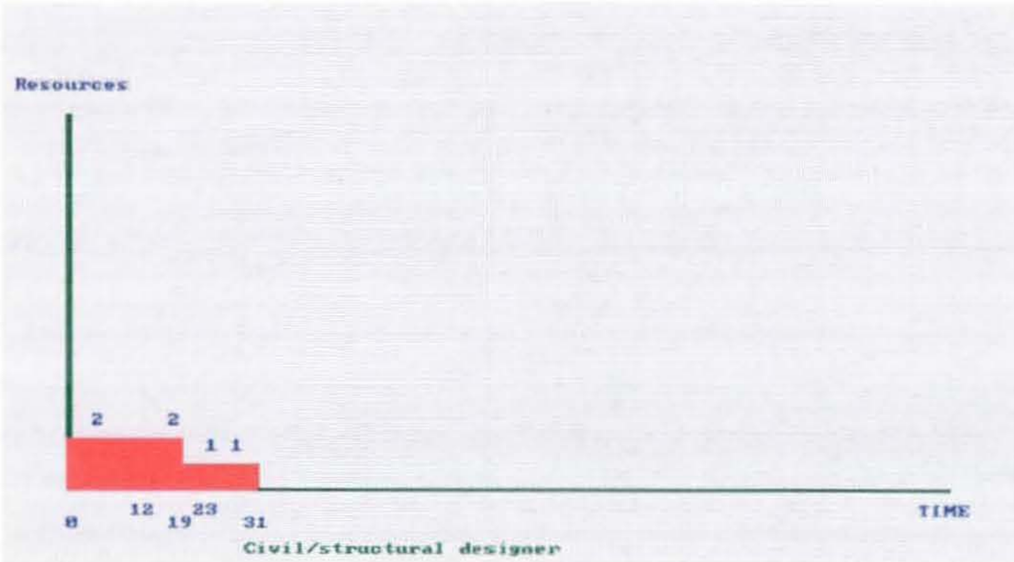
Example:
This example is based on data from the detailed design stage of a plant room described in section 8.8. Although due to the nature of such project the number of resources involved are relatively small, the purpose here is to illustrate the benefits that the produced tools would provide. The resources required for each of the design tasks are listed in Table 8.2.

No.	Task	Resources
1	Ground floor calcs.	1 civil/str. engineer
2	Ground floor drawings	1 civil/str. draftsman
3	Foundation calcs.	1 civil/str. engineer
4	Foundations drawings	1 civil/str. draftsman
5	Column casing drawings	1 civil/str. draftsman
6	Air handling unit drawings	1 services draftsman
7	Duct route drawings	1 services draftsman
8	Air handling unit / Ducting calcs.	1 services engineer
9	Mezzanine floor slab calcs.	1 civil/str. engineer
10	Mezzanine floor slab drawings	1 civil/str. draftsman
11	Mezzanine floor steelwork calcs.	1 civil/str. engineer
12	Column sizing calcs.	1 civil/str. engineer
13	Brick work wall details drawings	1 civil/str. draftsman
14	Steel work plan drawings	1 civil/str. draftsman
15	Steel work sections drawings	1 civil/str. draftsman
16	Electrical switchgear calcs.	1 services engineer
17	Electrical switchgear drawings	1 services draftsman

Table 8.2 Resources required for the detailed design of the plant room

The resources available for the project are : 2 civil/structural engineers, 2 civil/structural draftsmen, 2 services engineers and 2 services draftsmen. Running the simulation provides the design manager with the resource utilisation histograms shown in Figures 8.20 and 8.21. Figure 8.20 shows that two civil/str. draftsmen are required only during simulation time 43 to 51, otherwise one draftsman is enough. This also applies to the two services draftsmen whom will be required only during simulation time 27 to 31 as shown in Figure 8.21. Therefore the design manager can re-plan the design tasks to either dispose the two draftsmen to other jobs whilst they are idle or engage these draftsmen, if possible, in other tasks and reduce the durations of these tasks. Similar assessments may also be done for the services engineers and the civil/structural engineers.

If the design manager decides that the task mezzanine floor slab drawings requires two civil/structure draftsmen in lieu of one, then re-running the simulation with the revised data will present for the design manager the consequence. A message will appear on the screen 'P10 Insufficient resources' when this task has received its



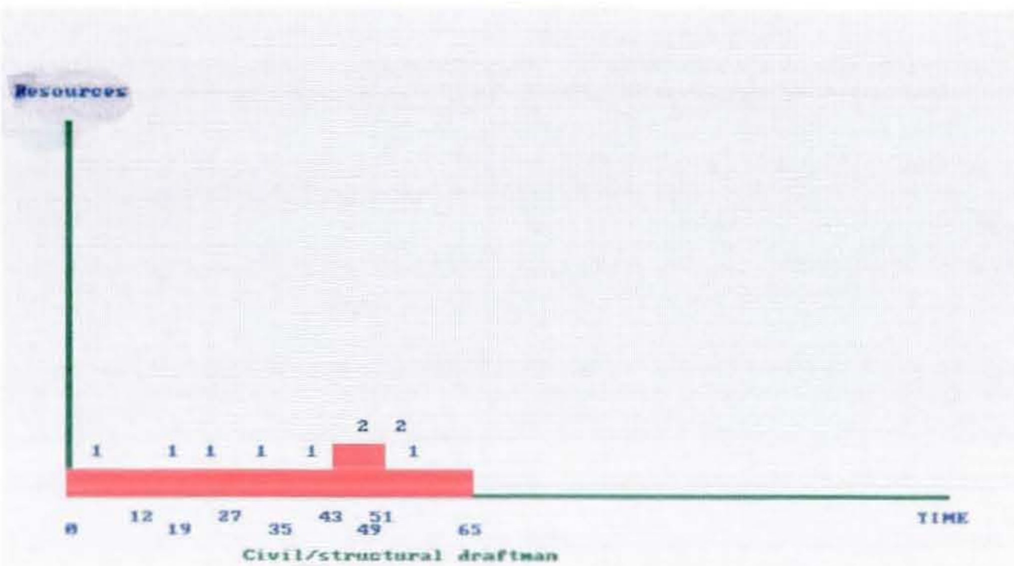
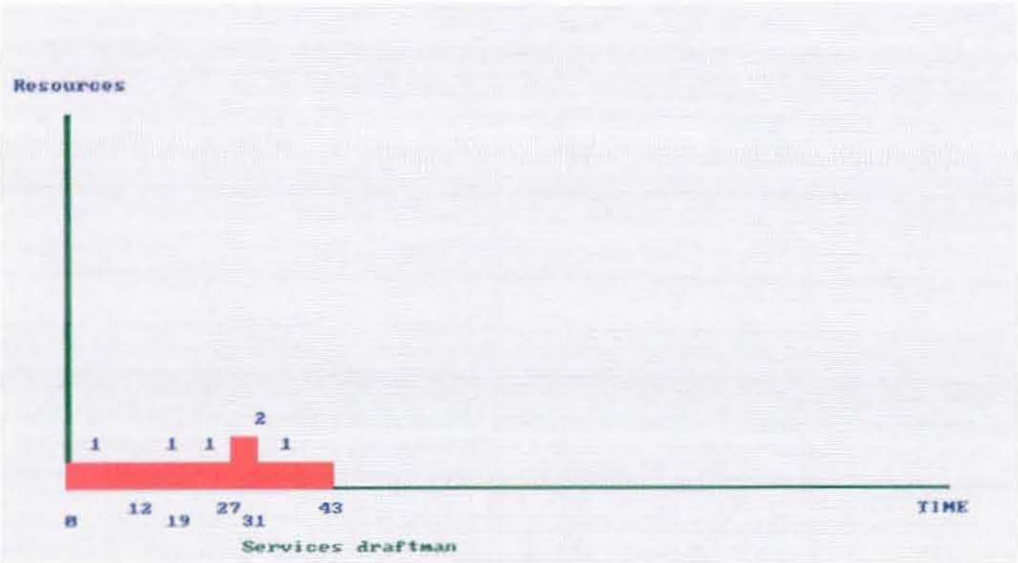


Figure 8.20 Resource histograms for civil/structure designers and draftsmen produced by the Simulation Model



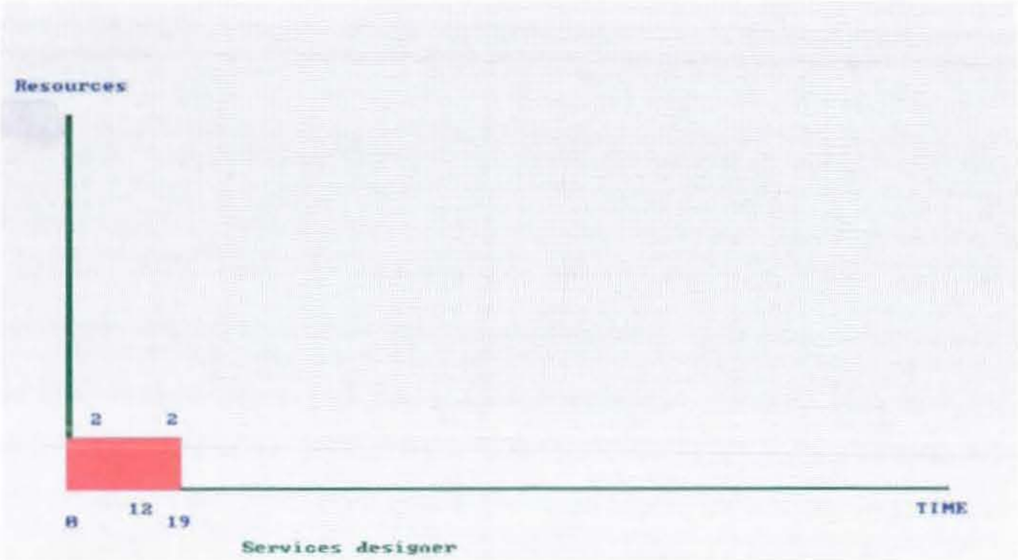


Figure 8.21 Resource histograms for services engineers and draftsmen

requisite information but cannot start due to insufficient resources. The task should be delayed until other task(s) that require civil/structure draftsmen are completed and release the engaged draftsmen. This results in a delay of eight design hours. This is illustrated in Figures 8.22 and 8.23.

8.10 ALLOWING FOR POOR QUALITY INFORMATION

One of the conclusions of the industry survey and interviews was that there is no formal way of measuring information quality and that the subjective estimation of information quality differs according to the sender and recipient of information. The simulation model can assist the design manager in overcoming this problem in the following manner:

- The design manager will allocate a global quality attribute for the design stage under consideration according to their judgement and experience. For example, this attribute may be 40% for scheme design while it would be 90% for detailed design. The quality of information exchanged must be equal to or greater than the global quality attribute in order that the task receiving the information is performed.

- The design manager will assess different scenarios of different quality attributes on the information links between design tasks or between a source (such as the Client) and a design task. The design manager will use their experience and perception for the expected quality of information among different designers. By running the simulation, the design manager will identify the design tasks which would be affected by a 'poor' information quality and will assess their impact on the overall design programme. The manager will identify also the 'critical' design tasks which <u>must</u> provide information of 'satisfactory' quality if there is not to be a considerable delay in the design duration or a delay in the commencement of a considerable number of design tasks. Particular attention will then be given to the management of these tasks to ensure that they provide information of 'satisfactory' quality.

- If the design manager judges that information of 'poor' quality is inevitable, this should be allowed for it in the design programme. The simulation tool may be used to demonstrate to the information originator (e.g. the Client or other designers) the impact of the quality of information they provide on the overall design process.

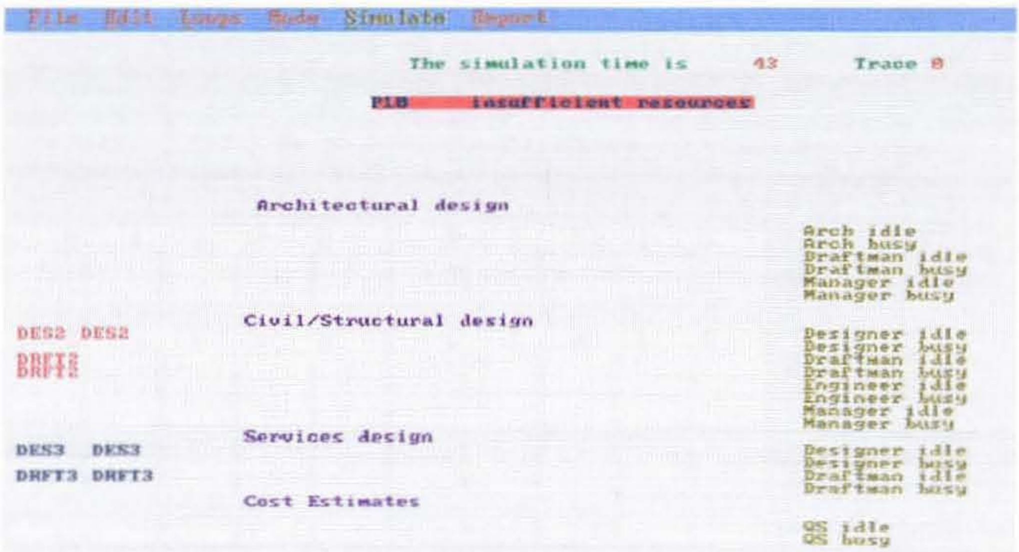


Figure 8.22 Simulating resource utilisation

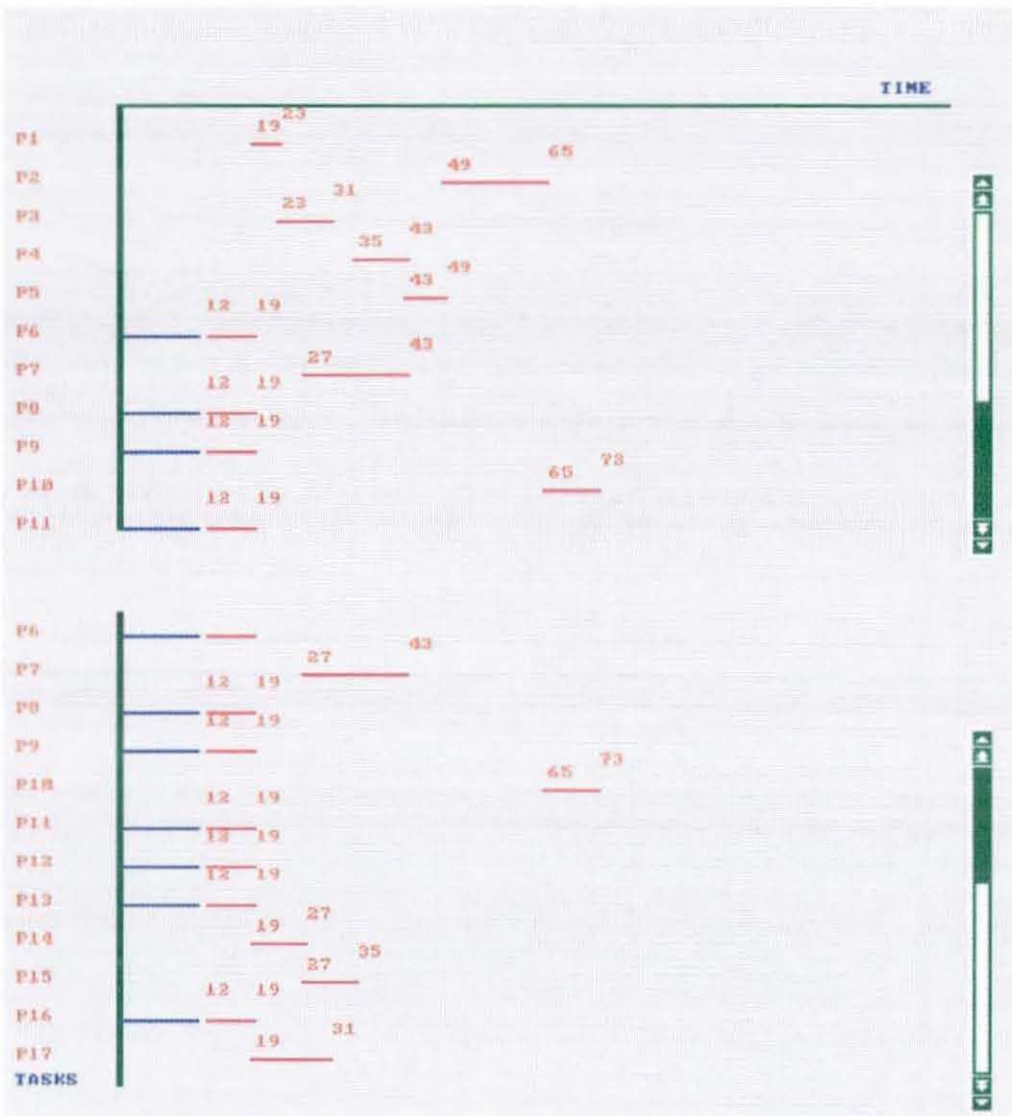


Figure 8.23 The impact of insufficient resources on the Detailed design schedule of the plant room

This feature is more useful in the later stages of design where there is more emphasis on the details of the information exchanged and hence quality of information. For example, drawings exchanged should show all dimensions, services engineers should provide precise information about dimensions of plant, maintenance requirements, allowance for vibrations, etc.

It should be noted that the simulation tool will not enhance the quality of information exchanged, but it will provide indications and awareness for the design managers to assess the implications of 'poor' quality information and allow for it in the design programme.

Example:

Considering the same example of the plant room described in section 8.8. The design manager envisages that the draftsman who will perform the task 'steel work plan drawings' will receive information of poor quality from the engineer performing the task 'mezzanine floor steel work calculations'. (For example, the steel sections are not dimensioned clearly.) In this case the design manager will refer to the simulation model and will allocate (according to their judgement) a global quality attribute for the information exchanged, say 80%. This means that the quality of information exchanged should be equal to or greater than 80% in order that the task receiving the information would be performed. The design manager will allocate to the information link representing the dependency of the task ' steel work plan drawings' on the task 'mezzanine floor steel work calculations' a quality attribute representing the envisaged poor information quality, say 40%. The remaining tasks that are expected to receive information of satisfactory quality are allocated quality attributes on their information links ranging between 80% and 100%.

Running the simulation will present the design manager with the impact of the poor quality information provided from the engineer to the draftsman. Figure 8.24 illustrates the state of each task after running the simulation. Seven tasks are affected and cannot commence due to such poor information quality. These tasks are:

- Ground floor drawings
- Foundation drawings
- Column casing drawings
- Duct route drawings
- Mezzanine floor slab drawings

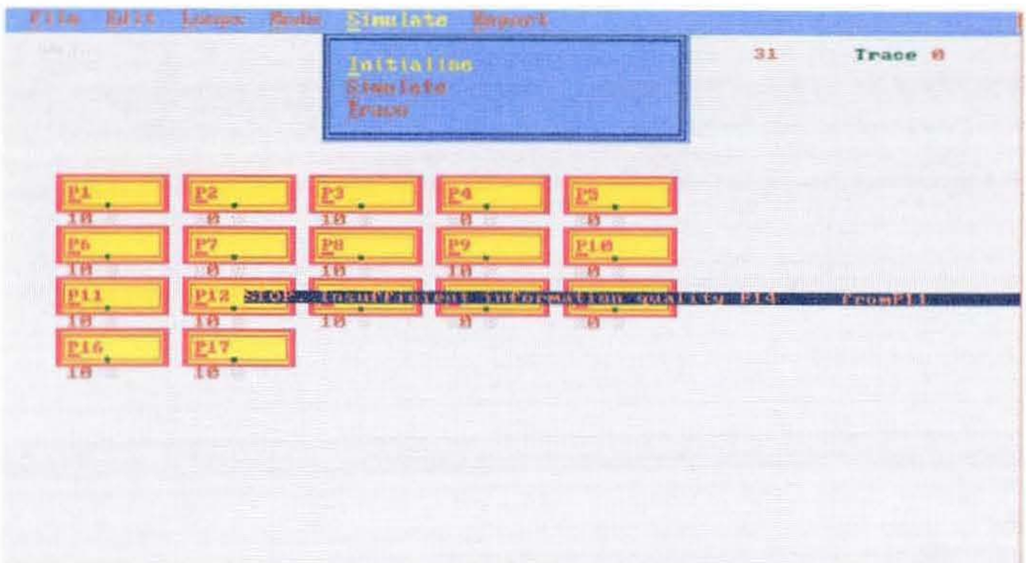


Figure 8.24 The impact of 'poor' quality information

- Steelwork plan drawings
- Steelwork section drawings

A message will also appear on the screen indicating that running the simulation cannot proceed further due to the poor information quality from the task 'mezzanine floor steel work calculations' to the task ' steel work plan drawings'. The design manager can then use the simulation tool to demonstrate to the engineer the impact of the poor quality information provided to the draftsman.

8.11 INDUSTRY FEEDBACK ON THE DEVELOPED TOOLS

The Simulation Model was validated through demonstrations held by the writer to industry professionals within Ove Arup and Partners, Nottingham branch; Ove Arup and Partners, Birmingham branch and AMEC Design and Management. The objective of the demonstrations was to acquire feedback on the contribution that the application of the simulation model will offer to improve the management of the design process across the different stages of design. A total of 10 demonstrations have been undertaken and a feedback document was issued to each attendee at the end of every demonstration. A copy of the feedback document is included in Appendix X. The responses showed the importance of the problems experienced during managing the design process and which are reflected in the features of the developed simulation model. They showed also the suitability of the developed tools to provide the solution to these problems and that the application of these tools will help to improve design management. A real mechanical design problem for a panel was presented by an organisation director during one of the demonstrations with regard to unavailability of information related to loads on the panel at the time of carrying out the design. The writer showed how the simulation model may be used to assess the impact of assuming the loads information at an early stage. One of the project managers highlighted the potential benefits that the simulation model provides for analysing historical projects especially in assessing the efficiency of the resource utilisation. Table 8.3 summarises the acquired responses. The values in the cells represent the average values of the acquired responses.

Problem	Importance (1-10) Conc/ Schem. des	Importance (1-10) Detailed des	Suitability of tools to provide the solution (1-100)
Assessing the impact of missing Information	7	8	76
Assessing the impact of assuming information	6	7	77
Assessing the impact of phased release of information	6	8	75
Assessing the impact of different levels of information quality	4	7	60
Assessing the impact of gate keeping of information	7	7	77
Assessing the impact of uncertainties and carrying out risk analysis	7	7	73
The problem of iteration	8	8	77
Resources management	8	9	75

Table 8.3 Summary of responses to the feedback document

The analysis of the responses showed the following:

(i) The problems in design management which were identified by the writer are valid with variable significance at the Conceptual/Schematic design stages and the Detailed design stage.

(ii) The maximum variation in this significance is for the problem of information quality. The importance of this problem was scored 4 (on a scale of 1-10) at the Conceptual/Schematic design while it was scored 7 at the detailed design. This confirms one of the conclusions of the case study presented in chapter 7 which showed that the main concern for the designers at the early stages of design is primarily to acquire the design information. It is at the later stages when refining the design that benefits are achieved through assessing design information quality.

(iii) The scores for the importance of the identified problems are >= 6 for the Conceptual/Schematic design stages (except for the problem of information quality) and >= 7 for the detailed design. This further proves that the identified problems are of prime importance to design management.

(iv) The range of the scores for the importance of the identified problems at the Conceptual/Schematic design stages is from 6 to 8 (with the exception of the problem of information quality) while the range of the scores at the detailed design stage is from 7 to 9. This indicates that the importance of the problems is more tangible as the design progresses.

(v) The score of the suitability of the tools to provide the solution to the identified problems ranged from 73 to 77 on a scale of 1-100 (with the exception of the problem of information quality). This confirms the benefits that the developed tools offer to improve the management of the design process. The least score was for the problem of the information quality which scored 60. This is due to the subjectiveness involved in simulating information quality.

The following modifications were suggested within the feedback document:

(i) To include the description of the design tasks in the input tables of the simulation model in lieu of the tasks' numbers in order to be more 'user friendly'.
This can be easily incorporated by adding a text column to the input tables describing each design task.

(ii) To associate resource utilisation with the bar chart of the design schedule to acquire all the necessary information in one screen.
The existing DOS version of the Genetik environment renders it difficult to incorporate such modification. However, Genetik supplier indicated their intention to produce a Windows based version of the simulation environment and hence it would be possible to combine several windows to show different formats for the results simultaneously.

(iii) To incorporate a facility of imposing a completion time to the design project or to specific design tasks and run the model 'backwards' to produce a design schedule and to calculate the required resources.
The concept of discrete event simulation is based on advancing the simulation clock when certain events are triggered. Therefore the simulation clock cannot move backwards. Additionally, the C-Events of the developed simulation model

are based on triggering design tasks when the specified conditions are being fulfilled (such as availability of the necessary information from preceding tasks, availability of resources, etc.) and hence it is not possible to start the simulation with a 'finish' time of a task because this task must receive information from the preceding task(s) and provide information to the succeeding task(s). The following alternative solutions are suggested:

- to run the simulation in the stochastic mode several times until the required completion dates are achieved and plan the design process based on the sampled durations
- to attempt different scenarios for the durations and resources on a 'what if' basis until the optimum scenario is achieved and plan the design based on this scenario
- to run the simulation in the stochastic mode several times and calculate the probability of completing the design project (or particular design tasks) in the required time.

All the attendees for the demonstrations confirmed that the application of the methodology used to produce the developed tools is not only valid for the Conceptual/Schematic stages of design but also for the Detailed design stage. A data flow model for the detailed design stage has been produced within the AMEC organisation and hence the use of data flow diagrams to model this stage has proved to be feasible. This is endorsed by Newton (1995) who produced data flow diagrams for the detailed design stage. The summary of the results in the feedback document (Table 8.3) shows that the typical events which occur during the design stage and which are reflected within the features of the developed simulation model are valid, with variable significance, across the different stages of design. However, minor adjustments would be required to apply the simulation model at the Detailed design stage. These adjustments are summarised as follows:

(i) A data flow model for the detailed design stage may include as much as 5000 functional primitive tasks. It would not be practical to assign durations and resources for such number of tasks. Therefore, the design tasks which would be included in the simulation model should represent the tasks of the data flow model at a level which is higher by one or two levels than the functional primitive tasks.

(ii) A separate 'cut off' value for the quality attribute of each information link should be incorporated in lieu of a global value for the whole design stage under consideration. This is because at the detailed design stage the design tasks are more defined and hence the required level of information quality may vary considerably. This feature may also represent a classification for the sensitivity and/or importance of information. The information links of higher 'cut off' values are more sensitive (and/or important) than the information links with lower 'cut off' values. This feature may be incorporated in the model easily by adding in the information links table a column which includes the 'cut off' value for each information link. The quality attribute for each information link would be checked against its corresponding 'cut off' quality value in lieu of a global value.

(iii) To incorporate in the model a facility which allows the assessment of the impact of different scenarios of information related criteria on the design fee. This would be of particular benefit in demonstrating to the Client the impact of missing information or late changes in the design information from the Client's side to support claims for variation orders. This feature may be incorporated by associating with every design task the cost/hour for carrying out the task. Therefore, the costs incurred due to any delays resulting from late information or re-design due to changes in the design information may be calculated.

PART IV

CONCLUSIONS AND RECOMMENDATIONS

CHAPTER 9

CONCLUSIONS AND RECOMMENDATIONS

9.1 INTRODUCTION

This research has focused on the improvement of the management of the building design process. The aim of this research was to study, model and simulate the information flow during the building design process to allow analysis of the effects of typical events and hence improve the management of the whole process. To meet this aim, the following research objectives were formulated:

(i) To study the nature of the design process in general and the building design process in particular.

(ii) To examine current practice for planning and managing the building design process.

(iii) To identify the main problems in design management.

(iv) To investigate existing models for the design process

(v) To model the information flow between the different participants within the building design process.

(vi) To identify typical events and information related problems.

(vii) To develop a computer based simulation tool to predict the effects of the identified events and problems and produce design schedules based on these predictions.

(viii) To assess the benefits that the developed tools offer to improve the management of the design process.

The main conclusions derived from this research are described in the following sections. From these conclusions the hypothesis of the research is proven i.e. existing planning techniques are unsuitable for the management of the design process. Techniques based on a combination of Data Flow Diagrams, Matrix Analysis and

Discrete Event Simulation will improve the management of the Conceptual, Schematic and Detailed design phases.

9.2 MAIN CONCLUSIONS

This section describes the main conclusions drawn from this research under the following headings:

- the nature of design;
- the current practice for planning and managing the building design process;
- the main problems in design management;
- the importance of information transfer for successful design management;
- the existing models of the design process;
- the use of Data Flow Diagrams to model the design process;
- typical events and information related problems during the building design process;
- the use of Discrete Event Simulation to simulate the flow of design information;
- evaluation of the developed tools; and
- the benefits that the developed tools offer to improve the management of the design process.

9.2.1 The Nature of Design

The literature review has shown that design is, by its nature, an iterative process. This iterative nature makes it complex and difficult to manage.

The nature of building design is not fundamentally different from the nature of manufactured product design. This has been recognised with the recent move of researchers towards considering construction as a manufacturing process. Most researchers regard the design process, although unique in itself, is not being affected by the product or process. However, a limited number of researchers propose that design processes are dependent on their product or processes. Defining design projects as a flow of information through time will achieve a compromise between these two schools of thought.

There is no consensus among researchers and practitioners with respect to the different stages of the design process. However, the RIBA plan of work represents the most well recognised model for the different stages of a construction project including the design stage.

9.2.2 The Current Practice for Planning and Managing the Building Design Process

Current planning techniques such as network analysis and PERT are suitable for planning deterministic activities which are either sequential or parallel. They are ill-suited to plan activities with an iterative nature, such as design activities, because they neither allow feedback loops nor any iterative procedures.

A review of the current practice for design management undertaken by the writer showed that in complex multi-disciplinary design situations, design managers lack sophisticated tools to aid them in managing the process. Such tools are required to aid design managers in planning design, taking into consideration its iterative nature, and foreseeing the effects of changing different parameters that affect information transfer and communications during the design process.

To date, few attempts have been made to apply concurrent engineering techniques used in the manufacturing industry to the construction industry. One potential technique considered appropriate to construction is the use of matrix analysis to achieve the optimum order for design tasks and highlight which tasks should be carried out in an iterative fashion. Although this technique has been recently applied by construction industry researchers as a management tool, there are other concurrent engineering techniques such as Quality Function Deployment and DFX (Design For Manufacture; Design For Assembly; Design For Inspection; etc.) that offer potential benefits to the construction industry but these have not been fully exploited.

9.2.3 The Main Problems in Design Management

The results of the extensive literature search, survey and interviews undertaken by the writer have shown that the problems in design management may be categorised into five categories: problems due to the inherent nature of design; problems due to technical aspects of design; client related problems; problems due to difficulties in managing information and problems due to difficulties in planning design.

Of these categories, the last two have been shown to be of great significance in the successful management of the design process. Therefore this research has concentrated on these categories of problems. Although the significance of these problems vary across the Conceptual/Schematic design stages on one hand and the

Detailed design stage on the other hand, the problems are valid for all of the three stages.

9.2.4 The Importance of Information Transfer for Successful Design Management

The design process is information driven. The main difficulties encountered during the management of the design process are predominantly information related. Information transfer and communication issues have been identified by different researchers as the key factors to the successful management of the design process. The review of previous research showed that little work had focused on managing information exchange during the early stages of design prior to the production of contract documents. In particular there was little research with regard to standardisation of ways of information exchange. This was confirmed through the survey and industry interviews undertaken by the writer which showed that the management of the Conceptual and Schematic stages of design is more difficult than the Detailed design stage. For this reason, this research focused on the Conceptual and Schematic stages of design.

Although some researchers have attempted to establish measures for information quality, there is no consensus over such measures. The literature review showed that measuring information quality is subjective, situation dependent and varies over time. This has been confirmed through the industry interviews undertaken by the writer which showed that there is no formal way to judge the quality of design information. One measure of quality of information is the satisfaction of the recipients. The measure of good quality is if the information provided is sufficient for the recipient to proceed to the next stage in the design. Design information is considered to be of poor quality if the information is insufficient or unsatisfactory for the recipient to proceed.

9.2.5 The Existing Models of the Design Process

Early models for the design process were either descriptive or prescriptive showing the different stages of design and emphasising its iterative nature. Some models addressed the different ways of thinking and learning styles of designers and the factors that influence them. However, a comprehensive review by the writer of the different models concluded that none of these models addressed in detail the information transfer and communication issues which have been identified by different researchers as the key factors to the successful management of the design process. It was not until the late 1980's when structured diagramming techniques developed for systems analysis purposes were consequently used by researchers to model the design process and to show the information exchange within the process.

9.2.6 The Use of Data Flow Diagrams (DFDs) to Model the Design Process

The literature review of the design process described in Chapters 2 and 3 emphasised the importance of communications and information transfer issues as key factors for successful design management. For this reason the research focused on the application of structured analysis diagramming techniques to the design process where information flows between processes could be modelled. An examination of the different categories of structured analysis diagramming techniques described in chapter 3 concluded that Data Flow Diagrams were the most suitable technique for modelling information transfer during the design process. The design tasks represented in the data flow model were analysed using partitioning techniques of the Design Structure Matrix to identify the inter-dependent tasks and loops of iterative design tasks

The following conclusions are drawn from the Generic Data Flow Model developed by the writer for the Conceptual and Schematic stages of design:

(i) Data Flow Diagrams are a useful technique for information transfer representation and can be easily understood by both researchers and industry representatives. The majority of the industry professionals who were involved in commenting on the DFDs had no prior instruction in the use of the technique. Within a short period of time however, they were conversant with the principles to the extent of being able to comment in a meaningful way

(ii) Data flow models provide a useful effective tool which may be used to improve communications during the design process, and hence improve the management of the process. These models assist in identifying information requirements for different design tasks, and in identifying other designers' problems. They may be used in the training of engineers and architects.

(iii) The model for the Conceptual and Schematic design stages is independent of the procurement strategy.

(iv) For the certain design tasks identified by the interviewees during the validation of the model, the difficulty in obtaining the information requirements for these tasks varied. However, the importance of the information did not vary.

(v) The difficulties with external information sources where approvals and regulations were necessary, (e.g. different authorities, insurers) were due to the fact that these sources are involved after a substantial part of the design has been already completed, and hence any input may require re-design and other implications on other design tasks. Also there was frequently a difficulty in interpretation of the regulations and the time taken by these sources to take decisions or provide approvals.

(vi) Difficulty in obtaining information and the importance of some information were seen from different perspectives according to the background and discipline of the design manager. A piece of information may be considered as important by one designer or manager but not by another.

(vii) Difficulties in communications or acquiring information are more frequent when dealing with external sources. Information required from sources or disciplines within the same organisation is easier to obtain than obtaining information from external sources as it is more difficult to control external sources.

(viii) There is no formal way to judge the quality of information exchanged. The measure of information quality varies according to the sender and the recipient of information. Missing information or information of insufficient quality from the recipients point of view are supplemented by assumptions.

(ix) The Generic Data Flow Model for the Conceptual/Schematic design stages may be subject to minor adjustments to fit a particular project and/or organisation.

(x) A few tasks and information requirements were regarded by some interviewees during the validation of the model as part of later design stages. This is partly due to the natural overlap between the different stages of design and partly due to the lack of consensus among researchers and professionals in the construction industry about the tasks that comprise every design stage. However, to maintain the generic nature of the model, it was decided to retain these elements to be used at the discretion of every user organisation.

9.2.7 Typical Events and Information Related Problems During the Building Design Process

The typical events and information related problems that occur during the design process were identified through a questionnaire survey followed by subsequent interviews with professionals in the construction industry. These events include:

(i) The variation of the quality of information exchanged between different design tasks.

(ii) Performing a design task based on assumed data inputs.

(iii) Changes in design information

(iv) The problem of missing information

(v) Releasing the information from different design tasks in packages.

(vi) 'Gate keeping' of information among design team members.

(vii) Resource allocation and assessment of their utilisation throughout the whole design process.

These events represent the main features of the developed simulation model.

9.2.8 The Use of Discrete Event Simulation to Simulate the Flow of Design Information

After thorough investigation of different simulation techniques to simulate the design process and the associated information related events, the writer concluded that the use of discrete event simulation technique using a simulation modelling environment which supports the three phase approach provides the most appropriate simulation technique for this research. Discrete event simulation allows the user to instantaneously observe the changes that occur in the model as the simulation clock advances and allows interaction with the model. A simulation modelling environment provides flexibility in the modelling aspects, including incorporating all necessary rules, which are not found in simulation data driven packages. The types of events involved in the three phase approach have been found suitable to represent events that occur during the design process.

The features of the Discrete Event Simulation Model developed reflect the typical events and information related problems described in section 9.2.7. The simulation model has transformed the developed Generic Data Flow Model from its static state to a dynamic state through allocating durations and resources to the design tasks. This has allowed the study of the impact of:

(i) Starting a design task at an earlier time based on assumed information.

(ii) 'Gate keeping' or withholding design information among design team members.

(iii) Changes in design information

(iv) Missing information

(v) The variation of the quality of information exchanged between different design tasks.

(vi) Releasing the information from different design tasks in packages or phases.

(vii) Allocating different resources to each design task and assessing their utilisation throughout the whole design process.

9.2.9 Evaluation of the Developed Tools

In order to evaluate the benefits and the validity of the Data Flow Model and the Design Process Simulation Model, a case study was conducted on the new engineering complex project at Loughborough University. The main conclusions of the study are:

(i) The Generic Data Flow Model for the Conceptual and Schematic design stages may be applied to any project with minor adjustments to fit the specific nature of each project. The value of the model for both stages is primarily that of a checklist to aid design management in identifying design tasks and their relevant information requirements.

(ii) The information exchanged during the design meetings at the Conceptual design stage falls into five categories:

(a) Information related to spatial and functional requirements.

(b) Information related to strategic requirements.

(c) Information related to operational requirements.

(d) Information related to the project aesthetics

(e) Cost information

This information was allocated to at least one of the information flows in the Generic Data Flow Model.

(iii) There was no impact of the Client's involvement in this case study on the Generic Data Flow Model other than that of a 'source' of the information represented in the model. There were no actual tasks undertaken by the Client apart from 'seeking information'.

(iv) There are no substantial benefits from running the simulation model to produce design schedules only for the Conceptual design stage. This is due to the following reasons:

(a) At the Conceptual design stage, architects usually 'think' of more than one task simultaneously. This aspect is practically impossible to simulate and allocating durations to these tasks would be unrealistic.

(b) At the Conceptual design stage, a substantial amount of time is spent 'thinking' around the design problem. This may vary according to the designer's experience and background and according to the project's particular circumstances.

(c) There is a substantial amount of 'waiting' time from the Client to make decisions or collect certain information. This may vary depending on the type of Client an/or type of project and hence, simulation would give unrealistic results.

(d) The resources involved at this stage are minimal and hence there are no substantial benefits from simulating resource engagement or utilisation.

(v) Since the information exchanged at the Conceptual design stage is used to <u>start</u> the design, there is no need to assess the information quality. This is because the main concern for the designers is primarily to acquire the design information. It is at the later stages when refining the design that benefits are achieved through assessing design information quality.

9.2.10 The Benefits that the Developed Tools Offer to Improve the Management of the Design Process

The benefits that the developed tools offer to improve the management of the design process across its different stages were demonstrated through practical examples. These benefits are summarised as follows:

(i) The Data Flow Model will be used by the design manager throughout the different design stages as a monitoring tool to ensure the completeness of the information requirements for the different design tasks.

(ii) The data flow diagrams will also assist the design manager to plan for the production of information and control design by ensuring that the required design information is included.

(iii) The matrix modelling for the functional primitive tasks of the data flow diagrams using the Design Structure Matrix will assist the design manager to identify loops of iterative design tasks. Knowing these tasks will assist the design manager in the interface management of different design disciplines. Designers performing tasks which fall within the same iterative loop should work in proximity to facilitate the communication process and increase the design efficiency.

(iv) The Discrete Event Simulation Model will provide the design manager with design schedules and resources histograms for different scenarios on 'what if' basis.

(v) The Simulation Model will also present to the design manager the different states for every design task as the simulation clock advances.

(vi) The Simulation Model will allow the design manager to assess the impact of:
- Changes in design information
- Missing design information
- Assuming design information
- 'Gate keeping' of design information

- The variation in the quality of information exchanged during the design process
- Phased release of information

(vii) The simulation model will assist the design manager in managing the resources by simulating the different states (idle v busy) for each resource as the simulation clock advances and by producing resources histograms for the utilisation of each type of resource.

Feedback from the industry for the developed tools showed the following:

(i) The problems in design management which were identified by the writer are valid with variable significance at the Conceptual/Schematic design stages and the Detailed design stage.

(ii) The maximum variation in this significance is for the problem of information quality. This confirms one of the conclusions of the case study presented in chapter 7 which showed that the main concern for the designers at the early stages of design is primarily to acquire the design information. It is at the later stages when refining the design that benefits are achieved through assessing the quality of design information.

(iii) The scores for the importance of the identified problems showed that the identified problems are of prime importance to design management.

(iv) The range of the scores for the importance of the identified problems at the Conceptual/Schematic design stages and at the Detailed design stage indicates that the importance of the problems is more tangible as the design progresses.

(v) The score of the suitability of the tools to provide the solution to the identified problems confirms the benefits that the developed tools offer to improve the management of the design process.

(vi) The methodology adopted in this research to improve the management of the design process is valid for all stages of design with minor adjustments for the Detailed design stage.

9.3 RECOMMENDATIONS FOR FUTURE WORK

The following recommendations for further research are derived from this study:

(i) This research has proved that techniques based on Data Flow Diagrams, Matrix Analysis and Discrete Event Simulation will improve the management of the design process. Further research should be undertaken to implement these techniques in design organisations and to assess the viability of such implementation.

(ii) The exchange of data between the Data Flow Model CASE tool, the Design Structure Matrix software, and the Genetik simulation environment has been undertaken manually. Further research should be undertaken to integrate the three software via a central database which can import and export data from the data dictionary of the CASE tool. A proposed prototype is illustrated in Figure 9.1. Further research should also be undertaken to link the simulation model with current project management software.

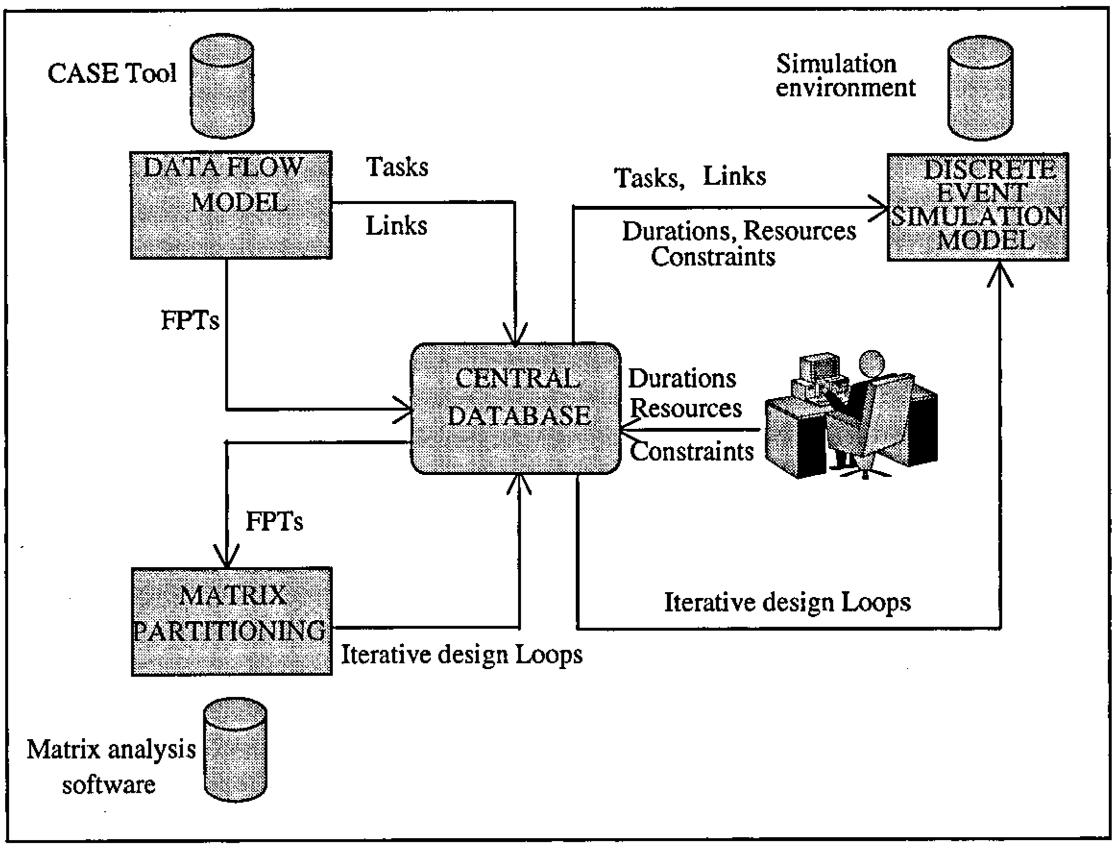


Figure 9.1 A proposed prototype for integration of software

(iii) The Design Structure Matrix has been applied in this research to identify loops of iterative design tasks. It is recommended that further research should be undertaken to investigate other methods of modelling building design iteration mathematically such as using signal flow graphs as described by Nukala et al (1995).

(iv) Further research should be undertaken to investigate the probability distributions which design tasks durations follow.

(v) The importance of standard forms of information exchange during the design process was highlighted within this research. Further research should be undertaken to investigate this area especially during the design stages prior to the production of contract documents.

(vi) This research has showed that the application of concurrent engineering concepts such as the Design Structure Matrix offer potential benefits to the construction industry. Further research should be undertaken to apply other concurrent engineering techniques such as the Quality Function Deployment technique, a method of designing and optimising the process of developing new products based on customer needs. This technique may be applied in incorporating the elements of the Client's brief of a construction project in the design of this project.

(vii) Although typical information related events and problems were incorporated in the developed simulation model, different communication routes reflecting different organisational structures were not considered. The writer recognises the importance of this aspect and recommends that future research should be undertaken to incorporate the organisational structure in the simulation model. This recommendation is endorsed by Jin et al (1995) who, in recent research, developed a computer simulation model for studying organisational aspects of concurrent design.

REFERENCES

Academic Press Dictionary of Science and Technology, 1991, Harcourt Brace Jovanovich Publishers, London

Addis, W, 1990, *Structural Engineering*, Ellis Horwood Limited, Chichester, UK

Adeli, H., 1988, *Expert Systems in Construction and Structural Engineering*, Chapman and Hall Ltd., New York

Ahituv, N. & Neumann, S., 1988, *Principles of Information Systems for Management*, 2nd edition, Dubuque, Iowa, W C Brown

Ahuja, H. N., 1984, *Project Management Techniques in Planning and Controlling Construction Projects*, John Wiley and Sons, Chichester

Ahuja, H and Nandakumar, V, 1985, *Simulation Model to Forecast Project Completion Time*, Journal of Construction Engineering and Management, ASCE Vol 111, pp 325-342.

Allwood, R. J., 1989, *Techniques and Applications of Expert Systems in the Construction Industry*, Ellis Horwood Limited, Chichester

Austin, S, Baldwin, A & Newton, A, 1993, *Modelling Design Information in a Design and Build Environment,* ARCOM conference, September 1993, Oxford, UK, pp 73-84.

Austin, S. A., Baldwin, A. N. & Newton, A. J., 1994, *Improved Building Design Programming by Manipulating the Flow of Design Information*, Journal of Construction Management and Economics, Vol. 12, pp 445-455

Austin, S., Baldwin, A., & Newton, A., 1994, *Manipulating Data Flow Models of the Building Design Process to provide effective Design Programmes,* ARCOM Conference, Loughborough, UK, September, pp 592-601

Austin, S. A., Baldwin, A. N. & Newton, A. J., 1995, *A Data Flow Model to Plan and Manage the Building Design Process,* Journal of Engineering Design, 7 (1), in press.

Austin, S. A., Baldwin, A. N., Thorpe, A., & Hassan, T. M., 1995, *Simulating the Construction Design Process by Discrete Event Simulation*, International Conference on Engineering Design, ICED 95, Prague, August, pp 762-772.

Baldwin, A., Austin, S., Thorpe, A. & Hassan, T., 1995, *Simulating* Quality *Within the Design Process*, American Society of Civil Engineers (ASCE) Second Congress on Computing in Civil Engineering, Atlanta, Georgia, June, pp1475-1482

Baldwin, A., Austin, S., Thorpe, A. & Hassan, T., 1994, *Simulating the Impact of Design Changes Upon Construction,* Proceedings of the 10th ARCOM Annual Conference, Loughborough, UK, pp 213-221.

Barrie, D. S. & Paulson, B. C., 1992, *Professional Construction Management,* McGraw Hill, London

Beakley, G. C. & Chilton, E. G., 1974, *Design Serving the Needs of Man*, Macmillan, New York, London

Belson, D., 1994, *Concurrent Engineering,* In handbook of design, manufacturing and automation, ed., Dorf, C. & Kusiak, A., John Wiley & Sons, Inc., New York

Bennett, J., Flanagan, R., Lansley, P., Gray, C. & Atkin, B., 1988, *Building Britain 2001*, Centre for Strategic Studies in Construction, Reading

BS1192, Part 1, 1984, *Construction Drawing Practice; Recommendation for General Principles*, BSI, London

BS1192, Part 5, 1990, *Construction Drawing Practice; Guide for Structuring Computer Graphic Information*, BSI, London

BS7000, Part 4, Draft, 1994, *Guide to Managing Design in Construction*, BSI, London

Builder, 1993, *Design Council,* An article in Builder, November 1993, under a section for Design Management, pp 20-41.

Byte, 1989, *The CASE Experience*, Byte, April 1989, pp 235-244.

Carr, R. I., 1979, *Simulation of Construction Project Duration*, Journal of the Construction Division, June 1979, pp 117-128.

Coleman, C. R., 1992, *Data Transfer - Designs on the Move*, In Architectural Management, ed. Nicholson, M., E & FN Spon, London

Coles, E. J., 1990, *Design Management: A Study of Practice in the Building Industry*, CIOB occasional paper No 40

Colquhoun, G., Baines, R. & Crossley, R., *A State-of-the-art Review of IDEF0*, International Journal of Computer Integrated Manufacturing, Vol. 6, No. 4, pp 252-264.

Cornick, T., 1991, *Quality Management for Building Design*, Butterworth - Heinemann Ltd., London

CPI, 1987, *Co-ordinated Project Information for Building Works*, a Guide with Examples, Co-ordinating Committee for Project Information, London

Cross, N., 1984, *Developments in Design Methodology*, John Wiley & Sons, Chichester

Cross, N., 1991, *Engineering Design Methods,* John Wiley & Sons, Chichester

Culley, S., McMahon, C. & Court, A., 1992, *The Information Requirements of Engineering Designers*, Design Research Study, May/June, pp 21-23.

Dawood, N., 1991, *A Computer-based Capacity Planning System for Precast Concrete Production,* PhD thesis, Loughborough University of Technology, Loughborough, UK

Day, A., & Faulkner, A., 1988, *Multi-Disciplinary Use of CAD in the Building Industry*, CICA, Cambridge, UK

Demarco, T., 1978, *Structured Analysis and System Specification*, Yourdon Press (Prentice Hall), Englewood Cliffs, NJ

Dias, W. P. S., 1993, *Product Modelling of Buildings,* Transactions of Institute of Engineering in Sri Lanka, pp 134-151

Dias, W. P. S. and Blockley, D. I., 1994, *The Integration of Product and Process Models for Design*, Design Studies, Vol. 15 No. 4, pp 417-432

Dieter, G. E., 1983, *Engineering Design. A Materials and Processing Approach,* McGraw Hill, New York

Edel, D. H., 1967, *Introduction to Creative Design*, Prentice Hall

Eppinger, S., Whitney, D., Smith, R. & Gebala, D., 1990, *Organizing the Tasks in Complex Design Projects,* ASME Conference on Design Theory and Methodology, September 1990, New York, pp 39-46.

Eppinger, S., Whitney, D., Smith, R. & Gebala, D., 1993, *A Model Based Approach for Organising tasks in Product Development*, working paper, MIT Sloan School of Management, WP 356-93

Eppinger, S., Whitney,D. & Gebala, D., 1992, *Organising the Tasks in Complex Design Projects: Development of Tools to Represent Design Procedures*, Proceedings of the 1992 NSF Design and Manufacturing Systems Conference, Georgia Institute of Technology, Atlanta, Georgia, pp 301-309

Eppinger, S., 1991, *Model Based Approaches to Managing Concurrent Engineering,* Journal of Engineering Design, Vol 2, No 4, pp 283-290.

ESD: The Electronic System Design Magazine, 1989, *Solving the CASE of Embedded Systems*, June 1989, pp 20-22.

Evans, R., 1978, *Hallmark of Real Change,* Building, January 1978, pp 69-70.

Evans, B., Powell, J., & Talbot, R., 1982, *Changing Design*, John Wiley and Sons, Chichester

Fisher, N., 1990, *The Use of Structured Data Analysis as a Construction Management Research Tool: The Technique,* Construction Management and Economics, No 8, pp 341-363.

Fisher, N., & Lin, S., 1992, *Information Management in a Contractor - A Model of the Flow of Project Data,* Thomas Telford, London

Fisher, N., 1993, *Construction as a Manufacturing Process?*, Inaugural Lecture, University of Reading

French, M. J., 1985, *Conceptual Design for Engineers*, Design Council, London

Frost, R., 1992, *A Converging Model of the Design Process Analysis and Creativity, the Ingredients of Synthesis*, Journal of Engineering Design, Vol 3, No 2, 1992, pp 117-126.

Gebala, D. & Eppinger, S., 1991, *Methods for Analysing Design Procedures*, ASME Conference on Design Theory and Methodology, Miami, September, pp 227-233

Gharib, K. M., 1991, *Modelling Information Flow Within a Design and Build Organization,* MSc Project. Loughborough University of Technology, Loughborough, UK

Glavan, J. & Tucker, R., 1991, *Forecasting Design - Related Problems, A Case Study*, Journal of Construction Engineering and Management, Vol. III, No. 1, pp 47 - 65

Gray, C., Hughes, W. & Bennett, J., 1994, *The successful Management of Design*, The University of Reading

Green, S. D., 1992, *A SMART Methodology for Value Management*, Occasional Paper No. 53, CIOB

Griffin, C. R. & Evans, F. J., 1994, *STEP Technology for Building Services Projects*, Building Services Engineering Research and Technology, Vol. 15, No. 4, pp 205 - 210

Guevara, J. M. & Boyer, L. T., 1981, *Communication Problems with Construction*, Journal of Construction Division. Proceedings of ASCE, Vol 107, No CO4, December 1981, pp 551-557.

Gupta, V. & Murthy, P., 1980, *An Introduction to Engineering Design Method*, Tata McGraw Hill, New Delhi.

Halpin, D. W., 1977, *CYCLONE: Method for Modelling of Job Site Processes*, Journal of the Construction Division, ASCE, Vol. 103, No. C03, Proc. Paper 13234, Sept. 1977, pp 489 - 499

Halpin, D., Riggs, L., 1992, *Planning and Analysis of Construction Operations*, John Wiley & Sons, New York.

Hanby, V., Hedges I. and Murray, M., 1993, *Logic Modelling of the Building Services Design Process*, CIBSE Conference, Manchester, UK, pp 28-39

Harmon, P., Maus, R. & Morrissey, W., 1988, *Expert Systems Tools and Applications*, John Wiley and Sons Inc., New York

Harris, F. and McCaffer, R., 1989, *Modern Construction Management*, Third Edition. BSP Professional Books, Oxford, UK

Hawkes, B. and Abinett, R., 1984, *The Engineering Design Process*, Pitman, London

Hedges, I., Hanby, V., & Murray, M., 1993, *A Radical Approach to Design Management*, CLIMA 2000 Conference, London, November, pp 295-314

Hegedus, P., 1990, *Information Production and Use: Issues and Questions in the Meeting of Needs*, In Information Quality, ed. Wormell, I., Taylor Graham, UK

Hunter, J., 1993, Communications in Construction, MSc Project, Loughborough University of Technology, Loughborough, UK

Huovilla, P., Koskela, L. & Lautanala, M., 1994, *Fast or Concurrent, The Art of Getting Construction Improved*, Second International Workshop on Lean Construction, Santiago, Chile

Huovilla, P., Koskela, L., Lautanala, M., Pietilainen, K.& Tanhuanpaa, V., 1995, *Use of the Design Structure Matrix in Construction*, 3rd International Workshop on Lean Construction, Albuquerque

Jergeas, G. F., 1989, *Detailed Design and Constructability*, PhD Thesis, Loughborough University of Technology, Loughborough, UK

Jergeas, G. F.& McCaffer, R., 1990, *Constructability and Detailed Design*, CIOB

Jewell, T., 1986, *A Systems Approach to Civil Engineering Planning and Design*, Harper & Row, London

Jin, Y., Levitt, R. E., & Kunz, J. C., 1995, *The Virtual Design Team: A Computer Simulation Framework for Studying Organizational Aspects of Concurrent Design*, Simulation, Vol. 64, March, pp. 160-174

Jin, Y., Levitt, R. E., Christiansen, T. R., & Kunz, J. C., 1995, *The Virtual Design Team; Modelling Organisational Behavior of Concurrent Design Teams*, Artificial Intelligence for Engineering Design, Analysis and Manufacturing: AI EDAM, Vol. 9, No. 2

Jones, J. C., 1981, *Design Methods: Seeds of Human Factors*, John Wiley and Sons, New York

Kalk, A., 1978, *INSIGHT - Interactive Simulation of Construction Operation Using Graphical Techniques*, Thesis presented to Stanford University, California, 1978, in partial fulfilment for the Degree of Engineer.

Kolb, D. A., 1976, *The Learning Style Inventory Technical Manual*, MacBer, Boston, USA, 1976.

Krishnan, N., Eppinger, S., & Whitney, D., 1993, *Overlapping Product Development Activities by Analysis of Information Transfer Practice*, Ninth International Conference on Engineering Design, ICED 93, The Hague, September, pp 685-688

Kusiak, A. & Larson, N., 1995, *Decomposition and Representation Methods in Mechanical Design*, Transactions of the ASME, Vol. 117, pp 17 - 24

Kusiak, A., 1994, *Concurrent Engineering: Issues, Models, and Solution Approaches*. In handbook of design, manufacturing and automation, ed., Dorf, C. & Kusiak, A., John Wiley & Sons, Inc., New York

Latham, Sir M., 1994, *Constructing the Team*, HMSO

Laurikka, P. J., 1993, *The Suitability of Computer Simulation Programs for Construction Planning*, CIB conference, Trinidad, September 1993, pp 221-229.

Lawson, B., 1980, *How Designers Think*, The Architectural Press, London

Layer Naming Convention for CAD in the Construction Industry, Version 2, November 1991, AutoCAD User Group and Autodesk Ltd, Surrey

Maji, R. K., 1988, *Tools for Development of Information Systems in CIM*, Advanced Manufacturing Engineering, 1, pp 26-34

Mandel, K., 1990, *Graphical Process Descriptions - Views and Diagrams*, International Journal of Computer Integrated Manufacturing, 3, (5), pp 314-327

Manyanga, J., 1993, *An Investigation of the Decision-Making Processes of Professional Designers on Engineering Projects,* MSc Project, Loughborough University of Technology, Loughborough, UK

Marca, D. & McGowan, C., 1988, *SADT- Structure Analysis and Design Techniques*, McGraw-Hill, New York

Marchand, D., 1989, *Managing Information Quality*, in Information Quality, ed. Wormell, I., Taylor Graham, UK

Martin, J. and McClure, C., 1985, *Structured Techniques for Computing*, Prentice-Hall, NJ.

McCord, K. & Eppinger, S., 1993, *Managing the Integration Problem in Concurrent Engineering*, Working paper WP 3594-93-MSA, Sloan School of Management, Massachusetts Institute of Technology, USA

McGee, 1992, Internal Confidential Report for AMEC Design and Management.

Mize, J. H. & Cox, J. G., 1968, *Essentials of Simulation,* Prentice Hall

Moxley, R., 1993, *Building Management by Professionals*, Butterworth, Oxford, UK

Mujtaba, S. M., 1994, *Simulation Modelling of a Manufacturing Enterprise with Complex Material, Information and Control Flows*, Computer Integrated Manufacturing, Vol. 7, No. 1, pp 29-46

Naylor, T. H., Balintfy, J. L., Burdick, D. S. & Chu, K., 1966, *Computer Simulation Techniques*, Wiley, New York

Ndekugri, I. & McCaffer, R., 1988, *Management Infomation Flow in Construction Companies*, Construction Management and Economics, Vol. 6, pp 273 - 294

Neale, R. & Neale, D., 1989, *Construction Planning*, Thomas Telford, London

NEDC Report, 1987, *Achieving Quality on Building Sites*, Building and Economic Development Committee, NEDO, London

NEDC Report, 1990, *Information Transfer in Building,* A report published by National Economic Development Office (NEDO), London

Neville, G., 1988, *Computational Models of the Design Process,* Proceedings on Design and Theory Methodology. N S F Grantee Workshop. Troy. New York.

Newton, A. J., 1995, *The Planning and Management of Detailed Building Design*, PhD thesis, Loughborough University of Technology, Loughborough, UK

Nukala, M., Eppinger, S., Whitney, D., 1995, *Generalized Models of Design Iteration Using Signal Flow Graphs*, DE-Vol. 83, Design Engineering Technical Conferences, Volume 2, ASME, pp 413-422

O'Brien, J., 1972, *CPM in Construction Management*, Fourth Edition, McGraw Hill, London

Ove Arup, 1994, *Building Engineering AutoCAD,* CAD Good Practice Guide, CICA, Cambridge

The Oxford English Dictionary, 1989, second edition, Clarendon Press, Oxford

Pahl, G. and Beitz, W., 1988, *Engineering Design. A Systematic Approach*, The Design Council.

Paul, R. and Balmer, D., 1993, *Simulation Modelling*, Chartwell-Bratt

Paulson, B., Douglas, S., Kalk, A., Touran, A., & Victor, G., 1983, *Simulation and Analysis of Construction Operations*, Journal of Technical Topics in Civil Engineering, ASCE, Vol 109, pp 89-104.

Pepper, S. C., 1942, World Hypothesis, University of California Press, Berkeley, California.

Pidd, M., 1992, Computer Simulation in Management Science, Third Edition, John Wiley & Sons, Chichester, UK.

Pilcher, R. & Flood, I., 1984, *The Use of Simulation Models in Construction,* Proceedings of the Institution of Civil Engineers, Part 1, 1984, 76, pp 635-652.

Pimmler, T.& Eppinger, S., 1994, *Integration Analysis of Product Decomposition*, DE-Vol. 68, Design Theory and Methodology DTM'94, ASME 1994, pp 343-351

Plaria, A. & Carrie, A., 1995, *Application and Assessment of IDEF3 - Process Flow Description Capture Method,* International Journal of Operations and Production Management, Vol. 15, No. 1, pp 63-73

Platt, D. G. & Blockley, D. I., 1993, *An Integrated Process Support Environment for the Management of Civil Engineering Design,* SERC N & N Workshop, University of Salford, UK.

Powell, J. & Newland, P., 1993, *Construction Information. An Integrating Interface to Data,* SERC N&N Workshop, University of Salford, UK

Price, G. M., 1995, *The Management of Building Services Engineering Design*, MSc Project, Loughborough University of Technology, Loughborough, UK

Production Drawings - A Code of Procedure for Building Works, 1987, Co-ordinated project information, Building Project Information Committee

RIBA Royal Institute of British Architects, 1973, *Plan of Work for Design Team Operation*, RIBA Publications

Roberts, K. H., & O'Reilly III, 1974, *Measuring Organisational Communication*, Journal of Applied Psychology, Vol. 59, No.3, pp 321-326

Rogers, J., 1989, *A Knowledge-Based Tool for Multilevel Decomposition of a Complex Design Problem,* Technical Paper, NASA, May 1989.

Rogers, J., 1989, *DeMaid - A Design Manager's Aid for Intelligent Decomposition User's Guide,* Technical Memorandum, NASA. March 1989.

Rogers, J. & Padula, S., 1989, *An Intelligent Advisor for the Design Manager, Technical Memorandum,* NASA, February 1989.

Ronen, B. & Spiegler, I., 1991, *Information as Inventory*, Information and Management, 21, pp 239 - 247, North Holland

Ross, D. T., Brackett, J. W., Bravoco, R. R. & Schoman, K. E., 1980, *Architects Manual ICAM Definition Method 'IDEF0'*, I-CAM, DR-80-ATPC

Ross, D. T., 1977, *Structured Analysis (SA): A Language for Communicating ideas*, IEEE Transactions on Software Engineering, SE-3, (1), pp 16-24

Sanvido, V. & Norton, K., 1994, *Integrated Design Process Model*, Journal of Management in Engineering, ASCE, Vol. 10, No. 5, pp 55-62

Schwuchow, W., 1990, *Problems in Evaluating the Quality of Information Services*, in Information Quality, ed. Wormell, I., Taylor Graham, UK

Shannon, R., 1975, *System Simulation - The Art and Science,* Prentice Hall, Englewood Cliffs, London

Smith, R. & Eppinger, S., 1995, *Identifying Controlling Features of Engineering Design Iteration*, Working paper WP 3348-91-MS, Sloan School of Management, Massachusetts Institute of Technology, U.S.A.

Sobieski, J. S., 1982, *A Linear Decomposition Method for Large Optimization Problems - Blueprint for Development,* Technical Memorandum, NASA, February 1982.

Stephenson, P., & Naylor, D., 1993, *A Prototype System for the Coordination of Design and Project Information*, CIB W-65 Conference, Trinidad, W.I., September, pp. 269-277.

Steward, D., 1981, *Systems Analysis and Management: Structure, Strategy and Design*, Petrocelli Books, New York

Steward, D., 1981, *The Design Structure System: A Method for Managing the Design of Complex Systems,* IEEE Transactions on Engineering Management, Vol EM-28, No 3, August 1981, pp 71-74.

Steward, D., 1991, *Using Information Flow to Manage the Design of Systems*, IEEE, Portland International Conference on the Management of Engineering and Technology. October 1991, New York.

Tiong, R.L.K., 1990, *Effective Controls for Large Scale Construction Projects*, Project Management Journal, March 1990, Vol. XXI, pp 32 - 42

Topalian, A., 1979, *The Management of Design Projects,* Associated Business Press, London

Touran, A., 1990, *Integration of Simulation with Expert Systems*, Journal of Construction Engineering and Management, ASCE, Vol. 116, pp 480-493

Usmani, A. & Winch, G., 1993, *The Management of the Design Process, The Case of Architectural and Urban Projects*, A Barlett Research paper, University College of London

Venegas, J. A., 1987, *A Model for Design / Construction Integration during the Initial Phases of Design for Building Construction Projects*, PhD Thesis, Stanford University, U.S.A.

Wagner, G., 1990, *The Value and the Quality of Information: the need for a theoretical synthesis*, In Information Quality, ed. Wormell, I., Taylor Graham, UK

Ward, P. and Mellor, S., 1985, *Structured Development for Real-Time Systems,* Prentice Hall, Englewood Cliffs, NJ

Warren, J., Stott, J. & Norcio, A., 1992, *Stochastic Simulation of Information Systems Designs from Data Flow Diagrams*, Systems Software, 1992, 18, pp 191-199.

Winner, R. I., 1988, *The Role of Concurrent Engineering in Weapons Systems Acquisition*, Institute for Defence Analysis, Alexandria, Va, 1988, Rep. No. R-338

Woolery, J. & Crandall, K., 1983, *Stochastic Network Model for Planning Scheduling*, Journal of Construction Engineering and Management. ASCE, Vol 109, pp 342-354.

Yadov, S. B., Bravoco, R. R., Chatfield A. T. & Rajkumar, T. M., 1988, *Comparison of Analysis Techniques for Information Requirements Determination*, Communications of ACM, 31 (9), pp 1090-1097

Zairi, M., 1994, *Measuring Performance for Business Results*, Chapman & Hall, London

APPENDIX I

EXAMPLES OF FORMS USED FOR EXCHANGING INFORMATION AND AN EXAMPLE OF INFORMATION RELEASE SCHEDULE (IRS)

AMEC Design and Management

AMEC Design and Management Limited
Timothy's Bridge Road, Stratford-upon-Avon, Warwickshire CV37 9NJ
Telephone: (01789) 204288, Fax: (01789) 29913[illegible]

SHEET NO. 1 OF

PROJECT DESIGN NOTE

DESIGN GROUP	E	PDN

PROJECT: Loughborough University of Technology
New Engineering Complex

REV Nº

PROJ Nº 00585

DATE 10 August 1995

TITLE

APPROVAL / COMMENT REQUIRED IN: .... DAYS/WEEKS INITIATED: .... CHECKED: ....

CLIENT'S COMMENTS:
- AGREED WITH NO COMMENTS
- AGREED IF COMMENTS BELOW ARE INCORPORATED
- NOT AGREED, REVISION NEEDED FOR REASONS BELOW

* (Delete as required)

SIGNED: .... DATE : ....

Distribution	M A P Murray	J French	D Hammond	C Underwood	D Webley	B Roberts	A Lin
	N Keaney	A Young					
	Main File	PDN File	Client - R Hill - LUT		G Howard - Howard Associates		

pdn00585.wp

AMEC Design and Management
AMEC Design and Management Limited
Timothy's Bridge Road, Stratford-upon-Avon, Warwickshire CV37 9NJ
Telephone: (01789) 204288, Fax: (01789) 299135

SHEET NO. 1 OF

DESIGN CHANGE CONTROL

DESIGN GROUP

DCC

REV Nº

PROJECT — Loughborough University of Technology
New Engineering Complex

PROJ Nº 00585

DATE

TITLE

DESCRIPTION:

REASON FOR CHANGE:

DISCIPLINE:	INITIATED:	APPROVED DL:

Effect on Design Programme Weeks

DESIGN LEADER'S COMMENTS

- Variation to be raised
- Client to be informed of Prime Cost change
- Change not approved
- Proceed with change
- Client approval required

NAME: SIGNED: DATE:

Distribution on Completion								
	M Murray		J French		D Hammond		C Underwood	
	B Roberts		A List		N Keaney		A Young	
	D Webley		DCC File		Main File			

AMEC Design and Management

AMEC Design and Management Limited
Stratford-upon-Avon, CV37 9NJ
Telephone (01789) 204288 Fax (01789) 299135

SHEET NO. 1 OF 1

VARIATION ENQUIRY/ORDER

E

Nº

PROJECT Loughborough University of Technology
New Engineering Complex

PROJ Nº 00585

DATE

VARIATION REQUESTED FOR:

SOURCE:

EFFECT ON COSTS:

	EXPENDED TO DATE	TO COMPLETE	TOTAL
Design/Engineering	£		
Prime Costs	£		
OVERALL COSTS	£		£
(Budget/Firm)			Add/(Omit)

* PROGRAMME EFFECT:

IF INSTRUCTION RECEIVED BY:

* Exclusive of any cumulative effect of variations

SIGNED (for AMEC)........................ Date: ....

CLIENT RESPONSE: (Delete as necessary)

a) DO NOT PROCEED WITH VARIATION

b) DO (NOT) PROCEED WITH VARIATION - FURTHER INFORMATION REQUIRED AS LISTED

c) PROCEED WITH VARIATION

SIGNED (for Client):.................................... DATE:.........................

Dist	M A P Murray	J French	D Hammond	C Underwood	D Webley	E File	Main File
	B Roberts	A Lee	M Kearsey	A Young	Client - B Hall	G Howard	

AMEC Design and Management

AMEC Design and Management Limited
Stratford-upon-Avon, CV37 9NJ
Telephone (01789) 204288 Fax (01789) 299135

SHEET NO. 1 OF

CONTACT REPORT

DESIGN GROUP	E	Nº	
PROJECT	Loughborough University of Technology New Engineering Complex	PROJ NO	00585
		DATE	

CONVERSATION BETWEEN: LUT

AMEC

SUBJECT

ıe following matter was discussed by the above on SITE/TELEPHONE on

ıgned: Date:

[illegible]TRIBUTION	M A F Murray	[illegible]	[illegible]	C Underwood	D Webley	E [illegible]		
	[illegible] Roberts	A [illegible]	H Keaney	A Young	Client - R Hall	G Howard		

CONTRACTOR	JOHN LAING CONSTRUCTION LTD - LAING EASTERN
CONTRACT	GRAFTON CENTRE CAMBRIDGE PHASE 2
SUBJECT	INFORMATION RELEASE SCHEDULE
DATE	1ST JULY 1994

BASIS OF SCHEDULE

The following dates are based on our

Revised Master Programme

CHL 605/DMP/2A

(Sheets 1 & 2)

Dated 30th June 1994

STATUS OF INFORMATION

The dates represent the issue of fully

co-ordinated and dimensioned drawings

for construction.

Any preliminary information.

(i.e Psum / other)

is covered elswhere

1.0 SPECIFIC DRAWINGS INFORMATION

A number of A4 Plans have been marked up to indentify and indicate the location of either existing sections that require updating or the production of new drawings that we require by the given dates.

1.1 Appendix 1 (1.50 Sections)

Section FF	11/7/94
Section EF (but not stepped)	11/7/94
Section BB	11/7/94
Section XX new drawing required by	11/7/94
Section YY new drawing required by	11/7/94
Section ZZ new drawing required by	11/7/94

Also we require a reissue of updated plans at 1.100 scale again, by 04/7/94.

1.2 Appendix 2/3 - 1:20 Strip Sections and any associated 1:5 details

Section 1	Core 8	30/6/94
Section 2		30/6/94
Section 3		30/6/94
Section 4		30/6/94
Section 5		30/6/94
Section 6	Core 5	22/7/94
Section 7		15/7/94
Section 8		22/7/94
Section 9	Core 7	15/7/94
Section 10		15/7/94
Section 11		22/7/94
Section 12		22/7/94
Section 13		22/7/94
Section 14	Atrium	01/8/94
Section 15		01/8/94
Section 16		01/8/94
Section 17		01/8/94
Section 18		01/8/94
Section 19		01/8/94
Section 20		01/8/94
Section 21		01/8/94
Section 22	Malls	29/7/94
Section 23		29/7/94
Section 24		22/7/94
Section 25		22/7/94
Section 26		22/7/94
Section 27		22/7/94
Section 28		22/7/94
Section 29		22/7/94
Section 30	Presto Mezzanine	22/7/94

1.3 Appendix 4 Existing / New Roof Details

Existing details and new details to be reviewed, discussed and co-ordinated with trade contractors by mid July at latest	15/7/94

2.0 BHS AREA

	received	+/- weeks
2.1 Steelwork and Primary Structure		
2.1.1 Brick Suooort		
2.1.2 Core 3 / 4 - Cavity Wall Suooort	4/7/94	
2.1.3 Core / Tower No 2 - Brick Suooort	4/7/94	
2.1.4 Core / Tower No 2 - Curtain Wall Head	8/7/94	
2.1.5 Core / Tower No 2 - Curtain Wall Cill	5/7/94	
2.2 Brickwork		
2.2.1 Cooinq details - see		
2.2.2 DPC details - see		
2.3 Roofing and Cladding		
2.3.1 Rainwater Down Pioes / Hooders (layout, setout, sizes, colours)	08/7/94	
2.3.1 Roof Screed Area - Fall / Inlet Layout	15/7/94	
2.3.3 Uodated / New details indicated on Aooendix (5)		
A - Atruim Roof / BHS Paraoets	08/7/94	
B - Mansard / Brickwork / Cooinq	08/7/94	
C - Mansard / Brickwork / Cooing	08/7/94	
D - Mansard / Brickwork / Cooing	08/7/94	
E - Atrium Roof / BHS Cladding / Blockwork	08/7/94	
F - Brickwork / Flat Roof / Uostand	08/7/94	
2.4 Stair Core 8 Finishes		
Joinery Items - Comoletion (Cills)	08/7/94	
Door Schedule - Comoletion	04/7/94	
Ironmongery - Comoletion	04/7/94	
Gyoroc Ceiling Structure / Suooort / Edoe detail	19/7/94	
Finishes Schedule - External	08/7/94	
- Internal	19/7/94	
Metalwork Package (Comoletion)	15/7/94	

3.0 ENTRANCE TOWER & FOOD COURT

		received	+/- weeks
3.1 Steelwork and Primary Structure			
3.1.1 Link Bridge Steel / RC Work	05/7/94		
3.1.2 RC Details around Link / Tower at roof level	11/7/94		
3.1.3 Parapet Steel / Cladding Rails	08/7/94		
3.1.4 Curtain Wall Support on Tower			
- Cill Detail	18/7/94		
- Head Detail	26/7/94		
- Intermediate Support	26/7/94		
3.1.5 Tower Brick Support System	15/7/94		
3.1.6 Lintel Schedule	01/8/94		
3.1.7 Core 3 / 4 Lift Shaft Slab	11/7/94		
3.1.8 Core 3 / 4 Flat Roof Structure	11/7/94		
3.1.9 Structural Layout (main steel and Purlins for out standing area, see plan)	11/7/94		
3.2 Brickwork and Blockwork			
3.2.1 Setting Out	01/8/94		
3.2.2 Stone Details	11/7/94		
3.2.3 Copings	26/7/94		
3.2.4 Brick Support	11/7/94		
3.2.5 DPC's	01/8/94		
3.2.6 Lintel Schedule (Internal)	11/7/94		
3.2.7 Head Restraint Details (Internal)	18/7/94		
3.2.8 Fire Protection Schedule	26/7/94		
3.3 Roofing & Cladding			
3.3.1 Rainwater Down Pipes System	22/7/94		
3.3.2 Circular Roofing Details	11/7/94		
3.3.3 Updated / New details (1:5) see Appendix 4	22/7/94		

3.4 Windows and Doors		received	+/- weeks
3.4.1 Opening / Setting Out (Windows / Curtain Wall	/94		
3.4.2 DPC Detailing (final- Windows / Curtain Wall	08/8/94		
3.4.3 Head / Cill / Jamb Details (final- Windows / Curtain Wall)	/94		

3.5 Finishes

3.5.1 Suspended Ceilings -			
(i) Reflected Ceiling Plan (fully co-ordinated)	18/7/94		
(ii) Ceiling Layout (Bulkheads, Edge Detail, Services)	01/8/94		
3.5.2 Flooring			
(i) Terrazzo Setting out / Layout	15/8/94		
(ii) Terrazzo Details - Edge, Upstands, Services	15/8/94		
(iii) Vinyl Flooring Layout / Details (incl. Thresholds, Joints	12/9/94		
(iv) Ceramic Flooring Layout / Details (incl. Thresholds, Joints)	05/9/94		
(v) Other Flooring Layout / Details (incl. Thresholds, Joints)	12/9/94		
3.5.3 Joinery / Metalwork / Plumbing			
( ) Door Schedule (incl. Jamb / Head Sections)	01/8/94		
(ii) Ironmongery Schedule	15/8/94		
(iii) Sanitary Ware Schedule	01/8/94		
(iv) Toilet Panel / Vanity Unit Layout / Details	12/9/94		
(v) Fixture / Fixing	22/8/94		
(vi) Signage Schedule / Layout / Detail	08/8/94		
(vii) General Joinery Package (final)	08/8/94		
(viii) General Metalwork Package (final)	08/8/94		

3.5.4 Wall and General Finishes		received	+/- weeks
(i) Finishes Schedule - Basic	18/7/94		
(ii) Finishes - Complete & Specifications	15/8/94		
(iii) Wall Tiling Layouts / Details	05/9/94		

3.6 Specialist

3.6.1 Shop Fitting

(i) Poduim - Layout / Details	01/8/94		
(ii) Waitress Units - Layout / Details	01/8/94		

3.7 Builders Work Details

3.7.1 Escalator / Floor Details			
3.7.2 Mat Well / Frame Details	05/9/94		

4.6 Specialist Items

		received	+/- weeks
4.6.1 Roof Lining Details	26/7/94		
4.6.2 Internal Gable Details	15/8/94		

4.7 BWIC

4.7.1 Lift Door / Threshold / Jam Details	08/8/94		
4.7.2 Stair / Floor Junction Joint	08/8/94		
4.7.3 Floor Joint Details	08/8/94		
4.7.4 Mat Well (Frame (North Wall) Details)	29/7/94		

4.5 Finishes	received	+/- weeks

4.5.1 Susoended Ceilings -

(i)	Reflected Ceiling Plan (fully co-ordinated)	18/7/94
(ii)	Ceiling Layout (Bulkheads, Edge Detail, Services)	01/8/94

4.5.2 Flooring

(i)	Terrazzo Setting out / Layout (final)	15/8/94
(ii)	Terrazzo Details - Edge, Uostands, Services	15/8/94
(iii)	Vinyl Flooring Layout / Details (incl. Thresholds, Joints	12/9/94
(iv)	Ceramic Flooring Layout / Details (incl. Thresholds, Joints)	05/9/94
(v)	Other Flooring Layout / Details (incl. Thresholds, Joints)	12/9/94

4.5.3 Joinery / Metalwork / Plumoing

(i)	Door Scnedule (incl. Jamb / Head Sections)	01/8/94
(ii)	Ironmongery Scnedule	15/8/94
(iii)	Sanitary Ware Schedule	01/8/94
(iv)	Toilet Panel / Vanity Unit Layout / Details	12/9/94
(v)	Fixture / Fixing	22/8/94
(vi)	Signage Schedule / Layout / Detail	08/8/94
(vii)	General Joinery Package	08/8/94
(viii)	General Metalwork Package	08/8/94

4.5.4 Wall and General Finishes

(i)	Finishes Schedule - Basic	18/7/94
(ii)	Finisnes Scnedule - Comolete & Specifications	15/8/94
(iii)	Wall Tiling Layouts / Details	05/9/94

	received	+/- weeks
5.5.3 Joinery / Metalwork / Plumbing		
(i) Door Schedule (incl. Jamb / Head Sections)	01/8/94	
(ii) Ironmongery Schedule	15/8/94	
(iii) Sanitary Ware Schedule	01/8/94	
(iv) Toilet Panel / Vanity Unit Layout / Details	12/9/94	
(v) Fixture / Fixing	22/8/94	
(vi) Signage Schedule / Layout / Detail	08/8/94	
(vii) General Joinery Package	08/8/94	
(viii) General Metalwork Package	08/8/94	
5.5.4 Wall and General Finishes		
(i) Finishes Schedule - Basic	18/7/94	
(ii) Finishes Schedule - Complete & Specifications	15/8/94	
(iii) Wall Tiling Layouts / Details	05/9/94	
5.6 Specialist Items		
5.6.1 Internal Gables, Cladding - Layout / Details	15/8/94	
5.6.2 Roof Lining - Layout / Details	11/7/94 - urgent	
5.6.3 Column Casing - Layout / Details	01/8/94	
5.7 BWIC		
5.7.1 Floor Joints	05/9/94	
5.7.2 Escalator / Flooring Joints	05/9/94	
5.7.3 Water Feature Waterproofing Details	22/8/94	

6.0 EXTERNAL WORKS

		received	+/- weeks
6.1	Layout / Levels - Setting out / Drawings 29/8/94		
6.2	Layout - Walls (Founds, Bkwk, Stone 29/8/94		
6.3	Layout - Underground Ducts / Crossings 29/8/94		
6.4	Layout - Signage. Furniture, Metalwork 15/8/94		
6.5	Layout - Paving 29/8/94		

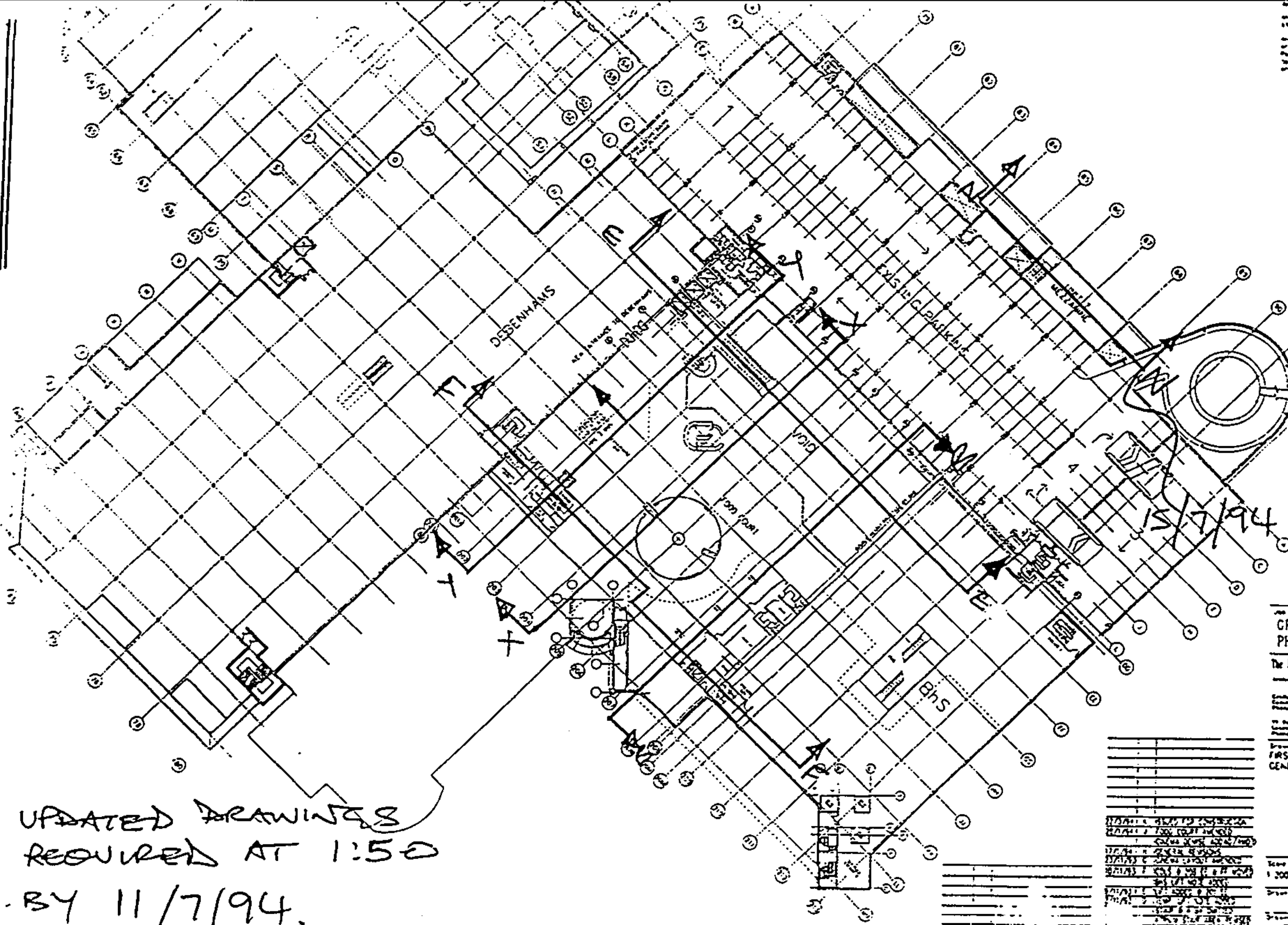
6.6 Details

		received	+/- weeks
6.6.1	Paving, Kerbs, Edging 29/8/94		
6.6.2	Furniture Schedule / Details 15/8/94		
6.6.3	Singage Schedule / Details 15/8/94		
6.6.4	Metalwork Details (Gates, Fencing, Barriers) 15/8/94		

4.0 CORE No. 7

	received	+/- weeks
4.1 Steelwork and Primary Steel		
4.1.1 Main Steel (stairs)	08/7/94	
4.1.2 Gable to Grid 203	04/7/94	
4.1.3 Tower Cladding, Rail System	05/7/94	
4.1.4 RC work to Landings	04/7/94	
4.1.5 Demolition & alterations (Remaining (i.e Car Park Flank Wall))	18/7/94	
4.2 Brickwork / Blockwork		
4.2.1 Lift Shaft / Car Park Flank Wall Setting out / Details (final)	18/7/94	
4.2.2 Plant Room Setting out / Details	26/7/94	
4.2.3 Toilet / Landing Setting out / Details (final)	26/7/94	
4.2.4 Lintel Schedule	18/7/94	
4.2.5 DPC	11/7/94	
4.2.6 Brick Support	11/7/94	
4.2.7 Copings, Stone Bands	18/7/94	
4.3 Roofing and New Cladding		
4.3.1 Updated 1:5 Sections (see Appendix 5)	15/7/94	
4.3.2 RW Pipes	01/8/94	
4.4 Windows and Doors		
4.4.1 Gable Curtain Wall Opening and Structural Details	04/7/94 - Urgent	
4.4.2 Co-ordinated Jamb / Head / Cill detail M.Price / Briggs Roofing	06/7/94 - Urgent.	

5.0 ATRIUM & SIDE MALLS	received	+/- weeks
5.1 Main Steel & Primary Structure		
5.1.1 Final Gable Cladding Rail Details	06/7/94	
5.1.2- Water Feature	01/8/94	
5.1.3 Internal Roof Lining Details	08/7/94	
5.2 Blockwork		
5.2.1 Side Malls - North set out	18/7/94	
5.2.2 Side Malls - South set out	08/8/94	
5.2.3 Perimeter Blockwork	08/8/94	
5.3 Roofing & Cladding		
5.3.1 Updated / New 1:5 Sections (as indicated in Appendix 5)	15/7/94	
5.4 Windows and Doors		
5.4.1 Aluminium Doors to Mall	15/7/94	
(N.B P-Sum information now 10 weeks overdue)		
5.5 Finishes		
5.5.1 Suspended Ceilings -		
(i) Reflected Ceiling Plan (fully co-ordinated)	18/7/94	
(ii) Ceiling Layout (Bulkheads, Edge Detail, Services)	01/8/94	
5.5.2 Flooring		
(i) Terrazzo Setting out / Layout	15/8/94	
(ii) Terrazzo Details - Edge, Upstands, Services	15/8/94	
(iii) Vinyl Flooring Layout / Details (incl. Thresholds, Joints)	12/9/94	
(iv) Ceramic Flooring Layout / Details (incl. Thresholds, Joints)	05/9/94	
(v) Other Flooring Layout / Details (incl. Thresholds, Joints)	12/9/94	



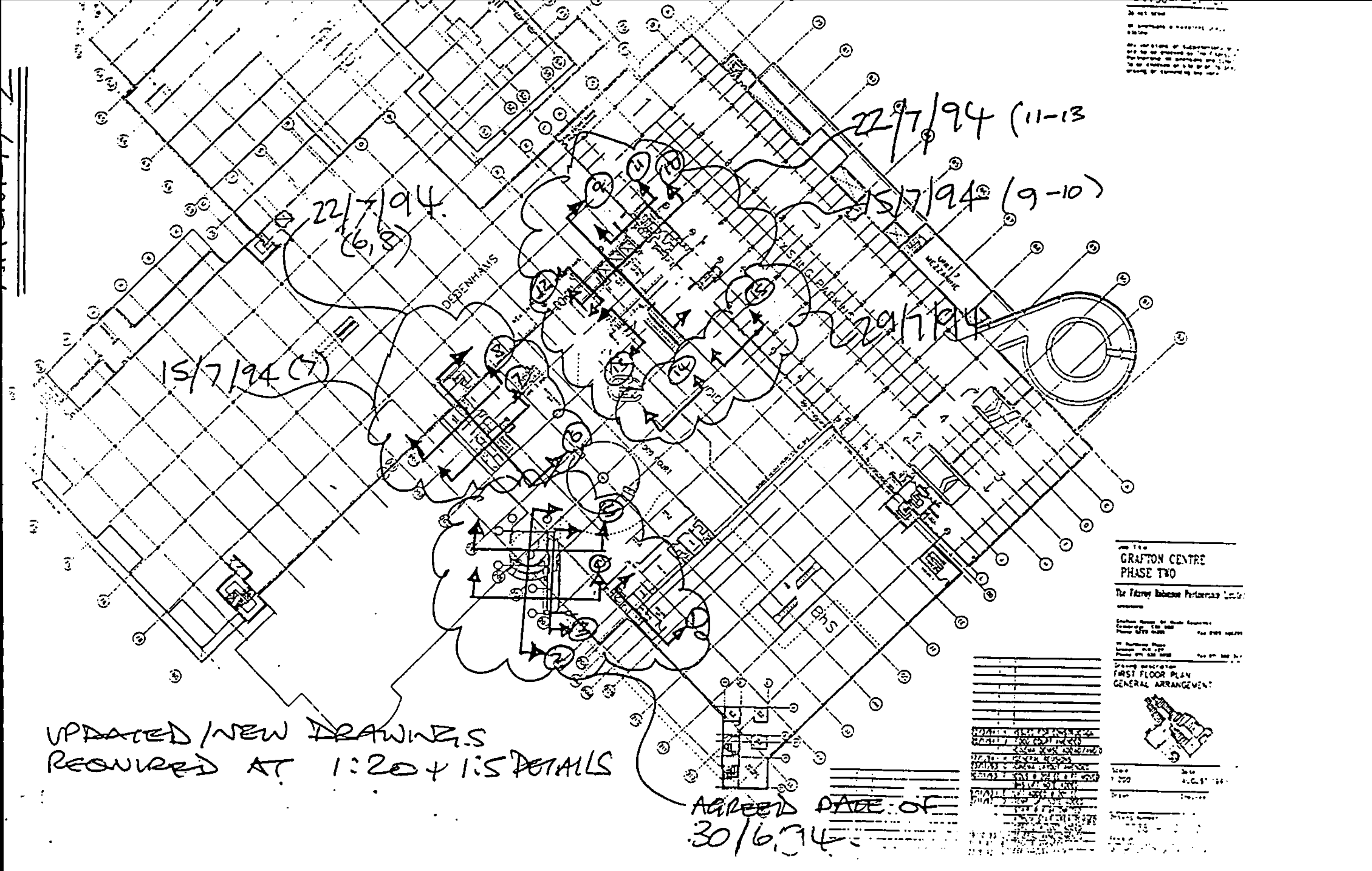
UPDATED DRAWINGS
REQUIRED AT 1:50
BY 11/7/94.

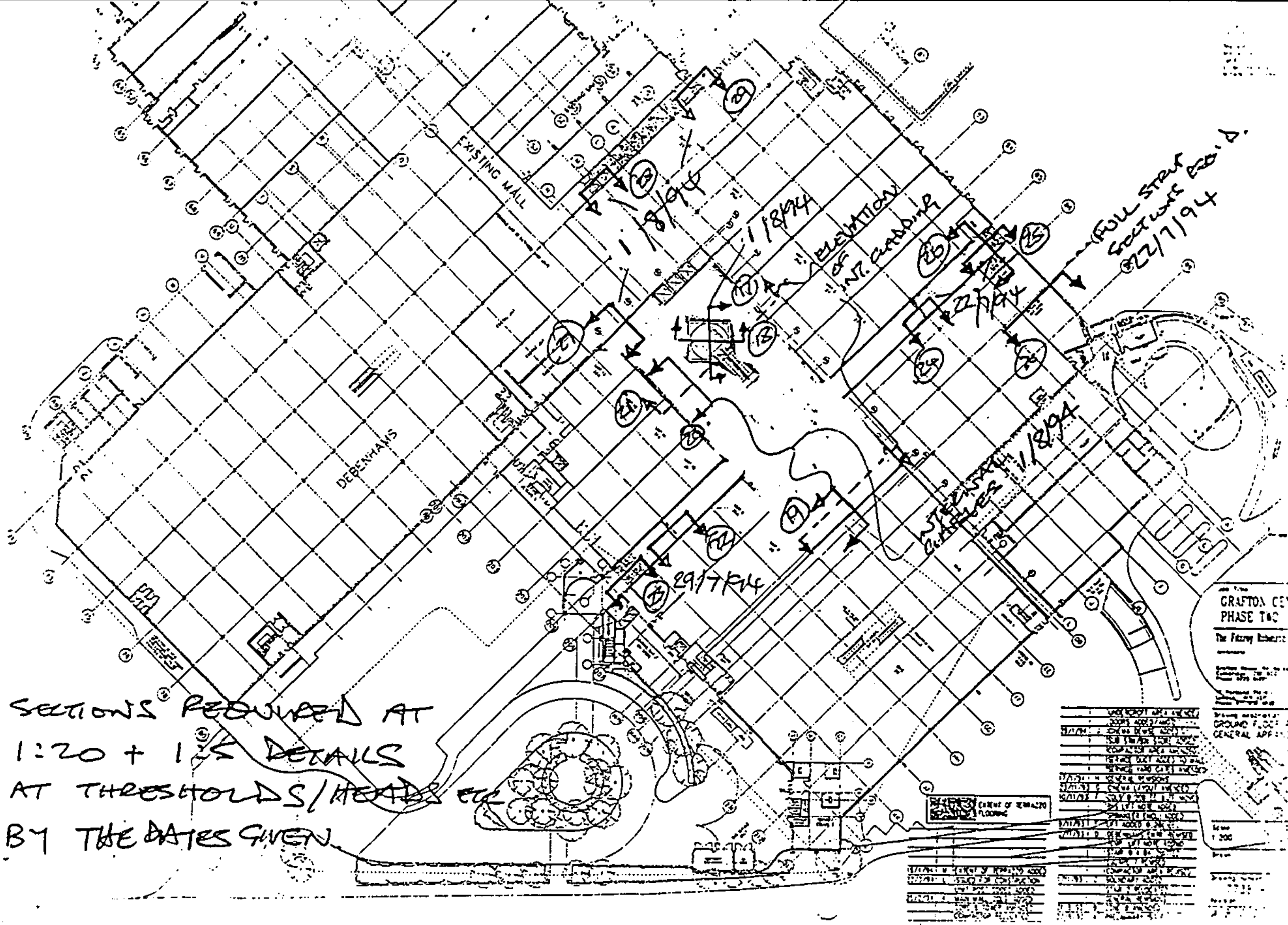
CRAFTON CENTRE
PHASE TWO

The Fitzroy Robinson Partnership Limited

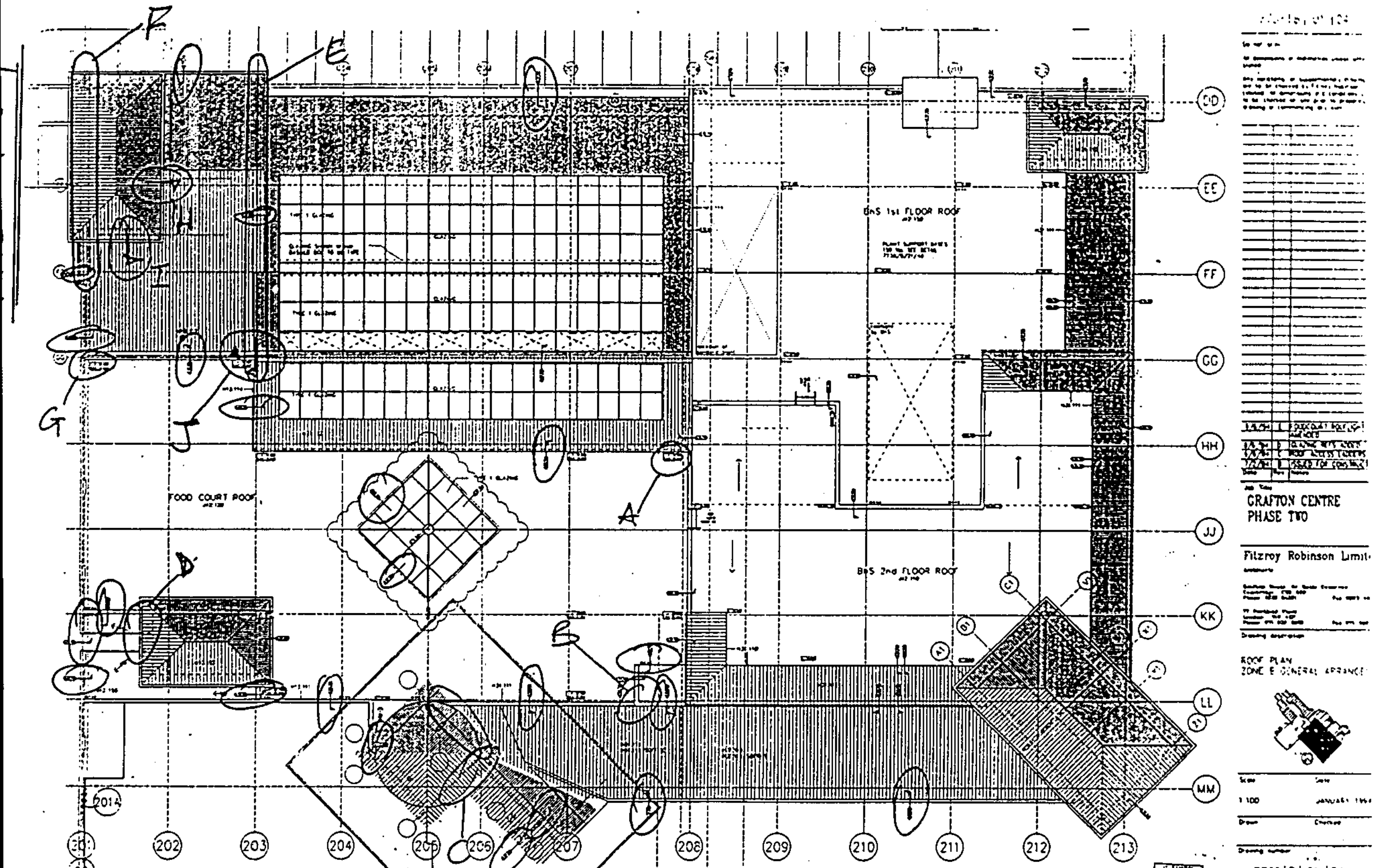
FIRST FLOOR PLAN
GENERAL ARRANGEMENT

1:200 AUGUST 199?


22/7/94 (11-13
15/7/94 (9-10)
22/7/94.
(6,8)
15/7/94 (7)
29/7/94
DEBENHAMS
UNIT 2 MEZZANINE
BhS
GRAFTON CENTRE
PHASE TWO
FIRST FLOOR PLAN
GENERAL ARRANGEMENT
UPDATED/NEW DRAWINGS
REQUIRED AT 1:20 + 1:5 DETAILS
AGREED DATE OF
30/6/94.



SECTIONS REQUIRED AT
1:20 + 1:5 DETAILS
AT THRESHOLDS/HEADS ETC
BY THE DATES GIVEN



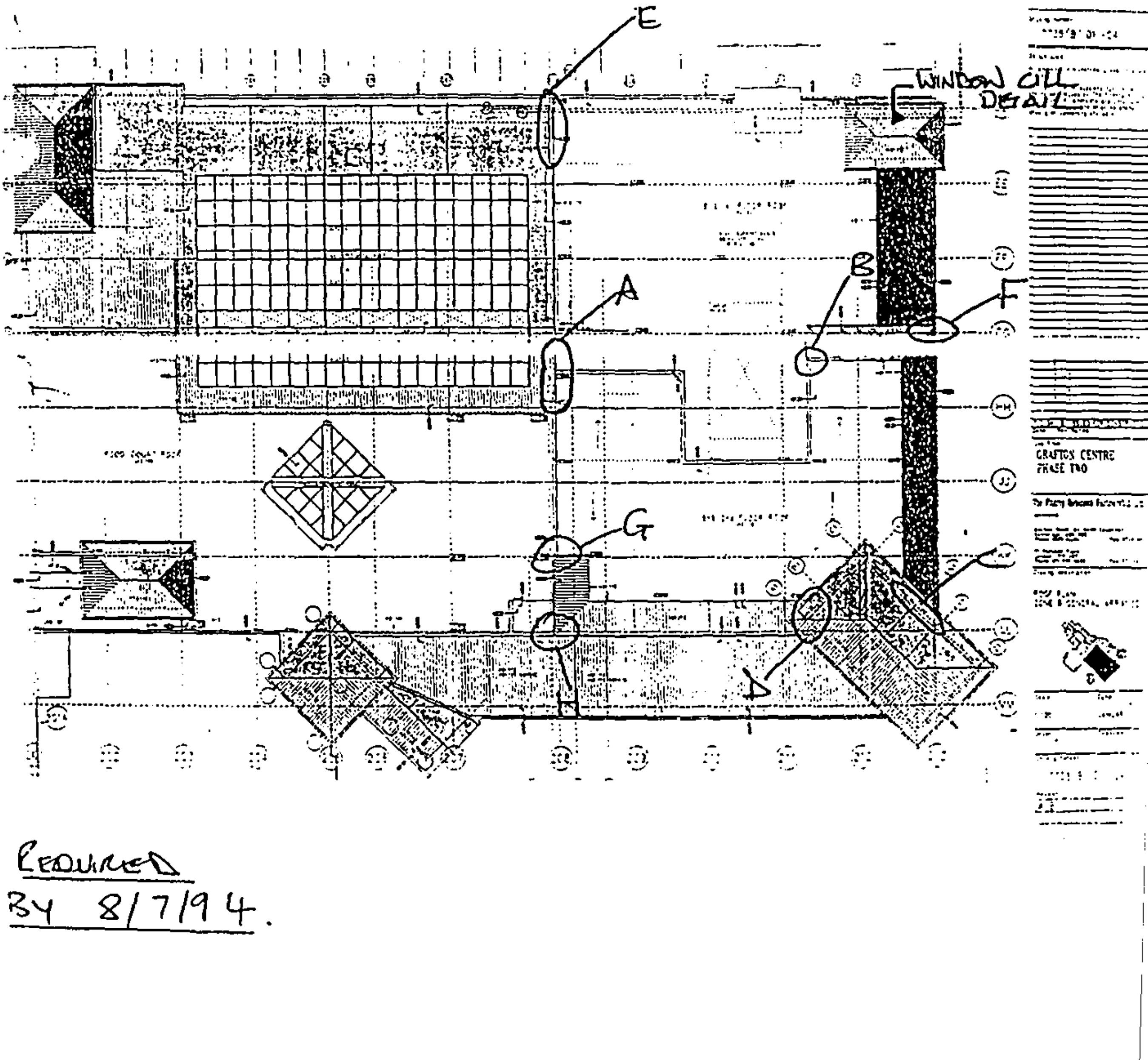
DETAILS TO BE FULLY REVIEWED / CO-ORDINATED AT 1:5 SCALE (AT A TO J HAVE YET TO BE ISSUED)

APPENDIX II

THE QUESTIONNAIRE SURVEY AND INTERVIEWS

APPENDIX

READ WITH IRS FOR BHS AREA



REQUIRED
BY 8/7/94.

Survey Document: The Management of the Design Process

This survey includes questions relating to all stages of the design process: Conceptual, Schematic, and Detailed Design. Please read the questions carefully and answer the questions by ticking the appropriate box or writing an answer in the space provided. *All responses are for the university research purposes and will be treated in strictest confidence. Please complete and return the questionnaire by 12th November 1994.*

Section 1

Organisational and Individual Questions

1. Please enter your Name and Position within Ove Arup organisation?

   ....

2. How many years of experience do you have in this position? .... years

3. How many projects are you involved with in a typical year? .... projects

.4. Are these projects generally

Design and Build ☐ Traditional ☐ Other ☐

Section 2

General Questions

1. The management of design at the *Schematic/Conceptual Design Stage* is more difficult than that at the *Detailed Design Stage*

   Agree ☐ Disagree ☐ No view

   Why? ....

2. The management of the *Schematic/Conceptual Design Stage* is the same irrespective of the form of procurement, (Traditional, Design & Build etc.)

   Agree ☐ Disagree ☐ No view ☐

   Why? ....

3. The management of the *Detailed Design Stage* is the same irrespective of the form of procurement, (Traditional, Design & Build etc.)

   Agree ☐ Disagree ☐ No view ☐

   Why? ....

4. Obtaining a realistic Design Brief for the new works is the most difficult task of the Design Manager

   Agree ☐ Disagree ☐ No view ☐

   Why? ....

5. The nature of the Design Brief obtained from the Client and the ease of obtaining it is dependant on the form of procurement, (Traditional, Design & Build etc.)

   Agree ☐ Disagree ☐ No view ☐

   Why? ....

6. What are the main difficulties for the design manager during the *Conceptual and Schematic Design Stages* and how may these be overcome?

....
....
....
....
....

Section 3

This section of the survey document relates specifically to the Schematic Design Stage. We have in our previous research identified the key data items and information flows relating to this design process. We now seek to focus on the important data items.

3.1 The following list of items relates to the SOURCES of information used in the *Schematic Design Process*. Please indicate on a scale of 1 to 5 the DIFFICULTY of obtaining accurate information from these sources (1 = very easy, 5 = very difficult)

Planning Authorities	1	2	3	4	5
Client	1	2	3	4	5
Building Control Officer	1	2	3	4	5
Local Authorities	1	2	3	4	5
National Rivers Authority	1	2	3	4	5
Highways Authority	1	2	3	4	5
Insurers	1	2	3	4	5

3.2 The following list of items relates to the Design Process in the *Schematic Design Stage*. Please mark on a scale of 1 to 5 the DIFFICULTY of completing these design tasks (1 = very easy, 5 = very difficult)

Site investigation	1	2	3	4	5
Schematic Drainage design	1	2	3	4	5
Schematic Architectural Design :					
Establish Fire Rating Requirements	1	2	3	4	5
Decide on Finishes & Materials	1	2	3	4	5
Roof Arch. Design	1	2	3	4	5
Developing Plans & Elevations	1	2	3	4	5
Developing Sections & Details	1	2	3	4	5
Outline Arch Specs Production	1	2	3	4	5
Landscaping Scheme Design	1	2	3	4	5
Architectural. Co-ordination	1	2	3	4	5
Schematic Structural Design :					
Foundations Schematic Design	1	2	3	4	5
Floor Slab Schematic Design	1	2	3	4	5
Ancillary Buildings Schematic Des.	1	2	3	4	5
Bearing Walls Schematic Design	1	2	3	4	5
Prelim. Structural Design Checks	1	2	3	4	5
Roof Schematic Structural Design	1	2	3	4	5
Structural Co-ordination	1	2	3	4	5
Structural Frame Schematic Design	1	2	3	4	5
Schematic Services Design	1	2	3	4	5
External Works Schematic Design	1	2	3	4	5
Outline Specs Production	1	2	3	4	5

Section 4

Other issues relating to the Conceptual and Schematic Design Process

4.1 During the *Conceptual and Schematic Design stages*, what are the communication difficulties commonly experienced between members of the design team and the *Client*?

....

....

....

4.2 During the *Conceptual and Schematic Design stages*, what are the communication difficulties commonly experienced amongst members of the *design team*?

....

....

....

4.3 Are there any Quality Assurance procedures in your organisation for communication at the *Conceptual and Schematic Design stages*, between the design team and the Client, or for communication between different members of the design team?

Yes ☐ No ☐

In your opinion, are such Quality Assurance procedures for communication at this stage of the design process essential?

Yes ☐ No ☐

Why? ....

....

Thank you for completing this questionnaire

Please indicate if you would be available for interview on 23rd of November 1994.

Yes ☐ No ☐

If your answer was no please indicate an alternative date ....

Summary of Response to Section 3

DIFFICULT >=4

	ARUP	AMEC	KS	Total
Planning Authorities		2	3	5
Client	2	1		3
Building Control Officer	2	3	1	6
Local Authorities	2	2	2	6
National Rivers Authority	1	2	1	4
Highways Authority	1	2	1	4
Insurers	2	2	1	5

[The numbers in the cells represent the number of interviewees at each organisation scoring >=4]

>=4	ARUP	AMEC	KS	Tot
Site investigation				
Schematic Drainage design		1		1
Schematic Architectural Design :				
Establish Fire Rating Requirements	1	1	1	3
Decide on Finishes & Materials		1	4	5
Roof Arch. Design	1			1
Developing Plans & Elevations	1	1		2
Developing Sections & Details		1		1
Outline Arch Specs Production		1	1	2
Landscaping Scheme Design				
Architectural. Co-ordination	1	1	1	3
Schematic Structural Design :				
Foundations Schematic Design	2			2
Floor Slab Schematic Design				
Ancillary Buildings Schematic Des.	1			1
Bearing Walls Schematic Design			1	1
Prelim. Structural Design Checks	1		1	2
Roof Schematic Structural Design				
Structural Co-ordination	1	1	3	5
Structural Frame Schematic Design				
Schematic Services Design	2	1	1	4
External Works Schematic Design			1	1
Outline Specs Production		1	1	2

[The numbers in the cells represent the number of interviewees at each organisation scoring>=4]

ANALYSIS OF DATA FROM THE SURVEY AND INTERVIEWS

For consistency purposes, it was decided to analyse the data collected from the twelve respondents to the survey document who were interviewed. However, the remaining eight respondents did not show any discrepancies with the results. The analysis showed the following:

- Nine of the interviewees agreed that the schematic design stage is more difficult to manage than the detailed design stage. Those who disagreed attributed the reason to the fact that there are less people involved and hence easier to control and co-ordinate, and that not all questions need to be answered at this stage. The same ratio was also within each organisation.

- Seven agreed that the management of the schematic design stage is the same irrespective of the procurement route. Two had no views over procurement related issues. There is no fixed point of time when the procurement route is decided.

- Nine disagreed that the management of the detailed design stage is the same irrespective of the procurement route. Two had no views.

- Nine agreed that obtaining the design brief is the most difficult task for the design manager during the conceptual/schematic design stage. The three who disagreed were a mechanical engineer, public health engineer, and a civil engineer mainly involved for managing design projects for a leading supermarket chain. This type of building, being repetitive in its nature usually presents a well structured brief. This gives some evidence that Architects have more problems in obtaining the Client's brief than other disciplines' engineers, and that the brief is more difficult to obtain for unique projects than in case of repetitive type projects.
  Obtaining a structured design brief was considered by some interviewees as the responsibility of the Clients' consultants, that there should be professional skilled personnel capable of extracting the brief from the Client and that the brief can be developed as the design progresses (room data sheets may be used for this purpose)

- Six interviewees disagreed that the ease or difficulty of obtaining the design brief depends on the procurement route. Two had no views.

- Eleven agreed that Quality Assurance procedures are essential for communication during the conceptual/schematic design stage.

- Due to the fact that all the interviewees were of different backgrounds (refer to Table 4.2), there was no real pattern in the difficult information sources identified nor the difficult design tasks identified. A summary of the response for the difficult tasks and information sources is included earlier in this Appendix. (For simplicity, some of the design tasks were indicated at a higher level than the functional primitive tasks).

- The difficulties with external information sources where approvals and regulations were necessary, (e.g. different authorities, insurers) were due to the fact that these sources are involved after substantial part of the design has been already completed, and hence any input will require re-design and other implications on other design tasks. Also there was a difficulty in interpretation of regulations and the time taken by these sources to take decisions or provide approvals.

- Although one manager identified some difficult design tasks to perform, he was not able to comment on details of information requirements for these tasks, as he saw that designers involved directly in these disciplines are better to comment.

- For the difficult design tasks identified by the interviewees, the difficulty in obtaining the information requirements for these tasks varied, but the importance of these information did not vary. The interviewees identified all technical information as important (ranked as >=5 on a scale of 1-7, 7 being most important) for the design task to proceed. (Information like the approved program and approved cost plan was seen by some as less important)

- Difficulty in obtaining information and the importance of some information were seen from different perspectives according to the background and discipline of the interviewee. A piece of information may be important or difficult for one designer or manager from his point of view, but not the same with another designer.

- Difficulties in communications or acquiring information are more when dealing with external sources. Information required from sources or disciplines within the same organisation is easier than obtaining information from externals, as it is more difficult to control the externals.

- Co-ordination between different design disciplines was considered by managers as the most important and difficult task. This task is an on-going managerial task throughout the design process. It cannot be modelled in DFD's.

- The most difficult elements in the design brief to obtain were related to users' requirements (number of users, spaces, machines sizes, etc.)

- The difficulties for the design manager during the schematic/conceptual design stage were seen as :

Client related issues:	- Establish relationship with Client - Clients' brief - Changes introduced by the Client - Fulfilling Clients' actual requirements
Project related issues:	- Time scale - Identifying project objectives - Allocating appropriate resources
Communication issues:	- Ensure all parties are aware of each others activities and requirements - Co-ordination of all design disciplines (communication problems are discussed later in this Appendix)

- The order of design tasks as decided by designers was determined in a very broad and global way. Frozen layouts were considered as a very important milestone where all design disciplines can proceed on its basis.

- All interviewees indicated that there is no formal way to judge the quality of information. The measure of good quality is if the information provided is enough to proceed to a next stage. The measure of poor quality is if the information is not enough or if there are complaints from the recipient. Some interviewees suggested adding status to information (e.g. preliminary, assumed) or adding a scale of accuracy on the drawings (e.g. 70% accurate). Quality judgement differs according to the sender and the recipient (e.g. single line drawings produced by mechanical engineers cannot be compared with architectural drawings).

- The interviewees indicated that information of insufficient quality or missing information are supplemented by assumptions from designers and confirmation is sought from the source. Any deviation from the assumptions may require re-design. One interviewee referred to past precedence in dealing with missing information or information of insufficient quality.

- Communication difficulties with the Client were considered by the interviewees as due to:
  - Frequent changes with lack of appreciation of the impact of changes
  - Client communicating only what he thinks is important
  - Decision making
  - Loose brief

- Communication problems among design team members showed some consistency within the same organisation. This indicates that communication problems are mainly attributed to organisational issues.

- Communication problems among team members were summarised as follows:
  - Conflicts due to different personalities and human behaviour issues
  - Lack of appreciation of effects of changes across disciplines
  - Unavailability of some team members during meetings due to being busy in other projects.
  - Geographical distances between team members. (If such distances were existing.)
  - Lack of awareness of some disciplines for other disciplines problems leading to thinking that others are asking irrelevant questions
  - Designers of each discipline do not know what other disciplines are expecting them to provide (e.g. in mechanical, drawings may be single line)
  - Speed of this design stage can prevent team members of becoming adequately familiar with each other or with the Client.
  - Lack of experience for some disciplines to advise other disciplines without carrying out the actual design.
  - Passing information between disciplines
  - Agree at which stage will the design development be frozen
  - Engineers pressurising Architects to provide scheme drawings quickly to enable them to start their design.

- There are special communication difficulties between Architects and engineers from other disciplines due to the difference in the way of thinking :

Engineers	Architects
- Think sequentially	- Do not think sequentially
- Think in terms of systems	- Think about overall problem
- Do not thinking 3D	- Think 3D
- Communicate more in writing	- Communicate more vocally

- Suggestions to overcome communication problems were:
  - Having a decision maker from the Client's side in the designer's offices
  - Holding regular formal and informal meetings and keeping every team member informed
  - All team members to be in the same geographical area
  - Personal training
  - Educate designers to see problems of other disciplines' designers
  - Train designers to understand key issues in other disciplines
  - More use of Information Technology
  - Prepare early framework for the Client to know what he will be asked
  - Architects to release information more quickly with stating assumptions or a scale of accuracy on the drawings.

- Having all members from different disciplines of the same design team in the same office area facilitates communications (as in the case of AMEC)

- Two organisations out of the three (AMEC, ARUP) had used electronic means of information transfer on most projects with different degrees. At Arup, the electronic information exchange had been used to link different international branches through a satellite. Within the UK., electronic transfer typically included exchanging drawings on disks.

  At AMEC, electronic information transfer includes e-mail, centralised CAD, desktop conferencing and modem links between site and office. However, electronic transfer of data was normally supported by exchange of data in paper format especially in design issues.

- Eleven out of the twelve interviewees indicated that the DFD models were useful and could improve communications. The areas of usefulness are:
  - Identification of information requirements for design tasks

- They help to identify other designers' problems
- High levels are useful for management purposes
- May be fed back in training of engineers and Architects

- Although it was expected that architects will be comfortable with DFD's as it is a graphical representation, one Architect pointed out that this is a more engineering approach and architects may be less comfortable with it. Another experienced design manager (with a civil and structures background) pointed out that high level DFD's are useful in case of modelling process engineering not building design.

- Communicating design information differs in multi-disciplinary organisations than in specialised ones. In house design team of a Design and Build contractor was considered by Kyle Stewart interviewees as offering better communications and better value analysis as design can be discussed more closely and in a more open way than in discussing design with a different party.

Conclusions of the study

The following conclusions were drawn from the study:

(a) The management of the Conceptual/Schematic design stages is more difficult than the Detailed design stage. However, further investigation is required to assess the benefits of simulating each of these stages. (This investigation has been carried out during the case study described in chapter 7.)

(b) There is no fundamental impact of the procurement strategy on the produced Generic Data Flow Model for the Conceptual/Schematic design stages. Therefore, the model is valid for all types of procurement.

(c) Difficulty in obtaining design information and the importance of some information vary according to the background and discipline of the designer or the design manager involved.

(d) Required elements of the design brief should be elicited by design staff in a structured way. The data dictionary of data flow models may be used to identify these elements according to the nature of each project.

(e) Communication problems experienced during the design process may lead to 'gatekeeping' (withholding) of information among participants intentionally or non intentionally.

(f) There is no formal way to judge the quality of information exchanged. The measure of information quality varies according to the sender and the recipient of information. Missing information or information of insufficient quality from the recipients point of view are supplemented by assumptions.

(g) Electronic information exchange provides fast effective data exchange. Such means of data exchange have not yet been fully exploited by the construction industry.

(h) Allocating appropriate resources and efficient resource utilisation is directly proportional to the efficient management of the design process.

(i) Data flow models provide a useful effective tool which may be used to improve communications during the design process, and hence improve the management of the process. These models assist in identifying information requirements for different design tasks, and in identifying other designers' problems. They may be used in the training of engineers and architects.

List of interviewees

Name	Organisation	Position	Background
B. Clifford	Arup	Assoc. Director	Design
C. Evans	Arup	Associate	Design, Civil
Ms. M. Whild	Arup	Arch. sub. consult	Arch., design manag.
D. Webley	AMEC	Group Manager	Mech., Design of process eng. projects
D. Starr	AMEC	Project Manager	Design, Design&Build, labs
M. Murphy	AMEC	Principal Arch.	Arch., build.design, refurb.,labor.
O. Vickery	AMEC	Principal Mech. engineer	Building services, design leader
J. French	AMEC	Principal Architect	Arch., Pharmac. projects
A. Robertson	Kyle Stewart	Associate	Public health eng., design manager, pharmac. projects
J. Dixon	Kyle Stewart	Principal Arch. Assistant	Arch. design, site manager
J. Cunliffe	Kyle Stewart	Assoc., Project design manager	Civil & structures
D. Carlile	Kyle Stewart	Director	Civil engineer,design manager

Total number of interviewees = 12

APPENDIX III

SAMPLE REPORTS FROM THE DATA DICTIONARY OF THE DATA FLOW MODEL FOR FACTORY DESIGN

All Definitions

Name Description	Type	Date
ACCEPTABLE-NOISE-LEVELS	Data Element	02/06/94
ACCESS FOR MACH VEH	Data Flow	21/02/94
ACCESS-EGRESS	Data Element	02/06/94
ADMIN AREA LAYOUT	Data Flow	10/02/94
ADMIN_AREA_LAYOUT	Data Element	09/02/94
AIR_COND	Data Element	09/02/94
APP SCHEME REPORT *APP-SCHEME-ARCH-DES+* *APP-SCHEME-STR-DES+* *APP-SCHEME-DRAIN-DES+* *APP-SCHEME-SERV-DES+* *APP-PROG+* *APP-COST-PLAN*	Data Flow	21/03/94
APP-CABLE-ROUTING-SCHEME	Data Element	21/02/94
APP-CLADDING-SCHEME	Data Element	21/02/94
APP-COM-ARCH-DES	Data Element	17/02/94
APP-COM-SERV-DES	Data Element	17/02/94
APP-COM-STR-DES	Data Element	17/02/94
APP-COMPUTER--SERV-PROV-SCHEME	Data Element	25/02/94
APP-COST-PLAN	Data Element	21/02/94
APP-EMERGENCY-LIGHTING-SCHEME	Data Element	21/02/94
APP-ENVELOPE-MAT	Data Element	21/02/94
APP-ENVELOPE-SCHEME *APP-ENVELOPE-MAT+* *APP-ROOF-SCHEME+* *APP-CLADDING-SCHEME*	Data Structure	23/02/94
APP-FIRE-ALARM-SCHEME	Data Element	21/02/94
APP-FIRE-FIGHTING-SCHEME	Data Element	21/02/94
APP-FLOOR-SLAB-SCHEME	Data Element	21/02/94
APP-FOUL-WATER-DRAIN-SCHEME	Data Element	21/02/94
APP-FOUND-SCHEME	Data Element	21/02/94
APP-GAS-PIPING-SCHEME	Data Element	21/02/94
APP-HEATING&PIPED-SERV-SCHEME	Data Structure	21/02/94

All Definitions

Name Description	Type	Date
APP-HVAC-SCHEME+ *APP-HEATING-SCHEME+* *APP-VENT-SCHEME+* *APP-VACUUM-PLANT-SCHEME+* *APP-FIRE-FIGHTING-SCHEME+* *APP-POTABLE-WATER-PIPING-SCHEME+* *APP-GAS-PIPING-SCHEME*		
APP-HEATING-SCHEME	Data Element	21/02/94
APP-HVAC-SCHEME	Data Element	21/02/94
APP-ILLUM-LEVELS-SCHEME	Data Element	21/02/94
APP-LIGHTED-AREAS-SCHEME	Data Element	21/02/94
APP-LIGHTNING-PROT-SCHEME	Data Element	21/02/94
APP-POTABLE-WATER-PIPING-SCHEME	Data Element	21/02/94
APP-POWER&LIGHTING-SCHEME *APP-POWER-LOADS+* *APP-CABLE-ROUTING-SCHEME+* *APP-COMPUTER--SERV-PROV-SCHEME+* *APP-LIGHTED-AREAS-SCHEME+* *APP-ILLUM-LEVELS-SCHEME+* *APP-EMERGENCY-LIGHTING-SCHEME*	Data Structure	25/02/94
APP-POWER-LOADS	Data Element	25/02/94
APP-PROG	Data Element	21/02/94
APP-ROOF-SCHEME	Data Element	21/02/94
APP-SCHEME-ARCH-DES *APP-SCHEME-LAYOUTS+* *APP-ENVELOPE-SCHEME+* *APP-SCHEME-FINISHES*	Data Structure	21/03/94
APP-SCHEME-DRAIN-DES *APP-SURF-WATER-DRAIN-SCHEME+* *APP-FOUL-WATER-DRAIN-SCHEME*	Data Structure	21/02/94
APP-SCHEME-FINISHES	Data Element	21/02/94
APP-SCHEME-LAYOUTS *APP-SCHEME-SITE-LAYOUT+* *APP-SCHEME-PROD-AREA-LAYOUT+* *APP-SCHEME-OFFICES-LAYOUT*	Data Structure	23/02/94
APP-SCHEME-OFFICES-LAYOUT	Data Element	21/02/94
APP-SCHEME-PROD-AREA-LAYOUT	Data Element	21/02/94
APP-SCHEME-SERV-DES *APP-HEATING&PIPED-SERV-SCHEME+* *APP-POWER&LIGHTING-SCHEME+* *APP-TELECOM&DATA-SCHEME+*	Data Structure	21/02/94

All Definitions

Name Description	Type	Date
APP-SPECIAL-SYSTEMS-SCHEME		
APP-SCHEME-SITE-LAYOUT *SITE-ACCESS+* *LOADING&UNLOADING-FACILITIES+* *PARKING-FACILITIES*	Data Structure	21/02/94
APP-SCHEME-STR-DES *APP-FOUND-SCHEME+* *APP-FLOOR-SLAB-SCHEME+* *APP-SUPERSTR-SCHEME*	Data Structure	25/02/94
APP-SECURITY-SYSTEM-SCHEME	Data Element	21/02/94
APP-SPECIAL-SYSTEMS-SCHEME *APP-FIRE-ALARM-SCHEME+* *APP-SECURITY-SYSTEM-SCHEME+* *APP-LIGHTNING-PROT-SCHEME*	Data Structure	21/02/94
APP-SUPERSTR-SCHEME	Data Element	21/02/94
APP-SURF-WATER-DRAIN-SCHEME	Data Element	21/02/94
APP-TELECOM&DATA-SCHEME	Data Element	21/02/94
APP-VACUUM-PLANT-SCHEME	Data Element	21/02/94
APP-VENT-SCHEME	Data Element	21/02/94
APPROVALS AND COMMENTS *APP-COM-STR-DES+* *APP-COM-ARCH-DES+* *APP-COM-SERV-DES*	Data Flow	21/02/94
ARCH-DES-DOC *ARCH-DRAWINGS+* *ARCH-SPECS*	Data Structure	21/02/94
ARCH-DRAWINGS	Data Element	21/02/94
ARCH-SPECS	Data Element	21/02/94
ARCH_COST_INF	Data Element	09/02/94
ASSUMED FOUNDATIONS *FOUNDATION_TYPE+* *FOUNDATION_DEPTH*	Data Flow	08/02/94
ASSUMED_GROUND_COND	Data Element	08/02/94
BLOCK-WORKS-DETAILS	Data Element	24/02/94
BUDGET	Data Element	09/02/94
BUILDING COST INFORMATION *ARCH_COST_INF+* *STR_COST_INF+*	Data Flow	09/02/94

All Definitions

Name Description	Type	Date
SERV_COST_INF		
BUILDING REGULATIONS *FIRE-WALLS-LOCATIONS+* *PETROL-INTERCEPTORS-REQ+* *HOSE-REELS-REQ+* *ESCAPE-RAMPS-SLOPES+* *ACCESS-EGRESS+* *FIRE-SAFETY-REQ+* *ACCEPTABLE-NOISE-LEVELS+* *ENVIRONMENTAL-STANDARDS*	Data Flow	02/0C/94
BUILDING_LAYOUT	Data Element	10/02/94
CLADDING-DETAILS	Data Element	24/02/94
COLOUR-SCHEDULES	Data Element	24/02/94
COMPRESSED_AIR	Data Element	09/02/94
COMP_AIR	Data Element	09/02/94
COORD-SPECS	Data Element	25/02/94
DESIGN TENDER DOCUMENTS *STR-DES-DOC+* *ARCH-DES-DOC+* *SERV-DES-DOC+* *DRAIN-DES-DOC+* *COORD-SPECS*	Data Flow	25/02/94
DESIGN_FLOOR_SLAB_UNIFORM_LOAD	Data Element	24/02/94
DESIGN_LOADS *DESIGN_FLOOR_SLAB_UNIFORM_LOAD*	Data Flow	24/02/94
DEVELOPED LAYOUTS *DIMENSIONED-OFFICE-AREA-LAYOUT+* *DIMENSIONED-PROD-AREA-LAYOUT*	Data Flow	17/02/94
DEVELOPEDLAYOUTS *DIMENSIONED-OFFICE-AREA-LAYOUT+* *DIMENSIONED-PROD-AREA-LAYOUT*	Data Flow	22/02/94
DEVELOPING SECTIONS& DETAILS *INCLUDES DEVELOPING SECTIONS&DETAILS FOR:* *GLAZING* *CLADDING* *BLOCK WORKS* *DRY PARTITIONS* *FALSE CEILING* *DOORS&DOOR SCHEDULES*	Process	24/02/94
DIMENSIONED-OFFICE-AREA-LAYOUT	Data Element	17/02/94
DIMENSIONED-PROD-AREA-LAYOUT	Data Element	17/02/94

All Definitions

Name Description	Type	Date
DOOR-SCHEDULES	Data Element	24/02/94
DRAIN-DES-DOC *DRAIN-DES-DRAWINGS+* *DRAIN-SPECS*	Data Structure	21/02/94
DRAIN-DES-DRAWINGS	Data Element	21/02/94
DRAIN-SPECS	Data Element	21/02/94
DRINK-WAT-PTS	Data Element	21/02/94
DRY-PARTITIONS-DETAILS	Data Element	24/02/94
ENVIRONMENTAL-STANDARDS	Data Element	02/06/94
ESCAPE-RAMPS-SLOPES	Data Element	23/02/94
EXISTING_DEVELOPMENTS	Data Element	08/02/94
EXIST_SEWER_FACIL	Data Element	08/02/94
FALSE-CEILING-DETAILS	Data Element	24/02/94
FINISHESOUTLINES *FINISHES_TYPES+* *FINISHES_MATERIALS*	Data Flow	09/02/94
FINISHES_MATERIALS	Data Element	09/02/94
FINISHES_STANDARDS *QUALITY OF FINISHING MATERIALS TO BE USED*	Data Element	08/02/94
FINISHES_TYPES	Data Element	09/02/94
FIRE-SAFETY-REQ	Data Element	23/02/94
FIRE-WALLS-LOCATIONS	Data Element	23/02/94
FIRE_ALARM_AND_DETECTION	Data Element	09/02/94
FL_SLAB_UNIF_LOAD	Data Element	25/02/94
FOUNDATION_DEPTH	Data Element	08/02/94
FOUNDATION_TYPE	Data Element	08/02/94
GLAZING-DETAILS	Data Element	24/02/94
HANDDRYERS-DIST	Data Element	21/02/94
HANDDRYERS-LOC	Data Element	21/02/94
HANDINGOVER_DATE	Data Element	08/02/94
HEATING&PIPED SERV	Data Flow	21/02/94

All Definitions

Name Description	Type	Date
HEATING_PIPED_SERV_PROD_AREA+ *HEATING_PIPED_SERV_LAB+* *HEATING_PIPED_SERV_OFFICES+* *HEATING_PIPED_SERV_EXTERNAL*		
HEATING_PIPED_SERV_EXTERNAL *PERIMETER_FIRE_RING_MAIN+* *INCOMING_POTABLE_WATER+* *INCOMING_FIRE_MAIN+* *SPRINK_TANK+* *PUMP_HOUSE+* *INCOMING_NAT_GAS*	Data Structure	21/02/94
HEATING_PIPED_SERV_LAB *AIR_COND+* *COMPRESSED_AIR+* *LOCAL_EXTRACT*	Data Structure	09/02/94
HEATING_PIPED_SERV_OFFICES *RADIATOR_SYSTEM+* *MECH_VENT_OFFICE_AREA+* *MECH_VENT_TOILETS+* *HOT_AND_COLD_WATER+* *KITCHEN_EXTRACT*	Data Structure	09/02/94
HEATING_PIPED_SERV_PROD_AREA *WARM_AIR_SYSTEM+* *SMOKE_VENTILATORS+* *COMP_AIR+* *VACUUM_PLANT*	Data Structure	09/02/94
HOSE-REELS-REQ	Data Element	23/02/94
HOT_AND_COLD_WATER	Data Element	09/02/94
INCOMING_FIRE_MAIN	Data Element	09/02/94
INCOMING_NAT_GAS	Data Element	09/02/94
INCOMING_POTABLE_WATER	Data Element	09/02/94
KITCH-FAC-LOC	Data Element	21/02/94
KITCHEN_EXTRACT	Data Element	09/02/94
LAYOUTS *SITE_LAYOUT+* *PROD_AREA_LAYOUT+* *ADMIN_AREA_LAYOUT*	Data Flow	21/03/94
LIGHTING *ILLUMINATION_LEVEL+* *FIXTURES_TYPES+* *CAR_PARK_ILLUMIN+* *FOOTPATHS_ILLUMIN+*	Data Element	09/02/94

All Definitions

Name Description	Type	Date
EMERGENCY_LIGHTING		
LIGHTING-FIXT-DIST	Data Element	21/02/94
LIGHTNING_PROTECTION	Data Element	09/02/94
LOADING&UNLOADING-FACILITIES	Data Element	21/02/94
LOCAL_EXTRACT	Data Element	09/02/94
MATERIALS SELECTION *SELECTION OF AUXILLARY MATERIALS WITH THE SKELETON, MATERIALS FOR THE ROOF, INTERNAL PARTITIONS AND INTERNAL AND EXTERNAL FINISHES.*	Process	10/02/94
MECH ROOMS *MECH_ROOMS_SIZES+* *MECH_ROOMS_LOCATION*	Data Flow	09/02/94
MECH_ROOMS_LOCATION	Data Element	09/02/94
MECH_ROOMS_SIZES	Data Element	09/02/94
MECH_VENT_OFFICE_AREA	Data Element	09/02/94
MECH_VENT_TOILETS	Data Element	09/02/94
PARKING-FACILITIES	Data Element	21/02/94
PERIMETER_FIRE_RING_MAIN	Data Element	09/02/94
PETROL-INTERCEPTORS-REQ	Data Element	23/02/94
PHASING_REQ	Data Element	08/02/94
POWER&LIGHT REQ *POWER_SUPPLIES+* *LIGHTING*	Data Flow	09/02/94
POWER-SUPPLY-DIST	Data Element	21/02/94
POWER_SUPPLIES *VOLTAGE_TYPE+* *ESTIMATED_LOADS+* *CABLE_TYPES+* *DISTRIBUTION_SYSTEM+* *COMPUTER_UPS_REQ*	Data Element	09/02/94
PROCUREMENT_METHOD	Data Element	08/02/94
PROD_AREA_LAYOUT	Data Element	09/02/94
PROJECT BRIEF *SITE_BOUNDARIES+* *SITE_LOCATION+* *EXISTING_DEVELOPMENTS+* *PHASING_REQ+* *TENDER_POLICY+*	Data Flow	21/03/94

All Definitions

Name Description	Type	Date
HANDINGOVER_DATE+ *FINISHES_STANDARDS+* *EXIST_SEWER_FACIL+* *ASSUMED_GROUND_COND+* *FL_SLAB_UNIF_LOAD+* *BUDGET*		
PUMP_HOUSE	Data Element	09/02/94
RADIATOR_SYSTEM	Data Element	09/02/94
SCHEME REPORT	Data Flow	07/02/94
SCHEME-REPORT	Data Element	17/02/94
SECTIONS&DETAILS *CLADDING-DETAILS+* *GLAZING-DETAILS+* *BLOCK-WORKS-DETAILS+* *DRY-PARTITIONS-DETAILS+* *DOOR-SCHEDULES+* *COLOUR-SCHEDULES+* *FALSE-CEILING-DETAILS*	Data Flow	24/02/94
SECURITY_SYSTEM	Data Element	09/02/94
SERV REQ BY ARCH *HANDDRYERS-LOC+* *DRINK-WAT-PTS+* *KITCH-FAC-LOC*	Data Flow	21/02/94
SERV-DES-DOC *SERV-DES-DRAWINGS+* *SERV-SPECS*	Data Structure	21/02/94
SERV-DES-DRAWINGS	Data Element	21/02/94
SERV-SPECS	Data Element	21/02/94
SERVICESDIST *LIGHTING-FIXT-DIST+* *POWER-SUPPLY-DIST+* *SPEAKERS-DIST+* *SPRINKLERS-DIST+* *TEL-POINTS-DIST*	Data Flow	21/02/94
SERV_COST_INF	Data Element	09/02/94
SITE-ACCESS	Data Element	21/02/94
SITE_BOUNDARIES	Data Element	08/02/94
SITE_LAYOUT	Data Element	09/02/94
SITE_LOCATION	Data Element	08/02/94
SMOKE_VENTILATORS	Data Element	09/02/94

All Definitions

Name Description	Type	Date
SPEAKERS-DIST	Data Element	21/02/94
SPECIAL SYSTEMS *TELECOM_AND_DATA+* *FIRE_ALARM_AND_DETECTION+* *SECURITY_SYSTEM+* *LIGHTNING_PROTECTION*	Data Flow	09/02/94
SPRINKLERS-DIST	Data Element	21/02/94
SPRINK_TANK	Data Element	09/02/94
STR-DES-DOC *STR-DES-DRAWINGS+* *STR-SPECS*	Data Structure	21/02/94
STR-DES-DRAWINGS	Data Element	21/02/94
STR-SPECS	Data Element	21/02/94
STR_COST_INF	Data Element	09/02/94
TEL-POINTS-DIST	Data Element	21/02/94
TELECOM_AND_DATA	Data Element	09/02/94
TENDER_POLICY	Data Element	08/02/94
VACUUM_PLANT	Data Element	09/02/94
WARM_AIR_SYSTEM	Data Element	09/02/94

Process Inputs

Process Name
    Input

ANCILLARY BUILD DESIGN
    REQ FOR PLANT ROOMS
    SITE SURVEY
ARCH COORDINATION
    APP-COM-ARCH-DES
    APP-COST-PLAN
    APP-PROG
    FINISHES SCHEDULES
    LANDSCAPING DRAWINGS
    PLANS&ELEVATIONS
    SECTIONS&DETAILS
    UNCOORDINATED SPECS
ARCHITECTURAL DESIGN
    FINAL PROD&OFF AREAS REQ
    BUILDING REGULATIONS
    PLANNING CONSTRAINTS
    ARCH SPECS
    LEVEL OF BRICK WALLS
    STEEL SKELETON LAYOUT&DET
    BEARING WALLS LOC
    SIZE&LOC OF PLANT ROOMS
    MANHOLES LOC&SIZES
    LIGHTED AREAS
    LIGHTING FIXTURES
    PENETRSCHED
    AHU LOC
    TOPSOILDEPTH
    FULL HEIGHTPARTITIONS
    APP-COST-PLAN
    APP-PROG
    APP-SCHEME-ARCH-DES
    APP-COM-ARCH-DES
    LOUVERSLOC
    SERVICESDIST
    ROADS LAYOUT
    FOOTPATHS MAT
    SITE LEVELS
ARRANGEMENT OF SPACES
    SITE LAYOUT
CLADDING SCHEME DESIGN
    CLADDING MATERIALS
DECIDE ON FINISHES &MATERIALS
    FIRE RATING REQ
    APP-ENVELOPE-MAT
    APP-SCHEME-FINISHES
DEVELOPED DESIGN
    APP SCHEME REPORT
    APPROVALS AND COMMENTS
    SITE SURVEY
DEVELOPING PLANS&ELEVATIONS
    SERVICESDIST
    SIZE&LOC OF PLANT ROOMS

Process Inputs

Process Name
Input

MANHOLES LOC&SIZES
AHU LOC
FINAL PROD&OFF AREAS REQ
PLANNING CONSTRAINTS
STEEL SKELETON LAYOUT&DET
FOOTPATHS MAT
LANDSCAPED AREAS
FIRE ESCAPE LOC
FIRE WALLS LOC
FIRE DOORS LOC
APP-SCHEME-LAYOUTS
DEVELOPING SECTIONS& DETAILS
LOUVERSLOC
FULL HEIGHTPARTITIONS
LIGHTED AREAS
LIGHTING FIXTURES
PENETRSCHED
LEVEL OF BRICK WALLS
BEARING WALLS LOC
ROOF PATTERN
ROOF LIGHT
APP-CLADDING-SCHEME
FINISHES
PLANS
DRAINAGE DESIGN
FINISHED LEVELS
DEVELOPED LAYOUTS
GUTTERS DETAILS

APP-SCHEME-DRAIN-DES
APP-PROG
DRAINAGEDISCHARGEQUALITY
DRAINAGE APPROVALS
ESTABLISH FIRE RATING REQ
BUILDING REGULATIONS
ESTIMATING COSTS
PRICES
BUDGET
BUILDING COST INFORMATION
FACTORY DESIGN
APPROVALS AND COMMENTS
PROJECT BRIEF
FLOOR SLAB DESIGN
APP-FLOOR-SLAB-SCHEME
FOUNDATIONS DESIGN
APP-FOUND-SCHEME
WATER TABLE
SOIL CONDITIONS
BEARING WALLS LOADS
SUPERSTR LOADS
FLOOR SLAB LOADS
SITE SURVEY

Process Inputs

Process Name
    Input

INTERNAL ROADS DESIGN
    ACCESS FOR MACH VEH

    APP-SCHEME-SITE-LAYOUT
    APP-COST-PLAN
    APPROVALS
    SOIL CONDITIONS
LANDSCAPING DESIGN
    TOPSOILDEPTH
    ROADS LAYOUT
    SITE LEVELS
MATERIALS SELECTION
    FINISHES MATERIALS
    SCHEME PLANS
    FINISHES_STANDARDS
ROOF DESIGN
    APP-ROOF-SCHEME
ROOF SCHEME DESIGN
    ROOF TYPE
SCHEME ARCH DESIGN
    SITE LAYOUT
    FINISHES_STANDARDS
    FEEDBACK
    FEEDBACK
SCHEME BUILDING DESIGN
    FINISHES_STANDARDS
    SITE LAYOUT
    FL_SLAB_UNIF_LOAD
    ASSUMED_GROUND_COND
SCHEME DESIGN
    APPROVALS AND COMMENTS
    PROJECT BRIEF
    PLANNING REGULATIONS
    SITE SURVEY
SCHEME DRAINAGE DESIGN
    SITE LAYOUT
    EXIST_SEWER_FACIL

SCHEME FLOOR SLAB DESIGN
    FL_SLAB_UNIF_LOAD
    LAYOUTS
SCHEME FOUNDATIONS DESIGN
    ASSUMED_GROUND_COND
SCHEME REPORT PRODUCTION
    SITE ACCESS
    PARKING FACILITIES
    LOADING&UNLOADINGFACILITIES
    SITE LAYOUT
    FOUL WATER DRAIN SYS
    SURFACE WATER DRAIN SYS

    POWER&LIGHT REQ

Process Inputs

Process Name
    Input

    HEATING&PIPED SERV
    SPECIAL SYSTEMS
    VENT SYSTEM
    ESTIMATED SLAB THICK&STRENGTH
    SKELETON MATERIAL
    ASSUMED FOUNDATIONS
    STRUCTURAL SYSTEM
    CLADDING REQ
    ELEVATIONS
    ADMIN AREA LAYOUT
    PROD AREA LAYOUT
    HANDINGOVER_DATE
    PHASING_REQ
    MECH ROOMS
    FINISHESOUTLINES
    TENDER_POLICY
    COST PLAN
    APPROVALS AND COMMENTS
SCHEME SERV DESIGN
    LAYOUTS
SCHEME STR DESIGN
    FL_SLAB_UNIF_LOAD
    ASSUMED_GROUND_COND
    LAYOUTS
    REQ FOR MECH ROOMS
    ROOF SCHEME
SCHEME SUPERSTR DESIGN
    REQ FOR MECH ROOMS
    ROOF SCHEME
    LAYOUTS
SERVICES DESIGN
    DEVELOPED LAYOUTS
    SMOKE DETECT&VENT REQ
    SERV SPECS
    APP-PROG
    APP-COST-PLAN
    APP-SCHEME-SERV-DES
    SERVREQ BYARCH
SITE PLANNING
    PLANNING REGULATIONS
    SITE_LOCATION
    EXISTING_DEVELOPMENTS
    SITE_BOUNDARIES
    SITE SURVEY
SPECS COORDINATION
    PROJ SERVSPECS
    PROJ STRSPECS
    PROJ ARCH SPECS
SPECS PRODUCTION
    ARCH SPECS
    FINISHES&MATERIALS
STEEL FRAME DESIGN

Process Inputs

Process Name
    Input

    APP-SUPERSTR-SCHEME
    GUTTERS DETAILS
    ROOF SECTIONS
    CLEAR HEIGHTS
STR DESIGN CHECKS
    UNCHECKED STR DES
STRUCTURAL COORDINATION
    APP-PROG
    APP-COST-PLAN
    DEVELOPED LAYOUTS
    STR SPECS
    APP-SCHEME-STR-DES
    APP-COM-STR-DES
    FOUND DES DOC
    ANC BUILD DRWGS
    FRAME DES DOC
    BW DETAILS
    COMMENTS&REQ MOD
STRUCTURAL DESIGN
    APP-SCHEME-STR-DES
    APP-COM-STR-DES
    SITE SURVEY
    GUTTERS DETAILS
    ROOF SECTIONS
    CLEAR HEIGHTS
    DEVELOPED LAYOUTS
    STR SPECS
    WATER TABLE
    SOIL CONDITIONS
    APP-PROG
    APP-COST-PLAN
    REQ FOR PLANT ROOMS

APPENDIX IV

A DETAILED DESCRIPTION OF THE SIMULATION MODEL

DETAILS OF EACH MODULE WITHIN THE GENETIK DESIGN SIMULATION MODEL

Tables

In the Tables module the maximum number of rows for the table are identified by the user. The minimum number of columns is a default of Genetik and includes:

- A column defining the number of each row. (This column cannot be edited)
- A column defining the ID of every row in the sequence each row is edited. (This column cannot be edited)
- A column defining the index of every row. The rows are ordered automatically in ascending order with regards to the index of each row. The index column in the tables provides flexibility in manipulating the data especially when cross referencing data in different tables and performing certain tasks on a previously defined range in a table.

The remaining columns in each table are defined by the user who has to specify the type of variable the column will handle e.g. integer, real, text.
The tables included in the Tables module are:

TNODE

A table containing information about design tasks that will be input by the user in addition to other information that will be generated as the simulation runs. The input information is:

- Task number
- Task icon (defined in the Icon Module)
- X co-ordinate for the icon on the screen
- Y co-ordinate for the icon on the screen
- Task duration
- Whether the task is iterative or non iterative

The information that will be generated for every design task includes:

- the start time
- the time for the end of first iteration for iterative tasks
- the start time of second iteration for iterative tasks
- the time at which the task will end
- if the task is conditional or non conditional

- the period of lapsed time which represents the delay of starting the task if a gate(s) is closed.
- the sampled duration of the design task in case of running the simulation stochastically
- the state of the task as the model runs:
  state 1: task is ready to start
  state 2: conditional tasks in loops re-start after receiving unconditional information
  state 3: end of first iteration for tasks on conditional basis
  state 4: end of first iteration for tasks
  state 7: non iterative tasks re-start after receiving unconditional information
  state 8: non iterative tasks end on conditional basis
  state 10: end of all activities done by a task
- Tags for design tasks whenever certain conditions or constraints are exercised.

<u>TSOURCE</u>

A table containing a list of different external sources of information that influence design tasks. These include:

- Client
- Planning authorities
- Building control officer
- Insurers
- Local authorities
- National rivers authority
- Highways authority

<u>TLINK</u>

A table containing information, which is input by the programmer, about links between non iterative tasks and iterative tasks. Those links are identified from the data flow model. This information includes:

- Task(s) that provide information to every design task
- The quality attribute for every information link (on a scale from 0 to 100)
- The status of the link switch (on or off)
- The status of the link gate (opened or closed)
- The period of time lapsed between finishing a task and releasing information
- The percentage of information required by a certain task from other task(s) to enable it to start.

- Information about conditional links that will be generated as the simulation runs depending on the task status (conditional or unconditional)
- Tags for links whenever certain conditions or constraints are exercised.

TLOOPDEF

This table defines the design tasks contained in each loop. These tasks are identified from the matrix analysis. The user may nominate the task that initiates every loop by assigning the integer (1) to this task in the relevant column. Alternatively, the model selects a task in the loop where all conditions to initiate the loop have been fulfilled. Information about this task will be generated in the relevant column as the model runs. Additionally, information about the duration of every task is generated for every iteration.

TLOOP

This table contains information about the different loops. The data input by the user is the duration of the second iteration of each loop. Other information about loops is generated as the simulation runs. This information includes:

- The state of every loop which is the same as that of tasks comprising the loop.
- The time at which first iteration starts and ends
- The duration of first iteration
- The time lapse representing the delay of starting a loop if a gate is closed.
- The time lapse representing the delay of starting second iteration of a loop if a gate is closed
- The time at which second iteration starts
- The status of the loop (conditional or unconditional).

COMPRES

A table containing information about different types of resources that will be input by the user. This information includes:

- The categories of each resource in the design firm represented by a table variable of the corresponding entity.
- The number of each employee for each category.
- The description of each resource category.
- The number of each resource category that will be allocated to the project to be simulated. This number will initialise the "idle" LIST of each resource type.
- Table variables representing the "idle" LISTS and "busy" LISTS for each type of resource.

- Table variables for each resource type that encapsulates the data required to draw histograms for resource utilisation for each resource category.
- Definitions of SCREENS and PICTURES corresponding to every resource category on which the relevant histograms are displayed.

TRES

A table containing information about resource requirements for each design task. This information are input by the user and can be amended through the results menu.

DES1, DRFT1, MANG1, DES2, DRFT2, MANG2, ENG2, DES3, DRFT3, MANG3, ENG3

Eleven tables which automatically capture the data required to plot the histograms for each resource type. This data includes the simulation time and the number of "busy" resources as the simulation clock advances.

RESLOOPS

A table used by the UTILITY CHEKRESL (explained later in this document) to capture data about required total number of resources for each resource type for a given iterative loop.

EVENTS

This table is filled with information about events which are executed as the simulation runs (typical events are explained later in this document). At the end of each simulation run the data in the table are deleted.

M-MAIN

This table includes information about items from the results main menu bar. The structure of the results main menu is :

FILE	EDIT	LOOPS	MODE	SIMULATE	REPORT
Finish	Processes	Any task ready	Deterministic	Initialise	Processes
Bar chart	Resources	Nominated task	Stochastic	Simulate	
Resources/ Toggle screen		Longest duration		Trace	
Histograms					

M_FILE, M_EDIT, M_LOOPS, M_MODE, M_SIM, M_REP

Six tables containing actions to be taken according to the user's selection from the results menu bar. The different actions resulting from the selection of each of these commands will be described later in this document.

TRPHDR

A temporary table for report headers, e.g. model name

TRPTXT

A temporary table that holds information for the report text. As the simulation runs, data about design tasks are imported from other tables by means of an action ARPNODE, to table TRPTXT showing tasks start and end times, timing for different iterations, sampled durations in case of stochastic mode, etc.

ENTITIES

ENTITIES are special types of tables. In addition to the default columns previously described in the TABLES module, the minimum number of columns in ENTITIES include a column for the description of each entity and a column for the colour of each entity. The data in these two columns is used when displaying Entities on the screen. Within the context of this model, Entities are used to represent the different types of resources involved in the design process.

The Entities included in the ENTITIES module include:

DESARCH

An Entity containing information about architects.

DESCIVST

An Entity containing information about civil/structures designers.

DESME

An Entity containing information about mechanical and electrical designers.

DRFTARCH

An Entity containing information about architectural draftsmen.

DRFTCVST

An Entity containing information about civil/structures draftsmen.

DRFTME

An Entity containing information about mechanical/electrical draftsmen.

ENGCVST

An Entity containing information about civil/structures engineers.

ENGME

An Entity containing information about mechanical/electrical engineers.

MANGARCH

An Entity containing information about architectural design managers.

MANGCVST

An Entity containing information about civil/structures design managers.

MANGME

An Entity containing information about mechanical/electrical design managers.

LISTS

LISTS are also special types of Tables. In addition to the default columns previously described in the TABLES module there is one additional column in each LIST representing a row pointer in the ENTITY of which this LIST is a member. In other words, every row in a LIST should be a member of a pre-defined Entity. There are twenty two Lists in the LISTS module representing the "idle" state and the "busy" state for each of the eleven previously mentioned entities. These include DES1BUS, DES1IDL, DRFT1BUS, DRFT1IDL, MANG1BUS, MANG1IDL, DES2BUS, DES2IDL, DRFT2BUS, DRFT2IDL, ENG2BUS, ENG2IDL, MANG2BUS, MANG2IDL, DES3BUS, DES3IDL, DRFT3BUS, DRFT3IDL, ENG3BUS, ENG3IDL, MANG3BUS and MANG3IDL.

C-EVENTS

C-Events are routines written in Genetik code. The occurrence of these events depends on specific conditions within the simulation being satisfied. For every C-Event, an associated B-Event is scheduled with a certain duration. B-Events are also routines written in Genetik code and, within the context of this model, are mainly related to changing the state of design tasks and calculating their start and end times. The C-Events within the simulation model are:

CDEPEND

An event that starts any design task not in an iterative loop upon receipt of the necessary information from other tasks and the availability of requisite resources. The start of the task may be conditional or non conditional according to its switch status or the status of the switches of its predecessors. Quality of information condition and gates status are also checked. The B-Event BEND is scheduled, on satisfaction of all conditions, with a duration equal to the design task duration.

CDEPENDN

This routine is the same as CDEPEND but is used when the simulation is run in stochastic mode. The design task duration is sampled from a normal distribution with a mean equal to its deterministic value. However, there is the possibility of applying other distributions such as triangular, binomial, etc. The associated B-Event is BENDST.

CSTPH1NP

A C-Event that starts the first iteration of a loop providing that the resources required for all design tasks within that loop are available and that any design task within the loops satisfies all conditions. These conditions include dependency, quality , gates status, and switches status. The duration of the first iteration is equal to the duration of that task. The associated B-Event is BENDPH1N.

CSTP1NPL

This event is identical to CSTPH1NP but the duration of the first iteration is equal to the longest task duration within the loop.

CSTPH1SN

This event is identical to CSTPH1NP but takes place when the simulation is running in stochastic mode.

CSTP1SNL

This event is identical to CSTP1NPL but takes place when the simulation is running in stochastic mode.

CSTPH1

A C-Event that starts first iteration of a loop if a design task nominated by the user satisfies all conditions including dependency, quality, gates status and switches status. The resources required for all design tasks within that loop should be also available. The associated B-Event is BENDPH1N.

CSTPH1ST

This event is identical to CSTPH1 but takes place when the simulation is running in stochastic mode.

On occurrence of any of the previously mentioned C-Events, the state of the relevant design task(s) changes to (1), and the relevant resources change from "idle" to "busy".

CCOND

An event that re-starts a conditional design task that is not in an iterative loop on receipt of all requisite information and availability of all resources, and providing all the conditions of its predecessors are satisfied. The associated B-Event is BENDCON with a duration equal to the original duration of the design task reduced by a certain factor specified by the user before running the simulation. The minimum total duration of any design task should not be less than the duration specified by the user. On occurrence of this event, the task state will change to 7 and the relevant resources state changes from "idle" to "busy". Within the same C-Event, there are different rules in case of running the simulation in stochastic mode.

CCONDL

Same as CCOND but for tasks in iterative loops. The associated B-Event is BENDP1CON with a duration equal to the original duration of the design task that initiates the loop reduced by a certain factor specified by the user before running the simulation. On occurrence of this event, the task state will change to 2 and the relevant resources will change to "busy".

CCONDLNG

This event is identical to CCONDL but takes place if the user selects from the results menu the duration of the first iteration to be the longest task duration within the loop.

CGATE1

A C-Event that schedules the time lapsed between finishing a design task and starting a dependant design task in the situation of a closed `gate` on their link. If a task requires information from more than one task where closed gates exist on their links this task will commence after considering the time lapse which starts it at the latest time. The associated B-Event is BGATE. This C-Event is applicable in case of tasks which are not in iterative loops or tasks initiating first iteration of loops.

CLOOP2

A C-Event that schedules the time lapsed before starting second iterations of loops. The associated B-Event is BLOOP2.

CSTARTGA

A C-Event that starts non loop design tasks or tasks which initiate first iteration of loops and which their start time have been delayed due to existence of closed gates. Other conditions like resource availability, information quality and switch status are also checked. The associated B-Event is BEND in case of non loop design tasks or BENDPH1N in case of tasks which initiate first iteration of loops. Within the same C-Event there are different rules in case of running the simulation in stochastic mode. On occurrence of this event, the state of the relevant design task changes to (1), and the state of the relevant resources changes from "idle" to "busy".

CSTARGAL

This event is identical to CSTARTGA but takes place if the user selects from the results menu the duration of the first iteration of a loop to be the longest task duration within the loop.

CPHASE

A C-Event that phases the release of information from a task to a subsequent task(s). This figure is based on the percentage of information required by each task from other task(s) to start which is specified by the user. A task will release information to a dependant one after a time equal to a percentage of its duration which is proportional with the same percentage of requisite information. The associated B-Event is BPHASE.

CSTPH22

A C-Event that starts the second iteration of loops providing that all design tasks within the loop satisfy all conditions including dependency, quality, gates status and resource availability. The associated B-Event is BENDPH2. On occurrence of this event the state of the loop changes to 5, and the state of the relevant resources changes to "busy".

CPH2GATE

A C-Event that starts second iteration of loop tasks which their start have been delayed due to the existence of closed gates. The associated B-Event is BENDPH2. On occurrence of this event the state of the loop changes to 5, and the state of the relevant resources changes to "busy".

B-EVENTS

B-Events are routines written in Genetik code and are scheduled when associated C-Events occur. The B-Events, within the context of this model, are mainly related to changing the state of design tasks and calculating the start and end times of design tasks. The B-Events within the simulation model are:

BEND

An event associated with the C-Events CDEPEND and CSTARTGA. On occurrence of this event, the state of design tasks changes to 8 if the task ends on conditional basis, otherwise it changes to 10 and the state of the relevant resources changes from "busy" to "idle".

BENDST

This event is identical to BEND which occurs when the simulation is run in stochastic mode.

BENDCON

A B-Event associated with the C-Event CCOND. On occurrence of this event the state of conditional design tasks changes to 10 and the `conditional` tag disappears. The state of the relevant resources changes to "idle".

BENDPH1N

A B-Event associated with C-Events CSTPH1NP, CSTPH1SN, CSTPH1, CSTP1NPL, CSTPH1ST, CSTP1SNL , CSTARTGA, and CSTARGAL. If loop design tasks are running conditionally, then their state will change to 3, otherwise the state of loop tasks will change to 4. The state of the relevant resources changes to "idle".

BNDP1CON

A B-Event which is scheduled on occurrence of C-Event CCONDL or CCONDLNG. The state of loop tasks changes to 4 and `Conditional` tag disappears. The state of the relevant resources changes to "idle".

BGATE

A B-Event associated with C-Event CGATE1.

BLOOP2

A B-Event associated with C-Event CLOOP2.

BPHASE

A B-event associated with the C-event CPHASE.

BENDPH2

A B-Event which is scheduled on occurrence of C-events CSTPH22 or CPH2GATE. The state of loop tasks changes to 10 and the state of the relevant resources changes to "idle".

ACTIONS

ACTIONS are routines written in Genetik code which perform certain tasks during running the simulation whenever they are called by other modules within the model. The actions developed by the writer for use within the simulation model are:

Actions called from the Results Menu Bar:

INITLZ

Is the action called when the user selects INITIALISE from the SIMULATE menu. This action should be taken before running any simulation to reset the values of all variables and dispose the results from previous runs.

EXECUTE

Is the action called when the user selects SIMULATE from the SIMULATE menu. It is the action that runs the simulation by selecting the appropriate events as the simulation clock advances.

ATRACE

Is the action taken if the user selects TRACE from the SIMULATE menu. This action stops the simulation run after every B-Event is scheduled, as the simulation clock advances, until the user clicks the mouse or presses a key. This action allows the user to follow the changes that occur to every design task as the simulation runs.

DETERM

This action is taken when the user selects DETERMINISTIC from the MODE menu. It allows the model to run in deterministic mode by selecting the deterministic events by means of action variables MODE and MODEPH1.

STOCH

This action is taken when the user selects STOCHASTIC from the MODE menu. It allows the model to run in stochastic mode by selecting the stochastic events by means of the action variables MODE and MODEPH1.

ANYTASK

This action is taken when the user selects ANY TASK READY from the LOOPS menu. It allows the model to select the duration of the first iteration of loops to be the duration of the task that initiates the loop. This is done by the model activating the relevant events by means of action variables MODEPH1, MODECOND, and MODEGATE. This can be undertaken in either stochastic or deterministic mode according to the user's choice from the MODE menu.

NOMIN

This action is taken when the user selects NOMINATED TASK from the LOOPS menu. It allows the model to select the duration of the first iteration of loops to be the duration of the task nominated by the user which initiates the loop. This is done by the model activating the relevant events by means of action variables MODEPH1, MODECOND, and MODEGATE. This can be undertaken in either stochastic or deterministic mode according to the user's choice from the MODE menu.

LONGST

This action is taken when the user selects LONGEST DURATION from the LOOPS menu. It allows the model to select the duration of the first iteration of loops to be the longest task duration within the loop. This is done by the model activating the relevant events by means of action variables MODEPH1, MODECOND, and MODEGATE.

This can be undertaken in either stochastic or deterministic mode according to the user's choice from the MODE menu.

FINISH

Is the action taken when the user selects FINISH from the FILE menu. It allows the user to exit from the main results menu.

BARCH

Is the action taken when the user selects BAR CHART from the FILE menu. It displays the results of running the simulation in a bar chart format showing the start and end times of every design task, and the start and end times of every iteration for loop tasks with different colours representing different iterations. Where applicable, tasks performed on conditional basis are also illustrated on the bar chart. If the size of the bar chart is greater than the physical size of the screen, there is the facility of scrolling the screen, up or down, by means of a scroll bar.

RESMODE

Is the action taken when the user selects RESOURCES/TOGGLE SCREEN from the FILE menu. It displays the twenty two lists of the eleven resource types each in the "busy" state and the "idle" state with different colours representing different disciplines. When the simulation runs, different resources move between their relevant "idle" and "busy" lists according to their state at every simulation time. This action also allows toggling the screen with the processes icons screen.

HISTRES1

Is the action taken when the user selects HISTOGRAMS from the FILE menu. This action displays the results of the resource utilisation throughout the whole design process in histogram format. A histogram is displayed for every resource type each in a separate screen.

ARPNODE

Is the action taken when the user selects PROCESSES from the REPORT menu. This action imports data about the model and the design tasks, as the simulation runs, and displays it in the temporary tables TRPHDR and TRPTXT. This data includes model name, task durations, start and end times for every task, timing for different iterations, sampled durations in case of stochastic mode, etc. There is the possibility of printing this report to a printer or a file.

AEDNODE

Is the action taken when the user selects PROCESSES from the EDIT menu. It allows the user to edit or amend tasks durations which will be transferred automatically to the table TNODE.

AEDRES

Is the action taken when the user selects RESOURCES from the EDIT menu. It allows the user to edit or amend the type and/or number of resources required for each design task. This data is transferred automatically to the table TRES.

Other ACTIONS within the Simulation Model:

CLIST

An action containing a list of all the C-Events that will be called as the simulation runs.

DRAWNODE

An action that draws the icons representing different design tasks.

QUALMES

An action that displays a message of insufficient information quality if the simulation stops due to finding a link with a quality attribute less than the cut-off quality value.

STARTUP

STARTUP is the essential ACTION UNIT required for the model to run. It displays the `simulation run` screen with the results main menu.

SWITCH2

Is the action that tags all tasks with an `off` link switch with `C` (for conditional). It tags also with `C` all tasks that start based on information from conditional tasks.

UTILITIES

Utilities are modules which when called, return certain parameters based on other given parameters. In addition to over 350 Genetik built in utilities, other utilities have been written in Genetik code to perform certain functions. The utilities developed by the writer are:

<u>CHECKDEP</u>

A utility to check the dependency of any design task on other tasks. If a task is found to be dependant on other task(s), then it will not start until the predecessor(s) have been completed (except in cases of `switches` or phased release of information). This utility is applicable in cases of starting non loop tasks or tasks that initiate first iteration of loops.

<u>CHEKDEP2</u>

As CHECKDEP but checks the dependency of loop tasks before starting second iteration.

<u>CHECKPRB</u>

A utility to test the information quality condition on every link between different design tasks and between any design task and an external source of information (e.g. Client, highways authority, etc.). The test will fail if a quality attribute on a link is less than the cut-off value defined by the user, and the associated design task will not be performed.

<u>CHECKRES</u>

A utility that checks the resources availability for each design task before its commencement. If the requisite resources are not available for a certain design task, this task will not start until other design task(s) are completed and release their resources. A message appears on the screen if the commencement of a task is delayed due to lack of resources. If the resources available are still not adequate, the simulation stops with a message identifying the task(s) with insufficient resources.

<u>CHEKRESL</u>

A utility that checks the resources availability for <u>all</u> design tasks comprising a loop before this loop is initialised. If the requisite resources are not available for a certain loop, this loop will not start until other design task(s) are completed and release their resources. A message appears on the screen if the commencement of a loop is delayed due to lack of resources. If the resources available are still not adequate, the simulation stops with a message identifying the loop(s) with insufficient resources.

ADRESBUS

A utility that adds to the "busy" list of each resource type the number of relevant resources who are occupied at each simulation time as the simulation clock advances.

ADRESIDL

A utility that adds to the "idle" list of each resource type the number of relevant resources who are idle at each simulation time as the simulation clock advances.

ADREBSL

A utility that adds to the "busy" list of each resource type the total number of relevant resources of a loop occupied at each simulation time as the simulation clock advances.

ADREIDP

A utility that adds to the "idle" list of each resource type the total number of relevant resources of a loop who become idle at each simulation time as the simulation clock advances.

UDWBUTTN

A utility to draw button icons for design tasks with the task number displayed on it.

UFDINRW

A utility to cross reference integer values between different tables. On retrieving the required integer value (mainly used for the index column in tables), a table row pointer can call any other value in the same row.

UFDTXROW

A utility to cross reference text values between different tables. On retrieving the required text values, (mainly used for Process names columns in tables), a table row pointer can call any other value in the same row.

INTEGER VARIABLES

In addition to local integer variables within every action, C-Event and B-Event, there are global integer variables within the whole model that can be called from any module. The most important global integer variables are:

PCCRIT

Should be specified by the user and represents the cut-off value for the information quality which should be less than or equal the quality attribute on any link of any design task for that task to be performed.

SCALE

Represents the scale of the bar chart.

SIMTIME

Represents the simulation time.

REAL VARIABLES

TIMEC

A global real variable that represents the reduction factor of tasks duration after a conditional task receives its requisite information. This variable should be specified by the user.

ACTION VARIABLES

MODE

An action variable that selects the relevant C-Events when running the simulation in deterministic mode or stochastic mode according to the user's choice. It is applicable for non loop tasks.

MODEPH1

An action variable that selects the relevant C-Events when running the simulation in deterministic mode or stochastic mode according to the user's choice. It is applicable for loop tasks. The duration of the first iteration is selected according to the user's choice from the LOOPS menu.

MODECOND

An action variable that selects the relevant C-Events for the duration of the first iteration of conditional loop design tasks.

MODEGATE

An action variable that selects the relevant C-Events for the duration of the first iteration of loop design tasks after passing through a 'closed gate'.

SCREEN

Screen is a module within Genetik to define different screen layouts and colours while running the model. The screens defined within the built model are:

SCREEN1

A definition for the screen where the main menu, simulation time and processes icons are displayed.

SCREEN2

A definition for the screen where the bar chart appears.

SCREEN3

A definition for the screen where the resources lists appear each in the "idle" state and the "busy" state with different colours representing different disciplines.

SCRDES1, SCRDRFT1, SCRMANG1, SCRDES2, SCRDRFT2, SCRENG2, SCRMANG2, SCRDES3, SCRDRFT3, SCRENG3, SCRMANG3

Definitions for eleven screens where results of the simulation run are displayed in resource utilisation histogram format for each resource type.

The layout and colours of every screen is defined by assembling different WINDOWS on different PICTURES, which represent other modules within Genetik. SCREEN1 is comprised of a WINDOW containing the results menu (developed also by the writer), a second WINDOW displaying the current simulation time, and a third blank WINDOW where pre-defined ICONS of design tasks will be displayed when the model runs.

SCREEN2 is comprised of a blank WINDOW where a bar chart representing a schedule of the design tasks will be displayed whenever the appropriate action is invoked from the results menu.

SCREEN3 is comprised of a WINDOW containing the results menu, a WINDOW displaying the current simulation time, and a third WINDOW displaying the resources lists.

The remaining eleven screens each is comprised of a blank window where a histogram of the resource utilisation for each resource type is displayed whenever the appropriate action is invoked from the results menu.

ICON

Icon is a module within Genetik used to define shapes and colours of icons. The icon defined within the built model is PROC which represents an icon for each design task.

APPENDIX V

EXAMPLES OF DATA INPUT AND SOME ROUTINES OF THE SIMULATION MODEL DEVELOPED BY THE WRITER

GENETIK 9.11 library file created by THH at 12:1? on 11-Jul-95

Model name : DESIGN as 17+resources+sources+

File contents : The TABLE unit TLINK

```
--------------------------------------------------------------------------------------------------------------------------
ABLE  * TLINK      :
AX ROWS*           :      150
 TO VAR* SYSINDEX  :    4   Y
 XT VAR* PROCESS   :   10   N
 TG VAR* PCPROB    :    4   N
 XT VAR* SWITCH    :    3   N
 XT VAR* GATE      :    2   N
 XT VAR* COND      :    1   N
 TN VAR* LICON     :    2   N
 TO VAR* TIMELAPS  :    4   N
 TO VAR* PHRBLOK   :    4   N
 XT VAR* SWITCH1   :    3   N
 AL VAR* PERCINP   :    4   N
 XT VAR* PHASE     :    1   N
 TO VAR* TESTGATE  :    4   N
 30   2147483647 *          *      3 *   * *  * * *    ?    0    0 *   * 0 0?000E+00 * *    0
 96        112 *S1          *    1?0 *on * *  * * *    ?    0    0 *   * 0.0?000E+00 * *    0
 13        112 *P11114      *    1?0 *on * *  * * *    ?    0    0 *   * 0.0?000E+00 * *    0
 14        112 *P113        *    100 *on * *  * * *    ?    0    0 *   * 0.0?000E+00 * *    0
 ?7        113 *S2          *    1?0 *on * *  * * *    ?    0    0 *   * 0.0?000E+00 * *    0
 ?3        113 *S1          *    1?0 *on * *  * * *    ?    0    0 *   * 0.0?000E+00 * *    0
 36        113 *P116        *    1?0 *on * *  * * *    ?    0    0 *   * 0.0?000E+00 * *    0
 97        114 *S1          *    1?0 *on * *  * * *    ?    0    0 *   * 0.0?000E+00 * *    0
 ?1        114 *P112        *    1?0 *on * *  * * *    ?    0    0 *   * 0 0?000E+00 * *    0
 0?        114 *P111235     *    1?0 *on * *  * * *    ?    0    0 *   * 0.0?000E+00 * *    0
 ?09       114 *P111232     *    1?0 *on * *  * * *    ?    0    0 *   * 0.0?000E+00 * *    0
 ?15       114 *P115        *    1?? *on * *  * * *    ?    0    0 *   * 0 0?000E+00 * *    0
 10        114 *P1113       *    1?? *on * *  * * *    ?    0    0 *   * 0 0?000E+00 * *    0
 ?0        114 *P11121      *    1?? *on * *  * * *    ?    0    0 *   * 0 0??00E+00 * *    0
 ?9        114 *P11122      *    1?? *on * *  * * *    ?    0    0 *   * 0 0?000E+00 * *    0
 ??        114 *P11112      *    1?? *on * *  * * *    ?    0    0 *   * 0 0??00E+00 * *    0
 ??        114 *P11111      *    1?? *on * *  * * *    ?    0    0 *   * 0 0??00E+00 * *    0
```

0001	114	'P111111	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0018	114	'P111115	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0016	114	'P111114	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0012	114	'P113	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0093	115	'S1	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0076	115	'X	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0026	122	'P114	'	100 'on ' ' ' ' '	?	0	1 '	'	0.10000E+00 'Y'	0
00103	124	'S3	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
00102	124	'S4	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0034	124	'P114	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
00107	125	'S6	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
00106	125	'S3	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
00105	125	'S5	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0037	125	'P114	'	100 'on ' ' ' ' '	?	0	1 '	'	0.40000E+00 'Y'	0
00104	126	'S7	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0036	126	'P122	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0035	126	'P114	'	100 'on ' ' ' ' '	?	0	1 '	'	0.80000E+00 'Y'	0
0067	127	'P1215	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0066	127	'P124	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0065	127	'P1237	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0092	1113	'S1	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0075	1113	'X	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0033	1212	'P114	'	100 'on ' ' ' ' '	?	0	1 '	'	0.70000E+00 'Y'	0
0055	1214	'X	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0063	1215	'P12134	'	100 'on ' ' ' 'C'	?	0	0 '	'	0.00000E+00 ' '	0
0062	1215	'P1217	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0061	1215	'P12113	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0060	1215	'P12114	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0059	1215	'P12111	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0058	1215	'P1216	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0057	1215	'P1232	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0056	1215	'P114	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0084	1216	'X	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0046	1217	'P124	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0083	1218	'X	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0099	1231	'S3	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0098	1231	'S4	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0022	1231	'P114	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0101	1232	'S1	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0100	1232	'S2	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0029	1232	'P1231	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0028	1232	'P114	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0027	1233	'P122	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0024	1234	'P1231	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0023	1234	'P114	'	100 'on ' ' ' ' '	?	0	1 '	'	0.40000E+00 'Y'	0
0030	1235	'P114	'	100 'on ' ' ' ' '	?	0	1 '	'	0.60000E+00 'Y'	0
0032	1236	'P1234	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0031	1236	'P114	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0025	1237	'P1234	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0045	1238	'P1236	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0044	1238	'P1232	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0043	1238	'P1233	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0042	1238	'P1237	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0041	1238	'P1234	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0040	1238	'P114	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0090	11112	'S1	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0073	11112	'X	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0072	11113	'X	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0071	11114	'X	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	.
0002	11121	'P116	'	100 'on ' 'c1' ' '	?	3	0 '	'	0.00000E+00 ' '	4
0091	11122	'S1	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0074	11122	'X	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0085	12111	'X	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0
0086	12112	'X	'	100 'on ' ' ' ' '	?	0	0 '	'	0.00000E+00 ' '	0

```
........................................................................................................................................

........................................................................................................................................

   GENETIK  9.11 library file created by THR at 12:16 on 11-Jul-95

   Model name     : DESIGN     as 17+resources+sources

   File contents : The TABLE    unit TRES

........................................................................................................................................

*TABLE   * TRES      :                               0
<MAX ROWS>           :        100
<INTG VAR> SYSINDEX :    4   Y
<TEXT VAR> PROCESS  :    8   N
<INTG VAR> DUR      :    4   N
 TEXT VAR> CXLOOP   :    1   N
<TEXT VAR> RESTYPE  :    9   N
 INTG VAR> NUMB     :    4   N
00000  2147483647 *        *          0 * * *          *          0
00049        112 *P112     *          6 * * *DES2      *          2
 0025        112 *P112     *          6 * * *DRFT2     *          1
 0050        113 *P113     *          7 * * *DES1      *          3
 0027        113 *P113     *          7 * * *DRFT1     *          2
 0056        114 *P114     *         18 * * *DES1      *          1
 0055        114 *P114     *         18 * * *MANG1     *          1
 0054        114 *P114     *         18 * * *DES2      *          1
 0053        114 *P114     *         18 * * *MANG2     *          1
 0052        114 *P114     *         18 * * *ENG3      *          1
 0051        114 *P114     *         18 * * *MANG3     *          1
 0063        115 *P115     *          4 * * *DES1      *          1
 0062        115 *P115     *          4 * * *DES2      *          1
 0023        115 *P115     *          4 *Y* *DES3      *          1
 0024        116 *P116     *         12 * * *DES2      *          2
 0026        122 *P122     *         23 * * *DES2      *          2
 0041        122 *P122     *         23 * * *MANG2     *          1
 0066        124 *P124     *          6 * * *DES3      *          2
 0065        124 *P124     *          6 * * *ENG3      *          2
 0064        124 *P124     *          6 * * *DRFT3     *          2
 0043        124 *P124     *          6 *Y* *MANG3     *          1
00068        125 *P125     *          8 * * *DRFT2     *          1
00045        125 *P125     *          8 *Y* *DES2      *          1
00067        126 *P126     *          9 * * *DRFT2     *          2
00044        126 *P126     *          9 *Y* *DES2      *          1
00058        127 *P127     *         11 * * *DES1      *          1
 0057        127 *P127     *         11 * * *DES2      *          1
00042        127 *P127     *         11 * * *ENG3      *          1
00022       1113 *P1113    *         14 *Y* *ENG3      *          3
00069       1212 *P1212    *         16 * * *DES2      *          2
00040       1212 *P1212    *         16 *Y* *DRFT2     *          1
00010       1214 *P1214    *          5 *Y* *DES2      *          2
 0039       1215 *P1215    *          6 *Y* *MANG2     *          1
 0070       1216 *P1216    *          5 * * *DES2      *          1
 0012       1216 *P1216    *          5 *Y* *DRFT2     *          1
 0059       1217 *P1217    *          6 * * *DES1      *          2
 0011       1217 *P1217    *          6 * * *DRFT1     *          1
 0071       1218 *P1218    *          8 * * *DES2      *          2
 0009       1218 *P1218    *          8 *Y* *DRFT2     *          2
 0060       1231 *P1231    *         21 * * *DES1      *          1
 0008       1231 *P1231    *         21 * * *ENG3      *          1
 0073       1232 *P1232    *          6 * * *DES1      *          4
```

006	1212	"P1212 "	6	"Y"	"DRFT1 "	1
033	1233	"P1233 "	4	" "	"DES1 "	1
301	1233	"P1233 "	4	"Y"	"DRFT1 "	2
005	1234	"P1234 "	9	" "	"DES1 "	2
034	1235	"P1235 "	0	" "	"DES1 "	2
302	1235	"P1235 "	16	"Y"	"DRFT1 "	1
035	1236	"P1236 "	22	" "	"DES1 "	1
307	1236	"P1236 "	22	"Y"	"DRFT1 "	2
004	1237	"P1237 "	11	" "	"DES1 "	2
003	1238	"P1238 "	7	" "	"HANG1 "	1
319	11112	"P11112 "	12	"Y"	"DES1 "	1
076	11113	"P11113 "	12	" "	"DES1 "	1
318	11113	"P11113 "	12	"Y"	"DRFT1 "	1
077	11114	"P11114 "	4	" "	"DES1 "	1
017	11114	"P11114 "	5	"Y"	"DRFT1 "	1
321	11121	"P11121 "	11	"Y"	"DES2 "	1
020	11122	"P11122 "	5	"Y"	"DES2 "	1
037	12111	"P12111 "	7	"Y"	"DES2 "	1
318	12112	"P12112 "	4	"Y"	"DES2 "	1
078	12113	"P12113 "	9	" "	"DES2 "	1
314	12113	"P12113 "	9	"Y"	"DRFT2 "	1
379	12114	"P12114 "	8	" "	"DES2 "	2
313	12114	"P12114 "	8	"Y"	"DRFT2 "	1
133	12131	"P12131 "	14	" "	"DES2 "	1
034	12132	"P12132 "	6	" "	"DES2 "	1
035	12133	"P12133 "	9	" "	"DES2 "	1
161	12134	"P12134 "	13	" "	"DES2 "	2
136	12134	"P12134 "	13	" "	"DRFT2 "	2
116	111111	"P111111 "	17	"Y"	"DES1 "	2
780	111112	"P111112 "	18	" "	"DES1 "	1
015	111112	"P111112 "	18	"Y"	"DRFT1 "	1
348	111113	"P111113 "	7	"Y"	"DES1 "	1
081	111114	"P111114 "	4	" "	"DRFT1 "	1
046	111114	"P111114 "	4	"Y"	"DES1 "	1
082	111115	"P111115 "	13	" "	"DES1 "	1
047	111115	"P111115 "	13	"Y"	"DRFT1 "	1
028	111231	"P111231 "	7	"Y"	"DES1 "	1
029	111232	"P111232 "	8	"Y"	"DES1 "	1
032	111233	"P111233 "	9	"Y"	"DES2 "	1
030	111234	"P111234 "	6	"Y"	"DES2 "	1
031	111235	"P111235 "	15	"Y"	"DES2 "	1

```
*************************************************************************************************************

  GENETIK  9.11 library file created by THH at 11 59 on 11 Jul-95

  Model name     . DESIGN      as 17+resources+sources

  File contents . The ACTION   unit BARCH

*************************************************************************************************************

TION  * BARCH     : Draw bars with time disp on each bar act 0
EXT VAR. TASKS    :    8 L *          *
NTG VAR. NUMBER   :    0 L                48
NTG VAR. MAXIMUM  :    0 L               100
NTG VAR. COUNT    :    0 L                49
ION VAR. TRNODE   :    0 L TNODE    .00031
NTG VAR. YPOSN    :    0 L                 3
EAL VAR. LENBAR   :    0 L      11.25000
NTG VAR. XPOSN    :    0 L                 0
EAL VAR. XENDPH1  :    0 L       7.50000
EAL VAR. XPOSN1   :    0 L       4.75000
NTG VAR. XTIME    :    0 L                 0
EAL VAR. XSTPH2   :    0 L       7.50000
EAL VAR. XPO      :    0 L       4.75000
NTG VAR. LIKEY    :    0 L                27
NTG VAR. LIX      :    0 L                80
NTG VAR. LIY      :    0 L                23
NTG VAR. LIXOFF   :    0 L                 6
NTG VAR. LIYOFF   :    0 L                 1
NTG VAR. LIPOS    :    0 L                94
NTG VAR. LICLICK  :    0 L                 3
NTG VAR. POS      :    0 L                 0
NTG VAR. A        :    0 L               114
NTG VAR. C        :    0 L               147
NTG VAR. XENDCOND :    0 L                22
NTG VAR. XSTNOCON :    0 L                38
ARCH1     Begin
          PICTURE4
          SCREEN2
          Setwindw : WINDOW4
          Size : NUMBER : MAXIMUM : TNODE
          A = NUMBER*3-30
          C = A+33
          Setwind : 1 : A
          Displtln : "W" : 10 : C : 1 : 1 : 75 : C : 1 : 1
          Displtln : "W" : 10 : C : 1 : 1 : 10 : 1 : 1 : 1
                                                                      display labels
          Disptext : "CK" : 2 : 1 : "TASKS"
          Disptext : "CK"   70 : C+1 : TIME
          Scrldefn : "SBAR" : "VG" : 77 : 1 : 30 : 2 : "V"
          Scrllmit : "SBAR" : 1 : NUMBER*3+2 : 33 : "BOTTOM"
          Scrlset : "SBAR" : NUMBER*3-30
                                                                      get each process
          Size   NUMBER   MAXIMUM   TNODE
PROCBAR1  Loop for COUNT = 1 to NUMBER step 1
            Getrow : TRNODE   TNODE   COUNT
            LENBAR = TRNODE CTEND/SCALE
                                                                      display bar
```

```
| |               YPOSN = C+COUNT+1
| |               XPO = TRNODE.START/SCALE
| |               Disptext : "yK" : 2 : YPOSN : TRNODE.PROCESS
| | [NORMAL]      Begin if TRNODE.CIEND<>0
| | |               Displtln : "c" : 10+XPO : YPOSN : 2 : 1 : 10+LENBAR : YPOSN : 2 : 1
| | |             Else
| | [NORMAL]      End
| | [ENDPH1]      Begin if TRNODE.ENDPH1<>0
| | |               XENDPH1 = TRNODE.ENDPH1/SCALE
| | |               XPOSN1 = TRNODE.START/SCALE
| | |               Displtln : "c" : 10+XPOSN1 : YPOSN : 2 : 1 : 10+XENDPH1 : YPOSN : 2 : 1
| | |               XSTPH2 = TRNODE.STPH2/SCALE
| | | [PH1ONLY]     Begin if TRNODE.STPH2<>0
| | | |               Displtln : "K" : 10+XENDPH1 : YPOSN : 2 : 1 : 10+XSTPH2 : YPOSN : 2 : 1
| | | |             Else
| | | [PH1ONLY]     End
| | | [NOZERO]      Begin if TRNODE.ENDPH1<>0
| | | |               Dispint : "yK" : 10+XENDPH1 : YPOSN+2 : TRNODE.ENDPH1 : 3
| | | |             Else
| | | [NOZERO]      End
| | | [DISPSTP2]    Begin if TRNODE.ENDPH1<>TRNODE.STPH2
| | | |               Leave [DISPSTP2] if TRNODE.STPH2==0
| | | |               Dispint : "yK" : 10+XSTPH2 : YPOSN+2 : TRNODE.STPH2 : 3
| | | |             Else
| | | [DISPSTP2]    End
| | |             Else
| | [ENDPH1]      End
| |                                                              display start times
| | [DISP]        Begin if TRNODE.START<>0
| | |               Dispint : "yK" : 10+XPO : YPOSN+1 : TRNODE.START : 3
| | |             Else
| | [DISP]        End
| |                                                              display end times
| | [DISPEND]     Begin if TRNODE.CIEND<>0
| | |               Dispint : "yK" : 10+LENBAR : YPOSN+2 : TRNODE.CIEND : 3
| | |             Else
| | [DISPEND]     End
| | [COND]        Begin if TRNODE.ENDCOND<>0
| | |               XENDCOND = TRNODE.ENDCOND/SCALE
| | |               XSTNOCON = TRNODE.STNOCOND/SCALE
| | |               Displtln : "K" : 10+XENDCOND : YPOSN : 2 : 1 : 10+XSTNOCON : YPOSN : 2 : 1
| | |               Dispint : "yK" : 10+XENDCOND : YPOSN+2 : TRNODE.ENDCOND : 3
| | |               Dispint : "yK" : 10+XSTNOCON : YPOSN+1 : TRNODE.STNOCOND : 3
| | |               Disptext : "mK" : 13+XENDCOND : YPOSN+2 : "C"
| | |             Else
| | [COND]        End
| [PROCBAR]   Next
| [ASCROLL]   Begin
| | [SCROLL]      Loop
| | |               Getinput : LIKEY : LIX : LIY : LIXOFF : LIYOFF
| | |               Leave [SCROLL] if LIKEY==27
| | | [TEST1]       Begin if LIKEY==1|LIKEY==2
| | | |               Scrlget : "SBAR" : LIPOS : LICLICK : LIX : LIY : LIXOFF : LIYOFF
| | | |               Setwind : 1 : LIPOS
| | | |               Scrlset : "SBAR" : LIPOS
| | | | [AX]          Begin if LIPOS==NUMBER*3-30
| | | | |               Displtln : "W" : 10 : C : 1 : 1 : 75 : C : 1 : 1
| | | | |               Disptext : "cK" : 70 : C+1 : TIME
| | | | |             Else
| | | | [AX]          End
| | | |               Displtln : "W" : 10 : C : 1 : 1 : 10 : 1 : 1 : 1
| | | | [LAB]         Begin if LIPOS==1
| | | | |               Disptext : "cK" : 2 : LIPOS : "TASKS"
| | | | |             Else
```

```
| | | | {LAB}       End
| | | |             Enter [PROCHAR]
| | | |           Else
| | | {TEST1}     End
| | {SCROLL}    Next
| |             Setlend   "SHAR"    "KK"
| {ASCROLL}  End
{BARCH}    End
```

****

```
****************************************************************************************************

GENETIK  9.11 library file created by THH at 12:06 on 11-Jul-95

Model name      DESIGN     as 17+resources+sources

File contents   The UTILITY  unit CHECKDEP

****************************************************************************************************

:LITY * CHECKDEP   check dependency                        0
TG VAR> COUNT          0 P           0
ET VAR> LXFLAG         2 R "  "
TG VAR> NNODE          0 L           0
TG VAR> MAXIMUM        0 L           0
OM VAR> TRNODE         0 L     ?.00000
TG VAR> TRLINK     :   0 L           0
OM VAR> TRLNK          0 L     ?.00000
TG VAR> COUNT2     :   0 L           0
OM VAR> TROW       :   0 L     ?.00000
TG VAR> LIDUR      :   0 L           0
ET VAR> TXNODE     :  10 L "         "
XT VAR> OK         :   1 L " "
TG VAR> NUMBER     :   0 L           0
TG VAR> TIME1      :   0 L           0
TG VAR> TIME2      :   0 L           0
AL VAR> TIME4      :   0 L     0.00000
TG VAR> TIME5      :   0 L           0
OM VAR> TRDEF      :   0 L     ?.00000
ECKDEP] Begin
         LXFLAG = "Y"
         Setrange : TLINK : COUNT : COUNT
LINK]    Loop
           Getrange : TRLNK : COUNT2
           Leave [LINK] if COUNT2<=0
                                                                  check switch on or off
 [SWITCH1]  Begin if TRLNK.SWITCH=="off"
 |            Next [LINK]
 |          Else
 [SWITCH1]  End
                                                                  check if loop node independ on outer mod
 [INDEPEND] Begin if TRLNK.PROCESS=="X"
 | [PRIOR]     Begin if NODEPH1==CSTPH1|NODEPH1==CSTPH1ST
 | |             UPDINRW : TRLNK.SYSINDEX : TNODE : SYSINDEX : TROW : OK
 | |             UPDTXROW : TROW.PROCESS : TLOOPDEF : PROCESS : TRDEF : OK
 | | [PRIOR1]    Begin if OK=="Y"&TRDEF.PRIORITY==1
 | | |             LXFLAG = "Y"
 | | |             Leave [LINK]
 | | |           Else
 | | [PRIOR1]    End
 | |           Else
 | [PRIOR]     End
 |             LXFLAG = "N"
 |             Leave [LINK]
 |           Else
 [INDEPEND] End
                                                                  start loop with independent process
 [STINDEP]  Begin if TRLNK.PROCESS=="Y"
```

```
 |              LXFLAG = "Y"
 |              Leave {LINK}
 |            Else
{STINOEP}     End
              TXNODE = TRLNK.PROCESS
                                                                   Find Node or Source
              UPDTXROW : TXNODE : TNODE : PROCESS . TROW   OK
                                                                   check phased release
{PHASE}       Begin if TRLNK.PERCINP<=0.0&OK=="Y"
 |              Next {LINK} if TRLNK.PHASE=="Y"
 |            Else
{PHASE}       End
{SOURCE}      Begin if OK=="N"
 |              UPDTXROW : TXNODE : TSOURCE : PROCESS : TROW : OK
 |            Else
{SOURCE}      End
                                                                   see if link source /node is finished
{NOTOK}       Begin if TROW.STATE<=10
 |              Leave {NOTOK} if TROW.STATE==4
 |              LXFLAG = "N"
 |            Else
{NOTOK}       End
LINK}       Next
            Endrange
IECKDEP}  End
```

```
GENETIK  9.11 library file created by THH at 12 :1 on 1: 2.: 95

Model name    : DESIGN     as 1?+resources+aour:+i

File contents : The C-EVENT  unit CCONDL

EVENT + CCONDL   : conditional events in loops
TG VAR+ NUMBER   :    0 L           0
TG VAR+ MAXIMUM  :    0 L           0
TG VAR+ COUNT    :    0 L           0
M VAR+ TRNODE    :    0 L     ?.00000
M VAR+ TRLNK     :    0 L     ?.00000
T VAR+ OK        :    1 L * *
TG VAR+ TIM      :    0 L           0
AL VAR+ TIMM     :    0 L    0.00000
M VAR+ TRDEF     :    0 L     ?.00000
KT VAR+ LXFLAG1  :    1 L * *
TG VAR+ L        :    0 L           0
TG VAR+ M        :    0 L           0
TG VAR+ N        :    0 L           0
TG VAR+ S        :    0 L           0
AL VAR+ H        :    0 L    0.00000
AL VAR+ W        :    0 L    0.00000
OM VAR+ TRLNK1   :    0 L     ?.00000
TG VAR+ COUNT2   :    0 L           0
OM VAR+ TRNODE1  :    0 L     ?.00000
TG VAR+ AR       :    0 L           0
TG VAR+ RT       :    0 L           0
TG VAR+ RZ       :    0 L           0
KT VAR+ AZ       :   40 L *
OM VAR+ TRLNK2   :    0 L     ?.00000
TG VAR+ FM       :    0 L           0
KT VAR+ LXFLAG6  :    1 L * *
ONDL]   Begin
          Size : NUMBER : MAXIMUM : TLINK
[COND]    Loop for COUNT = 1 to NUMBER step 1
            Getrow : TRLNK : TLINK : COUNT
            Next [COND] if TRLNK.COND<>"C"
            UPDTXROW : TRLNK.PROCESS : TNODE : PROCESS   TRNODE : OK
  [COND6]   Begin if OK=="Y"
  |           Leave [COND6] if TRLNK.PHASE=="Y"
  |           Next [COND] if TRNODE.STATE<>10
  |           Setrange : TLINK   TRLNK.SYSINDEX   TRLNK SYSINDEX
  | [COND14]  Loop
  | |           Getrange : TRLNK1 : COUNT2
  | |           Leave [COND14] if COUNT2==0
  | |           UPDTXROW   TRLNK1 PROCESS   TNODE   PROCESS   TRNODE1   OK
```

```
| | | [COND16]    Begin if TRNODE1 STATE<-10
| | | |             Endrange
| | | |             Next [COND]
| | | |           Else
| | | [COND16]    End
| | [COND14]    Next
| |             Endrange
| |             TIM - TRNODE CIEND
| |           Else
| [COND6]     End
|             UPDINRW : TRLNK SYSINDEX : TNODE : SYSINDEX : TRNODE : OK
| [COND2]     Begin if OK--"Y"
| |             Next [COND] if TRNODE.CKLOOP<>"Y"
| |             UPDTXROW . TRNODE.PROCESS : TLOOPDEF : PROCESS : TRDEF : OK
| | [COND5]     Begin if OK--"Y"
| | |             Leave [COND5] if TRDEF.NOCOND<>0
| | | [COND1]     Begin if TRNODE.STATE<<2|TRNODE.STATE--3
| | | |             Next [COND] if TRNODE.STATE--0
| | | |             CHEKRESL : TRDEF.SYSINDEX : LXFLAG6
| | | |             Next [COND] if LXFLAG6--"N"
| | | | [COND8]     Begin if TRNODE STATE--3
| | | | |             AR - TRNODE.SYSINDEX
| | | | |             Setrange : TLINK : TRNODE.SYSINDEX : TRNODE.SYSINDEX
| | | | | [COND18]    Loop
| | | | | |             Getrange : TRLNK2 : COUNT2
| | | | | |             Leave [COND18] if COUNT2<<0
| | | | | | [COND9]     Begin if OK--"Y"&TRLNK.COND--"C"
| | | | | | |             AZ - TRLNK.PROCESS
| | | | | | |             UPDTXROW : TRLNK.PROCESS : TNODE : PROCESS : TRNODE : OK
| | | | | | | [COND10]    Begin if OK--"Y"
| | | | | | | |             RT - TRNODE.SYSINDEX
| | | | | | | |             Leave [COND1] if SIMTIME>>TRNODE.CIEND
| | | | | | | |             UPDINRW : TRLNK.SYSINDEX : TNODE : SYSINDEX : TRNODE : OK
| | | | | | | |             FM - TRNODE.SYSINDEX
| | | | | | | |           Else
| | | | | | | [COND10]    End
| | | | | | |           Else
| | | | | | [COND9]     End
| | | | | [COND18]    Next
| | | | |             Endrange
| | | | |           Else
| | | | [COND8]     End
| | | |             L - SIMTIME
| | | | [COND4]     Begin if TRDEF.INIT--1|TRDEF.PRIORITY--1
| | | | | [COND7]     Begin if MODEPH1--CSTPH1ST|MODEPH1--CSTPH1EN
| | | | | |             W - SIMTIME+TRNODE.SAMPDUR*TIMEC
| | | | | | [TIK2]      Begin if TRNODE.START+TRNODE.SAMPDUR>>W
| | | | | | |             TRDEF.NOCOND - 1
| | | | | | |             Next [COND]
| | | | | | |           Else
| | | | | | [TIK2]      End
| | | | | |             TRNODE.TESTQUAL - 1
| | | | | |             CHECKPRB : TRNODE.SYSINDEX : LXFLAG1
| | | | | | [QUAL1]     Begin if LXFLAG1--"N"
| | | | | | |             TRNODE.ENDPH1 - SIMTIME
| | | | | | |             Next [COND]
| | | | | | |           Else
| | | | | | [QUAL1]     End
| | | | | |             TIMM - TRNODE.SAMPDUR*TIMEC
| | | | | |             SCHEDULE : ENDP1CON : TIMM : TRNODE
| | | | | |             ADRESBSL : TRDEF.SYSINDEX
| | | | | |             CFLAG - 1
| | | | | |             TRNODE STATE - 3
| | | | | |             DRAWNODE
```

```
| | | | | |             Next (COND)
| | | | | |           Else
| | | | | (COND7)       End
| | | | |               M = TRNODE.START+TRNODE.DUR
| | | | |               H = SIMTIME+TRNODE.DUR*TIMEC
| | | | |               N = 1+TRNODE.DUR*TIMEC
| | | | | (TIM1)        Begin if TRNODE.START+TRNODE.DUR =0
| | | | | |               TRDEF.NOCOND = 1
| | | | | |               Next (COND)
| | | | | |             Else
| | | | | (TIM1)        End
| | | | |               TIMM = TRNODE.DUR*TIMEC
| | | | |               TRNODE.TESTQUAL = 1
| | | | |               CHECKPRE : TRNODE.SYSINDEX : LXFLAG1
| | | | | (QUAL)        Begin if LXFLAG1=="N"
| | | | | |               TRNODE.ENDPH1 = SIMTIME
| | | | | |               Next (COND)
| | | | | |             Else
| | | | | (QUAL)        End
| | | | |               SCHEDULE : ENDP1CON : TIMM : TRNODE
| | | | |               ADRESBSL : TRDEF.SYSINDEX
| | | | |               CFLAG = 1
| | | | |               TRNODE.STATE = 2
| | | | |               DRAWNODE
| | | | |             Else
| | | | (COND4)       End
| | | |             Else
| | | (COND1)       End
| | |             Else
| | (COND5)       End
| |             Else
| (COND2)       End
(COND)        Next
(CONDL)     End
```

```
GENETIK  9.11 library file created by THU at 12:12 on 11-Jul 95

Model name     : DESIGN     as 1?+resources+sources

File contents    The B-EVENT  unit BENDPHIN

.........................................................................................................................

B-EVENT * BENDPHIN   End phase 1 with or without prior         c
FROM VAR. TRLOOP    .     0 L        ?.00000
FROM VAR. TRDEF     .     0 L        ?.00000
INTG VAR. COUNT2    .     0 L              0
TEXT VAR. TXNODE    .    10 L "            "
FROM VAR. TRNODE    :     0 L        ?.00000
TEXT VAR. OK        :     2 L "  "
TEXT VAR. TXTNODE   :    10 L "            "
INTG VAR. DURPH1    :     0 L              0
INTG VAR. NUMBER    :     0 L              0
INTG VAR. MAXIMUM   :     0 L              0
INTG VAR. COUNT     :     0 L              0
INTG VAR. TIM1      :     0 L              0
INTG VAR. DURLP     :     0 L              0
INTG VAR. DU        :     0 L              0
{BENDPHIN} Begin
           TRLOOP = CURRENT
           ADRESIDP : TRLOOP.SYSINDEX
           TIM1 = SIMTIME
           DURLP = TRLOOP.PH1DUR
 {TLOOPP}    Begin if TRLOOP.COND=="C"
  |            TRLOOP.STATE = 3
  |            Setrange : TLOOPDEF : TRLOOP.SYSINDEX : TRLOOP.SYSINDEX
  | {TLOOPP1}  Loop
  | |            Getrange : TRDEF : COUNT2
  | | {TLP}      Begin if TRDEF.NOCOND<>0
  | | |            Endrange
  | | |            Leave {TLOOPP}
  | | |          Else
  | | {TLP}      End
  | |            Leave {TLOOPP1} if COUNT2<<0
  | |            TXNODE = TRDEF.PROCESS
  | |            UFDTXROW : TXNODE : TNODE : PROCESS : TRNODE : OK
  | | {TLOOPP3}  Begin if OK=="Y"
  | | |            TRNODE.STATE = 3
  | | |          Else
  | | {TLOOPP3}  End
  | {TLOOPP1}  Next
  |            Endrange
  |                                                          calc starting time of loop processes
  |            Setrange : TLOOPDEF : TRLOOP.SYSINDEX : TRLOOP.SYSINDEX
  | {TLOOPP2}  Loop
  | |            Getrange : TRDEF : COUNT2
  | |            Leave {TLOOPP2} if COUNT2<<0
  | |            TXNODE = TRDEF.PROCESS
  | |            UFDTXROW : TXNODE : TNODE : PROCESS : TRNODE : OK
  | |            TRNODE.START = TIM1-DURLP
  | |            TRNODE.ENDCOND = TIM1
  | |            DU = TRNODE.START
  | |            TRDEF.PH1DUR = TRLOOP.PH1DUR
  | {TLOOPP2}  Next
  |            Endrange
  |            TRLOOP.STPH1 = TIM1-DURLP
  |            DRAWNODE
  |            Leave {BENDPHIN}
```

```
        Else
LOOPP]    End
        TRLOOP.STATE = 4
        TRLOOP.ENDPH1 = SIMTIME
        Setrange : TLOOPDEF : TRLOOP.SYSINDEX : TRLOOP.SYSINDEX
LOOPPH1] Loop
          Getrange : TRDEF : COUNT2
          Leave [TLOOPPH1] if COUNT2<=0
          TXNODE = TRDEF.PROCESS
          UPDTXROW : TXNODE : TNODE : PROCESS : TRNODE : OK
[CO]      Begin if OK=="Y"
|           TRNODE.STATE = 4
|           TRNODE.ENDPH1 = SIMTIME
|           TRNODE.COND = " "
|         Else
[CO]      End
LOOPPH1] Next
        Endrange
                                                    calc starting time of loop processes
        Setrange : TLOOPDEF : TRLOOP SYSINDEX : TRLOOP.SYSINDEX
TART]   Loop
          Getrange : TRDEF : COUNT2
          Leave [START] if COUNT2<=0
          TXNODE = TRDEF.PROCESS
          UPDTXROW : TXNODE : TNODE : PROCESS : TRNODE : OK
[CA]      Begin if OK=="Y"
|           TRDEF.PH1DUR = TRLOOP.PH1DUR
|           TRNODE.START = TRNODE.ENDPH1-TRLOOP.PH1DUR
|         Else
[CA]      End
TART]   Next
        Endrange
        TRLOOP.STPH1 = SIMTIME-TRLOOP.PH1DUR
        DRAWNODE
OPH1N] End

----------------------------------------------------------------------------------------------------

---------------------------------------------------------------------------
```

APPENDIX VI

SAMPLE REPORTS FROM THE DATA DICTIONARY OF THE GENERIC DATA FLOW MODEL OF THE CONCEPTUAL AND SCHEMATIC DESIGN STAGES

All Definitions

Name Description	Type	Date
ACCEPTABLE-NOISE-LEVELS	Data Element	02/06/94
ACCESS FOR PLANT VEH	Data Flow	29/06/94
ACCESS-EGRESS	Data Element	02/06/94
APP CONCEPT DESIGN REPORT *APP-OUTLINE-ARCH-DES+* *APP-OUTLINE-STR-DES+* *APP-OUTLINE-DRAIN-DES+* *APP-OUTLINE-SERV-DES+* *APP-PROG+* *APP-COST-PLAN*	Data Flow	31/08/94
APP SCHEME DES DOC *STR-SCHEME-DES-DOC+* *ARCH-SCHEME-DES-DOC+* *SERV-SCHEME-DES-DOC+* *DRAIN-SCHEME-DES-DOC+* *OUTLINE-PROJ-SPECS*	Data Flow	28/06/94
APP-BUILDING-MANAGEMENT-SCHEME	Data Element	01/09/94
APP-CABLE-ROUTING-SCHEME	Data Element	21/02/94
APP-CLADDING-SCHEME	Data Element	21/02/94
APP-COM-ARCH-DES	Data Element	17/02/94
APP-COM-SERV-DES	Data Element	17/02/94
APP-COM-STR-DES	Data Element	17/02/94
APP-COMPUTER--SERV-PROV-SCHEME	Data Element	25/02/94
APP-COST-PLAN	Data Element	21/02/94
APP-EMERGENCY-LIGHTING-SCHEME	Data Element	21/02/94
APP-ENVELOPE-MAT	Data Element	21/02/94
APP-ENVELOPE-SCHEME *APP-ENVELOPE-MAT+* *APP-ROOF-SCHEME+* *APP-CLADDING-SCHEME*	Data Structure	31/08/94
APP-FINISHES-OUTLINES	Data Element	28/06/94
APP-FIRE-ALARM-SCHEME	Data Element	21/02/94
APP-FIRE-FIGHTING-SCHEME	Data Element	21/02/94
APP-FLOOR-SLAB-OUTLINE-PROP	Data Element	28/06/94
APP-FOUL-WATER-DRAIN-OUTLINE-PR	Data Element	28/06/94

All Definitions

Name Description	Type	Date
APP-FOUL-WATER-DRAIN-SCHEME	Data Element	21/02/94
APP-FOUND-OUTLINE-PROP	Data Element	28/06/94
APP-GAS-PIPING-SCHEME	Data Element	21/02/94
APP-HEATING&PIPED-SERV-SCHEME *APP-HVAC-SCHEME+* *APP-HEATING-SCHEME+* *APP-VENT-SCHEME+* *APP-SPECIAL-PLANT-SCHEME+* *APP-FIRE-FIGHTING-SCHEME+* *APP-POTABLE-WATER-PIPING-SCHEME+* *APP-GAS-PIPING-SCHEME*	Data Structure	28/06/94
APP-HEATING-SCHEME	Data Element	21/02/94
APP-HVAC-SCHEME	Data Element	21/02/94
APP-ILLUM-LEVELS-SCHEME	Data Element	21/02/94
APP-LIGHTED-AREAS-SCHEME	Data Element	21/02/94
APP-LIGHTNING-PROT-SCHEME	Data Element	21/02/94
APP-OUTLINE-ARCH-DES *APP-PRELIMINARY-LAYOUTS+* *APP-ENVELOPE-SCHEME+* *APP-FINISHES-OUTLINES+* *APP-TYPICAL-SECTIONS*	Data Structure	31/08/94
APP-OUTLINE-DRAIN-DES *APP-SURF-WATER-DRAIN-OUTLINE-PR+* *APP-FOUL-WATER-DRAIN-OUTLINE-PR*	Data Structure	28/06/94
APP-OUTLINE-SERV-DES *APP-HEATING&PIPED-SERV-SCHEME+* *APP-POWER&LIGHTING-SCHEME+* *APP-TELECOM&DATA-SCHEME+* *APP-SPECIAL-SYSTEMS-SCHEME*	Data Structure	28/06/94
APP-OUTLINE-STR-DES *APP-FOUND-OUTLINE-PROP+* *APP-FLOOR-SLAB-OUTLINE-PROP+* *APP-SUPERSTR-OUTLINE-PROP*	Data Structure	28/06/94
APP-POTABLE-WATER-PIPING-SCHEME	Data Element	21/02/94
APP-POWER&LIGHTING-SCHEME *APP-POWER-LOADS+* *APP-CABLE-ROUTING-SCHEME+* *APP-COMPUTER--SERV-PROV-SCHEME+* *APP-LIGHTED-AREAS-SCHEME+* *APP-ILLUM-LEVELS-SCHEME+* *APP-EMERGENCY-LIGHTING-SCHEME*	Data Structure	28/06/94
APP-POWER-LOADS	Data Element	25/02/94

All Definitions

Name Description	Type	Date
APP-PRELIMINARY-BUILDING-LAYOUT	Data Element	28/06/94
APP-PRELIMINARY-LAYOUTS *APP-PRELIMINARY-SITE-LAYOUT+* *APP-PRELIMINARY-BUILDING-LAYOUT*	Data Structure	28/06/94
APP-PRELIMINARY-SITE-LAYOUT *SITE-ACCESS+* *LOADING&UNLOADING-FACILITIES+* *PARKING-FACILITIES*	Data Structure	28/06/94
APP-PROG	Data Element	21/02/94
APP-ROOF-SCHEME	Data Element	21/02/94
APP-SECURITY-SYSTEM-SCHEME	Data Element	21/02/94
APP-SPECIAL-PLANT-SCHEME	Data Element	28/06/94
APP-SPECIAL-SYSTEMS-SCHEME *APP-FIRE-ALARM-SCHEME+* *APP-SECURITY-SYSTEM-SCHEME+* *APP-LIGHTNING-PROT-SCHEME+* *APP-TELECOM-AND-DATA-SCHEME+* *APP-BUILDING-MANAGEMENT-SCHEME*	Data Structure	01/09/94
APP-SUPERSTR-OUTLINE-PROP	Data Element	28/06/94
APP-SURF-WATER-DRAIN-OUTLINE-PR	Data Element	28/06/94
APP-TELECOM&DATA-SCHEME	Data Element	21/02/94
APP-TELECOM-AND-DATA-SCHEME	Data Element	01/09/94
APP-TYPICAL-SECTIONS	Data Element	31/08/94
APP-VENT-SCHEME	Data Element	21/02/94
APPROVALS AND COMMENTS *APP-COM-STR-DES+* *APP-COM-ARCH-DES+* *APP-COM-SERV-DES*	Data Flow	21/02/94
ARCH-SCHEME-DES-DOC	Data Element	28/06/94
ARCH_COST_INF	Data Element	09/02/94
BUDGET	Data Element	09/02/94
BUILDING COST INFORMATION *ARCH_COST_INF+* *STR_COST_INF+* *SERV_COST_INF*	Data Flow	09/02/94
BUILDING REGULATIONS *FIRE-WALLS-LOCATIONS+*	Data Flow	02/06/94

All Definitions

Name Description	Type	Date
PETROL-INTERCEPTORS-REQ+ *HOSE-REELS-REQ+* *ESCAPE-RAMPS-SLOPES+* *ACCESS-EGRESS+* *FIRE-SAFETY-REQ+* *ACCEPTABLE-NOISE-LEVELS+* *ENVIRONMENTAL-STANDARDS*		
BUILDING_LAYOUT	Data Element	10/02/94
BUILDING_MANAGEMENT_SYSTEM	Data Element	01/09/94
CONCEPT DESIGN REPORT	Data Flow	31/08/94
DESIGN_GUIDELINES	Data Element	01/09/94
DEVELOPED LAYOUTS *DIMENSIONED-LAYOUTS*	Data Flow	28/06/94
DEVELOPEDLAYOUTS *DIMENSIONED-LAYOUTS*	Data Flow	28/06/94
DEVELOPING SECTIONS& DETAILS *INCLUDES DEVELOPING SECTIONS&DETAILS FOR:* *GLAZING* *CLADDING* *BLOCK WORKS* *DRY PARTITIONS* *FALSE CEILING* *DOORS&DOOR SCHEDULES*	Process	24/02/94
DIMENSIONED-LAYOUTS	Data Element	28/06/94
DRAIN-SCHEME-DES-DOC	Data Element	28/06/94
DRINK-WAT-PTS	Data Element	21/02/94
ENVIRONMENTAL-STANDARDS	Data Element	02/06/94
ESCAPE-RAMPS-SLOPES	Data Element	23/02/94
EXISTING_DEVELOPMENTS	Data Element	08/02/94
EXIST_SEWER_FACIL	Data Element	08/02/94
FINISHESOUTLINES *FINISHES_TYPES+* *FINISHES_MATERIALS*	Data Flow	09/02/94
FINISHES_MATERIALS	Data Element	09/02/94
FINISHES_STANDARDS *QUALITY OF FINISHING MATERIALS TO BE USED*	Data Element	08/02/94
FINISHES_TYPES	Data Element	09/02/94
FIRE-SAFETY-REQ	Data Element	23/02/94

All Definitions

Name Description	Type	Date
FIRE-WALLS-LOCATIONS	Data Element	23/02/94
FIRE_ALARM_AND_DETECTION	Data Element	09/02/94
FL_SLAB_UNIF_LOAD	Data Element	25/02/94
FOUNDATIONS TYPE *FOUNDATION_TYPE+* *FOUNDATION_DEPTH*	Data Flow	27/06/94
FOUNDATION_DEPTH	Data Element	08/02/94
FOUNDATION_TYPE	Data Element	08/02/94
GENERAL LAYOUTS *SITE_LAYOUT+* *BUILDING_LAYOUT*	Data Flow	31/01/96
HANDDRYERS-DIST	Data Element	21/02/94
HANDDRYERS-LOC	Data Element	21/02/94
HANDINGOVER_DATE	Data Element	08/02/94
HEATING&PIPED SERV *HEATING_PIPED_SERV_BUILDING+* *HEATING_PIPED_SERV_EXTERNAL*	Data Flow	28/06/94
HEATING_PIPED_SERV_BUILDING *RADIATOR_SYSTEM+* *MECH_VENT_TOILETS+* *HOT_AND_COLD_WATER+* *KITCHEN_EXTRACT*	Data Structure	28/06/94
HEATING_PIPED_SERV_EXTERNAL *PERIMETER_FIRE_RING_MAIN+* *INCOMING_POTABLE_WATER+* *INCOMING_FIRE_MAIN+* *SPRINK_TANK+* *PUMP_HOUSE+* *INCOMING_NAT_GAS*	Data Structure	28/06/94
HOSE-REELS-REQ	Data Element	23/02/94
HOT_AND_COLD_WATER	Data Element	09/02/94
INCOMING_FIRE_MAIN	Data Element	09/02/94
INCOMING_NAT_GAS	Data Element	09/02/94
INCOMING_POTABLE_WATER	Data Element	09/02/94
INS REQ *REQ-FOR-FIRE-WALLS+* *FIRE-SAFETY-REQ+*	Data Flow	01/09/94

All Definitions

Name Description	Type	Date
ENVIRONMENTAL-STANDARDS		
KITCH-FAC-LOC	Data Element	28/06/94
KITCHEN_EXTRACT	Data Element	09/02/94
LIGHTING *ILLUMINATION_LEVEL+* *FIXTURES_TYPES+* *CAR_PARK_ILLUMIN+* *FOOTPATHS_ILLUMIN+* *EMERGENCY_LIGHTING*	Data Element	09/02/94
LIGHTING-FIXT-DIST	Data Element	21/02/94
LIGHTNING_PROTECTION	Data Element	09/02/94
LOADING&UNLOADING-FACILITIES	Data Element	21/02/94
MATERIALS SELECTION *SELECTION OF AUXILIARY MATERIALS WITH THE SKELETON, MATERIALS FOR THE ROOF, INTERNAL PARTITIONS AND INTERNAL AND EXTERNAL FINISHES.*	Process	28/06/94
MECH ROOMS *MECH_ROOMS_SIZES+* *MECH_ROOMS_LOCATION*	Data Flow	09/02/94
MECH_ROOMS_LOCATION	Data Element	09/02/94
MECH_ROOMS_SIZES	Data Element	09/02/94
MECH_VENT_TOILETS	Data Element	09/02/94
OUTLINE-PROJ-SPECS	Data Element	28/06/94
PARKING-FACILITIES	Data Element	21/02/94
PERIMETER_FIRE_RING_MAIN	Data Element	09/02/94
PETROL-INTERCEPTORS-REQ	Data Element	23/02/94
PHASING_REQ	Data Element	08/02/94
POWER&LIGHT REQ *POWER_SUPPLIES+* *LIGHTING*	Data Flow	09/02/94
POWER-SUPPLY-DIST	Data Element	21/02/94
POWER_SUPPLIES *VOLTAGE_TYPE+* *ESTIMATED_LOADS+* *CABLE_TYPES+* *DISTRIBUTION_SYSTEM+* *COMPUTER_UPS_REQ*	Data Element	09/02/94
PRELIMINARY-COLOUR-SCHEME	Data Element	28/06/94

All Definitions

Name Description	Type	Date
PROCUREMENT_METHOD	Data Element	08/02/94
PROJECT BRIEF *SITE_BOUNDARIES+* *SITE_LOCATION+* *EXISTING_DEVELOPMENTS+* *PHASING_REQ+* *TENDER_POLICY+* *HANDINGOVER_DATE+* *FINISHES_STANDARDS+* *EXIST_SEWER_FACIL+* *SPECIAL_LOADING_COND+* *BUDGET+* *SPECIFIC_REQ+* *DESIGN_GUIDELINES*	Data Flow	18/10/95
PUMP_HOUSE	Data Element	09/02/94
RADIATOR_SYSTEM	Data Element	09/02/94
REQ-FOR-FIRE-WALLS	Data Element	01/09/94
SECURITY_SYSTEM	Data Element	09/02/94
SERV REQ BY ARCH *HANDDRYERS-LOC+* *DRINK-WAT-PTS+* *KITCH-FAC-LOC*	Data Flow	21/02/94
SERV-SCHEME-DES-DOC	Data Element	28/06/94
SERVICESDIST *LIGHTING-FIXT-DIST+* *POWER-SUPPLY-DIST+* *SPEAKERS-DIST+* *SPRINKLERS-DIST+* *TEL-POINTS-DIST*	Data Flow	21/02/94
SERV_COST_INF	Data Element	09/02/94
SITE-ACCESS	Data Element	21/02/94
SITE_BOUNDARIES	Data Element	08/02/94
SITE_LAYOUT	Data Element	09/02/94
SITE_LOCATION	Data Element	08/02/94
SMOKE_VENTILATORS	Data Element	09/02/94
SPEAKERS-DIST	Data Element	21/02/94
SPECIAL SYSTEMS *TELECOM_AND_DATA+* *FIRE_ALARM_AND_DETECTION+* *SECURITY_SYSTEM+* *LIGHTNING_PROTECTION+*	Data Flow	01/09/94

All Definitions

Name Description	Type	Date
BUILDING_MANAGEMENT_SYSTEM		
SPECIAL_LOADING_COND	Data Element	27/06/94
SPECIFIC_REQ *SPATIAL_REQ+* *FUNCTIONAL_REQ+* *STRATEGIC_REQ+* *OPERATIONAL_REQ*	Data Element	18/10/95
SPRINKLERS-DIST	Data Element	21/02/94
SPRINK_TANK	Data Element	09/02/94
STR-SCHEME-DES-DOC	Data Element	28/06/94
STR_COST_INF	Data Element	09/02/94
TEL-POINTS-DIST	Data Element	21/02/94
TELECOM_AND_DATA	Data Element	09/02/94
TENDER_POLICY	Data Element	08/02/94
TYPICAL SECTIONS&DETAILS *TYPICAL-CLADDING-DETAILS+* *TYPICAL-GLAZING-DETAILS+* *TYPICAL-BLOCK-WORKS-DETAILS+* *TYPICAL-DRY-PARTITIONS-DETAILS+* *TYPICAL-DOOR-SECTIONS+* *PRELIMINARY-COLOUR-SCHEME+* *TYPICAL-FALSE-CEILING-DETAILS*	Data Flow	29/06/94
TYPICAL-BLOCK-WORKS-DETAILS	Data Element	28/06/94
TYPICAL-CLADDING-DETAILS	Data Element	28/06/94
TYPICAL-DOOR-SECTIONS	Data Element	28/06/94
TYPICAL-DRY-PARTITIONS-DETAILS	Data Element	28/06/94
TYPICAL-FALSE-CEILING-DETAILS	Data Element	28/06/94
TYPICAL-GLAZING-DETAILS	Data Element	28/06/94
WARM_AIR_SYSTEM	Data Element	09/02/94

Data Flow Diagram Symbol List

Type	Name	Date
Data Flow	ENVELOPE MATERIALS	31/08/94
Data Flow	ROOF TYPE	22/03/94
Data Flow	FINISHES MATERIALS	22/03/94
Data Flow	ARCH_COST_INF	22/03/94
Data Flow	GENERAL LAYOUTS	31/01/96
Data Flow	FINISHESOUTLINES	22/03/94
Data Flow	ARCH ROOF SCHEME	23/09/94
Data Flow	ENVELOPE REQ	30/06/94
Data Flow	ELEVATIONS	22/03/94
Data Flow	GENERAL LAYOUTS	31/01/96
Data Flow	SITE LAYOUT	22/03/94
Data Flow	FINISHES_STANDARDS	22/03/94
Data Flow	SPECIFIC_REQ	18/10/95
Data Flow	FEEDBACK	27/06/94
Data Flow	REQ FOR PLANT	31/08/94
Data Flow	SKELETON MATERIAL	31/08/94
Data Flow	COST FEEDBACK	31/08/94
Data Flow	TYPICAL SECTIONS	31/08/94
Data Flow	SKELETON MATERIAL	31/08/94
Data Flow	SKELETON MATERIAL	31/08/94
Data Flow	PROPOSED STOREYS NO.	31/01/96
Data Flow	STRUCTURAL SYSTEM	01/09/94
Data Flow	ARCH_COST_INF	18/10/95
Data Flow	APPROVALS AND COMMENTS	18/10/95
Data Flow	APPROVALS AND COMMENTS	18/10/95
Data Flow	SITE LAYOUT	18/10/95
Data Flow	SPECIFIC_REQ	18/10/95
Data Flow	SPECIFIC_REQ	18/10/95
Data Flow	FEEDBACK	18/10/95
Data Flow	REQ FOR PLANT	18/10/95
Data Flow	SKELETON MATERIAL	18/10/95
Data Flow	STRUCTURAL SYSTEM	18/10/95
Data Flow	PROPOSED STOREYS NO.	31/01/96
Data Flow	STRUCTURAL SYSTEM	18/10/95
Data Flow	TYPICAL SECTIONS	18/10/95
Data Flow	SKELETON MATERIAL	18/10/95
Data Flow	REQ FOR PLANT	18/10/95
Data Flow	SPECIFIC_REQ	18/10/95
Data Flow	FEEDBACK	18/10/95
Data Flow	ELEVATIONS	18/10/95
Data Flow	GENERAL LAYOUTS	31/01/96
Data Flow	SITE LAYOUT	18/10/95
Data Flow	OPT FOR FOOTPRINT SHAPE	18/10/95
Data Flow	PROPOSED STOREYS NO.	31/01/96
Data Flow	PROPOSED STOREYS NO.	31/01/96
Data Flow	OPT FOR FOOTPRINT SHAPE	18/10/95
Data Flow	PROP FLOOR LAYOUTS	18/10/95
Data Flow	PROP FLOORLAYOUTS	18/10/95
Data Flow	OPT FOR FOOTPRINT LOC	18/10/95
Data Flow	PROPOSED STOREYS NO.	31/01/96
Data Flow	OPT FOR FOOTPRINT LOC	18/10/95
Data Flow	PROPOSED STOREYS NO.	31/01/96
Data Flow	PROP FLOOR AREAS	18/10/95

Data Flow Diagram Symbol List

Type	Name	Date
Data Flow	FOOTPRINT SHAPE	18/10/95
Data Flow	APPROVALS AND COMMENTS	18/10/95
Data Flow	APPROVALS AND COMMENTS	18/10/95
Data Flow	ARCH_COST_INF	18/10/95
Data Flow	ARCH_COST_INF	18/10/95
Data Flow	FLOOR AREAS	18/10/95
Data Flow	APPROVALS AND COMMENTS	18/10/95
Data Flow	APPROVALS AND COMMENTS	18/10/95
Data Flow	APPROVALS AND COMMENTS	18/10/95
Data Flow	MECH ROOMS	27/06/94
Data Flow	FINISHESOUTLINES	27/06/94
Data Flow	SPECIAL_LOADING_COND	27/06/94
Data Flow	GENERAL SOIL CONDITIONS	23/09/94
Data Flow	ARCH ROOF SCHEME	23/09/94
Data Flow	POWER&LIGHT REQ	31/08/94
Data Flow	HEATING&PIPED SERV	27/06/94
Data Flow	SPECIAL SYSTEMS	27/06/94
Data Flow	VENT SYSTEM	27/06/94
Data Flow	ESTIMATED SLAB THICK&STRENGTH	27/06/94
Data Flow	SKELETON MATERIAL	27/06/94
Data Flow	FOUNDATIONS TYPE	27/06/94
Data Flow	STRUCTURAL SYSTEM	27/06/94
Data Flow	ENVELOPE REQ	18/10/95
Data Flow	ELEVATIONS	31/08/94
Data Flow	GENERAL LAYOUTS	31/01/96
Data Flow	SITE LAYOUT	01/09/94
Data Flow	FINISHES_STANDARDS	18/10/95
Data Flow	GENERAL LAYOUTS	31/01/96
Data Flow	GENERAL LAYOUTS	31/01/96
Data Flow	REQ FOR MECH ROOMS	31/08/94
Data Flow	ARCH ROOF SCHEME	23/09/94
Data Flow	STR_COST_INF	27/06/94
Data Flow	ARCH_COST_INF	27/06/94
Data Flow	SERV_COST_INF	27/06/94
Data Flow	FEEDBACK	31/08/94
Data Flow	FEEDBACK	27/06/94
Data Flow	GENERAL LAYOUTS	31/01/96
Data Flow	SPECIFIC_REQ	27/06/94
Data Flow	SPECIFIC_REQ	18/10/95
Data Flow	SPECIFIC_REQ	31/08/94
Data Flow	REQ FOR PLANT	31/08/94
Data Flow	PRIMARY HVAC SYS	31/08/94
Data Flow	SKELETON MATERIAL	31/08/94
Data Flow	SKELETON MATERIAL	18/10/95
Data Flow	COST FEEDBACK	31/08/94
Data Flow	COST FEEDBACK	31/08/94
Data Flow	COST FEEDBACK	31/08/94
Data Flow	TYPICAL PENETRATIONS	31/08/94
Data Flow	TYPICAL SECTIONS	07/11/94
Data Flow	PROPOSED STOREYS NO.	31/01/96
Data Flow	STRUCTURAL SYSTEM	01/09/94
Data Flow	PROPOSED STOREYS NO.	17/04/96
Data Flow	APPROVALS AND COMMENTS	18/10/95

Data Flow Diagram Symbol List

Type	Name	Date
Data Flow	APPROVALS ANDCOMMENTS	18/10/95
Data Flow	PLANNING REGULATIONS	27/06/94
Data Flow	SITE_LOCATION	27/06/94
Data Flow	EXISTING_DEVELOPMENTS	27/06/94
Data Flow	SITE_BOUNDARIES	27/06/94
Data Flow	SITE SURVEY	27/06/94
Data Flow	FINISHES_STANDARDS	27/06/94
Data Flow	SITE ACCESS	27/06/94
Data Flow	PARKING FACILITIES	27/06/94
Data Flow	LOADING&UNLOADINGFACILITIES	27/06/94
Data Flow	SITE LAYOUT	27/06/94
Data Flow	SITE LAYOUT	27/06/94
Data Flow	SITE LAYOUT	27/06/94
Data Flow	SITE LAYOUT	27/06/94
Data Flow	EXIST_SEWER_FACIL	27/06/94
Data Flow	FOUL WATER DRAIN SYS	27/06/94
Data Flow	SURFACE WATER DRAIN SYS	27/06/94
Data Flow	ARCH ROOF SCHEME	23/09/94
Data Flow	POWER&LIGHT REQ	27/06/94
Data Flow	HEATING&PIPED SERV	27/06/94
Data Flow	SPECIAL SYSTEMS	27/06/94
Data Flow	VENT SYSTEM	27/06/94
Data Flow	ESTIMATED SLAB THICK&STRENGTH	27/06/94
Data Flow	SKELETON MATERIAL	27/06/94
Data Flow	FOUNDATIONS TYPE	27/06/94
Data Flow	STRUCTURAL SYSTEM	27/06/94
Data Flow	ENVELOPE REQ	30/06/94
Data Flow	ELEVATIONS	01/09/94
Data Flow	GENERAL LAYOUTS	31/01/96
Data Flow	HANDINGOVER_DATE	27/06/94
Data Flow	PHASING_REQ	27/06/94
Data Flow	SPECIAL_LOADING_COND	27/06/94
Data Flow	APP CONCEPT DESIGN REPORT	31/08/94
Data Flow	PRICES	31/08/94
Data Flow	BUDGET	27/06/94
Data Flow	BUILDING COST INFORMATION	02/09/94
Data Flow	MECH ROOMS	27/06/94
Data Flow	FINISHESOUTLINES	27/06/94
Data Flow	TENDER_POLICY	27/06/94
Data Flow	COST PLAN	27/06/94
Data Flow	APPROVALS AND COMMENTS	27/06/94
Data Flow	GENERAL SOIL CONDITIONS	23/09/94
Data Flow	GENERAL SOIL CONDITIONS	23/09/94
Data Flow	GENERAL SOIL CONDITIONS	23/09/94
Data Flow	SPECIFIC_REQ	27/06/94
Data Flow	APPROVALS	31/08/94
Data Flow	OUTLINE APPLICATION	31/08/94
Data Flow	COST FEEDBACK	31/08/94
Data Flow	TYPICAL SECTIONS	31/08/94
Data Flow	PROPOSED STOREYS NO.	31/01/96
Data Flow	APPROVALS AND COMMENTS	18/10/95
Data Flow	APP SCHEME DES DOC	27/06/94
Data Flow	CONCEPT DESIGN REPORT	31/08/94

Data Flow Diagram Symbol List

Type	Name	Date
Data Flow	APPROVALS AND COMMENTS	27/06/94
Data Flow	PROJECT BRIEF	27/06/94
Data Flow	DESIGN TENDER DOCUMENTS	27/06/94
Data Flow	CONFIRMED FOUNDATIONS TYPE	31/01/96
Data Flow	FOUNDATIONS DEPTH	23/09/94
Data Flow	APPROX LOADS	23/09/94
Data Flow	APP-FOUND-OUTLINE-PROP	23/09/94
Data Flow	FOUNDATIONS OPTIONS	23/09/94
Data Flow	APP-COST-PLAN	23/09/94
Data Flow	STANDARDS	23/09/94
Data Flow	SITE SURVEY	23/09/94
Data Flow	SUPERSTR LOADS	23/09/94
Data Flow	FOUND SCHEME DES DOC	23/09/94
Data Flow	FLOOR SLAB LOADS	23/09/94
Data Flow	BEARING WALLS LOADS	23/09/94
Data Flow	APP-FOUND-OUTLINE-PROP	23/09/94
Data Flow	SOIL CONDITIONS	23/09/94
Data Flow	DEVELOPED LAYOUTS	23/09/94
Data Flow	FRAMING	23/09/94
Data Flow	LOADS FROM ROOF	23/09/94
Data Flow	STANDARDS	23/09/94
Data Flow	STR SOL FOR DUCTS	23/09/94
Data Flow	REQ FOR PLANT AND DUCTS	23/09/94
Data Flow	FRAME SCHEME DES DOC	23/09/94
Data Flow	SUPERSTR LOADS	23/09/94
Data Flow	APP-SUPERSTR-OUTLINE-PROP	23/09/94
Data Flow	CLEAR HEIGHTS	23/09/94
Data Flow	SKELETON LAYOUT	23/09/94
Data Flow	SKELETON LAYOUT	23/09/94
Data Flow	STR MEMBERS SIZES	23/09/94
Data Flow	STR MEMBERS SIZES	23/09/94
Data Flow	STR MEMBERS DEAD LOADS	23/09/94
Data Flow	SUPERSTR LOADS	23/09/94
Data Flow	APP-SUPERSTR-OUTLINE-PROP	23/09/94
Data Flow	STANDARDS	23/09/94
Data Flow	REQ FOR PLANT AND DUCTS	23/09/94
Data Flow	FRAME TYPICAL SECTIONS	23/09/94
Data Flow	SKELETON LAYOUT	23/09/94
Data Flow	SUPERSTR LOADS	23/09/94
Data Flow	SERVREQ BYARCH	18/10/95
Data Flow	LOUVERSLOC	01/09/94
Data Flow	SERVICESDIST	01/09/94
Data Flow	APP-COM-ARCH-DES	01/09/94
Data Flow	APP-COST-PLAN	01/09/94
Data Flow	APP-PROG	01/09/94
Data Flow	TOPSOILDEPTH	01/09/94
Data Flow	FULL HEIGHTPARTITIONS	01/09/94
Data Flow	REQ FOR PLANT AND DUCTS	01/09/94
Data Flow	MANHOLES LOC&SIZES	01/09/94
Data Flow	LIGHTED AREAS	01/09/94
Data Flow	LIGHTING FIXTURES	01/09/94
Data Flow	AHU LOC	01/09/94
Data Flow	DEVELOPEDLAYOUTS	18/10/95

Data Flow Diagram Symbol List

Type	Name	Date
Data Flow	FINISHED LEVELS	01/09/94
Data Flow	FINAL REQ	01/09/94
Data Flow	BUILDING REGULATIONS	23/09/94
Data Flow	PLANNING CONSTRAINTS	01/09/94
Data Flow	ARCH SPECS	01/09/94
Data Flow	LEVEL OF BRICK WALLS	01/09/94
Data Flow	ROOF SECTIONS	01/09/94
Data Flow	CLEAR HEIGHTS	01/09/94
Data Flow	SKELETON LAYOUT	01/09/94
Data Flow	BEARING WALLS LOC	01/09/94
Data Flow	FOOTPATHS MAT	01/09/94
Data Flow	FIRE RATING REQ	01/09/94
Data Flow	ROOF PATTERN	01/09/94
Data Flow	LANDSCAPED AREAS	01/09/94
Data Flow	ROOF LIGHT	01/09/94
Data Flow	FINISHES SCHEDULES SCHEME	01/09/94
Data Flow	LANDSCAPING SCHEME DRWGS	01/09/94
Data Flow	SCHEME PLANS&ELEVATIONS	01/09/94
Data Flow	TYPICAL SECTIONS&DETAILS	01/09/94
Data Flow	OUTLINE PROJ ARCH SPECS	01/09/94
Data Flow	FIRE ESCAPE LOC	01/09/94
Data Flow	FIRE WALLS LOC	01/09/94
Data Flow	FIRE DOORS LOC	01/09/94
Data Flow	FINISHES&MATERIALS	01/09/94
Data Flow	ROADS LAYOUT	01/09/94
Data Flow	ARCH-SCHEME-DES-DOC	01/02/96
Data Flow	APP-PRELIMINARY-LAYOUTS	01/09/94
Data Flow	APP-ENVELOPE-MAT	01/09/94
Data Flow	APP-FINISHES-OUTLINES	01/09/94
Data Flow	APP-CLADDING-SCHEME	01/09/94
Data Flow	APP-ROOF-SCHEME	01/09/94
Data Flow	FINISHES	01/09/94
Data Flow	PLANS	01/09/94
Data Flow	REQUEST FOR APP.	31/01/96
Data Flow	SITE LEVELS	01/09/94
Data Flow	OUTLINE PROJ ARCH SPECS	01/09/94
Data Flow	APP-TYPICAL-SECTIONS	01/09/94
Data Flow	INS REQ	01/09/94
Data Flow	STANDARDS	01/09/94
Data Flow	STANDARDS	01/09/94
Data Flow	STANDARDS	01/09/94
Data Flow	STANDARDS	01/09/94
Data Flow	STANDARDS	01/09/94
Data Flow	STR SOL FOR DUCTS	01/09/94
Data Flow	REQ FOR PLANT AND DUCTS	01/09/94
Data Flow	STR SOL FOR DUCTS	01/09/94
Data Flow	APP-OUTLINE-ARCH-DES	14/08/95
Data Flow	PLANNING SUB	18/10/95
Data Flow	PLANNING CONSTRAINTS	18/10/95
Data Flow	FEEDBACK	18/10/95
Data Flow	DEVELOPED LAYOUTS	18/10/95
Data Flow	COST INF	18/10/95
Data Flow	APP-OUTLINE-STR-DES	01/09/94

Data Flow Diagram Symbol List

Type	Name	Date
Data Flow	APP-COM-STR-DES	01/09/94
Data Flow	SITE SURVEY	01/09/94
Data Flow	EXISTING INFRASTR	18/10/95
Data Flow	FINISHED LEVELS	01/09/94
Data Flow	DEVELOPED LAYOUTS	01/09/94
Data Flow	FINAL REQ	01/09/94
Data Flow	BUILDING REGULATIONS	01/09/94
Data Flow	PLANNING CONSTRAINTS	18/10/95
Data Flow	ARCH SPECS	01/09/94
Data Flow	LEVEL OF BRICK WALLS	01/09/94
Data Flow	ROOF SECTIONS	01/09/94
Data Flow	CLEAR HEIGHTS	01/09/94
Data Flow	SKELETON LAYOUT	01/09/94
Data Flow	BEARING WALLS LOC	01/09/94
Data Flow	DEVELOPED LAYOUTS	01/09/94
Data Flow	STR SPECS	01/09/94
Data Flow	STR-SCHEME-DES-DOC	01/09/94
Data Flow	SOIL CONDITIONS	23/09/94
Data Flow	REQ FOR PLANT AND DUCTS	01/09/94
Data Flow	MANHOLES LOC&SIZES	01/09/94
Data Flow	LIGHTED AREAS	01/09/94
Data Flow	LIGHTING FIXTURES	01/09/94
Data Flow	AHU LOC	18/10/95
Data Flow	DEVELOPED LAYOUTS	01/09/94
Data Flow	SMOKE DETECT&VENT REQ	01/09/94
Data Flow	TOPSOILDEPTH	01/09/94
Data Flow	FULL HEIGHTPARTITIONS	01/09/94
Data Flow	SERV SPECS	01/09/94
Data Flow	ACCESS FOR PLANT VEH	01/09/94
Data Flow	APP-PRELIMINARY-SITE-LAYOUT	01/09/94
Data Flow	DRAIN-SCHEME-DES-DOC	01/09/94
Data Flow	SERV-SCHEME-DES-DOC	01/09/94
Data Flow	APP-PROG	01/09/94
Data Flow	APP-COST-PLAN	01/09/94
Data Flow	APP-COST-PLAN	01/09/94
Data Flow	APP-PROG	01/09/94
Data Flow	APP-PROG	01/09/94
Data Flow	APP-COST-PLAN	01/09/94
Data Flow	APP-OUTLINE-SERV-DES	01/09/94
Data Flow	APP-OUTLINE-DRAIN-DES	24/05/95
Data Flow	APP-OUTLINE-ARCH-DES	01/09/94
Data Flow	APP-COM-ARCH-DES	01/09/94
Data Flow	APP-COM-SERV-DES	01/09/94
Data Flow	SERVREQ BYARCH	01/09/94
Data Flow	LOUVERSLOC	01/09/94
Data Flow	SERVICESDIST	01/09/94
Data Flow	ROADS LAYOUT	01/09/94
Data Flow	ARCH-SCHEME-DES-DOC	01/09/94
Data Flow	REQUEST FOR APP.	31/01/96
Data Flow	APP-COST-PLAN	01/09/94
Data Flow	APP-PROG	01/09/94
Data Flow	APPROVALS	01/09/94
Data Flow	DRAINAGEDISCHARGEQUALITY	01/09/94

Data Flow Diagram Symbol List

Type	Name	Date
Data Flow	DRAINAGE APPROVALS	01/09/94
Data Flow	FOOTPATHS MAT	01/09/94
Data Flow	SITE LEVELS	01/09/94
Data Flow	SOIL CONDITIONS	23/09/94
Data Flow	REQ FOR PLANT AND DUCTS	18/10/95
Data Flow	GROUND CONDITIONS	01/09/94
Data Flow	OUTLINE-PROJ-SPECS	01/09/94
Data Flow	OUTLINE PROJ SERV SPECS	01/09/94
Data Flow	OUTLINE PROJ STR SPECS	01/09/94
Data Flow	OUTLINE PROJ ARCH SPECS	01/09/94
Data Flow	EXISTING DRAINAGE	01/09/94
Data Flow	INS REQ	01/09/94
Data Flow	INS REQ	01/09/94
Data Flow	STANDARDS	01/09/94
Data Flow	STANDARDS	01/09/94
Data Flow	STANDARDS	01/09/94
Data Flow	STR SOL FOR DUCTS	01/09/94
Data Flow	APP CONCEPT DESIGN REPORT	23/09/94
Data Flow	APPROVALS AND COMMENTS	18/10/95
Data Flow	REVISED COST ESTIMATE	18/10/95
Data Flow	COST INF	18/10/95
Data Flow	COST INF	18/10/95
Data Flow	COST INF	18/10/95
Data Flow	COST INF	18/10/95
Data Flow	APPROVALS ANDCOMMENTS	18/10/95
Data Flow	PLANNING SUB	18/10/95
Data Flow	APP CONCEPT DESIGN REPORT	31/08/94
Data Flow	APP SCHEME DES DOC	27/06/94
Data Flow	APPROVALS AND COMMENTS	22/03/94
Data Flow	PROJECT BRIEF	22/03/94
Data Flow	CONCEPT DESIGN REPORT	31/08/94
Data Flow	PLANNING REGULATIONS	31/08/94
Data Flow	APPROVALS AND COMMENTS	22/03/94
Data Flow	SITE SURVEY	22/03/94
Data Flow	SITE SURVEY	22/03/94
Data Flow	SITE SURVEY	22/03/94
Data Flow	APPROVALS	31/08/94
Data Flow	OUTLINE APPLICATION	31/08/94
Data Flow	APP-FOUND-OUTLINE-PROP	01/09/94
Data Flow	APP-FLOOR-SLAB-OUTLINE-PROP	01/09/94
Data Flow	APP-SUPERSTR-OUTLINE-PROP	01/09/94
Data Flow	APP-PROG	01/09/94
Data Flow	APP-COST-PLAN	01/09/94
Data Flow	REQ FOR PLANT AND DUCTS	01/09/94
Data Flow	OUTLINE PROJ STR SPECS	01/09/94
Data Flow	LEVEL OF BRICK WALLS	01/09/94
Data Flow	ROOF SECTIONS	01/09/94
Data Flow	CLEAR HEIGHTS	01/09/94
Data Flow	SKELETON LAYOUT	01/09/94
Data Flow	BEARING WALLS LOC	01/09/94
Data Flow	DEVELOPED LAYOUTS	01/09/94
Data Flow	STR SPECS	01/09/94
Data Flow	STR-SCHEME-DES-DOC	01/09/94

Data Flow Diagram Symbol List

Type	Name	Date
Data Flow	SOIL CONDITIONS	23/09/94
Data Flow	APP-OUTLINE-STR-DES	01/09/94
Data Flow	APP-COM-STR-DES	01/09/94
Data Flow	SITE SURVEY	01/09/94
Data Flow	BEARING WALLS LOADS	01/09/94
Data Flow	SUPERSTR LOADS	01/09/94
Data Flow	FOUND SCHEME DES DOC	01/09/94
Data Flow	FLOOR SLAB LOADS	01/09/94
Data Flow	ANC BUILD SCHEME	01/09/94
Data Flow	FRAME SCHEME DES DOC	01/09/94
Data Flow	BW TYPICAL DETAILS	01/09/94
Data Flow	SITE SURVEY	01/09/94
Data Flow	SITE SURVEY	01/09/94
Data Flow	COMMENTS&REQ MOD	01/09/94
Data Flow	UNCHECKED SCHEME STR DES	01/09/94
Data Flow	FRAMING	01/09/94
Data Flow	LOADS FROM ROOF	23/09/94
Data Flow	REQ FOR PLANT AND DUCTS	01/09/94
Data Flow	STANDARDS	01/09/94
Data Flow	STANDARDS	01/09/94
Data Flow	STR SOL FOR DUCTS	01/09/94
Data Flow	REQ FOR PLANT AND DUCTS	01/09/94
Data Flow	STR SOL FOR DUCTS	01/09/94
Data Flow	REQ FOR PLANT AND DUCTS	01/09/94
Data Flow	DEVELOPED LAYOUTS	23/09/94
Data Flow	DEVELOPED LAYOUTS	23/09/94
Data Flow	APP-COST-PLAN	23/09/94
Data Flow	FLOOR SLAB COST INF	23/09/94
Data Flow	FOUND COST INF	23/09/94
Data Flow	STR_COST_INF	22/03/94
Data Flow	REQ FOR MECH ROOMS	22/03/94
Data Flow	ARCH ROOF SCHEME	31/01/96
Data Flow	SUPERSTR COST INF	23/09/94
Data Flow	SPECIAL_LOADING_COND	27/06/94
Data Flow	GENERAL SOIL CONDITIONS	23/09/94
Data Flow	ESTIMATED SLAB THICK&STRENGTH	27/06/94
Data Flow	SKELETON MATERIAL	23/09/94
Data Flow	FOUNDATIONS TYPE	27/06/94
Data Flow	STRUCTURAL SYSTEM	22/03/94
Data Flow	GENERAL LAYOUTS	31/01/96
Data Flow	GENERAL LAYOUTS	31/01/96
Data Flow	FEEDBACK	02/06/94
Data Flow	JOINTS LAYOUT	02/06/94
Data Flow	APPROX QUANTITIES	27/06/94
Data Flow	SPECIFIC_REQ	27/06/94
Data Flow	COST FEEDBACK	31/08/94
Data Flow	PRIMARY HVAC SYS	31/08/94
Data Flow	COST FEEDBACK	31/08/94
Data Flow	COST FEEDBACK	31/08/94
Data Flow	TYPICAL PENETRATIONS	31/08/94
Data Flow	TYPICALPENETRATIONS	31/08/94
Data Flow	PRIMARY HVAC SYS	31/08/94
Data Flow	PROPOSED STOREYS NO.	31/01/96

Data Flow Diagram Symbol List

Type	Name	Date
Data Flow	SPECIAL_LOADING_COND	23/09/94
Data Flow	STRUCTURAL SYSTEM OPTIONS	23/09/94
Data Flow	SKELETON MATERIAL OPTIONS	23/09/94
Data Flow	SKELETON MATERIAL	23/09/94
Data Flow	SKELETON MATERIAL	23/09/94
Data Flow	TYPICALPENETRATIONS	23/09/94
Data Flow	PRIMARY HVAC SYS	23/09/94
Data Flow	COST FEEDBACK	23/09/94
Data Flow	SPECIFIC_REQ	23/09/94
Data Flow	APPROX QUANTITIES	23/09/94
Data Flow	FEEDBACK	23/09/94
Data Flow	GENERAL LAYOUTS	31/01/96
Data Flow	REQ FOR MECH ROOMS	23/09/94
Data Flow	ARCH ROOF SCHEME	24/05/95
Data Flow	SUPERSTR COST INF	23/09/94
Data Flow	SKELETON MATERIAL	23/09/94
Data Flow	STRUCTURAL SYSTEM	23/09/94
Data Flow	COST FEEDBACK	23/09/94
Data Flow	STRUCTURAL SYSTEM	23/09/94
Data Flow	FEEDBACK	23/09/94
Data Flow	FEEDBACK	23/09/94
Data Flow	SPECIAL_LOADING_COND	23/09/94
Data Flow	SPECIFIC_REQ	23/09/94
Data Store	FINISHES MATERIALS	22/03/94
Data Store	PRICES	27/06/94
Data Store	SPECS	01/09/94
Data Store	GEOTECH REPORT	01/09/94
Data Store	INFRASTR DRWGS	01/09/94
Data Store	SPECS	01/09/94
Data Store	SPECS	01/09/94
Data Store	SERV SPECS	01/09/94
Data Store	STANDARDS	01/09/94
Data Store	STANDARDS	01/09/94
Data Store	STANDARDS	01/09/94
Data Store	TOPOGRAPHICAL MAPS	01/02/96
Data Store	SPECS	01/09/94
External	CLIENT	27/06/94
External	MAIN CONTRACTOR	28/06/94
External	DESIGN & BUILD CONTRACTOR	27/06/94
External	CLIENT	01/09/94
External	BUILDING CONTROL OFFICER	01/09/94
External	PLANNING AUTHORITIES	01/09/94
External	BUILDING CONTROL OFFICER	01/09/94
External	HIGHWAYSAUTHORITY	01/09/94
External	NATIONALRIVERSAUTHORITY	01/09/94
External	BUILDINGCONTRCLOFFICER	01/09/94
External	LOCAL AUTHORITY	01/09/94
External	INSURERS	01/09/94
External	INSURERS	01/09/94
External	CLIENT	18/10/95
External	CLIENT	18/10/95
External	PLANNING AUTHORITIES	22/03/94
Process	ROOF ARCH CONCEPT DESIGN	23/09/94

Data Flow Diagram Symbol List

Type	Name	Date
Process	ENVELOPE CONCEPT DESIGN	31/08/94
Process	MATERIALS SELECTION	22/03/94
Process	ARRANGEMENT OF SPACES	01/09/94
Process	CONSIDER BUILDING LAYOUT OPT	31/01/96
Process	CONSIDER FOOTPRINT SHAPE OPT	18/10/95
Process	CONSIDER NO OF STOREYS	18/10/95
Process	CONSIDER FOOTPRINT LOC OPT	18/10/95
Process	APPROX FLOOR AREAS DISTRIBUTION	31/01/96
Process	SERV CONCEPT DESIGN	31/08/94
Process	STR CONCEPT DESIGN	31/08/94
Process	ARCH CONCEPT DESIGN	31/08/94
Process	SITE PLANNING	27/06/94
Process	CONCEPT DESIGN REPORT PRODUCTIO	31/08/94
Process	DRAINAGE CONCEPT DESIGN	31/08/94
Process	BUILDING CONCEPT DESIGN	31/08/94
Process	ESTIMATING COSTS	27/06/94
Process	PRELIMINARY SITE INVESTIGATION	27/06/94
Process	SCHEME DESIGN PROCESS	27/06/94
Process	DETAILED DESIGN	28/06/94
Process	PROD FOUND SCH DES SKET&TABLES	23/09/94
Process	CONFIRM FOUNDATIONS TYPE	31/01/96
Process	APPROX FOUND LOAD CALCULATIONS	23/09/94
Process	CONSIDER FOUNDATIONS OPTIONS	23/09/94
Process	APPROX FRAME LOADS CALCULATIONS	24/10/95
Process	PRODUCE TYPICAL FRAME SECTIONS	24/10/95
Process	PRODUCE SKELETON LAYOUT	23/09/94
Process	CALC APPROX SIZE FOR STR MEMBER	24/10/95
Process	DEVELOPING SECTIONS& DETAILS	01/09/94
Process	ARCH DESIGN FOR ROOF	24/10/95
Process	DECIDE ON FINISHES &MATERIALS	01/09/94
Process	LANDSCAPING SCHEME DESIGN	01/09/94
Process	DEVELOPING PLANS&ELEVATIONS	01/09/94
Process	ESTABLISH FIRE RATING REQ	01/09/94
Process	ARCH DESIGN REVIEW	18/10/95
Process	PRODUCE OUTLINE ARCH SPECS	24/10/95
Process	PLANNING SUB CONSULTATION	18/10/95
Process	SCHEME STRUCTURAL DESIGN	01/09/94
Process	EXTERNAL WORKS SCHEME DESIGN	01/09/94
Process	SCHEME DRAINAGE DESIGN	01/09/94
Process	SCHEME SERVICES DESIGN	01/09/94
Process	SCHEME ARCHITECTURAL DESIGN	01/09/94
Process	SITE INVESTIGATION	01/09/94
Process	PRODUCE OUTLINE PROJECT SPECS	24/10/95
Process	REVISE COST ESTIMATE	18/10/95
Process	SCHEME DESIGN	27/06/94
Process	CONCEPT DESIGN	31/08/94
Process	BEARING WALLS SCHEME DESIGN	01/09/94
Process	STRUCTURAL DESIGN REVIEW	18/10/95
Process	FOUNDATIONS SCHEME DESIGN	23/09/94
Process	FLOOR SLAB SCHEME DESIGN	01/09/94
Process	FRAME SCHEME DESIGN	23/09/94
Process	ANCILLARY BUILD SCHEME DESIGN	01/09/94
Process	PRELIMINARY STR DESIGN CHECKS	01/09/94

Data Flow Diagram Symbol List

Type	Name	Date
Process	ROOF STR SCHEME DESIGN	24/10/95
Process	SUPERSTR CONCEPT DESIGN	23/09/94
Process	FLOOR SLAB CONCEPT DESIGN	31/08/94
Process	FOUNDATIONS CONCEPT DESIGN	31/08/94
Process	DECIDE ON STRUCTURAL SYSTEM	23/09/94
Process	ESTIMATE SUPERSTR COSTS	23/09/94
Process	CONSIDER STR SYSTEM OPTIONS	23/09/94
Process	DECIDE ON SKELETON MATERIAL	23/09/94
Process	CONSIDER SKELETON MAT OPTIONS	23/09/94

APPENDIX VII

A TYPICAL VALUE MANAGEMENT SESSION

DEPARTMENT OF MANUFACTURING PROCESSES
LOUGHBOROUGH UNIVERSITY OF TECHNOLOGY
VALUE MANAGEMENT WORKSHOP - 24TH MARCH 1995

· provide a friendly 'Fraunhofer like' manufacturing engineering department
include a responsive world class research and teaching environment
oviding a natural interaction within the department, the rest of the school,
dustry and academia.

e project must:-

- Provide enough space
- Be of benefit to the department
- Enable the department to maintain an individual identity

(P1) Flexibility
17

- (A) No loss of effectiveness after move 33
- (B) Flexibility of all services 25
- (C) Low cost change 23
- (D) Growth allowance 19

(P2) Internal Interaction
10

- (A) Cohesion 22
- (B) Largely self contained environment 22
- (C) Maximum opportunity for appropriate personnel interaction 15
- (D) Retain integrated way of working 26
- (E) Realise balance of shared efficiency 15

(P3) Low Running Costs
12

- (A) Effective security 28
- (B) Ease of access 24
- (C) Ease of space management 28
- (D) Low maintenance 21

(P4) Maximise Research Income
18

- (A) Continue to respond to research 41
- (B) Support a range of research 59

(P5) Quality Shop Window
14

- (A) Quality management 45
- (B) Professional aesthetic 55

(P6) Effective Recruitment
15

- (A) Stimulating/Creative Environment 55
- (B) Quality of Learning Experience 45

(P7) Undergraduate Experience
14

- (A) Quality of Learning Experience 32
- (B) Quality of Space Allocation 28
- (C) Access to School Social Facility 16
- (D) Quality of Facilities 24

Value Management Workshop
Evaluation of Objectives
Designation

Objective	flexibility	interaction	low costs	maximise research income	quality shop window	effective recruitment	under-graduate experience				TOTAL	Ckeck
R.BELL	9	9	7	10	8	10	8	0	0	0	61	
%	15	15	11	16	13	16	13	0	0	0	100	100
D.WILLIAMS	5	1	3	7	2	6	4	0	0	0	28	
%	18	4	11	25	7	21	14	0	0	0	100	100
C.BACKHOUSE	100	90	70	80	80	90	80	0	0	0	590	
%	17	15	12	14	14	15	14	0	0	0	100	100
J.EDWARDS	70	50	40	90	80	60	60	0	0	0	450	
%	16	11	9	20	18	13	13	0	0	0	100	100
T.DOWNHAM	100	40	100	80	90	50	75	0	0	0	535	
%	19	7	19	15	17	9	14	0	0	0	100	100
% TOTAL	84	52	62	90	68	76	68	0	0	0	500	500

TEAM AVERAGE	17	10	12	18	14	15	14	0	0	0	100

LOUGHBOROUGH UNIVERSITY OF TECHNOLOGY
NEW ENGINEERING COMPLEX
VALUE MANAGEMENT DESIGN BRIEFING WORKSHOP
DEPARTMENT OF AERONAUTICAL AND AUTOMOTIVE AND TRANSPORT TECHNOLOGY

30TH MARCH 1995

A G E N D A

Jonathan French and David Hammond from AMEC will facilitate the session.

9.15 am.	Introduction	J French
9.30 am.	Project Objective Structuring	J French D Hammond
10.45 am.	Refreshment Break	
11.00 am.	Value Tree Evaluation & Weighing	J French D Hammond
11.45 am.	Review and Agree Objectives	J French
12.00 pm.	Refreshment Break	
12.15 pm.	Speculation Brainstorm	J French D Hammond
12.45 pm.	Review Actions	J French
1.00 pm.	CLOSE	

APPENDIX VIII

AN EXAMPLE OF A ROOM DATA SHEET (RDS)

LOUGHBOROUGH UNIVERSITY OF TECHNOLOGY

ENGINEERING COMPLEX

ROOM DATA SHEET DATA SHEET NO:................

DEPARTMENT: AAETS	
ROOM NAME: LIFTING BAY	
ROOM NUMBER: (to be completed by ADAM Design)	

BUILDING CONSIDERATIONS

CRITICAL DIMENSIONS None / Yes $39\ m^2$ Length: Width: Height: 5 METRES MAX. Nº OF OCCUPANTS: 5 Use is continuous/intermittent 24 hour/daytime/night time (please ring as appropriate)	FUNCTIONAL DESCRIPTION: VEHICLE MODIFICATION AND REPAIR AREA. (RESEARCH & TEACHING) TO HOUSE VEHICLE LIFT WITH SPACE FOR ADDITIONAL VEHICLE ALONGSIDE LIFT.
OTHER CONSIDERATIONS Direct Access to outside "Yard" required. VEHICLE EXHAUST GAS EXTRACT SYSTEM REQUIRED	RELATIONSHIPS WITH OTHER ROOMS (or departments) Adj: to dynamometer and electrical vehicle lab.

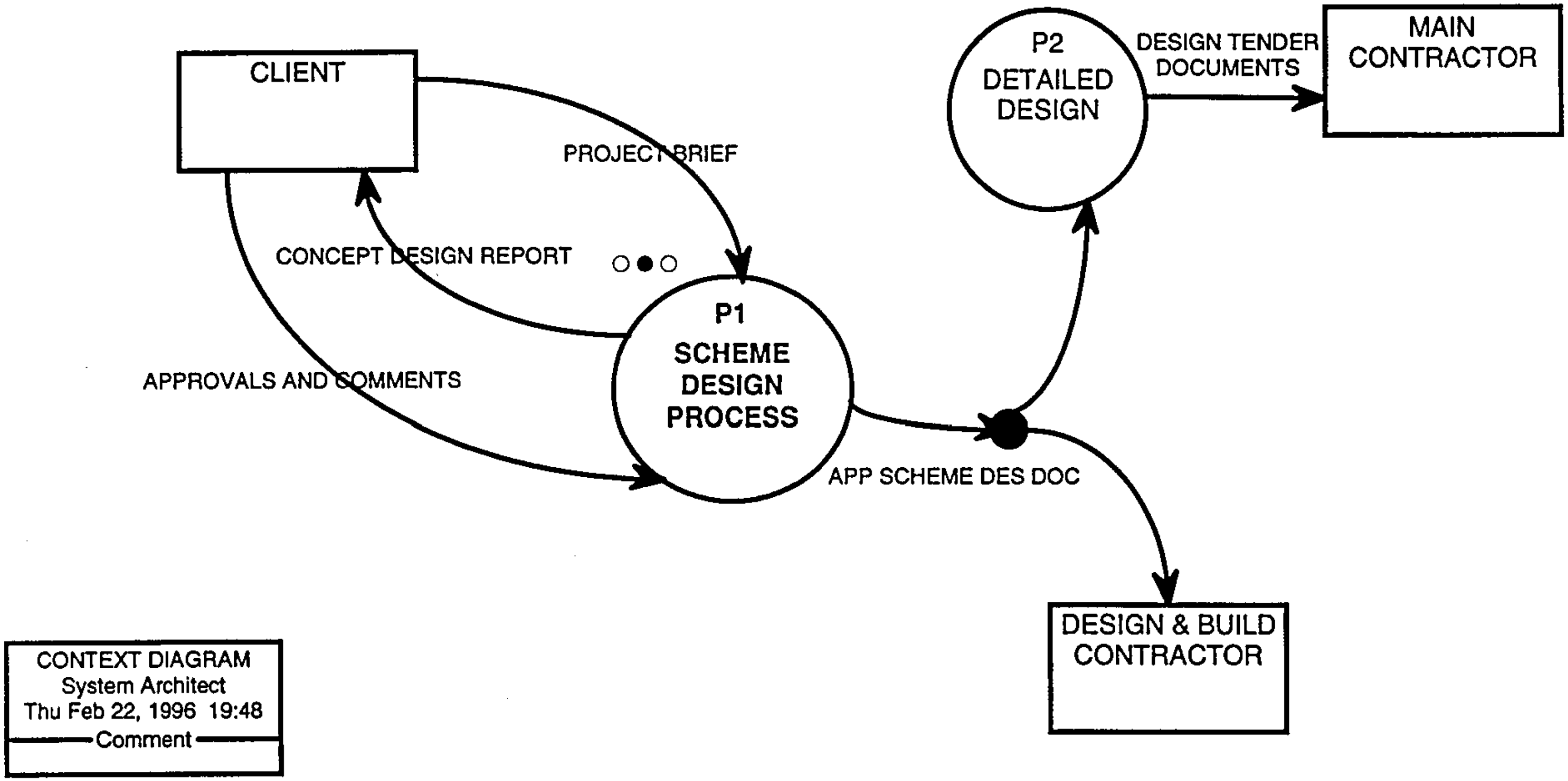
Issue	Purpose	Date
Word Proc. Reference: Tables	Document Number: 00585-IRY-0003	Rev.

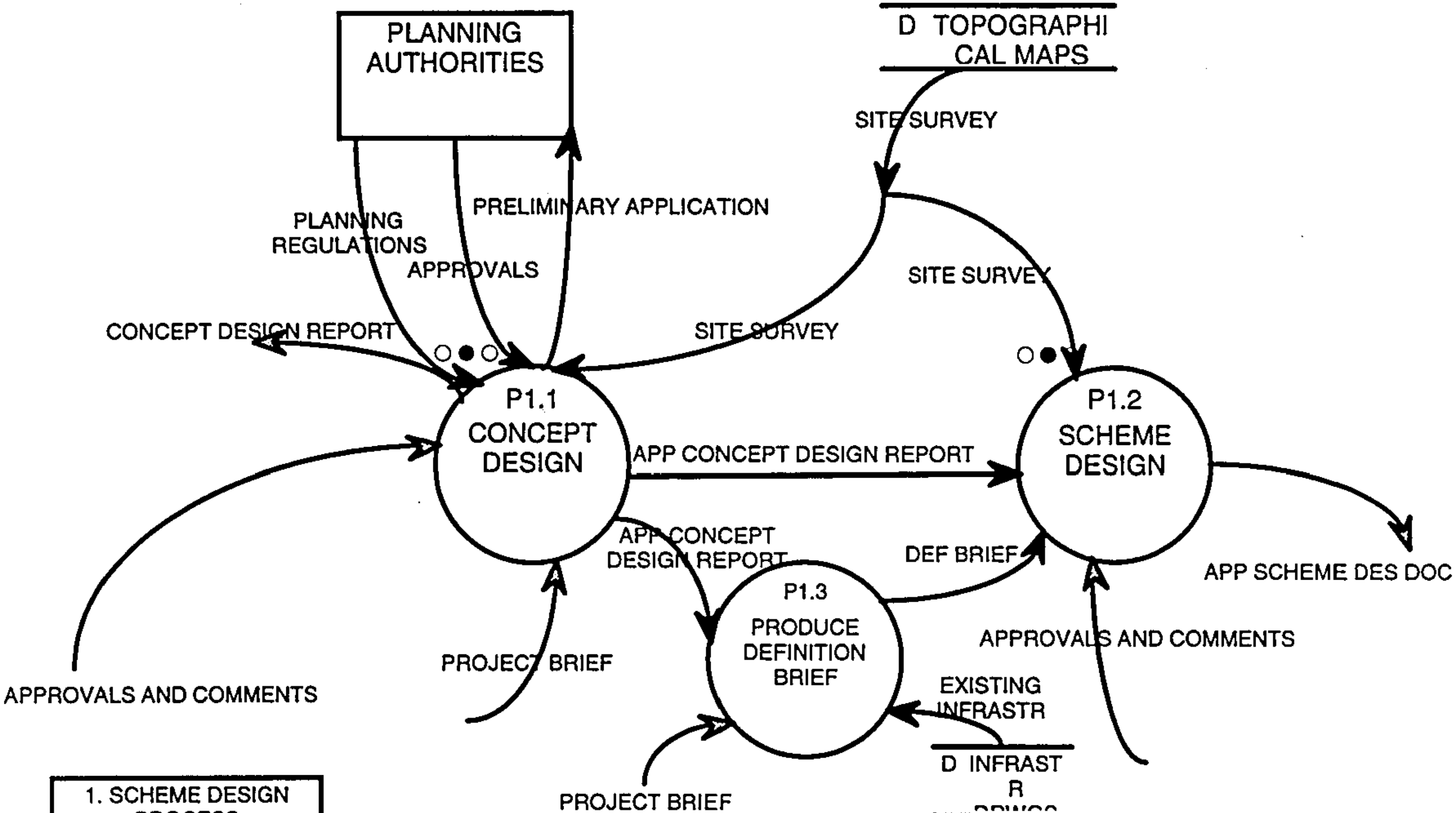
AMEC Design and Management

AMEC Design & Management Ltd
Timothy's Bridge Road
Stratford-upon-Avon,
Warwickshire CV37 9NJ

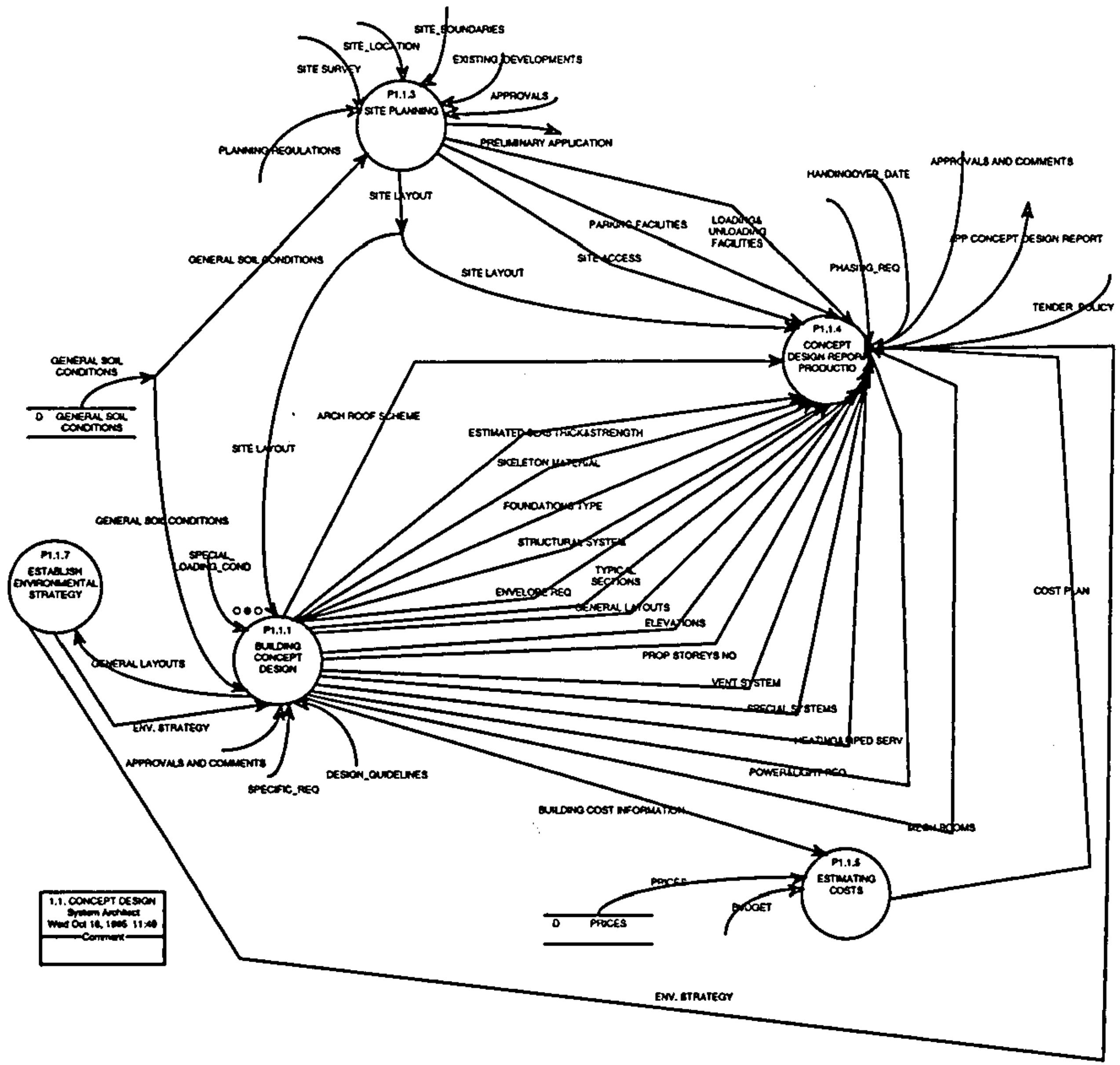
APPENDIX IX

THE DATA FLOW MODEL FOR THE LUT ENGINEERING COMPLEX CASE STUDY

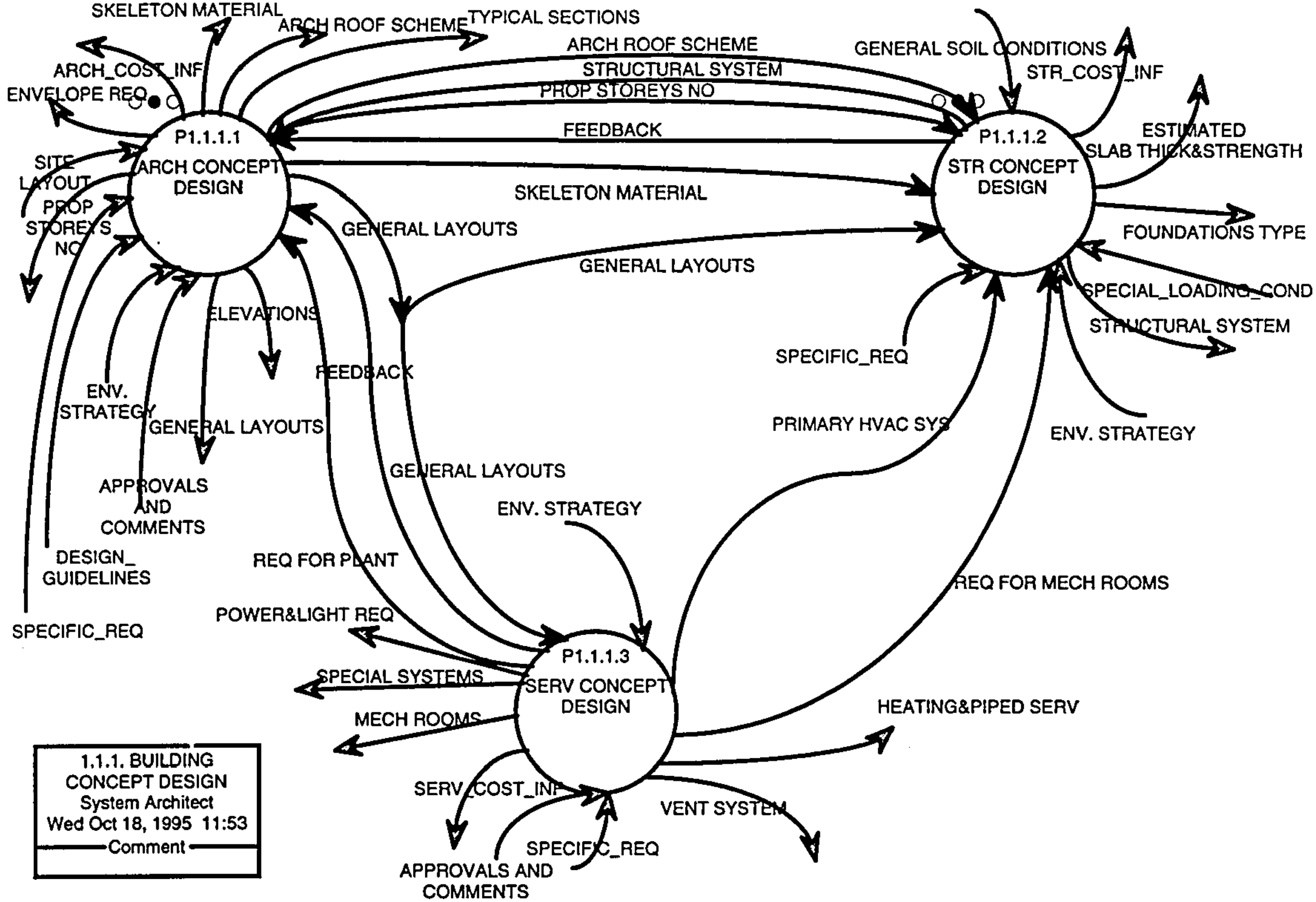

CLIENT
PROJECT BRIEF
CONCEPT DESIGN REPORT
APPROVALS AND COMMENTS
P1
SCHEME
DESIGN
PROCESS
APP SCHEME DES DOC
P2
DETAILED
DESIGN
DESIGN TENDER
DOCUMENTS
MAIN
CONTRACTOR
DESIGN & BUILD
CONTRACTOR
CONTEXT DIAGRAM
System Architect
Thu Feb 22, 1996 19:48
Comment



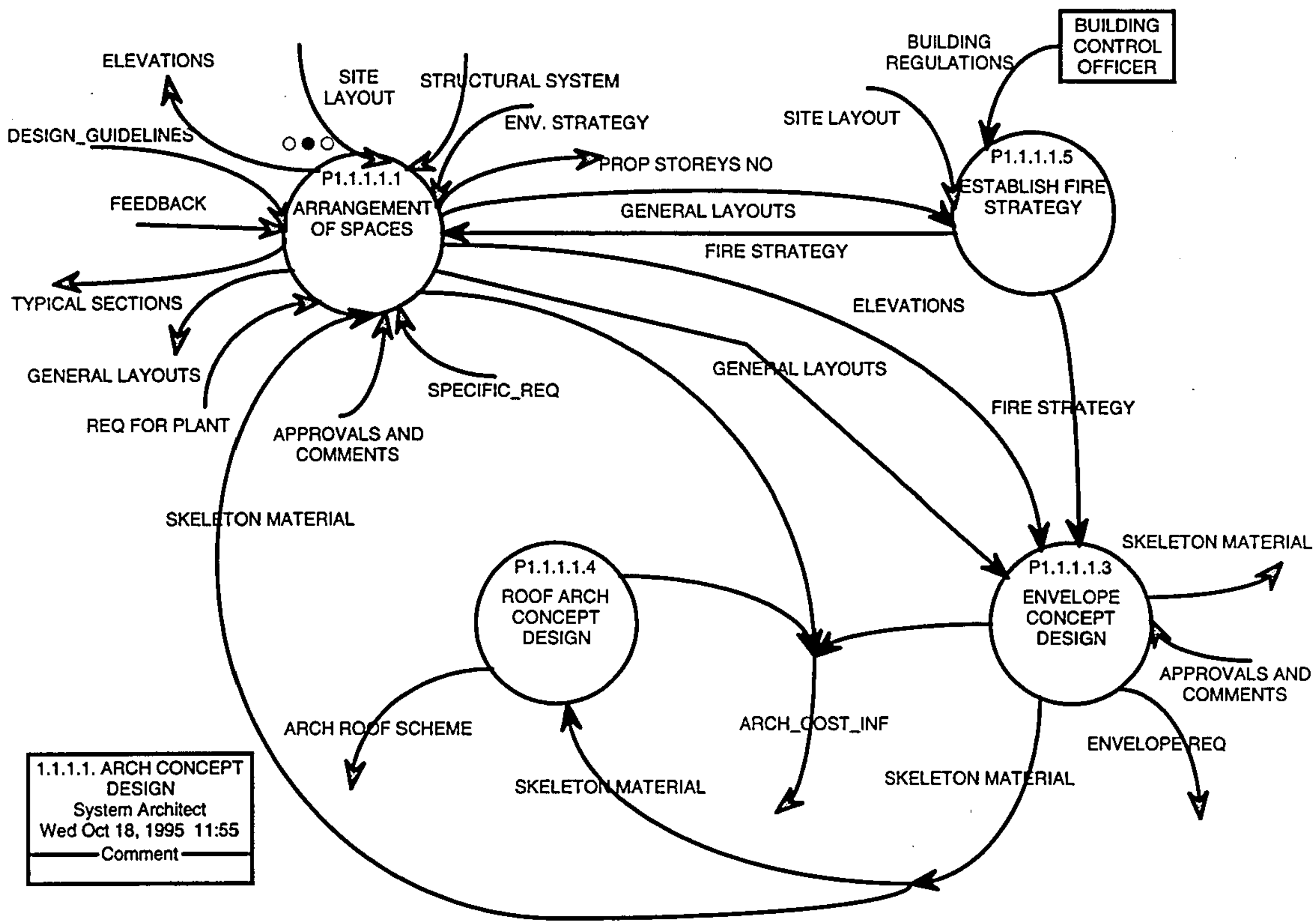
1. SCHEME DESIGN PROCESS
System Architect
Wed Oct 18, 1995 11:22
Comment



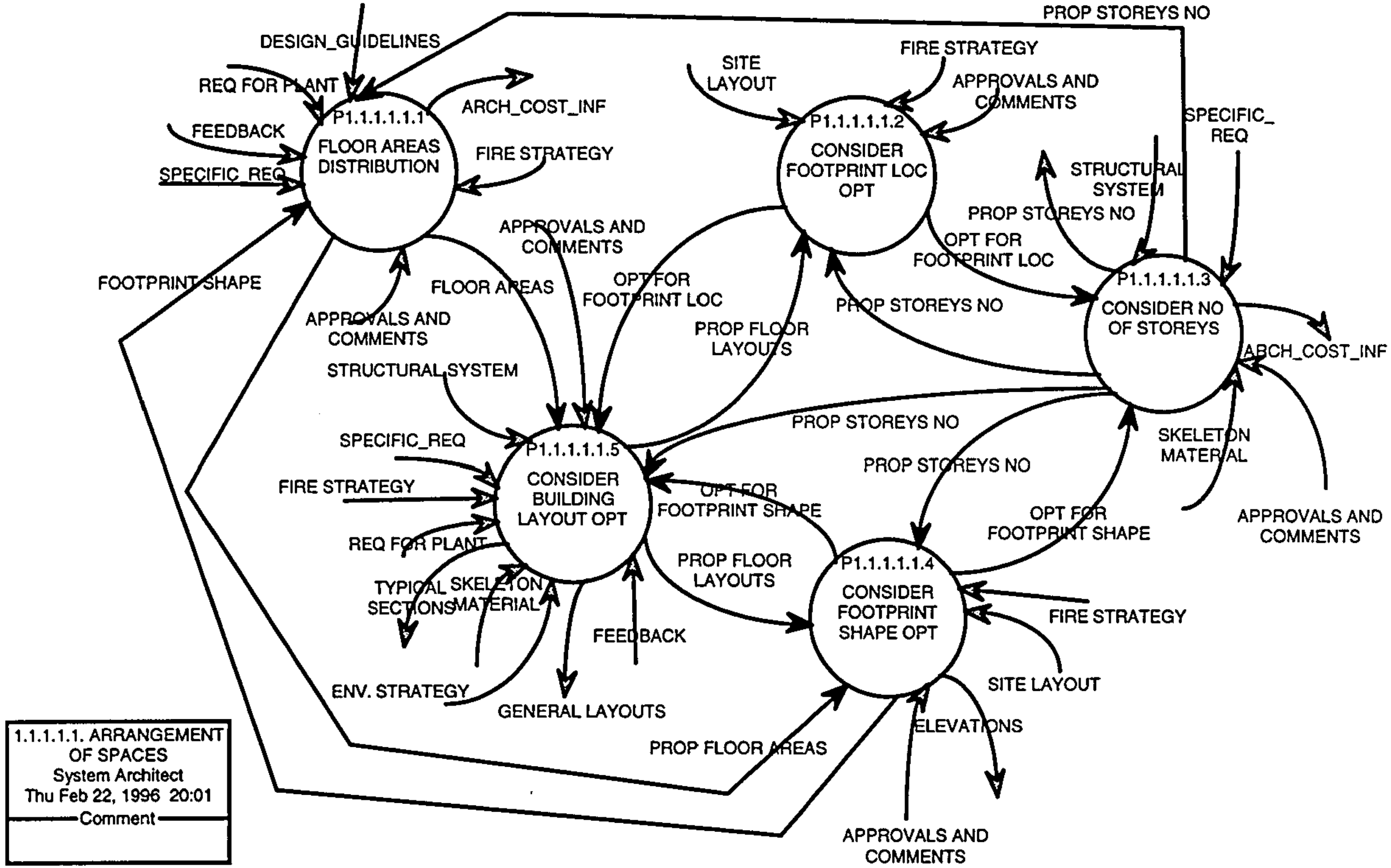
P1.1.3
SITE PLANNING
SITE_BOUNDARIES
SITE_LOCATION
SITE SURVEY
EXISTING DEVELOPMENTS
APPROVALS
PLANNING REGULATIONS
PRELIMINARY APPLICATION
SITE LAYOUT
GENERAL SOIL CONDITIONS
PARKING FACILITIES
LOADING& UNLOADING FACILITIES
SITE ACCESS
HANDINGOVER_DATE
APPROVALS AND COMMENTS
APP CONCEPT DESIGN REPORT
PHASING_REQ
TENDER_POLICY
P1.1.4
CONCEPT DESIGN REPORT PRODUCTIO
D GENERAL SOIL CONDITIONS
ARCH ROOF SCHEME
ESTIMATED SLAB THICK&STRENGTH
SKELETON MATERIAL
FOUNDATIONS TYPE
STRUCTURAL SYSTEM
TYPICAL SECTIONS
ENVELOPE REQ
GENERAL LAYOUTS
ELEVATIONS
PROP STOREYS NO
VENT SYSTEM
SPECIAL SYSTEMS
HEATING&PIPED SERV
POWER&LIGHT REQ
MECH ROOMS
COST PLAN
P1.1.7
ESTABLISH ENVIRONMENTAL STRATEGY
SPECIAL_LOADING_COND
P1.1.1
BUILDING CONCEPT DESIGN
ENV. STRATEGY
APPROVALS AND COMMENTS
SPECIFIC_REQ
DESIGN_GUIDELINES
BUILDING COST INFORMATION
P1.1.5
ESTIMATING COSTS
PRICES
BUDGET
D PRICES
1.1. CONCEPT DESIGN
System Architect
Wed Oct 18, 1995 11:49
Comment

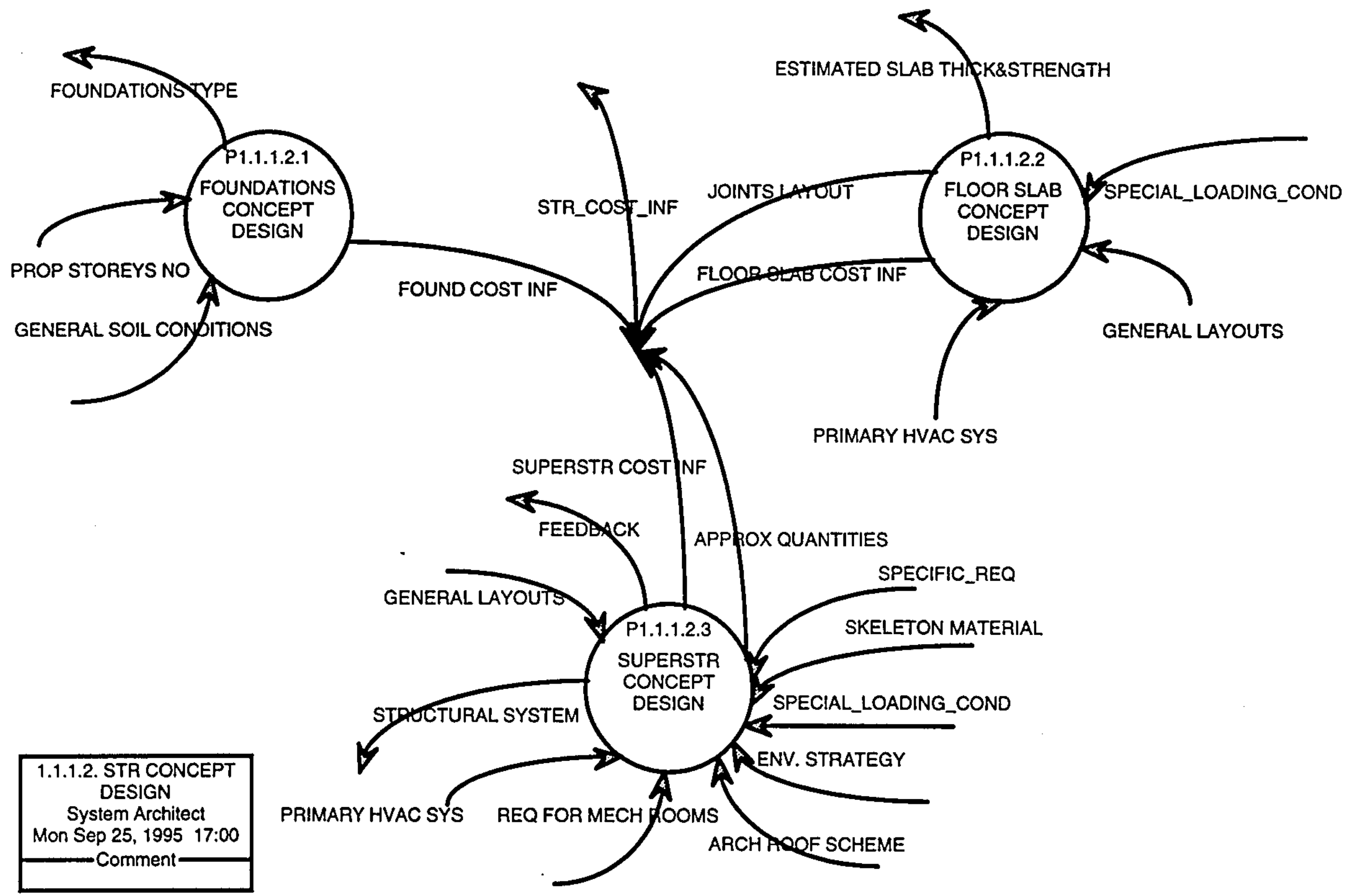


P1.1.1.1
ARCH CONCEPT DESIGN
P1.1.1.2
STR CONCEPT DESIGN
P1.1.1.3
SERV CONCEPT DESIGN
SKELETON MATERIAL
ARCH ROOF SCHEME
TYPICAL SECTIONS
ARCH_COST_INF
ENVELOPE REQ
SITE LAYOUT
PROP STOREYS NO
ELEVATIONS
ENV. STRATEGY
GENERAL LAYOUTS
APPROVALS AND COMMENTS
DESIGN_ GUIDELINES
SPECIFIC_REQ
ARCH ROOF SCHEME
STRUCTURAL SYSTEM
PROP STOREYS NO
FEEDBACK
SKELETON MATERIAL
GENERAL LAYOUTS
GENERAL LAYOUTS
FEEDBACK
GENERAL LAYOUTS
ENV. STRATEGY
REQ FOR PLANT
POWER&LIGHT REQ
SPECIAL SYSTEMS
MECH ROOMS
GENERAL SOIL CONDITIONS
STR_COST_INF
ESTIMATED SLAB THICK&STRENGTH
FOUNDATIONS TYPE
SPECIAL_LOADING_COND
STRUCTURAL SYSTEM
SPECIFIC_REQ
PRIMARY HVAC SYS
ENV. STRATEGY
REQ FOR MECH ROOMS
HEATING&PIPED SERV
SERV_COST_INF
VENT SYSTEM
SPECIFIC_REQ
APPROVALS AND COMMENTS
1.1.1. BUILDING CONCEPT DESIGN
System Architect
Wed Oct 18, 1995 11:53
Comment

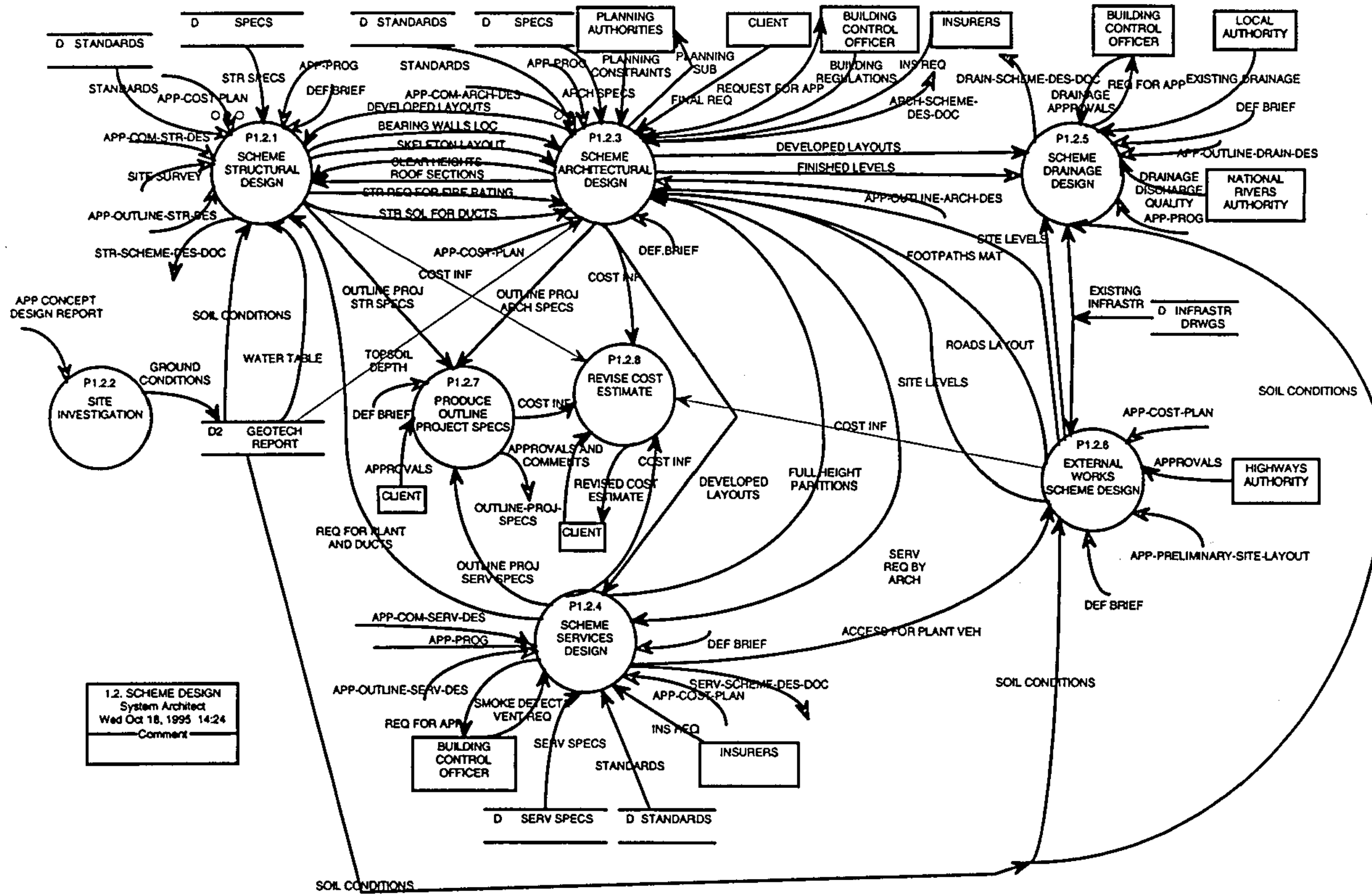


ELEVATIONS
SITE LAYOUT
STRUCTURAL SYSTEM
ENV. STRATEGY
BUILDING REGULATIONS
BUILDING CONTROL OFFICER
SITE LAYOUT
DESIGN_GUIDELINES
PROP STOREYS NO
P1.1.1.1.1
ARRANGEMENT OF SPACES
FEEDBACK
GENERAL LAYOUTS
FIRE STRATEGY
P1.1.1.1.5
ESTABLISH FIRE STRATEGY
TYPICAL SECTIONS
ELEVATIONS
GENERAL LAYOUTS
GENERAL LAYOUTS
SPECIFIC_REQ
REQ FOR PLANT
FIRE STRATEGY
APPROVALS AND COMMENTS
SKELETON MATERIAL
SKELETON MATERIAL
P1.1.1.1.4
ROOF ARCH CONCEPT DESIGN
P1.1.1.1.3
ENVELOPE CONCEPT DESIGN
APPROVALS AND COMMENTS
ARCH ROOF SCHEME
ARCH_COST_INF
ENVELOPE REQ
SKELETON MATERIAL
SKELETON MATERIAL
1.1.1.1. ARCH CONCEPT DESIGN
System Architect
Wed Oct 18, 1995 11:55
Comment

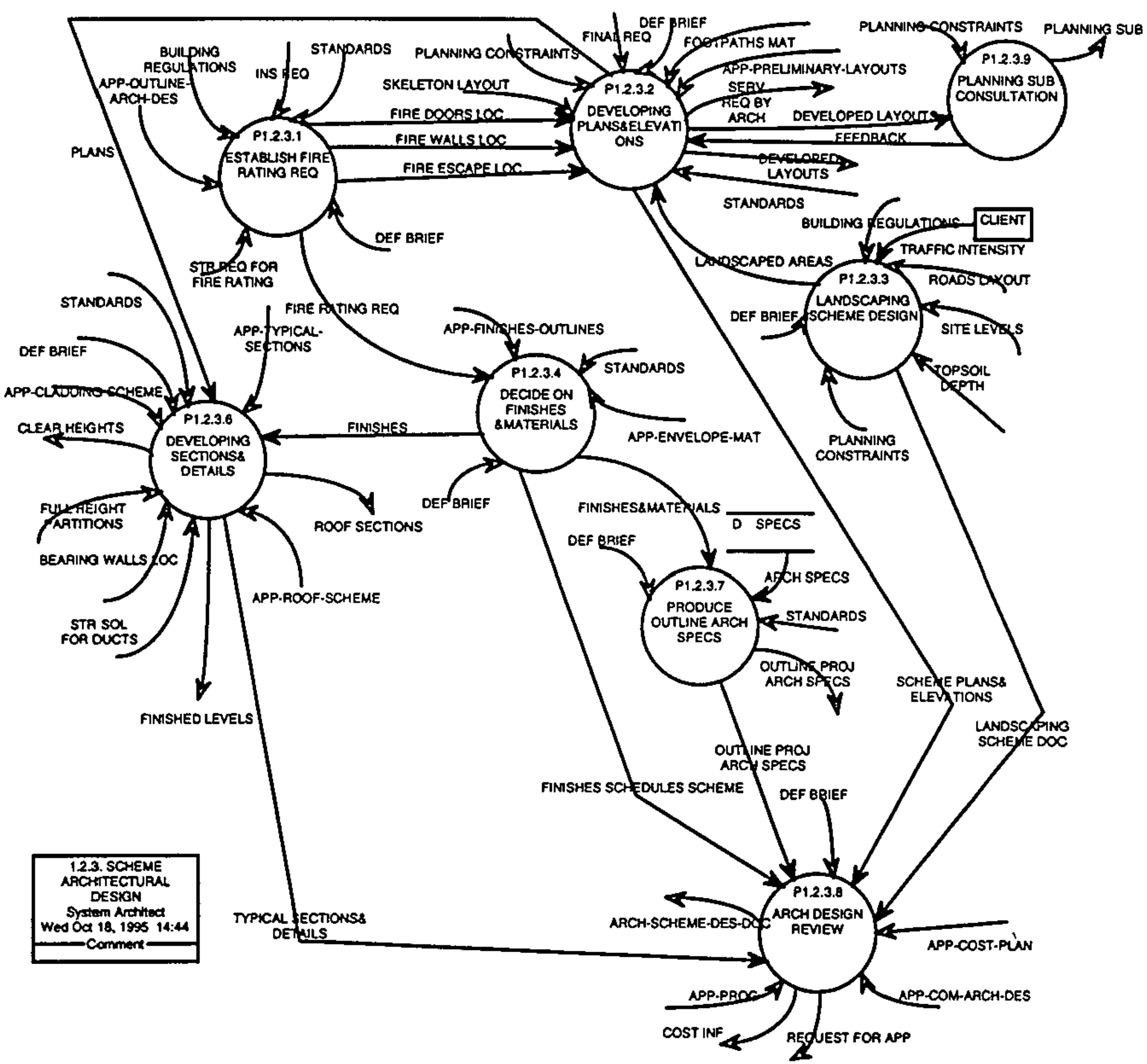

P1.1.1.1.1.1
FLOOR AREAS DISTRIBUTION
P1.1.1.1.1.2
CONSIDER FOOTPRINT LOC OPT
P1.1.1.1.1.3
CONSIDER NO OF STOREYS
P1.1.1.1.1.4
CONSIDER FOOTPRINT SHAPE OPT
P1.1.1.1.1.5
CONSIDER BUILDING LAYOUT OPT
DESIGN_GUIDELINES
REQ FOR PLANT
FEEDBACK
SPECIFIC_REQ
ARCH_COST_INF
FIRE STRATEGY
FOOTPRINT SHAPE
APPROVALS AND COMMENTS
FLOOR AREAS
OPT FOR FOOTPRINT LOC
PROP FLOOR LAYOUTS
STRUCTURAL SYSTEM
SITE LAYOUT
PROP STOREYS NO
SPECIFIC_REQ
SKELETON MATERIAL
OPT FOR FOOTPRINT SHAPE
TYPICAL SECTIONS
ENV. STRATEGY
GENERAL LAYOUTS
PROP FLOOR AREAS
ELEVATIONS
1.1.1.1.1. ARRANGEMENT OF SPACES
System Architect
Thu Feb 22, 1996 20:01
Comment

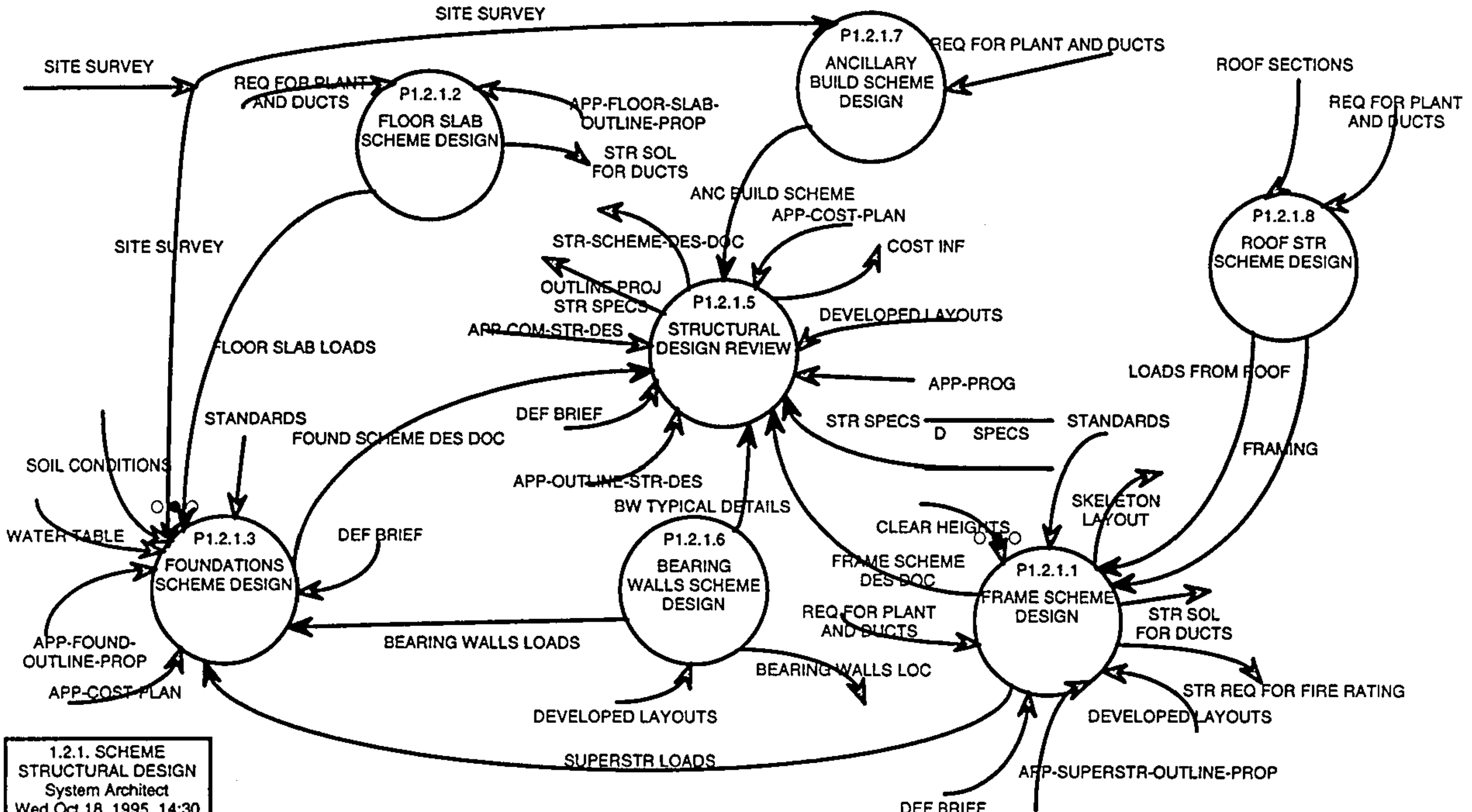


FOUNDATIONS TYPE
P1.1.1.2.1
FOUNDATIONS CONCEPT DESIGN
PROP STOREYS NO
GENERAL SOIL CONDITIONS
FOUND COST INF
STR_COST_INF
JOINTS LAYOUT
ESTIMATED SLAB THICK&STRENGTH
P1.1.1.2.2
FLOOR SLAB CONCEPT DESIGN
SPECIAL_LOADING_COND
FLOOR SLAB COST INF
GENERAL LAYOUTS
PRIMARY HVAC SYS
SUPERSTR COST INF
FEEDBACK
APPROX QUANTITIES
SPECIFIC_REQ
SKELETON MATERIAL
GENERAL LAYOUTS
P1.1.1.2.3
SUPERSTR CONCEPT DESIGN
STRUCTURAL SYSTEM
SPECIAL_LOADING_COND
ENV. STRATEGY
PRIMARY HVAC SYS
REQ FOR MECH ROOMS
ARCH ROOF SCHEME
1.1.1.2. STR CONCEPT DESIGN
System Architect
Mon Sep 25, 1995 17:00
Comment

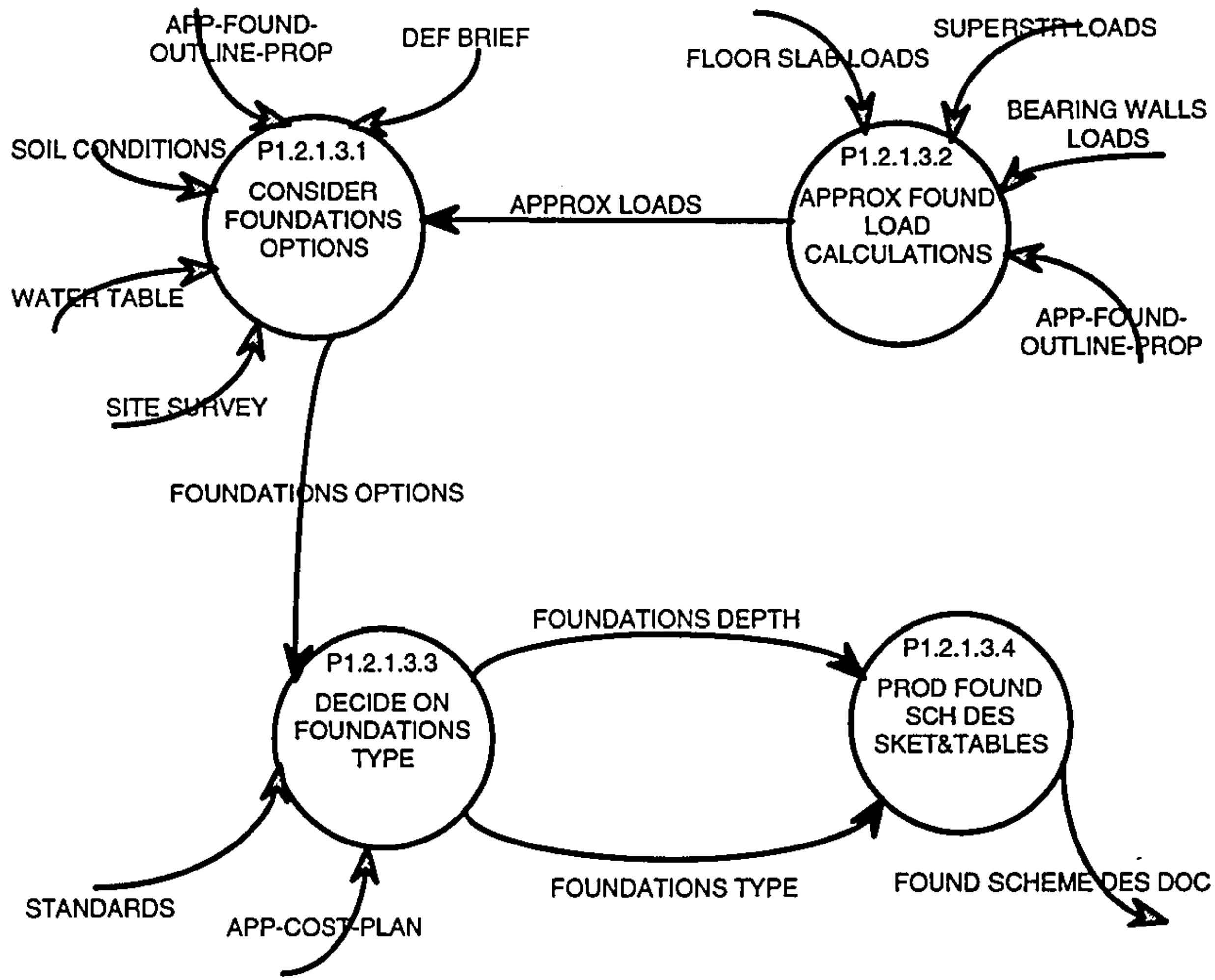


P1.2.1
SCHEME STRUCTURAL DESIGN
P1.2.2
SITE INVESTIGATION
P1.2.3
SCHEME ARCHITECTURAL DESIGN
P1.2.4
SCHEME SERVICES DESIGN
P1.2.5
SCHEME DRAINAGE DESIGN
P1.2.6
EXTERNAL WORKS SCHEME DESIGN
P1.2.7
PRODUCE OUTLINE PROJECT SPECS
P1.2.8
REVISE COST ESTIMATE
D STANDARDS
D SPECS
D STANDARDS
D SPECS
PLANNING AUTHORITIES
CLIENT
BUILDING CONTROL OFFICER
INSURERS
BUILDING CONTROL OFFICER
LOCAL AUTHORITY
NATIONAL RIVERS AUTHORITY
HIGHWAYS AUTHORITY
D INFRASTR DRWGS
D2 GEOTECH REPORT
CLIENT
CLIENT
BUILDING CONTROL OFFICER
INSURERS
D SERV SPECS
D STANDARDS
STANDARDS
APP-COST-PLAN
STR SPECS
APP-PROG
DEF BRIEF
STANDARDS
APP-PROG
PLANNING CONSTRAINTS
PLANNING SUB
ARCH SPECS
APP-COM-ARCH-DES
FINAL REQ
REQUEST FOR APP
BUILDING REGULATIONS
INS REQ
ARCH-SCHEME-DES-DOC
DRAIN-SCHEME-DES-DOC
DRAINAGE APPROVALS
REQ FOR APP
EXISTING DRAINAGE
DEF BRIEF
APP-OUTLINE-DRAIN-DES
DRAINAGE DISCHARGE QUALITY
APP-PROG
DEVELOPED LAYOUTS
FINISHED LEVELS
APP-OUTLINE-ARCH-DES
DEVELOPED LAYOUTS
BEARING WALLS LOC
SKELETON LAYOUT
CLEAR HEIGHTS
ROOF SECTIONS
STR REQ FOR FIRE RATING
STR SOL FOR DUCTS
APP-COM-STR-DES
SITE SURVEY
APP-OUTLINE-STR-DES
STR-SCHEME-DES-DOC
APP-COST-PLAN
COST INF
DEF.BRIEF
COST INF
OUTLINE PROJ STR SPECS
OUTLINE PROJ ARCH SPECS
SITE LEVELS
FOOTPATHS MAT
EXISTING INFRASTR
ROADS LAYOUT
SITE LEVELS
SOIL CONDITIONS
APP CONCEPT DESIGN REPORT
GROUND CONDITIONS
SOIL CONDITIONS
WATER TABLE
TOPSOIL DEPTH
DEF BRIEF
COST INF
APPROVALS
APPROVALS AND COMMENTS
OUTLINE-PROJ-SPECS
REVISED COST ESTIMATE
COST INF
DEVELOPED LAYOUTS
FULL HEIGHT PARTITIONS
COST INF
APP-COST-PLAN
APPROVALS
APP-PRELIMINARY-SITE-LAYOUT
DEF BRIEF
REQ FOR PLANT AND DUCTS
OUTLINE PROJ SERV SPECS
SERV REQ BY ARCH
ACCESS FOR PLANT VEH
APP-COM-SERV-DES
APP-PROG
DEF BRIEF
APP-OUTLINE-SERV-DES
SMOKE DETECT VENT REQ
REQ FOR APP
SERV-SCHEME-DES-DOC
APP-COST-PLAN
INS REQ
SERV SPECS
STANDARDS
SOIL CONDITIONS
SOIL CONDITIONS
1.2. SCHEME DESIGN
System Architect
Wed Oct 18, 1995 14:24
Comment

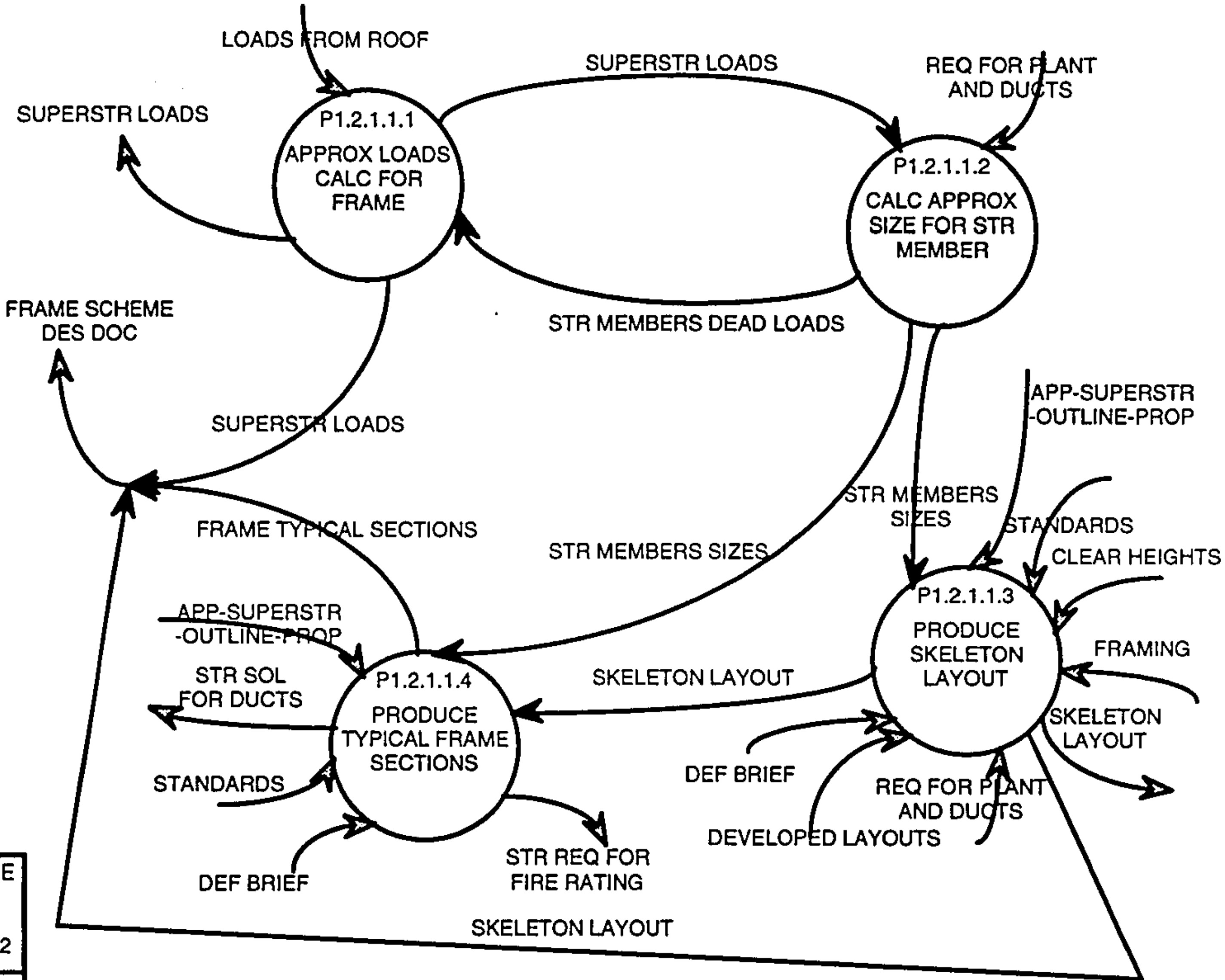

P1.2.3.1
ESTABLISH FIRE RATING REQ
P1.2.3.2
DEVELOPING PLANS&ELEVATIONS
P1.2.3.3
LANDSCAPING SCHEME DESIGN
P1.2.3.4
DECIDE ON FINISHES &MATERIALS
P1.2.3.6
DEVELOPING SECTIONS& DETAILS
P1.2.3.7
PRODUCE OUTLINE ARCH SPECS
P1.2.3.8
ARCH DESIGN REVIEW
P1.2.3.9
PLANNING SUB CONSULTATION
CLIENT
D SPECS
BUILDING REGULATIONS
APP-OUTLINE-ARCH-DES
INS REQ
STANDARDS
PLANS
PLANNING CONSTRAINTS
SKELETON LAYOUT
FINAL REQ
DEF BRIEF
FOOTPATHS MAT
APP-PRELIMINARY-LAYOUTS
SERV REQ BY ARCH
DEVELOPED LAYOUTS
FEEDBACK
PLANNING SUB
FIRE DOORS LOC
FIRE WALLS LOC
FIRE ESCAPE LOC
STR REQ FOR FIRE RATING
FIRE RATING REQ
TRAFFIC INTENSITY
ROADS LAYOUT
SITE LEVELS
TOPSOIL DEPTH
LANDSCAPED AREAS
APP-TYPICAL-SECTIONS
APP-FINISHES-OUTLINES
APP-CLADDING-SCHEME
CLEAR HEIGHTS
FINISHES
APP-ENVELOPE-MAT
FULL HEIGHT PARTITIONS
BEARING WALLS LOC
STR SOL FOR DUCTS
ROOF SECTIONS
APP-ROOF-SCHEME
FINISHED LEVELS
FINISHES&MATERIALS
ARCH SPECS
OUTLINE PROJ ARCH SPECS
SCHEME PLANS& ELEVATIONS
LANDSCAPING SCHEME DOC
FINISHES SCHEDULES SCHEME
TYPICAL SECTIONS& DETAILS
ARCH-SCHEME-DES-DOC
APP-COST-PLAN
APP-PROC
APP-COM-ARCH-DES
COST INF
REQUEST FOR APP
1.2.3. SCHEME ARCHITECTURAL DESIGN
System Architect
Wed Oct 18, 1995 14:44
Comment



1.2.1. SCHEME STRUCTURAL DESIGN
System Architect
Wed Oct 18, 1995 14:30
Comment



1.2.1.3. FOUNDATIONS
SCHEME DESIGN
System Architect
Wed Oct 18, 1995 14:33
Comment



1.2.1.1. FRAME SCHEME DESIGN
System Architect
Wed Oct 18, 1995 14:32
Comment

All Definitions

Name Description	Type	Date
ACCEPTABLE-NOISE-LEVELS	Data Element	02/06/94
ACCESS FOR PLANT VEH	Data Flow	29/06/94
ACCESS-EGRESS	Data Element	02/06/94
APP CONCEPT DESIGN REPORT *APP-OUTLINE-ARCH-DES+* *APP-OUTLINE-STR-DES+* *APP-OUTLINE-DRAIN-DES+* *APP-OUTLINE-SERV-DES+* *APP-PROG+* *APP-COST-PLAN*	Data Flow	31/08/94
APP SCHEME DES DOC *STR-SCHEME-DES-DOC+* *ARCH-SCHEME-DES-DOC+* *SERV-SCHEME-DES-DOC+* *DRAIN-SCHEME-DES-DOC+* *OUTLINE-PROJ-SPECS*	Data Flow	28/06/94
APP-BUILDING-MANAGEMENT-SCHEME	Data Element	01/09/94
APP-CABLE-ROUTING-SCHEME	Data Element	21/02/94
APP-CLADDING-SCHEME	Data Element	21/02/94
APP-COM-ARCH-DES	Data Element	17/02/94
APP-COM-SERV-DES	Data Element	17/02/94
APP-COM-STR-DES	Data Element	17/02/94
APP-COMPUTER--SERV-PROV-SCHEME	Data Element	25/02/94
APP-COST-PLAN	Data Element	21/02/94
APP-EMERGENCY-LIGHTING-SCHEME	Data Element	21/02/94
APP-ENVELOPE-MAT	Data Element	21/02/94
APP-ENVELOPE-SCHEME *APP-ENVELOPE-MAT+* *APP-ROOF-SCHEME+* *APP-CLADDING-SCHEME*	Data Structure	31/08/94
APP-FINISHES-OUTLINES *DURABILITY+* *REFLECTENCE+* *AESTHETICS+* *REQ-FOR-ANTISTATIC-MATERIALS+* *SOUND-REDUCTION*	Data Flow	15/08/95
APP-FINISHES-OUTLINES	Data Element	28/06/94
APP-FIRE-ALARM-SCHEME	Data Element	21/02/94

All Definitions

Name Description	Type	Date
APP-FIRE-FIGHTING-SCHEME	Data Element	21/02/94
APP-FLOOR-SLAB-OUTLINE-PROP	Data Element	28/06/94
APP-FOUL-WATER-DRAIN-OUTLINE-PR	Data Element	28/06/94
APP-FOUL-WATER-DRAIN-SCHEME	Data Element	21/02/94
APP-FOUND-OUTLINE-PROP	Data Element	28/06/94
APP-GAS-PIPING-SCHEME	Data Element	21/02/94
APP-HEATING&PIPED-SERV-SCHEME *APP-HVAC-SCHEME+* *APP-HEATING-SCHEME+* *APP-VENT-SCHEME+* *APP-SPECIAL-PLANT-SCHEME+* *APP-FIRE-FIGHTING-SCHEME+* *APP-POTABLE-WATER-PIPING-SCHEME+* *APP-GAS-PIPING-SCHEME*	Data Structure	28/06/94
APP-HEATING-SCHEME	Data Element	21/02/94
APP-HVAC-SCHEME	Data Element	21/02/94
APP-ILLUM-LEVELS-SCHEME	Data Element	21/02/94
APP-LIGHTED-AREAS-SCHEME	Data Element	21/02/94
APP-LIGHTNING-PROT-SCHEME	Data Element	21/02/94
APP-OUTLINE-ARCH-DES *APP-PRELIMINARY-LAYOUTS+* *APP-ENVELOPE-SCHEME+* *APP-TYPICAL-SECTIONS*	Data Structure	14/08/95
APP-OUTLINE-DRAIN-DES *APP-SURF-WATER-DRAIN-OUTLINE-PR+* *APP-FOUL-WATER-DRAIN-OUTLINE-PR*	Data Structure	28/06/94
APP-OUTLINE-SERV-DES *APP-HEATING&PIPED-SERV-SCHEME+* *APP-POWER&LIGHTING-SCHEME+* *APP-TELECOM&DATA-SCHEME+* *APP-SPECIAL-SYSTEMS-SCHEME*	Data Structure	28/06/94
APP-OUTLINE-STR-DES *APP-FOUND-OUTLINE-PROP+* *APP-FLOOR-SLAB-OUTLINE-PROP+* *APP-SUPERSTR-OUTLINE-PROP*	Data Structure	28/06/94
APP-POTABLE-WATER-PIPING-SCHEME	Data Element	21/02/94
APP-POWER&LIGHTING-SCHEME *APP-POWER-LOADS+* *APP-CABLE-ROUTING-SCHEME+* *APP-COMPUTER--SERV-PROV-SCHEME+*	Data Structure	28/06/94

All Definitions

Name Description	Type	Date
APP-LIGHTED-AREAS-SCHEME+ *APP-ILLUM-LEVELS-SCHEME+* *APP-EMERGENCY-LIGHTING-SCHEME*		
APP-POWER-LOADS	Data Element	25/02/94
APP-PRELIMINARY-BUILDING-LAYOUT	Data Element	28/06/94
APP-PRELIMINARY-LAYOUTS *APP-PRELIMINARY-SITE-LAYOUT+* *APP-PRELIMINARY-BUILDING-LAYOUT*	Data Structure	28/06/94
APP-PRELIMINARY-SITE-LAYOUT *SITE-ACCESS+* *LOADING&UNLOADING-FACILITIES+* *PARKING-FACILITIES*	Data Structure	28/06/94
APP-PROG	Data Element	21/02/94
APP-ROOF-SCHEME	Data Element	21/02/94
APP-SECURITY-SYSTEM-SCHEME	Data Element	21/02/94
APP-SPECIAL-PLANT-SCHEME	Data Element	28/06/94
APP-SPECIAL-SYSTEMS-SCHEME *APP-FIRE-ALARM-SCHEME+* *APP-SECURITY-SYSTEM-SCHEME+* *APP-LIGHTNING-PROT-SCHEME+* *APP-TELECOM-AND-DATA-SCHEME+* *APP-BUILDING-MANAGEMENT-SCHEME*	Data Structure	01/09/94
APP-SUPERSTR-OUTLINE-PROP	Data Element	28/06/94
APP-SURF-WATER-DRAIN-OUTLINE-PR	Data Element	28/06/94
APP-TELECOM&DATA-SCHEME	Data Element	21/02/94
APP-TELECOM-AND-DATA-SCHEME	Data Element	01/09/94
APP-TYPICAL-SECTIONS	Data Element	31/08/94
APP-VENT-SCHEME	Data Element	21/02/94
APPROVALS AND COMMENTS *APP-COM-STR-DES+* *APP-COM-ARCH-DES+* *APP-COM-SERV-DES*	Data Flow	21/02/94
ARCH-SCHEME-DES-DOC	Data Element	28/06/94
ARCH_COST_INF	Data Element	09/02/94
BUDGET	Data Element	09/02/94
BUILDING COST INFORMATION *ARCH_COST_INF+*	Data Flow	09/02/94

All Definitions

Name Description	Type	Date
STR_COST_INF+ *SERV_COST_INF*		
BUILDING REGULATIONS *FIRE-WALLS-LOCATIONS+* *PETROL-INTERCEPTORS-REQ+* *HOSE-REELS-REQ+* *ESCAPE-RAMPS-SLOPES+* *ACCESS-EGRESS+* *FIRE-SAFETY-REQ+* *ACCEPTABLE-NOISE-LEVELS+* *ENVIRONMENTAL-STANDARDS*	Data Flow	02/06/94
BUILDING_LAYOUT	Data Element	10/02/94
BUILDING_MANAGEMENT_SYSTEM	Data Element	01/09/94
CONCEPT DESIGN REPORT	Data Flow	31/08/94
DEF BRIEF *PROJECT_DESCRIPTION+* *STATEMENT_OF_OBJECTIVES+* *PROJECT_STRATEGY+* *EXISTING_INFRASTRUCTURES+* *PRINCIPLES_OF_DESIGN+* *PROJECT_PROGRAMME+* *ENVIRONMENTAL_OBJECTIVES+* *SCHEDULE_OF_DATA*	Data Flow	18/10/95
DESIGN_GUIDELINES	Data Element	01/09/94
DEVELOPED LAYOUTS *DIMENSIONED-LAYOUTS*	Data Flow	28/06/94
DEVELOPEDLAYOUTS *DIMENSIONED-LAYOUTS*	Data Flow	28/06/94
DEVELOPING SECTIONS& DETAILS *INCLUDES DEVELOPING SECTIONS&DETAILS FOR:* *GLAZING* *CLADDING* *BLOCK WORKS* *DRY PARTITIONS* *FALSE CEILING* *DOORS&DOOR SCHEDULES*	Process	24/02/94
DIMENSIONED-LAYOUTS	Data Element	28/06/94
DRAIN-SCHEME-DES-DOC	Data Element	28/06/94
DRINK-WAT-PTS	Data Element	21/02/94
ENVIRONMENTAL-STANDARDS	Data Element	02/06/94
ESCAPE-RAMPS-SLOPES	Data Element	23/02/94
EXISTING_DEVELOPMENTS	Data Element	08/02/94

All Definitions

Name Description	Type	Date
EXIST_SEWER_FACIL	Data Element	08/02/94
FINAL REQ *FROZEN_DESIGN_INFORMATION*	Data Flow	14/08/95
FINISHES_MATERIALS	Data Element	09/02/94
FINISHES_STANDARDS *QUALITY OF FINISHING MATERIALS TO BE USED*	Data Element	08/02/94
FINISHES_TYPES	Data Element	09/02/94
FIRE-SAFETY-REQ	Data Element	23/02/94
FIRE-WALLS-LOCATIONS	Data Element	23/02/94
FIRE_ALARM_AND_DETECTION	Data Element	09/02/94
FL_SLAB_UNIF_LOAD	Data Element	25/02/94
FOUNDATIONS TYPE *FOUNDATION_TYPE+* *FOUNDATION_DEPTH*	Data Flow	27/06/94
FOUNDATION_DEPTH	Data Element	08/02/94
FOUNDATION_TYPE	Data Element	08/02/94
GENERAL LAYOUTS *SITE_LAYOUT+* *BUILDING_LAYOUT*	Data Flow	28/06/94
HANDDRYERS-DIST	Data Element	21/02/94
HANDDRYERS-LOC	Data Element	21/02/94
HANDINGOVER_DATE	Data Element	08/02/94
HEATING&PIPED SERV *HEATING_PIPED_SERV_BUILDING+* *HEATING_PIPED_SERV_EXTERNAL*	Data Flow	28/06/94
HEATING_PIPED_SERV_BUILDING *RADIATOR_SYSTEM+* *MECH_VENT_TOILETS+* *HOT_AND_COLD_WATER+* *KITCHEN_EXTRACT*	Data Structure	28/06/94
HEATING_PIPED_SERV_EXTERNAL *PERIMETER_FIRE_RING_MAIN+* *INCOMING_POTABLE_WATER+* *INCOMING_FIRE_MAIN+* *SPRINK_TANK+* *PUMP_HOUSE+* *INCOMING_NAT_GAS*	Data Structure	28/06/94

All Definitions

Name Description	Type	Date
HOSE-REELS-REQ	Data Element	23/02/94
HOT_AND_COLD_WATER	Data Element	09/02/94
INCOMING_FIRE_MAIN	Data Element	09/02/94
INCOMING_NAT_GAS	Data Element	09/02/94
INCOMING_POTABLE_WATER	Data Element	09/02/94
INS REQ *REQ-FOR-FIRE-WALLS+* *FIRE-SAFETY-REQ+* *ENVIRONMENTAL-STANDARDS*	Data Flow	01/09/94
KITCH-FAC-LOC	Data Element	28/06/94
KITCHEN_EXTRACT	Data Element	09/02/94
LANDSCAPING SCHEME DOC *DRAWINGS+* *PLANT-TYPES+* *MAINTENANCE-PERIODS+* *DURABILITY-OF-HARD-LANDSCAPE*	Data Flow	15/08/95
LIGHTING *ILLUMINATION_LEVEL+* *FIXTURES_TYPES+* *CAR_PARK_ILLUMIN+* *FOOTPATHS_ILLUMIN+* *EMERGENCY_LIGHTING*	Data Element	09/02/94
LIGHTING-FIXT-DIST	Data Element	21/02/94
LIGHTNING_PROTECTION	Data Element	09/02/94
LOADING&UNLOADING-FACILITIES	Data Element	21/02/94
MECH ROOMS *MECH_ROOMS_SIZES+* *MECH_ROOMS_LOCATION*	Data Flow	09/02/94
MECH_ROOMS_LOCATION	Data Element	09/02/94
MECH_ROOMS_SIZES	Data Element	09/02/94
MECH_VENT_TOILETS	Data Element	09/02/94
OUTLINE-PROJ-SPECS	Data Element	28/06/94
PARKING-FACILITIES	Data Element	21/02/94
PERIMETER_FIRE_RING_MAIN	Data Element	09/02/94
PETROL-INTERCEPTORS-REQ	Data Element	23/02/94

All Definitions

Name Description	Type	Date
PHASING_REQ	Data Element	08/02/94
POWER&LIGHT REQ *POWER_SUPPLIES+* *LIGHTING*	Data Flow	09/02/94
POWER-SUPPLY-DIST	Data Element	21/02/94
POWER_SUPPLIES *VOLTAGE_TYPE+* *ESTIMATED_LOADS+* *CABLE_TYPES+* *DISTRIBUTION_SYSTEM+* *COMPUTER_UPS_REQ*	Data Element	09/02/94
PRELIMINARY-COLOUR-SCHEME	Data Element	28/06/94
PROCUREMENT_METHOD	Data Element	08/02/94
PROJECT BRIEF *SITE_BOUNDARIES+* *SITE_LOCATION+* *EXISTING_DEVELOPMENTS+* *PHASING_REQ+* *TENDER_POLICY+* *HANDINGOVER_DATE+* *SPECIAL_LOADING_COND+* *BUDGET+* *SPECIFIC_REQ+* *DESIGN_GUIDELINES*	Data Flow	18/10/95
PUMP_HOUSE	Data Element	09/02/94
RADIATOR_SYSTEM	Data Element	09/02/94
REQ-FOR-FIRE-WALLS	Data Element	01/09/94
SECURITY_SYSTEM	Data Element	09/02/94
SERV REQ BY ARCH *HANDDRYERS-LOC+* *DRINK-WAT-PTS+* *KITCH-FAC-LOC*	Data Flow	21/02/94
SERV-SCHEME-DES-DOC	Data Element	28/06/94
SERV_COST_INF	Data Element	09/02/94
SITE-ACCESS	Data Element	21/02/94
SITE_BOUNDARIES	Data Element	08/02/94
SITE_LAYOUT	Data Element	09/02/94
SITE_LOCATION	Data Element	08/02/94
SMOKE_VENTILATORS	Data Element	09/02/94

All Definitions

Name Description	Type	Date
SPEAKERS-DIST	Data Element	21/02/94
SPECIAL SYSTEMS *TELECOM_AND_DATA+* *FIRE_ALARM_AND_DETECTION+* *SECURITY_SYSTEM+* *LIGHTNING_PROTECTION+* *BUILDING_MANAGEMENT_SYSTEM*	Data Flow	01/09/94
SPECIAL_LOADING_COND	Data Element	27/06/94
SPECIFIC_REQ *SPATIAL_REQ+* *FUNCTIONAL_REQ+* *STRATEGIC_REQ+* *OPERATIONAL_REQ*	Data Element	18/10/95
SPRINKLERS-DIST	Data Element	21/02/94
SPRINK_TANK	Data Element	09/02/94
STR-SCHEME-DES-DOC	Data Element	28/06/94
STR_COST_INF	Data Element	09/02/94
TEL-POINTS-DIST	Data Element	21/02/94
TELECOM_AND_DATA	Data Element	09/02/94
TENDER_POLICY	Data Element	08/02/94
TYPICAL SECTIONS&DETAILS *TYPICAL-CLADDING-DETAILS+* *TYPICAL-GLAZING-DETAILS+* *TYPICAL-BLOCK-WORKS-DETAILS+* *TYPICAL-DRY-PARTITIONS-DETAILS+* *TYPICAL-DOOR-SECTIONS+* *PRELIMINARY-COLOUR-SCHEME+* *TYPICAL-FALSE-CEILING-DETAILS*	Data Flow	29/06/94
TYPICAL-BLOCK-WORKS-DETAILS	Data Element	28/06/94
TYPICAL-CLADDING-DETAILS	Data Element	28/06/94
TYPICAL-DOOR-SECTIONS	Data Element	28/06/94
TYPICAL-DRY-PARTITIONS-DETAILS	Data Element	28/06/94
TYPICAL-FALSE-CEILING-DETAILS	Data Element	28/06/94
TYPICAL-GLAZING-DETAILS	Data Element	28/06/94
WARM_AIR_SYSTEM	Data Element	09/02/94

APPENDIX X

THE FEEDBACK DOCUMENT

Feedback Document For the Developed Tools

Organisation:
Name:
Position:

Problem	Importance (1-10) Conc/ Schem. des	Importance (1-10) Detailed des	Suitability of tools to provide the solution (1-100)
Assessing the impact of missing Information			
Assessing the impact of assuming information			
Assessing the impact of phased release of information			
Assessing the impact of different levels of information quality			
Assessing the impact of gate keeping of information			
Assessing the impact of uncertainties and carrying out risk analysis			
The problem of iteration			
Resources management			

Do you believe the application of these tools will help to improve design management?

Can you suggest any modifications to the developed tools? If yes please specify.

If the answer to the above question is yes how important, in your opinion, is this modification to enhance these tools (1-100)?