RICE UNIVERSITY

ARCHITECTURAL PROGRAMMING: Problem Definition and its relation to Design Process

PETER I. KARP

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER IN ARCHITECTURE

Houston, Texas

May, 1969
Abstract

ARCHITECTURAL PROGRAMMING:
Problem Definition and its relation to the
Design Process

Peter I. Karp

There is, currently, considerable evidence of confusion
and conflict within the profession of architecture. This
conflict stems from three major factors. These are:

i) The Technological Age

ii) The Complexity of Problems

iii) The Client.

Each of these is examined; the conclusion being that for
architecture to develop, defined Process or Method is needed.

Scientific Method is then examined to find the roots of Process.
The argument is that architecture can, and must, learn Method
or Process from the Technologies. The conclusion is that for
any Process, the first step is the initial definition of the
problem.

This problem statement (with respect to Architecture) is dif-
ficult, since it involves non-quantifiable goals statements
and concepts. The Process whereby this Problem Statement is
achieved has been defined as Programming.

This definition involves Areas of Concern and results in
Criteria for Programming, which are:
These criteria are then used for evaluation of the Case Study. The first part of the case study examines, in detail, the Programming Process of a large architectural firm. Part two examines four other Programming Processes, in less detail.

The study is then evaluated by re-stating the definition of Programming, and describing various potential uses of Programming and concludes in the statement of two over-riding considerations for Programming. These are:

1) The Statement of the Problem is the product of Programming.
2) Programming and Design are Processes within an overall Project Delivery System. The envelope which contains them is Management; and their interface is the Statement of the Problem.

The appendix contains illustrations of techniques used by the firm for the specific case study, together with documentation and evaluation of three different uses of their Programming Process.

A comprehensive bibliography completes the document.
TABLE OF CONTENTS

Section A: INTRODUCTION

1.0 Architectural Programming. .......... 1
2.0 Aims of the Study. ................. 2
3.0 Scope of the Study ................. 3

Section B: HYPOTHESIS

4.0 Architecture in a Technological Age. .. 6
5.0 Conclusion ......................... 17

Section C: SCIENTIFIC METHOD

6.0 The Scientific Approach .............. 20
7.0 The Gulf: Humanities - Technologies. 21
8.0 The Systems Approach. .............. 31

Section D: PROGRAMMING: DEFINITION

9.0 Generally .......................... 42
10.0 Areas of Concern ................... 43
11.0 Conclusion ......................... 61

Section E: CASE STUDY

PART ONE: (Specific)

12.0 Large Architectural Firm ............ 65
13.0 CRS Programming. .................. 67
14.0 Methods Used ....................... 78
15.0 Conclusions ......................... 81

PART TWO: (Generally)

16.0 Other Programming Systems. ........ 85

Section F: ASSESSMENT

17.0 Programming and the Profession ....... 95
18.0 Programming: The Final Overview .... 96

Section G: APPENDIX I

19.0 Case Study Photographs ............. 99

APPENDIX II

20.0 Caudill Rowlett Scott ............... 102
21.0 BIBLIOGRAPHY ..................... i.
Section A: INTRODUCTION

1.0 Architectural Programming.

1.1 Choice of subject for study. For some time now, the author has been interested in design process, as applied to architecture. This interest stems from a realization of certain inadequacies in his own make-up which are strongly related to some of the problems that are facing architects today. The choice of subject is related to the author's position at the end of a period of study, with respect to practising architecture in the near future.

It was felt that by defining a position and an attitude to the design process, the author would be best prepared for his future role as practitioner.

Many journal articles and books surveyed over the last several years have led to an increasing awareness of the conflicts and inadequacies of the profession. It should be noted that almost as many articles point out the progress of architecture, and if the hypothesis seems to be rather negative it is because the author is attempting to define and study areas of concern and conflict in order that he may better understand what progress has been made

* See Bibliography.
and where the problem areas lie.

Any exaggeration of the problems is intentional in that by exaggeration these conflicts can be best examined and responded to. The overall aim of the hypothesis is however distinctly optimistic in that it leads to the definition of a point of departure, for the author, in the process and production of architecture.

2.0 Aims of the Study.

2.1 The intention is to make a case for, and point out that, the central problem facing an architect today is the initial definition of the problem and the way this relates to the process of design employed and thus the effect upon the final product, the building.

Problem definition is concerned largely with conceptual and goals statements, since these are qualitative rather than measurable criteria the problem is compounded. The architect's problem remains the searching out and definition of the 'unique' quality of each design problem so that his overall design concept can reflect the nature and purpose of the final building. He must be concerned, in the first instance, with accurately setting up the limits for his design solution.
In this study this initial statement of the problem has been defined as Programming*. The intention is to achieve an overview of Programming, and the criteria used in this process. Thus the thesis will deal with principles and generalizations and result in a point of departure rather than a solution, which would be a separate study, and is the logical next step for the author.

3.0 Scope of the Study.

3.1 The method of study employed is, generally, hypothesis based on various references, citing principles and generalizations rather than specifics, to find common threads and differences, both related to architecture and to the broader field of scientific method in problem solving.

3.1.1 Section B outlines the confusion and conflicts involved with architecture, under various sub-headings. The conclusion is that a method and process is needed; and this leads into Section C.

3.1.2 Scientific method as related problem definition and problem solving are cited. The argument being that architects have much to learn from the technologies, and defines (*)

* A more precise definition of this term is central to Section D.
what it is that architects can learn.

Central to this section is the argument 'The Humanities-Technologies Gulf', as postulated by Lord Snow\(^1\) and elaborated by Mallows\(^2\). The author leans heavily on Professor Mallows' thesis, since a study of this 'Gulf' from first principles is a vast study in itself. The purpose here is to point out the similarities of approach on either side of this 'Gulf', arguing that the architect will not have to change his whole method of work in order to bring system to his decision making process.

The section then examines "Scientific Method" in general, points out its central features and concludes that for architects to employ system the first pre-requisite is a clear definition of the problem.

3.1.3 Section D defines Programming and formulates criteria for problem-seeking and the roles of the individuals or groups involved. First in a general sense and then some methods are examined in broad outline. The relevance of Programming is defined as well as some of its potential uses. This is the synthesis of the thesis argument and formulates criteria for the case study which follows.

\(^1\) Snow, C. P. (1964).
\(^2\) Mallows, E. W. N. (1965).
3.1.4 Section E examines the Programming method of a large firm of architects-planners; by describing their approach and evaluating it by means of an appendix which cites three different uses of the system. Four other Programming processes are examined, in a less specific manner.

3.1.5 Section F is an evaluation of the study in terms of the stated goals. The conclusion consists of criteria for Programming as well as an overall view of the value of the study.

3.1.6 Section G is the appendix (cited above) and a bibliography of sources of quotations as well as a listing of all books and periodicals used in the study.
Section B: HYPOTHESIS

4.0 Architecture in a Technological Age/Society.

4.1 There is, currently, much published on the confusion within the profession of architecture, the lack of positive answers from the architect and the confusion created by the misunderstanding of the architect and his work. "People don't want what architects want"\(^1\) is one of the favourite cries. Very often architects are excluded from the decision making process - they are thought of as either too arty to wrestle with real problems; or else merely as tools - people to get things done - to draw the plans.

More than a problem of misunderstanding, this seems to be to some extent a vote of no-confidence in architects, rather than architecture per se. The architect is very often accused of creating monuments to his own ego rather than really trying to discover and satisfy the needs of his client or the users of his buildings.

Architects do not present anything of a unified front, either to these attacks, or in their approach to problems. The "prima donna" syndrome lingers in this field more than

---

1. Personal experience in Sociology seminars. Rice University, Fall 1968.
any other, the most common excuse being the ignorance of the general public.

"The present situation is confused and puzzling. From the client we hear constant complaints about the architect's lack of ability to satisfy him, from a practical as well as from an aesthetical and economical point of view. The authorities give us to understand that, it is often doubtful whether the architects are qualified to solve the problems which society poses. And the architects themselves disagree on issues so fundamental that their discussion must be interpreted as an expression of groping uncertainty. The disagreement does not only concern the so-called 'aesthetic' problems, but also the fundamental questions of how man should live and work in buildings and cities."²

It is clear, however, that the architect's problems have a qualitative rather than a quantitative base - that is to say architecture is not lacking a technological base for buildings - it is possible, physically, to erect almost any structure or form that we desire - the hang-up is the inability to understand and interpolate the needs and desires of the users of our buildings.

"But advances in the material product of architecture have been at the expense of understanding the process of design and the extension of design theory."³

It is seldom when a client or user can look at his building and recognize the solution of the problems he foresaw. He had to try and understand his problems when he recognized the need to build, and now he must begin all over and attempt to understand the building. It is more than the "education" of the client and user; in this complex age more is demanded of the architect and if he fails to deliver he will be increasingly relegated to the position of "...a 'necessary evil' with the sole task of trimming the ideas' of the client."4

What are the origins of this situation; and where do the solutions lie? Only by examining the roles of the participants will we be able to identify the causes, and thus begin to prescribe the solutions. It would be nihilistic to lay all the blame on the shoulders of the architects, but it would be of no value at all to try and deflate the new responsibilities facing architects.

4.2 The nature of the problem. What is new or unique about
a) the technological age;
b) the complexity and size of projects;
c) the client-architect relationship.

4.2.1 The technological revolution of the past few decades has rubbed off onto almost every sphere of our life and daily activities. The mass media, the communication networks and the problem solving tools that were little more than dreams a few years back, are hard realities now. The extent to which these developments have affected our daily lives should not be underestimated.

"In the face of growing interest in the 'system approach', many communities and agencies are evaluating the applicability to their own circumstances of these techniques that purport to improve the basis for decision making, policy construction, and program design and execution."5

Much less do we now rely on intuitive decision-making. There is much more information available as a base for these decisions and we also have the tools necessary for this information to become digestable. Investments are larger and are implemented more swiftly and consequently the decision maker is asking for much more information before he decides, and for that information to be presented to him in a succinct and concise form.

The architect as a decision maker has fallen palpably far behind. The architect, in a technological age, has made little use of the knowledge available to him.

"The architect who proposes to run with technology knows now that he will be in fast company, and that, in order to keep up, he may have to emulate

---

the Futurists and discard his whole cultural load, including the professional garments by which he is recognized as an architect. If, on the other hand, he decides not to do this, he may find that a technological culture has decided to go on without him."

The architect is being asked many and more precise questions today, and it is becoming increasingly difficult for him to hide in a "cloud of creativity". Sophisticated clients are asking precise and incisive questions. As a result of the sophistication of their decision-making in other fields they are demanding hard clean answers to their questions.

"What is worse, in an era that badly needs designers with a synthetic grasp of the organization of the physical world, the real work has to be done by less gifted engineers, because the designers hide their gift in irresponsible pretension to genius." 7

Clearly the analogy should not be taken too far - the architect is faced with qualitative rather than measurable problems - but unless he formulates, and works under conditions which allow and foster communication he will be relegated to the role of skilled worker.

It may be that this is overstating the case, but when a businessman or a school-board is facing real issues, that they can measure, what is to prevent them from acting on the advice of one of their quantitative consultants and

using the architect as an implementor, rather than a
decision maker?

Only when the architect learns a means for a) understanding
the sophisticated demands of his client and b) for communi-
cating his ideas and his approach to a problem to the
demanding client, will he regain and retain his seat on
the technological bus. The point is that the architect
can no longer have several parts to his kit. (For example,
one which is "economics" - how he intends to meet the
budget and another, of "aesthetics" or "architecture") He
must be able to show how each of these is a part of a
larger whole - and how these parts are interrelated and why
they cannot be considered separately. He must be able to
show how he has tied them together, and where the joints
and overlaps are.

"The modern designer relies more and more on his
position as an 'artist' on catchwords, personal
idiom, and intuition - for all these relieve him
of some of the burden of decision, and make his
cognitive problems manageable. Driven on his
own resources, unable to cope with the complicated
information he is supposed to organize, he hides
his incompetence in a frenzy of artistic indivi-
duality."^8

The architect's approach has not, largely, kept pace, and
he has lost the confidence of his client. The communication

gap is increasing, and the architect as the hired professional, is losing his role in the decision making process because he can no longer match the sophistication of his client in the definition and solution of problems. The society demands method and system in problem solving, architecture must recognize and respond to this.

"It is my belief that the architect, who is the professional supposed to provide design solutions to environmental problems, has little knowledge on how to deal in abstract with the structure of problems as such, and, moreover, in spite of the fact that his solutions will ultimately adopt physical form, he has little understanding of many of the meanings of the forms which he manipulates."  

4.2.2 Complexity and size of projects. Architecture has to face, and respond to, a new scale and order of problems. The architect is faced with problems not only larger, but often of a nature different from those of the past. (Feasibility studies, eg., which may or may not lead to building projects are being done for almost all building types.) Very often these studies are not given to the architects, but are done by urban economists or specialized analysis groups. The results of these studies then become binding upon the architect. He has to work within a framework set up by others, by specialists with different goals and priorities.

If the architect is to perform a decision making function

---

at this stage he must have demonstrable techniques that can be used in collaboration with various specialties. It seems logical that the architect, as the innovator in the project itself, should play, at the very least, an overall, "steering" role in commissions of this nature.

Very often in this situation the architect can argue only on qualitative grounds, with people orientated to numbers, graphs and computers. (When he is able to demonstrate a rationale, a method by which he arrives at his set of criteria, he will be listened to with equal, at least, emphasis.) Unless a professional can show a rational basis for qualitative, or value-judgements, he will have to hide behind his professional mask and will have only limited credibility. Only when he achieves this understanding will he be able to become a full member of this decision making situation - and only then will he be able to challenge the numbers and graphs of the economists and managers.

"The public does not easily understand that issues such as the relations between technics and form, or form and function, really are important." ¹⁰

As an extension of this, then, the pure size and complexity of the problems facing architects is, advancing and changing constantly. More factors have to be considered,

even at the 'small project' level.

Large-scale development is becoming the order of the day. No longer is it a part of a city block, it is the whole block - or perhaps two or three blocks that are being developed. Naturally the architect has to respond to these new forces. Traditionally the architect has fallen back on increased quantities of tracing paper to cope with the new influx and complexity of these situations. To remain a decision maker he will have to evolve methods of organizing and digesting this information in order to derive meaningful solutions. Increased number of specialists will have to be consulted and their inputs directed and channelled - generalizations will have to be made - but generalizations that can be measured and traced to their origins for amplification or change.

The architect has to have the ability to organize and utilize the information surrounding him rather than selecting only that which fits. Preconceived ideas have no relevance in this new order of problems. "Wonderful solutions to the wrong problem" have had their day, and the day has been long. These monuments to an architect's 'inspiration' no longer hold water.

Architects have to refrain from "leading by the nose".
They must lead by skill and initiative, in all senses of the words.

"Architects are not prepared for the immensity of the task, for the plethora and diffusion of information overwhelms most of them and they cling to handicraft methods of design and production, not having learned to augment their mental processes through the ordering of the design processes and the expansion of their intuition."\(^{11}\)

4.2.3 We have already alluded to the new sophistication of the client. He asks (and expects precise answers to,) more questions and he is more capable of measuring the responses he gets.

But who is this client? He is no longer the patron with money in the bank who wants a building. He now has many faces, not all of them looking in the same direction. He is now the "Building Committees" or the "Building Project Management Groups" or even the multi-purpose board of directors of a company or institution. He has many hands pulling tight on all sorts of reigns. This is one of the new complexities the architect must order.

This "client" has brought a new factor to the architect's attention. He is now concerned with the user or occupant of his building, and how the architect intends to measure

\(^{11}\) Jackson, B. (1966) p. 47.
and allow for this variable.

We have heard for years that 'buildings are for people' but never before has the user become such a central figure in the demands of the client. School boards want to try new methods of education, developers want a building that will have all the stores doing thriving business, city corporations want thriving centres for their cities. These people are no longer satisfied with photographic models. They are concerned, for various reasons to be sure, about the people in their buildings. How will the building react to different occupants, uses and users.

The architect thus has a variable to measure that has never been so central an issue before.

It seems strange and even a little disillusioning to think that users are a new concern for architects. They are, in that he has now to measure and respond to human behaviour as never before. It is a new level of sophistication that society has reached and the architects will have to meet this, and on all fronts. Architects must work closer with client and user - he has to recognize them as never before. The challenge of this new limit can only improve architecture, but in order that the limit be viable the architect must, first, be able to order and understand it.
"The architects of the future unquestionably will have to have greater sensitivity to human needs and values."\(^\text{12}\)

5.0 Conclusion.

5.1 Communication. The problems which we have cited should not be considered in a negative or pessimistic way. The aim of the preceding was to identify the roots of the architect's dilemma. In order that we might resolve these problems we must look at their manifestation and then try to see where the solution lies.

Architecture today is suffering from a communication gap. The architect must be able to communicate with various disciplines and individuals.

Contemporary society is essentially fast-track and relies on the rapid absorption and dissemination of information. Change is of the essence. Feedback, or the implications of changing forces in our society and way of life, is now being measured and considered in all that we do.

It is clear that for architecture to assume a central role, it must become more of a part and contributor to this

system than it has been in the past. Architecture is the logical resolution of a series of unequal forces in a physical form and must, in its formulation reflect this. Even the architect who orders by means of reams of tracing paper is, to a degree, a logical decision maker. The degree to which his design responds to logical progression is a function of the personality involved.

Architects claim logical thought but seem loath to recognize any real system or method in their way of working. Most architects have a system, albeit few are "efficient" in the accepted sense.

"Trial-and-error design is an admirable method. But it is just real world trial and error which we are trying to replace by a symbolic method, because real trial and error is too expensive and too slow."13

What is needed is a recognition, firstly, by the architects of their own methods, and secondly efforts to refine and record these methods. Until architects can present methods as a core of their decision making process they will be a) "left out" to an increasing degree and b) experience an inability to cope with more complex and demanding situations.

Architects must define and study their methods, and re-

organize them to allow for the management and utilization of the new and complex inputs. They must then be able to clearly define and explain, to client, user, and specialist consultant alike, the nature of their decision making process.

Mutual understanding is a prerequisite to co-operation and thus implementation of the valid generalizations necessary for the creation of a physical environment. Architecture needs systematic method.

What then is the essence, what are the essentials of 'method' and 'systems' and how might these best be related to architecture. What, for architecture, is the central issue (or issues) upon which to build this communication which it lacks?

"Design requires a language, tools for communication and means to store information. Little knowledge is exchanged between designers in any regular fashion.

The design community fails to systematically extend its knowledge because it has never agreed or defined what the proper concern of design is, and thus it has never defined a basic framework of information."14

Section C: SCIENTIFIC METHOD

6.0 The Scientific Approach.

6.1 The validity of scientific method is clearly a characteristic of the society in which we operate. Systems analysis, operations research and like technologies are being more widely used and sought after as a means of gaining a more precise insight and more definitive answers to the complexity of present day problems.

Problems themselves have indeed become more complex primarily as a result of the application of scientific analysis. We are now able to consider more variables together and consequently are able to broaden the scope or limits of our problems.

There exists, however, a schism between the technologies and the humanities on the validity of this systematized examination. Architects are often skeptical if not overtly antagonistic to this type of problem solving; the charge being that the use of any precise methodology or process stifles, or even does away with, creativity. The gulf that exists between the technologies and the humanities is considerable, but there is certain evidence to suggest that a close and careful examination of the way decisions are
made on either side of this gulf, will show a similarity of approach not inconsistent with close cooperation and understanding.

Architects are reluctant to look at how they work and how the technologies work - they are "not interested". Technology is their tool - it is only information, nothing more.

"The awareness of the problems has reached a clamor, but the designers still take the simple approaches to problems and have not yet realized that some things are too complex to grasp and order by random and intuitive processes."  

An examination of this gulf, leading to an examination of the systems approach in general is invaluable in identifying the criteria for a valid design process. The direction is clear; the sophistication of our society in terms of decision making is patent. That architecture evolve a process of design which can match this sophistication is inevitable. The question is what are the roots of this process, where does it come from and how should it be used?

7.0 The Gulf: Humanities Technologies*.  

---

* The argument outlined here is derived from Mallows, 1965, and (since it is only supplemental to the present thesis) relies strongly upon it.
7.1 The nature of the gulf: Historically the schism between theorists and practical craftsmen stretches at least as far back as the days of Aristotle.

"...the possessor of experiences is recognized as wiser than the possessor of any form of sense perception, the artist as wiser than the mere possessor of experiences, the master craftsman than the manual worker, the speculative sciences than the productive."\(^2\)

Also in describing theatre audiences he says

"...of two kinds...one of free and educated men, the other a vulgar crowd of mechanics and day labourers and the like."\(^3\)

Thus at the very start of "Western Civilization" we find definition of the differences between Technologies and Humanities, and that this is both inevitable and desirable. The Renaissance of the sixteenth century was largely Humanitarian; art, sculpture, literature and architecture, placing the purveyors of this new knowledge on a distinct, and higher plane, from the "do-ers". The Scientific Revolution of Galileo, Descartes and Newton came when the Humanitarian Renaissance was already starting to decline.

Technology, or applied Science, lagged even further behind, and in its greatest moment, the Industrial Revolution, was frowned upon by Science as rather vulgar and lower in

\(^2\), \(^3\) Aristotle, quoted in Mallows, E. W. N. (1965) p. 3.
stature. The pure Sciences moved over to another portion of the higher plane, still apart from the Humanities but both far "superior" to the practical, constructive, applied scientists. The Scientific method of technologists today is still considered a lower form of thought than pure science. And Pure Science is considered of a lower, or different type of intellect (depending upon which side you stand), than that of the Humanities. The schism here prompted Lord Snow's lecture of 1959, "The Two Cultures", where he outlines this gap. Today "scientific method" in the analysis of real problems is gaining tremendous momentum and popularity in our society. We are, as has already been outlined, becoming more and more systems oriented. But architects as "creative artists" and "Humanists" hold that this is to be avoided.

"The architects have shown themselves rather unwilling to work out a theoretical basis for their field, mostly because of the prejudice that theory kills the creative faculty."\(^4\)

There are however, on examination, some very very powerful parallels in the method used, traditionally both by the Humanist and Technologist, and the contention is that we should understand and recognize these parallels, and understand our own methods in order that we can find some

---

4. Snow, C. P. (1964)
common ground for communication and understanding.

Asking an architect to become more scientific, does not mean throwing off his traditional way of working. It means recognizing how he works and understanding how similar this is, in basic concept, to those "heartless" engineers and technologists.

(This is two-sided - the Technologists are as much "at fault" as the Humanist but this paper is exaggerating the role of the Humanist in a Technological age and consequently the argument might read as "all the fault of the architect" - this is not so. The initial exaggeration is for the purposes of the thesis - "How can architecture relign?")

7.1.1 Scientific Investigation. Based on the writings of W.J.B. Beveridge (1960), Emile Boirac (1950), P. Freedman (1949) and others as outlined by Mallows⁶ it is possible to outline scientific procedure as follows:

a) to state an objective
b) to observe the facts relevant to the objective
c) to group the observed facts in categories and to interrelate these categories
d) to form generalization or over-all patterns from

---

these categories and their interrelationships, and so arrive at some interpretation or conclusion. This stage may or may not include further verification of the conclusion.

On examination it is possible to illuminate these stages as hypotheses, observation, illumination, intuition or imagination. Without a clear statement of intent the system has no strength. Clearly stating the objective allows for selection and unearthing of the pertinent facts as well as the ability to recognize redundant facts. This hypothesis may well change or become more amplified as a result of the second stage, illumination of the objective. This grouping of the facts allows us to understand more clearly and fully what the implications of the objectives are. This stage reduces the data to manageable form, leading on to the fourth stage Patterns, or generalizations based on intuition can only come from an understanding of the forces of related groups of facts as seen through the overall framework of the objective. It is within this fourth stage that creativity is housed. That single unifying act which scientists are often loath to recognize as creativity, (preferring to define it as inevitability, with respect to the method) is a part of the fourth stage.

"It is the construction of these patterns that Scientists and Mathematicians use intuition
and imagination and admit that the scientist approaches the artist."  

7.1.2 The Study of Man. The Humanist studies man and his behaviour rather than facts and phenomena of the physical world. Mallows defines the Humanist's process of investigation as having five stages: -

a) Mental Freedom (curiosity)
b) Mental Discipline (orderly thought)
c) Evaluation
d) Generalization (from particulars)
e) Creation (of new patterns)

The first is the ability to think not only without restraint but without internal or external prejudice. Freedom demands mental curiosity. It is clear that most of modern society has come to be based on this freedom and curiosity. The second characteristic lends the power of reason to this freedom and curiosity. It gives it a structure upon which to grow and develop. It allows us to break a concept down to its salient parts. Even in the humanist field, the trained, disciplined mind has always been the mark of education.

Evaluation is the weighing, and arranging of the thoughts

unearthed in the mind. Without this ordering no development can come. This is the fundamental notion that values of things lie primarily not with the things themselves, but in their relationship. Based on these relationships generalized statements can be made which illuminate, and unify the related phenomena. The true value of any analysis or ordering is the resultant overview which then frees us and enables us to take the fifth step.

The "full-blooded creative act" as a defined clearly stated step in study, is the main difference between the science and the arts. The emphasis, the importance that the artist attaches to this "creation". But creation must have a base, it cannot spring from nowhere, a base in the understanding of Man and his ideals and aspirations.

"Reason and analysis cannot reach the springs of human action for these lie too deep: imagination can both reach and translate them into new habits ready to transform individuals into new beings and so society into a new pattern."8

But, the importance here is that, no matter how "creative" a man is, he cannot innovate or change unless (or until) he understands.

7.1.3 The Common Ground. The four steps we have defined

in the technologies and the five under Humanism when placed side-by-side reveal some interesting points of "common ground". The gulf is largely in our subconscious, in actual fact the methods of problem-solving are very similar.

<table>
<thead>
<tr>
<th>Science and the Technologies</th>
<th>The Humanites</th>
</tr>
</thead>
<tbody>
<tr>
<td>i) The statement of the objective.</td>
<td>i) Mental objectivity and curiosity.</td>
</tr>
<tr>
<td>ii) The observation of the facts relevant to the objective.</td>
<td>ii) Orderly and analytic thinking.</td>
</tr>
<tr>
<td>iii) The grouping of the facts and the interrelating of the groups.</td>
<td>iii) Evaluation of experience.</td>
</tr>
<tr>
<td>iv) The forming of generalizations and over-all patterns.</td>
<td>iv) Generalizing from particular instances.</td>
</tr>
<tr>
<td>v) Creation of new, ideal, patterns.</td>
<td></td>
</tr>
</tbody>
</table>

The first step is simply to think clearly without prejudice, about the over-all end or aim of any work. It is upon the precision and clarity of this statement that the ultimate worth of the solution depends, under both disciplines. This step, which demands the clear, or over-all view might thus, (in both cases) be defined as the THESIS or PROPOSITION.

The second step is the patient collection of facts, or at least the study of the facts. It is clear that the technologies might lay more emphasis on this step than the humanities, it is clear that this emphasis is based upon
accuracy; (now reinforced by advanced computation and measurement technique) the humanist at this stage is seeking the essence of the facts - what is behind them. His emphasis is orderly thought process rather than the exact accuracy of the individual facts. But in essence both might be called ANALYSIS or CLASSIFICATION.

The third step represents the beginnings of a new order, the sorting and grouping, finding the "strain of validity". This is a qualitative step and the technologist will assess the relevance of the experiment with respect to the initial statement before proceeding. The humanist also assesses or reassesses, the essential difference being his reference to human values, the quality or human value of an experience. This is surely in both cases DIAGNOSIS or EVALUATION.

The fourth step is very closely parallel on both sides, it is making of generalizations. This is the attempt to see the "woods rather than the trees" both stress this attempt to integrate, either by cold logic or by intuitive guess; both of which are then measured back against the preceding steps.

Here we see that the two sides part. Much of what the Humanist ascribes to the fifth step is implicit in the
fourth step of the Technologists but there is a definite statement of the "full-blooded creative act" on the Humanist side. Human beings cannot be divided into two species; one theoretical and imaginative and one rational and practical. Human minds have more similarities than dissimilarities; and laws of mental and imaginative behaviour are basically the same whatever the task.

"The truth is that the two mental procedures are not so divorced as sometimes assumed, for much of the fifth step is implicit in the fourth step of the Technologies, however explicitly it may be excluded by many scientists or technologists."  

It is time for the architects (and indeed the technologist, the sociologist, the engineer, doctor or whatever) to recognize their common ground and try to understand themselves and their behaviour and then those of others.

"There are, as I said before, common attitudes, common standards and assumptions."  

What are the scientific methods that engineers say they use, and what are the qualities and what are the strengths and weaknesses of the humanist tradition - the trained, creative mind? Rather than scoff at technologists with their systems, architects should endeavour to a) learn from these specialists and b) help to point out the limits

of such systems for their own uses. Only then will the profession of architecture learn from the Technologies only then can communication begin to grow and thrive.

"The technologist and the humanist differ rather according to their objectives than their procedures; and if the one is following the external world and the other the inner man, to be productive of results their method of approach is still forced into well-known paths of mental behaviour and they leave those paths at their peril........ it can be seen there is a common denominator of the mind, and that common denominator is mental procedure."^11

What is it that the architect can learn about decision making, (and how can he relate this to his own process) from the "Scientific or Systems Approach"? Where more exactly is the root of this common denominator? What, essentially, is the "Systems Approach"?

8.0 The Systems Approach

8.1 Definition. The Systems Approach can be variously defined, but it is almost always described as in terms of the steps or operations involved in carrying out a system analysis or design. A succinct definition of Systems Approach might be

"...the application of pre-determined rational procedures to coordinate management, programming, planning and design into a disciplined method of Problem Identification and Problem Solving."^12

Systems Approach, or even more broadly, Scientific Method, is always concerned with a series of steps which are always taken, in order, in the understanding and resolution of problems, be these management or design problems. A broad description of the methodology of diverse authors reveals some interesting parallels, be they architects, engineers or mathematicians. There is a consistent concern for two major issues. Firstly the defined series of steps to be followed and secondly the interconnection between steps.

8.1.1 Polya,* discussing Mathematical problems and their solution formulated "the list". It reads as follows:

a) Understanding the problem
b) Devising a plan
c) Carrying out the plan
d) Looking back.

The first step he defines as "...you have to understand the problem". Identification of the unknown, the data the condition. Whether or not it is possible to satisfy the condition. The introduction of a representative figure with suitable notation. The separation of the various parts of the condition.

The second part he describes as "Find the connection

* Polya, G. (1948)
between the data and the unknown. You may be obliged to consider auxiliary problems if an immediate connection cannot be found. You should obtain eventually a plan of the solution. This section in detail involves the solutions of sub-problems, the restating of the problem if the end is not evidencing, and an assessment of the data used. "Try and think of a related problem that you know". This is the synthesis of data into a form.

Polya's explanation of the third and fourth step are self-explanatory. The third step:- "Carry out your plan - check each step. Can you see clearly that the step is correct? Can you prove that it is correct?"

The fourth step:- "Examine the solution obtained. Can you check the result? Can you check the argument? Can you use the result, or method, for some other problem?"

Polya also recognizes the universality of an approach to problem solving in its application to other fields, when he says:-

"Although the present book pays special attention to the requirements of students and teachers of mathematics, it should interest anybody concerned with the ways and means of invention and discovery."

8.1.2 Carl E. Gregory*, a California professor of Business

* Gregory, C. E. (1967)
Management, defines "scientific Problem Solving" as a procedure which has nine steps divided into three broad categories.

("Stages of thinking")

a) Preparation
b) Incubation, insight or illumination
c) Verification

The nine steps are:-

a)  i) Deciding an Objective
    ii) Analyzing Problems
    iii) Gathering Data
b)  iv) Organizing Data
    v) Inducting
c)  vi) Planning
    vii) Prechecking
    viii) Activating Plans
    ix) Evaluating

Each step is defined as having a creative or judgemental substep, "...and each step of scientific problem solving must be understood and applied beyond a reasonable doubt of effectiveness before passing on to the next".

Central to his argument being that creativity is an inescapable feature of Scientific Problem Solving.

8.1.3 Marvin Adelman* is Senior Consultant to the System Development Corporation in California. He approaches systems from a different point of view. His definition is

* Adelman, M. (1967)
of criteria for Systems Approach rather than steps of procedure. "The systems approach is organized, creative, empirical, theoretical and pragmatic".

a) Organized, in that it may require a team to attack large problems. A team with a "language basis, and common goal" for communication.

b) Creative. "The system approach is a highly creative process, and its outcome is sensitively dependent on the people who do it and the resources on which they can readily draw. The solutions arrived at are not likely to be uniquely implicit in the data or the stated objectives; but they must be generated or devised so that they are consistent with the data and objectives."

c) Empirical, in that great reliance is placed upon data. Interest not only in the facts themselves but more strongly in their relationships to, and interreactions with, one another.

d) Theoretical. The system must rely, not only on any direct theory about the problem, but also on theory from related or sub-fields of activity. "To be most effective then, the system approach should involve both a central team, committed to the subject area and task on a more or
less continuing basis, and means of having access to people and organizations with additional relevant capabilities."

e) Pragmatic. "...a crucial characteristic of the system approach is that it is action-orientated. Its products are intended to be useful in the real world of practical affairs".

Adelman stresses the role of the team in the process, and underlines the communication and cooperation necessary. "Thus, an adequate understanding of the character of each of the agencies must be generated by the system team."

To foster this communication, the essential first-step of a common goal, (Problem-Statement) is central to any System Approach.

8.1.4 John P. Eberhard*, (Dean of the School of Design at University of New York, Buffalo) suggests that most of the formal methodologies include (with variations in complexity and terms), the following general steps:-

i) A period of problem definition.
ii) The invention of alternative solution.
iii) The selection of means to test the validity or effectiveness of the alternatives

iv) A testing of the alternatives by the means selected, and a return to problem restatement or invention of additional solutions if none of the first set of alternatives qualifies.

v) The implementation of the alternative finally selected.

vi) An evaluation of the results to improve the next attempt at solving problems of this kind.

He underlines that the overall "control" or limit is the "problem definition" or conceptual statement - that it is this which gives direction and structure to the hierarchical organization of the system. He also points out that a system, inherently, can cope with different scales and orders of problems.

"As long as we are reasonably rigorous in insuring that what we call a system has an inherent unity which we can define, we may encompass as small a set of components as we wish or as large an aggregation as we can understand."

8.2 Conclusions

8.2.1 From the two foregoing chapters (7.0 and 8.0) two major arguments emerge. These are:-

a) That the architect is not as far removed from the technologist, in his approach to problems, as is supposed; and can thus learn from, and communicate with, the technologies.

b) That the essential first step in any 'Systems
Approach' to problem solving is a clear initial statement of the problem.

8.2.2 Architect-technologist. Thus, the contention is that, in fact, the Gulf is not as wide or unbridgeable as is often supposed. In terms of basic approach the two disciplines are very near. What is needed is a realization by architects (and equally by other disciplines) of these similarities, and the use of this as a basis for communication in decision making.

"Because the systems approach sounds much like the methods now used by architects, it is necessary to caution practitioners not to dismiss it a 'nothing new' or embrace it without really understanding what they really need to do is different."{13}

Understanding of problems in terms of basic concept and then the structuring and definition of a system or framework are prerequisite to a new and more meaningful role for architecture. Obviously, in detail rather than broad concept, this 'bridging of the Gulf' is not simple; but the means, the material for building the bridge is there, and should not be wasted.

"The shapes of mathematics are abstract, of course, and the shapes of architecture are concrete and human. But that difference is inessential. The crucial quality of shape, no matter of what kind, lies in its organization, and when we think of it this way we call

it form. Man's feeling for mathematical form was able to develop only from his feeling for the process of proof. I believe that our feeling for architectural form can never reach a comparable order of development, until we too have first learned a comparable feeling for the process of design."

In order for the architect to make that crucial fifth step (of the humanities) "the full-blooded creative act" he must hone his limits, and the framework within which he works in order to give emphasis to the art of design. He must be able to get to that step with maximum accuracy and minimum time-loss.

Caudill places architecture "on the beach", and perhaps sums-up this argument best, for he alludes to the role of science without deflating the creative aspects* of architecture.

"The architect's position as the liason between arts-humanities and science-engineering will become more important. We will continue to practice on the beach. Like the beach where water overlaps land, architecture flourishes where these two worlds join; where science-engineering overlaps the arts-humanities. Warning! Whether or not we practice as a profession will depend upon how well we not only preserve but clarify this uniqueness. Regardless of our speciality on the spectrum of architectural practice we must not be completely submerged in either. In a war between science and art we remain neutral. We must have increased sensitivity to protect and enrich human endeavor through both science and art. We do this by staying on the beach. We are amphibians. If we go too far out we'll end up fish. If we go too far inland,

---


* Creativity will be examined more closely in the conclusion to Section D.
8.2.3 Problem Definition. The thread that runs through all of the systems cited is the initial, clear statement of the problem, as the first task. All subsequent activity is measured back to this statement. So too, must it be for architecture.

"The first task of the designer is to find out what the problem really is. The destruction of preconceptions of both the problem and its solution and the acceptance of the notion that all previous solutions to similar problems probably conflict with the needs of a new system in terms of the vast array of variables through which any system can be characterized." 16

The difficulty of this in architectural terms has been noted. This non-measurable, qualitative goal (or concept) statement must be defined, before design work of any relevance can result.

There is considerably more involved here than the client's first statement, or the architect's first conception of the problem. This statement has to be derived, checked and measured before it becomes the guideline for the designer.

"Another caution is that we should not assume we are prepared to make adequate problem statements. We seldom explore, in our present practice, the underlying requirements or the intended user behaviour.

for the buildings we design - we accept the client's statement of them, a statement usually based on the client's preconceived notions of solutions. We lack formal or rigorous means for evaluating alternative solutions."

Programming in this thesis is the process and methods used in the formulation of this initial statement.

Programming is problem seeking, design is problem solving - two separate and yet related activities.

Definition of problem statement processes and their relationships to design procedures; architectural programming and its relationship to design process.

Section D: PROGRAMMING: DEFINITION

9.0 Generally

9.1 Programming in this context is defined as the process by which the problem is defined. The process by which criteria and hierarchies are developed for use in the design of a space, building, facility, physical environment.

Since Programming is problem definition the process can also be used for problems where the end product is the problem (such as feasibility studies, master planning studies) as opposed to normal use for pre-design studies.

Programming is problem seeking and design is problem solving; separate processes, whose interface is the problem statement.

Such a program of criteria will include all of the characteristics, or functional requirements expected of the facility and should include aesthetic or other non-quantifiable goals statements that should influence the decision making, (or synthesis) which follows.

The program, for a task, is the 'set' of criteria and
hierarchies which will be the springing-off point in the
design process. The package however will contain the
necessary basic information which led to this 'set'. Thus
a quantitative list of space requirements is not a Program,
although it might form a part of the 'package'. As defined,
the package may or may not contain detailed, or secondary,
information which has been gathered. (This might well be
a separate package.)

The Program then, must be a simple statement; a list
perhaps, of conceptual generalizations about the problem,
which the designer can use as overall limits for his
work.

The Program must be simple, but simple as opposed to
simplistic. That is to say, the concepts or generalizations must be derived from, and related to, a body of
factual data which has been systematically collected,
collated and recorded.

What are the areas of concern, who is involved with the
process? How is it done - what constitutes the system-
atic analysis which results in the program.

10.0 Areas of Concern.
10.1 Problem Seeking - Problem Solving. The first, and one of the most important concerns is the differentiation between problem seeking and problem solving. Many systematic methods are involved with definition of the problem, or goals, as the first step. The essential difference being that, with architecture, problem definition is more complex and needs as much of an analytic procedure as the problem-solving area. Programming is concerned primarily with analysis; design with synthesis.

"What we need is a conscious clarification of our problems, that is, the definition of our building tasks and the means to their solution."¹

The end product of the programming process is the starting point in the design process, and thus it is clear that any programming procedure must be cognizant of this fact.

Programming must be strongly related to design, by definition, so that labour and resources can best be divided and utilized.

Only when architects separate the functions of problem seeking and problem solving will they be prepared to comprehend what is involved with Programming.

"Finding the right design program for a given problem is the first phase of the design process.

It is, if we like, the analytical phase of the process. This first phase of the process must of course be followed by the synthetic phase, in which a form is derived from the program.\textsuperscript{2}

Consequently the roots of any process of design hinge very heavily on the initial breakdown of the overall complexion of the problem situation. Programming design, evaluation.

Evaluation or feedback is only possible when these phases are clearly defined since without this definition it is almost impossible to find procedures to measure results (the user and the building) against.

"Firstly we have the questions concerning the relationship between buildings and those who use them, that is, the prerequisites and effects of architecture. Subsequently follows the question about the organization of the means, seen independently of their effects. Finally we question whether particular means correspond to particular prerequisites and effects. Taken together the questions cover all aspects of architecture as a human product. The theory thus becomes complete, if we succeed in answering these questions."\textsuperscript{3}

10.2 Differentiation between Wants and Needs. As a direct extension of the foregoing, one of the programmer's most important functions is to unravel and order client-goals. He has to be able to get "outside" of information

\textsuperscript{2} Alexander, C. (1964) p. 84.

given to him by a client or user, evaluate this and translate it into a form that he can use.

Very often a client is able to describe his requirements only by referring to an example that he knows or has seen. Typical here is the client with a glossy magazine photo of a building. "My offices should look like this". The point is that he has no vocabulary for describing his concept. Is it "grandeur", or "monumentality" or "prestige"? The programmer must question this information - examine it, and try to find what is behind it.

A more graphic illustration is the example of the school teacher, who when asked what her classroom (in a new facility) should have, that her present one did not, thought and replied that she wanted "pinning board to run the full width of the room above the blackboard". She recognized a need for pinning surface, and the only way she could articulate it was by describing a situation that she had seen. The architect, with more of a vocabulary of possible means of solving her "pinning space problem", described various ways this could be done and together they eventually decided that what she needed was a clear area of pinning space as opposed to isolated 'bits' around the room.
So the programmer (architect) had managed to get behind the solution she described and find out the need. In terms of design it is clear that the new definition of her need is more valuable to a designer than the manner in which she initially described it.

The programmer must always be very careful of separating wants from needs. Clients may express wants that bear no relation, say, to the size of their budgets. The programmer must be sure that the client understands the limitations of his budget and what this means in terms of his space (or other) needs. The final decision is the client's, the programmer must be sure this is as closely related to the real needs to a) produce a valid project, and b) avoid costly changes and omissions as the project develops (costly in terms of his, and the client's time and resources).

10.3 Communications. Implicit in the foregoing is the need for communication and cooperation between the parties concerned. The basis for this must be clearly established "modus operandi". The key is client involvement. Unless he can clearly see where the investigation is leading and what the means are he will never become fully involved. The programmer's role here is to direct and order the aspirations of the client. Some clients have a plethora
of information about their projects, others only minimal limits for the designer.

The client must be made aware what information is required of him and at what stage. In order that the client's aspirations and involvement be heightened, he must thus become an integral part of the programming team. He must not be placed in the role of "checker". That is to say he should not be asked to comment on achieved goals statements, (much as a professor examining student work) but rather he must be a direct participant in the formulation of the goals.

Goals and criteria established in this way become a sound base for progress. Involvement must ensure that the client comprehends fully the procedure and his goals that are set up; minimizing the possibility of his "changing his mind" at a later stage when this would be costly. Misunderstanding must be avoided - cooperation and involvement achieved.

(It should be noted that the word "client" as used above is pluralistic in that it not only often represents a body of people; but is intended to incorporate, wherever possible, the user. A more correct term, for the above argument, might be "client-user".)
This argument implies some sort of "programming team" rather than a "programmer". What constitutes a programming team?

10.4 The Programming Team. The team concept is central to comprehensive Programming. Obviously it will consist of two or more programmers, and the client might well be more than one person. Also involved are various specialists (economists, sociologists, business consultants, etc.). For any such team to progress there must be a definition of tasks. Broadly speaking we have two sets. The programmers and their consultants, and the user-clients and their consultants. It should be made clear that final decisions are taken by one member of each group*.

This team of decision makers will obviously have to be supported by, and be able to refer to, their consultants and "data-gatherers" at all times; and it is thus clearly possible that this team will change in size and composition during the Programming process.

The success or failure of the team is in direct relationship to the control of the two coordinators and the degree of cooperation that they are able to achieve.

* Two was used for illustration - it might well constitute three or more.
It is the programmer's responsibility to develop a language of communication with the client, and what this involves depends on the techniques employed. Audio-visual aids, charts, diagrams or whatever; but the programmer must be constantly conscious of the degree of communication which is being achieved.

10.5 Fact Collection and Correlation. Fact collection is, potentially, one of the best bases for accurate Programming. It can however very easily hamstring the process in that undue attention is paid to the "wrong" information. Programming involves various stages; it has been defined here as the initial problem statement, and care must be taken to ensure that facts collected bear directly on this. Goals must be tested early since this is the means by which the programmer tests the relevance of the data he is ordering.

Often the client, initially, provides much information of different kinds. Details with regard to furnishings and fixtures, finishes, lighting requirements etc. are important, but not at the Programming stage. This is important information for design development and that is where it should be ordered and used.

The programmer must extract the facts relevant; concepts
and facts, about the site (ingress, egress, topography, climate, etc.) the facility, (centralized, decentralized, use criteria etc.) and any other conceptual considerations. Other information should not be discarded but rather stored for later correlation and use.

Individual Programming methods will vary, obviously, and what is needed here is the establishment of a Programming process. The framework, which does not vary, in principle, under which all problems are studied.

For example, a team might have the broad categories of Site, Function, Demography and Environment, as their framework.

The importance is that, in every problem the team collect facts on the same base.

The "modus operandi" becomes the same, the teams become attuned to it and can decide very quickly whether a) they have satisfied their framework b) a fact or collection of facts has relevance to the problem statement.

The depth to which the search for facts goes then becomes a direct function of the problem in hand. Once a system is established it is soon clear where the "holes" will be.
The programmer will know what kind of information he needs. This framework makes the recording of past projects much simpler. If the firm has had experience in the field under study, the programmer can go back to past projects and look at, say, the Functional (or whatever) section and gain the information. If there is no experience in the project-type, his limits of research are still defined and this simplifies his research.

The setting up of this framework relative to the overall process of problem seeking and problem solving is the crux of Programming, and at this stage it seems imperative that the role of the designer in the overall process be defined.

10.6 The Role of the Designer. This is one of the most controversial aspects of Programming - whether or not the designer is involved, and if so, to what extent. This seems, to the author to be largely a question of the scale of operations - the size of the practice.

Programming is, primarily, analysis; design, synthesis. It seems conceivable, even reasonable, that the designer himself do the Programming for the project, if the firm (and thus its projects) is small. To have one or two programmers, as such, would mean that they might be idle
for periods of time. One of the primary concerns of Programming is the better utilization of capital and human resources and this would be negated by idle programmers.

One of the other prime aims of Programming is to allow the designer to tackle a problem (relatively) free of preconceptions; to give him rather, valid, clear criteria enabling him to approach the problem "fresh".

Whether the designer does, or does not, become involved in Programming is subordinate to the concept that Programming and Design are separate (and yet related) functions; and so, if the designer does do the Programming he should do it as a separate activity. First do the Programming and then do the designing based upon that Programming.

There are many proponents of "Programming through Design" but the arguments that precede have defined the two activities as separate, the contention being that, until the designer has a clear statement of the problem he should not proceed. By programming through design he is obviously designing in order to program - quite the opposite pole - implying some pre-conception, since where does the designer begin if not from an understanding of the problem?
Ideally, then, Programming and Design are separated and done by different people. Why is this ideal? Programming has been defined as being primarily analytical and design as synthesis. Creativity is central to design and subordinate in Programming*. Purely in terms of utilization of talent and potential within a firm, designers should be engaged in design. It has been postulated that design must begin with the Program statement.

Thus, where the scale of operations warrants it, the author believes separation of task is imperative, simply because of the difference between analysis and synthesis, as stated above. Where operations are on a smaller scale, Programming might be done by the designer, but definitively. That is, he must be cognizant that he is analyzing the problem with a statement of it as the end-product; which end-product is then the springboard to his design process.

By definition then, there are two (at least) separate divisions within the firm, yet related. How is this relationship and intermix achieved? How is the programmer sure that his statement is in the form that the designer wants? How does the programmer formulate the criteria for the designer's use?

* Discussed more fully under 10.8.
10.7 The Process. In section 10.5 the author alluded to the "Framework" within which the programmer works. This framework is the connection between Programming and Design.

Any firm actively engaged in Programming must have a "framework" or "set" which envelops the whole organization. That is to say, the criteria sought out by the programmers must be the same concerns by which the designers choose to create, and evaluate, their forms.

This is obviously a policy decision and may be quite different from firm to firm. For example a firm might choose to work under "Cost-Efficiency and Beauty" or "Form-Function -Economy"*, or "Time-Cost-Maintenance-Structure" or whatever.

Once these parameters have been set up, they become parameters for
a) Programming
b) Design
c) Evaluation

The same controls are used for each phase of the system. The individual processes unrelated to each other have no validity. Without the "Framework" how does an architect

* See Section E. Case Study.
set about evaluating his project? Or in a large firm how is it possible for quality control of any kind to be instituted and implemented without a frame of reference?

The connection then, is "the framework", "criteria for performance", "controls", "philosophy", and the Program statement should be in these terms, that is the goals be stated with reference to each of these stated areas of concern.

10.8 Creativity. There are two main charges with respect to creativity, programming and design, each of which could be the basis for a study in itself. It is of value however, to cite these and state a position on these issues.

The first is that Programming or a systematic approach stifles creativity in the design phase. It is a well-established fact that there is a definable limit to the number of issues a designer can consider simultaneously, and this is one of the aims of Programming - to reduce complex information into manageable proportions.

"There are limits to the difficulty of a laboratory problem which he can solve, to the number of issues he can consider simultaneously; to the complexity of a decision he can handle wisely. There are no absolute limits in any of these cases (or usually even any scale on which such limits could be specified); yet in practice it is clear that there are limits of some sort. Similarly, the very frequent failure of individual designers to produce
order from a series of abstracts and related facts.

"Physical concepts are free creations of the human mind and are not, however it may seem, uniquely determined by the external world.... with the help of physical theories we try to find our way through the maze of observed facts, to order and understand the world of our sense perception...."\(^8\)

The role of creativity in design should never be diminished for it is, ultimately, the "brilliant flash" or "the lonely jump" and its clarity and accuracy that not only solve a problem but present a new level of insight into it.

"If he is a good designer the form he invents will penetrate the problem so deeply that it not only solves it but illuminates it."\(^9\)

The second charge is that there is no creativity involved in Programming and that it might be tackled by anyone with analytical skills or even by a computer. The preceding definition of Programming centres upon the assumption that the result is valid generalizations - the problem stated and defined - based upon the relevant facts. Obviously then, the ability to "see the woods after examining the trees" is a task not easily accomplished and the reduction of facts into generalized conceptual statements does require a degree of creativity.

The programmer must be creative in that he must be able to see beyond the problem specifics. His task, as has been stated, is to "extract the essence" within differing

---

well organized forms suggests strongly that there are limits to the individual designer's capacity.\(^4\)

Alexander also draws an analogy to illustrate this point.

"Two minutes with a pencil on the back of an envelope lets us solve problems which we could not do in our heads if we tried for a hundred years. But at present we have no corresponding way of simplifying design problems for ourselves."\(^5\)

There is currently little resistance to the view that limits placed on a problem do not stifle creativity but rather enhance the possibility for it. Stravinsky stated

"...the more art is controlled, limited, worked over, the more it is free....My freedom consists in my moving about within the narrow frame that I have assigned myself for each one of my undertakings."\(^6\)

Creativity has often been described as "the long lonely jump" between the facts and the synthesis of them. The author argues for Programming, not to do away with creativity but rather to heighten its potential.

"Such belief in the program as direct determinant of form is no more than another extremism in a long gallery of designers in search of a ritual. "A design program does not determine a design's aesthetic form but it restricts the possible number of forms by stipulating the level of operational performance that the design's form is to satisfy."\(^7\)

The designer must be "freed" in order that he concentrate his unique ability upon synthesis, the development of an

problem situations.

The emphasis, however, remains analytical rather than creative. This question is perhaps best summed up by Alexander:-

"What designers disagree about is the relative importance of different requirements. In the present theory this would have to be expressed, if it were expressed at all, by assigning some sorts of weights or values to different variables. However, few designers will actually disagree about the variables themselves. While the relative importance of different requirements usually is a matter of personal opinion, the decision that a requirement either is a requirement or isn't, is less personal." 10

10.9 The Role of the Computer. The degree to which computers are related to Programming is, fundamentally, simple. Whatever the computer can do is a direct result of the input it receives; it cannot create, it processes data. Consequently before architecture can fully utilize the "ability" of the computer to analyze data and generate useful output, it must first have a structured process whereby parts (or problems) can be extracted for analysis.

The computer, then, is essentially an aid to Programming. A sequence of activities, once set-up, can then be examined to see where the computer can be applied. Architects must understand the problems very clearly and

understand what they want from an aggregate of data; and then understand the capabilities of the computer, before valid use can be made of such a tool.

It is not difficult to imagine how useful the computer could be in the storage and retrieval of information and the potential for its uses in the Programming field is enormous. Manipulation of variables, storing of data for later use etc. etc.

But first architects must learn to order the processes they use; to define their problems more clearly, only then will the computer realize its potential.

"In my opinion the question, "How can the computer be applied to architectural design?" is misguided, dangerous and foolish....

'A digital computer is, essentially, the same as a huge army of clerks, equipped with rule books, pencil and paper, all stupid and entirely without initiative, but able to follow exactly millions of precisely defined operations. There is nothing a computer can do which such an army of clerks could not do, if given time...

'In asking how the computer might be applied to architectural design, we must, therefore, ask ourselves what problems we know of in design that could be solved by such an army of clerks...

'At the moment, there are very few such problems. Although we speak a great deal about the complexity of problems, the complexity of architecture, and the complexity of the environment, this talk, so far, is rarely more than hand waving. In the present state of architectural and environmental design, almost no problem has yet been made to exhibit complexity in such a well defined way that it actually requires the use of a computer....

'It is ironic that the very tool which has been invented to unravel complexities imposes such severe restrictions on the design problems which it can
solve that the real source of complexities has to be eliminated before the tool can even get to it. But for the moment that is the situation.\textsuperscript{11}

And Myer and Krauss state, in a study done to evaluate the role of computers in design

"Architects are not conscious of their method; they're too involved in substance to focus on method. They can't articulate verbally what they're after, and we're surprised ourselves to feel so strongly about some of our conclusions - for instance, that the final form is the direct result of the order in which the designer considers the variables. We wouldn't have thought this, before we began."\textsuperscript{12}

11.0 Conclusion.

11.1 The conclusion to this chapter should be read not only as a conclusion to the above but also as an introduction to the Case Study which follows in Section E. This conclusion might best be defined as providing criteria for the Case Study, and breaking down these criteria into four broad categories.

11.1.1 Flexibility. Any Programming system must, by definition, be able to handle problems of varying scales of complexity and size. At the same time Programming should be able to be used for various other functions such as


Master Planning Studies, Feasibility Studies and the like; since Programming is **problem seeking**. Flexibility, to accommodate unique applications and varying emphasis, is a prerequisite to a good Programming process.

Flexibility should be taken further to include "internal flexibility". Is the system easily explained to designers, new programmers and to clients, and is there the possibility for growth and change within the process?

11.1.2 Feedback. The new order of problems demands that the architects evaluate completed projects. Does the Programming process, the design process and hence the overall system have an area or segment which can receive, evaluate and re-use information in the interests of better product? Programming as a basis for this research or feedback is, perhaps, one of its greatest potential functions. Without structuring every phase of the design process, architects will never be able to identify and correct problem areas.

Architecture can never respond to people without measuring, in some way, the way people use and abuse built facilities*

11.1.3 Experience. Past projects have import in that

* Feedback is discussed more fully under Section G, Appendix II.*
they are sources of information as well as the potential identification of design deficiency. Programming procedures must be adjusted and grow, as stated above; but unless the information is able to be recorded and retrieved without undue complexity the system will have only temporal benefit. Programming procedures must be seen through the ability they have to use experience.

11.1.4 Communication. The system must foster communication, not only between programmer-client-designer-user-specialist etc., but also in that the system should be a continuing one throughout the design process. The author indicated the need for different information at different stages. Thus Programming fits not only in schematic design but later, in a different form, in design development etc. What is important here is communication between the designer and programmer. There must be the potential for dialogue. The designer must be able to challenge the client's goals-statements and the programmer must only advise revision of the client's goals in terms of the designer's input.

The less changing of these goals or criteria that can be done, the more accurately has the Programming been done. What is important is that it is an open avenue rather than a dead-end street, between designer and programmer;
both architects serving the client.

11.2 The Case Study of Section E will be examined through the matrix

a) Flexibility
b) Feedback
c) Experience
d) Communication

and all its implied sub-sections.
Section E: CASE STUDY

PART ONE: SPECIFIC

12.0 Large Architectural Firm.

12.1 It has been pointed out that a large firm can do many things, with regard to programming, that smaller firms cannot do, simply because of its size and the variety of problems with which it deals. But any Programming process, as outlined previously must be able to handle problems of varying size, and the lessons to be learnt from this firm are not limited to large projects. There is much here that is relevant, in principle, to smaller firms.

12.2 The firm of Caudill, Rowlett, Scott* (hereinafter referred to as CRS) was selected for the specific study for the following reasons:-

(a) Programming is a defined facet of their Project Delivery Process.

(b) Their national reputation.

(c) The availability of information.

12.2.1 CRS defines problem seeking, or Programming, as part of their overall 'problem solving approach' to architecture. They have a department of Programming that works

* Caudill, Rowlett, Scott. Architects, Engineers, Planners. 3636 Richmond Avenue, Houston, Texas 77027. (Approximately 270 employees.)
under a defined process for all projects, be they building or non-building studies. The emphasis on parts of the process will vary for projects differing in size and nature, but the approach to Problem Definition is always the same.

12.2.2 CRS enjoys a national reputation, (not only as an architectural firm, but more importantly here) as a leader in the field of Programming. Several members of the firm have published material on the subject and the partner in charge of Programming, Mr. William Pena, has addressed several conferences (architectural and other) on Programming.

12.2.3 The CRS system has developed over several years of practical experimentation and the firm has abundant material (on their process and in general) available. The firm itself being in Houston, was physically, immediately available to the author. CRS was throughout this study willing to participate a) in the research b) in the exposure of their method.*

12.3 The definition of the CRS process (below) is from published articles by the firm, as well as from intra-office memoranda.

* The author wishes to record his gratitude to Mr. W. M. Pena, General Partner, Head of Department of Programming, and Mr. J. W. Focke, Programming Architect; of Caudill Rowlett Scott, for their invaluable contributions to this study.
13.0 CRS Programming.

13.1 The CRS approach to programming is COOPERATIVE, ANALYTICAL and CREATIVE.

Cooperative Approach:

There are many people involved in arriving at the architectural solution for a client's problem. At the programming stage, there are the owner, the users, the architect, special consultants, governmental agencies, etc. Successful programming can be accomplished only if there is total cooperation among all participants.

Programming requires the joint effort of two groups -- the client group and the architect group. Together they form the programming team. It is important that each group designates a responsible person with complete authority to make decisions or to cause decisions to be made. This simplifies the communication between the two groups. The collaboration of the two group leaders can then result in a successful project.

The client group is primarily responsible for information concerning functional and organizational requirements, financial objectives and the physical conditions of the site.

The client group includes the group leader (coordinator), the building committee, and special consultants (functional, fiscal, equipment, etc.). The work of client group might precede the work of the architect group or the two groups might work concurrently.

The architect group is responsible for the analyses of the functional program and the space requirements. The architect group includes the group leader (project manager), specialists (programming, design, building type, cost estimators, etc.) and special consultants (technologists).

Together the two groups as a total team analyze the facts and identify functional-architectural concepts for the facilities. Together they must balance the space budget and the cost budget.

The client's presence on the team stimulates a group interaction that would not otherwise be possible.
The team concept requires a high degree of communication. Organized programming provides the format for dialogue. It is necessary to graphically document every program item for quick reference. This documentation is the responsibility of the architect -- not only to demonstrate that he understands the information but also to provide feedback to the client. A glossary of terms may be required. It is amazing how confusing the program may seem with the use of esoteric terms.

The squatters technique is an on-the-site, intense work session involving the total team. The technique requires preplanning, cooperation and effective communication. The programming squatters brings the total team together to focus on identifying the problem in all its aspects. The short intense time period requires that the process be organized to bring the proper team members together for various levels of analysis and decision making. The same squatters technique is used in the problem solving session in which the client is again a part of the team.

Analytical Approach

In any problem solving approach there is an analytical stage. This stage is usually associated with problem seeking. This is reinforced by the fact that another common name for programming is "architectural analysis." Problem seeking requires an objective attitude in dealing with the realities of the problem. During this exposure stage all pertinent information must be brought to light regardless of the individual bias of the team members. Objectivity in decision making requires rational thinking and judgment. Preconceptions at this stage are to be avoided.

Creative Approach

The programming process is basically analytical, but it is also creative -- not in seeking solutions but rather in generating programmatic ideas which will influence design decisions. Programmatic ideas refer to functional concepts as opposed to physical concepts or solutions. The analytical attitude in programming must not preclude the intuitive insight into the problem. Programming allows for creative thinking within the rational framework of procedures. Problem seeking is creative in establishing the uniqueness and essence in the statement of the problem.

Two-Phase Process

The client-architect team should seek to organize the
information under a priority for use in design. This is no attempt to stop the flow of information but merely to determine its relevance for the phases in design. During the schematic design phase a designer intends to solve the big overall problems first. During the design development phase he intends to solve progressively more detailed problems. The programming process should provide the appropriate information at each design phase.

The first phase of programming must necessarily deal with a wealth of information, but this information must be analyzed to reveal the relevant facts above the unimportant details, the basic concepts above the small features.

The second phase then must provide the detailed information for the development and refinement of the design. By then the details will not obscure the significant statements that emerged in the first phase.

Background

If the programmer is not experienced in the area of the client's problem, he must develop a background understanding of the nature of the subject through a survey of similar projects, library research, etc. This can be limited research simply to understand the jargon of the client at first contact, and the general nature of the client's needs. He must seek information, not solutions.

With enough background information, the number and kinds of consultants which will be required and when they might be most useful can be determined.

Steps to Define the Problem

The search for the definition of the problem calls for a step-by-step analytical procedure. There are many step-by-step procedures for problem solving which mix the steps for problem seeking and for problem solving. This mixing of steps leads to confusion in the architectural process. The separation of problem seeking and problem solving is absolutely necessary if we are to clarify the confusion and go on to strengthen design methods.

The typical scientific method for problem solving begins with the definition of the problem and ends with the selection of the solution. The five steps in problem seeking are as follows:

1. Establish goals
2. Collect, organize and analyze facts
3. Uncover and test concepts
4. Determine the real needs
5. State the problem

The sequence of the steps may vary, but the steps themselves form an orderly framework for the classification and documentation of information coming from many sources.

In practice, the first three steps may be concurrent and cause a re-evaluation of goals and a recycling for confirmation of facts and concepts. The fourth step is taken after evaluating the first three to determine the space requirements, the performance criteria, and the project budget. An imbalance here would cause a recycle analysis to review and adjust the first four steps. The fifth step is taken after re-evaluating the previous steps to establish the uniqueness and the essence of the whole problem. Other scientific methods work from a hypothesis to a conclusion. This approach is not generally applicable to problem seeking, but rather to problems to prove.

Basic Considerations

If the design of the facility is to solve problems of Function, Form, Economy and Time, then the programming must treat these as basic considerations and classifications of information.

Function for this purpose, deals with the functional aspects in terms of aims, methods to be used to meet them and the numbers and characteristics of the people involved.

For may be used to evoke questions regarding the physical and emotional environment to be provided, the quality of construction and the condition and the conditions of the site.

Economy emphasizes the need for early cost control and brings up for consideration the initial budget, the operating cost and the long term cost which may be affected by initial quality of construction.

Time involves those considerations of change, growth, and escalation which affect Function, Form and Economy.
STEPS AND CONSIDERATIONS

Form, Function, Economy and Time are the basic considerations, the content of the programming process... and later they will comprise the criteria for evaluating the programming package as well as the design solution. But these considerations are not in themselves a process. One is not considered before the others. All four are considered simultaneously at each of the five steps in the analytical procedure. We use them as key words in seeking information and as general categories or classifications for organizing information at every step of the programming procedure.

Form, Function, Economy and Time within the framework of the process provide:

1. A format for collecting information
2. Classifications for organizing information
3. Criteria for evaluating the results of programming and design.

A MATRIX CHECKLIST

Another way of coordinating the steps and considerations is to establish a simple matrix which is a useful tool in generating the necessary information to complete the programming phase.

The matrix can be used not only as a check list for missing information but also as a device to display the emphasis or amount of information regarding form, function, economy and time at each step. This emphasis could also be indicative of the priority of the considerations. We consider form, function, economy and time simultaneously, but within the context of the project each might be given a particular priority. Obviously each project would have a different profile in the matrix.

(1) ESTABLISH GOALS

It is usually easier for the client to verbalize his goals for the project at the very beginning when he has the total project in mind and before his thinking gets involved with details. Practically it is better for goals to be established at the beginning since they will establish a direction for programming. The gathering of facts will be related to the goals and the tests for programatic concepts will determine whether the goals are indeed being implemented.
The client might place particular emphasis on one or more considerations through his statements of goals. However, he should be encouraged to state his goals in terms of all four considerations. For example, the client is prone to think only of functional goals; the programmer should be ready to ask questions regarding the goals for the other considerations.

The client-owner may establish general overall goals. The client-user usually establishes more specific goals. If there is a conflict between the owner and user goals, the conflict must be clarified and resolved. The subsequent steps depend on clear-cut, coordinated goals.

(2) COLLECT, ORGANIZE AND ANALYZE FACTS

Facts by themselves will tell us nothing. They have to be organized and analyzed before they will reveal their importance. The classification of facts under Form, Function, Economy, and Time is useful in organizing and analyzing the information.

The goals will determine the kinds of information which will be meaningful. Nevertheless, we have to discriminate between pertinent facts which will be immediately useful and details which will be useful at a later phase. This does not mean the stopping of the flow of information but rather classification for relevancy. We must not let the details distort or confuse what is really important in the immediate programming phase. Hence, the idea of two-phase programming.

Check-lists may be developed for the collection and documentation of information for each project type. The following are categories which might be generally applicable:

FORM: We need data on the site... its physical characteristics, climate conditions, legal aspects, coincident planning by other agencies. We need information regarding the availability of materials and the local construction industry. We need information on building codes as they might affect the form of the building.

ECONOMY: We need data on budget limitations, local cost index, building cost per square foot, operating costs and long-term costs where applicable. Method of financing, economic influence of other agencies.

TIME: We need data on project schedule, phasing and growth, price escalation; anticipated changes and projections.
UNCOVER AND TEST CONCEPTS

While step 2 is said to deal with facts and step 3 deals with concepts, it is difficult to separate facts from ideas. The grouping of facts stem from ideas, concepts of organization stem from facts. They depend on each other. For analysis, facts and quantitative, concepts and qualitative.

Programmatic concepts are methods of implementing the goals. Most of these concepts are organizational in nature which implement the functional goals. This heavy emphasis on function results from the clients participation on the team. It is here that the client can exercise his most creative thinking in determining his functional relationships.

The client must be stimulated to make decisions in regard to his organizational structure and functional relationships. If these decisions are not forthcoming, the testing or concepts will provide a means of stimulating the client's decision-making authority.

The programming architect must be creative in the sense of finding alternatives to cause the client to participate and react with the required decision. It is not the programmer but the client who makes the decision. The programmer must provide the analyses to uncover concepts and to stimulate the decisions.

It is difficult to understand the difference between programmatic concepts and physical concepts. Most often concepts are thought of only as design solutions. This misconception is reinforced by the fact that physical concepts in design respond to programmatic concepts and thereafter become so closely related that it is difficult to know which came first. To identify programmatic concepts one must think abstractly in terms of organizational structure, relationships and other functional requirements.

Concepts are uncovered and tested through the use of evocative words. These words indicate the information which might be most fruitful. These words may be found in the subcategories of the basic considerations of Form, Function, Economy and Time or they may be identified with recurring concepts. In discussions with the client the programmer must be alert to identify concepts and to record them. It takes a receptive mind to identify concepts when they are expressed however briefly.
CLASSIFICATION OF CONCEPTS

The classification of concepts under Form, Function, Economy, and time is simply a means of analyzing the implications. The specific classification is a matter of interpretation—a particular concept could easily be listed under two or more classifications. The following classification is based on our experience in testing these recurring concepts:

FORM:

Concepts stemming from Function, Economy and Time can be stated abstractly; it is more difficult to state form concepts abstractly but they can be—in terms of the site, the physical and psychological environment, and quality. The client should be encouraged to state form concepts abstractly.

When the client cannot express a form concept except in physical or concrete terms it turns out to be a premature design solution. In this case the programmer must look behind this solution to examine the validity of the reasons that lead to it.

Preconceptions, prejudices and opinions by the client can be useful information and should be deliberately exposed during the programming phase. These can be discussed and analysed and they can lead to a better understanding of the requirements. Avoiding prejudicial might lead to difficulties later in the project. Evocative words for form concepts are Site, Physical and Psychological Environment and Quality.

FUNCTION:

The programmer might test a number of recurring concepts with the client to give him a means of expressing his functional needs.

CENTRALIZATION VS DECENTRALIZATION

This concept deals with centralization or decentralization of activities, services or personnel. It can establish organizational structures, functional relationships or requirements, and over-all affinities. It is important not to misconstrue this programmatic concept as the physical concept of Compactness vs. Dispersion. The programmatic concept can have several alternatives of Compactness or Dispersion.
INTEGRATION VS. COMPARTMENTALIZATION

We must find out if activities should be integrated or compartmented. A family of activities or closely related functions would indicate integration; the need for some degree or kinds of privacy (acoustical or visual), compartmentalization. Again we should recognize the difference between the programmatic concept of integration and the responding physical concept of the Open Plan.

HIERARCHY

Hierarchy is an evocative word from which spring a series of concepts regarding the order of importance or priority such as relative position, size, social value and others.

FLOW

The concept concerns the flow of people, vehicles, goods, services and information in terms of priority, sequence and mix or separation. This concept expands on affinities and relationships (it is not concerned with a table of organization.) We can numerically code connections between corresponding units, construct abstract flow diagrams and manipulate the diagrams to minimize the circulation conflicts. This could be a computer or a manual function.

PEOPLE

This is another evocative word which can generate concepts derived from the physical, social and psychological characteristics of people—as individuals in small groups and in large groups. This is an area in which there is no substitute for experience and expertise.

ECONOMY:

Concepts regarding economy are particularly useful in the recycle analysis if there has been a mis-balance between budget, space requirements and quality.

MULTI-FUNCTION is the first and obvious concept which might be tested as a means of achievery and balance; however, it should be understood that it could result in reducing efficiency for each of the combined functions. There could be other functional alternatives.

Other evocative words which can generate economy concepts are: Phasing, optimization, Efficiency and Cost/Effectiveness.
TIME:

CONVERTIBILITY

This concept is a response to anticipated change in functional requirements. The degree of convertibility must be established as (1) immediate, (2) weekend or (3) long range convertibility.

EXPANSIBILITY

Anticipated growth triggers the programmatic concept or expansibility.

PHASING

This time-economy concept is useful in the recycle analysis to attain functional and/or economic feasibility.

There may be others but these recurring concepts have been generated through an accumulated experience with many building types.

DETERMINE NEEDS

The determination of needs is an analytical step to establish (quantitative) needs of the client in terms of quality (cost per square foot), space requirements, and a budget predicted at the time of construction.

The proposed space requirements and the expected level of quality must be tested with the proposed budget.

Balance must be achieved between space, quality, budget and time. In case of an imbalance at least one of the four elements must be negotiable. If agreement is reached on quality, budget and time, the adjustment must be made in the amount of space. The client's space requirements must be reanalyzed to determine the realistic space needs as opposed to the client's wants. Other alternatives would involve the re-evaluation of quality, budget and time. A more serious imbalance might require recycling to re-evaluate goals, facts and concepts.

FORM

The proposed quality of construction is express in the quantitative terms of cost per square foot. At this point this expression is based on experience and/or background survey and analysis. Agreement on the quality must be
reached with the client.

Both the physical and psychological environment will affect the quality of construction and in turn the cost per square foot.

Site conditions will affect the form of the building and will influence the construction budget.

FUNCTION

Space requirements are the results of functional needs as determined by facts and concepts. They are generated by people and activities. Parameters are predictive units of space based on a knowledge of typical activities. Allowance must be made for a reasonable building efficiency (the predicted relationship or net to gross areas).

ECONOMY

The cost estimate analysis must be as comprehensive and realistic as possible. Every possible cost category must be incurred in the analysis. There must be no doubt as to what comprises the total budget required. The building cost is based on cost per square foot and percentages or the building cost are used for most of the other line items unless more refined figures are available. Most often these cost parameters are based on experience tempered by the luck situation.

TIME

In determining the cost per square foot a realistic escalation factor must be included to cover the time lag between programming and mid-construction.

Phasing of construction may be considered as an alternative:

1. When the initial budget is limited
2. When the funds are available over a period of time
3. When the function needs are expected to grow.

STATE THE PROBLEM

The statement of the problem is the interface between problem definition (programming) and problem solution (design). The programmer and the designer work together in formulating very succinct statements after evaluating the first four steps. The problem must be stated in qualitative terms emphasizing the essence and the uniqueness.
The essence might be reached by limiting the number of statements to a minimum of four—dealing with form, function, economy and time. There should be no more than ten statements, more than ten might indicate either that the project is very complex or that details are being used as premises for design.

UNIQUENESS
OBVIOUS
GENERAL - DETAIL

The statements must deal with unique not the common aspects of the problem. Sometimes the obvious leads to the uniqueness—obvious only after the first four steps.

These statements should be made in terms of performance and not physical form. They should be made so as not to close the door to different expressions in architectural form.

A statement might be derived from any of the previous four steps—goals, facts, concepts, needs—as long as it is an important form giver.

14.0 Methods Used.

14.1 Three important features of the CRS Programming process, mentioned in the above, merit specific note here.

14.1.1 The Triad Theory.¹ The overall Project Delivery System of CRS is governed by the equilibration of FORM, FUNCTION and ECONOMY through TIME. A high magnitude of equilibrium of these three forces (a dynamic equilibrium as opposed to a simple, or low level, equilibrium) governs all facets of the system; Programming, Design, the overall Management envelope, etc.

The validity, or otherwise, of this philosophy is not an issue here, what is important is that this "Triad" is used as the framework defining the scope and structure of each of the processes within the system.

This runs throughout - Programming Schematic Design, Design Development, even to quality control and project evaluation. The Programming process thus has its roots firmly in the larger Project Delivery System.

14.1.2 The Problem Solving Approach. CRS believe that architecture, to a great degree, is a process of solving a series of problems. Thus, unless there is a clear statement of the problem, (that is, unless you know what the problems are) there will never be a good architectural solution.

14.1.3 The Team Concept. CRS believes that solving problems today can best be done by a team of people with a variety of talents. That the demands of society can best be met through the efforts of a co-ordinated and creative team.

14.2 Communications. These considerations underline CRS's concern with effective communications. Two of the more important techniques used by them are outlined below.
14.2.1 Analysis Cards*. 8-1/2 X 5-3/4" analysis cards are used for describing each of the many points on which the client should be informed and to which his reaction is wanted. Only one idea to each card - for emphasis. Thus the client can examine facts and discuss concepts in an organized simplified manner, and can understand fully and quickly which factors are of concern and what other factors his decision will influence.

14.2.2 "Brown-Sheet" technique. The firm presents relative space quantities - drawn to scale in block form - on brown wrapping paper sheets. During work sessions these are pinned to the wall and the client is able to see the relationships and relative sizes of parts of the facility. Changes are then made to these sheets as these requirements are affected by cost realization (or other reasons); new data is pasted over the old until a satisfactory space allocation is reached. By the time the study is completed these sheets are sometimes book-thick.

Brown paper is used to psychologically stress their impermanence. Computer printouts could show the same graphic information but CRS believes that a) the size would be wrong b) there is still an aura of "finality" or "authority" about a computer print-out which will impede easy and frequent alteration.

* See Appendix I
15.0 Conclusions

15.1 This conclusion is based upon the definitions of Section D, and subjective evaluation as seen through Appendix II.

15.2 The criteria established for Programming (11.2), were:

(a) Flexibility
(b) Feedback
(c) Experience
(d) Communication

15.2.1 Flexibility. The process shows remarkable flexibility in that it is used not only for projects of varying size, but also for the definition of non-building problems*. The process has been successful in that its applications have spread over time. Several non-building studies have been completed and their success is reflected by the fact that an increasing amount of this type of problem is being examined by the firm. Thus the flexibility of the process within an overall system of project delivery is patent; and the "internal flexibility", the fact that changes and refinements are constantly made from within, reflects progressive developments.

* See Appendix II
15.2.2 Feedback.

(a) Internal. The internal feedback has been mentioned above; but it should be noted that attempts at refinement and development are not incidental to this process, but are a defined function.

(b) Design Feedback. This is one area that the firm has not pursued to full potential but have recently begun a program of evaluation of their buildings using the program requirements as a base. It should be noted that evaluation of buildings does not necessarily reflect the success of the Programming process, but is an evaluation of the way in which the design process has interfaced with the Programming process in terms of the established project delivery system.

A clear process of evaluation of completed projects is the obvious "next-step" for CRS, and it is clear that their Programming process can provide both the base and tools for this research.

15.2.3 Experience. The process has been built and refined over several years, and this attests to the central role that experience has played. The firm has built considerable records with regard to building types, and they do, consciously, exploit the information gathered over time.
CRS has not preserved individual "programming packages" as well as they might have. These, as stated earlier, rely on analysis cards (for example) and these are often lost during the design process. Recent attempts have been made to offset this, and file copies are made (immediately post Programming) before these cards are turned over to the design team.

This is the one facet of the process that needs "tightening-up".

15.2.4 Communication

(a) Internal. Not all the designers are as familiar with the process as they might be, and obviously since different personalities are involved, some utilize the Programming process more fully than others. Efforts are being made at this time, to define, intraoffice, the Programming process in order that all members of the firm understand the goals of programming as well as the potential of the process.

(b) External. CRS is outstanding in this phase and the degree of involvement they have achieved with their clients is remarkable. The program is reached by client decision, the programmer's role being to make him aware of the problems and also to be sure that he is exposed to them. The
"squatters technique" (both for Programming and schematic design) has proved invaluable in terms of client involvement. With this involvement the client's aspirations are increased and thus the final product, potentially, is improved.

The firm is also incorporating (or attempting to incorporate) different special disciplines in order to make the problem statement more valid. A behavioural scientist is attempting to increase the potential of the Programming process by introducing new insights into the measurement of human activity.

Their process is both "open" and yet structured, with regard to communication, and this is one of the finest features.

15.3 The most important characteristic of the CRS process is its simplicity. Simple, but simple as opposed to simplistic. It is this very simplicity that makes it so difficult to evaluate "from without". The author, early on in the study felt "there is nothing to it", but soon realized the value of the process and its relationship to the overall system as defined in this simple way. None of the criteria previously cited are truly possible until simplicity is achieved.
Simplicity and the constant state of flux (in terms of improvement and expansion of the process) are its greatest assets. This is one of the few Programming processes, defined for use in terms of practising architecture, that have been successful. Successful in that it is used and also that its uses are increasing.

PART TWO: GENERALLY

16.0 Other Programming Systems.

16.1 Four Programming processes are described below; three theoretical (not related to actual in-office use), one practical.

16.1.1 Richard Seaton* defines Architectural Programming as the following:-

" (i) stated human goals, targets, or benefits;
(ii) a set or sequence of operations or activities in time conducing to goals or benefits;
(iii) a statement of material and environmental conditions necessary to or encouraging such operations;
(iv) a specification of the characteristics, including costs, of objects acheiving those conditions; and
(v) a definition of the criteria by which the human goals, targets or benefits are to be measured."²

* Associate Research Psychologist and lecturer in Behavioral Sciences in Architecture, College of Environmental Design, University of California, at Berkley.

He adds also that programming need not be restricted to a single building; building components and or sets of buildings may be programmed. And that

"In building programming statements of purpose or goals and their measurement are crucial."\(^3\)

He does not discuss, in this paper, how the programmer goes about achieving this program; confining his attention to the "role of the programmer", and the difficulties involved in Research in Programming.

The task of the programmer is to:-

"(a) examine stated target benefits for different population elements with an eye to uncovering implicit objectives and expressing them through objective criteria;
(b) specify operations (tasks and functions) contributing to criteria;
(c) ascertain attributes of the designed environment which promote operations and uncover attributes which interfere with operations;
(d) specify alternative design forms which conform to constraints and provide environmental attributes that promote operations or ends, and choose among them in terms of benefits derived for dollar outlays;
(e) determine promotive (spillover) inter-relationships between design forms;
(f) allocate available resources to facilities and forms providing the larger marginal benefits and positive spillovers."\(^4\)

Seaton does define the importance of "Feedback" to architecture. He indicates that Programming may be the necessary

---


base for evaluation, and more definitely that feedback research is different from, and supplemental to, research in Programming. Both however are necessary for the improvement of the methods used for creating meaningful spaces for people.

"Surely in the long run, if not in the short, both clients and designers will come to seek and rely on feedback research as well as research for programming."  

16.1.2 Christopher Alexander*. Alexander, probably the most quoted author on the evolution and synthesis of design process uses mathematical logic and form as the key to solution of architectural problems.

He defines very clearly that the first step in any design process must be a clear comprehension of the problem.

"The following argument is based on the assumption that physical clarity cannot be achieved in a form until there is first some programmatic clarity in the designer's mind and actions; and that for this to be possible, in turn, the designer must first trace his design problem to its earliest functional origins and be able to find some sort of pattern in them."  

He states that the difficulty in architects not understanding the "field" within which their problems lie, and points out how "misfit" in the understanding of a field

* Professor, Dept. of Architecture, University of California at Berkeley.
is easier to define than "fit". He then develops a mathematical system for assigning to "fit" and "misfit" variables, and placing a quantifiable value on each.

Alexander then expands this concept into slightly more specific terms in that he suggest that any problem can be broken down into sub-problems, which subdivision will result in manageable problems. He believes that the solutions of problems lies in the understanding of the interaction between sub-problems (the degree of "fit" and "misfit") not in the sub-problems.

Laborious mathematical relationships are then developed for the understanding of these sub-problems. And then the solution progresses to the next higher problem set of sub-problems; developing until the initial major problems are understood and can be responded to.

Essentially his system attempts to break the defined problem down into sub-problems, understand these and then build up solutions in almost reverse procedure - building up again, as it were.

The system is involved and laborious and hinges upon the quantification of variables based on human behaviour, and
just this is an enormous and questionable task. Questionable in that the basis is value judgements at an ever increasing scale.

"(The work of Alexander may be considered this kind of analysis, finding the best solution to a series of questions, moving to the next higher question, finding the best solution to that, and so on.) Myer and Krauss don't think this is a useful system for the architect, for reasons brought out by their case study - because you can't put a quantitative value on every variable, and because you don't want to avoid the hot dynamic issues of design, and because you want to consider the design as a whole, and consider variables at any time, and so on. It is possible that Alexander himself is dissatisfied with the limitations of his approach, they suggest."[7]

While Alexander does define Programming very clearly, the emphasis of his study is synthesis, and thus its direct relevance to this study is limited; the value being in his thoughts, quoted earlier, with reference to problem statement, as a precondition to design method.

16.1.3 Herbert H. Swinburne*. Defines his thoughts on "A Systematic Approach to Architecture".

Two principle points he makes are of significance here. The first being his concern for Feedback as a necessary part of the system and in this regard that 'design' and 'performance' must have the same base. He describes this

---

* Architect; Philadelphia, Pennsylvania
as follows:-

"Strategy and Sequence.

Relevance of the DESIGN process to the total ARCHITECTURAL process.

1. Definition
2. Analysis------Design------
3. Synthesis------
4. Development
5. Implementation
6. Operation
7. Evaluation Performance--

The criteria for evaluating the performance of any facility in use must be the same criteria used for the design concept. Design can't be established on one base and its performance judged on another.

Performance is not one explicit value. It is a profile of many elements, viewed by many evaluators, and ranging from poor to excellent - from minimum to maximum."8

Secondly he cites the areas of concern, or the concerns of Programming. His sub-title here is:

"Programming - Prelude to Design.

Investigation - Research - Analysis and Systematic Organization of Data for:

1. OPERATIONS. The defined physical arrangements of machinery, equipment, furnishings and their supply and energy systems required to accomplish any task or assignment such as: sleeping, teaching, manufacturing.

2. HUMAN. The specific or casual arrangement of people, singly or in groups, in any prescribed active or passive situation, that considers their needs and desires - their prejudices and aspirations - and the degree of control or permissiveness over their movements, privacy or initiative.

3. SPATIAL. Those positive three dimensions of length, breadth and height - of clearances, distances and relationships - of flexibility, adaptability, expandibility - all required for symbiosis between people and each other, between man and machine, and between machine and machine.

4. ENVIRONMENTAL. The definition, interior and exterior, of the physical setting dealing with thermal, luminous and sonic conditions required by HUMAN beings performing certain OPERATIONS in a specific SPACE. Also the establishment of a behavioural setting that generates impact through mood, emotions, symbol and image.

He does not however, become specific, in this paper, as to methods that ought to be employed in actually gathering this information. The weakness of his Programming definition is that is does not differentiate the uses of all the data he cites as part of Programming. There seems to be very little structuring of information implied; that is there is little concern as to what kind of information is needed and at what stage in the process of design. He implies that all information is part of this "Prelude to Design". He does emphasize the organization of the data; what is lacking is, perhaps, hierarchical structuring.

16.1.4 Harley, Ellington, Cowin and Stirton (Hereinafter referred to as HECS)*


* Harley, Ellington, Cowin and Stirton, Inc., Architects, Planners; Detroit, Michigan.
Under the direction of J. Arthur Miller, AIA, HECS has redefined its professional objectives and set new directions for its problem-solving expertise. Not all of the firm's activities are in building projects; some being the "non-building" type of study mentioned before.

The firm's objectives are:-

"1. To identify with owner and user's goals, and to define all relevant issues to insure that the right problems are being solved.
2. To discover all opportunities inherent in the problem and to generate creative answers which solve all aspects in a simple and direct manner.
3. To insure that the design solutions, when there is to be a physical product, are implemented efficiently, economically, and true to the design intent.

HECS sees five important requirements necessary to insure that these objectives for performance will be met:
1. A tightly knit creative team which includes owner and user.
2. A communications framework which insures a productive dialogue with the owner and user.
4. The ability for thorough documentation.
5. Conscientious and capable construction management abilities."

HECS also discusses six important features of their process.

i) The Team. A Multidisciplinary team is used. It consists of programmers, Comprehensive planners, specialists in financial analysis and architectural and engineering analysis and design. They stress that the team acts as a

---

"single professional generalist".

ii) Communications. Clients are encouraged, (and expected) to join the team in the initial statement of the problem as well as later establishment of quality controls. Whenever possible clients are exposed to existing standards, and where this is not possible are "trained" via slides, diagrams and other visual aids.

iii) Meaningful Dialogue. Early establishment of a "common language" is one of the early functions of the Team.

iv) Rational Process. All office procedures are systematized to improve understanding, solution and production.

v) Thorough Documentation. All aspects are documented for use during "project-life" as well as for office records.

vi) Construction Management Capability. This is, HECS believes, the "key" to the firm's ability to deliver-within budget and time limits - the physical products which usually result from its professional services.

The published matter on the HECS process is very limited, but certain points of interest emerge.

a) The firm believes that by their Process they are able
to "deliver," on time and within budget.

b) They **start** with a statement of the problem examine the data and then **re-state** the problem. This is questionable — how are they able to state the problem **before** examining the data? Unless this is a simplistic statement such as "School," merely as a starting point.

c) The process is not the same for each project, only similar, and this is perhaps its weakness. It lacks some overall "Policy Control" to "tighten up" the system and simplify it.

d) The firm has done several non-building studies, and this attests to the flexibility of the process.

The process is relatively new, and over time will obviously improve; but in terms of examined processes is worthy of note and study.
Section F: ASSESSMENT

17.0 Programming and the Profession.

17.1 The Fee Structure. Obviously techniques such as Programming, as defined, raise some questions as to fees for services. As the service of the architect improves and increases this might well mean a change in the fee structure. The author feels it necessary to state a position with regard to this.

17.1.1 Since Programming, as defined herein implies a better allocation of human resources as well as a potential reduction in design time and fewer charges during design development, the author believes that Programming can and should be done by the architect (it is in his interest to do it) under the existing Scale of Fees.

17.1.2 With regard to the non-building uses of Programming, fee arrangements would have to be agreed upon with the client.

The AIA is doing a study currently (on Programming) which deals with both these problems, and the author felt that in the context of this thesis further, more detailed study would obscure the direction of the argument.
18.0 Programming; The Final Overview.

18.1 It is the contention of the author that a central issue facing Architecture today is the initial statement of the problem. This has been defined as Programming. The criteria for Programming, as defined in Section D are:-

The Programming Process must recognize and respond to

(1) Flexibility
(2) Feedback
(3) Experience
(4) Communication

as overall controls.

18.1.1 Programming is not an isolated activity but a process within a larger system. In architectural terms, a process of problem definition which interfaces with a process of design within a total project delivery system. The author emphasized the need for a "theme" running through the processes that relate each to the other and hence to the overall "envelope".

18.1.2 Programming, by virtue of the fact that it is problem definition has applications beyond use only as the first stage of a project delivery system. Uses where the statement of the problem is the end product and could become the programmer's domain. Feasability studies and
Master Planning studies were two examples cited.

18.1.3 Programming also has relevance in the equitable division of labour within an architectural firm.

18.1.4 Important too is the understanding that different information is needed at different stages within the total system, and the Programming process should be defined around this concept.

18.1.5 Perhaps one of the greatest effects that Programming could have on the improvement of the product (ultimately the environment) is the base it forms for the evaluation of completed projects, and this should be a part of the definition of any Programming process. That is the role that the Programming package will play in later analysis.

18.1.6 Several firms which came under study during research for this study, use Programming techniques of some kind or another; but very few use the same procedure for each project and even fewer document their process in any way. Consequently communication with respect to different processes used in practice is of a very low level, and feedback with respect to (in this case) Programming is not often encountered.
The author believes that, at this point in time, for Programming as a procedure or process to develop, architectural firms must define and document their processes.

There can then be a) initial realization, within the firm, of their process' shortcomings' and b) the development of a dialogue on Programming which will be of benefit to all concerned.

18.2 Most of the concepts of the above are scope for study within themselves. Two concepts, however, have overriding importance with respect to an understanding of Programming, and these are illustrated (overleaf) by means of two diagrams.

18.2.1 (See Figure 1.) The Statement of the Problem is the product of Programming. Programming starts with goals and ends with the Statement of the Problem.

18.2.2 (See Figure 2.) Programming and Design are processes within an overall Project Delivery System. The envelope which contains them is Management; and their interface is the Statement of the Problem.
19.0 Case Study Photographs

19.1 The following are examples of the CRS Analysis Cards referred to in Section E (14.2.1)
Credit Dept. shouldn't have this traffic.

Do this precisely:
1. Necessary?
2. Desirable?
3. Unnecessary?

The computer is

Interesting to see
(assume that it's
displayed to the
public).
1. Desirable
2. Undesirable

People bank where it is most
CONVENIENT

Data Processing Personnel

Is a mechanical garage
PROGRAMMING PROCESS

• ESTABLISH AIMS
• COLLECT, ORGANIZE & ANALYZE FACTS
• UNCOVER & DEVELOP CONCEPTS
• DETERMINE NEEDS
• STATE THE PROBLEM
APPENDIX II

20.0 Caudill Rowlett Scott. An evaluation of three different applications of the CRS Programming Process.

20.1 Master Planning Study; Technical and Community College.*

20.1.1 Contents of the "package". The study has been divided into five broad categories.

(a) Background. A short case history of the project defining the client, the history and broad description of the parameters of the firm's commission. This opening statement concludes with CRS's statement as to their view of the meaning of a Master Planning Study.

"Programming and planning for the Northern Branch began in late 1968. From the beginning of the planning process, the college and the CRS team realized that the only sure constant on a campus today is change, and that the adoption of a static 'master plan' would only breed obsolescence. This report, therefore, not only contains the concepts and planning recommendations developed, but also summarized the goals, problems and premises which led to the report then becomes twofold:

Serves as a guide for the long range development of the Downtown Campus.

* Derived from draft copy of proposed published report.
Provides a basis for continuous refinements as new problems arise or premises change."

(b) Programming. This section has been very carefully defined. The five steps in Programming (See Section E.13.4) are explained under the heading of "Procedure". Each of these "steps" are then examined individually.

(i) College Aims: Scope and Guidelines, (AIMS). The aims are a series of broad statements of intent, with regard to role of the college in the community - facility that will change constantly - with the emphasis on technical skills, educational guidance etc.

Scope and Guidelines consist of general statements such as:-

"Within the context of the college's aims a general compilation of anticipated programs and enrollments has been made. This has served as the basis for the calculation of space requirements."

There are also several short statements, presented one at a time each with a small diagram illustrating the point. For example:-

"Facilities will be available eight hours per day, five days a week for day programs.

"For programming purposes it is anticipated that each teaching station will be utilized 75% of the available time.

Each time a teaching station is in use it is anticipated that 80% of the student stations will be utilized.

...Technical Community College will strive to maintain
a faculty/staff ratio of 1 faculty member for each 16 full-time equivalent students.

The commitment to a strong, effective guidance system will be implemented by a staff of one professional counselor or for each 100 full-time equivalent student."

(ii) Site and Environs (FACTS). Once again, the information is presented one fact at a time, each with a small diagram illustrating all facts affecting site and environs, such as winds, climate, position in city, topography, movement patterns of traffic around site, relationship to transport routes etc.

(iii) Concepts. The same techniques of singular statements each illustrated with a small freehand sketch (or "bubble diagram") make comprehension very simple and clear. These are educational concepts in that they are decisions of policy on how the facility is to be designed with respect to the educational methods. For example:-

"The Learning Resources Center will function as a learning materials warehouse. Materials will be distributed to students and faculty through the LRC outlet in each student service center.

Faculty Offices will be developed within a framework of division chairmen, department chairmen, and non-supervisory faculty. Widely dispersed groups of faculty offices will facilitate coordination at each level while promoting an interdisciplinary faculty mix.

A Student/Faculty Commons will be developed for service functions of a centralized nature. Areas for student activities and organizations, a faculty
lounge, multiuse conference areas, and a book store will comprise the commons."

(iv) Parking and Building Requirements (NEEDS). General statements which refer to the more detailed analysis results of the appendix are given here.

"Building space requirements are given in detail in the Appendix. Net areas required for buildings at two levels are as follows:

<table>
<thead>
<tr>
<th>Students</th>
<th>SF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000</td>
<td>109,675</td>
</tr>
<tr>
<td>3000</td>
<td>283,475</td>
</tr>
</tbody>
</table>

The majority of students, faculty and staff are anticipated to arrive by automobile. Parking requirements are, therefore, based upon heavy automobile use. Criteria for these calculations may be found in the Appendix."

(v) Statement of the Problem. The Programmatic Statement results from all the above analysis. This is the interface between Programming and the schematic design process that follows. Nine general statements are made here; as before, each separately, with its own "bubble diagram".

"The campus should posses an Identity within the context of Northern Delaware.

As a major resident in an Urban Redevelopment the campus should Respond to the Adjacent Environment. The plan should recognize the need for Interaction with the adjacent community and facilities on-campus development of Community Use facilities.

Elements of both Community and Campus Circulation should be recognized as major influences.

The plan must respond to the Climate of the region.

The ever-changing nature of Educational Needs and Community
Influences call for a Flexible campus plan.

The campus must be planned for economical Phasing to later accommodate a Tripling of the initial enrollment. This expansion should recognize the phasing of the adjacent elements of the central city redevelopment.

The form of the Steeply sloping site should influence the form and location of both Buildings and Exterior Spaces on the campus.

The college's commitment to Interdisciplinary mix necessitates a plan which will allow Maximum Exposure of students and faculty to all learning experiences.

The Student Service Centers should be recognized as major conceptual elements fulfilling the Home Base needs of the commuter student.

20.1.2 Comment. The statement of the problem is clear in itself and also in that its derivations can be clearly seen. For later reference all the pertinent facts, concepts etc. are clearly recorded, and each separate.

The progression is clear, and is explained at the outset. The appendix contains the specific area requirements obtained from the "brown sheets" technique as well as a cost estimate broken down into understandable chunks.

The package is "clear" both in its content from Programming through Design Process to Comprehensive Plans.

Information is clearly available for later reference either by the designer or the client. It is easy to "find" the
information as the structure of the report is clear.

The value then is in its clarity its conscieness and the development of each stage of the Study upon the foregoing stages.

The weakness in this report, with respect to overall CRS "Triad Theory" is that the triad of Form, Function and Economy is not clear, in that facts, concepts, etc. and the final problem statement are not structured into this framework. Obviously they were gathered in this way and the design will be done through this framework, so the Form-Function-Economy stratification could have been stronger (and thus the final product even clearer).

20.2 Flexibility Study: Convention Center for a Coastal City.*

"In October 1968, the City of Corpus Christi authorized the firms of Swanson Hiester Wilson Boland and Caudill Rowlett Scott to undertake a Convention Facilities study. The purpose of this study has been to determine the types and sizes of facilities needed, to investigate and recommend a site, to survey existing buildings, to develop a site utilization plan, and to illustrate a design concept in order to project estimated costs and establish possible methods of phasing."

*Corpus Christi Convention Center. Published report (Feb 1969) Caudill Rowlett Scott with Swanson Hiester Wilson Boland ( Architects Planners Engineers, Corpus Christi.)
The study is ideally suited to Programming (as defined in Section D) since essentially there was "no problem" - the firms were asked to formulate the problem based on investigated flexibility.

20.2.1 Contents of the "package". The table of contents illustrates very clearly the comprehensive nature of the study.

"(A) GOALS

(B) RESEARCH & PROGRAMMING
   Economic Background
   Investigation
   User's Conference
      Participants
   Program
   Analysis of Existing Facilities

(C) ANALYSIS & PLANNING
   Land Use
   Major Traffic Arteries and Traffic Volumes
   Topography
   Site Selection
   Site Analysis
   Response to the Bay

(D) DESIGN STUDY
   Summary of the Problem

(E) APPENDIX
   Cost Estimate Analysis
      Option I
      Option II
      Option III
      Legend"

(A) Goals. Are defined in terms of Economic Goals, Form Goals, and Functional Goals. This is the initial requirement,
"where are we going in this study". It is very simple and clear. Some examples of these statements are:-

i) "Emphasize the uniqueness of Corpus Christi's being water orientated and take advantage of water activity potential. (FORM)

ii) The new facility must reflect a high quality in design and construction. (FORM)

iii) Capitalize on Corpus Christi's growth potential. Tourism and convention activity have grown 850% in the Coastal Bend Region in the 10-year period ending in 1967. (ECONOMIC)

iv) The facilities must attract the revenue of delegates and their return value as tourists. (ECONOMIC)

v) The convention facility must provide the possibility for more than one function to take place at a given time. (FUNCTION)

vi) The latest trends indicate that the related convention facilities should be 'under one roof' for maximum convenience of convention-eers. (FUNCTIONAL)

(B) Research and Programming. The comprehensive nature of this portion is clear in the contents. Two techniques used to gather information are of special note. Firstly the "User's Conference".

"The second phase of the investigation was to invite a group of highly qualified professional executives of state, regional and national convention-orientated organizations to Corpus Christi to review their convention requirements and to assist in the programming and planning stages of the study. The conference was conducted in the form of a round table discussion to promote the exchange of ideas and opinions. The User's Conference contributed immeasurably toward developing a building program fulfilling the projected needs of Corpus Christi."

Some of the executives were potential users, some administrators, a management consultant etc., together with
The report then summarizes the user's conference (Priority of Needs, need for flexibility, "under one roof" concept etc.). The point is that much came out of this "mix of specialities", in terms of conceptual considerations.

Secondly the investigation and tabulation of the features of several convention centers around the country which were visited or researched by the Programmers. This clear tabulation makes it very easy for the reader (user, client, financiers etc.) to draw comparisons. Further matrixes showed recently opened facilities within the region, subdivision by included facilities etc. (Refer to table of contents above).

(C) Analysis and Planning. Once again the contents "tells the story" - and the facts are presented with clear carefully drawn diagrams making it easy for

(i) initial decision making

(ii) later perusal by readers of the report.

Facts are presented one at a time and cover existing as well as proposed facilities. Thus the progression to the next section the "Statement of the Problem" was clear and ordered.
(D) Design Study. (Summary of the Problem). The problem was stated as a series of short phrases. These were:

"The Convention Center includes an exhibit hall with meeting rooms, an arena, and an auditorium. The facilities serve for convention functions and community use. Therefore, the center should function for a single occupancy by a large group or simultaneous occupancies by separate small groups.

Recent trends in convention planning involve the convention delegate's family in planning activities in conjunction with the convention. Therefore, the Convention Center should provide unique facilities to accommodate this trend.

The Convention Center site is adjacent to waterfront property presently serving public use. Therefore, the center should be a good neighbor to the adjacent properties.

The presence of the Convention Center generates parking requirements for large numbers of vehicles. Therefore, the center should be designed to allow for the required parking without restricting off-site traffic flow.

The Convention Center is to occupy and share the site with the existing coliseum facility. Therefore, the new facility should allow for its continued operation.

The waterfront site is unique to Corpus Christi. Therefore, the Convention Center should touch the water and establish an activity connection at the water.

The building site is adjacent to the water. Therefore, because of soil conditions, the new facilities must be designed for storm conditions (high water, wave action, high winds) and must be structurally supported on deep piles.

The Convention Center must accommodate phased construction. Therefore, the probability exists that the total project will not be built at one time.

The Convention Center is bound by major through traffic arteries. Therefore, the new facilities should
minimize the pedestrian-vehicular conflict.

The Convention Center will be viewed from all sides, including the bay. Therefore, it should be handsome from all sides and from above.

The new facilities are to be distinctive (show place), and clearly identifiable as a convention center. Therefore, they must reflect a high quality in design and construction.

The proposed budget is adequate for good quality construction. However, it is not without design constraints.

These statements were the basis for the schematic design study. A series of plans, drawings and model photographs give a clear idea of one possible solution of this problem. Presentation is clear and uncluttered and once again, "easy to read".

(E) Appendix. The appendix is interesting in that it gives not only a carefully constructed cost analysis (closely related to the gathered facts and summary of the problem) but also projected maintenance costs and a potential revenue calculation.

Again these are (i) related back to the User's Conference, as well as other derived facts, (ii) clearly documented and "easy to read".

Information as to costs for different Phasing possibilities
is clear; and the information with regard to potential revenue is not only useful but put down in a way so as to be easily understood in "pieces" by the reader.

20.2.2 Evaluation. The document is comprehensive, clearly stratified, and very "readable". The comprehensive yet unclouded nature is patent to the reader, but the important features for the author are (i) that the information has been stratified with apparent simplicity, and more importantly, (ii) was produced in such a relatively short span of time. (Commission was in October 1968, date of the published report is February 1969).

The CRS programming process not only has the capability for such 'non-building' studies but is obviously well suited to handling them. The document (and hence the study) is remarkable.

20.3 Problem Statement as a basis for "feedback" (Project Evaluation).

20.3.1 The Problem* was stated as twenty four goals for the design, each derived under Form Function or Economy.

* Jesse H. Jones Hall for the Performing Arts, Houston. Intra-office record, CRS.
Since this was too large a number of variables for the designer to consider it was condensed to the following.

A  "SUMMARY: The six tasks of the architects

How to accommodate the movement of 3000 people.

How to seat 3000 to best see and hear a performance.

How to do this also for 1800.

How to make a stage serve a variety of performances.

How to do these things within a $6 million first-cost and a self-sustaining operation cost for the city.

B  THE CHALLENGE: The essence of the problem

The purpose of any performing art is to communicate a thought or feeling in a real or abstract manner.

Our challenge is to design a building that will prepare the patron for receiving that communication. The preparation should be through a series of experiences that can be enhanced or amplified by the architecture—and the experience should begin when one first sees the building from a distance—and should not end until he is on his way home."

This is the statement of the problem. Obviously it was backed up by area and space requirements etc. The question is; is this the base for Feedback research, or should it include the specific area requirements? It is the opinion of the author that the program statement does provide the necessary basis for feedback research, since feedback must examine the way in which design has responded to the stated goals. Was the uniqueness of the problem achieved?

20.3.2 Points of Concern. Certain factors must be borne
in mind with regard to "feedback". Feedback measures the Design Process, not the Programming Process.

If the building is examined with respect to this problem statement the result will be the success of the Design Process in responding. It will not measure the validity of the Program. Obviously there are many ways a designer could respond to the problem statement, and feedback on this base will examine only the degree to which he was able to respond successfully. It is not possible to evaluate the program statement by examining the final product.

Whether the problem statement is correct or not is examined during design - it is here that the goals may be challenged.

Feedback must have a base; obviously one must be able to structure this examination, but it is Design we are examining, not Programming.

It is clear however, that Programming (as defined in Section D) when it is the statement of the problem in general terms as with the case outlined above, will not be able to be measured by for example, user reports. This is an examination of the way design has responded to the general constraints of the Program with respect to the information at its disposal.
21.0 BIBLIOGRAPHY


BARNETT, J. "Will the Computer Change the Practice of Architecture?" Architectural Record. (January 1965) P. 143-150.


BROADBENT, G. "Design Methods in Architecture". Interbuild/Arena (February 1968) p. 14-19 (Reprint)


BUILDING FOR THE YEAR 2,000 (Cover story) Time (August 2, 1968) p. 39-43.


COLQUHOUN, A. "Typology and Design Method". Interbuild/ Arena (June-July 1968) p. 11-14.


EBERHARD, J. P. "Technology for the City" Science and Technology. (September 1966) Reprint.


FREEDMAN, P. *The principles of scientific research.*


HALL, A. D. *A Methodology for Systems Engineering.*


JACKSON, B. "The Relationships between needs are the Elements of Form." *Zodiac 17.* Reprint.


PENA, W. M. "People and Program....Background for the Architect." Reprinted from the 38th Yearbook, Department of Elementary School Principals. Washington, D. C.


