Fourth Edition

Problem Seeking

An Architectural Programming Primer

William M. Peña Steven A. Parshall

Problem Seeking

This page intentionally left blank

Fourth Edition

Problem Seeking

An Architectural Programming Primer

William M. Peña Steven A. Parshall

John Wiley & Sons, Inc.

Acknowledgments

HOK Team

Editor:	Sofia Fonseca de Niño
Desktop Publishing	Sharon Burton
Production Manager:	Lauren Gibbs
Text Production:	Allison Ehrlich
Special Contributors:	Kevin Williams, Bill Senseman
Research Assistants:	Yuji Mizoue, Scott Tucker, Gagan Singh, Heather Jordan, Conny Brown
Cover Graphics:	Hal Kantner
Photography:	Jim Olive, pp. 17, 67, 190, 205
Photo Enhancements:	Adrian Arroyos

We are grateful to those programmers, past and present, who have contributed to this book— some much more than others—but all contributing more than they realize.

Published by John Wiley & Sons, Inc., New York.

This title is also available in print as ISBN 0-471-12620-9. Some content that may appear in the print version of this book may not be available in this electronic edition.

For more information about Wiley products, visit our web site at www.wiley.com

Foreword

William Peña directed the first edition of *Problem Seeking* to clients, to business and facilities planning officials on the staffs of institutions, corporations, and various public bodies. This was because client participation is so critical to the success of projects. However, practicing architects and architecture students discovered that first booklet, and as a result, it was used in 1973 by the National Council of Architectural Registration Boards as a basis for the predesign part of the professional exam.

The second edition, then, was written for clients, as well as architects and students. With that edition, the Problem Seeking method joined a long list of programming methods being introduced in the late 1970s.

The third edition, published in 1987, recognized that traditional architectural services had expanded to include predesign programming at the beginning of a client's decision-making process and building evaluation services at the end. Postoccupancy evaluation makes the client's role even more vital in reaching decisions on which architects can base sound design solutions and can test them after the building is occupied.

In 1994 Hellmuth, Obata + Kassabaum (HOK) acquired CRSS Architects, which had evolved from the original firm of Caudill, Rowlett and Scott (CRS). Now HOK is undertaking the publication of this fourth edition of *Problem Seeking*. HOK's practice was founded on the same principle as CRS—both firms viewed design as problem solving. Furthermore, many of the principles and techniques presented in this book can be attributed to Bill Caudill, one of the founders of CRS, and an AIA Gold Medalist.

More importantly, the broad range of principles and techniques presented in this book has evolved over a long period of architectural practice. They are not the product of one person, but the accumulated efforts of many members of CRS and now HOK. We are proud to continue the tradition of involving and interacting with clients in architectural programming as the first step of the design process.

Gyo Obata
Co-Chairman
Hellmuth, Obata + Kassabaum, Inc.

Contents

Preface

Part One
An Architectural Programming Primer

Part Two
The Appendix

Preface

This book is the fourth edition of *Problem Seeking*. The first edition in 1969 was based on 20 years of research and practice in architectural programming. The subsequent editions evolved over the next 30 years through changes in communication techniques and expanded scope of applications, although the theory remained intact. This edition, then, has the advantage of experience of some 50 years of professional application—**indicating a practice-tested validity**.

This is a two-part book. **Part One** is a primer on programming. It is written to help you understand **one programming method**, whether you are an architect, a student, or a client getting ready to start a building project. **Part Two** is the appendix—a collection of definitions, examples, considerations, activities, and techniques that expands upon the principles explained in the primer.

What is new in this edition?

Throughout **Part One**, we inserted quotes from speeches by Willie Peña to enrich one's understanding of problem seeking. In response to client interest in participation and consensus building through programming, we added sections on ways of thinking and effective group action, again based on speeches given by Willie Peña.

Duncan Sutherland coined the term "officing," and in 1987 he and Steve Parshall published *Officing: Bringing Amenity and Intelligence to Knowledge Work*. Soon after, Martha Whitaker published an HOK print forum on Alternative Officing, which led to an International Facility Management Association (IFMA) research report on Alternative Officing, defining emerging types of officing settings. Summarizing this work, we added six officing settings to the section on programming concepts.

In light of new forms of electronic publishing, **Part Two, The Appendix**, has undergone a major rewrite, not only to add new material, but also to improve the reader's ability to find supplemental information when undertaking a programming assignment.

We expanded the glossary and examples in the section On Needs. It includes broader definitions of area and methods of measurement that are consistent with industry standards. We provided a set of building efficiency factors that are related to the area of definitions and that explain the mathematical relationships among them.

Along with updated building costs, we added further explanation of the relationship between building costs and interior fit-up costs, including considerations for renovation projects. More clients are using financial analysis for selecting programmatic options; therefore, we added a primer on discounted cash flow analysis and net present value analysis to complement the section on cost estimating.

We organized the problem statement examples so that they are referenced by project phase (master plan, schematic design, interior design) as well as building type.

We reworked the section on programming activities to provide a better reference to the discussion of degrees of programming and type of deliverable to be produced. By moving the section on typical programming activities, we provided a better focus to the section on useful techniques. Beginning with the use of questionnaires, this section covers data management, functional relationship analysis, updated sections on brown sheets and analysis cards, and a new section on electronic presentations.

Finally, the reformatting of pages anticipates the future electronic publication and access to a living document of tools, techniques, and guidelines for the advanced programmer.

William M. Peña, FAIA
Founder
Caudill, Rowlett and Scott, Inc.

Steven A. Parshall, FAIA
Senior Vice President
Hellmuth, Obata + Kassabaum, Inc.

This page intentionally left blank

Part One

An Architectural Programming Primer

The Primer

Good buildings don’t just happen. They are planned to look good and perform well, and come about when good architects and good clients join in thoughtful, cooperative effort. Programming the requirements of a proposed building is the architect’s first task, often the most important.

There are a few underlying principles that apply to programming—whether the most complex hospital or a simple house. This book concerns these principles.

Programming concerns five steps:

1 Establish **Goals**.

2 Collect and analyze **Facts**.

3 Uncover and test **Concepts**.

4 Determine **Needs**.

5 State the **Problem**.

The approach is at once simple and comprehensive—simple enough for the process to be repeatable for different building types and comprehensive enough to cover the wide range of factors that influence the design of buildings.

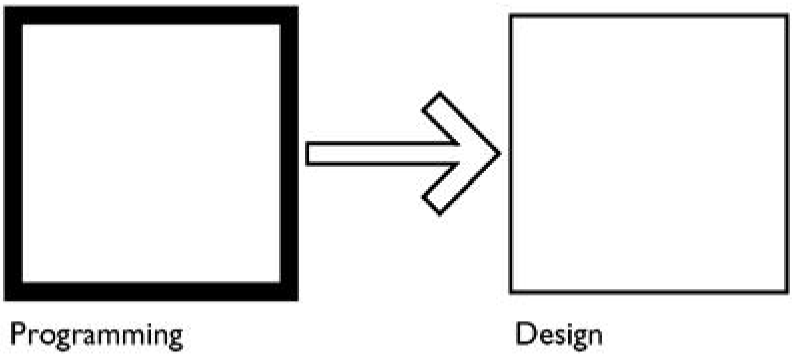
The five-step process can be applied to most any discipline—banking, engineering, or education—but when applied specifically to architecture, it has its proper content, that is an architectural product….a room, a building, or a town. The principle of this process is that a product will have a much better chance of being successful if, during the design, the four major considerations are regarded simultaneously.

These considerations (or design determinants) indicate the types of information needed to define a comprehensive architectural problem:

Function   Form   Economy   Time

Architectural programming, therefore, involves an organized method of inquiry…a five-step process…interacting with four considerations.

The Search



Programming is a process. What kind? Webster spells it out specifically: “ A process leading to the statement of an architectural problem and the requirements to be met in offering a solution. ”

This process, derived from the definition and referred to as the five-step process, is basic. The word “basic” is used advisedly. Since the advent of systematic programming five decades ago, different degrees of sophistication have evolved. But the procedures presented here remain basic to all.

Back to the definition. Note “statement of an architectural problem.” This implies problem solving. Although usually identified with scientific methods, problem solving is a creative effort. There are many different problem-solving methods, but only those few that emphasize goals and concepts (ends and means) can be applied to architectural design problems.

Almost all problem -solving methods include a step for problem definition—stating the problem. But most of the methods lead to confusing duality—finding out what the problem is and trying to solve it at the same time. You can't solve a problem unless you know what it is.

What, then, is the main idea behind programming? It's the search for sufficient information to clarify, to understand, and to state the problem.

If programming is problem seeking, then design is problem solving.

These are two distinct processes, requiring different attitudes, even different capabilities. Problem solving is a valid approach to design when, indeed, the design solution responds to the client's design problem. Only after a thorough search for pertinent information can the client's design problem be started. "Seek and you shall define!"

Programmers and Designers

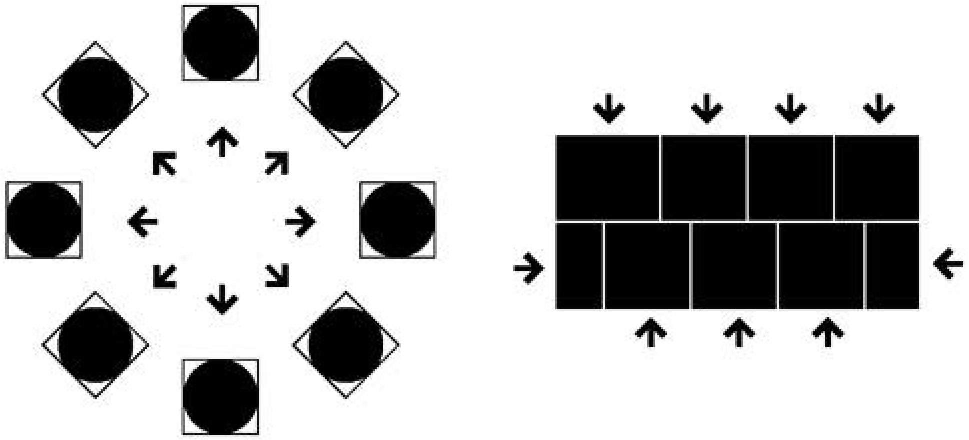
If I were given one hour to save the planet, I would spend 59 minutes
defining the problem and one minute resolving it.
—Einstein

Who does what? Do designers program? They can, but it takes highly trained architects who are specialized in asking the right questions at the right time, who can separate wants from needs, and who have the skills to sort things out. Programmers must be objective (to a degree) and analytical, at ease with abstract ideas, and able to evaluate information and identify important factors while postponing irrelevant material. Designers can't always do this. Designers generally are subjective, intuitive, and facile with physical concepts.

Qualifications of programmers and designers are different. Programmers and designers are separate specialists because the problems of each are very complex and require two different mental capabilities, one for analysis, another for synthesis.

It may well be that one person can manage both analysis and synthesis. If so, he or she must be of two minds and use them alternately. However, for clarity, these different qualifications will be represented by different people—programmers and designers.

Analysis and Synthesis



The total design process includes two stages: analysis and synthesis. In analysis, the parts of a design problem are separated and identified. In synthesis, the parts are put together to form a coherent design solution. The difference between programming and design is the difference between analysis and synthesis.

Programming IS analysis. Design IS synthesis.

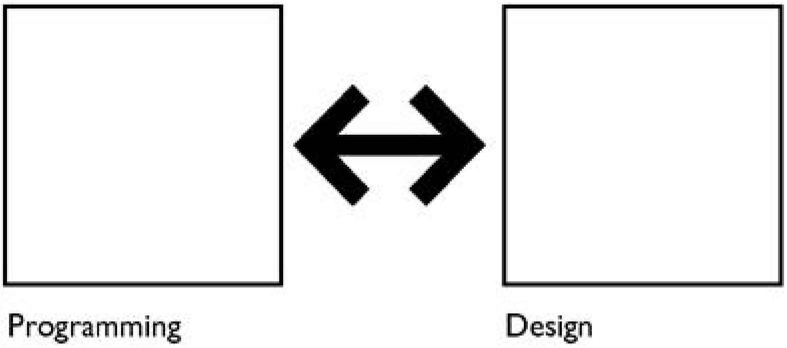
You may not perceive the design process in terms of analysis and synthesis. You may even question problem solving as an approach. You may think of the design process as

Sometimes I think we arrive at a solution before we know what the problem is. We say: "My next design will be ***Round!****," without logic or analysis.*
—William Peña

a creative effort. It is. But the creative effort includes similar stages: analysis becomes preparation or exposure, and synthesis becomes illumination or insight. The total design process is, indeed, a creative process.

Does programming inhibit creativity? Definitely not! Programming establishes the considerations, the limits, and the possibilities of the design problem. (We prefer "considerations" to "constraints" to avoid being petulant.) Creativity thrives when the limits of a problem are known.

The Separation



Programming precedes design just as analysis precedes synthesis. The separation of the two is imperative and prevents trial-and-error design alternatives. Separation is central to an understanding of a rational architectural process, which leads to good buildings and satisfied clients.

The problem-seeking method described in this book requires **a distinct separation of programming and design.**

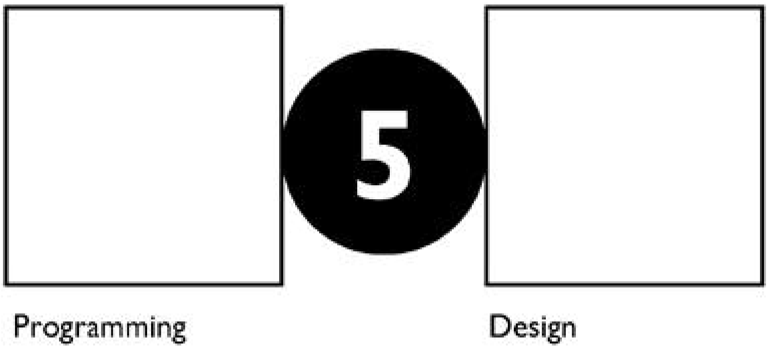
Most designers love to draw, to make “thumbnail sketches,’’ as they used to call these drawings. Today, the jargon is “conceptual sketches” and “schematics.” Call them what

Corita Kent, artist and educator, wrote, "Rule Eight: Don't try to create and analyze at the same time. They are two different processes."
—Today You Need a Rule Book, *1973.*

you will, they can be serious deterrents in the planning of a successful building if done at the wrong time—before programming or during the programming process. Before the whole problem is defined, solutions can only be partial and premature. A designer who can't wait for a complete, carefully prepared program is like the tailor who doesn't bother to measure a customer before starting to cut the cloth.

Experienced, creative designers withhold judgment and resist preconceived solutions and the pressure to synthesize until all the information is in. They refuse to make sketches until they know the client's problem. They believe in thorough analysis before synthesis. They know that programming is the prelude to good design—although it does not guarantee it.

The Interface



The product of programming is a statement of the problem. Stating the problem is the last step in problem seeking (programming), and it is also the first step in problem solving (design). **The problem statement, then, is the interface between programming and design.** It's the baton in a relay race. It's the handoff from programmer to designer. In any case, the problem statement is one of the most important documents in the chain that is the total project delivery system.

While many theorists extol the virtues of the problem statement, few practitioners stop to formulate a statement, to verbalize it. This programming method requires that you actually write out a clear statement of the problem. Since this statement is the first step in design as well as the last step in programming, its composition must be the joint effort of the designer and the programmer.

Five Steps

The competent programmer always keeps in mind the steps in programming: **(1) Establish Goals, (2) Collect and Analyze Facts, (3) Uncover and Test Concepts, (4) Determine Needs,** and **(5) State the Problem.** The first three steps are primarily the search for pertinent information. The fourth is a feasibility test. The last step is distilling what has been found.

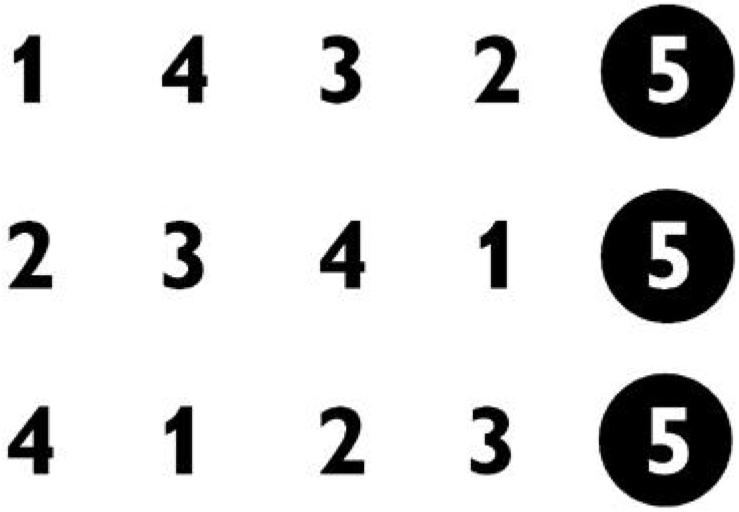
Curiously enough, the steps are alternately qualitative and quantitative. Goals, concepts, and the problem statement are essentially qualitative. Facts and needs are essentially quantitative.

Programming is based on a combination of interviews and worksessions. Interviews are used for asking questions and collecting data, particularly during the first three steps. Worksessions are used to verify information and to stimulate client decisions—particularly during the fourth step.

Briefly, the five steps pose these questions:

1. **Goals**—*What* does the client want to achieve, and *Why*?
2. **Facts**—*What* do we know? *What* is given?
3. **Concepts**—*How* does the client want to achieve the goals?
4. **Needs**—*How* much money and space? What level of quality?
5. **Problem**—*What* are the significant conditions affecting the design of the building? What are the general directions the design should take?

Procedure



1 4 3 2 ❺

2 3 4 1 ❺

4 1 2 3 ❺

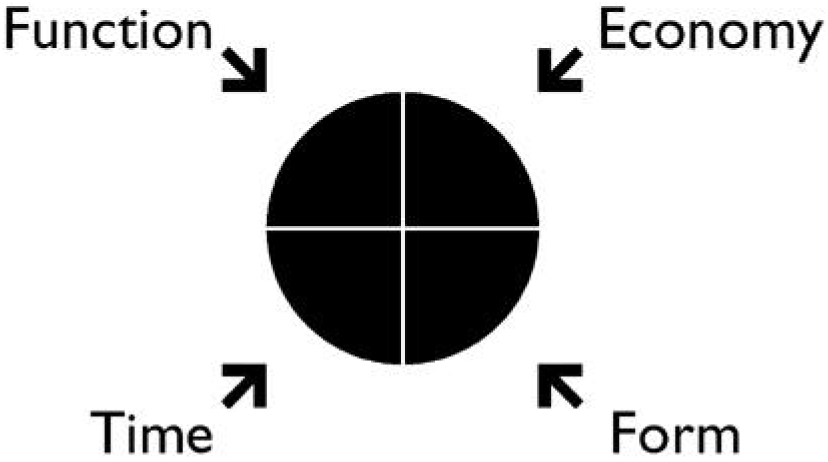
The five steps, then, are not inflexibly strict. They usually have no consistent sequence, nor is the information scrupulously accurate. For example, a 10,000-student university, a 300-bed hospital, and a 25-student classroom are only nominal rather than actual sizes. Information sources are not always reliable, and predictive capabilities may be limited.

The steps and the information, then, do not have the rigor or the accuracy of a mathematical problem. Programming, therefore, is a heuristic process and not an algorithm. As such, even good programming cannot guarantee finding the right problem, but it can reduce the amount of guesswork. The method is just as good as the judgment of the people involved.

Working through the steps in numerical sequence is preferable; theoretically, this is the logical order. But, in actual practice, **steps may be taken in a different order or at the same time**—all but the last step. It is frequently necessary, for example, to start with a given list of spaces and a budget (fourth step) before asking about Goals, Facts, and Concepts (first, second, and third steps). It usually is necessary to work on the first four steps simultaneously, cross-checking among them for the integrity, usefulness, relevance, and congruence of information.

The fifth step is taken only after marshalling all the previous information, extracting, abstracting, and getting to the very essence of the problem.

The Whole Problem

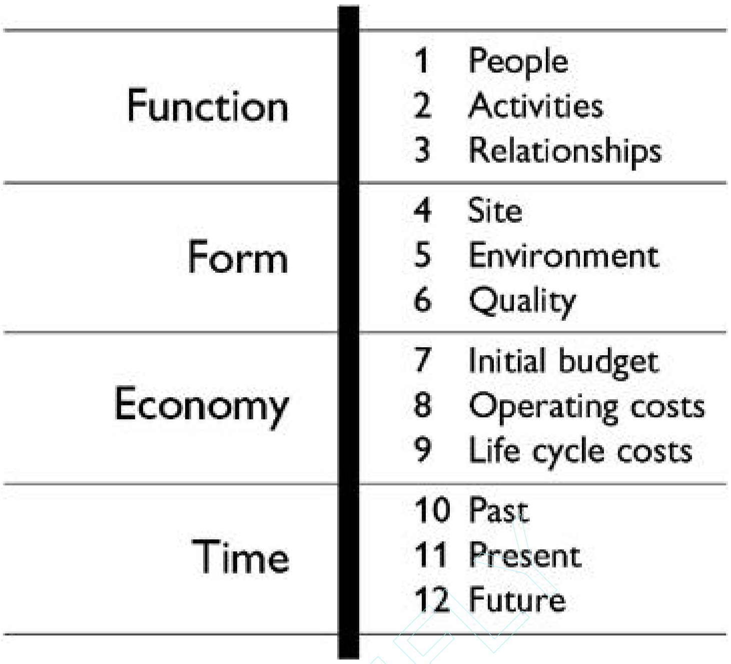


It's important to search for and find the whole problem. To accomplish this, the problem must be identified in terms of **Function, Form, Economy, and Time.** Classifying information accordingly simplifies the problem while maintaining a comprehensive approach. A wide range of factors makes up the whole problem, but all can be classified in the four areas that serve later as design considerations.

Too little information leads to a partial statement of the problem and a premature and partial design solution. The appropriate amount of information is broad enough in scope to pertain to the whole design problem, but not so broad as to pertain to some universal problem. As the Spanish proverb states: “He who grasps too much, squeezes little.” Grasp only what you can manage and what will be useful to the designer.

As a professor might say, “Before you answer individual questions, be sure to look at the whole examination.” Designers should look at the whole problem before starting to solve any of its parts. How can a designer who does not have a clear understanding of the whole problem come up with a comprehensive solution?

Four Considerations



Function	1 People 2 Activities 3 Relationships
Form	4 Site 5 Environment 6 Quality
Economy	7 Initial budget 8 Operating costs 9 Life cycle costs
Time	10 Past 11 Present 12 Future

Take a closer look at Function, Form, Economy, and Time. **There are three key words to each consideration.**

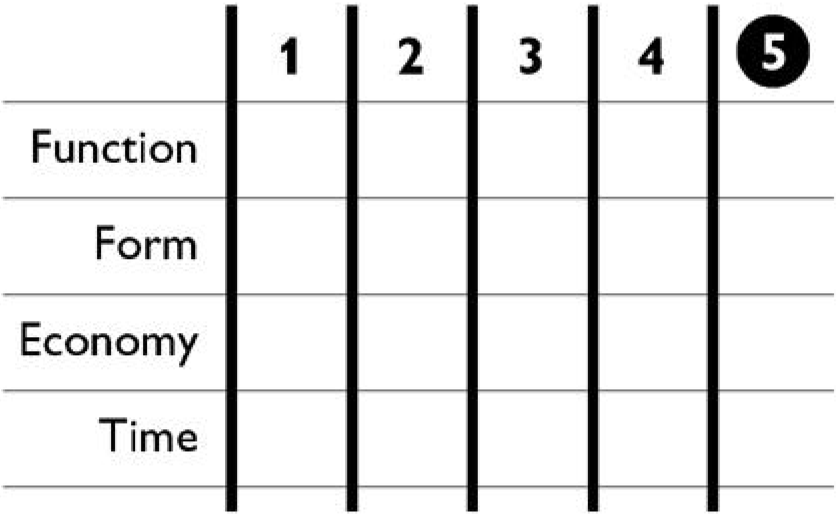
Function implies "what's going to happen in their building." It concerns activities, relationship of spaces, and people—their number and characteristics. Key words are (1) people, (2) activities, and (3) relationships.

Form relates to the site, the physical environment (psychological, too) and the quality of space and construction. Form is what you will see and feel. It's "what is there now" and "what will be there." Key words are (4) site, (5) environment, and (6) quality.

Economy concerns the initial budget and quality of construction, but also may include consideration of operating and life cycle costs. Key words are (7) initial budget, (8) operating costs, anad (9) life cycle costs.

Time has three classifications—past, present, and future—which deal with the influences of history, the inevitability of changes from the present, and projections into the future. Key words are (10) past, (11) present, and (12) future.

Framework



	1	2	3	4	❺
Function					
Form					
Economy					
Time					

Use the four considerations to guide you at each step during programming. By establishing a systematic set of relationships between the steps in problem seeking and these considerations, between process and content, a comprehensive approach is assured. The interweaving of steps and considerations forms a framework for information covering the whole problem.

All four considerations interact at each step. For example, in the first step when goals are investigated, function goals, form goals, economy goals and time goals should emerge. With each of these having three subcategories, the process includes asking twelve pertinent questions regarding goals alone. Since the first three steps constitute the main search for information, three times twelve provides the basis for thirty-six pertinent questions.

Consider these to be key questions. The answers will provide opportunities for further questions. The Information Index on the following pages indicates more than ninety items in these three steps.

Programmers do not have to know everything the client knows, but they should know enough of the client's aspirations, needs, conditions, and ideas that will influence the design of the building. For this, programmers have to know the right questions to ask; they start with the thirty-six subcategories.

The considerations interact in the fourth step to test the economic feasibility of the project, and in the last step, they interact to state the whole problem.

This interaction provides a framework for classifying and documenting information. The classification qualities inherent in this framework are particularly useful in preventing information clogs when dealing with massive quantities of information. The categories are broad enough to classify the many bits of information gathered during programming without nit-picking and indecision.

The framework can be used as a checklist for missing information. As such, the orderly display of information on a wall becomes a good visual scoreboard. One glance at a wall display of graphic analysis material can spot what is missing and needs to be documented. It also provides a format for dialogue among the members of the team.

Information Index

The framework can be extended to serve as an Information Index—**a matrix of key words used to seek out appropriate information.** These key words are specific enough to cover the scope of major factors and are universal enough to be negotiable for different building types. Even if some key words do not seem to apply in a particular project, it is useful to test them—to ask a question based on those key words. If the test proves they are applicable, then those key words will encourage a thorough search for information. They may offer a better and quicker understanding of the project.

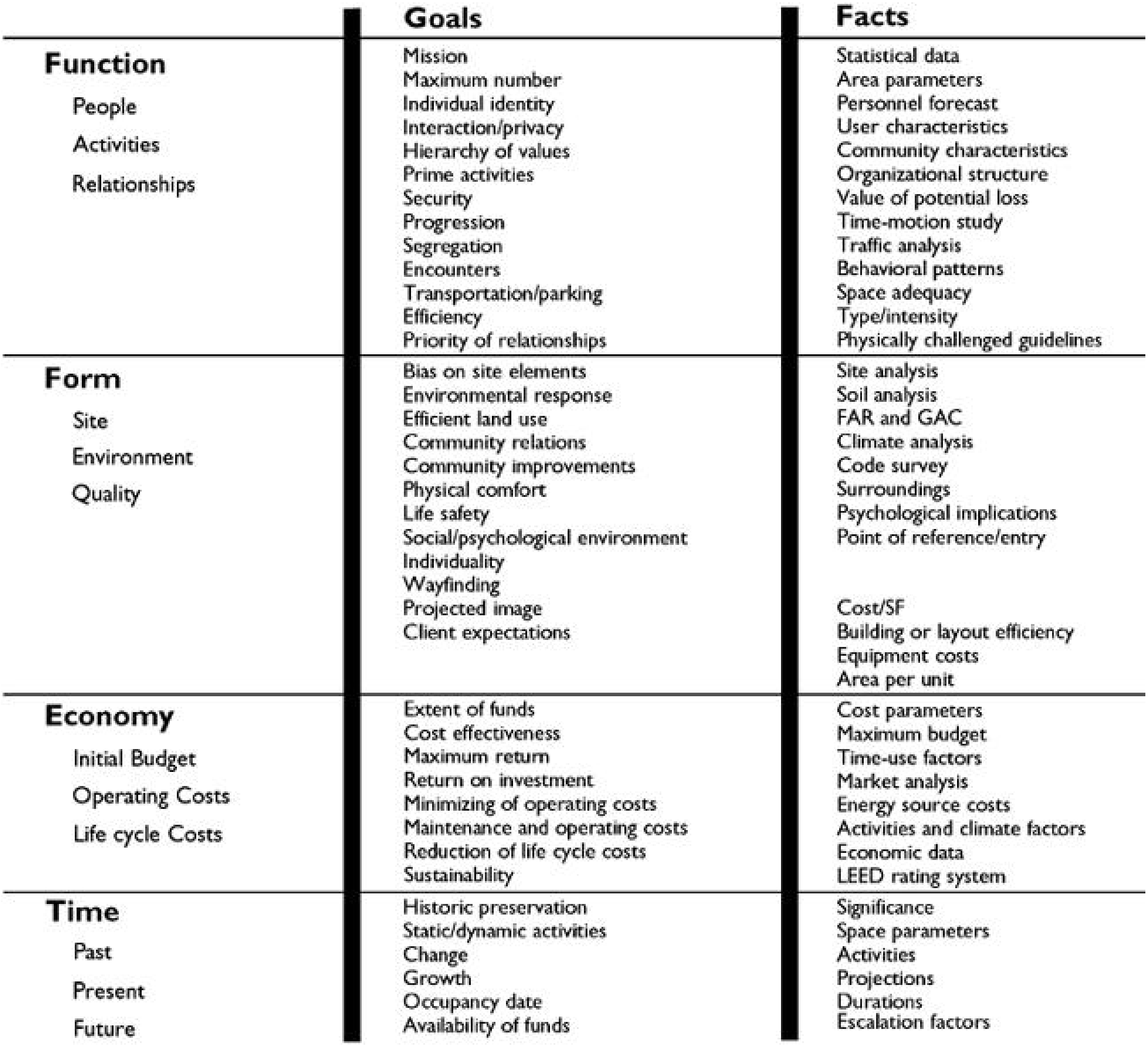
Most key words are "evocative words." They trigger useful information. Charged with emotion as well as meaning, they tend to evoke a response, or even to suggest likely substitutions.

An Information Index may be designed to be very specific and tailored to one building type; but as all such checklists, it would soon be obsolete. A general character prolongs its usefulness.

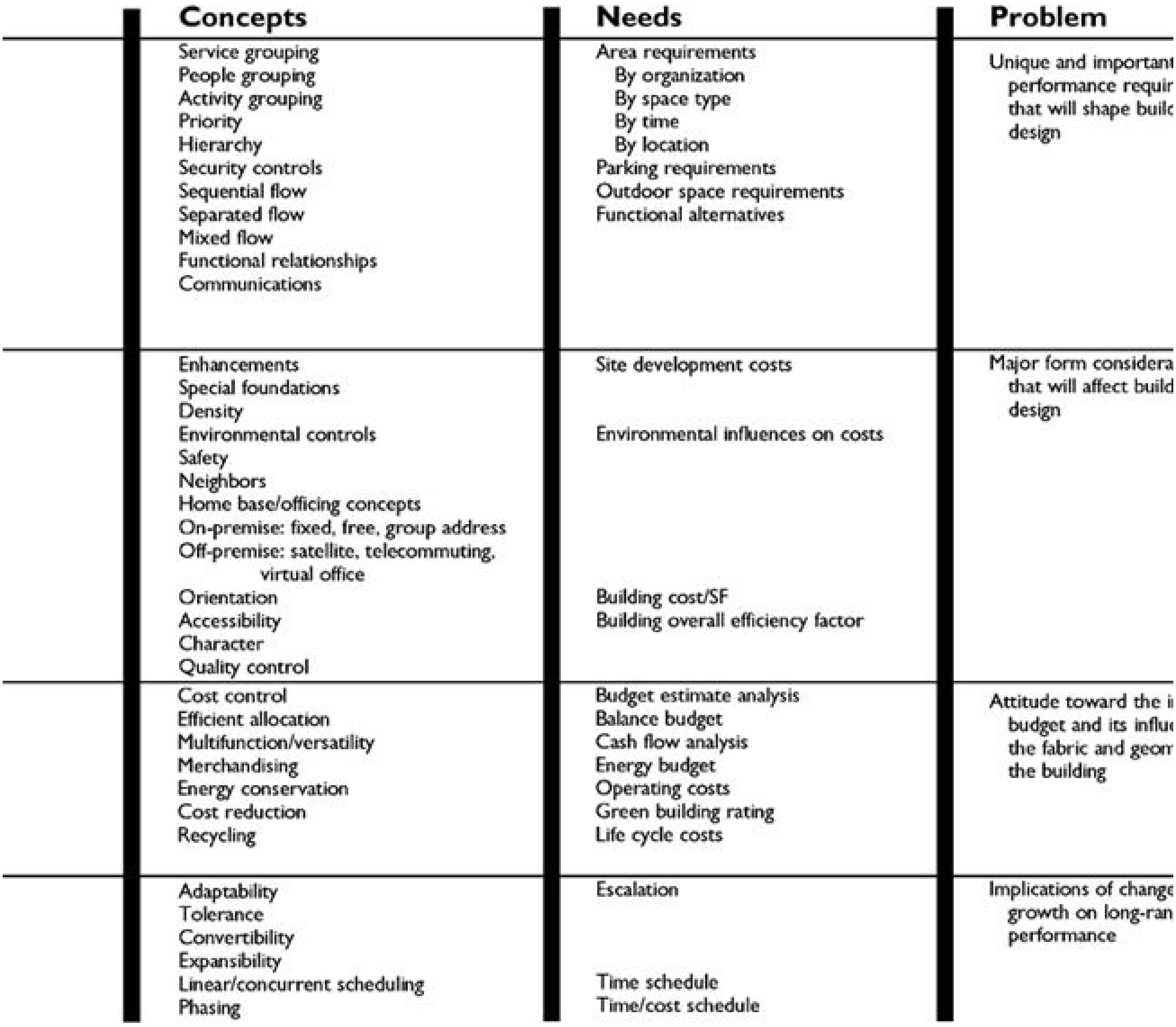
Note that the Information Index establishes the interrelationship of information regarding Goals, Facts, and Concepts. For example, a functional goal for "efficiency" is related to "space adequacy" and is implemented by effective "relationships"—reading horizontally on the index. Also note that items under Needs and Problem are more limited since the fourth step is a feasibility test and the last step is abstracting the essence of the project.

We have adapted the following chart from the ***Architectural Registration Handbook, A Test Guide for Professional Exam Candidates,*** published jointly by the National Council of Architectural Registration Boards and Architectural Record, 1973.

Information Index

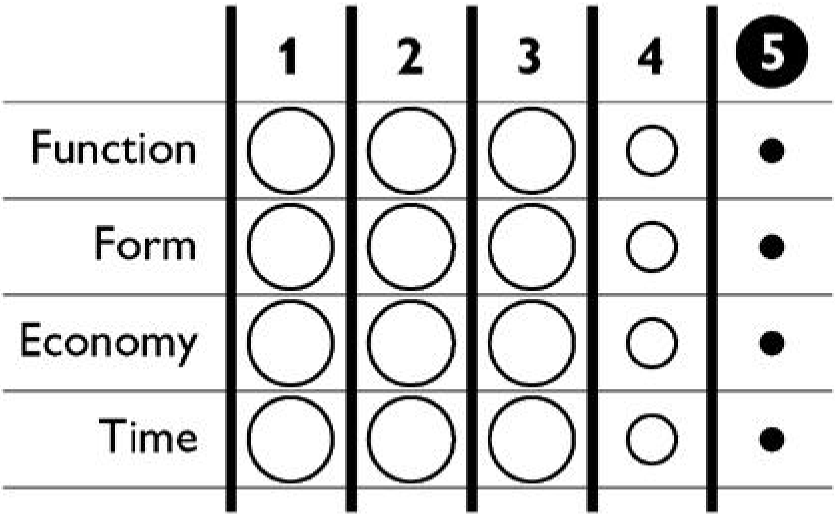


	Goals	Facts
Function People Activities Relationships	Mission Maximum number Individual identity Interaction/privacy Hierarchy of values Prime activities Security Progression Segregation Encounters Transportation/parking Efficiency Priority of relationships	Statistical data Area parameters Personnel forecast User characteristics Community characteristics Organizational structure Value of potential loss Time-motion study Traffic analysis Behavioral patterns Space adequacy Type/intensity Physically challenged guidelines
Form Site Environment Quality	Bias on site elements Environmental response Efficient land use Community relations Community improvements Physical comfort Life safety Social/psychological environment Individuality Wayfinding Projected image Client expectations	Site analysis Soil analysis FAR and GAC Climate analysis Code survey Surroundings Psychological implications Point of reference/entry Cost/SF Building or layout efficiency Equipment costs Area per unit
Economy Initial Budget Operating Costs Life cycle Costs	Extent of funds Cost effectiveness Maximum return Return on investment Minimizing of operating costs Maintenance and operating costs Reduction of life cycle costs Sustainability	Cost parameters Maximum budget Time-use factors Market analysis Energy source costs Activities and climate factors Economic data LEED rating system
Time Past Present Future	Historic preservation Static/dynamic activities Change Growth Occupancy date Availability of funds	Significance Space parameters Activities Projections Durations Escalation factors



	Concepts	Needs	Problem
	Service grouping People grouping Activity grouping Priority Hierarchy Security controls Sequential flow Separated flow Mixed flow Functional relationships Communications	Area requirements By organization By space type By time By location Parking requirements Outdoor space requirements Functional alternatives	Unique and importan performance requir that will shape builc design
	Enhancements Special foundations Density Environmental controls Safety Neighbors Home base/officing concepts On-premise: fixed, free, group address Off-premise: satellite, telecommuting, virtual office Orientation Accessibility Character Quality control	Site development costs Environmental influences on costs Building cost/SF Building overall efficiency factor	Major form considera that will affect build design
	Cost control Efficient allocation Multifunction/versatility Merchandising Energy conservation Cost reduction Recycling	Budget estimate analysis Balance budget Cash flow analysis Energy budget Operating costs Green building rating Life cycle costs	Attitude toward the i budget and its influe the fabric and geom the building
	Adaptability Tolerance Convertibility Expansibility Linear/concurrent scheduling Phasing	Escalation Time schedule Time/cost schedule	Implications of change growth on long-ran performance

Organizing Information



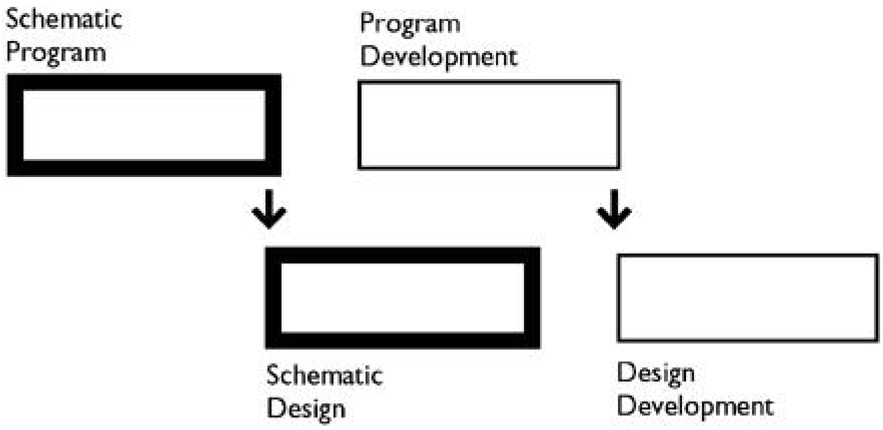
	1	2	3	4	5
Function	◯	◯	◯	○	●
Form	◯	◯	◯	○	●
Economy	◯	◯	◯	○	●
Time	◯	◯	◯	○	●

Programmers establish orders so that information can make sense and can be used effectively in discussions and decision making. **Programmers organize and classify information.** They extract information and display it. They stimulate decisions from client groups. They organize the client's vast world of information with a rational framework. Without this framework, their verification with the client and their hand-off to the designer would not be possible.

With this framework, programmers can classify information, placing it into broad compartments. Since the main search for information is made in the first three steps, it can be expected that the largest quantities of information are found in those first compartments. Refer to the accompanying diagram. Note that the space requirements and their economic feasibility represent a diminished amount of information in the fourth step. And, of course, the fifth step represents the least, yet most important, information.

The handoff package—the programming document including a clear, simple statement of the problem—must represent the epitome of organized, edited information free of irrelevance.

Two-Phase Process



Schematic program and program development provide the information needed at the two successive design phases, going from the general scope to particular details. **Programming is a two-phase process related to the two phases of design**—schematic design and design development.

Schematic design depends on major concepts and needs, which should not be lost in the mass of information unusable in this phase. Designers must have information that clarifies major design determinants—those factors that will shape the broad composition of the building. The schematic program must provide this important overall information useful in schematic design.

However, equally critical is the filtering and the postponing of information that is

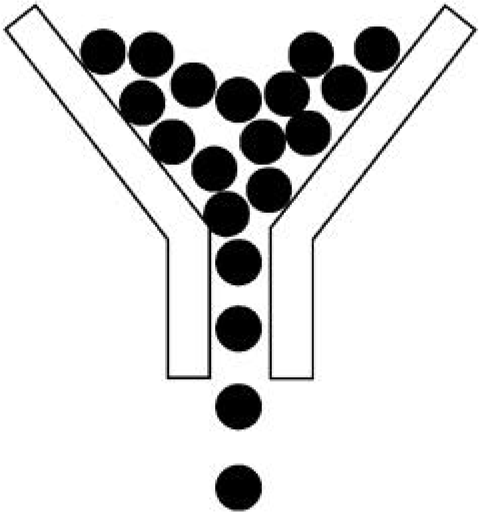
We have to establish the major concepts for a project and the flood of details to follow will not cover up what is real in planning.
—Bill Caudill

not needed in schematic design. Give designers only the information they need at the time they need it.

Design development is what the words imply: the detailed development of schematic design. Program development provides the specific room details—furniture and equipment requirements, environmental criteria (atmospheric, visual, and acoustic), and service requirements (mechanical and electrical). The second phase of programming may be in progress when the designer is doing schematic design.

It should be pointed out, however, that the programmer, in dealing with an unfamiliar building type and critical functional areas, must seek and collect specific details earlier than normally needed in order to establish adequate and generalized space requirements for schematic design.

Data Clog

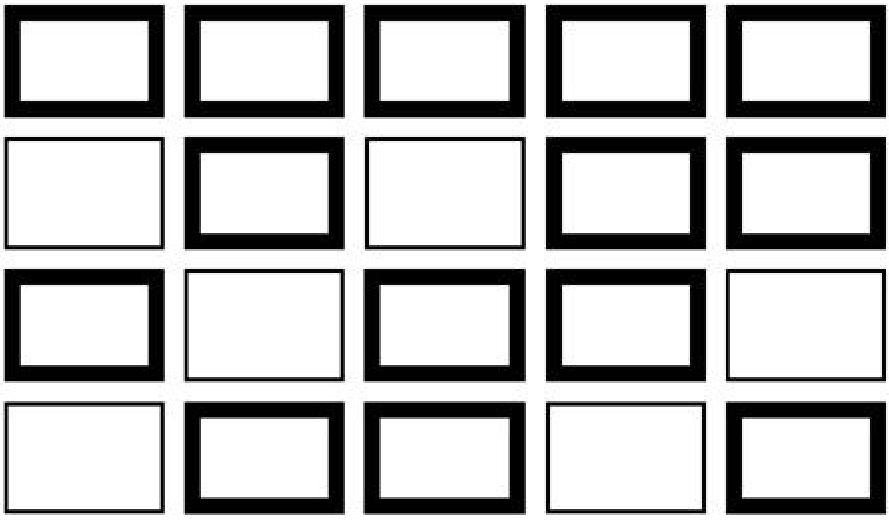


The amount of information received from a client can be staggering. Don’t let the flood of information bother you. One trick is to determine when the information will be most useful—in schematic design or in design development. Any quantity of client-furnished information can be organized for use at the appropriate phase. A programmer needs experience and good judgment to know in which phase to use the information—and needs even more experience and judgment to cull trivial and irrelevant information to eliminate data clog.

Yes, people become data clogged with too much unorganized information, which causes confusion and prevents clear conclusions. Data clog paralyzes the thought processes, and a mental block against all information develops. Unable to comprehend it, designers may decide to ignore it all, throw up their hands and say, “Don’t bother me with all those facts. I know what I must do—I’ll limit the information to what I already know.”

One can assimilate any amount of information as long as it is pertinent, meaningful, and well organized for effective use. Large amounts of highly organized material are required to expand the range of possibilities before a new and useful combination of ideas can be generated by the designer.

Processing and Discarding

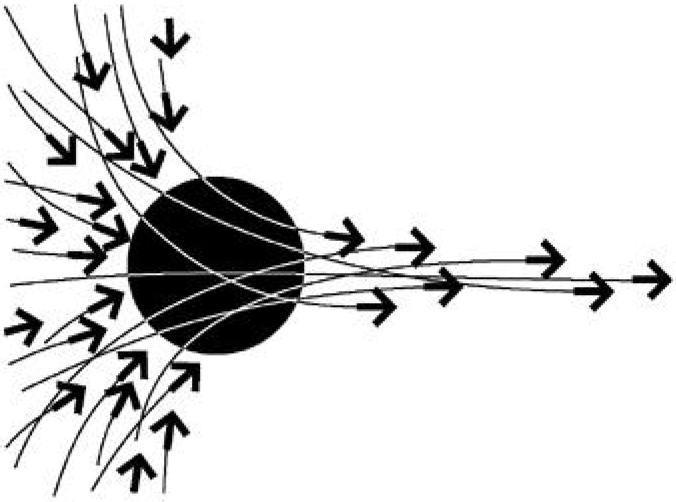


Programming concerns the processing of raw data into useful information. For example, course enrollments at a college are not useful information—until they can be manipulated mathematically with average class size, periods attended per week, total periods available for scheduling, and classroom utilization. Only when the process produces the number and size of classrooms required does the raw data become useful information.

Raw data relating to climate analysis or soil analysis also becomes meaningful information only when architectural implications are determined. After that's accomplished, the raw data can be discarded or placed in an appendix of the program report, where it will not cause data clog.

To quote an old saying, “Any fool can add, it takes a genius to subtract.” It takes a “genius” to discard information as being irrelevant to a design problem or merely too trivial to affect the design one way or another. Although programming is primarily conscious analysis, intuition has its place—**the sensitivity to know what information will be useful and what should be discarded.** The risk in discarding useful information is minimized with experience.

Abstract to the Essence



Architects are taught to take a holistic view of the problem, and even to go beyond the sphere of direct influences to explore other possibilities. However, going too far afield increases the prospects of irrelevant information.

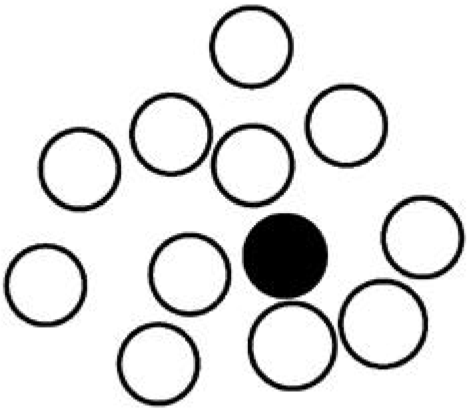
Architects are also taught to bring order out of chaos, to establish an order of importance, to get to the heart of the matter. Abstracting—distilling—to the essence must be an essential talent of the programmer. **There must be a filtering process that brings out only the major aspects of information.** This is especially true in arriving at the statement of the problem.

One reason for limiting one thought, one fact per card, is to be able to reduce the number of cards without losing something important.
—Steve Parshall

There is always the danger of oversimplification in abstracting to find the essence. Yet the danger of leaving something out can be minimized by analyzing and consciously including all the complicating factors.

There is need to amplify in order to view the whole problem, but there is also need to abstract. You amplify and then narrow down; you seek the ramifications of the information gathered, and then turn around to determine the bare implication. It's a continual process. You must be able to see the trees *and* the forest—not both at once but consecutively, from two different points of view.

User on Team



Users are experts in the use of the building. They may assume that they know what they want better than anyone else. They may be right, or they may ask the architect or a consultant to find out what they need. **Users must be contributing members of the project team.**

Dealing with users calls for different strategies to determine reasonable requirements; nevertheless, the building should benefit by intensive user participation in the programming process.

Users are sometimes suspicious that a building will represent only the architect's self-expression. This concerns the familiar argument involving form and function.

On the other hand, the architect is sometimes suspicious that users are being idiosyncratic in their requirements and that no one else can use the building in the future without major remodeling.

Usually, architects love to design buildings tailor-made to specific user requirements that provide opportunities for novel designs. This is particularly true of tailored residences, in which the owner/users are directly responsible for the outcome.

Organizations and institutions with static or dynamic conditions bring up the issue of idiosyncratic versus negotiable requirements. But remember, the users' first concern is how their needs will be met when the building is occupied.

Ways of Thinking

To achieve effective group action, it is important to understand how people think. Planning a large, complex building project involves many people of many minds. We are beginning to appreciate the multiplicity of ideas that emerges from the total planning team with its multiheaded client and multiheaded architect: the client group and the architect group.

By definition, each of these groups carries a different set of baggage with distinct needs, values, and objectives—indeed, different ways of thinking. This is inevitable. Recognizing the differences is no less important than reconciling them, whether they exist between the client group and the architect group or among individual group members. The greatest differences exist within the architect group. These differences usually emerge in architectural programming, the first phase in the design process. Problem seeking recognizes analysis and synthesis as two different processes calling for two different ways of thinking.

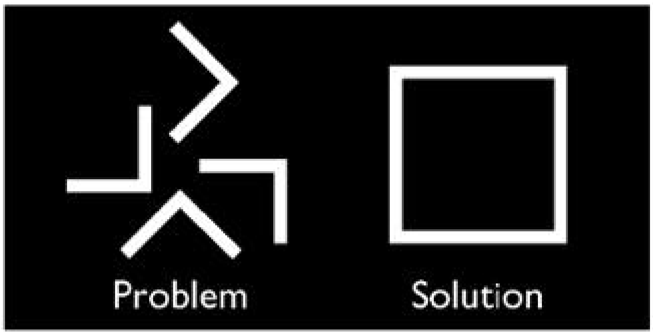
To determine an organization's needs related to a proposed building project, senior management generally assigns a group to work with architectural programmers, the first contingent of the architect group. This group might include people from the top to the bottom of the organization chart. If needed, management brings in outside consultants to augment the know-how of the client group. As to be expected, each participant comes with certain biases and viewpoints that are all valid and important.

Programmers seek consensus among these diverse viewpoints through a series of meetings. The objective is to cope with the multiplicity of thought and to lessen the differences of so many minds. This doesn't mean there must be a poor compromise. But we know this: **participants in group action will argue their heads off unless they believe that "together we can do a better job than we can separately." Without this maxim, we're in trouble.**

First, there is a kickoff meeting of the entire client group with the programmers in which the format and goals of the programming sessions are clearly spelled out. Scheduled meetings of individual organizational components follow, which lead to preliminary conclusions and program requirements. Worksessions with senior management are required to resolve issues and make decisions. Finally, there is a wrap-up meeting with the entire group to review how the conclusions affect individual needs and desires. That's when minds can clash and communication bogs down.

Team action is not easy. There are always risks. But risks are minimized when group participants understand and appreciate the different ways people think during the search for consensus. Interaction between the client group and the architect group pays off in more functional, beautiful, and economical buildings. We think such results make the risks worthwhile.

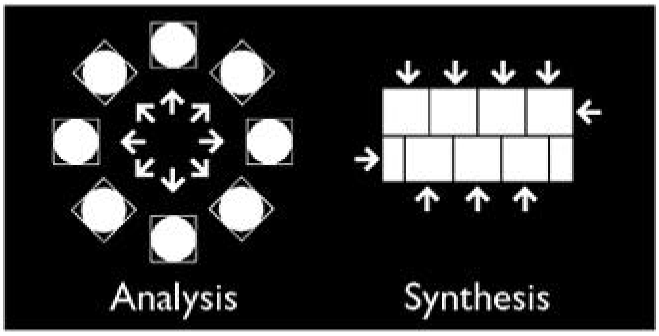
On the next pages are 12 antinomies—different ways of thinking—that are prevalent among the client-architect team during the programming process.



Some people are solution-oriented. This kind of thinking seeks a solution before distinguishing the parts of the problem. Transplanting a California building to New York—or vice versa—is an example of this approach toward design thinking.

This kind of thinking led to dominance of the International Style of building: the same style, the steel -and-glass building, in vastly different geographic locations. This kind of thinking also explains the Texas Cape Cod. In these cases, solutions were identified before the problems were solved.

We contend that problem solving is a valid approach to design; therefore, problem definition should be the first step in the design process. Architectural design is like most everything else: you can't solve a problem unless you know what it is.

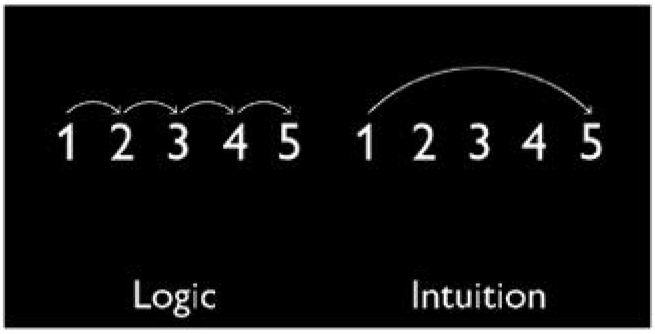


Analytical thinking is said to be based on the left side of the brain, along with logical and verbal functions. The right side handles the ability to synthesize, along with intuitive and spatial capabilities. This is why programmers and designers predominantly use one part of the brain more than the other.

If we accept this notion, we can cope with the multiplicity of thought between programmers and designers. If we practice group action, we can put the many ways of thinking to work for us.

Analysis is what the explicit process of programming is all about. Yet some solution-oriented and intuitive people tend to resist analysis where the parts are separated and clearly identified.

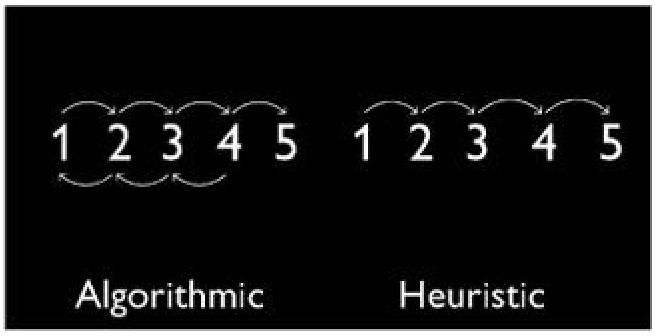
Successful programming relies on analysis. Successful design relies on synthesis. The possibility for creativity depends on the unexpected, integrated arrangement of the parts.



Logical thinkers do well in programming. They use an orderly, well-documented, step-by-step process.

Intuitive thinkers do well when chunks of information are missing. They are scanners. The systematic approach bores them. They skip steps in the process to reach valuable insights. Their weakness is not seeing the necessity of documentation for others. They make poor programmers, but they often make good designers.

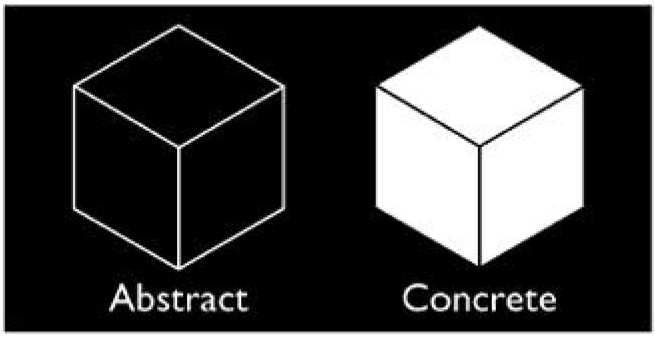
Programming requires logic in its systematic search for information. Designers find that intuition is important in deciding which information will prove most useful. **Since the design process encompasses programming and design, both logical and intuitive thinkers are needed on the planning team.**



The quantitative aspect of information gathering in programming makes some people expect too much exactness. On the other hand, the qualitative aspect provides an evocative ambiguity needed for creativity. Although the intent of programming is to reveal the problem, there is no assurance of precision. That's not all bad. Precision may deter creativity during design.

Programming is heuristic: steps are not rigorously sequential, and information is hardly ever precise or complete.

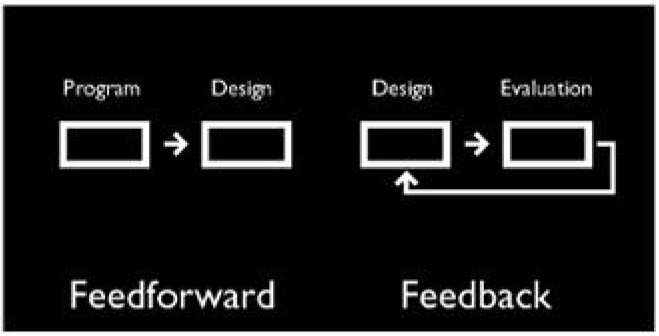
When the problem is crucial, such as life safety, an algorithmic approach is taken. Each step is rigorously retraced in its proper sequence and rechecked for the precision of the information. Exactitude is not necessary for the creation of design concepts. Designers don't paint by the numbers.



Architects and engineers think in three dimensions. They perceive ideas in concrete, tangible terms. Abstract thinking, dealing with ideas generalized from particular instances, is very difficult for some of them, particularly if they are trained to visualize solutions.

Programming needs abstract thinking—keeping parts malleable, jellylike, and loose until design synthesizes the physical solution.

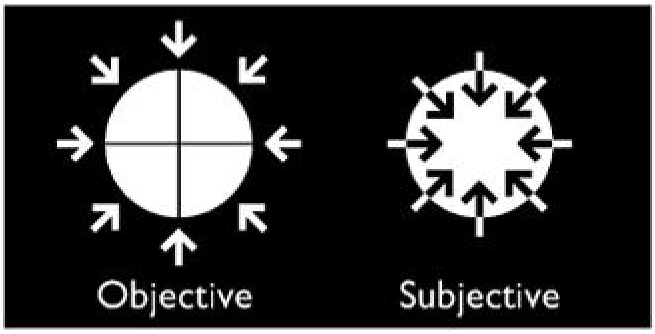
Abstract ideas help to suspend judgment and prevent preconceptions until all the information is gathered and processed. This ambiguity provides that leeway necessary for alternative design solutions. Many design concepts can be derived from one programmatic concept.



Programming implies looking ahead, or feedforward. Programming is the prelude to design, but it does not guarantee good design. Postoccupancy evaluation is feedback to modify a design or to improve a subsequent program. Unquestionably, feedback is a great device to fine-tune a new design or a future program.

Ideally, we should have both feedforward and feedback. The building program, as information feedforward, forms the basis of design. The evaluation, as information feedback, offers refinement of design.

Architects are taught to think in predictive terms—to visualize the way things will be in the future. They must look ahead and, occasionally, use the rear view mirror. In a medical analogy, if programming is diagnosis, post-occupancy evaluation is postmortem! And we learn from both.

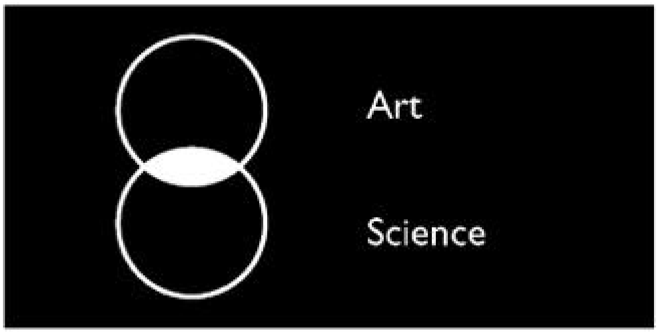


Programming demands objectivity. We know, of course, that complete objectivity is not possible. On the other hand, we need to face facts squarely—to hear what we might not want to hear. Objective thinking relates to the realistic view of facts without distortion, but objectivity does not mean insensitivity to social conditions.

Yet some people approach programming subjectively — as they would design.

Subjectivity deals with personal prejudices brought to the process.

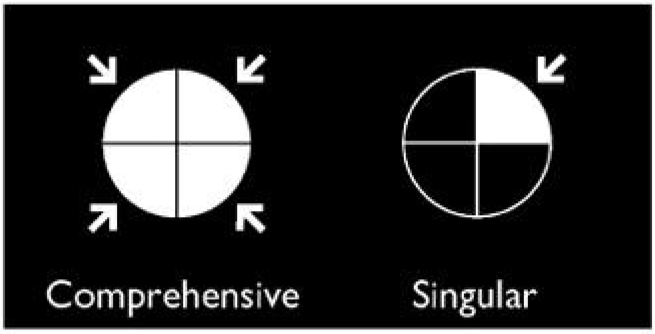
As programmers, when we search for a clear, rational statement of the problem, our minds must think objectively.



These days we hear a lot about the art of architecture as a product of skill and taste applied to certain popular aesthetic principles. We also hear about the science of architecture as a product of knowledge that has been tested and verified.

Artistic activities emphasize intuitive, subjective thinking. Scientific activities emphasize logical, objective thinking. Architecture deals with both.

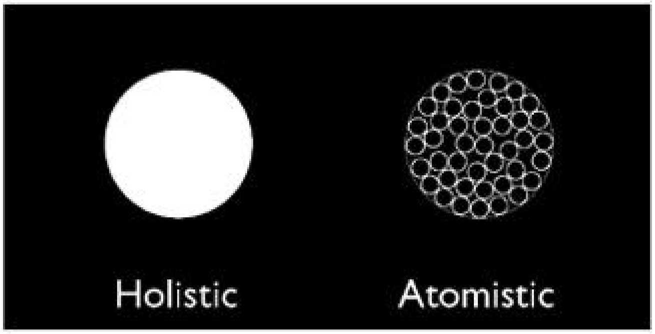
This causes a lot of confusion. The way we cope with this antinomy is to think of architects as practicing on the beach where two worlds meet—the world of arts and the world of science. Architects often walk too far inland and forget how to swim, or swim too far out to sea and forget how to walk. Nevertheless, we love our beach where the arts and science overlap. By nature, architectural design must be open to both worlds.



There are four major design considerations: function, form, economy, and time. All four, not just one, must be included in a rational design process and, in some cases, simultaneously.

But some people work best on a singular approach. They focus on one aspect of design. Some users are single-minded about function, some architects are obsessed with form, and managers emphasize economy and time. Since most people limit their thinking to their specialties, this is the best argument for an inclusive team with a broad range of views.

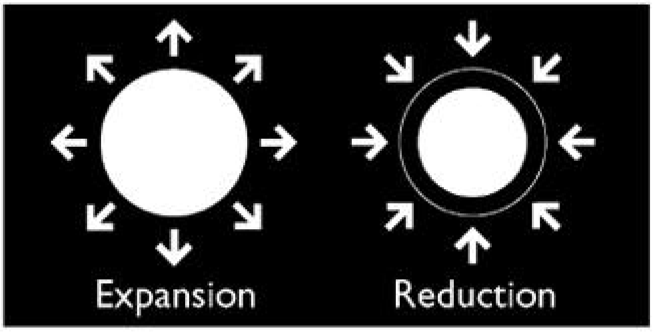
A wide mental grasp is needed to account for all pertinent considerations; however, the individual team member can have a single-track mind devoted solely to his or her specialty. **Unless empathic to other views—to how other specialists think—an individual probably won't make a good member of the team as either a programmer or as a designer.**



Some people are prone to view the design problem in a holistic approach. They see the forest. Others see the trees; they love the details that make up the whole. This is an atomistic approach.

Some are big picture people—conceptual thinkers. Others are detail people who like to work in design development or in interior design. These are opposite ways of thinking. **Programming and design require both ways of thinking.**

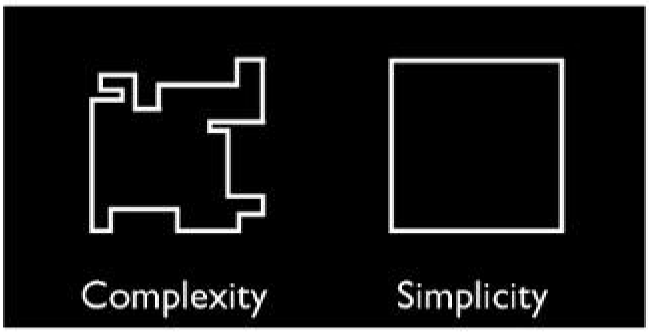
The team is the new genius. We want different eyes—some to see the forest, and others to see the trees. Although it's not absolutely necessary, seeing the forest first has certain advantages.



Programmers and designers alike often expand the design problem beyond the sphere of direct influences. They want to explore other possibilities—to be all-inclusive. This is good. Browning said, "Ah, but a man's reach should exceed his grasp, or what's a heaven for?"

But some people take it beyond the pull of gravity into the universe. It becomes a universal problem that no one can define, much less solve. A Spanish proverb states, "Who grasps too much, squeezes little."

Other people think that to focus on the heart of the matter, one should distill the information reduction down to the essence; however, there is always the danger of oversimplification. **In the search for the problem (programming) and the search for the solution (design), both kinds of thinking have their place. The trick is to decide when one should take precedence over the other.**



Complexity in programming can mean too many tortuous steps, too much detail too soon, too many categories, dubious problems, obscure jargon, multiheaded clients, and unclear terms.

Some people enjoy tension, ambiguity, and complexity. Other people enjoy the intellectual challenge of simplifying it—boiling it down to its essence. We advocate the latter. We generally start with complexity and work toward simplicity throughout the entire design process. Oversimplification occurs through the tendency to concentrate on a single aspect of a problem to the exclusion of all complicating factors. When this happens, the program becomes simplistic and design quality is endangered.

But it is possible to strive for a simplicity that promotes clarity and intelligibility. **Fundamental simplicity is difficult to achieve and requires disciplined analytical skills to discriminate among staggering amounts of information.**

Effective Group Action

Knowing different ways of thinking gives one a better understanding of individuals and how they behave in groups, as well as their distinctive patterns of thinking, perception, and problem solving—specifically, how they intermix in a team endeavor **to develop a building program**. You can sharpen your perception if you see the other side of the coin. You don't have to like what you see, but remember these points when organizing the programming process.

1. **The reconciliation of different ways of thinking cannot be made with a middle-of-the-road mentality.** Consistently riding the median won't do it. There's a time when intuition must dominate logic and a time when it's the other way around; a time for abstract thinking and a time for concrete thinking; and a time to put science over art and a time for the reverse.

2. **People think differently because of background and experience.** That's why an intermixture of distinct individuals is so interesting. Group action often produces unpredictable results. This may not please those who try to program with a prejudice—with a building design in mind. But that's not playing by the rules. For innovative results, let group action set its own course.

3. **Valid information is sometimes elusive.** When a large group of design professionals meet, expect many different points of view, different attitudes, and different opinions that may modify the information itself.

4. **When in group disagreement, keep cool.** Remember there is great value in the interaction of the architect group and the client group. Try to understand the different ways of thinking during the melee. A cool head can tolerate confusion. Remember that we can learn to cope with many different minds and approaches—how to collaborate with people who think differently.

5. **Consensus is difficult, yet it is possible.** The problem -seeking method of programming acknowledges the real needs and desires of users. The end result is to reach agreement on how the proposed buildings should respond to those needs and desires. When there are insurmountable disagreements, then, obviously, management must step in. Delay this as long as you can, and give group action a chance to take hold. When it does, you may be delightfully surprised at the results.

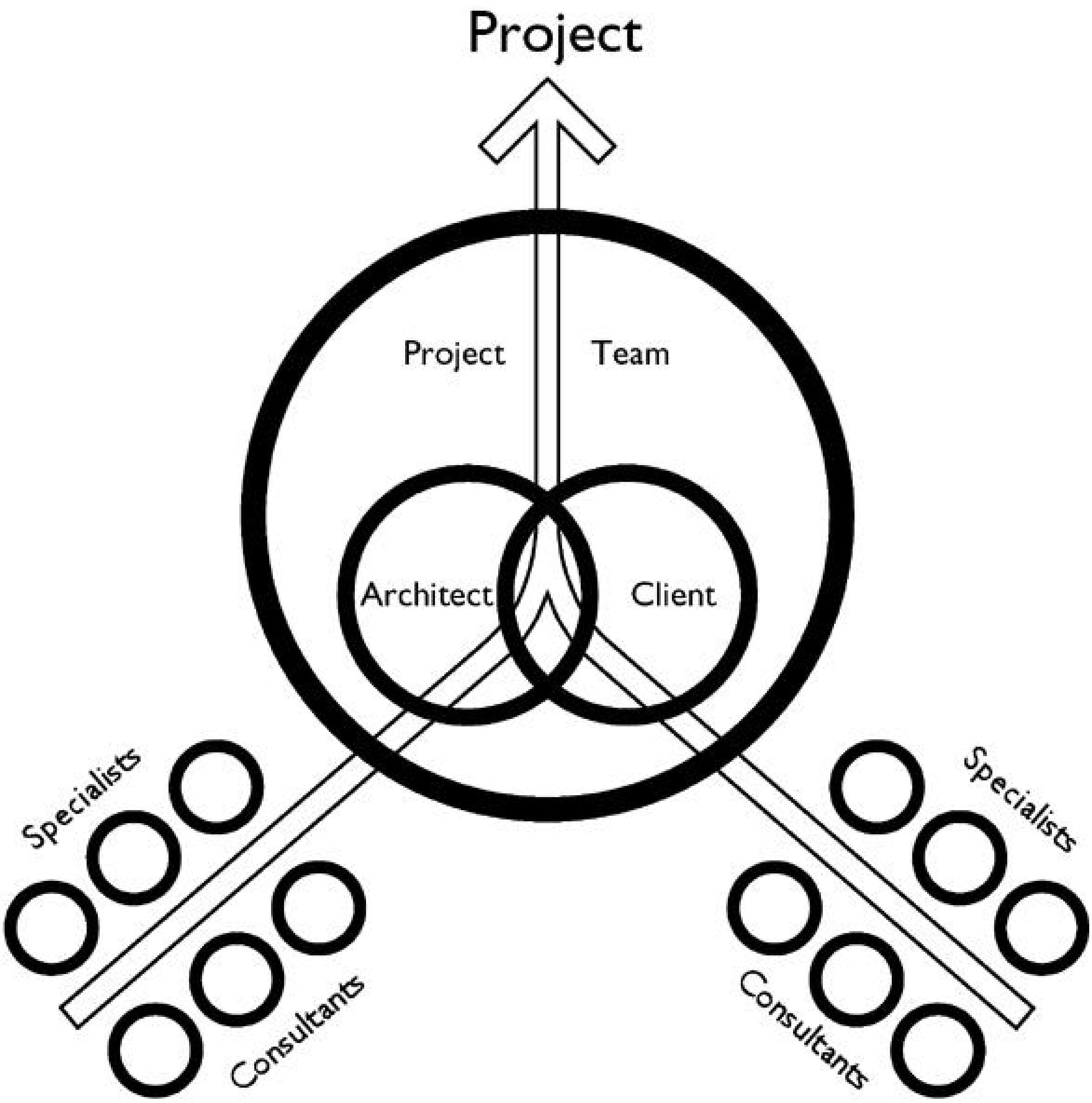
Team

Programming requires team effort. **The project team should be led by two responsible group leaders—** one to represent the client and the other to represent the architect. They must work together toward a successful project. Each leader must be able to:

- Coordinate the individual efforts of his or her group members.
- Make decisions or cause them to be made.
- Establish and maintain communication within, and between, the two groups.

The project team must have good management.

Many people participate in programming a project. There is the traditional participation of the client/owner and the client/manager. More and more, however, the client/users and the client/spectators (community people) are becoming active in programming. This means that the approach to programming should be rational enough to withstand public scrutiny and analytical enough to achieve a mutual understanding of the issues.


Project
Project
Team
Architect
Client
Specialists
Consultants
Specialists
Consultants

Participatory Process

Greater client/user participation generates much more data. This increased involvement also causes more conflicting information. The users are concerned with the hope for greater satisfaction of their needs; the owner is concerned with cost reduction and cost control. Exposure of the owner's and users' differences is the first step in reconciliation. Conflicts are often reconciled by the introduction of human values not previously considered by the owner.

Participants on the team must communicate and be willing to cooperate with one another. This precludes the prima donna client or the prima donna architect who competes to play every role on the team, so as to make every decision in programming and in design.

Clients have the major responsibility to be creative in programming, for they are the ones responsible for the operational outcome. Programmers can act as catalysts in seeking new combinations of ideas. They can test new ideas and spawn alternatives.

Designers must be creative in the design phase, for they are responsible for the physical and psychological environment. Programmers must keep the client from making premature design decisions during programming. They should raise the client's appreciation and aspiration for better buildings. In short, programmers should prepare for designers the best possible environment for creativity.

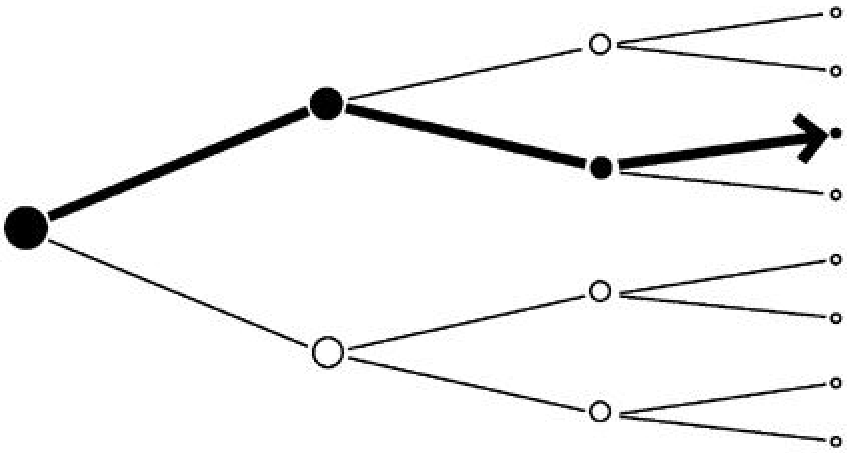
Background Information

Although the five-step process is the same for any building type, **there may necessarily be a preparatory step.** This will depend on the experience (or inexperience) that the programmer brings to the project. For example, if the project were a school and the programmer had no experience in programming schools, then he or she should develop a background understanding of schools. The programmer should visit similar schools, do library research, and talk to educators and consultants. He or she would need to understand the jargon of the client and the general nature of the building type.

Programmers start with an analytical attitude. They approach the project in an organized manner. Their background and experience relate to the specific type of building. If not, the preparatory step is required.

With proper background information, programmers help the client to determine the number and kinds of consultants and when they might be most effectively brought into the total design process.

Decision Making



Good programming is characterized by timely and sound decision making by the clients—not the programmer. During programming, clients decide what they want to accomplish and how they want to do it. Programmers may have to evaluate the gains and risks in order to stimulate a decision. They must identify for clients those decisions that need to be made prior to design.

Although complete objectivity is not required, programmers must emphasize the client's decisions and not their own, and their questions should not be based on a preconceived solution. They may stimulate client decisions by spawning options and by testing programmatic concepts. Programmers may ask, for example, "Have centralized kitchen services been considered as opposed to several decentralized kitchens?" Goals and concepts must be displayed, so that decision makers can understand alternative concepts and evaluate their effect on goals.

You ask, “Do you want two or three bedrooms?” And if the client can’t make up his mind, you arrive at two designs.

“Do you want two or three baths? ” With no decision, you have to arrive at four designs.

“Do you want a two- or three-car garage?” With no decision, you have to arrive at eight designs.

—William Peña

The programmer must stimulate client decisions. This prevents having to reprogram after the designer is at work. When the client’s decisions lead to a well-stated problem, any needed recycling back from design to programming is a minor activity and will not seriously affect the design solution.

When a client postpones decisions, the design solutions tend to be unfeasible. If the client cannot decide on how much money to spend until he or she sees the design, the inevitable will happen. The design solution will exceed the extent of funds available.

Decisions made during programming eliminate the expense of numerous design alternatives. If merely two design alternatives are made for each indecision, the number of alternatives increases exponentially. Indecision, then, increases the complexity of the design problem, which is definitely to be avoided. On the other hand, **every decision the client makes during programming simplifies the design problem** by reducing the number of alternative design solutions to those that meet the program requirements. Organizational and functional decisions produce clear requirements that lead to limited design alternatives, which is highly desirable.

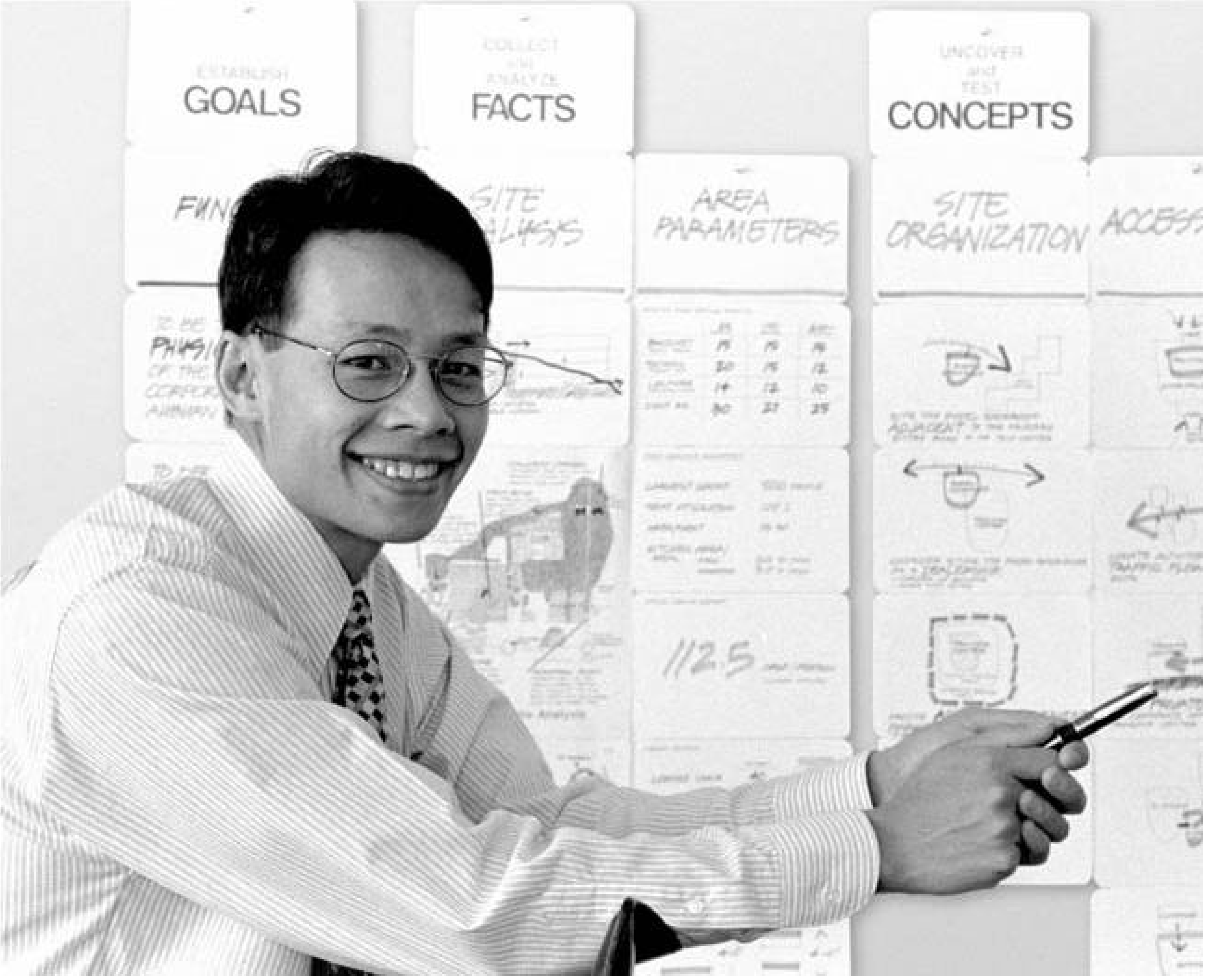
While emphasis is placed on client decision making, it must be realized that this authority is often vested on other people and agencies. Understanding who will actually make which decisions is crucial. The Chief Administrator? The administrators of funding and the code agencies? Generally, the individual who has the responsibility for the outcome has the authority to make the decision. Interview this person! Then insist on his or her approval of the program.

Communication

To achieve effective, clear communication among many people—professionals, clients, and users—information collected must be carefully documented. Undocumented information is not likely to be considered and evaluated by the client and the designer. The programmer collects, organizes, and displays the information for discussion, evaluation, and consensus. **Team effort demands communication**.

Clients and designers require graphic analysis in order to fully comprehend the magnitude of numbers and the implication of ideas. This means there is a need to use communication techniques (brown sheets, analysis cards, and gaming cards) to promote thorough understanding, which leads to sound decision making.

A flow-chart diagram is comprehended more quickly than a written description. Use graphic images that are simple and include only one thought at a time. Keep the images specific enough to clarify the thought, but general and abstract enough to evoke a range of design possibilities. These should help client understanding and cater to the designer's thinking and drawing skills.


ESTABLISH
GOALS
COLLECT and ANALYZE
FACTS
UNCOVER and TEST
CONCEPTS
SITE ANALYSIS
AREA PARAMETERS
SITE ORGANIZATION
112.5

Establish Goals

Goals are important to designers who want to know the what and why of things rather than a list of spaces. They won't find inspiration in a list. They will find it in goals. **Project goals indicate what the client wants to achieve, and why.**

However, goals must be tested for integrity, for usefulness, and for relevance to the architectural design problem. To test them, it is necessary to understand the practical relationship between goals and concepts.

If goals indicate what the client wants to achieve, concepts indicate how the client wants to achieve them. In other words, goals are implemented through concepts.

Goals are the ends. Concepts, the means. Concepts are ways of achieving goals. The relationship of goals and concepts is one of congruence. The test for the integrity of goals depends on their congruence with concepts.

Practical goals have concepts to implement them. Lip-service goals, on the other hand, have no integrity and should be disregarded. They may well be faithless promises in a public relations publication with no plan to keep them. Regardless of good intentions, it is not always what the client says but what he or she really means.

No one can argue against "motherhood" goals. They are unassailable; however, they are too general to be directly useful. Who can argue against the goal "to provide a good environment?" or the goal "to get the most for the money?" There's nothing wrong with including a few "motherhood" goals, especially if they can be processed to be specific enough to clarify the situation; however, intellectually hard, clear project goals are absolutely essential.

On the other hand, a few "motherhood" goals are needed to inspire designers, who like ambiguity to trigger the subconscious in their search for design concepts.

Do not forget that trying to mix problems and solutions of different kinds causes never-ending confusion. To put it positively, a social problem calls for a social solution. After there is a social solution, then it can be part of a design problem for which there will be a design solution. You cannot solve a social problem with an architectural solution.

Programmers must test goals and concepts for relevance to a design problem and not to a social or some other related problem that cannot be solved architecturally. This test for relevance includes testing goals and concepts for design implications that might qualify them as part of a design problem.

Collect and Analyze Facts

Facts are important only if they are appropriate. Facts are used to describe the existing conditions of the site, including the physical, legal, climatic, and aesthetic aspects. These facts about the site should be documented graphically to be really effective. Other important facts include statistical projections, economic data, and descriptions of the user characteristics. There's no end to facts. **Yet programming must be more than fact finding.**

The facts (and figures) can become too numerous to promote definite conclusions. Collect only those that might have a bearing on the problem, and organize them into categories. Seek facts that are pertinent to the goals and concepts. Massage these facts and figures so that become useful information. Process them to determine the architectural implications.

Facts may involve many numbers—such as the number of people that generates space requirements: 2000 seats in a concert hall. Numbers need to be accurate enough to ensure the impartial allocation of space and money, yet rounded out enough to allow for a loose fit: 150 square feet per office occupant. Predictive parameters have to be just accurate enough to be realistic: 15 square feet per dining seat.

When programmers ask questions, what they hear may not be what they want to hear; nevertheless, they must try to avoid a bias so as to collect impartial information. They must avoid preconceptions and face the facts squarely. They must be realistic, neither optimistic nor pessimistic. Programmers must separate fact from fantasy. They must seek what is true or even what is assumed to be true. Assumptions in this case are things to be lived with. Programmers must tell the difference between established fact and mere opinion. They must evaluate opinions and test their validity.

Uncover and Test Concepts

It is critical to **understand the difference between programmatic concepts and design concepts**, which is very difficult for some people to grasp.

Programmatic concepts refer to abstract ideas intended mainly as functional solutions to clients' performance problems without regard to the physical response. On the other hand, design concepts refer to concrete ideas intended as physical solutions to clients' architectural problems, this being the physical response. The key to comprehension is that programmatic concepts relate to performance problems and design concepts relate to architectural problems.

The difference between programmatic concepts and design concepts is illustrated in these examples: *convertibility* is a programmatic concept; a corresponding design concept is a *folding door. Shelter* is a programmatic concept; a corresponding design concept is a *roof*.

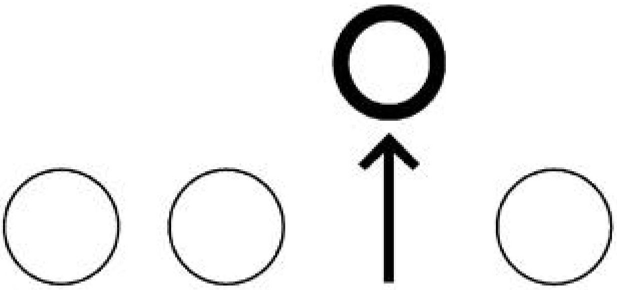
Abstract ideas are required. Ideas must be kept in a pliable, vague form until the designer jells them into a physical solution. It's really best if design can wait until all the information is available. Should the client prescribe independent, concrete ideas or three-dimensional design concepts, the designer would have difficulty in articulating solid -form solutions into an integrated whole.

Such is the case when a house client drops on your desk a big scrapbook full of magazine clippings representing a parade of actual design solutions—a Dutch kitchen, a French Provincial dining room, a Japanese living room, together with a Shangri-La porch. The scrapbook is the nemesis of the experienced programmer, yet it can be used as a means to seek the problems behind the solutions.

There are twenty-four programmatic concepts that seem to crop up on nearly every project, regardless of the building types—housing, hospitals, schools, shopping centers, or factories. The next series of diagrams explains briefly these recurring concepts. The programmer will find them useful by testing to see if they are applicable to his or her current project.

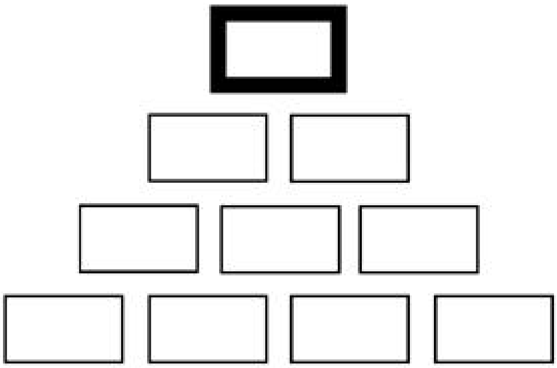
1. Priority

The concept of priority evokes questions regarding **the order of importance**, such as relative position, size, and social value. This concept reflects how to accomplish a goal based on a **ranking of values**. For example, “To place a higher value on pedestrian traffic than on vehicular traffic” may relate to the precedence in traffic flow.



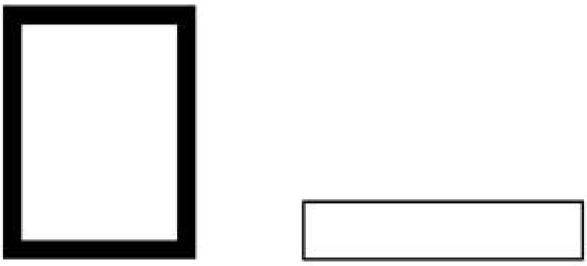
2. Hierarchy

The concept of hierarchy is related to a goal about the exercise of authority and is expressed in symbols of authority. For example, the goal “to maintain the traditional hierarchy of military rank” may be implemented by the concept of a hierarchy of office sizes.



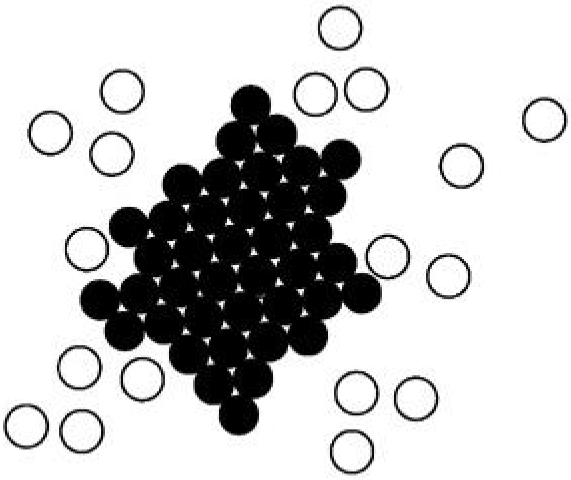
3. Character

The concept of character is based on a goal concerning **the image** the client wants to project in terms of values and the generic nature of the project.



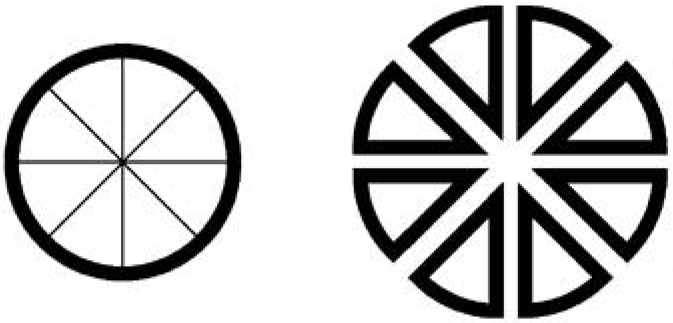
4. Density

A goal for efficient land or space use, a goal for high degrees of interaction, or a goal to respond to harsh climatic conditions may lead to the appropriate degree of density—**low, medium,** or **high density.**



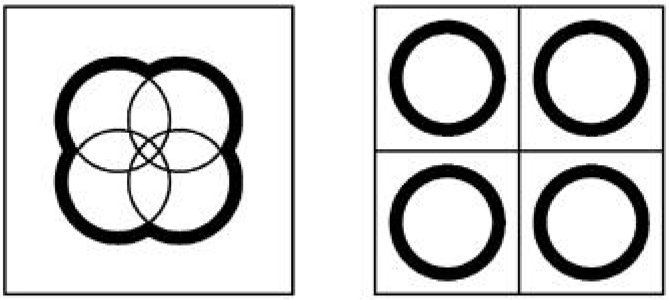
5. Service Grouping

Should services be **centralized** of **decentralized?** Test the many services as being best centralized or best decentralized. Should the heating system be centralized or decentralized? The library? And dining? And storage? And many other services? Evaluate the gains and risks to simulate client decisions. But remember each distinct service will be centralized or decentralized for a definite reason—to implement a specific goal.



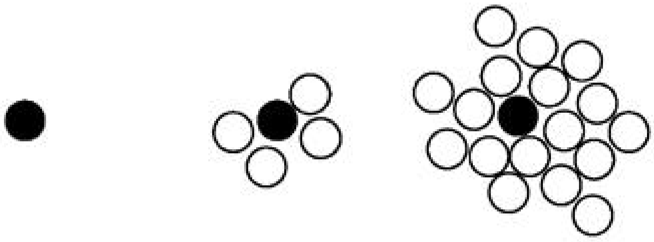
6. Activity Grouping

Should activities be **integrated** or **compartmentalized?** A family of closely related activities would indicate integration to promote interaction, while the need for some kinds and degrees of privacy or security would indicate compartmentalization.



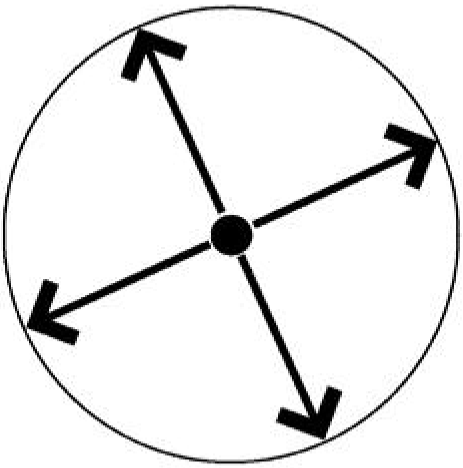
7. People Grouping

Look for concepts derived from the physical, social, and emotional characteristics of people—**as individuals, in small groups,** and **in large groups**. If a client wants to preserve the identity of individuals while in a large mass of people, ask what size grouping would implement this goal. Look to the functional organization and not to the organizational chart, which merely indicates pecking order.



8. Home Base

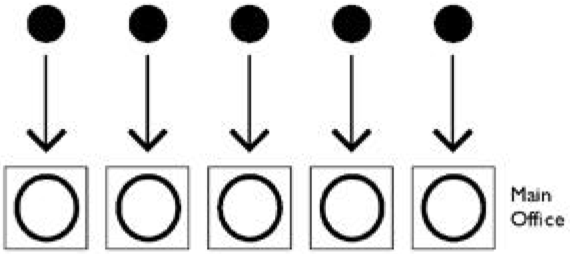
Home base is related to the idea of **territoriality—**an easily defined place where a person can maintain his or her individuality. While this concept applies to a wide range of functional settings—for example, a high school or manufacturing plant—recently, many organizations have recommended new settings for office work. These officing concepts are described in the following pages as on-premise or off-premise work settings.



On-Premise Officing Concepts

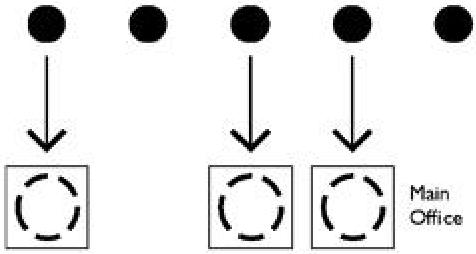
8a. Fixed Address

This concept refers to a traditional work setting where one person is assigned to a workspace. The concept of a shared address is similar, for example, a single office assigned to two or more people—double occupancy.



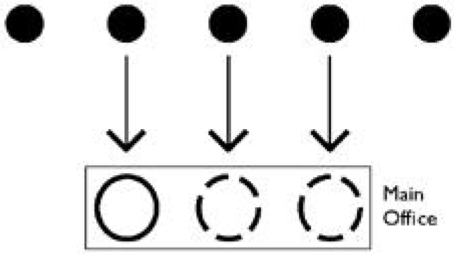
8b. Free Address

This concept refers to workspaces that are unassigned and shared on a first-come, first-serve basis. **Hoteling** refers to the reservation of shared workspaces on a predetermined schedule.



8c. Group Address

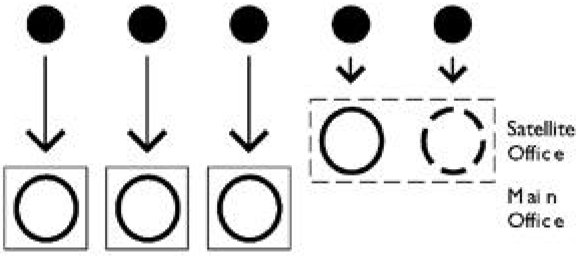
This concept refers to a designated group or team space assigned for a specified period of time. Within the team area, individuals are assigned workspace on an as-needed (free-address) or first-come, first-serve basis.



Off-premise Officing Concepts

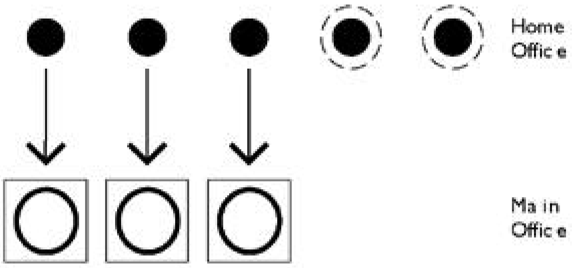
8d. Satellite Office

A goal for providing convenient office centers leads to the concept of satellite offices or remote telecenters. These places provide offices close to employees' residences or customer sites and are used on a full-time or drop-in basis.



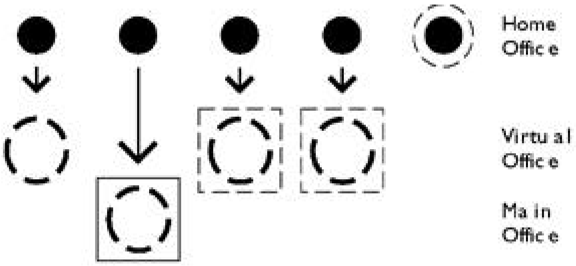
8e. Telecommuting

These concepts refer to an individual's use of his or her residence as a workspace. Electronic communication and computer technology combine to serve as a substitute for travel to an office center.



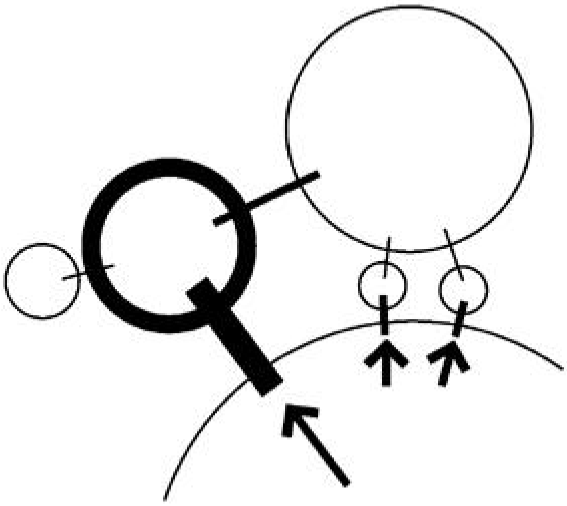
8f. Virtual Office

Virtual officing uses portable computer and communication technology to allow an individual to work in a variety of settings: at home, while traveling, at a client location, in a hotel, or in a satellite office center.



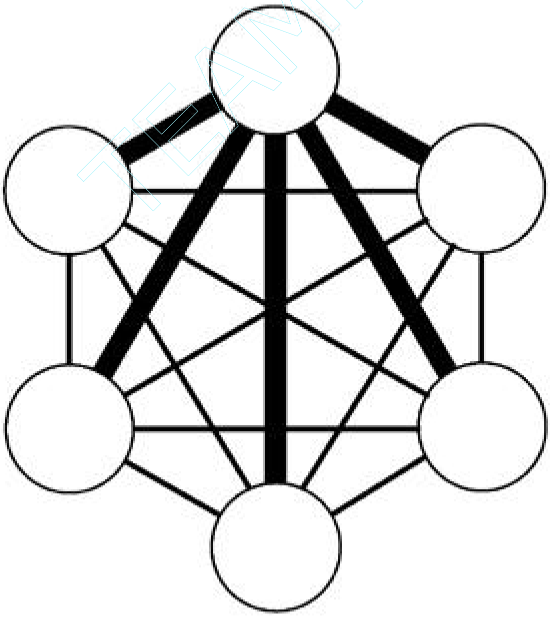
9. Relationships

The correct interrelation of spaces promotes efficiencies and effectiveness of people and their activities. This concept of **functional affinities** is the most common programmatic concept.



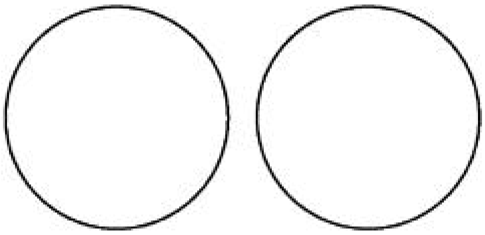
10. Communications

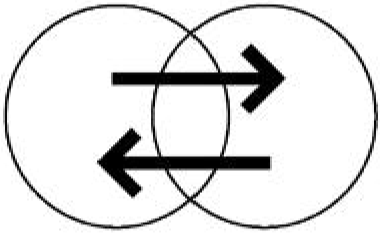
A goal to promote the effective exchange of information or ideas in an organization may call for **networks or patterns** of communication: Who communicates with whom? How? How often?



11. Neighbors

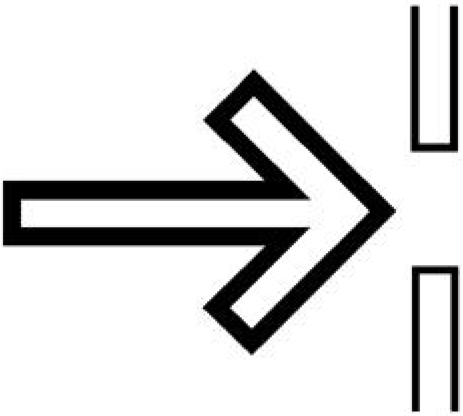
Is there a goal for sociability? Will the project be completely **independent** or is there a mutual desire to be **interdependent**, to cooperate with neighbors?





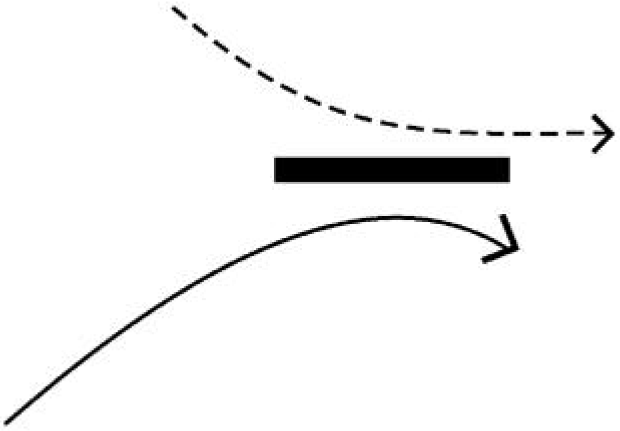
12. Accessibility

Can first-time visitors find where to enter the project? The concept of accessibility also applies to provisions for the handicapped beyond **signs and symbols**. Do we need single or multiple entrances?



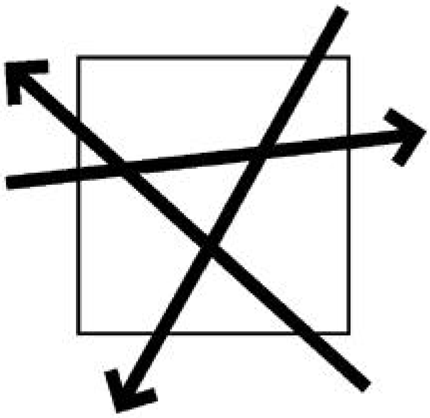
13. Separated Flow

A goal for segregation may relate to people (such as prisoners and public), to automobiles (such as campus traffic and urban traffic), and to people and automobiles (such as pedestrian traffic and automobile traffic). For example, **separate traffic lanes with barriers,** such as walls, separate floors, and space.



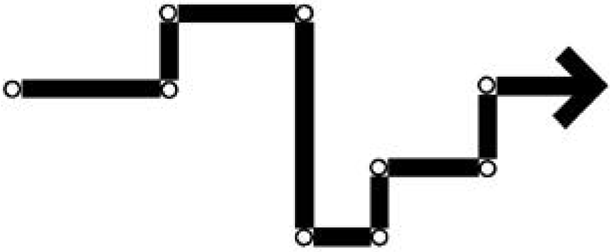
14. Mixed Flow

Common social spaces, such as town squares or building lobbies, are designed for **multidirectional, multi-purpose** traffic—or mixed flow. This concept may be apropos if the goal is to promote chance and planned encounters.



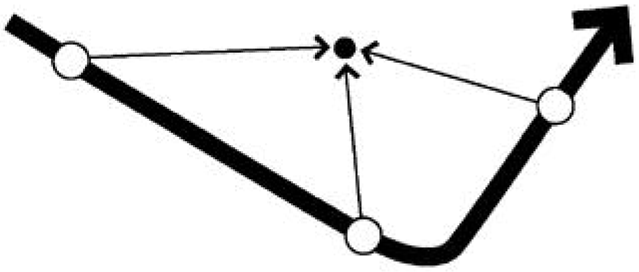
15. Sequential Flow

The progression of people (as in a museum) **and things** (as in a factory) must be carefully planned. A flow-chart diagram will communicate this concept of sequential flow much easier than words will.



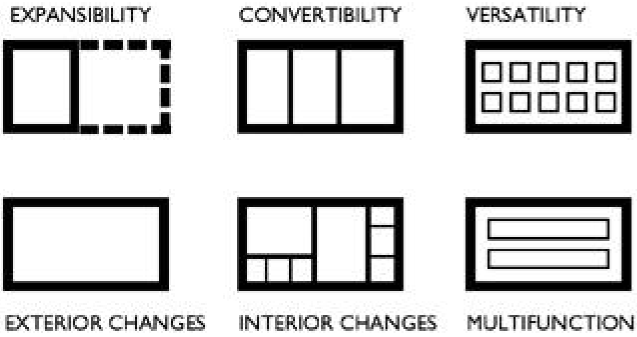
16. Orientation

Provide a bearing—a **point of reference** within a building, a campus, or a city. Relating periodically to a space, thing, or structure can prevent a feeling of being lost.



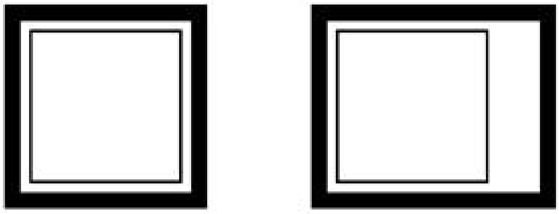
17. Flexibility

The concept of flexibility is quite often misunderstood. To some, it means that the building can accommodate growth through expansion. To others, it means that the building can allow for changes in function through the conversion of spaces. To still others, it means that the building provides the most for the money through multi-function spaces. Actually, flexibility covers all three— **expansibility, convertibility, and versatility**.



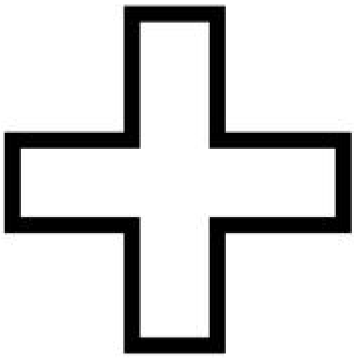
18. Tolerance

This concept may well add space to the program. Is a particular space **tailored** precisely for a static activity, or is it provided with a **loose fit** for a dynamic activity—one likely to change?



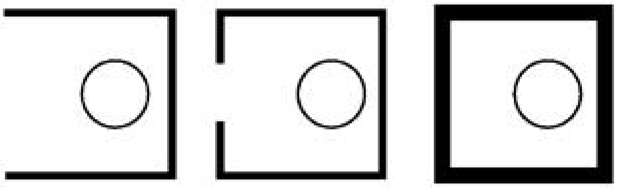
19. Safety

Which major ideas will implement the goal for life safety? Look to **codes and safety precautions** for form-giving ideas.



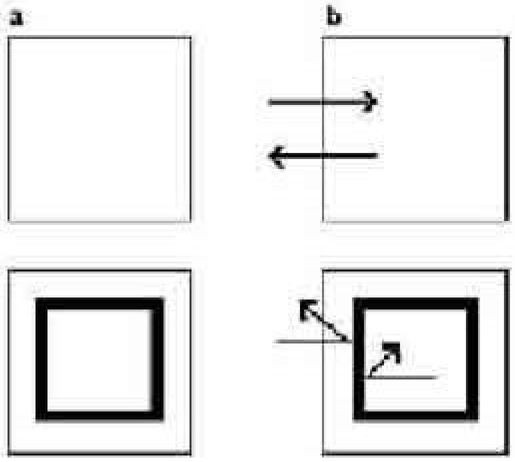
20. Security Controls

The degree of security control varies depending upon the value of the potential loss—**minimum, medium,** or **maximum**. These controls are used to protect property and to guide personnel movement.



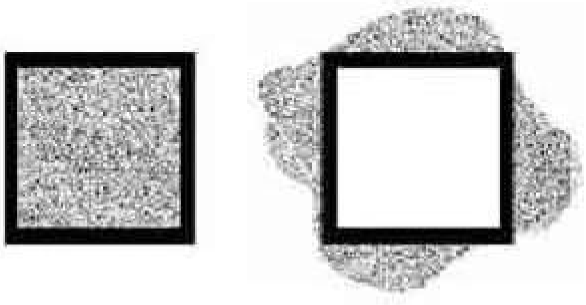
21. Energy Conservation

There are two general ways to lead to energy -efficient buildings: **(a) keep heated area to a minimum** by making use of conditioned, but non-heated, outside space, such as exterior corridors; and **(b) keep heat-flow to a minimum** with insulation, correct orientation to sun and wind, compactness, sun controls, wind controls, and reflective surfaces.



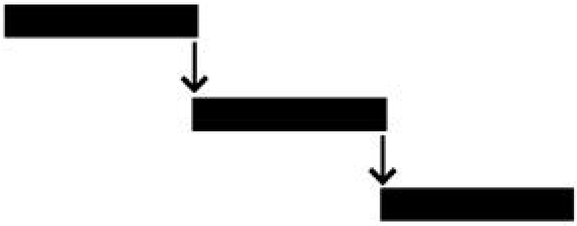
22. Environmental Controls

What controls for **air temperature, light, and sound** will be required to provide for people comfort **inside and outside** the building? Look to the climate and sun angle analysis for answers.



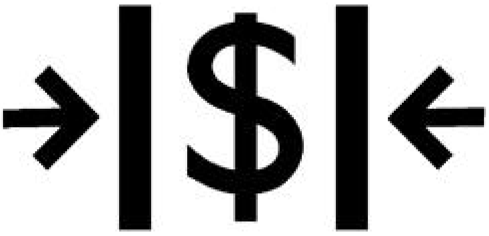
23. Phasing

Will phasing of construction be required to complete the project on a **time-and-cost schedule** if the project proved infeasible in the initial analysis? Will the urgency for the occupancy date determine the need for concurrent scheduling or allow for linear scheduling?



24. Cost Control

This concept is intended as a search for economy ideas that will lead to a **realistic preview of costs** and a balanced budget to meet the extent of available funds.



Determine Needs

Few clients have enough money to do all the things they want to do. Therefore, distinguishing *needs* from *wants* is important. What the rich man considers a necessity, the poor man thinks a luxury. Thus, judgments on the quality and adequacy of space are difficult to make. It is also difficult to identify real *needs*. The client usually wants more than he or she can afford. So the client and the architect must agree on a quality level of construction and on a definite space program relating to funds available at a specific time.

The fourth step is, in effect, an economic feasibility test to see if a budget can be determined, or a fixed budget balanced. It should be noted that the best balance is achieved when all four elements of cost are to some extent negotiable: (1) the space requirements, (2) the quality of construction, (3) the money budget, and (4) time. At least one of these four elements must be negotiable. Thus, if agreement is reached on quality, budget, and time, the adjustment must be made in the amount of space. A serious imbalance might require the re-evaluation of Goals, Facts, and Concepts.

The client's functional needs have a direct bearing on space requirements, which are generated by people and activities. Allowance must be made for a reasonable building efficiency as expressed by the relationship of net areas to gross areas. The proposed quality of construction is expressed in quantitative terms as 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:

- When the initial budget is limited.
- When the funds are available over a period of time.
- When the functional needs are expected to grow.

Cost control begins with programming, and is basic to the whole architectural design problem to be solved. Cost control does not inhibit an architect's creativity; economy is a major consideration, not a constraint. An architect might petulantly think that cost control is a constraint, but not if he or she is committed to giving clients what they need, what they can afford.

Predicting costs at programming is not too difficult since the total planning proceeds from the general to the specific, from the broad scope to details. During programming, cost estimates can be made by successive approximations from the roughest tally of gross area, testing it with different quality levels of construction, while keeping an eye on building cost and other anticipated expenditures. First-phase programming (for schematic design) requires schematic estimates. Second-phase programming (for design development) requires more detailed estimates. As the project advances in refinement, it is possible to test, to rebalance, and to update the budget estimate.

Cost Estimate Analysis

There are many times at which we can exercise cost control. But if we don't establish and balance the budget toward the end of programming we jeopardized the project.

—William Peña

It is imperative to establish a realistic budget from the very beginning. **Realistic budgets are predictive and comprehensive**. They prevent major surprises. They tend to include all the anticipated expenditures as line items in a cost estimate analysis. The architect must look to past experience and published material to derive predictive parameters.

The budget depends upon three realistic predictions: (1) a reasonable efficiency ratio of net to gross area, (2) cost per square foot escalated to mid-construction, and (3) other expenditures as percentages of building cost. These predictions have become so common a practice that they are not considered as predictions but as planning factors.

What happens when a trial-run cost estimate analysis results in a total budget amount required (Line K) larger than the extent of funds available? In other words, the client cannot afford the total cost. If the budget is fixed for a specific time, only two other factors can change: cost per square foot and gross area. This means that the quality of construction or the amount of space or both must be reduced.

Cost Estimate Analysis

A. Building Costs	200,000 S.F.at $90.00/GSF	$18,000,000
B. Fixed Equipment	(8% of A)	1,440,000
C. Site Development	(15% of A)	2,700,000
D. Total Construction	(A + B + C)	$22,140,000
E. Site Acquisition/Demolition		500,000
F. Moveable Equipment	(8% of A)	1,440,000
G. Professional Fees	(6% of D)	1,328,400
H. Contingencies	(10% of D)	2,214,000
J. Administrative Costs	(1% of D)	442,800
K. Total Budget Required	(D + E through J)	$28,065,200

State the Problem

Programming is a process leading to an explicit statement of an architectural problem. It's the handoff package—from programmer to designer.

After pondering information derived from previous steps, *designer and programmer must write down* the most salient statements regarding the problem, the kind of statements that will shape the building. These, if skillfully composed, can serve as **premises for design**, and later as **design criteria** to evaluate the design solution.

There should be a minimum of four statements concerning the four major considerations, components of the whole problem: Function, Form, Economy, and Time. Typically, they cover the functional program, the site, the budget, and the implications of time. Rarely should there be more than ten statements. More than this would indicate that the problem is still too complex or that minor details are being used as premises for design. Statements must represent the essence of the problem.

The problem statements must be clear and concise—in the *designer's own words* so there is no doubt that he or she understands. The problem statements should focus on the obvious—which is often overlooked. Stress the uniqueness of the project.

This is where we do the most reductions. However, there are some people who would expand the problem to make it universal—which no one can solve.

—William Peña

The format for a problem statement can vary with individual designers, but it is good practice to acknowledge a significant and specific condition and establish a general direction for design. While each condition must be precisely stated, the direction (what should be done) should be ambiguous enough to prevent the feeling of being locked into one solution. This direction should be made in terms of performance, so as not to close the door to alternative solutions nor to different expressions in architectural form.

These qualitative statements relate to the whole problem by including all the complicating factors, yet they must represent the essence of the previous steps. They anticipate a comprehensive solution to the whole problem—not by discarding the information in the previous steps (which is kept on display), but by resolving the initial complexity of the design problem into simple and clear statements. The act of resolution pervades the programming process, but it is most vividly expressed in this fifth step. Resolution requires an intensity of intellectual effort. It is hard work to simplify and clarify the statement of the problem, yet this is necessary so that everyone on the project team can cooperate toward the same end.

Programming Principles

To reinforce the concept of Architecture by Team, Bill Caudill believed:

A. The Principle of Product
A product has a much better chance of being successful if, during the design process, the four major considerations (function, form, economy, and time) are regarded simultaneously.

B. The Principle of Process
Every task requires three kinds of thinking action relating to the disciplines of architectural practice: management, design, and building technology. Teamwork is in the overlap.

Expanding on these two principles of team action, the following principles are the foundations of the problem-seeking method.

1. The Principle of Client Involvement
The client is a participating member of the project team and makes most decisions at programming.

2. The Principle of Effective Communication
Clients and designers require graphic analysis to understand the magnitude of numbers and the implications of ideas.

3. The Principle of Comprehensive Analysis
The whole problem covers a wide range of factors that influence design, but they can all be classified in a simple framework of five steps and four considerations.

4. The Principle of Bare Essentials
Programming requires abstracting—distilling—to the essence to bring out only the major aspects of information.

5. The Principle of Abstract Thinking
Programming deals with abstract ideas known as programmatic concepts, which are intended mainly as operational solutions to clients' performance problems, without regard to the physical design response.

6. The Principle of Distinct Separation
The problem-seeking method recognizes programming and design—analysis and synthesis—as two different processes calling for different ways of thinking.

7. The Principle of Efficient Operation
The programming team requires good project management, clear roles and responsibilities, a common language, and standard procedures.

8. The Principle of Qualitative Information
The requirements of a proposed building include the clients' goals (what is to be achieved) and concepts (how it is to be achieved).

9. The Principle of Quantitative Information
Certain project facts and needs are essentially numbers—numbers of people and things generate area numbers and cost numbers—and they can lead to cost control and a balanced budget.

10. The Principle of Definite Closure
Programming is a process leading to an explicit statement of an architectural problem—compensating for the missing parts and resolving the initial complexity to simple and clear statements.

This page intentionally left blank

Part Two

The Appendix

Introduction

Terms and Examples

There is a series of seven sections covering the terms, definitions, and examples of applications used in architectural programming. Each section defines terms used in the text of this book and related terms not in the text. The terms are not in alphabetical order, because it is more important to explain their interrelationships. For example, Values, Beliefs, and Issues are grouped to explain their logical relationship to Goals. To find a definition, then, first look in the Index; it will cite the proper page.

1. On Theory and Process

2. On Considerations

3. On Goals

4. On Facts

5. On Concepts

6. On Needs

7. On Problem Statements

Programming Procedures

The Information Index uses key words and phrases to trigger specific questions in the context of the project at hand. Behind these key words are detailed procedures that are universal enough to be negotiable for a wide variety of building types. These words are meant to evoke questions and further inquiry for the analysis of a design problem. Therefore, their organization follows the five steps of the problem-seeking process.

1. Establish Goals

2. Collect and Analyze Facts

3. Uncover and Test Concepts

4. Determine Needs

5. State the Problem

Programming Activities

The Primer covers the basic programming process. This section explains how to apply the fundamental process to a typical project, and then to more complex projects and different client situations. Finally, it explains three ways to simplify design problems.

1. Typical Programming Activities

2. Four Degrees of Sophistication

3. Variable Conditions

4. Three Ways to Simplify Design Problems

Useful Techniques

The most important techniques in programming deal with methods of communication with users and later with the design team: first, how to collect, organize, and analyze data, and then how to interview the client for information and how to use that information during decision-making worksessions with the client. Graphic -communications techniques help clients and designers understand the magnitude of numbers and the implication of ideas. Ultimately, one should be able to evaluate the programming package—without reference to the resulting design. Is it a good architectural program? Use a question set, and find out. Programming reports are often required for program approval.

1. Questionnaires

2. Data Management

3. Functional Relationship Analysis

4. Interviews and Worksessions

5. Preparation of Brown Sheets

6. The Analysis Card Technique

7. Electronic Presentations

8. Outline for Programming Reports

9. Evaluation Technique

10. Building Evaluation

Terms

The following section contains more than definitions of terms with very special usage in architectural programming. They contain examples to illustrate the terms themselves, and also examples of related terms to show their relationships. They contain information linking the technical terms to the process—linking meaning and usage. The section on Needs is, perhaps, the most complex case of definitions and examples. The components of a cost estimate analysis lead to typical line item allocations, to the variations of building efficiency, and, finally, to the mysteries of quality levels.

On Theory and Process

Architectural Programming: A process leading to the statement of an architectural problem and the requirements to be met in offering a solution.

Systems Analysis: The process of studying an activity typically by mathematical means in order to determine its essential end and how this may most efficiently be attained.

Scientific Method: The principles and procedures used in the systematic pursuit of interdependent, accessible knowledge and involving, as necessary conditions:

(1) The recognition and formulation of a problem

(2) The collection of data through observation and, possibly, experiment

(3) The formulation of hypothesis and

(4) The testing for confirmation of the hypothesis formulated.

Traditional Problem-Solving Steps:*

(1) Definition of the problem

(2) Establishment of objectives

(3) Collection of data

(4) Analysis of the problem

(5) Consideration of solutions

(6) Solution of the problem

* **Compare with the five problem-seeking steps.**

Hypothesis: A proposition, condition, or principle that is assumed, without belief, in order to draw out its logical consequences and by this method to test its accord with facts that are known or may be determined.

Analysis: Separation, or breaking up, of a whole into its fundamental elements or component parts.

Synthesis: Composition, or combination of parts or elements, so as to form a coherent whole.

Research: Critical and exhaustive investigation or experimentation having for its aim the discovery of new facts and their correct interpretation.

Operations Research: The application of scientific and especially mathematical methods to the study and analysis of complex overall problems.

Theory: Principles and generalizations plus their interrelationships that present a clear, rounded, and systematic view of a complex problem or field.

Principle: An empirically derived conclusion about irreducible qualities of a system. The particular abstractions that summarize the phenomena of a given subject field.

Generalization: A general statement, law, principle, or proposition.

Generalize: To derive or induce (a general conception or principle) from particulars.

Induction: Reasoning from a part to a whole, from particulars to generals, from the individual to the universal.

Deduction: Deriving a conclusion by reasoning. Inferring from a general principle.

Reductionism: A procedure or theory that reduces complex data or phenomena to simple terms.

Resolution: The process of reducing to simpler form. The art of analyzing or converting a complex notion into a simpler one or into its elements.

Heuristic: Serving to guide, discover, or reveal. Valuable for stimulating or conducting empirical research but unproved or incapable of proof.

Algorithm: A rule or procedure for solving a mathematical problem that frequently involves the repetition of an operation.

Comprehensive: Covering a matter under consideration completely or nearly completely, accounting for all, or virtually all, pertinent considerations.

Complex: Combining various parts. Needing considerable study, knowledge, or experience for comprehension or operation.

Complicated: May heighten notions of difficulty in understanding.

Organize: To put in readiness for cooperative action. To arrange elements into a whole of interdependent parts.

Unorganized: Not brought into a coherent or well-ordered whole.

Simplism: Oversimplification. The tendency to concentrate on a single aspect (as of a problem) to the exclusion of all complicating factors.

Method: A particular approach to problems of truth or knowledge. A systematic procedure, technique, or mode of inquiry employed by a particular discipline.

Methodology: The approaches employed in the solution of a problem. A branch of logic that analyzes the procedures that should guide inquiry in a particular field. Methods of inquiry, techniques, and procedures used in a particular field.

Reasonable: Carries a much weaker implication of the power to reason in general. Rather, refers to actions, decisions, or choices that are practical, sensible, just, or fair.

Rational: The power to make logical inferences and draw conclusions that enable one to understand the world about oneself and relate such knowledge to the attainment of goals.

Logical: That which is in harmony with sound reasoning and agrees with accepted principles of logic.

Logic: The science of correct reasoning that deals with the criteria of validity in thought and demonstration.

Key Words: Words with a crucial meaning.

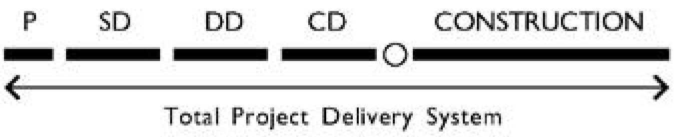
Evocative Words: Words that trigger useful information; words charged with emotion, as well as meaning, that tend to evoke ideas or associations.

Coded Words: Words assigned to arbitrary meanings.

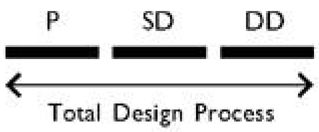
Framework: An open work frame. A frame of reference. A systematic set of relationships.

Information Index: A matrix or rectangular format of key and evocative words arranged to express the relationships of steps and considerations and the typical classification of pertinent information.

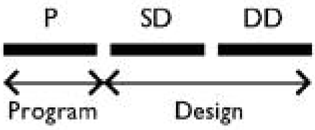
Total Project Delivery System: A complete series of operations leading to the occupancy of a completed building: (1) programming (P), (2) schematic design (SD), (3) design development (DD), (4) construction documents (CD), (5) bidding, and (6) construction.



Total Design Process: The first three phases in architectural practice: (1) programming, (2) schematic design, and (3) design development. Programming is a part of the total design process in this definition, but it is separate from schematic design.



Design: The second and third phases of the total design process: schematic design and design development.



Schematic Design: The interpretation of the owner's project requirements by studies and drawings that illustrate basic architectural concepts, space requirements and relationships, primary circulation, scale, massing, use of site, general appearance, and scope of the project. Included is a statement of adequacy of the stipulated project budget.

Design Development: Following approval of schematic design, development includes the determination, design, and coordination of architectural, structural, mechanical, and electrical systems; equipment layouts; and all related site development. This phase results in drawings and documentation, plus additional material as necessary to illustrate "final" development and insure that all significant design questions and/or problems have been answered.

Construction Documents: This phase transforms the preceding approved "DD Package" into a set of detailed, legal bidding documents that relate to the construction industry. These documents control and direct the construction process via construction drawings and detail materials and building systems specifications.

On Considerations

Considerations: Relate to an architectural product and indicate the four major types of information needed in programming.

Content: Refers to four considerations that constitute a comprehensive architectural problem: Function, Form, Economy, and Time.

Function: How the design product will work to do the job it is supposed to do. The performance.

The "Do"—the way people and things will move about to do the tasks they have to do.

Functions: The action for which a person or thing is specially fitted, used, or responsible, or for which it exists.

Functional: Designed chiefly from the point of view of use: utilitarian work, operations, and/or performance.

Activities: Organized units for performing a specific function.

Form: In design, form means the shape and structure of a building as distinguished from its materials. It is what you see and feel.

In programming, form refers to what you will see and feel, avoiding the suggestion of a design solution. It's the "what is there now" and "what will be there."

Economy: The efficient and sparing use of the means available for the end proposed. Implies an interest in achieving maximum results from the initial budget and the maximum cost effectiveness of the operating and life cycle costs.

Time: Deals with the influence of history, the inevitability of change from the present, and projections into the future.

Operational: Refers to goals and concepts dealing with the process—how the client/architect team will proceed through the total project delivery system to fulfill the contract.

On Goals

Goal: The end toward which effort is directed. Suggests something attained only by prolonged effort.

Goals can be classified as (1) project goals and (2) operational goals.

Project goals are concerned with product; operational goals are concerned with process.

Project goals are established by the client working with the architect. These are elicited from the considerations of Function, Form, Economy, and Time…and their subcategories.

The following can be used as synonyms for the term “goals”: objectives, aims, missions, purposes, reasons, philosophies, aspirations, and policies.

Any of these terms can be used to generate statements that specify what is to be achieved toward the success of the project—what the client wants to accomplish and why.

PROJECT GOALS

1. Function

a. Mission Statement

(1) Explains Reasons

(2) Answers Why

(3) States Purpose

b. Philosophy

2. Form

3. Economy

4. Time

Consider the following use of these commonly used synonyms for the word “goal.”

Objective: A more detailed delineation of a particular goal. Implies something tangible and immediately attainable.

Goals tend to be general; objectives tend to be specific.

Objectives are more time bound and quantitative—and, therefore, a better measure for evaluating the degree of achievement than generalized goals.

Example:

Goal: *To serve as many students from the state of Texas as possible.*

Objective: *To increase enrollment by the amount of 1,000 students per year.*

Policy: A definite course of action selected from among alternatives and in the light of given conditions to guide and determine present and future decisions.

Policies are rules or guidelines that implement goals and objectives. While a goal or an objective stresses the effort of action, a policy represents a selected course of action.

Example:

Goal: *To promote academic efficiency.*

Objective: *To reduce student travel time between classes.*

Policy: *That service courses be decentralized where desirable.*

Concept: *Decentralized cluster of activity.*

Concepts are functional or organizational ideas that also implement goals and objectives. While policies are classified under goals, concepts are not.

Intention: A determination to act in a certain way. Implies little more than what one has in mind to do or bring about.

Aim: Something intended or desired to be attained by one's effort. Implies effort directed toward attaining or accomplishing.

Vision: A target that beckons. A mental image of a desirable future state that is different in an important way from what exists today.

Vision Session: A goal-setting meeting with the client/user groups used to communicate and document a client's vision and goals.

Mission: A task or function assigned or undertaken.

A mission statement of an organization simply explains the reason for its existence.

A functional goal answers the question "Why?"

It should state the purpose of the organization to provide guidance to all subordinate programs and activities.

Example:

This university's mission is to build knowledge and to prepare the future leadership for change and improvement.

The mission statement should include the general functions or services to be performed, without anticipating implementing concepts.

Example:

The functions of a university are: (1) teaching, (2) research, and (3) service.

End: The goal toward which an agent should act. Stresses the intended effect of action, often distinct from the action or means as such.

Philosophy: A basic theory concerning a particular subject, process, or sphere of activity. Asking the client for the philosophy behind the functional program often results in answers and information that are too esoteric and too vague to be directly useful.

Purpose: Something set up as an end to be attained. Suggests a more settled determination.

Aspiration: (1) A goal aspired to or (2) A condition strongly desired. The latter indicates the informality with which a goal can be stated.

Top management responsible for comprehensive planning will necessarily establish the broadest project goals, while middle management will develop more specific goals… consistent with the broad goals. The user usually establishes objectives.

Project goals can be established with no immediate means of achievement available. However, it might be well to remember that goals must eventually be tested to determine their integrity and usefulness—depending on means of achievement.

PROJECT GOALS

1. "Motherhood"
2. Lip Service
3. Inspirational
4. Practical

Consider the following kinds of project goals:

"Motherhood" Goals: These are unassailable goals; however, they are too general to be directly useful.

Example:

To provide a good environment for children.

Lip-Service Goals: These are showpieces that look good in a public relations publication but after testing are found lacking in sufficient backup for accomplishment.

Inspirational Goals: These are general "Motherhood" goals whose ambiguity may serve to trigger the designer's subconscious to uncover a design concept.

Example:

To project the dynamic, progressive spirit of the bank.

Practical Goals: These goals may provide guidance to the collection of pertinent facts.

These goals are intended to be accomplished through known concepts and may affect the statement of the problem.

Example:

Goal: *To help maintain the individual student's sense of identity within the large mass of enrollments.*

Fact: *Enrollments in this school will grow from the initial 1,000 students to 2,700 students.*

Concept: *Decentralize the mass of 2,700 students into schools of 900 students with four houses within each school.*

Goals are derived from values, beliefs, and/or issues, either consciously or unconsciously. In fact, with a client/user who is not goal-oriented or is even nonverbal, it might be easier to bring out values, beliefs, and/or issues that may lead to goals.

Value: Something intrinsically valuable or desirable. Relative worth, utility, or importance. Aims and objectives that act as a basis for motivation. Basic interests or motives.

Example:

Value: *The worth of the individual as a human being.*

Goal: *To help maintain the individual student's sense of identity within the large mass of enrollments.*

Issue: A point of debate or controversy. A matter that is in dispute between parties.

Example:

Issue: *The racial issue.*

Goal: *To develop the performing arts to such an outstanding level that all races will be attracted to this school.*

Belief: Mental acceptance of something offered as true, with or without certainty.

Example:

Belief: *That a better environment can help people live better lives.*

Goal: *To produce forms and spaces with the quality of architecture.*

Operational Goals These goals generally result from the architect's contract or from operational decisions made by the client/architect team. These goals will affect how the team will proceed through the project to fulfill the contract. They will give rise to operational concepts.

Operational goals describe what the team wants to accomplish in terms of the total project delivery system—the process, not the product.

The effort is to identify the best possible course of action in terms of time, people, and cost, and often in terms of information, techniques, and location. It is advisable for the programmer to address the response to these goals when preparing the proposal for services.

OPERATIONAL GOALS

1. Time
2. People
3. Cost
4. Information
5. Techniques
6. Location

Examples:

Time: *To occupy the finished building by September 1998.*

Time and Location: *To keep the present hospital operational while the new wing is being constructed.*

Information and Techniques: *To process enrollment/space data.*

Time and Technique: *To develop a schedule that will compress the total project delivery time.*

Cost: *To effect a 20 percent gross profit on the whole project.*

People: *To coordinate the team ’s activities to make the most effective use of consultants.*

On Facts

Information: Knowledge obtained from investigation, study, or instruction.

Fact: Information presented as having objective reality; truth.

Data: Factual material used as a basis for reasoning, discussion, or decision.

Relevant: Properly applying to the matter at hand. Having a (logical) connection with a matter under consideration.

Pertinent: Interchangeable with "relevant." Often stresses a more significant relationship that contributes to the understanding of a problem or matter at hand.

Assumption: A statement accepted or supposed true without proof or demonstration. In programming, classified under facts as assumed hard facts or fixed opinions.

Truth: Conformity to knowledge, fact, actuality, or logic.

Empirical: Based on factual information. Observation or direct sense experience as opposed to theoretical knowledge.

User Characteristics: Those physical, social, emotional, and intellectual qualities that typify the users and affect their behavior patterns.

Common characteristics include physical size, age and sex, social class, likes and dislikes, intellectual ability.

Parameter: The mathematical term for a symbolic quantity that may be associated with some measurable quantity in the real world, such as cost/GSF.

An arbitrary constant characterizing by each of its particular values some particular member of a system.

Disinterestedness: An objectivity toward uncovering of information. Detached scrutiny.

Objectivity: The use of facts without distortion by personal feelings or prejudices.

Skepticism: Suspension of judgment until all the data is analyzed.

On Concepts

Concept: Something conceived in the mind: an idea or notion.

Programmatic Concepts: These refer to ideas intended mainly as functional and organizational solutions to the client's own performance problems. These concepts are general or abstract ideas generalized from particular instances.

Design Concepts: These refer to ideas intended as physical solutions to the client's architectural problems.

Example:

Programmatic Concept: *Decentralize the mass of 2,700 students into schools of 900 students with four houses within each school.*

Design Concepts: *The physical responses to the programmatic concept of decentralization above may be: (1) the dispersion of three buildings, (2) the dispersion/compactness of three floors in one building, or (3) the compactness of a single building with three identifiable schools on one floor.*

In programming, programmatic concepts are emphasized and design concepts avoided. It is essential to understand the difference between these two kinds of concepts.

To deal with design concepts during programming would mean: (1) jumping to conclusions, (2) synthesizing too early, and (3) determining subsolutions before the subproblems were identified.

PROJECT GOALS (Ends)

PROGRAMMATIC CONCEPTS (Means)

DESIGN CONCEPTS (Response)

Programmatic concepts attempt to implement practical goals. They are a means of accomplishing goals. If goals are ends, programmatic concepts are means, and design concepts are the physical response to them and to the design premises in the statement of the problem.

Programmatic concepts are further classified under Function, Form, Economy, and Time. Since they are intended as functional and organizational solutions, it might be thought that most of them are functional. This is not so. It might also be thought impossible to avoid the physical aspects of concepts. This may be so, but the intent is to state a programmatic concept in such a way as to elicit alternative responses in design.

Recurring Concepts: These refer to programmatic concepts that not only appear in just one project or type of institution, but also appear as potential aspects of any project or institution. These concepts, then, are worth testing in any project to find their applicability.

Operational Concepts: These refer to ideas intended as procedural solutions to the client/architect team's procedural problems. These concepts indicate how the team will proceed through the project to fulfill the client/architect contract. Operational concepts implement operational goals in terms of time, people, and cost, and often in terms of information, techniques, and location.

Examples:

Operational Goal: *To occupy the finished building by September 1988.*

Operational Concept: *Scheduling and critical-path method.*

Operational Goal: *To keep the present hospital operational while the new wing is being constructed.*

Operational Concept: *Concurrent activities.*

Operational Goal: *To process enrollment/space data.*

Operational Concept: *Automation.*

On Needs

Needs: Requirements. Something necessary. An indispensable or essential thing or quality.

Wants: Something lacking and desired or wished for.

Requirement: Something wanted or needed.

Space Requirement: A detailed listing of the amounts of each type of space designated for a specific purpose.

Performance: Something accomplished or carried out. The execution of an action that fulfills agreed-upon requirements.

Performance Requirements: Those requirements stemming from the unique user needs in terms of the physical, social, and psychological environment to be provided. These will involve the adequacy, the quality, and the organization of space.

Functional Requirements: Those requirements dealing chiefly with the way people will use the project with convenience, efficiency, and effectiveness. These also will involve the adequacy, the quality, and the organization of space.

Human Requirements: Those requirements stemming from generalized human needs in terms of the physical, social, and psychological environment to be provided. These human needs involve such general categories as self-preservation, physical comfort, self-image, and social affiliation—initially expressed as specific goals.

Area Definition and Measurement Methods

The definition of building area and its measurements varies with the purpose for how architects, facility managers, or real estate professionals intend to use the definitions and measures. A diligent programmer should review with the client the definition and measures being used on each project. Each standard below responds to a unique industry purpose for its application.

"Classification of Building Areas," Publication Number 1235, National Academy of Sciences, National Research Council, Washington, DC, 1964.

"Methods of Calculating Areas and Volumes of Buildings," AIA Document D101, The American Institute of Architects, New York, NY, 1995.

"Standard Classification for Building Area Measurements," Designation E 1836–96, American Society for Testing and Materials, West Conshohocken, PA, 1996.

"Standard Method for Measuring Floor Area in Office Building," ANSI/BOMA Z65.1-1996, Approved by American National Standards Institute, Inc., Building Owners and Management Association International, New York, NY, 1996.

While there are several industry standards for measuring building area, the purpose of determining the total area for a building program is primarily to predict the size of a new building and to provide a sound basis for estimating the budget for building construction. This size represents the building gross area, which is the sum of the net assignable and unassigned areas.

Unassigned Areas: These consist of all other spaces in the building, specifically circulation areas, mechanical areas, general toilets, janitor closets, unassigned storage, walls, and partitions. The distribution of unassigned areas is shown below as typical percentages of the building gross.

Tare Area: The remainder after the net assignable area is subtracted from the gross building area. The tare area consists of the unassigned areas listed below.

Circulation Areas: These include interior corridors, exterior covered walks (half of full area), and phantom corridors, which are undefined circulation paths through assigned areas, such as a pathway through a programmed lobby space. Note that circulation areas are, by far, the largest single component of unassigned space and it is useful often to distinguish between primary and secondary.

Primary Circulation: Lobbies, corridors, and vertical circulation between elevators, public toilets, building entrances and exits required to satisfy the building code.

Secondary Circulation: Corridors providing access from net assignable areas to the primary circulation.

Mechanical Areas: Areas for the building heating, ventilation, air conditioning, electrical, plumbing, and communications distribution. These areas vary considerably based on the building type. For example, an 8 percent mechanical area for an office building may simply include heating, ventilating, and air-conditioning equipment to meet minimum code requirements. In contrast, a 14 percent mechanical area for a corporate research building may require more sophisticated mechanical systems to meet safety and strict environmental control requirements.

Walls, Partitions, Structure: Building area for structure walls, columns, and dividing partitions. Generally, this amounts to 7 percent to 9 percent of the gross building area.

Public Toilets: Public restrooms required by the building code range from 1.5 percent to 2 percent of the gross building area.

Janitor Closets: Space for general cleaning supplies, normally requires less than 0.5 percent.

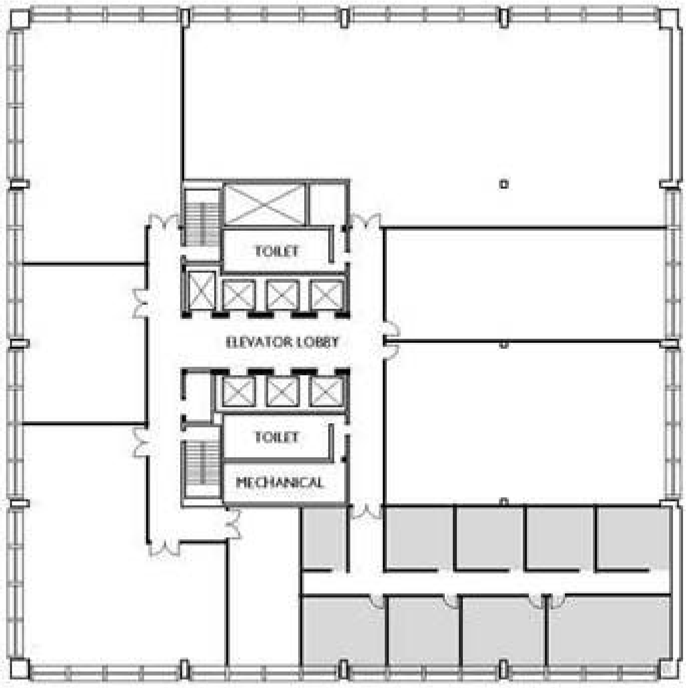
Building Storage: General building storage, normally requires less than 0.5 percent.

Distribution of Unassigned Area as a Ratio of the Building Gross Area

Circulation	.160	.180	.200	.220	.260	.300
Mechanical	.050	.055	.055	.075	.080	.080
Walls, Partitions, Structure	.070	.070	.070	.080	.085	.090
Public Toilets	.015	.015	.015	.015	.015	.020
Janitor Closets	.002	.005	.005	.005	.005	.005
Unassigned Storage	.003	.00	.005	.005	.005	.005
Unassigned Area	**.30**	**.33**	**.35**	**.40**	**.45**	**.50**

Net Assignable Area or Net Area: The area required to accommodate a function, equipment, an occupant, or an occupant group. Net assignable area includes interior walls, building columns, and projections. Net assignable area excludes exterior walls, major vertical penetrations, building core and service areas, primary circulation, and secondary circulation.

Compute the Net Assignable Area: Measure to the inside surface of the exterior building walls, to the finished surface of walls surrounding major vertical penetrations, building core areas, and service areas, and to the center of partitions separating net assignable area adjoining net assignable areas and from secondary circulation space.

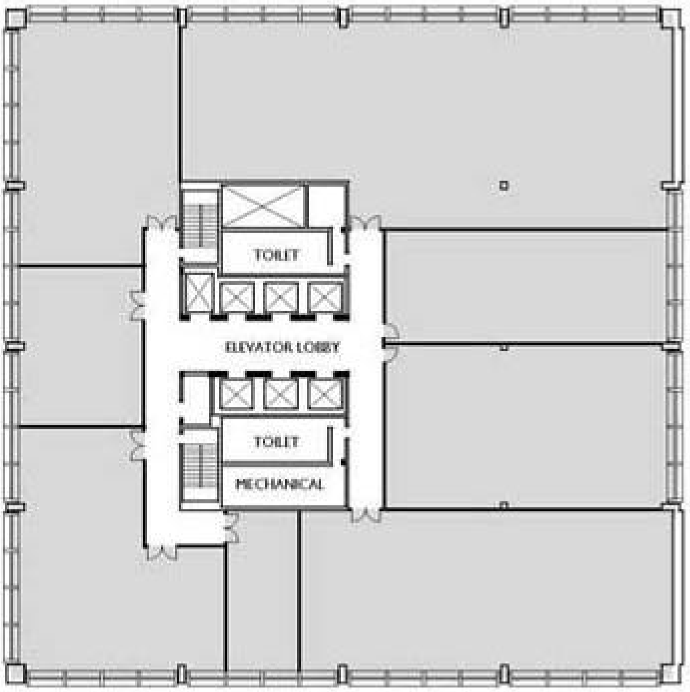


Net Area

Useable Area: The floor area of a building assigned to occupant groups or available for assignment. Usable area includes net assignable areas of interior walls, building columns and projections, and secondary circulation. Usable area excludes exterior walls, major vertical penetrations, primary circulation, building core, and building service areas.

Departmental Gross Area: The net assignable areas and required secondary circulation assigned to an occupant group or department. Same as Useable.

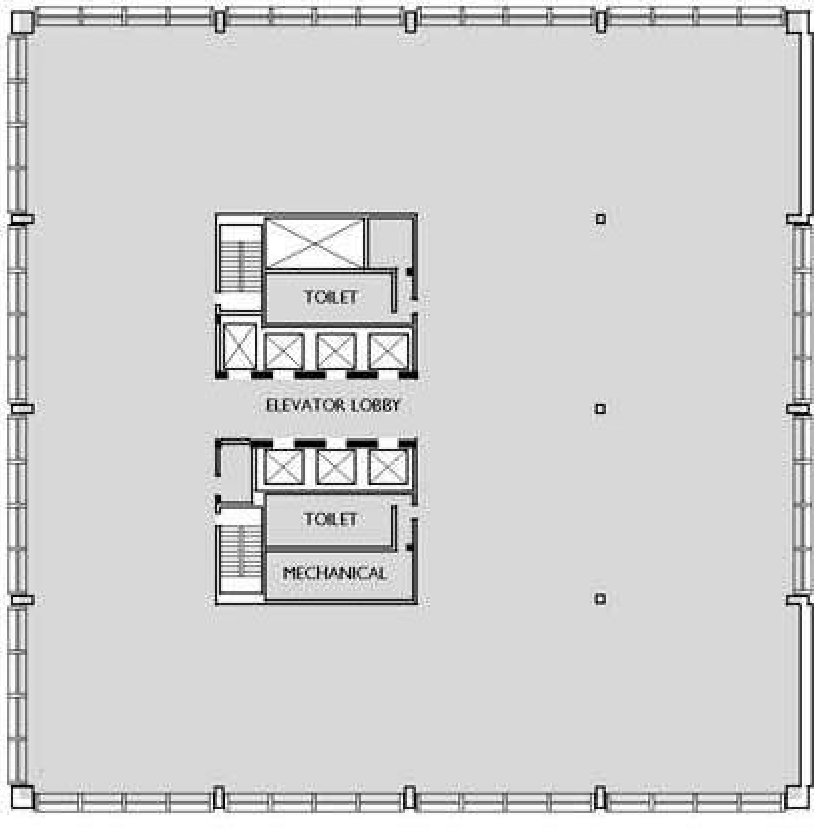
Compute the Usable Area: Measure to the inside finished surface of the exterior building walls, to the finished surface of the walls surrounding major vertical penetrations and building core and service areas, and to the center of the walls dividing the space from adjoining usable areas.



Useable Area

Rentable Area: The floor area of a building that is available for assignment to a tenant as a basis for calculating rent. This area provides a consistent basis of comparison with other buildings whether leased or owner occupied. Rentable area includes the useable area, building core and service areas, and primary circulation. It excludes major vertical floor penetrations, such as elevator shafts and stairs. The definition of rentable area may vary according to the terms of a specific lease.

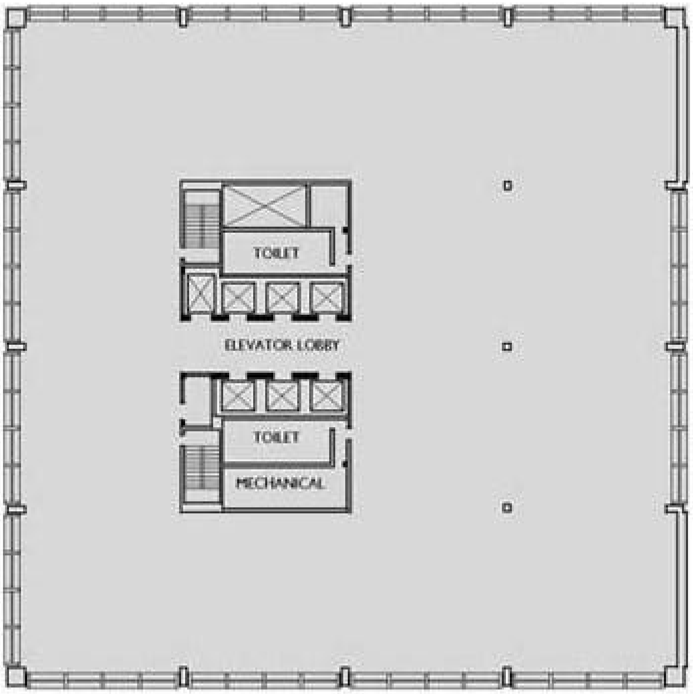
Compute the Rentable Area: Measure to the inside finished surface of the exterior building walls, excluding any major vertical penetrations of the floor. For sloping walls, measure floor areas at the floor plane. Include the areas of columns and building projections in the rentable area. Exclude exterior walls and major vertical penetrations from the rentable area.



Rentable Area

Building Gross Area or Gross Area: The floor area of a building for all levels that are totally enclosed within the building envelope, including basements, mezzanines, or penthouses.

Compute Building Gross Area: Measure to the outside face of exterior walls, disregarding cornices, pilasters, and buttresses, that extend beyond the wall face. The building gross area of basement space includes the area measured to the outside face of basement foundation walls.



Gross Area

Building Efficiency Factors

Building efficiency factors express the relationships among the various area definitions. Variations in application lead to several types of efficiency factors as shown in the accompanying definition and example. There is a mathematical relationship between the efficiency factors as shown below:

Example of Efficiency Factors:

Interior Layout Efficiency	x	Base Building Efficiency	=	Overall Building Efficiency
.61	x	.84	=	**.51**

Efficiency by Building Type	**Interior Layout**	**Base Building**	**Overall Building**
Corporate Office	.620	.80	**.50**
Corporate R&D	.625	.80	**.50**
University Administration	.687	.80	**.55**
University R&D	.750	.80	**.60**
Dormitory	.750	.80	**.60**
Student Center	.750	.80	**.60**
Auditorium	.750	.80	**.60**
Museum	.813	.80	**.65**
Food Service	.813	.80	**.65**
Conference Center	.813	.80	**.65**
Library	.813	.80	**.65**
Academic Classrooms	.813	.80	**.65**
Physical Education	.875	.80	**.70**
Building Services	.938	.80	**.75**
Warehouse	1.000	.90	**.90**

Overall Building Efficiency

Differences in predominating room sizes, occupancy levels, circulation requirements, and special mechanical requirements lead to different overall building efficiency factors for various building types.

For example, the predominance of small rooms requiring higher percentages in circulation and partitions leads to an overall building efficiency of 55 percent in a university administration building. On the other hand, the large gym areas in physical education would indicate small percentages in circulation and partitions, leading to an overall building efficiency of 70 percent. Large spectator areas demanding large areas of circulation would result in factors of 65 percent and 60 percent.

1/Multiplier = Divisor	**1/Divisor = Multiplier**
1 / 1.25 = .80	1 / .80 = 1.25
1 / 1.33 = .75	1 / .75 = 1.33
1 / 1.42 = .70	1 / .70 = 1.42
1 / 1.54 = .65	1 / .65 = 1.54
1 / 1.67 = .60	1 / .60 = 1.67
1 / 1.82 = .55	1 / .55 = 1.82
1 / 2.00 = .50	1 / .50 = 2.0

Clients and architects use efficiency factors either as divisors or multipliers, which are comparable as shown in the table above.

$$\textbf{Overall Building Efficiency} = \frac{\textbf{Net Assignable Area}}{\textbf{Building Gross Area}}$$

Overall Building Efficiency: The ratio of the net assignable areas to the building gross area expressed as a percentage of the gross area. In the programming phase, this factor is used to calculate the total building gross area requirements using the net area requirements as a base. To do this, divide the sum of the net assignable areas by the appropriate overall efficiency. This factor is commonly used for public and educational building design applications.

Example: 60% Overall Efficiency

$$\frac{\text{60,000 net assignable area}}{\text{.60 overall efficiency}} = \text{100,000 building gross area}$$

$$\textbf{Base Building Efficiency} = \frac{\textbf{Useable Area}}{\textbf{Building Gross Area}}$$

Base Building Efficiency: The ratio of the useable areas to the building gross area expressed as a percentage of the gross area. In the programming phase, this factor is used to calculate the total building gross area requirements using the useable area requirements as a base. To do this, divide the useable area by the appropriate building efficiency. This factor is commonly used for commercial building design applications.

Example: 80% Building Efficiency

$$\frac{\text{80,000 useable area}}{\text{.80 building efficiency}} = \text{100,000 building gross area}$$

$$\textbf{Interior Layout Efficiency} = \frac{\textbf{Net Assignable Area}}{\textbf{Useable Area}}$$

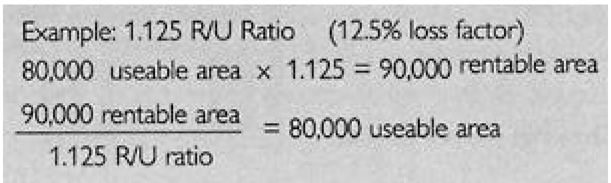
Interior Layout Efficiency: The ratio of the net assignable areas to the useable area expressed as a percentage of the useable area. In the programming phase, this factor is used to calculate the total useable area requirements using the net assignable area requirements as a base. To do this, divide the sum of the net assignable areas by the appropriate layout efficiency. This factor is commonly used for interior design applications.

Example: 75% Layout Efficiency

$$\frac{\text{60,000 net assignable area}}{\text{.75 layout efficiency}} = \text{80,000 useable area}$$

$$\textbf{R/U Ratio} = \frac{\textbf{Rentable Area}}{\textbf{Useable Area}}$$

R/U Ratio: The ratio of the rentable areas to the useable area expressed as a multiplier. Use this R/U ratio, sometimes referred to as the “loss factor,” to calculate the total rentable area requirements using the useable area requirements as a base. To do this, multiply the useable area by the appropriate R/U ratio. This factor is commonly used to calculate rentable area for lease agreements or financial analysis.

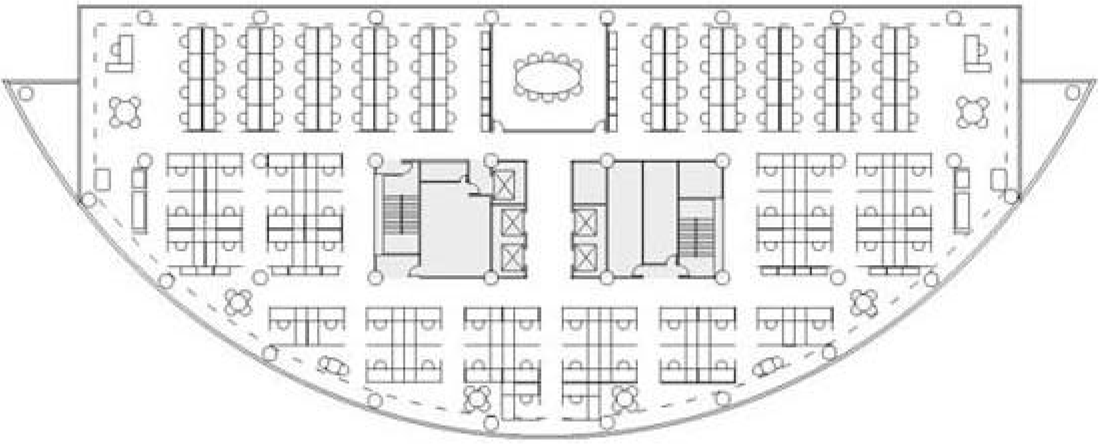


Example: 1.125 R/U Ratio (12.5% loss factor)

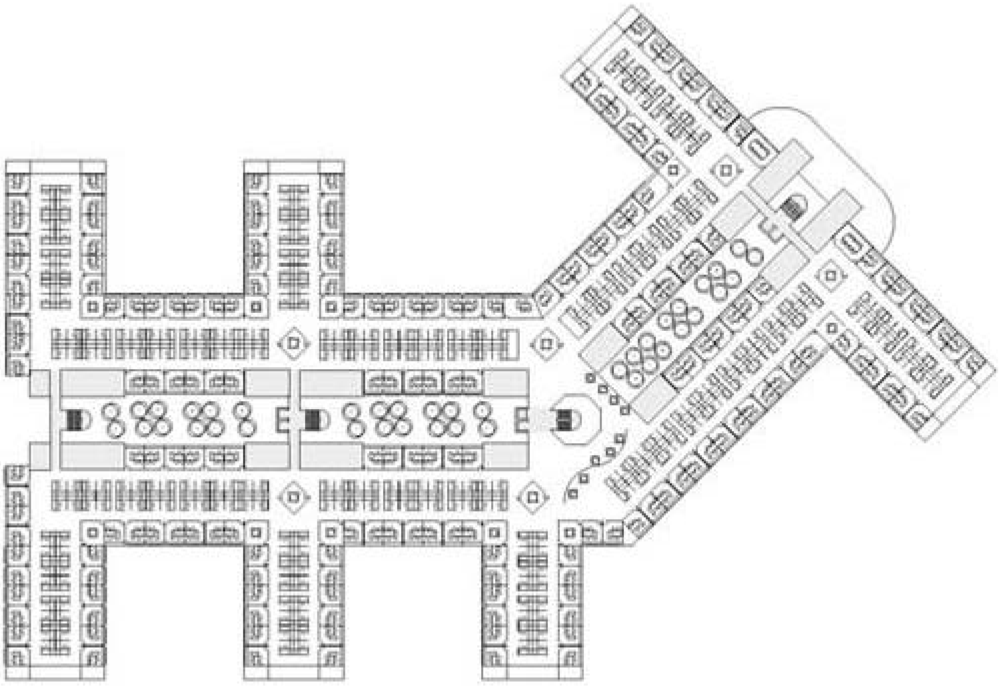
80,000 useable area × 1.125 = 90,000 rentable area

$$\frac{90{,}000 \text{ rentable area}}{1.125 \text{ R/U ratio}} = 80{,}000 \text{ useable area}$$

Typical Open-Plan Layout*



Typical Enclosed-Plan Layout*



*Drawing are not to scale.

Open Plan	Interior Layout	Base Building	Overall Building
1. Circulation			0.36
Secondary	0.33		
Primary		0.08	
2. Mechanical		0.03	0.03
3. Walls, Partitions, Structure			0.07
Interior	0.06		
Base Building		0.03	
4. Public Toilets		0.02	0.02
5. Building Storage		0.01	0.01
Unassigned Area	**0.39**	**0.17**	**0.49**
Net Assigned Area	**0.61**	-	**0.51**
Useable Area	1.00	**0.83**	-
Building Gross Area	-	1.00	1.00

Enclosed Plan	Interior Layout	Base Building	Overall Building
1. Circulation			0.32
Secondary	0.24		
Primary		0.10	
2. Mechanical		0.07	0.07
3. Walls, Partitions, Structure			0.07
Interior	0.08		
Base Building		0.05	
4. Public Toilets		0.02	0.02
5. Building Storage		0.01	0.01
Unassigned Area	**0.32**	**0.25**	**0.49**
Net Assigned Area	**0.68**	-	**0.51**
Useable Area	1.00	**0.75**	-
Building Gross Area	-	1.00	1.00

Base Building Efficiency

Often a building design separates the building shell (exterior walls, foundations, and columns) and building core (primary circulation, mechanical areas, public toilets, janitor closets, and building storage) from the interior layout of occupant specific use of a building. When determining the building gross area for core and shell design, divide the required useable area by the base building efficiency (typically ranges from 75 percent to 85 percent). For a commercial building, the shell and core design is based on the rentable area required to meet the client's financial goals. In this case, multiply the useable area by the estimated R/U ratio to calculate the rentable area.

Interior Layout Efficiency

When determining the areas for an interior design program, one is predicting the size of the useable area and providing a sound basis for estimating the budget for interior construction or "tenant fit-up." The layout efficiency will vary according to the office planning concept, as shown in the accompanying charts. For example, an enclosed office arrangement may require a 70 percent layout efficiency. On the other hand, an open plan layout can range from 60 percent to 65 percent layout efficiency depending on the size of the net assignable areas and adequacy of secondary circulation. Conversely, the client may have established the rentable area, and the task is to determine the useable area available for the design of the interior space. In this case, divide the rentable area by the estimated R/U ratio.

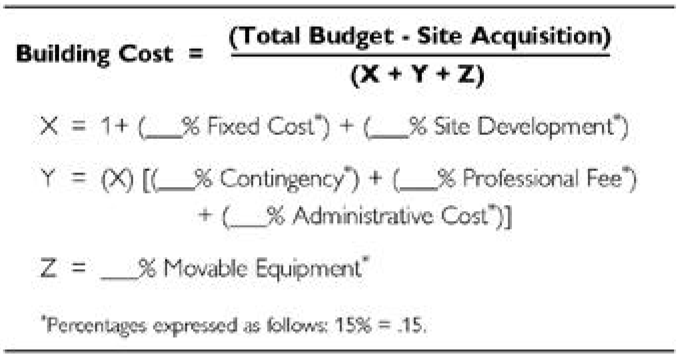
Cost Estimate Analysis

The cost estimate analysis for a new building must be as comprehensive and realistic as possible, with no doubt as to what constitutes the total budget required. Once the programmer has determined the total net assignable area of a project, it is an easy task to arrive at a reasonable efficiency factor and then calculate the total gross building area. This area, multiplied by a realistic unit cost, will produce the estimated building cost (Line A), upon which depend estimates of many cost items.

Cost Estimate Analysis Example

A. Building Costs	200,000 GSF @ $90.00/GSF	$18,000,0
B. Fixed Equipment	(8% of A)	1,440,0
C. Site Development	(15% of A)	2,700,0
D. Total Construction	(A + B + C)	$22,140,0
E. Site Acquisition/Demolition		500,0
F. Moveable Equipment	(8% of A)	1,440,0
G. Professional Fees	(6% of D)	1,328,4
H. Contingencies	(10% of D)	2,214,0
J. Administrative Costs	(1% of D)	442,8
K. Total Budget Required	(D + E through J)	$28,065,2

Even before determining the total gross area from the space program, it may be judicious for the programmer to start with the available funds as comprising the total budget (Line K) and to work back to building cost (Line A) to find the approximate area that may be feasible to build within the total budget. The following formula may be used to reduce Line K, Total Budget Required to Line A, Building Cost:



$$\textbf{Building Cost} = \frac{\textbf{(Total Budget - Site Acquisition)}}{\textbf{(X + Y + Z)}}$$

X = 1+ (___% Fixed Cost*) + (___% Site Development*)

Y = (X) [(___% Contingency*) + (___% Professional Fee*)
+ (___% Administrative Cost*)]

Z = ___% Movable Equipment*

*Percentages expressed as follows: 15% = .15.

A. Building Cost: This includes all costs of construction within 5 feet of the building line, all items required by codes (fire extinguisher cabinets, fire alarm systems, etc.), and items normally found in buildings regardless of type (drinking fountains).

B. Fixed Equipment: This includes all equipment items that may be installed before completion of the building and that are a part of the construction contract, such as lockers, food service equipment, fixed seating, fixed medical equipment, security equipment, stage equipment, stage lighting, etc.

C. Site Development: This includes all work required that lies within the site boundary and 5 feet from the edge of the building, that is, grading and fill, fencing, electronic perimeter system, roads and parking, utilities, landscape development, athletic fields, walks, site lighting, street furniture, site graphics, on-site sewage treatment plant, and unusual foundation conditions.

D. Total Construction: This represents the total budget for construction, usually the contract documents base bid.

E. Site Acquisition and/or Demolition: The money budgeted for purchasing the project site and/or demolishing existing structures.

F. Movable Equipment: This category includes all movable equipment and furniture items, but does not include operational equipment (i.e. microscopes, library books, and so on, purchased from operating funds).

G. Fees: Costs of architectural and engineering services and of consultant services.

H. Contingency: A percentage of the total construction cost is included to serve as a planning contingency, bidding contingency, and construction reserve (change orders, etc.)

J. Administrative Costs: Items the owner is responsible for during the planning process, that is, legal fees, site survey, soil testing, insurance, and material testing.

K. Total Budget: This represents the total budget required to occupy the new facility and/or renovated areas.

Line Item Cost Allocations

Use historical percentages of project cost to calculate the Total Budget Required (Line K). The percentages listed indicate the usual ranges of variation depending on the building type, existing conditions, and other factors.

Cost Estimate Analysis Line Items

A. Building Cost:		
Building Gross Area	× Unit Cost	= Building Cost
200,000 GSF	× $90/GSF	= $18,000,000
B. Fixed Equipment:		**Percentage of Line A**
Low		5%
Medium		10–15%
High		20%
Especially High		30%
C. Site Development:		**Percentage of Line A**
Low		5%
Medium		10–15%
High		20%
Specially High		30%
D. Total Construction Cost:		**Sum of A + B + C**

E. Site Acquisition and/or Demolition:	
Varies widely	$ Lump Sum Amount
F. Movable Equipment:	**Percentage of Line A**
Low	5%
Medium	10%–15%
High	20%
G. Professional Fees, Including Consultants:	**Percentage of Line D**
Varies	5% to 10%
H. Contingencies:	
Low	5%
Medium	10%
High	15%
J. Administrative Costs:	**Percentage of Line D**
Varies	1% to 2%

K. Total Budget: **Sum of D + E + F + G + H + J = K**

Components of Building Cost

When the Uniform Classification is used, the components of building cost (Line A) include: foundations, sub- and superstructure, exterior enclosure, roofing, interior construction, mechanical systems, electrical systems, conveying systems, and general conditions.

A 1. Foundations: Wall and column foundations and pile caps, plus special conditions.

A 2. Substructure: Slab on grade, basement excavation, structure walls.

A 3. Superstructure: Floor, roof, stair construction.

A 4. Exterior enclosure: Exterior walls, louvers, screens, balcony walls, handrails, soffits, doors, windows.

A 5. Roofing: roof coverings, traffic toppings, paving membrane, roof insulation and fill, flashing, roof openings.

A 6. Core Finish, Interior Fit-Up: Partitions, interior finishes and specialties, such as lockers, toilet accessories, counters, kitchen cabinets, closets.

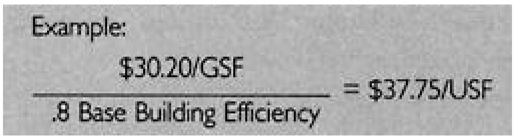
A 7. Mechanical: Plumbing, HVAC, fire protection, special systems.

A 8. Electrical: Service distribution, lighting and power, special electrical systems.

A 9. Conveying systems: Elevators, moving stairs and walks, dumbwaiters, general construction items.

A 10. General conditions and profit: Mobilization, site overhead, demobilization, office expense, profit.

The following chart represents components of building cost for an office building at a moderate/excellent level of quality. The chart also shows both the overall building and costs between the base building unit cost and interior layout cost. To convert the building gross area unit cost for interior fit-up to a useable area parameter, divide the gross area unit cost by the base building efficiency.



Example:

$$\frac{\$30.20/\text{GSF}}{.8 \text{ Base Building Efficiency}} = \$37.75/\text{USF}$$

Building Systems Performance Criteria: The performance criteria used for the evaluation and selection of building systems. They define the functionality sought from building systems to meet quality level expectations.

Building Systems: Components of a building organized by a specific discipline, such as architectural, structural, mechanical, electrical, and plumbing.

For detailed programming, a client/user often defines the building systems performances criteria for the whole building or for each space type. The unit cost allocated should achieve the building system performance criteria. For example, comfort control increases with smaller Heating Ventilating Air Conditioning (HVAC) zone areas. As a result, more mechanical equipment may be necessary to achieve this performance and the unit cost is greater, as shown in the chart below.

Performance Criteria	HVAC Zone	Air Distribution $/GSF
Low Control	3,000 SF	$5
Moderate Control	1,000 SF	$9
High Control	300 SF	$30

Components of Building Cost	Overall Building ($/GSF)	Base Building ($/GSF)	Interior Fit-Up ($/GSF)	Base Building Efficiency
A 1. Foundations	$2.50	$2.50		
A 2. Substructure	$11.00	$11.00		
A 3. Superstructure	$12.00	$12.00		
A 4. Exterior Enclosure	$12.50	$12.50		
A 5. Roofing	$3.50	$3.50		
A 6. Core/Interior Fit-up	$21.70			
Base Building		$4.50		
Interior Fit-Up			$17.20	.80
A 7. Mechanical	$18.20			
Base Building		$13.00		
Interior Fit-Up			$5.20	.80
A 8. Electrical	$13.80			
Base Building		$9.00		
Interior Fit-Up			$4.80	
A 9. Conveying Systems	$2.50	$2.50		
A 10. General Conditions and Profit	$10.76			
Base Building		$7.76		
Interior Fit-Up			$3.00	.80
Line A Unit Cost	**$108.46**	**$78.26**	**$30.20**	**-**

*Based on 1998 costs.

Interior Cost Estimate

Interior fit-up is usually defined as the tenant component of new construction or as the remodeling of existing interior space. Area references for interior fit-up usually ignore core and shell components of a building, so only the useable area applies. The programmer calculates the useable area, then, instead of the gross area as the primary parameter for Line A unit cost.

Example:

Line A Interior Fit-Up

Useable area × Unit Cost = Interior Fit-Up Cost

100,000 USF × $32/USF = $3,200,000

In the case of new construction, the tenant space is empty except for the air distribution system, usually the perimeter diffusers. Light fixtures may be stacked on the floor. Tenant fit-up typically includes partitions, doors, casework, finishes on all surfaces, limited plumbing for coffee bars or private toilets, relocation of sprinkler heads, heating, ventilating and air-conditioning distribution, supplementary exhaust or cooling systems, lighting installation, power distribution, and telecommunications rough-in and fixed equipment. But it is not necessarily limited to these.

Interior fit-up costs vary as a function of the ratio of enclosed space to open space, quality of finishes, and performance level of building systems. Refer to the following list of components for interior fit-up.

Open-plan offices with minimal enclosed spaces have a lower interior fit-up cost. In comparison, an enclosed-

Interior Fit-Up Costs	**Open Plan**	**Enclosed Plan**
A 6a.Partitions Drywall, concrete masonry unit, folding and glazed	**$3.50**	**$7.50**
A 6b.Doors Wood, hollow metal, and glass with frames and hardware	**$2.00**	$4.00
A 6c.Finishes Floor, walls, and ceilings	**$4.50**	**$7.00**
A 6d.Casework Office shelving, storage, private toilets, work surfaces, and coffee bars	**$0.50**	**$2.00**
A 6e.Specialties Chalkboards, tack boards, audio-visual equipment, access flooring, graphics, lockers, and rest-room accessories	**$0.50**	**$1.00**
A 7a.Plumbing Private toilets, coffee bar sinks, and appliance hookups	**$0.50**	**$1.00**
A 7b.Fire Protection Sprinkler head installation	**$0.25**	**$0.50**
A 7c.Mechanical Air distribution, chilled-or hot-water distribution, fan coil units, computer air conditioning, and special exhausts	**$3.50**	**$5.00**
A 8a.Electrical Lighting installation, special lighting,	**$3.50**	**$5.00**

power distribution, fire alarm modifications, public address, and security		
A 8b.Telecommunications	**$0.75**	**$1.00**
Phone and data rough-in, cabling, and/or equipment		
A 10.General Conditions and Profit	**$2.15**	**$3.75**
Line A Unit Cost per USF	**$21.65**	**$37.75**

* Based on 1998 costs.

plan office with more permanent partitions, casework, and finishes will have a higher fit-up cost. While the Line A unit cost for open plan is lower than for enclosed plan, open plan most likely has a higher allocation for moveable equipment (Line F) for the cost of the open plan layout furniture system.

Interior fit-up costs can range from $12.00 to $100.00 or more per useable area. The level of quality will also determine the variance in the unit cost. In a lease agreement, a landlord may provide tenant contract allowances (work order credits) that cover the cost to build out the interior space using the building standard systems and materials, normally at an economical level of quality. If a tenant does not use the building ’s standard and the actual interior fit-up exceeds the quality and scope of materials furnished, the tenant must fund the additional cost. Levels of quality are discussed in the section that follows.

Site Development Costs

Site development costs (Line D) vary widely depending on the requirements of the building type, the nature and location of the site, and the quality level of the development. Site development costs vary from a low of 5 percent of the Line A building cost and a medium level between 10 percent and 15 percent to a high of 20 percent. The especially high percentage of 30 percent allows for extraordinary conditions such as rock excavation, very steep slopes and intensive development requirements. Use the accompanying chart of site development components as a checklist to define requirements and develop a more detailed estimate.

Site Development Costs

D 1. Site Preparation

Estimate 1% to 3% of building costs.

D 2. Parking

On Grade:	= 350 SF–400 SF/car.
Allow 125 cars per acre	= $1,200–$1,500/car
Estimate lump sum	
Structural:	
Allow 280–325 SF/car	
Estimate lump sum	= $6,000–$7,500/car

D 3. Roadways
Estimate lump sum per linear foot.

D 4. Sidewalks and Terraces
Estimate 1–7% of building cost.

D 5. Walls and Screens
Estimate .5–2.5% of building cost.

D 6. Outdoor Sports Facilities
Estimate lump sum per unit and type.

D 7. On-Site Utilities
Estimate 1–3% of building cost.

D 8. Off-Site Utilities (if required)
Estimate 3–5% of building cost

D 9. Storm Drainage
Estimate .5–.5% of building cost.

D 10. Landscaping
Estimate planting 1–2% of building cost.

D 11. Outdoor Equipment
Estimate lump sum.

D 12. Outdoor Lighting
Estimate pedestrian lighting 1% of building

cost, and parking lighting lump sum per car.

*Based on 1998 costs.

Building Quality Levels

The building cost (Line A of the Cost Estimate Analysis) depends on (1) the total net area (the sum of all space needs), (2) a reasonable efficiency ratio of net to gross area, and (3) the cost per square foot escalated to mid-construction. Of these, it is the cost per square foot, the unit cost, that is usually associated with the quality of the building.

Types of Quality: It is true that the cost per square foot represents the quality of materials, systems, and construction — the quality of the architectural fabric. But it should come as no surprise that both the total net area and the building efficiency also represent aspects of quality—functional and spatial qualities, respectively.

Levels of Quality: But before covering the types of quality in more detail, it is helpful to discuss different levels of quality. It should be obvious that the architect and the client must reach an agreement on the level of quality for the project. The client must be conscious of a wide range of choices.

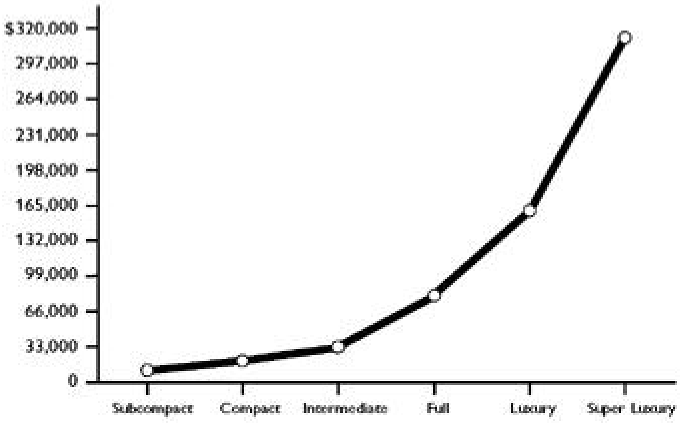
Automobile Analogy: One useful device is the analogy of automobiles. A client can be expected to understand the difference in quality between a Volkswagen Beetle and a Rolls Royce—between an austere and a superb quality—without having to resort to a detailed analysis. To round out the analogy without using trade names, consider the following six levels of quality:

For Automobiles	For Buildings
Super Luxury	Superb
Luxury	Grand
Full	Excellent
Intermediate	Moderate
Compact	Economical
Subcompact	Austere

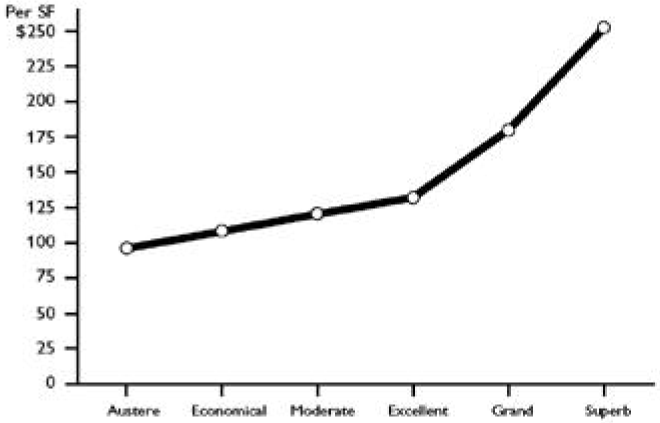
The accompanying chart shows the relationship between the unit cost for automobiles and quality level. The unit costs are taken from a consumer publication indicating the “best buys” in each category. Note that the difference in levels is gradual until the last two. The super-luxury level is actually shown at one third of its potential.

The point of the analogy is this: autos and buildings share the same wide range in levels of quality. They also share similar quality factors, based on (1) materials, systems, and construction, (2) function and performance, and (3) spatial qualities. Clients and architects must be aware of the wide range in levels of quality, and they must agree on a realistic quality level for which funds are available.

Quality Levels/Unit Costs for Automobiles*



Quality Levels/Unit Costs for Civic Auditoriums*



*Based on 1998 costs.

Efficiency and Quality: One aspect of architectural quality, spaciousness, is inversely proportional to the overall efficiency of a building. Therefore, it is important to predict and assign a reasonable efficiency for a building that would contribute to its expected quality.

For example: one of the factors affecting the architectural quality of a civic auditorium is plan efficiency. A civic auditorium intended as a statement of community pride would surely have an efficiency of 50 percent net floor area to 50 percent unassigned area. On the other hand, a civic auditorium intended merely as a necessary solution would have an austere 70 percent overall efficiency.

With a superb and an austere on opposite ends of a scale, a value judgment can be made regarding the quality intended and the reasonable efficiency that can be assumed for planning purposes. Further still, this scale can be expanded to provide a full range of quality levels, but not for the same building type.

Using six levels seems appropriate for most building types; however, they may not be the same six levels or there may be more than six levels. Building services, for example, would start with a skeletal 90 percent efficiency depending on the components and on the predominance of warehousing.

To help clients visualize the quality level implications of different efficiency factors, a programmer should do area takeoffs of existing floor plans and illustrate graphically how the areas are measured.

Quality of Construction: The construction quality level is represented by a unit cost figure, such as cost per gross square foot. The unit costs typically include architectural, structural, electrical, plumbing, and mechanical work, but do not include site development and fixed equipment.

The average unit costs are typically identified with different types of construction or building types related to building code fire ratings, but these average unit costs represent only the average quality level of construction in each type. The average quality represents good standard construction with adequate mechanical and electrical services and an average level of finishes. These average unit costs can be used to advantage; however, when programming, there is a great need to know a wider range of unit costs than those representing national averages.

The accompanying chart indicates the possibility of six choices in quality—ranging from austere to superb. The chart is a heuristic device to find the appropriate level of quality for a project and to be aware of the wide range of unit costs. National averages usually span more than three of these unit cost figures—most often in the lower end of the range. The level of quality depends on the level of construction, mechanical and electrical services, and interior and exterior finishes.

Civic auditoriums range from high school auditoriums used by the community to halls for the performing arts. Their unit costs represent a wide range in levels of quality.

Offices also have a wide range of types: low-rise offices, high-rise offices, medical offices, and municipal offices. Most industry sources provide three or four levels of quality, but for clients that want higher or lower levels, the chart below provides six levels of choice.

Building Type	**Austere**	**Economical**	**Moderate**	**Excellent**	**Grand**
	($/GSF)	($/GSF)	($/GSF)	($/GSF)	($/GSF)
Civic Auditoriums	96	108	120	132	180
Research Laboratories	125	146	168	208	250
Correctional Facilities	90	108	138	162	180
Hospitals	108	124	128	155	169
Offices	48	66	84	114	162
Libraries	72	84	96	108	139
Civic Centers	78	90	102	114	132
Education Facilities	50	69	81	94	106
Warehouses	29	35	41	76	58

Approximate national average unit cost per gross building square foot as of January 1998.

Under educational facilities, the wide range of unit costs can be justified by the wide range in educational levels: elementary schools, secondary schools, community colleges, and university buildings.

While warehouses have unit costs covering the lower ranges because they are not usually of high quality in construction, services, and finishes, there are those warehouses that are the exception.

Various national organizations publish timely unit costs based on national averages. They also publish regional modifiers or location factors for each state …and even for individual cities.

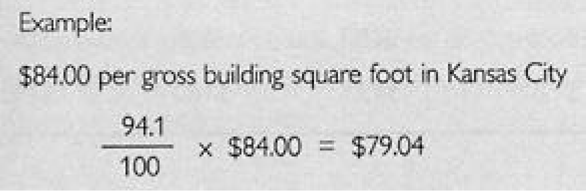
Some widely used sources of cost information include:

RS Means Square Foot Cost, annual edition. RS Means CMD Group, Kingston, MA.

Building Design and Construction magazine, Cahers Publishing, Deplaines, IL.

Design Cost Data magazine. DC&D Technologies, Inc., Tampa, FL.

To adjust the unit cost for a particular building type (based on national averages) for the specific location, multiply the unit cost by the location factor divided by 100 (the average location factor).



Example:

$84.00 per gross building square foot in Kansas City

$$\frac{94.1}{100} \times \$84.00 = \$79.04$$

City	Location Factor
Atlanta	88.3
Boston	118.5
Chicago	108.1
Dallas	86.4
Denver	93.1
Houston	89.7
Kansas City	94.1
Los Angeles	112.3
New York	135.2
Orlando	87.4
Philadelphia	110.8
Phoenix	91.2
San Francisco	125.7
Seattle	105.4
St. Louis	101.5

Further, unit costs become obsolete with time due to inflation. It is necessary, therefore, to escalate the adjusted unit cost by a reasonable percentage per year projected to midway through the construction period.

Example:

8% per year for two years to midpoint construction

$$\$79.04 \times [1+ (2x\ .08)] = \$91.69$$

Functional Adequacy: The six levels of quality are also applicable to the functional adequacy of a building. Theoretically, this term refers to the total net area per unit. Actually, most references of this type are made in building gross area per unit—which is complicated by a variable building efficiency that makes comparisons difficult. Nevertheless, this area per unit is intended to indicate the level of service and support per unit. Here are some examples: the area per bed in a hospital; the area per student in a high school, college or university; and the area per seat in an auditorium.

A 1500-student high school, with an overall efficiency of 65 percent, could have an austere 120 GSF per student but without an auditorium and a spectator gym. It could have a moderate 140 GSF per student but with limited vocational facilities. The superb 200 GSF per student would include educational enrichments wanted by many communities. The student capacity of the school is an important factor related to the central service facilities.

A 1000-student high school would have higher areas per student and a 2000-student school, lower areas.

Similarly, an auditorium would have a wide range in levels—from an austere 20 GSF per seat to a superb 90 GSF per seat for the same general capacity of 2500 seats. Again, the capacity is an important factor. A 500-seat auditorium would have a higher range in levels—higher areas per seat. Refer to the chart showing the 20 GSF to 90 GSF per seat range. The austere 20 GSF per seat would indicate limited lobbies, offices, storage, stage, and backstage facilities. These facilities would all increase with the rising levels of quality, even to include public restaurants.

Establishing Quality Level: Consider the six levels of quality and the automobile analogy as heuristic devices to expand the usual narrow range of quality levels and to establish the appropriate level of quality for a project. These might help both the client and the designer to

Civic Auditoriums within a 2000-3000 Seat Capacity Range

Kinds of Quality	**Austere**	**Economical**	**Moderate**	**Excellent**	**Grand**
Gross Area/Seat	20 GSF	25 GSF	30 GSF	40 GSF	60 GSF
Overall Building Efficiency	70%	67%	65%	60%	55%
Cost per GSF	$96	$108	$120	$132	$180

strive for the same appropriate level among the kinds of quality. This would prevent the total mismatch of a Volkswagen Beetle body with a Rolls Royce engine—although some inconsistency may be necessary to balance the budget.

The example of civic auditoriums is shown in the accompanying chart. It can be assumed that a small total budget imposes a low unit cost and area per seat—which, in turn, affect the overall efficiency. Compare the figures for the austere quality with those of the grand quality. Functional adequacy increased (by a ratio of 1:3), overall efficiency increased (by a ratio of 1:1.5), and the building cost increased (by a ratio of 1:2.35).

Building Renovation: Renovation work is becoming very popular with many organizations that face changing missions yet often have existing buildings that have become obsolete or do not fit required new uses. Facilities are left standing empty. So it is only natural to assume that these buildings can be renovated more easily and cheaply than building new construction. But renovation work can be very complex and expensive. It can range from a simple open plan office renovation with minimal impact, to hard construction and utilities, to the renovation of an old building for new occupancy that fails to comply with a variety of codes, and may have hazardous materials to abate.

The age of a building is directly proportional to the cost of renovation. Issues that make an old building expensive are prior occupancy; floor-to-floor height; mechanical, electrical, and plumbing systems; energy efficiency; structural capacities; seismic codes; and life safety and disabilities access guidelines.

If the previous use is not easily adapted to the new occupancy, expect to achieve a lower layout efficiency that will contribute to higher project cost. Even site development expenses are possible if utility capacities parking and site development are inadequate.

Major renovations almost always require compliance with all current codes. If the floor-to-floor height is less than desirable, the mechanical, electrical, and plumbing design will incur cost penalties.

Often, the original structural drawings are unavailable, forcing one to do expensive tests to determine structural conformance to new codes. Exterior wall glazing may fail to comply to energy codes.

In some cases, the only systems that can be salvaged are structure and solid exterior walls. A renovation of this nature will rival new construction in cost.

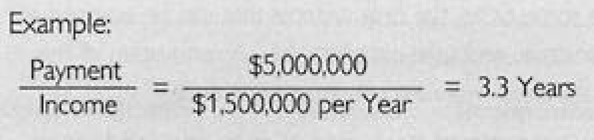
Always compare major renovation to new construction, even if it is desirable to salvage the building for historical purposes. Generally, the programmer should base a reliable renovation cost estimate on a building condition assessment that defines the degree of improvement required.

Financial Analysis: Financial analysis addresses the time value of money when a client is evaluating programmatic alternatives. The analysis adjusts varying economic values to comparable figures or to values consistent with other financial measures used by the client's organization. Generally, the assumptions and outcome from the analysis vary depending on the client's economic point of view: as an owner or as an investor.

Life Cycle Cost Analysis: The client as an owner will address a combination of income (cost savings) and payouts (capital and expense cash flows) over a period of years.

Investment Performance Analysis: The client as an investor will address the combination of income generation and payouts (capital and expense investments) over a period of years.

Payback: A simple indicator of the benefit of an investment is the calculation of the point in time when the income (or savings) equals the payment (or cost) of the investment.



Example:

$$\frac{\text{Payment}}{\text{Income}} = \frac{\$5,000,000}{\$1,500,000 \text{ per Year}} = 3.3 \text{ Years}$$

Discounted Cash Flow Analysis (DCFA)

Dissecting this phrase reveals the fundamental meaning behind the term—DCFA is the analysis of cash flows (or "streams"), discounted (or "brought back) to an equivalent dollar amount (in today's dollars). The benefit of discounting is that it levels the playing field by bringing all the future payments (rent, utilities, taxes, insurance, janitorial costs, and maintenance and repairs) back to a common date. One of the most common methods applied to this type of analysis is the determination of Net Present Value (NPV).

Net Present Value (NPV): The value of an investment based on a discount rate over a series of future payments (or costs) and income (or savings). NPV is very similar to (but the exact opposite of) calculating interest.

Example:

Assuming that you could put your $1 received today into a bank at 10 percent interest per year, it would be worth $1.10 at the end of the year. Similarly, if you will receive $1.10 at the end of a year and the bank's interest rate is 10 percent, the net present value is $1.00.

Present Value of Annuity (PV): The value now of a level series of payments to be received each period for a finite number of equal periods.

Technically, Net Present Value and Present Value are not synonymous; however, the two terms are often used interchangeably. NPV recognizes cash flows only at the end of a period, and accommodates variable payments or income streams that occur at regular periods. PV is based on constant payments made over continuous and equal periods.

Compounding

Most of us have a savings account. And most savings accounts accumulate compound interest over time. The concept is relatively straightforward. Your money in the account and the interest earned over each earning "period" (year, quarter, month, and day) both earn interest. "Compounded Interest" is simply interest on interest.

Example:

Assume that you placed $1.00 in an interest bearing account and left it there for five years. Also assume that the account pays 10 percent interest, compounded *annually*. The following table represents your investment and its earnings. At the end of the five-year period, your initial investment of $1.00 has grown to $1.61.

Period	Investment	Cumulative
Year 1	$1.00	$1.10
Year 2	$1.10	$1.21
Year 3	$1.21	$1.33
Year 4	$1.33	$1.46
Year 5	$1.46	$1.61

Discounting

Discounting is the opposite of compounding. Discounting is equivalent to asking, "What dollar amount do I need to invest today (assume 10 percent interest) to ensure that I will have $1.61 five years from now?" The answer to that question is easy since it is the opposite of the above example. However, without a financial calculator or spreadsheet, the answer becomes increasingly more complicated as you add more variables (like rent, utilities, taxes, insurance, etc.). Therefore, the introduction of spreadsheets (especially those with embedded financial-analysis functions) has led to more common use of discounted cash flow analysis.

Discount Rate: A compound interest rate used to convert expected income, expenses, or future cash flows to a present value.

Discount Factor: A factor equal to the present value of one discounted for a particular time period and at a specific compound discount rate. See table below.

Discount Factor for
Present Value of 1 at Compound Discount Rates

Time Periods	1%	3%	5%	7%	9%
1	0.9901	0.9709	0.9524	0.9346	0.9174
2	0.9803	0.9426	0.9070	0.8734	0.8417
3	0.9706	0.9151	0.8638	0.8163	0.7722
4	0.9610	0.8885	0.8227	0.7629	0.7084
5	0.9515	0.8626	0.7835	0.7130	0.6499
6	0.9420	0.8375	0.7462	0.6663	0.5963
7	0.9327	0.8131	0.7107	0.6227	0.5470
8	0.9235	0.7894	0.6768	0.5820	0.5019
9	0.9143	0.7664	0.6446	0.5439	0.4604
10	0.9053	0.7441	0.6139	0.5083	0.4224

Hurdle Rate: The minimum rate of return for a particular discounted cash flow analysis. The hurdle rate may vary depending on the risk profile for the investment.

The general rule of thumb is the higher the risk, the higher the hurdle rate.

Internal Rate of Return: The percentage rate earned on each dollar that remains in an investment each year. The IRR of an investment is the same as the discount rate at which the sum of the present value of future cash flows equals the initial capital investment.

Balance Sheet: A detailed listing of assets and liabilities for a person or business. The delta between assets and liabilities is “net worth” or “equity.”

Capital: All funds employed in a business, including debt and equity.

Example:		
Equity	$42,000,000	70%
Debt	$18,000,000	30%
Total Capital	$60,000,000	100%

Cost of Capital: The rate of return from an investment with similar risk (compared to the base investment).

Cost of Equity: The rate of return required by a shareholder or investor.

Cost of Debt: The rate of return required by a bank or lender.

Tax Rate: The ratio of a tax assessment to the amount being taxed.

After-Tax Cost of Debt: The cost of debt adjusted for the benefit of tax deductions at the tax rate.

In the example below, the cost of debt (e.g., a loan from bank) is 8 percent, and the tax rate is 40 percent.

Example:		
After-Tax Cost of Debt	=	Cost of Debt × (1 - the Tax Rate)
After-Tax Cost of Debt	=	.08 × (1- .4)
	=	.08 × .6
	=	.048

Weighted Average Cost of Capital (WACC): This is synonymous with a organization’s hurdle rate or discount rate. It is calculated using the debt and equity positions for an organization and their relative percentages. In the example below, 70 percent of the organization’s capital is equity with a cost of 11 percent, and 30 percent is debt with an after tax cost of 4.8 percent. Therefore, the WACC is 9.1 percent.

Example:

$$\text{WACC} = \left(\begin{matrix}\text{Cost of}\\ \text{Equity}\end{matrix} \times \begin{matrix}\text{Percentage}\\ \text{of Equity}\end{matrix}\right) + \left(\begin{matrix}\text{After-Tax}\\ \text{Cost of Debt}\end{matrix} \times \begin{matrix}\text{Percentage}\\ \text{of Debt}\end{matrix}\right)$$

	Weight	x	Cost	=	Average
Equity	.7		11.0%		7.7%
Debt	.3		4.8%		1.4%
WACC					9.1%

Inflation Rate: The rate at which the cost of living and working is expected to change.

Economic Life: The useful life of an investment.

Pro Forma: A hypothetical financial analysis involving assumptions commonly used to analyze "what-if'' scenarios. In general the client will provide the programmer with the assumptions on which the financial analysis should be based, including inflation rate, cost of capital (or discount rate), and economic life.

For example, for an inflation rate of 3 percent, cost of capital 9.1 percent (assume a 9 percent discount rate), an economic life of five years, which programmatic alternative of equal risk should the programmer recommend?

Programmatic Alternative A requires a total project cost of $29,000,000 for design and construction, and results in an annual operating cost of $1,000,000.

Programmatic Alternative B requires a total project cost of $24,000,000 for design and construction, and results in an annual operating cost of $2,500,000.

In the pro forma example below, the net present value is –$22,670,000 for Alternative A and –$24,000,000 for Alternative B. Since the risks are the same for both alternatives, the net present value of Alternative A is $1,330,000 less than Alternative B, and would be the preferred alternative based on financial criteria.

Discounted Cash Flow Analysis Example (Dollars in $1,000s)

(Annual operating cost savings for Alternative A equals Alternative B operating cost of $2,500,000 less Alternative A c

	Initial Period 0	**Fut Per 1**
Payments		
–Initial Design and Construction Cost	–$29,000	
Income		
–Annual Operating Cost Savings	$0	$1,
–Inflation @ 3%/year	x 1.000	x 1
–Escalated Annual Operating Cost Savings	$0	$1,
Cash Flow	–$29,000	$1,
–Discount Factor @ 9%	x 1.000	x .
Net Present Value ($000) -$22,670 =	–$29,000 +	$1,4

Net Present Value of Alternative B	**$24,070,000**
Net Present Value of Alternative A	**–$22,670,000**
Difference	**$1,330,000**

On Problem Statements

Problem Statement: A description of the critical conditions and design premises that become the starting point for schematic design.

Hypothesis: An assumed or real condition taken as a basis for inference from which to draw conclusions.

Condition: Something established or agreed upon as a requisite to the doing of something else.

Premise: A condition stated as leading to a conclusion.

Design Premise: A specific condition leading to a general design directive.

Criteria: The standards by which performances are tested or judged.

Design Criteria: The problem statements used as standards to judge a design solution.

Abstract (Adjective): Having no reference to a thing or things: opposed to concrete.

Abstract (Noun): A synopsis or the concentrated essence of a larger whole, after the filtering out of unneeded details.

Essence: The intrinsic of indispensable properties. The essential nature of a thing.

The following pages contain Problem Statements from actual projects covering different phases and building types. These have been written by different programmer/designer teams over the past 50 years. Note the different styles and formats—even different titles.

Yet the statements follow the format of identifying a condition leading to a general design directive. Moreover, each is a comprehensive statement covering Function, Form, Economy, and Time.

Military Academy
Master Plan
November 1974

Function

Since the emphasis must be placed on pedestrian movement in the cadet zone and in the family housing/community service center, **the master plan must provide for the separation of pedestrian movement and vehicular traffic.**

Since the predominant cadet formation will be a company with platoons in line, **the master plan should respond with broad aprons and sidewalks.**

Economy

Since the Academy will be a military showcase, **the quality of design and construction must be of a high level.**

Form

Since the cadet zone must locate facilities within a 5-to-6- minute walking distance, **the master plan must respond with the appropriate density.**

Since the area is barren and austere, **the master plan should create green planted areas for the psychological effect.**

Since the projected image of the academic campus must reflect the military values of strength, order, and discipline, **the master plan should respond to this image.**

Time

Since the Academy may grow even beyond the two planned phases, **the master plan must allow an open-ended framework for expansion.**

Satellite Office
Consolidation Plan
March 1997

Function

Since the corporation intends to focus on people as its strength, **the project should balance the need for group and client spaces: neighborhood settings, technology showcases, and interaction area with the need for private spaces: work booths, increased meeting rooms, and individual storage areas.**

Since the corporation intends to share one vision/act as one team, **the proposed plan should provide spaces that allow a shift from the individual to team focus: project rooms, enclaves, team spaces, customer presentation rooms, and business center.**

Since the corporation intends to have one standard of excellence, **the project should "put the company on the map for Marketing in Asia," creating an environment that adapts to change and uses the latest technology in a variety of office settings (group address, free address, fixed address) to fit the needs of each particular team.**

Form

Since the corporation intends to create superior value for the customers, **the project should create an innovative image to showcase its technology and people to clients, including real-time demos of their products.**

Economy

Since the corporation intends to provide shareholder value, **the project will improve the cost effectiveness through the relocation to the proposed location and will create operational synergy by moving two locations into one.**

Time

Since the corporation experiences temporary transformations in organizational structure to create new teams, **the facility must accommodate these changes.**

Headquarters
Site Selection Plan
October 1996

Function

Since the corporation may expand its hosting capabilities for a variety of event visitors, **the proposed location(s) within the target city should consider the potential for development of lodging accommodations to support conferencing and training facilities, and/or the proximity to major hotel chains, airport, ground transportation centers, and access to other entertainment facilities.**

Since internal communication between groups is critical, **the proposed facility should strive to satisfy the adjacency requirements on at least 30,000–33,000 GSF floorplate/footprint.**

Since the exhibition hall is a high-profile function, possibly independent of the offices, that anticipates large amounts of visitors, **the location proposal should seek locations that receive high public traffic.**

Economy

Since the development depends on return on investment, **location proposals must consider short-term as well as long-term effects of owning versus leasing.**

Form

Since the entrance to any space forms a first and lasting impression, **the entry sequence to the new location of the headquarters should communicate the importance of athletics and education.**

Since there is a desire to project an "open" feeling to the public, yet there are varying degrees of confidentiality requirements, **the location strategy should accommodate the desire to secure certain functions from the public while also providing an open, welcome feeling to visitors.**

Time

Since the corporation is interested in achieving ownership sooner rather than later, **the location proposal should provide flexible exit strategies for each proposed location.**

University Research Park
Master and Capital Plan
December 1983

Function

Since area requirements for ground tenant sites are not yet known, **the master plan must be designed with a flexible lot subdivision system.**

Economy

Since the municipal improvement district will be developed in Phase One, **the master plan should allocate as much of the site development to Phase One as feasible.**

Form

Since there will be a public street right-of-way dividing the site, **the master-plan design must integrate the two areas into a cohesive whole, as well as provide appropriate security for tenant site.**

Since the site is relatively featureless, **the master-plan design must provide the required image for the park.**

Time

Since the park will be built in phases, **the master plan must locate the common support facilities and amenities to serve all phases equally well.**

Medical Center and School of Medicine
Master Plan
July 1971

Function

The School of Medicine has strong functional and administrative ties with the existing university campus; **hence, a physical and visual connection between the two campuses is important.**

Ambulant patient care is the dominant aspect of this medical education, and **the character and positioning of the clinics must visibly reflect their key role.**

Economy

Recognizing the severe limitation of the budget, **continue to use appropriate cost control techniques and seek creative expression of this "lean and clean" quality in the architecture.**

Form

The Medical School educational and service programs are marked by their accessibility—health care for the walking patient, as well as the acutely ill bed patient, extension services to the region, and air transport for emergency care; **therefore, the school should have a corresponding sense of physical openness and outward orientation.**

Since there will be a large daily influx of patients at the clinics, many making their first visit, **special consideration must be made concerning patient orientation and direction.**

Time

The Medical School will be the core of the future Medical Center; **therefore, the school must be able to evolve and to grow to meet these new responsibilities and affiliations.**

Medical education concepts and programs will continue to evolve; **therefore, the architecture must have the convertibility to accommodate change.**

Specialized Office Center
Concept Design
March 1992

Function

Since some existing product line functions are in place, **locate the related functions adjacent to the anchors.**

Since maximizing utilization of existing facilities is important for efficient use of resources, **find opportunities for compatible fit of facilities and shared use.**

Since the consolidated group represents an opportunity to create a research, development, test, and evaluation center of excellence for aircraft development, **the facility should foster a work environment with optimum facility proximity, interaction areas, site amenities, and quality workspace.**

Economy

Since the budget is fixed, **prioritize construction dollars for R&D facilities and address feasibility of renovation on a building-by-building basis.**

Since it is desired to take advantage of the bid climate, **identify, prioritize, and add alternates up to 15 percent beyond the budgeted concept design.**

Form

Since the sites have sensitive environmental ecosystems, **develop mitigation plans in conjunction with the base environmental committee.**

Since the additional population at the base will result in substandard roadways, **optimize the improvement budget allocation to create a better base-wide transportation system.**

Time

Since the corporate requirements will likely change many times during the life of the building, **the facility must accommodate changes in corporate philosophy, the organizational structure and work process.**

Senior High School
Schematic Design
December 1950

Function

That students spend as much time in halls (more than an hour a day) as they do in any one classroom or laboratory. Therefore, halls and other circulation elements should be designed to help achieve the aims of the educational program. (Note: Perhaps this consideration provides the fundamental difference between the high school plant and the elementary school plant.)

That the school plant will be used year-round for community improvement, education, and recreation. Those elements that are to be used by both students and the public, such as the gymnasium and auditorium, should be grouped in one zone for efficient use and economical maintenance.

Economy

That within each individual teaching area, such as Homemaking, English, or Speech, there will always be changes in teaching techniques. Therefore, classrooms, laboratories, and shops should be designed for economical and efficient adaptations to these changes.

Form

That a well-balanced, effective program of education will accent communications among students in the classroom, as well as communication between the teacher and the student group. Therefore, teaching areas should be designed to allow flexibility of seating arrangement.

Time

That high school population will continue to grow and that courses of study will continue to be added to, or subtracted from, the curriculum. Therefore, the school must be designed so that it can be expanded economically and efficiently without marring the beauty of the school.

Urban Community College
Schematic Design
August 1973

Function

Since there is a diversity of student population and lifestyles, **there is the need to achieve a strong sense of place to foster interaction.**

Since the major user is the adult part-time student spending a short time in the facility, **careful consideration should be given to orientation and to circulation systems.**

Since the district has adopted an educational merchandising concept, **the visibility of the activities becomes a major design objective.**

Since the classrooms in the pool are shared by diverse teaching groups, **their physical distribution should be a major design determinant.**

Economy

Since the budget establishes the quality of construction at “above average,” **the design must consider the effect of urban conditions on materials and costs.**

Form

Since there is a need for capturing the spirit of a new urban-building type that combines educational, commercial, and office activities, **the design should respond to this unique need.**

Since the small urban site has numerous external physical and legal constraints, **the design should respond to these external influences, as well as to the needs for functional requirements.**

Time

Because of the indeterminacy of the academic programs now, and in the future, **convertibility and negotiability of classrooms should be a major design objective.**

Operational

To meet the goal for September 1976, occupancy, unique scheduling techniques, efficient construction methods, timely decision making in review and approvals, and availability of funds must be coordinated.

Community Mental Health and Retardation Center
Schematic Design
March 1969

Function

Because of the importance of the functional duality of the Center as both a state and a community center for mental health and for mental retardation, **the solution should express this duality.**

Since the goal for coordinated service, training, and research affects the multifunctional aspects, the solution **should encourage an interdisciplinary mix between these aspects of mental health and mental retardation.**

Because of the psychological-sociological nature of the people of the community, **the solution should provide the user with a clear sense of orientation.**

Economy

Because of the community's interest in "economy of means" and because of the numerous functions to be provided within a low-to-medium unit cost of $30.46 SF, **and the solution should strive for economy and multiuse of space.**

Form

Because of the relative position of the site to the university and the community, **the solution must provide for the interfacing of activities and of scale between the university and the immediate community.**

Time

Because the methods of mental health and mental retardation will change and because the needs of the community will change, **the Center must be adaptable to these changes.**

Because the facility will be used by the community on a continuing basis, **the solution should capture the spirit of a 24-hour concourse.**

Research and Development Center
Schematic Design
July 1984

Function

Since a high priority is placed on encouraging interaction between the research and the office personnel, **the design should maximize the relationship between office and lab as an operating unit.**

Since there is no particular "typical division," **the site plan and building design should be based on a general model of a division, group, and sector organization.**

Economy

Since this will be a corporate site, **building costs and site amenities should be consistent with those at other corporate sites.**

Since energy efficient design is important, **those energy conservation measures that show a four -year-or-better payback should be considered.**

Form

Since the development of the adjacent land is unknown at this time, **it is important to control access to the connector road.**

Since the development of this site will serve as a model for future growth in the area, **the site should communicate that "this quality company leads in quality growth in a sensitive area."**

Time

Since the project will be developed in preplanned phases, **the project delivery strategy should allow for occupancy of Phase One facilities by May 2002, and for occupancy of Phase Two facilities by June 2004.**

Operational

Because Phase Two construction will begin within months of the completion of Phase One, **the site design and phasing plan should locate Phase Two buildings to prevent serious construction obstructions to the users of the Phase One facilities.**

Convention Center
Schematic Design
December 1968

Function

The presence of the Convention Center generates parking requirements for large numbers of vehicles. Therefore, **the Center should provide adequate parking facilities without restricting off-site traffic flow.**

The exhibit hall generates a requirement for large truck/tractor access to, and egress from, the site. Therefore, **the site must accommodate maneuvering and storage of truck/tractor units without interfering with off-site traffic flow.**

Since the Convention Center site is bounded by major through-traffic arteries, **the new facilities should minimize the pedestrian-vehicle conflict.**

Economy

The budget is adequate for good-quality construction; however, **it is not without design implications.**

Form

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

Since the waterfront site is a unique feature of the city's image, **the Convention Center should touch the water and establish an activity connection at the water.**

Time

The current hotel capacity will have to expand to meet the ultimate requirements of the convention facilities (1500 to 2000 committed rooms). The success of the facilities depends on this expansion.

Phasing the building program will permit the interim time necessary for the response of the business community.

Additional Headquarters Office
Schematic Design
October 1984

Function

Since more than 30 separate departments or organizational groups will be co-located on the same site, **the design should strive to maintain departmental identity while locating departments for more efficient interaction and communication.**

The number of automobiles on the site is projected to grow by more than 150 percent by 1997. **On-site circulation and traffic to and from town will require careful and creative solutions to minimize traffic problems.**

Economy

Although the budget is adequate for a moderate-quality level of construction, **prudent and judicious use of materials and systems that reinforce the solid image of the company is advised.**

The plan should maintain and reinforce the natural beauty of the site and the integrity of the formal entry by the careful placement of new facilities.

Form

Since the new building will probably be in a more contemporary architectural style than the existing Headquarters, **the design should sensitively integrate a new facility that complements and does not clash with the existing structure.**

The existing and future facilities will share organizations and departments that will require constant interaction and movement. Appropriate site location of the new building and some form of a connection **between facilities are major design factors.**

Time

Phased growth of the staff population between move-in 1987 and 1997 will provide for built-in expansion space in the early years. **The plan should recognize this and locate these expansion areas for maximum availability and flexibility.**

Growth of departments over time may mean relocation and movement both within and between buildings. **The design should recognize this and consider buffer areas that easily allow for departmental movement and interim usage of space.**

Manufacturing Plant
Schematic Design
November 1980

Function

Since the operating center and team concepts lead to a strong and evolving organizational structure, **the design should respond with clear identity of areas and flexibility for change.**

Since safe and efficient traffic is a requirement, **the design must respond with a clear separation of pedestrian and vehicular traffic—and of car and truck movement.**

Since the production goals relate to layout efficiency, **the design must meet these criteria for efficiency.**

Since the program indicated different environmental conditions for machining and assembly, **the design should respond with a separation of these conditions.**

Economy

Since the type of construction is of moderate cost, **the design must proceed with rigorous cost control.**

Form

Since the partnership creates a completely new company, **the design should recognize the facility as a distinct corporate entity, as well as a functioning manufacturing plant.**

Since the surrounding community is an important consideration, **the design must respond with enhancement of the environment through sensitive site development.**

Time

Since the program indicated three potential stages of development, **the design must respond with strategies for growth.**

Energy

Since the manufacturing produces excess heat, **the design should take advantage of it when it is needed and dispose of it efficiently when it is not needed.**

Criminal Justice/Youth Center
Schematic Design
July 1975

Function

Since the living unit forms the background for the resident's identity and well-being, **the design must respond to a concept sensitive to this requirement.**

Since the functional organization calls for centralized service facilities surrounded by decentralized living units, **the design must respond to this grouping of activities.**

Since this is to be a medium-security facility, **the design must include provisions for adequate supervision and control.**

Economy

Since the budget is adequate, but not luxurious, **the design must respond with simplicity and directness.**

Form

Since the residents will be between the ages of 18 and 25, **the design must respond with a dynamic, playful, youthful character.**

Since the Environmental Impact Statement prescribes an image with a noninstitutional character, **the design should respond with forms of a scale and a proportion appropriate to satisfy this requirement.**

Since a normal, real-world psychological environment is sought, **the design should respond with an atmosphere similar to a college campus.**

Time

Since expansion of the facility is uncertain, **the design should provide visual and functional unity at each stage of development.**

Performing-Arts Hall
Schematic Design
March 1978

Function

Since all the performing arts need to be seen and heard under the best conditions, **the design should achieve superior sight lines and acoustical qualities.**

Since performing-arts events occur primarily in the evening, **the design should emphasize the nature of night activity.**

Since convenient flow of sets, costumes, and properties will reduce setup and breakdown time and costs, **the design should locate the stages at the same elevation as the receiving area, the scene shop, and the loading dock.**

Economy

Since the large hall must accommodate the symphony, opera, and ballet, **the multipurpose stage design must reconcile the different requirements of these arts.**

Since the cost for the architectural fabric of the large hall has been established within an excellent-to-grand quality, **the design should respond accordingly.**

Form

Since extraneous noise must be buffered from the performance area, **the design must acoustically isolate the mechanical room and scene shop.**

To reconcile the different seating-capacity preferences of the performing arts in the large hall, **the design must provide simple mechanical/electrical technology to reduce the capacity from 2100 seats to 1400 seats.**

Time

Because change is inevitable, **the concept of convertibility is important, particularly in offices for organizations and in the large hall (multiform).**

Professional Organization Offices
Space Planning and Interior Design
April 1979

Function

Since the office is accessible to the general public during working hours and must be accessible to employees during evenings and weekends, **the design should resolve the inherent security requirements.**

Although the company seeks an identity as one firm through uniform spatial and finish standards, **the design should respond to the unique functional requirements of each department.**

Several types of people visit the office, each with unique circulation requirements: 1) employees, 2) clients, 3) recruits, and 4) vendors; therefore, **the design should clearly separate conflicting circulation patterns.**

Economy

Since the company has a substantial investment in existing finishes and furniture, **the design should reuse these items when appropriate.**

Since the company will expand incrementally over the next 10 years in the building, **the space plan should establish the most economical mix of finished and furnished spaces.**

Form

Since the company is a prestigious international organization, **the design should convey an appropriate and distinguished corporate image.**

Since the core elements in the building are arranged asymmetrically, **the space plan should resolve special layout requirements for elevator access, and for cross-and vertical circulation.**

Since the company partners and managers are accustomed to the idea of hierarchy, **the design should maintain the arrangement of window offices.**

Time

The most economical leasing strategy requires some departments to switch floors at different time intervals; therefore, **the space plan should minimize disruption at each move, while considering the ultimate office arrangement.**

Since the exact growth of each department is uncertain, **the space plan should couple departments that might have offsetting growth patterns.**

Headquarters Office
Interior Design
June 1997

Function

Since the corporation, like any other cutting-edge business, considers reorganization and technological change to be constants, **the layout should strive for a highly flexible universal plan, which reduces the cost of frequent moves and changes.**

Since accommodation of the projected population and minimum workstation size (3.24 square meters) are the key drivers, **workspace standards should strive to provide functionality with modularity, allowing flexibility for the changing population and workstation units.**

Since support/common areas are truly "common" and various components of the corporation have changing common/project function needs, **common spaces should be located to provide easy accessibility for all users of the building, and should be designed for easy reconfigurations to satisfy the diverse needs of users.**

Economy

Since the budget must remain within the corporate guidelines, **the design should emphasize areas of higher quality by "putting the money in public areas."**

Form

Since the new headquarters is one of the few physical manifestations of a highly distributed business, **the design should be unique and communicate a distinct identity, while embodying the principles of partnership, economy, efficiency, and quality.**

Since visiting distributors are the primary focus (and measures of business success) of areas hosting tours and visits, **these floors (especially) should be designed to be warm, welcoming, accessible, and structured around the directed nature of a tour. It should be as much a place to visit as a place to work in.**

Time

Since the growth of departments over time may mean relocation and movement both within and between buildings, **the design should recognize this and consider buffer areas that easily allow for departmental movement and interim usage of space.**

Programming Procedures

There is a direct relationship between the Information Index (see page 36) and the Programming Procedures listed in this section. The Information Index uses key and evocative words and phrases to trigger specific questions about the project. The Programming Procedures give meaning to those words—charging them with meaning so that, thereafter, the words evoke questions beyond any prepared checklist.

These programming procedures are intended to provide stimulus to the programming process. There are more than enough procedures here to get the project under way. Certain procedures may apply in a specific project, while others may not; you'll have to test them to find out. You should then generate other procedures that apply to the specific project—still keeping the whole problem in mind.

The following procedures apply to architectural design programming as covered in the Primer. Applying problem seeking to other problem types requires defining new programming procedures. For example, there are information indexes for: master planning, interior design, engineering design, and management consulting. Each problem type requires the search for specific types of information. Therefore, while the five-step process remains the same, the considerations (or content) change accordingly.

Establish Goals

Function

(1) Understand why the project is being undertaken.

(2) Investigate the policy concerning the maximum number of people to be accommodated.

(3) Identify goals to maintain a sense of individual identity within a large mass of people.

(4) Identify goals for degrees and types of privacy and for group interaction.

(5) Investigate the hierarchy of values of the client/user.

(6) Identify goals concerning the promotion of certain activities as prime interests and their quality level.

(7) Identify the goal concerning the types of security required.

(8) Identify the goal concerning the effective continuity of progression (flow) of people and things.

(9) Investigate policies concerning the segregation of people, vehicles, and things.

(10) Identify goals dealing with the promotion of chance and planned encounters.

(11) Identify the policy concerning transportation (parking).

(12) Understand the implications of a goal for functional efficiency.

(13) Identify the goal concerning the priority of relationships.

Form

(14) Identify any client attitudes toward existing elements on the site (trees, water, open space, facilities, and utilities).

(15) Identify client attitudes toward the facility response to its environment.

(16) Investigate the land use policy for efficiency and environmental character.

(17) Identify policies concerning coincident planning and relations with the neighboring community.

(18) Identify policies concerning the investment in, or improvement of, the neighboring community.

(19) Identify the level of physical comfort required.

(20) Identify critical life safety considerations.

(21) Identify client attitudes toward the social/ psychological environment to be provided.

(22) Identify goals concerned with the promotion of the personal individuality of the user.

(23) Identify goals dealing with the flow of people and vehicles to provide wayfinding with a sense of orientation (knowing where you are), or a sense of entry (knowing where to enter).

(24) Identify the image that must be projected.

(25) Identify client attitudes toward the quality of the physical environment and the balance of space and quality.

Economy

(26) Identify the extent of available funds.

(27) Investigate the goal for cost effectiveness.

(28) Investigate the goal for maximum return—the most for the money.

(29) Investigate the goal for return on investment, for achieving financial gain.

(30) Identify the goal for minimizing the operational costs of the physical plant.

(31) Identify the goal for minimizing maintenance and operating costs.

(32) Identify the goal for establishing a priority on life-cycle costs or initial costs.

(33) Identify the client's goals for achieving a sustainable environment.

Time

(34) Identify client attitudes toward historic preservations.

(35) Determine client attitudes toward being static or dynamic as a social or functional organization.

(36) Identify client attitudes toward anticipated change.

(37) Identify client expectations for growth.

(38) Identify the desired occupancy date.

(39) Identify client goals for availability of funds over time.

Collect and Analyze Facts

Function

(40) Process raw statistical data into useful information.

(41) Generate area parameters from general activities (e.g., 150 GSF per office worker).

(42) Organize the personnel forecast listing the number of persons in each category and possibly their workloads.

(43) Analyze the physical, social, emotional, and intellectual characteristics of the people to be served.

(44) Analyze the characteristics of the community involved.

(45) Understand client organizational structure.

(46) Evaluate the potential loss to determine the degree of security controls required.

(47) Study the time-distance movement requirements.

(48) Analyze the different kinds of traffic lanes required by building occupants, pedestrians, and vehicles.

(49) Analyze the behavioral patterns of the client/user.

(50) Evaluate the space adequacy for the number of people and their activities to be housed.

(51) Identify the type and intensity of functional relationships.

(52) Analyze the requirements of special groups of people, such as the physically challenged.

Form

(53) Analyze the existing site conditions to include: contours, views, natural features, buildable areas, access and egress, utilities, size, and capacity.

(54) Evaluate the soil test report, and determine its implications on cost and design.

(55) Evaluate the floor area ratio, the ground area coverage, people per acre, and other comparative measures of density.

(56) Analyze the climate to include climatological data on seasonal temperatures, precipitation, snow, sun angles, and wind direction.

(57) Evaluate the form-giving significance of code and zoning requirements.

(58) Analyze local materials and the immediate surroundings of the site for possible influences.

(59) Understand the psychological implications of form on territoriality and the movement of people and vehicles.

(60) Define points of reference and entry.

(61) Establish a mutual understanding of building quality on a quantitative basis (cost per square foot).

(62) Understand the effect of building layout efficiency (commonly referred as net to gross ratio)on quality.

(63) Understand the effect of equipment cost on quality.

(64) Establish the functional adequacy (area/unit)of spaces as an indication of quality.

Economy

(65) Establish cost per square foot, considering escalation factor, local cost index, and construction quality level.

(66) Establish on a trial run the maximum budget required.

(67) Analyze the time-use factors for the different functions tentatively considered for combination.

(68) Evaluate the market-analysis report, and determine the implications on design.

(69) Analyze the different costs for the alternative energy sources.

(70) Analyze the climate factors, the wear -and-tear level of activities, and their implications on building materials.

(71) Analyze economic data related to initial versus life-cycle costs.

(72) Analyze the application of the Leadership in Energy and Environmental Design rating system.

Time

(73) Establish the full significance of the existing and neighboring buildings as having historic, aesthetic, and/or sentimental values.

(74) Generate space parameters from specific activities and the number of participants (e.g., 15 SF per dining seat).

(75) Identify the existing activities most likely to change.

(76) Identify long-term functional projections indicating growth or no growth.

(77) Determine a realistic time schedule for the complete project delivery.

(78) Analyze the implications of escalation factors.

Uncover and Test Concepts

Function

(79) Test the many services as best being centralized or decentralized.

(80) Investigate the sizes and kinds of groups to be housed—both now and in the future, including the physical, social, and emotional characteristics of people.

(81) Uncover the need for a family of closely related activities to be integrated into a unit, or the need for privacy (audio and/or visual)and for the degree of isolation (minimum, maximum).

(82) Uncover concepts establishing an order of importance, a priority based on what is valued or preferred and affecting relative position, sizes, and quality.

(83) Test the concept of hierarchy related to goals for the expression of symbols of authority.

(84) Understand how security controls are used to protect property and control personnel movement.

(85) Evaluate the flow charts dealing with the sequential movement of people, vehicles, services, goods, and information.

(86) Identify the need to separate completely traffic lanes to segregate different kinds of people (prisoners and public), different kinds of vehicular traffic (campus and urban traffic), or pedestrian and vehicular traffic.

(87) Identify the need for a common space dedicated to multidirectional, multipurpose traffic and intended to promote chance and planned encounters.

(88) Understand the organizational concepts and the functional relationships.

(89) Understand the use of networks or patterns of communication to promote the exchange of information.

Form

(90) Evaluate the natural features of the site, and identify those to be preserved or enhanced.

(91) Evaluate the climate analysis, and determine the implications on climate controls.

(92) Evaluate the form-giving implications of the code survey, and identify the salient safety precautions.

(93) Evaluate the soil analysis report, and determine the possibility of special foundations and their costs.

(94) Evaluate climate, demographic data, site conditions, and land value to establish general density standards.

(95) Evaluate the policy concerning the neighboring community to uncover the concept of sharing or interdependence.

(96) Uncover the need for an individual's home base or territoriality.

(97) Uncover the need for good orientation, maintaining a sense of direction through a building or campus.

(98) Uncover the need for the concept of accessibility, which promotes a sense of entrance and of arrival, providing direct access to public-oriented facilities.

(99) Uncover the general character of the architectural form that the client intends to project as an image.

(100) Understand that quality control is an operational concept used to provide the highest quality level feasible after the balance of quality/cost factors.

Economy

(101) Understand that cost control is an operational concept used to provide a realistic preview of probable costs after evaluating the pertinent facts.

(102) Understand that the efficient allocation of funds is an operational concept that utilizes formulas for the impartial allocation of space and money.

(103) Evaluate the time-use factors to determine the feasibility of combining various functions into a versatile, multifunction space.

(104) Uncover the need for the concept of merchandising used to promote business activities.

(105) Test the concept of energy conservation to determine the design and cost implications.

(106) Identify ways to reduce cost yet provide effective solutions.

(107) Identify ways to use renewable resources to achieve a sustainable environment.

Time

(108) Uncover the concept of adaptability in recycling an historic building for new activities and functions.

(109) Test the concept of tailored precision versus loose fit in determining the area requirements for an organization that might be static or dynamic.

(110) Uncover the concept of convertibility used to provide for interior changes in a building to accommodate future changes in activities.

(111) Uncover the concept of expansibility used to provide for exterior wall changes in a building to accommodate future growth.

(112) Test the conventional and fast-track procedures against the occupancy date to determine a realistic time schedule.

(113) Consider the phased approach to implement the project given constraints of time and cost.

Determine Needs

Function

(114) Identify the appropriate method of measuring: net, useable, rentable, and gross building area.

(115) Establish the area requirements for each activity by organization, location, space type, and time.

(116) Establish parking and outdoor-area requirements.

(117) Understand the cost implications of functional alternatives to providing facility, building, or site solutions.

Form

(118) Identify the components of site development cost.

(119) Consider the factors of the physical and psychological environment, as well as site conditions, as influences on the construction budget.

(120) Establish mutual agreement with the client on the construction quality expressed arements for each activity by organization, location, space type and time.

(121) Evaluate the efficiency factor that was used to determine the useable, rentable, or gross area requirements.

(122) Analyze the cost estimate and test for comprehensiveness and realism, leaving no doubt as to what comprises the total budget required.

(123) Establish a balance between space requirements, the budget, and quality.

(124) Analyze the cash flow required over time.\

(125) Evaluate the energy budget (if required).

(126) Evaluate the outline on operating costs (if required).

(127) Evaluate the report on life cycle costs (if required).

(128) Evaluate the level of sustainability desired using a rating system.

Time

(129) Evaluate the realism of the escalation factor to cover the time lag between programming and midconstruction.

(130) Determine a realistic time schedule for project delivery.

(131) Establish a time cost schedule of construction as an alternative to building the project in a single phase.

State the Problem

Function

(132) State the unique performance requirements to satisfy the personal or popular needs of the client/user.

(133) State the unique performance requirements to accommodate the major activities in the project.

(134) State the unique performance requirements created by the relationship among activities in the project.

Form

(135) Identify and abstract the major form-giving influences of the site on the building design.

(136) Identify the salient environmental influences on the building design.

(137) Identify the quality of the project and its implications on the building design.

Economy

(138) Establish an attitude toward the initial budget and its influence on the fabric and geometry of the building.

(139) Determine if operating costs are critical issues, and establish a design directive.

(140) Reconcile the possible difference between the initial budget and life cycle costs.

Time

(141) Consider the possible influences of historic surroundings.

(142) Consider which major activities will most likely remain static and fixed and which might be dynamic and flexible.

(143) Consider the implications of change and growth on long-range performance.

Programming Activities

There are different programming applications with many variations, and different degrees of sophistication within each application reflect the increasing complexity of building projects. Note that each of the four degrees described in this section builds on the experience of the previous degree and on the basic principles and elementary techniques of the first degree of sophistication.

There is a close relationship between the degrees of programming and the variable conditions under which services must be provided. Programmers must learn to make adjustments and modifications to the typical programming activities without having to invent a new programming method.

A beginner in programming must also learn not to be perplexed by the complexity of a project. The final section describes how this method, the considerations, and the client decisions can bring order and simplification to any design problem.

Typical Programming Activities

The typical programming activity described here is appropriate for medium-sized projects. Small- and large-sized projects would require adjustments in this approach. Each project schedule involves management decisions that will determine how concurrent or how sequential the programming activities will be. In order for these activities to be understood more clearly, they have been listed in the logical sequence that follows.

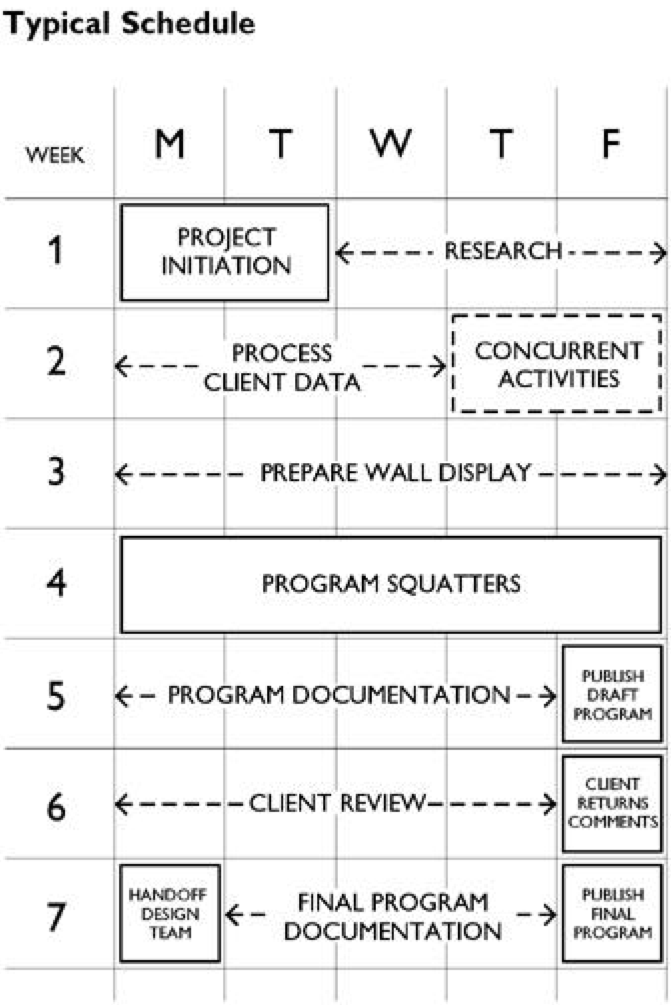
Project Initiation

Before meeting with the client for the first time, the lead programmer organizes the project team and assigns tasks according to the work plan. The team may consist of a lead programmer, an assistant programmer, a project manager, and, sometimes, a specialist for a particular building type. A work plan includes a tentative time schedule and defines activities, deliverables, and the team members assigned to complete them. The team prepares a list of the initial data needed from the client.

The project team goes to the client's premises for an organization meeting. One of the main reasons for the meeting is to identify the decision makers. It can be assumed that those people who have the responsibility and accountability for the product have the authority for decision making. The client/owner is usually identified as the main decision maker; however, the client/user and governmental agencies influence decisions.

Since project goals can determine the nature of the data to be gathered, it is prudent to elicit an initial set of goals from the owner and senior management—before they get down to details. This is also the time to explain the programming process and schedule of activities, including critical meeting dates and times. It is prudent to verify the client's expectations for the content and organization of the final report, as well as coordinate the use of computer applications for the whole team.

Typical Schedule



Typical Schedule

WEEK	M	T	W	T	F
1	PROJECT INITIATION		←---- RESEARCH ----→		
2	←--- PROCESS CLIENT DATA ---→			CONCURRENT ACTIVITIES	
3	←------ PREPARE WALL DISPLAY ------→				
4	PROGRAM SQUATTERS				
5	←– PROGRAM DOCUMENTATION –→				PUBLISH DRAFT PROGRAM
6	←------ CLIENT REVIEW------→				CLIENT RETURNS COMMENTS
7	HANDOFF DESIGN TEAM	←– FINAL PROGRAM DOCUMENTATION –→			PUBLISH FINAL PROGRAM

Information Request

This is the time to obtain data from existing records and to obtain the project's maximum capacity and the personnel forecast. Data may come from a variety of sources, including human resources, accounting/payroll, group manager, facilities department. For educational clients, for example, data may come from the enrollment, or scheduling offices, deans, or principals. The project manager seeks to obtain the site survey and the soils analysis, as well as plans of the existing facilities.

Optimally, the data will be available electronically. Find the proper channels to make the data transferable and readable. This is a critical time to coordinate the compatibility of computer applications between the project team and the client team.

Once the client/manager has been designated, he or she is asked to distribute the questionnaire to the users with instructions for their return at a certain time. The questionnaire serves to identify the type of information and the level of detail to be discussed at the interview. In some cases, consider an orientation meeting with the client/user representatives.

Concurrent Activities

Several concurrent activities need to take place sometime during the second week: the site analysis, the tour of existing or similar facilities, and the work of the client/manager. The client/manager assigns a workroom, selects the users to be interviewed, and prepares a schedule for the interviews during the squatters' week.

Office Preparation

Back at the office, the programmers research the pertinent building type, user characteristics, and area parameters. They contact cost estimators for cost data at various construction quality levels.

When the users' questionnaires are received, they are processed and tabulated. All the data received from the client is analyzed and interpreted into useful information. The data is organized and classified through the use of the Information Index.

Once the information arrives back from the client, determine: Is the data up to date? Is it complete and consistent? If new data is necessary, are there adequate resources to collect and process it in a timely manner?

Often, the information required resides in several places within an organization, and the programmer must reconcile the information received. For example, a facilities department has an accurate list of existing spaces with specific labels for each workspace, but these units and spatial labels may not coincide with those provided by accounting to report on people, because accounting uses full-time equivalents rather than space units. Computer applications help to sort the information received and find the missing or disparate components.

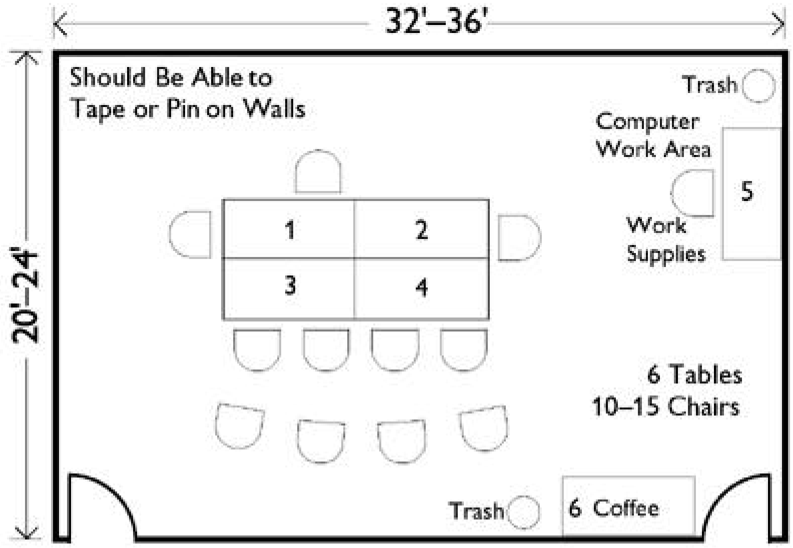
It usually takes five working days to collect the background information and prepare the wall display. The programmers compile and produce the initial space requirements graphically on brown sheets, and prepare a skeletal set of analysis cards around the initial goals, researched facts, and obvious concepts. A review of the Information Index will indicate missing information and questions to be asked during the squatters' interviews. A trial run on balancing the total budget is useful at this time. The project manager may prepare a preliminary project delivery schedule as well.

Programming Squatters

The squatters' technique solves a communication problem with clients at a long distance from the office. Setting up an "office," practically in view of the site and on the client's premises, is certainly a good solution. The users and the owners are easily available for interaction and decision making. Working efficiency is achieved by isolating the team members from the office telephone and other projects. In this way, they can concentrate on the task at hand.

A programming squatters follows a well-thought-out agenda. It begins on Monday morning with setting up the workroom. The most important feature of the workroom is plenty of wall space for pinning up displays.

Squatters' Workroom

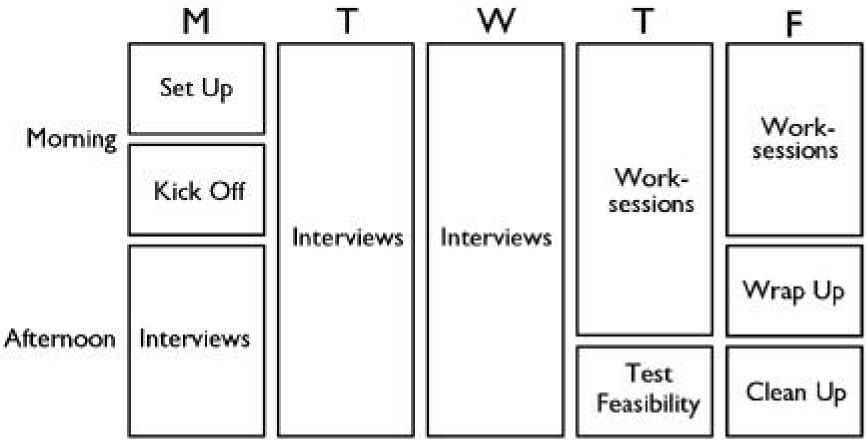


The programming team holds a kickoff meeting for key participants as a group. This session involves an explanation of the programming process, the schedule of activities, and an overview of the status of the project at that time. The participants are told what the interviewer needs to know from them and by when.

A programming squatters proceeds through Wednesday with interviews of individual client/user groups. Most interviews can be accomplished within a period of one hour. The schedule should provide an hour's break between interviews to allow the transcription of rough notes to analysis cards. Each user group reviews its previously submitted "want list" and modifies it realistically on the brown sheets. The programmer uses interviews to further clarify the responses to the questionnaire and to confirm the programmer's conclusions.

Using the Information Index, the interviewer can pursue the uncovering of new data. Here, one must act as a catalyst for decision making. One may present alternatives or evaluate gains and risks to stimulate a decision. Details concerning minor equipment are documented but postponed for use in design development.

Squatters' Agenda



The client/user group might emphasize specific objectives and functional relationships, as well as the physical and psychological environment.

Interviews with the client/owner and management staff are a good source for project and operational goals and overall concepts. This group is concerned with organization, finances, change, and cost and quality control.

Interviews depend on the amount and kind of client participation. With or without interviews, it is difficult to avoid worksessions. On Thursday, the programmers consolidate and display all the information reviewed over the past three days. The display of information may take the form of feedback to the client. In effect, the display indicates what the programmer perceives to be the important and pertinent information. The programmer, then, asks the client for confirmation and for decisions in the case of conflicting information, or the programmer may identify issues and ask for their resolution.

Technology allows real-time output of information and becomes essential during the worksessions. Provide a work area for a portable computer and printer, perhaps with telephone access to the Internet. A dedicated team member must keep up with the information changes during the interviews, point out missing information when appropriate, and contribute summary reports to be used in the brown sheet discussions as changes are made.

But the most critical function of the worksessions is to balance the total budget with the space requirements and the quality of construction. Graphic -analysis cards and brown sheets are used as working tools to determine the space program and to balance the budget. Electronic spreadsheets can be particularly useful during management presentations. It is possible to use them to help the client make decisions, by creating alternatives that can be weighted even sometimes as the client speaks about them. The use of electronic projectors directly from a lap -

top to a screen makes the spreadsheets available to a large audience. But translating the spreadsheets to a graphic format, such as that of brown sheets, is important to emphasize the relative size of spaces requested.

A preliminary cost estimate analysis is presented toward the end of Thursday to key client decision makers, to determine the project feasibility.

Often, the user requests (the wants) are more than are possible within the budget. It is important, then, to set priorities, to consider alternatives, and to make decisions about the project scope. After this session, it may be necessary to meet again with the individual groups to adjust the requirements. Friday morning is reserved for this purpose and for preparing the final presentation.

Early Friday afternoon, the wrap-up presentation is made to all participants as a group, and a preliminary approval of the program as it stands is requested. The squatters' week ends with the cleaning of the workroom and packing to go back to the office.

Program Documentation

The report for formal approval need not be more than photocopy reductions of the analysis cards and brown sheets together with enough text to explain the total program. This could be done within a standard report outline based on the programming steps, or the team prepares a more elaborate and refined report. The programming team submits this draft program to the client for review and formal approval.

Approval and Handoff

With such intensive client participation, formal approval is not difficult. The team incorporates the client's review comments into the wall display and into the report. It is essential to present the wall display to the design team since the information is usually encoded. The graphic analysis and the concise nature of the program, together with the verbal presentation, make it possible for the design team to assimilate what would have been a complex program.

The programmer, then, helps the designer to state the problem by flagging the information perceived to be a potential form giver.

The statement of the problem is added to the wall display and to the final report. All that remains, then, is to reprint and distribute the final report to the client and the design team members. However, it is the wall display, and not the bound report, that communicates the information to the design team.

Project Closeout

The programming team closes out the assignment by archiving the reports and wall display in the library, by entering the references into a document index, and by placing the critical electronic files in the deliverable folder.

Outline of Typical Programming Activities

A. **Project Initiation**
   1. Office organization
      A. Organize project team
      B. Prepare workplan
      C. Prepare list of data needs
      D. Establish computer applications and file sharing protocols
      E. Set up project directory
      F. Set up project Web site
   2. Organizational meeting with client/manager
      A. Identify client decision makers
      B. Elicit initial set of goals from owner/senior management
      C. Schedule client/user for programming squatters' interviews and worksessions
      D. Obtain data from existing records
      E. Obtain capacity/staff requirements
      F. Obtain site survey and soil analysis
      G. Obtain plans of existing facilities
      H. Arrange for distribution/return of questionnaire to users (if required)
   3. Orientation meeting with client/user representatives (optional)

B. **Concurrent Activities**
   1. Conduct site analysis
   2. Tour existing and/or similar facilities
   3. Have client/manager arrange participants for squatters' week interviews and worksessions
   4. Arrange through the client/manager for squatters' workroom near users and site
   5. Collect user questionnaires

C. **Office Preparation**
   1. Research building type/client
   2. Research cost data and area parameters
   3. Process and tabulate users' questionnaire
   4. Analyze data received from the client
   5. Prepare wall display:
      a. Present initial space requirements on brown sheets
      b. Draw initial-analysis cards
   6. Prepare squatters' interview questions

D. **Programming Squatters**
   1. Set up workroom and wall display
   2. Hold kickoff meeting with users as group
      a. Explain approach
      b. Explain what the interviewer needs to know and by when

3. Main body of interviews
    a. With client/user groups
        (1) Collect specific data
        (2) Test documented information on wall display
        (3) Plan for next level of detail
    b. With client owner/management
        (1) Confirm previous data
        (2) Reveal new data
4. Conduct worksessions
    a. Report implications of information to client for confirmation
    b. Identify conflicts needing reconciliation
    c. Identify issues yet to be resolved
    d. Test feasibility of project
        (1) Balance total budget with space requirements and quality of construction
        (2) Consider alternatives that result in balanced budget
    e. Make final revisions
5. Hold wrap-up meeting with client/owner and users as group
    a. Present wall display resulting from week's activities
    b. Receive informal approval of program
6. Clean up workroom, and pack up to go back to office

E. **Program Documentation**
1. Follow standard outline
2. Photocopy and reduce analysis cards for draft program
3. Submit draft program to client for formal approval

F. **Approval and Handoff**
1. Receive client review comments
2. Obtain client approval of program
3. Correct wall display and report
4. Present wall display to design team
5. Write problem statements with designers
6. Reprint and distribute final report

G. **Project Closeout**
1. Place wall display and report in archive library
2. Update document index
3. Place electronic file in deliverable folder on the common server

Four Degrees of Sophistication

The development of programming has been cumulative through four degrees of sophistication. This development is the result of many years in the professional field, working with clients in a wide variety of situations. The identification of the four degrees is empirical and well tested.

The problem-seeking method involving the five-step process and four basic considerations is applicable to all four degrees. In the fourth degree, the four basic considerations are expanded to five to include the political considerations in urban problems.

First Degree

First-degree programming consists largely of the **traditional architectural services** in which the architect merely organizes the information received from the client, adds the information on the site analysis, and tests the simple economic feasibility of the project. The information is sufficient to formulate the statement of the problem.

The **two-phase process** provides the appropriate information for the two phases of the design process: schematic design and design development. First-generation programming leads to the design of a **simple, perhaps single, building**—usually a **familiar building type.**

If the programmer is inexperienced in the client's building type, he or she needs to obtain a background through **library research**, a survey of similar projects, etc. This background will improve communication with the client and understanding of the nature of the problem.

Decision making is **centralized** in the client/owner who is also the user. With a **simple client structure**, the client is an active, working member of the team. As a result, the **client/user participates** through the process. A wall display with brown sheets and analysis cards, supported by spreadsheet and word-processing applications, are the primary techniques involved.

Second Degree

The expanded scope of second-degree programming takes advantage of computer applications to process large amounts of data as a tool that reinforces the architect in problem seeking. These extended computer applications include **spreadsheets or databases** for:

(1) Generating space requirements

(2) Manipulating the space inventory

(3) Analyzing functional affinities

(4) Calculating economic information

(5) Analyzing programmatic options

The **two-phase process** may become a **three-phase process** on projects that require a master-planning phase, as well as schematic design and design development phases. The idea of providing the appropriate information for each phase still applies.

In second-degree programming, the architect begins to provide **consulting services** to lead the client through the decision-making process. The architect takes the leadership to develop the program and provides most of the information through extensive interviewing, statistical analysis, and long-range projections.

Goal setting and the resolution of conflicting values are time-consuming, but extremely important, aspects of programming at this level. This had best be left to the **specialists** who have the experience in the building type and the social and political awareness to communicate effectively with the **complex client organization.**

Second-degree programming deals with a **complex building group.** The architect must be "specialized" in the building type, with extensive experience and **bench-marking databases** as a background for space parameters and workloads. The architect's experience will be useful in testing functional and organizational relationships and concepts and in understanding the implications of the client's organizational structure.

The programming team becomes more **interdisciplinary.** Specialists are needed to deal with problems in analysis, and with complex functional organizational requirements.

The client is still the final authority in decision making. Characteristically, the client in this level is a **multi-headed group** in which the owner is not necessarily the user. The user group may be composed of several groups with conflicting interests.

Third Degree

At this level, programming is still aimed at facilities design; however, there are generally **many preprogramming issues** that must be resolved before a design program can be developed. The analysis includes a survey of existing operational and functional plans dealing with the management activities concerned with efficient operation and the social and functional organization of an institution or organization.

The **management of the project team** and client organization becomes a major aspect at this level—the organization of work, the logistics of trips, the preparation of presentation material, and the timing of critical decisions to permit work to progress without recycling.

This level deals with **extremely large, mixed-use projects,** such as an entire industrial community, a military community, or a university city. The projects involve a full spectrum of building types within a comprehensive master plan. This level of programming will probably remain the exclusive domain of the large, highly specialized practice of multicompany, joint-venture organizations.

The program development requires an extensive background of experience from a **variety of consultants** and volumes of **detailed documentation** to justify and support every decision and recommendation made by the architect and the consultants.

One important characteristic of programming at this level (beyond the size of the project) is the total leadership of the architect to develop the program without the involvement of the client organization, or with minimum involvement at best.

There is likely to be a very **complex administrative organization** between the client owner and the architect that processes approvals. Yet, high-level decisions tend to be **autocratic,** whether by corporation presidents or governmental executives. The **user group may, or may not, be available to the process**. Still, the architect has to create a model of the user organization and a profile of the characteristics of the user. To link a large team working in multiple locations, **electronic-presentation technology** is useful for large group meetings, along with **electronic mail** and **Web-based publishing.**

Fourth Degree

This degree is involved with **urban-planning problems,** and, therefore, the major considerations of function, form, economy, and time are expanded to include the **political consideration.** Involvement by the architect/planning consultant is at the **bureaucratic** level, where planning problems are commingled with political issues and power struggles.

Fourth-degree programming deals with a whole series of loosely connected problems in **urban development.** These problems are **not always facilities oriented.**

Typical of these problems might be publicly financed projects in which the planning and design of facilities is a secondary issue to the larger issues of land location and use, financing, and public acceptance.

Research must be extensive enough for the recommendations to withstand **public scrutiny**. The architect/planner who wishes to serve in this environment must cope with all of the issues surrounding the project. He or she must seek alternatives and strategies.

This level of programming is an area for **specialty firms** of all sizes involving all types of publicly funded building projects and for architects with a strong sense of public service and a high tolerance for the bureaucratic process.

The Information Index is expanded to accommodate **political motivation**. This should indicate that decision making may put all logic aside for public image and expediency. The structure of this complex client would indicate more conflicting values, longer funding schedules, and public presentations involving **advocacy groups and bureaucratic organizations**.

Summary

The following table makes use of key phrases to describe the four degrees. It may be clear that, essentially, the different degrees depend on the levels of complexity of the problems and the client structure, and on the team and services required to deal with them.

Characteristics	First Degree	Second Degree
Phase of Process	Two phase	Two or three phase
Services Provided	Traditional	Consulting
Building Type	Familiar	Building-type specialists
Scope	Simple, single building	Complex building types
Project Team	Single team	Interdisciplinary team
Client Organization	Simple client structure	Complex client organization
Decision Making	Centralized	Multihead client
User Participation	Client/owner/user Involvement	Conflictive user groups
Research	Background research is project focused	Benchmarking data on building typ
Computer Applications	Word-processing and spreadsheets	Extensive use of spreadsheets/databases

Variable Conditions

In approaching a project, the programmer must be able to identify those conditions that will determine the scope of programming services required, as well as the techniques to be used. Different situations call for different responses. The following list might help to identify those conditions:

1. Identify the type of problem.

It makes a difference if the programmer is defining a rationalization, conceptual shift, or a strategic problem.

A rationalization problem emphasizes the Facts and Needs steps and seeks justification for the area requirements requested and budgeted. Generally, departmental managers must sign-off on the requested allocations.

A conceptual-shift problem involves the search for new ideas and emphasizes the Goals and Concepts steps. Good ideas occur throughout an organization, so this type of problem tends to use a highly participatory process often organized on a focus group basis.

A strategic analysis involves all steps in the process at a broad level of detail. The purpose is to clarify thinking about a problem and participation tends to be on a "need-to-know" basis.

2. Identify the type of program.

It makes a difference if the program is for a site master plan, building design, or interior design.

The sources of information vary: Board of Trustees' policies for master planning, management decisions for schematic design, and detailed room-by-room user requirements for design development.

3. Define the level of detail required.

It makes a difference if programming is in two phases: (1) for schematic design, and (2) for design development; or if programming is in three phases: (1) for master planning, (2) for schematic design, and (3) for design development.

It is a matter of level of details. Programming for master planning can be based on crude figures and rough information that must be refined for schematic design and further refined for design development. It is like going from a reducing glass to a magnifying glass. The most efficient process collects the appropriate level of information for the analysis required.

4. Determine if specific information will become obsolete at time of use.

It makes a difference if the conditions call for tightly tailored requirements or loosely fit requirements.

In the first instance, the building will work well initially and, thereafter, it must be altered to fit changing conditions. In the second instance, the building works in a spacious fashion, but the loose fit postpones initial alterations.

5. Quantify the degree of participation.

It makes a difference if the client is essentially one person or a group of persons.

To identify the participants ask: Who are the decision makers? Who must buy into decisions? Who knows the information? Who needs to receive the information?

6. Establish the attitude about participation.

It makes a difference if the client is willing to participate in the process or if the client relies on the programmer and consultants for specific recommendations.

The client's reliance on proposals and recommendations places a heavy responsibility on the programmer and consultants to do research and comparative analyses to justify each recommendation.

7. Indicate the level of decision making.

It makes a difference if the decision making is centralized or decentralized.

When the decision making is decentralized, the programmer faces the most serious challenge to reconcile the different points of view through documentation and graphic-analysis techniques. When the decision making is centralized, the programmer must seek out the decision maker and interview that person as early as possible. An important decision maker may be protected by a large staff from such interviews; yet, the staff may easily misread his or her intentions.

8. Indicate the availability and validity of existing information.

It makes a difference whether the information is handed to the programmer by the client and the consultant, or whether the information is generated by the client and the programmer.

In the first instance, the information is likely to be incomplete; few consultants would provide site and budget analyses. Even fewer would provide a reasonable building efficiency. In the second instance, it is the programmer's responsibility to see that the information is complete and predictively reasonable.

9. Determine the user and quality of deliverable.

It makes a difference if the programming report is a working document for the project team, or if a refined document with computer graphics and additional narrative is required for third-party use.

A working document requires copying the analysis cards with supplemental pages of text and numerical tables. It takes more time and resources to publish a refined document when using desktop publishing and when the client requires electronic files of the document.

10. Define the size and type of facility.

It makes a difference if the building type has unique requirements, such as a nuclear-science center might have had in 1958, or if it is a familiar building type.

With a familiar type, it is possible to use space parameters derived from past experience. With a unique type, the programmer is more dependent on background research and the user for space parameters.

11. Identify expected durations and key dates.

It makes a difference if the process is on a conventional schedule or on a concurrent schedule.

Concurrent (known as fast-track) scheduling requires that some decisions be made sooner, that the money be locked in earlier, that the space program be looser, and that the predictive parameters be shorter and more general. The overall amount of time in programming is the same as for conventional scheduling but, for concurrent scheduling, the initial programming period is shorter and requires more experienced programmers.

12. Determine if the client has a fixed budget.

It makes a difference if there is an established limit to the client's available funds or if the funds required are undetermined.

Actually, every client's budget has a limit. Sooner or later this limit becomes evident. An open-ended budget implies carte-blanche freedom; however, it merely postpones the balancing of the budget. In either case, an early trial-run cost estimate can be used to advantage in approximating the inevitable fixed budget.

13. Determine if a cost estimator is required.

It makes a difference if the cost and quality of construction are based on general experience (cost, location, time, quality), such as $30/GSF, $50/GSF, or $100/GSF, or if these are dependent upon performance specifications breaking unit costs into subsystems.

When the definition of performance specifications for building systems occurs during programming, the cost estimate is more precise and more time-consuming. For technical buildings or for renovation projects involving building condition assessments, a cost-estimating specialist is often part of the programming team.

How to Simplify Design Problems

Some architectural design problems are quite simple and familiar. They are easy to manage. On the other hand, there are those architectural design problems that are indeed complex and unique. These problems must be simplified and clarified before they become manageable.

There is no need to panic; start in an organized manner. Use the Information Index or just the basic framework of steps and considerations. If you start with the recommended method of inquiry—the five-step process—you won't lose time thrashing aimlessly. You will know what the end product will be: the statement of the problem. And it is when the problem is complex and unique that analysis is really effective in clarification. Use the four considerations as the major classifications of information; they are the components of the whole problem.

Undoubtedly, there are many ways to make a design problem manageable. Clients must be stimulated intellectually to make sound decisions at the right time. Sound decisions are needed to simplify the problem. Good communication techniques and graphic analysis certainly help. Take a look at the three ways that follow, and note how they might help to simplify a design problem.

1. Use the Five-Step Process.

a. To collect information and determine its validity—separating fact from fantasy by identifying the inter-relationships of information in the different steps.

b. To spot pertinent information—by testing goals and concepts for design implications that might qualify them as part of the design problem.

c. To process voluminous facts into useful concise information—by determining the bare implications of data, what it means.

d. To analyze a client's preconceived solution, to pinpoint the actual requirement—by tracing the solution back to a programmatic concept and even back to a goal.

e. To focus on information critical to schematic design—by filtering out information more appropriate to routine engineering or to design development.

f. To distinguish major concepts from minor details—going from the general to the particular.

g. To organize the information for cooperative evaluation, consensus and decision-making—to be able to trace the resultant Needs back to Goals, Facts and Concepts.

h. To lead to a clear statement of the problem—by seeking the essence, recognizing the obvious, and discovering the uniqueness of the problem.

i. To guide individual members of the project team toward a unified effort.

2. Use the Four Major Considerations and Their Subcategories.

a. To search for enough information to provide a clear, well-rounded perception.

b. To classify the wide range of factors that constitute the whole problem.

c. To concentrate on the whole problem without excluding the major factors.

d. To analyze the whole problem—to identify the subcategories as subproblems and to understand their interrelationships.

e. To analyze the subproblems separately within the limits of their interrelationships.

f. To focus on the elements of an architectural design problem as opposed to some other kind of problem outside the grasp of control.

3. Stimulate Client Decision Making.

a. To establish the program requirements.

b. To reduce the number of unknowns.

c. To provide more complete information.

d. To limit the number of alternative design solutions to those responding to the design problem.

Useful Techniques

Information is a basic element in programming. Facts and ideas, conditions and decisions, statistics and estimates—all these and many more constitute the information needs. This section emphasizes communication techniques—how to facilitate the transfer of information. This section also involves feedback and feedforward of information as well as covers the processing of data into information. The section ends with a technique to evaluate the programming package and a technique for building evaluation.

Questionnaires

Questionnaires can be an integral part of background research; however, they can provide only a part of the data required for a successful project. The extent of their value must be understood, and a programmer must use them judiciously and intelligently.

To be successful, questionnaires must be well thought out, consciously and carefully designed for a specific audience, and aimed like a rifle shot, not broadcast like a shotgun blast.

A questionnaire or survey form is often the first impression an architect makes on his or her client and the facility users. Since questionnaires can help or hurt the architect's reputation in the client's eyes, they must be designed and used carefully. When designing a questionnaire, consider these guidelines:

1. Determine the data that is needed and the best way to get it. Ask these questions:

   What is needed?

   Who probably has it?

   How should the question be asked and answered?

   What is the best vehicle for asking it?

2. Consider two or more questionnaires: one for executives (broad, strategic, and qualitative), and one for users (focused, operational, functional, and quantitative).

3. Customize each questionnaire to gather the right data from the right people.

4. Strive for legibility, clarity, and simplicity.

5. Use filled-in sample responses—include examples of the types of responses.

6. Provide clear directions—do not assume the reader has done this before.

7. Create the shortest and most specific form possible—people are busy, and your questionnaire is just one more unscheduled task for them.

8. Provide enough space for responses/answers.

9. Test the newly designed questionnaire with colleagues before you distribute it.

10. Use the most efficient delivery method available to ensure a faster response—electronic versus hard-copy distribution.

Questionnaire Use

The use of questionnaires can be a valid method of gathering data before the programming squatters. Questionnaires are very useful for collecting existing and proposed personnel, space, and vehicular requirements. The data will be tabulated by organizational or functional group, so organizational charts are very useful in creating your questionnaires. Analyze questionnaires prior to onsite worksessions with the client.

The scope of the project dictates how detailed the quantitative information request should be. For a small program, a programmer could have personnel lists by name. For large projects, however, involving even millions of square feet, the requests for information vary. In some cases, given a known building type, a generic program with predictive area parameters for the forecasted personnel may suffice. In others, detailed departmental listings may include space requirements for specific and unique functions. Adjust your information request according to the scope of each project.

Electronic Questionnaires

The intranet of an organization can provide a quick and effective tool for conducting questionnaire surveys. Electronic questionnaires may simply be an electronic-mail message with the questionnaire in an electronic-file format that the respondent can access, complete, and return using the mail system. These electronic responses can then be collated into meaningful data in a manner not unlike the collation of hard-copy responses. In situations where the number of respondents is very large and the information requested is clearly quantified, a Web-based questionnaire may be used. Web-based questionnaires reside on a Web site where the respondents can access them and fill in requested information. The Web site is ideally linked to a database where responses are automatically collected and summarized into predetermined categories.

While using electronic questionnaires:

1. Insure that all respondents have the necessary access, software, and skill to use the electronic questionnaire.

2. Test the use of the questionnaire with one of the client team members before general distribution.

3. Allow options where qualitative information and comments may be captured in addition to form-based quantitative information.

4. Test responses for completeness, accuracy, and interpretive errors.

5. Web-based responses are best managed through multiple-choice answers where the respondent does not have much leeway to make interpretive mistakes.

6. Explain the intent behind the questions in an easy-to-access help facility.

7. Manage electronic responses and data in an organized and easily retrievable form.

8. Always back up data.

Data Management

Programming steps are alternately qualitative and quantitative. Goals, Concepts, and the Problem Statement steps are essentially qualitative. Facts and Needs steps are essentially quantitative. Computer programs offer a variety of functions that can help in the **management and analysis** of data both quantitative and qualitative. While computers are typically used to analyze quantitative information, the programmer should use computer capabilities to complement the qualitative and interactive nature of a wall display or electronic presentation using the computer. Knowledge of computer-based applications is an integral part of today's programming process.

Data Sources

An organized programmer seeks to minimize the time and effort spent in data collection and input while maximizing the time available for analysis. However, it is extremely important that valid and clean data be used as a base for all analysis. It is much easier to set up correctly in the first place than to correct a wrong setup later. To this end, discover the most appropriate data sources and allocate adequate resources for data entry and cleanup. Data from various sources may be available in hard-copy or electronic formats. Electronic data is often easier to set up and link. However received, the programmer should validate the data received through the interviews and worksessions with the client/user.

Hard-Copy Formats
Questionnaire Responses
Organization Charts
Existing Facility Plan/Area Takeoffs
Existing Facility Walk-Through Notes
Interview/Analysis Cards
Space Allocation Standards

Electronic Formats
Electronic-Questionnaire Response
Human Resources (HR) Database
Facilities/Real Estate Database
Computer-Aided Facility Management (CAFM)
  Database
Computer-Aided Design (CAD) Drawings
Room Utilization Logs

The Structuring of Data

The purpose of data organization is to manage information received from the users such that it is analyzed and outputted in a manner that is both useful for client decision making and for the design of the building. While the process of information collection and analysis is often iterative, it is very useful to spend some time at the outset of a project to determine the possible information categories and analysis types that may be required during the programming process. The early determination of a potential data structure assists in efficient collection and analysis of information.

At the outset, set up consistent nomenclature and classification schemes to organize the data as it is collected. During this process, an understanding of the limitations of various information sources helps to create a data structure at a level of detail that optimizes the requirements of the project and the time available for data collection. Avoid duplication and collection of redundant information.

Electronic Information Management

Establish a minimum level of team skill set and software proficiency as the first step to efficient information management and exchange. Decide the following:

1. Compatibility of computer applications
2. Methods of file exchange
3. Consistent file-naming conventions
4. Accessible file storage locations
5. Protocols for saving original and updated files
6. Ways of exchanging information:
   - LAN/WAN-based sharing
   - Internet—E-mail
   - Internet—Web/FTP Site

Selecting a Data Processing Application

To analyze quantitative data, use spreadsheets or databases. Each processing method has unique sorting, summarizing, and reporting capabilities that makes one more appropriate than the other, depending on the project.

Characteristics of a database application

Database applications are very fast with large data sets. They are designed for continuous use, maintenance, and feedback. They can be used by several programmers/clients at once. The critical data is kept in a central source. They integrate better into an organization’s information systems’ landscape. Reporting is repetitive, static, and less interactive than that of spreadsheets. All data is in separate, distinct tables, unlike that in spreadsheets. Relevant items are “linked” (relational database). Ideally, set-up requires the design of input and output formats prior to collecting data.

Characteristics of a spreadsheet application

A spreadsheet preferably has fewer than 1,000 records in the data set (e.g., employees or spaces). The spreadsheet works well when reporting is changing and moving. It allows interaction while collecting the data but a highly customized, one-time answer. The results can then be used to make a decision, rather than to maintain the data collected. Electronic data is not easily accessible.

Consider a database over a spreadsheet when:

There are more than about 1,000 records in a data set (e.g., employees or spaces). There is detailed data available from electronic sources (CAD, HR databases).

Customization versus reusablity

Reusing templates and boilerplates provides opportunities for saving time. However, problem seeking works

Outline for Structuring Data

By Organizational Hierarchy

1. Enterprise
2. Division
3. Department
4. Section
5. Subsection

By Activity

1. Working Hours
2. Frequency of Visitors/Customers
3. Frequency and Duration of Meetings
4. Different Work Tasks/Settings

By Time Period

1. Existing—Actual
2. Present Period—Requirement
3. Move-in—Requirement
4. Long-Range—Requirement
5. Ultimate—Requirement/Site Capacity

By Location

1. Country/Region
2. Complex (Site)
3. Building
4. Floor
5. Zone

By Space Type

1. **Office Space**

2. **Office Support**
   Conference (can be assigned to
   Central Services)
   Departmental Storage
   Departmental Files/Libraries
   Departmental Equipment Rooms: Copiers, etc.

3. **Central Services**
   Cafeteria
   Vending
   Classrooms
   Break Areas
   Locker Rooms
   Central Library
   Mail/Reproduction
   Security
   Medical
   Lobbies
   Auditorium
   Special Events
   Facility Maintenance Operations

4. **Specialty Facilities**
   Laboratories
   Classrooms
   Other Special Functions

best with unique, unfamiliar, and complex design problems. Most often, projects radically change from client to client and scope to scope even for the same building type. Usually customization of electronic templates is required.

In general, a programmer can reuse the questionnaire templates by reformatting the appropriate sections. The spreadsheet templates created for a given program type tend to follow the same internal logic. But time will be needed to input all the information in the appropriate categories, and steps to create customized reports may vary. Templates of the boilerplate text explaining the methodology and the components of the report are helpful. Caution is essential to avoid using proprietary client data.

Develop new systems when time for this has been included in the project. But understand, too, that many breakthroughs in the use of technology for programming occur when answering a new question in the midst of a project. Technology introduces creative analytical processes, and, as such, management of time is important.

Output of Information

Programmers process data to provide useful information to the designer and to the client for approval. While the content of the user's information is the same, the format is often different for the purposes of the designer. The user needs information back for validation, and the client needs it for approval. Computer applications allow for quick output of the same information in a number of different formats. Besides the designer and users, data output can also assist in cost estimation and due diligence reports. In any case the programmer must insure that the data output is complemented and validated with the wall display information.

Data output may be a comprehensive, detailed listing of every space and its characteristics, or take the form of summaries. Two common summary forms are:

1. Summaries by space type

These summaries show in relative terms the area allocated to functional types of space, such as conference rooms versus break areas, in relation to the actual workspace. This type of summary may be for an entire campus or a single floor. It provides the designer with a relative listing of the various spaces that are needed while allowing the user to understand the functionality of the proposed facility.

2. Summaries by client organization

These summaries show the total allocated each client group, such as a department or other appropriate business unit, relating personnel to the square feet of each group. Users are often most concerned with these summaries as they represent the amount and cost of space for the particular group. A programmer uses these reports to justify allocations and to plan functional relationships among groups.

Functional Relationship Analysis

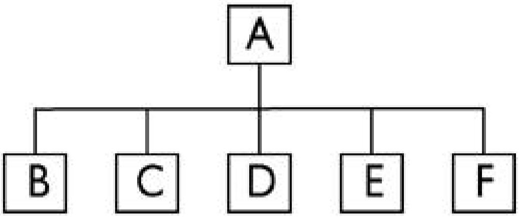
One of the qualitative components of the programming process involves the collection and analysis of organizational structure, concepts, work process, and functional relationships. The purpose of the analysis is to determine the required proximity of the different user groups.

The following are concepts that indicate types of functional relationship requirements:

Flow: The movement of people, material, products, or information from location to location.

Proximity: The shortest distance required among groups to ensure a high degree of communication and interaction and access.

Organization Chart



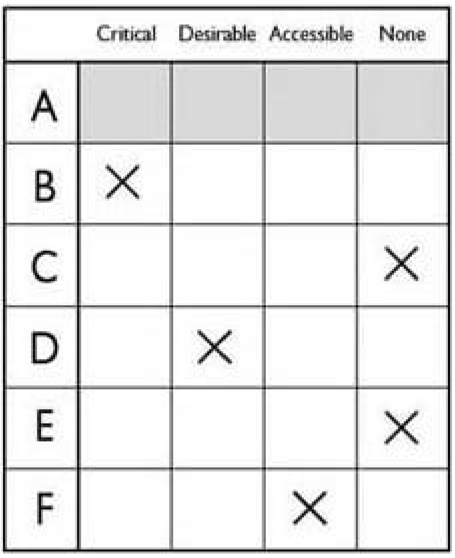
Virtual: An exception to the concept that proximity is necessary to ensure communication, because communication technology provides interface.

While the proximity of people and services is the predominant factor influencing the location of spaces, flow and access to communication networks are often key considerations in building organization and design.

To undertake a functional relationship analysis, begin by collecting the formal organization charts and classifying groups at a consistent level of hierarchy.

A programmer may use a questionnaire to identify the desired proximity among groups. An adjacency chart records the perception of each user group's functional-relationship requirements to all other user groups or among functional areas. In a questionnaire, limit the

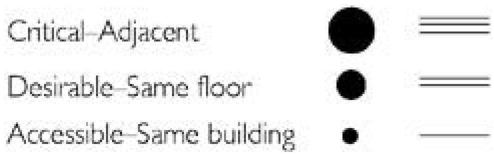
Adjacent Requirements for A



	Critical	Desirable	Accessible	None
A				
B	X			
C				X
D		X		
E				X
F			X	

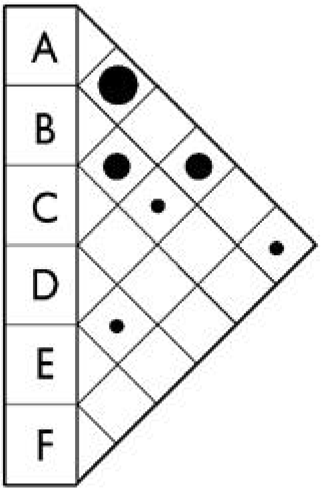
proximity codes to a few choices, such as critical, desirable, and accessible.

It is also important to define what the user may mean by each proximity code, for example:



Next, transpose the questionnaire responses to an interaction matrix. Use size of dots or color coding in an interaction matrix to record adjacency requirements among groups or specific program areas, such as mailroom, loading dock, or laboratory. During the interviews, cross-check questionnaire responses to validate requirements from all groups. During test fits of area allocations for specific loca-

Interaction Matrix

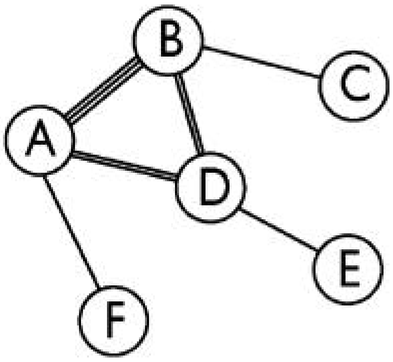


tions, it is also possible to use the matrix to record achieved relationships and score alternative plans.

Once all relationships are checked, create a bubble diagram. A bubble diagram is a simplified graphic description of an organization’s functional relationships. It is useful to record these at two scales:

1. **Micro relationships:** The depiction of individual user groups and their specific relationships. Using questionnaire information and circles to represent groups, prioritize adjacencies with different weight lines to indicate **critical, desirable, and accessible** relationships. You may also indicate flow and access.
2. **Macro relationships:** A diagram summarizing the overall requirements for interaction and communication among all user groups or functional areas.

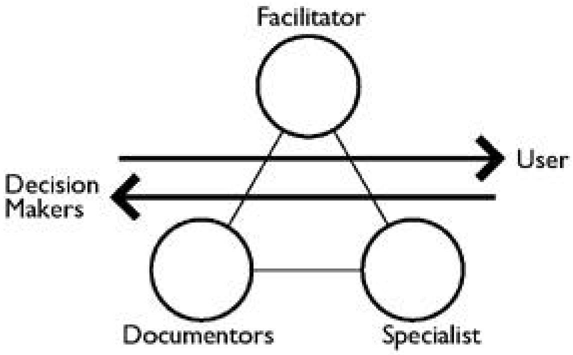
Bubble Diagram



Interviews and Worksessions

The programming process affects two-way communication between the end users and client decision makers through interviews and worksessions. There should be a clear distinction between interviews for data gathering and worksessions for summaries and decision making. Data is gathered as a basis for analysis, calculation, discussion, and decision; and after having its implications determined, it becomes useful information.

The communication role of a programmer (or a programming team) encompasses the subroles of facilitator, documentor, and building type specialist.



TWO-WAY COMMUNICATION

In the role of facilitator, the programmer represents an objective party that guides the processes of inquiry and causes the open exchange of ideas and data among the end users and decision makers.

Experienced facilitators will:

1. Focus discussion on the goals of the project.

2. Ask questions pertinent to the project.

3. Periodically summarize or recap.

4. Continue to return to the main ideas until they are clarified.

5. Remember that clients do not need to tell you *all* they know—only what you need to know!

The documentation role of a programmer is critical to successful communications during the programming process. The documentation of the responses in an interview should include (1) who the respondent was, (2) when the interview took place, and (3) the classification of data according to the Information Index. A verbatim record with opinions and attitudes is not necessary or desirable. However, accuracy and completeness are necessary with the kind of raw data that needs to be processed before yielding significant information.

Data should be documented for continuous team reference. Lost information can lead to wrong conclusions. The recording programmer should know when direct quotations may be desirable as documented data in or-der to clarify opinions and attitudes, goals, and concepts.

Generally, the recorder must extract the essence of a response as he or she records in order to avoid data clog. Further still, having reduced the response to its essence, one might find that a diagram or some other form of graphic representation communicates the response more vividly.

Finally, as building-type specialists, the programming team provides professional expertise to the analysis of user requirements and the implications of programmatic options.

Identifying Decision Makers

It can be assumed that those people who have the responsibility and accountability for the product have the authority for decision making. This assumption would indicate that the client/owner usually be identified as the main decision maker; however, the client/user and governmental agencies influence decisions.

The client/owner might be a corporate or governing board represented by the management group, the senior administrative staff, or an appointed building committee. In many cases the individual who is responsible on the organization chart is not the real decision maker. All too often, it becomes the matter of a guessing game as to who is the final decision maker. Nevertheless, it is important to identify the decision-making structure in each specific situation prior to the interviews and worksessions. Conflicts can be expected to arise in this complex decision-making body. When issues are identified, they should be dealt with privately by the consultant, not in a public hearing where decision makers are exposed.

The client/user might include the mid-management or mid-administrative group and, indeed, the actual or prospective user. Lately groups of interested citizens have joined the client/user group. While this second group is not the final decision-making group, it can *cause*, *influence*, and *recommend* decisions to be made. *Do not expect a group to make decisions on data that is not available to them*.

A third group would include governmental regulating agencies that exercise control of functional requirements, public expenditures, and public safety. These agencies are decision makers on specific issues and need to be identified early.

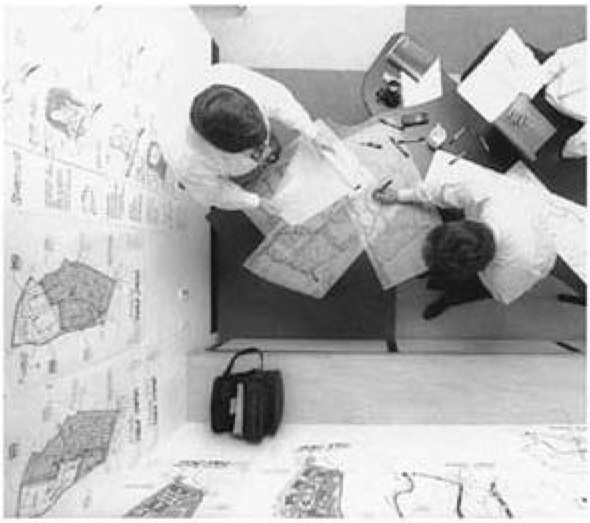
Preplanning Interviews

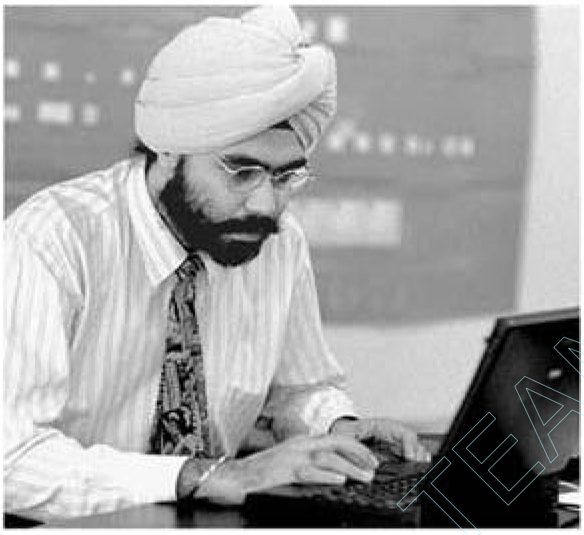
The programmer should not approach the interviews empty handed. Identify conflicting issues that need to be reconciled. Prepare graphic presentations leading to the impartial allocation of space and to sound decisions. Prepare a list of key words to guide an inquiry and discussion.

An important aspect of preplanning is identifying what needs to be gathered through the interview technique. The Information Index is not only a key-word checklist of questions, but also the format for the classification of responses. The client need not be aware of the Information Index, nor should the interview be overly structured. The obvious use of a checklist inhibits responses.

When arranging appointments, it is best to let the people to be interviewed know ahead of time what is to be discussed. This allows them time to prepare and to collect pertinent information they wish to discuss.

A series of interviews is best scheduled by the client/manager, who may have to arrange not only the best appointment time for various individuals, but he may also have to arrange for work substitutes for those individuals.





The programming squatters bring together the client team and the programming team, including special consultants, so that all are aware of decisions regarding the allocation of space and money and the consensus on quality—made within a balanced budget.

Types of Interviews

Interviewing techniques vary with the number and type of participants. Therefore, it would be well to consider four generalized categories:

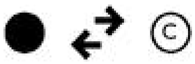
A. Individual interviews

B. All group interviews

C. Medium-group interviews

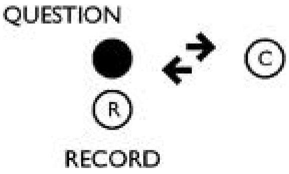
D. Large-group interviews

A1. Individual interviews involve essentially two people: the interviewer and the client respondent (C). The interviewer asks the questions and records the answers. The recording function (R) is the most likely to suffer.

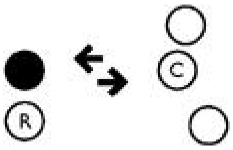


A tape recorder may be used, but there is a chance that it might intimidate the respondent who is reluctant to make commitments. Journalists are specialists at asking questions and recording them. But still, most people are afraid of being misquoted.

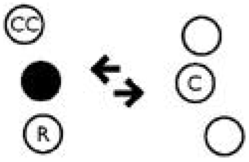
A2. It takes two people to conduct a good interview: one to ask questions, another to record the answers. This frees the interviewer from having to record and allows him or her to pursue questioning with more spontaneity.



B1. Small-group interviews usually involve a client leader in a discipline accompanied by one or two assistants or resource people. For all intents and purposes, the interaction between the interviewer and the leader has all the characteristics of the individual interview.

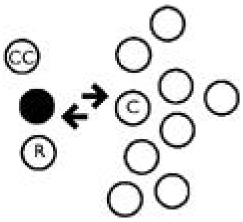


B2. A series of small-group interviews might well include the presence of a client coordinator (CC) who monitors the interview.



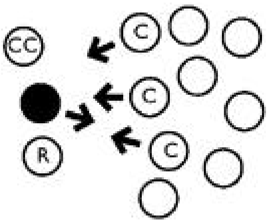
There are many advantages to having a monitor, such as checking the integrity of the answers, gaining valuable insight into opposing points of view, and providing follow-through action some interviews might generate. The main disadvantage is intimidating the respondent.

C1. Medium-group interviews introduce the possibility of single-discipline or multidiscipline groups. A group of six to ten people within the same discipline will most likely have a designated leader who will provide most of the answers. Nevertheless, the democratic process will provide the opportunity for different points of view.

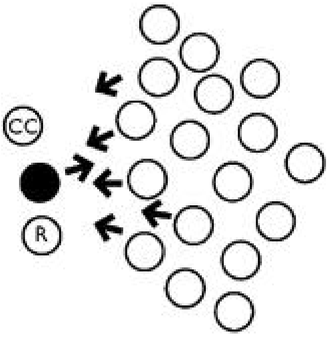


Medium groups require a fairly elaborate initial presentation to serve as a background for the questions to be asked. The presentation might go as far as identifying the issues that must be reconciled or alternatives that call for decisions. These might be used as a frame of reference for the type and pertinence of the data sought.

C2. When a medium group involves several disciplines or subgroups, members of each discipline might rally behind a leader. The multidiscipline aspects emphasize the need for a clear initial presentation or a frame of reference, so that each discipline can express itself on the same issues before launching new ones. To give everyone in a large group the opportunity to participate, rotate those sitting in the front row of seats. This rotation allows time for each discipline to contribute.



D. Large-group interviews involving fifteen or twenty people may be single-discipline or multidiscipline in composition. With these large numbers, only half of them are likely to participate actively and then only through the motivation provided by the interviewer.



A single-discipline group would very likely be headed by a leader. This group might have met previously to discuss the major issues involved in the project.

Large-group interviews require an initial presentation that will inform everyone of the background of the project and the framework for the type of data sought.

Preparation of Brown Sheets

Brown sheets graphically indicate space needs that have been derived from project goals, facts, and concepts. The brown sheets are intended to convey the magnitude of numbers and sizes. A client and a designer can visualize the number and sizes of spaces more easily if they are indicated graphically and to scale. Brown sheets serve well as a graphic technique for comparative analysis of the project's area requirements. One glance can tell where the major allocations of area have been made, the predominance of small spaces requiring a higher percentage of circulation spaces, or the unjustified size of different functional areas.

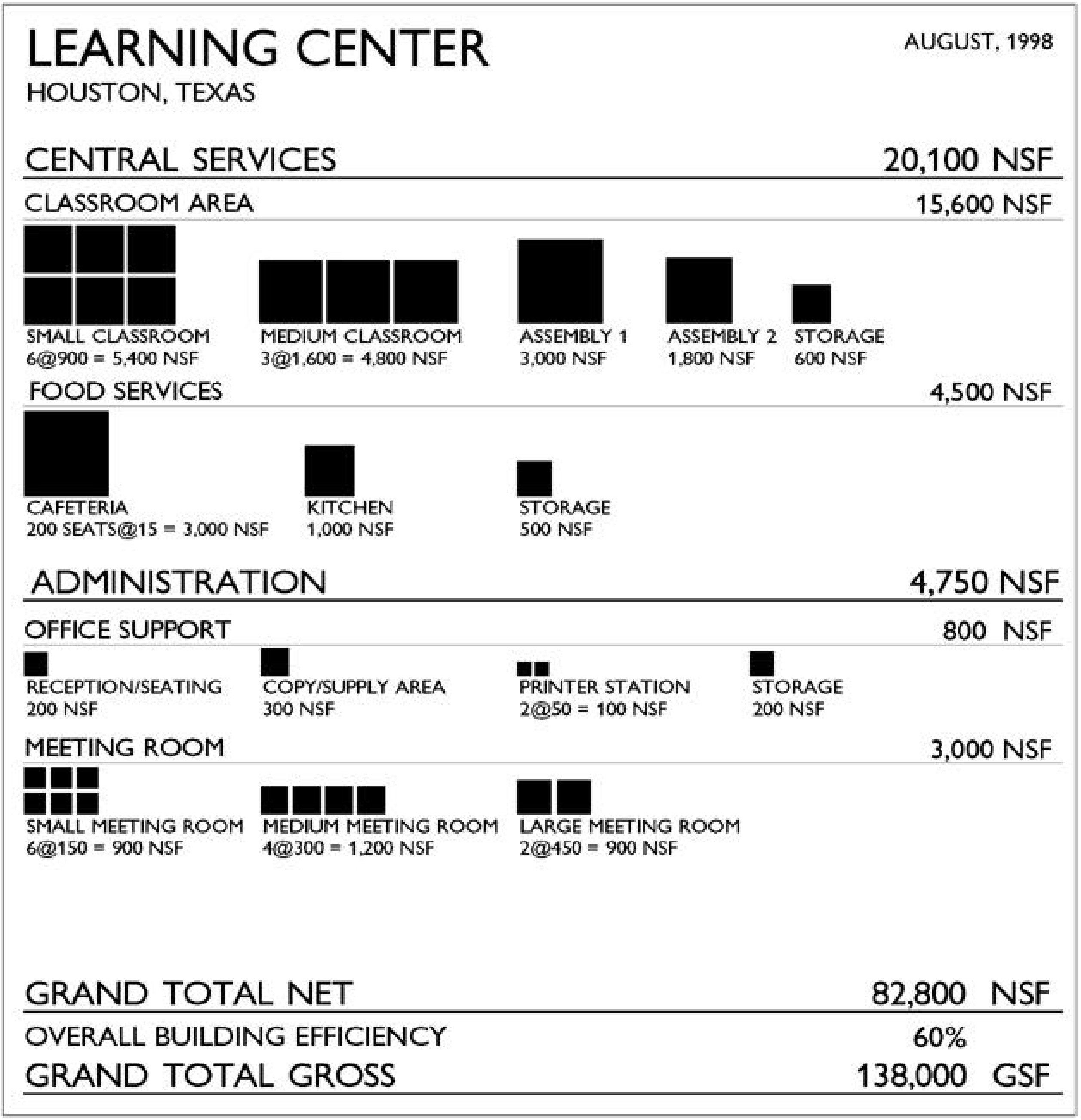
The first purpose of brown sheets is to present the area requirement as determined during the interviews or by some predetermined formula for the impartial allocation of space. For a schematic design program, net assignable areas are shown; however, the client is informed that unless an assigned area is shown on the brown sheets, it is not considered to be an area requirement. This is intended to check and recheck all net area requirements.

The second purpose of brown sheets is to serve as worksheets during worksessions. For that purpose, they are made of informal materials that not only lend themselves to revision, but even invite revision. The feedback to the users starts with the statement, "These are the area requirements you have indicated to us." The confirmation starts with the question, "Are these correct?" And if worksessions on the balancing of the budget indicate reallocations, changes, additions, and subtractions, the brown sheets must be revised on the spot: adding notes, changing figures, and adding or deleting the scaled squares representing areas. The brown sheets displayed on a wall are used to represent the latest revisions and the latest total tally at all times.

Time and again, the brown sheets have proven to be excellent communication devices. The total scope of a project can be communicated through brown sheets to large groups of people, often representing diverse disciplines and agencies in a much more efficient manner than through a typed list of spaces. Changes and revisions made on a set of brown sheets over a period of several days on a master copy are readily available for group display and discussion.

Computer applications allow the updating of spreadsheets and databases containing space lists that correspond with the brown sheets. It is possible to sort this information by client organization or by type of space. It is also possible to use computer applications to plot the sheets, or simply to be a tool for keeping a running tally of the calculations and totals.

The traditional brown sheets, as shown in the accompanying picture, were made from brown paper and white squares. While this technique is still in use, we also see the use of sheets generated by a computer plotter on white paper with contrasting dark squares. Regardless of format, the value of the brown sheets is the ability to perceive all the squares (all the areas) at one glance.



LEARNING CENTER

HOUSTON, TEXAS

AUGUST, 1998

CENTRAL SERVICES 20,100 NSF

CLASSROOM AREA 15,600 NSF

SMALL CLASSROOM
6@900 = 5,400 NSF

MEDIUM CLASSROOM
3@1,600 = 4,800 NSF

ASSEMBLY 1
3,000 NSF

ASSEMBLY 2
1,800 NSF

STORAGE
600 NSF

FOOD SERVICES 4,500 NSF

CAFETERIA
200 SEATS@15 = 3,000 NSF

KITCHEN
1,000 NSF

STORAGE
500 NSF

ADMINISTRATION 4,750 NSF

OFFICE SUPPORT 800 NSF

RECEPTION/SEATING
200 NSF

COPY/SUPPLY AREA
300 NSF

PRINTER STATION
2@50 = 100 NSF

STORAGE
200 NSF

MEETING ROOM 3,000 NSF

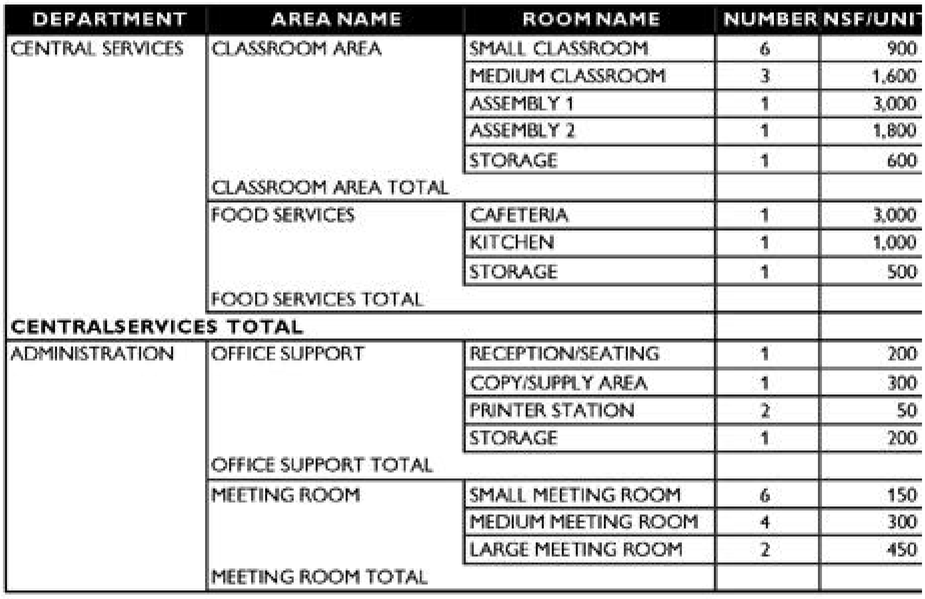
SMALL MEETING ROOM
6@150 = 900 NSF

MEDIUM MEETING ROOM
4@300 = 1,200 NSF

LARGE MEETING ROOM
2@450 = 900 NSF

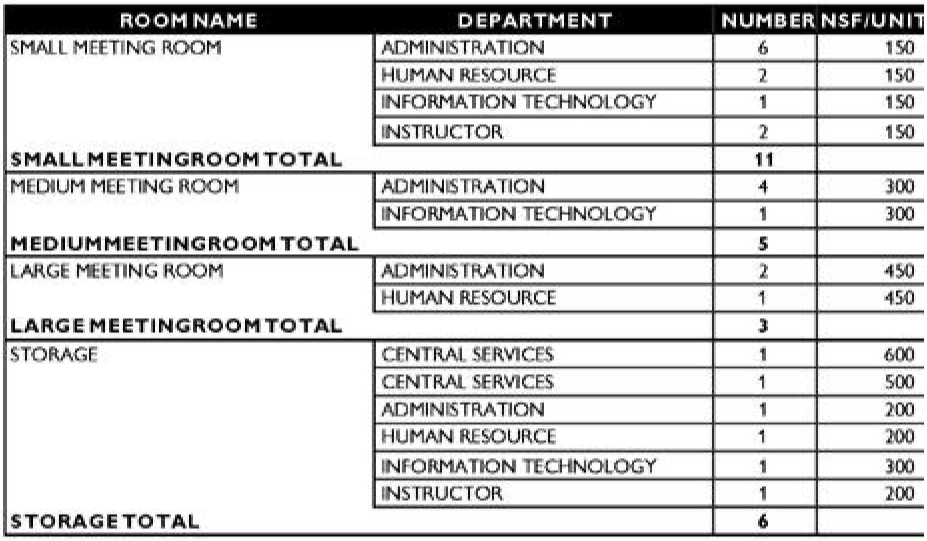
GRAND TOTAL NET	82,800 NSF
OVERALL BUILDING EFFICIENCY	60%
GRAND TOTAL GROSS	138,000 GSF

Space List Sorted by Client Organization



DEPARTMENT	AREA NAME	ROOM NAME	NUMBER	NSF/UNIT
CENTRAL SERVICES	CLASSROOM AREA	SMALL CLASSROOM	6	900
		MEDIUM CLASSROOM	3	1,600
		ASSEMBLY 1	1	3,000
		ASSEMBLY 2	1	1,800
		STORAGE	1	600
	CLASSROOM AREA TOTAL			
	FOOD SERVICES	CAFETERIA	1	3,000
		KITCHEN	1	1,000
		STORAGE	1	500
	FOOD SERVICES TOTAL			
CENTRAL SERVICES TOTAL				
ADMINISTRATION	OFFICE SUPPORT	RECEPTION/SEATING	1	200
		COPY/SUPPLY AREA	1	300
		PRINTER STATION	2	50
		STORAGE	1	200
	OFFICE SUPPORT TOTAL			
	MEETING ROOM	SMALL MEETING ROOM	6	150
		MEDIUM MEETING ROOM	4	300
		LARGE MEETING ROOM	2	450
	MEETING ROOM TOTAL			

Space List Sorted by Space Type



ROOM NAME	DEPARTMENT	NUMBER	NSF/UNIT
SMALL MEETING ROOM	ADMINISTRATION	6	150
	HUMAN RESOURCE	2	150
	INFORMATION TECHNOLOGY	1	150
	INSTRUCTOR	2	150
SMALL MEETING ROOM TOTAL		**11**	
MEDIUM MEETING ROOM	ADMINISTRATION	4	300
	INFORMATION TECHNOLOGY	1	300
MEDIUM MEETING ROOM TOTAL		**5**	
LARGE MEETING ROOM	ADMINISTRATION	2	450
	HUMAN RESOURCE	1	450
LARGE MEETING ROOM TOTAL		**3**	
STORAGE	CENTRAL SERVICES	1	600
	CENTRAL SERVICES	1	500
	ADMINISTRATION	1	200
	HUMAN RESOURCE	1	200
	INFORMATION TECHNOLOGY	1	300
	INSTRUCTOR	1	200
STORAGE TOTAL		**6**	

The Analysis Card Technique

The Analysis Card Technique is a method of recording graphically information intended to be displayed, discussed, discriminated, decided upon, and, sometimes, discarded during the programming phase of a project. This graphic communication technique is also used in the schematic design phase. Selected cards from these two phases can, then, become part of the presentation of the design solution for client approval.

Size and Kind

The size of a card is proportional to the frame of a 35mm slide. The standard 2 × 3 proportion can be expressed in a card 5½ × 8¼-inch or in any other convenient and proportional size. The face of the card has an almost imperceptible, non-photo, blue grid based on .5-cm. The grid is helpful in sketching diagrams, charts, and lettering. However, a white face is all that is required. The card is made of 100-pound pasted Bristol stock.

Wall Display

Use strings of analysis cards and brown sheets to prepare a wall display of the pertinent programming information. Organizing the display according to the five-step process, beginning with Goals, and then Facts, Concepts, Needs. and Problem Statements. It is useful to organize the strings of cards by subcategory. Use header cards with titles to identify the topic of the card string.

Working Advantages

The technique provides the following working advantages during the programming process:

1. The cards are relatively small and easy to handle. They are deliberately kept small to accommodate only one thought or one idea, simply and economically stated. This should encourage a sharp focus on each card. The single thought on a single card encourages easy comprehension. To single out a clear thought and put it in clear, graphic terms is couching basic truths. The cards are small enough to force the avoidance of unnecessary detail. This helps to keep the freshness of a small sketch.

2. The cards may be used freely, sorted, grouped, and sequenced. Their best use is as a wall display—tacked and grouped under the process sequence of Goals, Facts, Concepts, Needs, and Problem Statements. The visual display, together with proper classification, helps to make comparisons easier and to avoid duplications.

3. The cards are ideal for recording information as discussion with the client progresses during a worksession. These can join other cards in the wall display.

4. Typically, interview notes and preprogramming infor-

mation lead to the making of analysis cards. These are displayed and tested during he worksessions. It is a process of feedback and feedforward. "In essence, is this what you said?" and "Good! We'll pass this information on to the designer at the right time."

5. A wall display of analysis cards makes it easy to test the interrelationships among Goals, Facts, and Concepts that lead to Needs, and eventually to the Problem Statement.

6. A wall display of analysis cards shows, in effect, the progress of programming at any point in time. As committees review the cards, they can comment, make additions, and make deletions.

7. A wall display of analysis cards should be seen at a glance or two to represent the first cut toward the essence of the project. (Average display: 150 cards.) Too many cards could mean that it is time to reevaluate, to postpone, or to discard information.

8. A wall display of analysis cards can be presented to any new members of the client team coming aboard and eventually to the design team. The oral presentation can explain the coded nature of the cards, investing their brief graphic messages with potent meaning.

9. Since the cards are proportional to a 35mm slide, they may be photographed and presented to a large audience in slide form. Alternately, they may be presented, one at a time, using an opaque projector. It is also possible to scan the cards into the digital format, and using a computer, display them with an electronic projector.

10. The cards can be photocopied two or three to a page on regular 8½ × 11-inch paper. Grouped in terms of the programming steps, the photocopies can be augmented by the typed backup data placed in an appendix. In this plain format, the programming package can be stored for future reference. In a more explicit format, including captions to represent the original oral explanation, the programming package can be submitted as a report for formal client approval—and it can be used by team members at later stages in the project.

The schematic design team will not need to read the report. They will use the wall display of the original analysis cards. A design team in action must survey and check the information with hardly more than a glance.

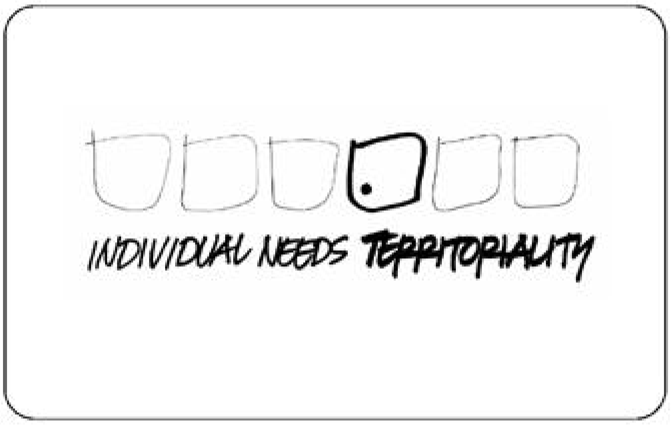
More sophisticated packaging would depend on the large number of copies required for approval and on a specific contract requirement.

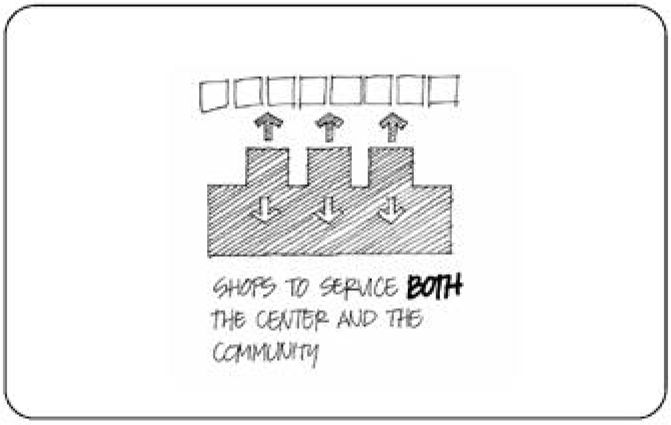
How to Draw an Analysis Card

It takes two related activities to make a good analysis card: thinking and drawing. One needs to think through one's hands. The skill of drawing gives expression, precision, and clarity to one 's thinking.

Here are eight pointers that lead to good analysis cards:

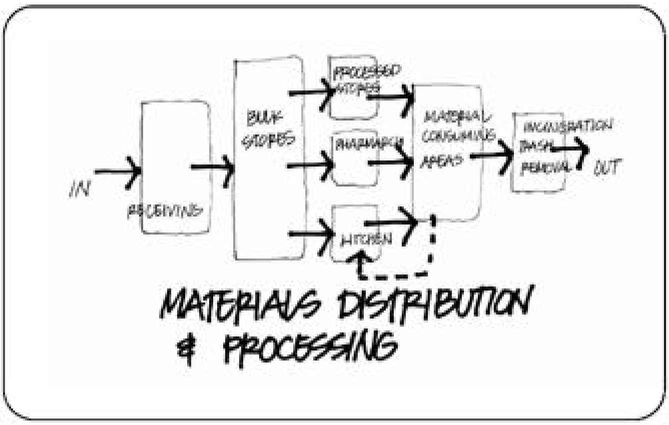
1. Think Your Message Through.

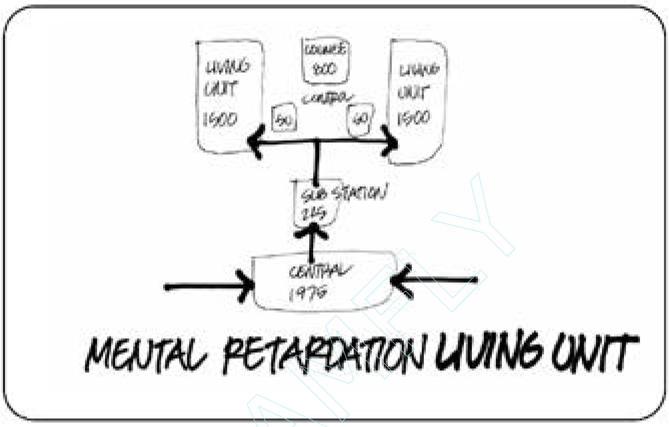




- Deal with it as if it were a telegram. Think what must be said. Reduce it to one thought.
- Put it down graphically, with very few elements.
- Write it out with very few words.
- Add color only for emphasis or for coding.
- (The illustrations represent a 40 percent reduction of the actual card size).

2. Use Visual Images.

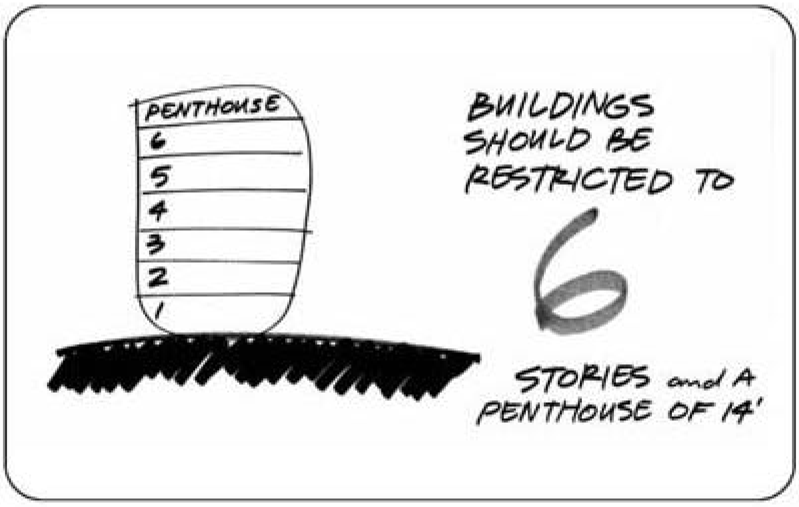


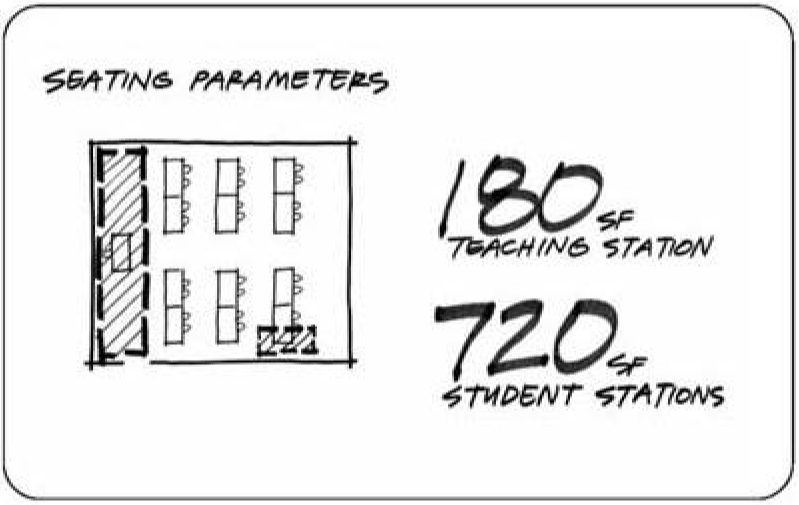


- Use diagrams, symbols, charts, and sketches to aid communication.
- Assume that a visual image is more easily retained than a verbal image.
- Label the parts, and give the card a title.
- A flow chart is understood more quickly than a written description.
- Keep the images simple and specific for clarity, but abstract enough to evoke a range of possibilities.
- Use an appropriate scale for the graphic image to project the magnitude of numbers and the implication of ideas.

Avoid minute detail as it is inappropriate.

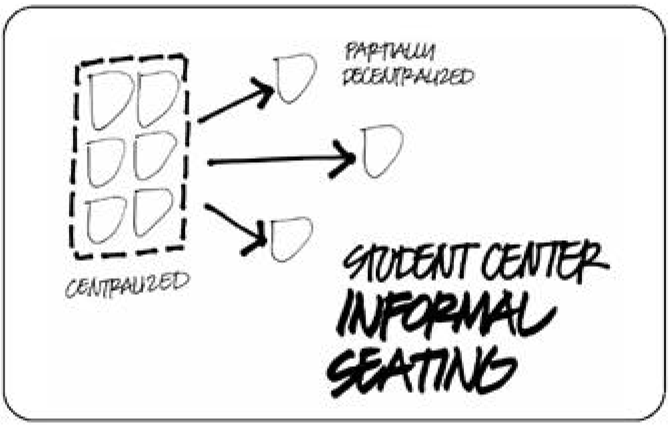
3. Use Very Few Words.

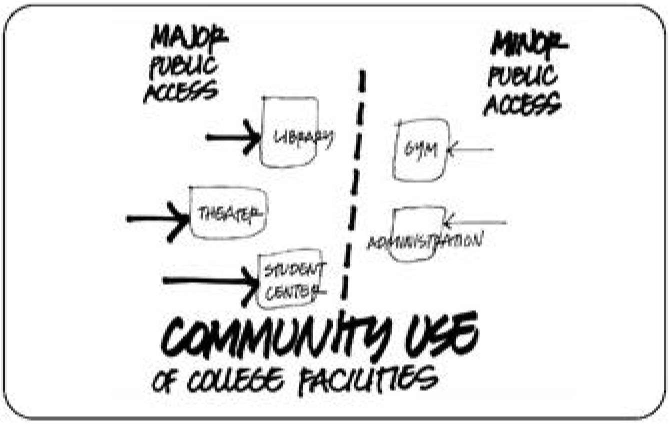

PENTHOUSE
6
5
4
3
2
1
BUILDINGS SHOULD BE RESTRICTED TO
6
STORIES and A PENTHOUSE OF 14'



- Label the drawings properly.
- Reinforce the drawing with a short sentence.
- State the point in as few words as possible.
- Long statements impose small, difficult-to-read lettering on the card.
- Sometimes the critical information is a number.

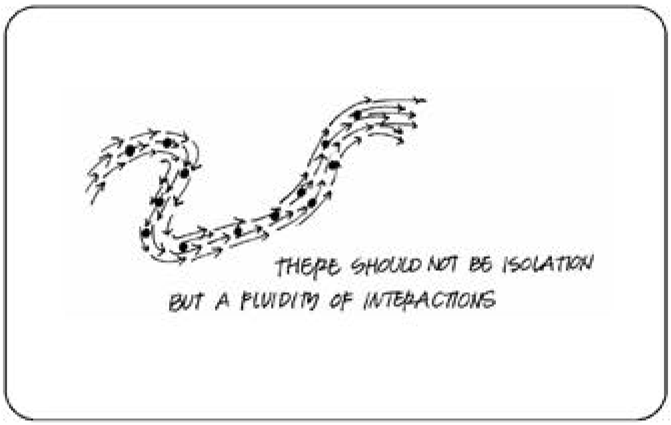
4. Strive for Legibility.

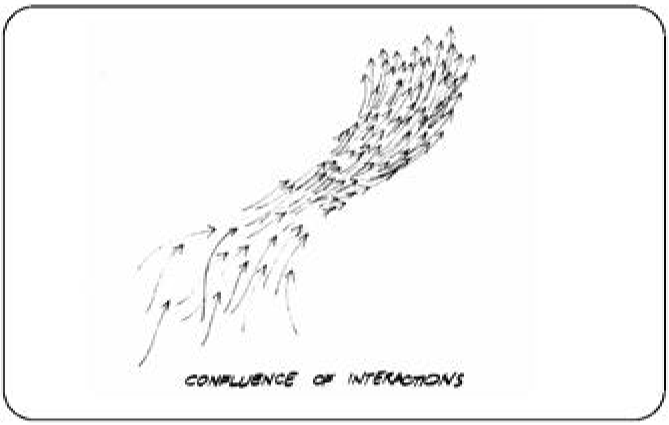




- Legibility is a function of line width and letter height.
- Use letters 1/8-inch high or larger.
- Use a range of pen sizes.
- The use of an opaque projector or slides will not improve illegible lettering.
- Letters on typewritten copy are usually too small and have too thin a stem width.

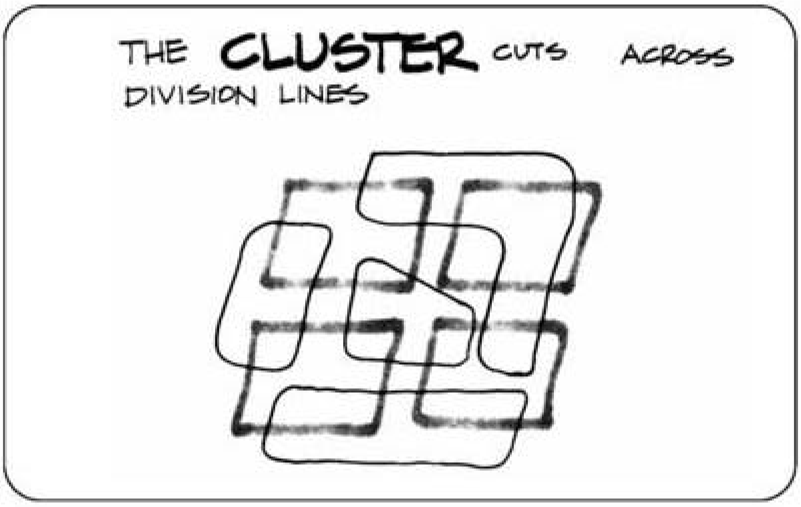
5. Design for Display.

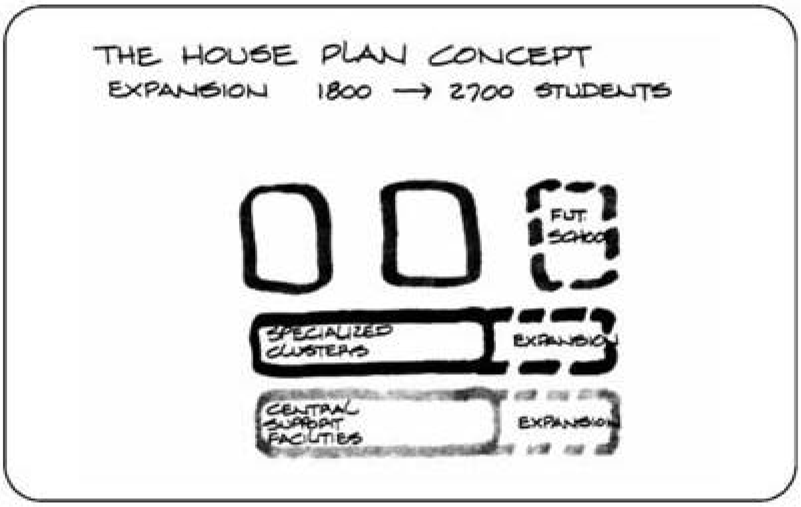




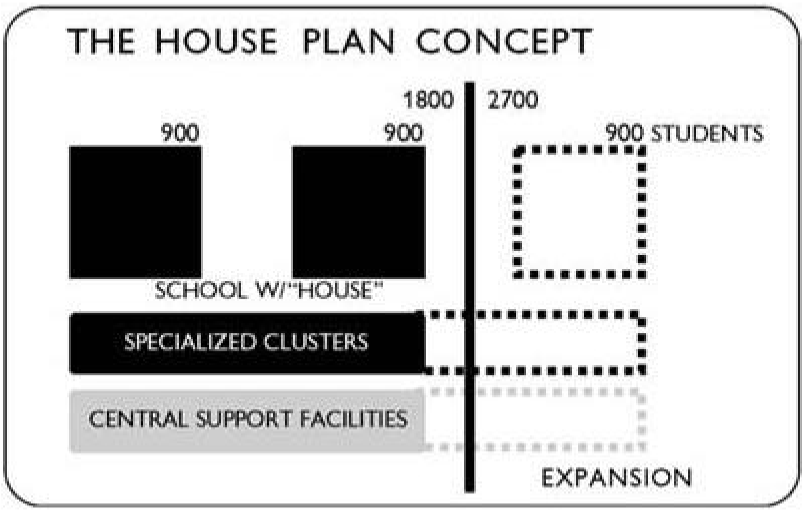
- The difference between analysis cards and book illustrations is in the viewing distance.
- Design analysis cards for a wall display.
- There is a certain look about good analysis cards.
- The bad ones are generally too bold and heavy or too delicate and light.
- If you have to be wrong, be too heavy.
- The two accompanying illustrations are too light for a wall display, but they are excellent book illustrations.

6. Plan for Cards of Different Finish.

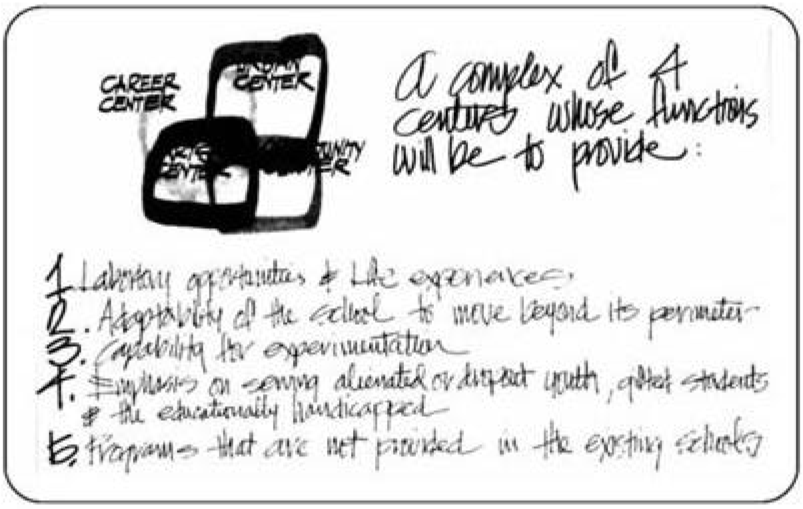




- *"Think"* cards are done quickly by anyone who has a bit of information for consideration.
- *"Working"* cards are sketched carefully enough to clarify the thinking.
- *"Presentation"* cards are meticulously drawn for greater precision. Assign one person to prepare the set for consistency.
- All cards are process documents and as such should have an informal, loose look (as opposed to final documents).

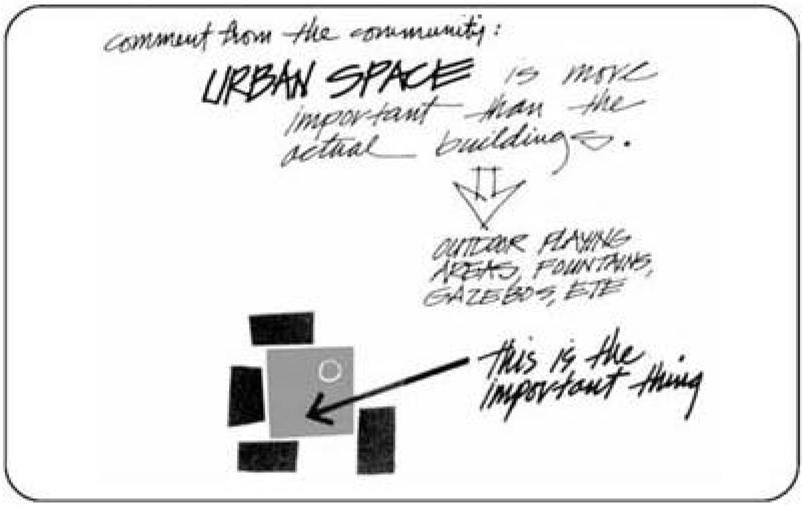


7. Encourage Documentation.



a complex of 4 centers whose functions will be to provide:

1. Laboratory opportunities & Life experiences.
2. Adaptability of the school to move beyond its perimeter
3. Capability for experimentation
4. Emphasis on serving alienated or dropout youth, gifted students & the educationally handicapped
5. Programs that are not provided in the existing schools



comment from the community:

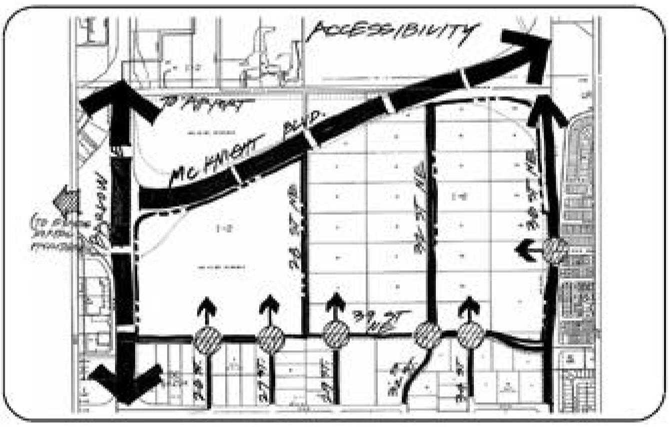
URBAN SPACE is more important than the actual buildings.

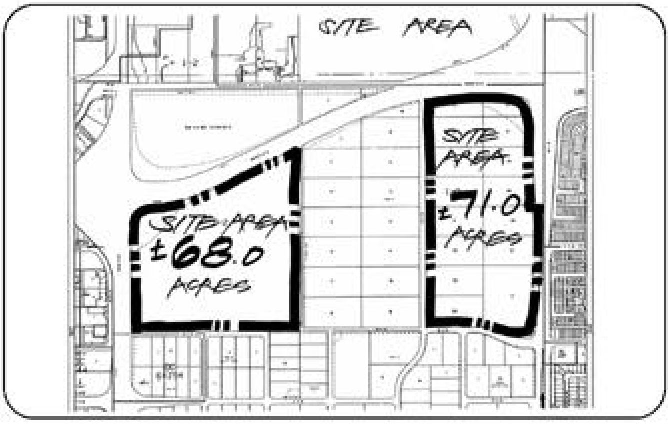
OUTDOOR PLAYING AREAS, FOUNTAINS, GAZEBOS, ETC

this is the important thing

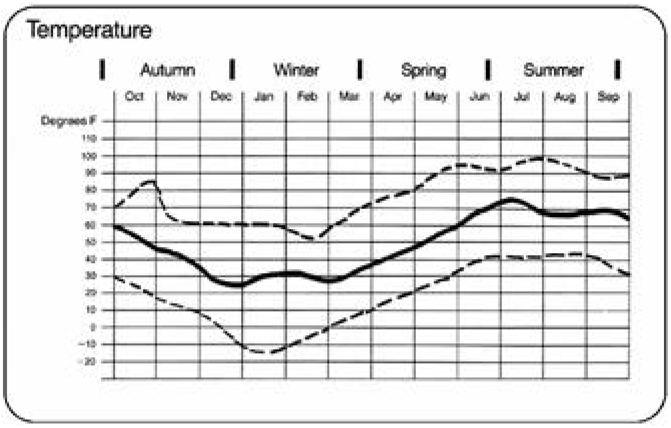
- Encourage everyone on the team to produce the initial analysis cards.
- Remove those inhibitions caused by the high standard of “presentation” cards.
- Promote the production of “think” cards.
- Be concerned with documentation first.
- Evaluate and determine which cards need to be redrawn—later.
- The two accompanying cards document information— too much of it. These cards need to be redrawn and simplified. The information may deserve not one but six separate cards—one thought per card.

8. Preplan “Routine” Cards.





- Order two dozen printed base maps on analysis cards. Document site information to be considered separately on separate cards.
- Document climate data on preprinted cards.
- This is “routine” information.
- If the information is not used in schematic design, it will be used later. The time spent is a matter of minutes.
- But if it is useful, or even a form-giver, the project has gained immeasurably.



Electronic Presentations

Technology extends the communication tools of the presenter. The programmer has access to media technology enabling different types of interaction through electronic connectivity.

Synchronous, face-to-face interaction implies that the presenter and viewers are in the same physical location and interacting in real time. The traditional squatters' worksession is the prime setting for a work in progress. Electronic media (projection of a computer screen) can aid in the review and analysis of data during these sessions, complementing the wall display, particularly during a decision-making period.

Synchronous, virtual interaction implies that the presenter and viewers are interacting in real time but may not be in the same physical location. Communicating with clients in separate locations includes the use of long-distance video and audio conferencing; Internet-based, real-time electronic document sharing, and other virtual technologies. These technologies allow for geographically dispersed teams to share information and mobilize quickly, especially during the organization stage of the project and while the team refines a set of conclusions after the squatters' sessions.

Asynchronous, virtual interaction implies the presenter and viewers are not only in separate physical locations but are also not interacting in real time. A good example is the use of the Internet to present information. The presenter creates a Web-based presentation and communicates to the target audience the location of this material. The viewers are then able to access the presentation at their own convenience and post comments back to the presenter or to each other in a Web-based discussion group. Web pages are also excellent repositories of live information through the posting either of questionnaires for data collection or of project findings and recommendations for the client's information dispersal process to decision makers and end users.

Some of the same pointers suggested to draw good analysis cards apply to the design of electronic presentations using computer applications:

1. Reduce the message of each frame or slide to one thought. A picture is worth a thousand words.

2. Use visual images and diagrams to aid communication.

3. Keep the ideas simple and specific for clarity.

4. View the presentation with a large screen before delivering it in public.

5. Options, such as black-and-white reproducibility, may have an impact on the design of the presentation. Preprint the document to test readability.

6. Consider the file size for electronic transfer. Photographs and diagrams can greatly increase the size of files. Large files take longer to transmit.

7. Make the presentation interactive whenever possible.

8. Use standard templates and consistent symbols.

Analysis
A systematic process of aski
An analysis problem i
Leads to a nt of the
HELLMUTH, OBATA + KASSABAUM

Outline for Report

Title Page

0.0 Preface

Purpose

Organization of Report

Participants

1.0 Executive Summary

2.0 Goals

Function
Economy
Form
Time

3.0 Facts

Summary of Statistical Projections

Staffing Requirements

User Description

Evaluation of Existing Facilities

Site Analysis:

Urban Context
Catchment Area
Vicinity Land Use
Accessibility
Walking Distances
Traffic Volume
Existing Structures
Views From/To Site
Location
Site Size/Configuration
Topography
Tree Cover
Buildable Areas
Land Acquisition Potential

Climate Analysis

Zoning Regulations

Code Survey

Cost Parameters

4.0 Concepts

Organizational Structure

Functional Relationships

Priorities

Operational Concepts

5.0 Needs

Area Requirements Summary

By Organizational Unit
By Space Type
By Project Phasing

Detailed Area Requirements

Outdoor-Space Requirements

Parking Requirements

Land Use Requirements

Budget Estimate Analysis

Project Delivery Schedule

6.0 Problem Statements

Function
Form

Economy Time

A.0 Appendix

Issue-Tracking Sheet

Detailed Statistical Data

Workload and Space Projection Methods

Existing Building Space Inventory

Departmental Evaluations

Outline for Programming Reports

Often, the client or a funding agency requires a report for formal approval. The report could amount to no more than photocopies of the analysis cards and photo reductions of the brown sheets together with enough text to explain the total program. This working document could be done within a standard report outline.

The report could also be a very elaborate document intended to be approved by many agencies concerned with many different levels of detail. In this case, one may seek approval on a format to make program evaluations and approvals comparatively easy for those many agencies.

When publishing a refined document, establish word-processing templates and style guides for consistency of format. Especially when a multidisciplinary team writes sections of the report, coordinate the use of computer applications among the project team members and with the client.

A standard outline based on the programming steps has the advantage of easily accommodating subject matter that has already been classified according to the steps; these steps become chapters in the report. Preventing overlapping among chapters, then, is not a problem. The problem becomes often what to leave out. Use an appendix for supplemental data.

The appendix should contain the bulky, statistical data and detailed information that the programmer used to reach conclusions in the main body of the report. The location of details in the appendix tends to improve the readability of the report.

A primary purpose for a program is the client’s review and formal approval. Some clients require signature approvals to indicate acceptance of the program as a basis for design. The preface of the report might contain the following purpose statement:

> The purpose of this program is to convey an understanding of the problem prior to its solution. This document serves as a record of the decision-making process and is for agreement and approval.

The designer does not write the Problem Statements until the client approves the program. These statements, however, are presented to the client as the beginning of the schematic design.

Develop a library to store and retain program reports and wall displays. A document library is a great resource for background research on building types. A comparative analysis of each program provides a basis for identifying recurring client Goals and Concepts. Furthermore, a comparative analysis of Facts and Needs reveals guidelines for parameters for size spaces, establishing ranges of functional adequacy, and typical allocations for the budget estimate.

Use indexing tools and coded filing/shelving systems to assure that you can retrieve the documents. For electronic files, set guidelines for the naming of files and a standard directory structure for storing them.

When a client requests electronic transfer of reports, provide documentation of the file structure, and indicate the application and version of the software required to use the files.

Evaluation Technique

What is quality evaluation?

It is the evaluation of the degree of excellence of the programming package (the product, not the process).

The evaluation of products should be measured in terms of Function, Form, Economy, and Time. The real value of process is found in the quality of the product.

Why do we need to quantify quality?

Most people like to quantify things. We ask such questions as: “What’s the score?” and “What grades did you make?” A symbol, such as “score,” is a good way to immediately perceive a situation. For that reason, we need to quantify quality—to have a “score.”

We know all the reasons why we should not quantify quality, too—it is subjective, it is based on a value judgment that is different in every individual. It is not scientifically accurate, and so on. Nevertheless, everyone, particularly users, judges our buildings—the ultimate products of our services. That is primarily why we are interested in evaluating our own intermediate products.

Throughout the course of a project, we need to check on its quality and to see if we can improve the project during the “next step.” We need to know what we have after we complete the project to see if it measures up to predetermined quality goals, and to see if we can improve the “next” project.

We need to evaluate a project at every stage in the total design process—starting with programming. For now, the evaluation of the finished building is another matter—requiring a different question set.

How can we quantify quality?

There are many ways. Here is one way. The method consists of three factors:

1. Using question sets as evaluative criteria.

2. Scoring on the basis of the whole problem—not just function.

3. Arriving at a single figure called “quality quotient,” which recognizes the strengths of Function, Form, Economy, and Time, and the equilibrium of the four.

How are value measurements made?

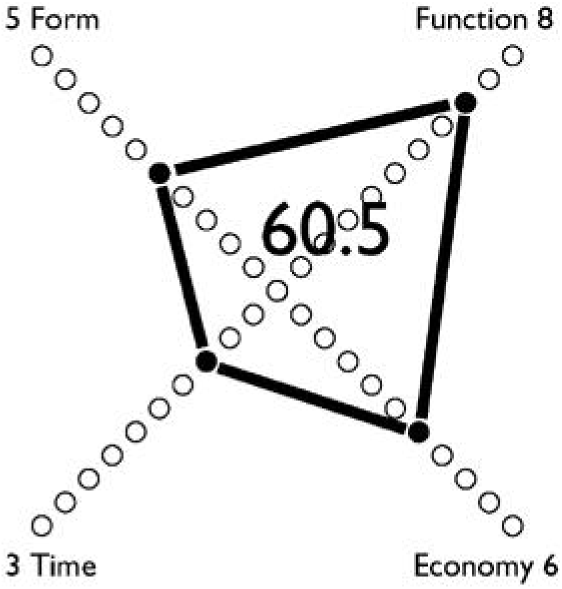
The whole problem concerns the equilibrium of the forces of Function, Form, Economy, and Time—the four forces that shape every product. Equally important as the equilibrium of these forces, however, is the magnitude of each force.

The magnitude of each force can be determined empirically with the following value measurement scale:

Complete Failure	1
Critically Bad	2
Far Below Acceptable	3
Poor	4
Acceptable	5
Good	6
Very Good	7
Excellent	8
Superior	9
Perfect	10

To aid in determining accurate values for each of the four forces, we have developed question sets. By our using the same value-measurement scale to respond to individual questions covering each of the four categories, the final values can be determined more easily. The final value for each category does not necessarily have to be the numerical average of the individual question responses, but the numerical average helps to understand how the final value was determined. The area of the quadrilateral formed by the final values of the four forces yields the quality quotient.

For example, the illustration shows a quadrilateral formed by the following values: Function 8, Form 5, Economy 6, and Time 3. We can assume that these values represent the numerical averages of the responses to the five questions in each category. The area of the quadrilateral can be determined by the following formula:



Graphic Analysis of Quality Quotient

Area= .5 (Function + Time) (Form + Economy) =
= .5 (8 + 3) (5 + 6) = 60.5

Question Sets

The only difference between the accompanying two question sets is the format. The full-sentence question set is intended for those people without experience in its use. After using it several times, a person could change to the key word question set—an abbreviated form with implied wording. For example: “Organizational concept meaning the big functional idea.” and “Functional goals and relationships meaning convenient and efficient operations.”

Full-Sentence Question Set For Programming

Function

A. To what extent have *organizational concepts* been uncovered?

B. How well documented are the client's *functional relationships* and goals?

C. How much discrimination has been used to distinguish between important *form givers* and details?

D. How *realistic* are the *space requirements* based on statistical projections, client needs, and building efficiency?

E. How well identified are the *user's characteristics* and needs?

Form

F. How clearly expressed are the client's *form goals?*

G. To what degree was *rapport* established with the client and the design team on *quality* as the cost per square foot?

H. How thoroughly are the *site* and *climate data* analyzed and documented?

I. To what extent has the *surrounding* neighborhood been analyzed for its social, historical, and aesthetic implications?

J. To what extent have *psychological environment* concepts been uncovered?

Economy

K. To what extent are the client's *economic goals* and *budget limitations* defined?

L. How well documented is the *local cost data* considering methods of financing, planning, and construction?

M. How well documented are the factors of climate and activities considering *maintenance* and *operation* costs?

N. How comprehensive and realistic is the *cost estimate analysis?*

O. To what extent have *economy concepts* been uncovered?

Time

P. To what extent does the program consider *historical preservation* and *cultural values?*

Q. To what degree have *major activities* been identified as *static* or *dynamic?*

R. To what extent does the program anticipate the effects of *change and growth?*

S. How well has the time factor been utilized to *escalate costs* and *determine phasing?*

T. How realistic is the *time schedule* for the *total project delivery?*

Key Word Question Set For Programming

Function

A. *Organizational Concept*
(the big functional idea)

B. *Functional Goals and Relationships*
(convenient and efficient operations)

C. *Form Givers versus Details*
(avoiding information clog)

D. *Realistic Space Requirements*
(statistical projections, client needs, and building efficiency)

E. *Users' Characteristics and Needs*
(physical, social, emotional, and mental)

Form

F. *Client's Form Goals*
(attitudes, policies, prejudices, and taboos)

G. *Rapport on Quality*
(quality versus space, quality as cost per S.F.)

H. *Site and Climate Data*
(physical and legal analysis)

I. *Surrounding Neighborhood*
(social, historical, and aesthetic implications)

J. *Psychological Environment*
(order, unity, variety, orientation, and scale)

Economy

K. *Economic Goals*
(budget limitations)

L. *Local Cost Data*
(local index and labor market)

M. *Maintenance/Operation Costs*
(factors of climate and activities)

N. *Cost Estimate Analysis*
(balanced initial budget)

O. *Economy Concepts*
(multifunction and maximum effect)

Time

P. *Historical Preservation and Cultural Values*
(evaluating significance and continuity)

Q. *Static or Dynamic Activities*

(fixed and tailored or flexible and negotiable spaces)

R. *Anticipated Change and Growth*
(effects of time)

S. *Cost Escalation/Phasing*
(effects of time on cost and construction)

T. *Project Schedule*
(realistic delivery)

Evaluating Facilities

Evaluating facilities is different from facility programming. The former is feedback to design; the latter is feedforward to design. Both are needed to improve the quality of the design product.

Evaluating facilities involves a systematic assessment by an evaluation team. The objectives are twofold:

1. To detect, observe, and report accurately on existing conditions and changes from the original intent as represented by the program.

2. To modify programmatic factors and design criteria, to recommend corrective actions, and to state lessons learned for programming, designing, building, and managing buildings.

The most common application is evaluating the performance of a facility once it is occupied—a postoccupancy evaluation (POE). Then the evaluation team can consider responses from facility users. After solving the shakedown problems and after the novelty has worn off, the first major performance evaluation should take place between six months and two years after occupancy.

Five Steps and Four Considerations

There are many evaluation methods, each suited to a particular application. Some are rigorous and strive for objectivity; others must provide expedient answers and are more subjective. This method is pragmatic—it is comprehensive, yet simplified enough for practice.

The process has five steps:

1. Establish the **purpose.**

2. Collect and analyze **quantitative** information.

3. Identify and examine **qualitative** information.

4. Make an **assessment.**

5. State the **lessons learned.**

The process is general enough to be suitable for many types of facilities. The content makes the evaluation specific. For the evaluation of building performance, it is important to address four major considerations:

Function

Form

Economy

Time

Like programming, evaluating involves an organized process of inquiry, which is comprehensive in content. The organization of an evaluation (feedback) corresponds to the framework used in programming (feedforward). The similarity of organization, content, and format increases the usefulness of the results.

1. Purpose

It is essential that everyone involved has a clear understanding of why the evaluation is being undertaken. While there are several reasons for conducting a POE, establish these at the initiation meeting.

An evaluation may serve many purposes:

To *justify* actions and expenditures.

To *measure* design quality (conformance to requirements).

To *fine-tune* a facility.

To *adjust* a repetitive program.

To *research* people/environment relationships.

To *test* the application of new ideas.

2. Quantitative Description

The second step, preparing a quantitative description, includes collecting factual data on the building as designed; for example, the floor plan. Analyzing parametric data provides a basis for comparing this facility with similar ones.

Functional Adequacy. A measure of the amount of area per the facility's primary unit of capacity. Example: gross area per seat in an auditorium.

Space Adequacy. The gross area of a building is the sum of the net assigned area and the unassignable area. The ratio of net assigned area to the gross building area measures the efficiency of the floor plan layout.

Construction Quality. The unit cost associated with the quality level of the building measured as the building cost per gross area.

Technical Adequacy. The cost of fixed and special equipment, such as stage equipment in an auditorium. Measured as a percentage of the building cost, though it is also possible to represent as a unit cost.

Energy Performance. A measure of the amount of energy per gross square foot consumed for the standard operation of a building.

User Satisfaction. Obtaining some form of a reading on how satisfied users are with the facility.

3. Qualitative Description

A qualitative description includes examining the client's goals for the facility, the programmatic and design concepts for achieving those goals, and the statements representing design problems that the designer intended to solve. This step also includes identifying both changes that have taken place since occupancy and current issues confronting the occupants and owner:

Goals. These convey the client's stated intentions. Sometimes clients express great aspirations that are not fully achievable in the end.

Concepts. These are ideas for realizing goals. Programming concepts represent abstract relationships and functional arrangements. Design concepts are physical responses providing a unifying theme to the solution.

Problem Statements. These represent a recognition of the critical project conditions, as well as a direction for the design effort.

Changes. These are indicators, since occupancy, of new requirements, or inadequacies. Changes are actions taken to alleviate undesirable conditions.

Issues are unsettled and controversial decisions that are in dispute. They are posed by the occupants or the owner of the facility or raised by the evaluation team.

4. Assessment

The assessment requires interpretation and judgment by an evaluation team. This team should represent different points of view and have a unique set of experiences, prejudices, and expertise. The collection of these diverse judgments leads to a more objective evaluation.

A team might encompass the following roles:

1. Owner

2. Facilities Manager

3. User Representative

4. Programmer

5. Designer

6. Project Manager

The evaluation criteria are standard questions that reflect important values. The evaluation team should review the question set prior to reaching a judgment to understand the meaning of the criteria. Each evaluator forms a subjective response as to the *degree of excellence* attained by the facility. Refer to page 209.

A comprehensive evaluation concerns the equiliibrium of all the forces that shaped the project, and is represented by a quality quotient (QQ). Refer to page 209 for the equation that yields this quotient.

Quality is a value judgment that varies with every individual. It is subjective. Nonetheless, quantification is useful.

First, a rating provides a mechanism for identifying the *differences in perception* of a building by the various evaluators. Better understanding is possible when the evaluation team discusses these differences.

Second, a rating provides an *explicit pattern* of how the parts contribute to the whole assessment. Clearer knowledge of the strengths and weaknesses is possible when the evaluators compare these patterns and discuss them.

5. Lessons Learned

Lessons learned are conclusions about strengths or weaknesses. Rarely should an evaluation conclude with more than twelve statements. At a minimum, four statements will cover each of the major considerations: function, form, economy, and time.

With a trained evaluation team, it is possible to complete the evaluation procedure within a week. For a typical building evaluation, however, the duration of the procedure might last four weeks. Elaborate user satisfaction surveys may extend the duration of the preparation phase. Sophisticated reports may lengthen the documentation phase. The accompanying chart lists typical activities for an evaluation.

Function

When evaluating functional performance, refer to the original Goals and Concepts of the program. The original program provides an immediate focus on the important client decisions that influenced the design.

Form

The evaluation must include aesthetic standards to determine the physical design excellence of the building. This is the most difficult part of the evaluation since aesthetic standards are ever-changing.

Economy

It is important to consider the original quality level of the facility—the quality commensurate with the initial budget. It is unrealistic to wish for a grand quality if the original budget allowed for no more than an economical level.

Time

Because two or three years may elapse between programming and occupancy, the initial users may be different from those involved in the initial planning. A certain amount of user satisfaction, therefore, depends on periodic interior design, or on the degree that partition and utility service changes are possible within the basic structure.

Typical Evaluation Activities

1. **Initiation**
   - Establish the purpose of the evaluation.
   - Identify the background data requirements.
2. **Preparation**
   - Research the background.
   - Prepare the quantitative and qualitative descriptions on analysis cards.
3. **Tour**
   - Make a visual inspection of the facility.
   - Possibly, undertake random interviews with users and probe for responses about performance.
4. **Discussion**
   - Meet to discuss observations after the tour.
5. **Assessment**
   - Make a judgment about the facility's success by assigning a score.
   - Record the ratings on a special graph, which illustrates the pattern of each assessment.
6. **Summation**
   - Review the wall display.
   - Prepare a statement of the lessons learned.
7. **Presentation**
   - Using the analysis cards, present the conclusions.
8. **Documentation**
   - The team leader prepares a report by photocopying the analysis cards.

Key-Word Question Set For Evaluating Facilities

Function

A. *Response to the Major Task*
(intended prime function)

B. *The Overall Organizational Idea*
(the big functional concept)

C. *Effective Arrangement of Spaces*
(functional activities and relationships)

D. *Exciting, Efficient Circulation of People and/or Things*
(flow, orientation, and kinesthetic experience)

E. *Provision of an Appropriate Amount of Space*
(programmed and unprogrammed)

F. *Response to User Physical Needs*
(comfort, safety, convenience, and privacy)

G. *Response to User Social Needs*
(health, interaction, and sense of community)

Form

H. *Creativity and Excellence in Design*
(imagination, innovation, and ingenuity)

I. *A Strong, Clear Statement of Total Form*
(plastic, planer, skeletal form)

J. *Response to the Nature of the Site*
(physical, historical, and aesthetic)

K. *Provisions for Psychological Well Being*
(order, unity, variety, color, and scale)

L. *Integration or Expression of Systems*
(structural, mechanical, and electrical)

M. *Design Excellence of Connections*
(ground, sky, and details)

N. *Symbolism of a Generic Nature*
(appropriate expression and character)

Economy

O. *Appropriate Simplicity or Complexity*
(clarity or ambiguity)

P. *Ease of Maintenance and Operation*
(response to climate and activities)

Q. *Most for the Money*
(good return of investment)

R. *Realistic Solution to a Balanced Budget*
(cost control)

S. *Maximum Effect with Minimum Means*
(elegance and efficiency)

T. *Lean/Clean or Rich Elaboration*
(machine aesthetics or ornamentalism)

U. *Energy Conservation*
(energy efficient)

Time

V. *Use of Materials and Technology of the Time*
(spirit and expression of the time)

W. *Fixed Spaces for Specific Activities*
(major static activities)

X. *Convertible Spaces for Changes in Function*
(dynamic activities)

Y. *Provision for Growth*
(expansibility)

Z. *Vitality and Validity over Time*
(sustaining quality)

A1. *Historical and Cultural Values*
(significance, continuity, and familiarity)

A2. *Advanced Materials and Technology*
(new forms and supportive tools)

Selected Bibliography

In 1959, we wrote an article on “Architectural Analysis” based on ten years’ practice of programming. We were long on practice, short on theory. Serious students of programming may be interested in the following selected bibliography, which influenced the theory behind the evolving problem-seeking method.

Books

Bruner, J. *The Process of Education*. Cambridge, MA Harvard University Press, 1962.

Haefele, John W. *Creativity and Innovation*. NY: Reinhold Publishing Co., 1962.

Osborn, Alex F. *Applied Imagination Principles and Procedures of Creative Problem Solving*. New York, NY: Scribner, 1963.

Polya, G. *How To Solve It*. Garden City, NY: Doubleday Anchor, 1957.

Taylor,Irving A.“The Nature of the Creative Process.’’ *In Creativity, An Examination of the Creative Process*,

Paul Smith, ed. New York, NY: Hastings House, 1959.

Periodicals

Archer, L. Bruce.“Systematic Method for Designers.” *Design*, No.172 (April 1963), pp.46–49.

Peña, William M., and William W. Caudill. “Architectural

Analysis—Prelude to Good Design.” *Architectural Record*, May 1959, pp.178–182.

Current References

Cherry, Edith. *Programming for Design, From Theory to Practice*. New York: John Wiley & Sons, 1998.

Duerk, Donna P. *Architectural Programming, Information Management for Design*. New York: Van Nostrand Reinhold, 1993.

Hershberger, Robert. *Architectural Programming and Predesign Manager*. New York: McGraw Hill, 1999.

IFMA and Haworth, Inc. *Alternative Officing Research and Workplace Strategies*, 1995.

Kumlin, Robert R. *Architectural Programming, Creative Techniques for Design Professionals*. New York: McGraw Hill, 1995.

Parshall and Sutherland. *Officing: Bringing Amenity and Intelligence to Knowledge Work*. Tokyo: LibroPort Co.,Ltd. 1988.

Index

Index

William M. Peña, FAIA

Author of *Problem Seeking: An Architectural Programming Primer*, now in its 4th Edition

After graduating from Texas A&M University in 1948, Willie Peña joined the architectural firm of Caudill Rowlett Scott (CRS). A year later, he became the firm's fourth partner and programmed his first of many building projects.

As a practicing researcher, Willie Peña advanced architectural programming to a sophisticated, analytical science benefiting both architects and clients. He gave to the profession the tools demanded by the complexities of design problems and to the clients, the communication techniques to make their needs known.

In 1950 Peña programmed his first project. By the time he retired in 1984, he had personally participated in the programming of more than 400 projects—one-third of CRS projects completed in 38 states and 9 foreign countries. During his career, Peña also conducted programming workshops and lectured at over 100 professional, corporate and academic sessions.

After 20 years of practice, he developed the Problem Seeking ® programming process. In 1969 he wrote the first edition of *Problem Seeking*. This publication became a standard text in architectural programming courses. *Problem Seeking* was republished as a second edition in 1977 and as a third edition in 1987. This is the publication's fourth edition.

In 1972 the American Institute of Architects elevated Peña to Fellow in recognition of his contributions to the field. Today, he is a consultant to Hellmuth, Obata + Kassabaum.

Honors/Awards

Received the *Chapter Citation for 1990* award from the American Institute of Architects–Houston Chapter

Was named *Officer of the Year* in 1978 at Caudill Rowlett Scott

Was named one of *Ten Most Distinguished Alumni* of Laredo High School by the faculty in 1963

Was named *Outstanding Alumnus of the College of Architecture*, Texas A&M University, May 1998

Received the *Thomas Jefferson Award* from the American Institute of Architects–Houston Chapter in April 2000. This award is given annually to an individual who has demonstrated the creativity, expansive vision, and renaissance approach to the public good, exemplified by Thomas Jefferson.

Steven A. Parshall, FAIA

Steve Parshall is Senior Vice President with Hellmuth, Obata + Kassabaum and Practice Director of HOK Consuting, and a member of the HOK Board of Directors.

With over 25 years of contributions to the practice of architecture, Steve has expanded the architect's role—in architectural programming and in research and evaluation of the built environment—adding value for clients throughout the world. Through benchmarking, training and publishing, his work has enabled architects to understand their clients' needs better, enhancing the profession's capability for providing predesign services.

Parshall, a Fellow of the American Institute of Architects, is a registered architect in Texas. Parshall received a bachelor of science in architecture degree, master of architecture, and master of business administration degrees from the University of Illinois.

Parshall served as chairman of the Research Committee of the International Facility Management Association. He was managing editor of the book *Officing: Bringing Amenity and Intelligence to Knowledge Work*, a joint publication effort with Matsushita Electric Works, Ltd. The bilingual publication focuses on the quintessential twentieth-century workplace. Parshall is coauthor with William Pena of the third and fourth editions of *Problem Seeking: An Architectural Programming Primer*.

Recent project activities include developing real estate and facility plans that make a business case for facility change—both for long-term strategies and immediate implementation. He manages projects for major corporations, defining planning models that tie business forecasting, work process improvement, occupancy cost reduction, and workplace solutions into a comprehensive real estate plan.

Parshall, his wife, and their four children make their home in Houston, Texas.