# HORIZONS IN COMPUTER SYSTEM DESIGN

By
Walter F. Bauer
Intellectronics Laboratories
Ramo-Wooldridge
Canoga Park, California

## Summary

"Computer system design", or using the new phrase "computer organization design", has not received nearly the attention that the design of circuitry and components has received. Circuitry and component development tend to provide the user with higher operating speeds at lower cost but do not necessarily provide the user with a system easily adaptable to his particular needs. There is a great need for computers which can be adapted, as a total system, to the particular needs of the user. The changing needs of the user result from considering the computer as one part of a larger information handling system where man-machine communication is of great importance. There are a number of trends in the organization of computers such as concurrent operation, modularity, control hierarchy, and integral interrogation and display, which help meet the needs of these more sophisticated applications. Analysis shows some of the benefits of modular design. In addition to the design of the computers themselves, the analysis techniques for investigating the application, are appearing. These involve queuing theory and the development of methodologies for information systems analysis and the evaluation of computer applications.

## Introduction

It is interesting to note the phrase "computer organization" in the title of this session. For some time now, those of us interested in the broader systems aspects and user aspects of computer design, have been using the phrase "computer system design". To many, however, the phrase "computer system design" still has the connotation of circuitry and component design or even that branch referred to as "logical design". Although Webster defines "organization" and "system" with phrases that have identical meanings, to many the phrase "computer organization" will have a different meaning than "computer system". The former clearly refers to broader systems aspects of computers such as instruction repertoire, communication with magnetic tapes, systems expandability, etc. Perhaps "computer organization" will stick; at any rate, the word "system" is slightly less overworked.

The thesis of my talk is that insufficient time and energy has been devoted to the analysis of computer organization; the developments in organization have been overshadowed by developments in circuitry and components. There have been some recent developments, however, in computer organization which indicate growing emphasis in this direction. It is my plan to review some of the changing needs, and the trends in computer organization to meet these needs, and to further discuss certain techniques in the analysis of information handling systems.

## The Status of Computer Organization

In the general design of computer systems, three areas of activity come to mind: circuits and components, programming, and systems design or computer organization. There have been noteworthy developments in circuitry and components, many of which could be classified as breakthroughs: the development of large-scale magnetic core memories, the development and use of the transistor, the use of printed circuitry, and computer automated design techniques. We are in the midst of still higher speed circuitry developments with microwave techniques and nanosecond speeds becoming discussed more frequently. Similarily, there have been somewhat less noteworthy, but significant, developments in programming: sophisticated compilers, formula translators, complete programming systems for automatic computer operation; more recently we have seen the development of problem oriented languages such as COBOL (Common Business Oriented Language) and programs for translating between various computer languages.

But the area which has, from my point of view, received far too little attention, and the area which we are discussing here this morning, is computer systems design or computer organization. Until approximately 1958 there was little progress in system design. There had been a gradual improvement in instruction logic with index registers becoming common and certain concepts like the indirect address, coming into use. The idea of buffering data between the input - output was advanced and the interrupt was adopted as a powerful technique for matching the computer approach to asynchronous external devices. While these advances are significant, they do not in any sense compare with the giant steps taken in circuitry and component development.

Happily, during the period since 1958, a number of significant advances have been made. The idea which now appears as obvious, of having a computer operate in a truly parallel fashion was first embodied in the GAMMA 60 and the LARC computers. The designers of these machines saw the possibilities of increasing speed through having large parts of the computer operating simultaneously rather than, or perhaps in addition to, increasing circuitry speeds. While it cannot be proven that parallelism or concurrent operation is more economical than higher speed circuitry, I believe that few would argue that increased parallelism can bear fruit and that it has not been pushed far enough or fast enough.

We are on the verge of other breakthroughs in computer organization. The concept of modularity coupled with sophisticated intramachine communication, such as that found in the Ramo-Wooldridge RW-400 computer, will undoubtedly receive a great play in the future.

## Application Horizons

For the purposes of our discussion here, consider the applications field as being divided into three parts: computing systems, control systems, and information systems. "Computing systems" here refers to the mechanization by electronic computers of processes which are being done or can be done semi-automatically or manually. For example, partial differential equations were solved for many years by manual means. Similarly, business data processing was performed by a variety of machines before the main bulk of the activity was taken over by the electronic digital computer. Computing systems constituted the first application phase of electronic computers; this application will continue to grow and flourish and is, and will remain for many years, the major application area of digital computers in terms of dollars spent.

The "control systems" applications area of electronic computers is a much newer one. Perhaps the first notable control system was the Air Force's SAGE System for Air Defense. Other systems such as refinery process control are appearing. This applications area is possibly best characterized by saying that it provides closed loop control. It frequently requires specialized input - output for the computer, and seldom requires human intervention -- the human only serves to monitor the process. Indeed, an important characteristic of control systems application of computers is that the human can not carry out the process through manual or semi-automatic means because of accuracy requirements, speed requirements, and the requirement for reliability in spite of monotony.

But the newest and most exciting applications area for electronic computers is the so-called "information systems". The lines of definition between the three applications areas are not clear cut and it is difficult in some cases (and unimportant) to decide in which area a particular application belongs. However, information systems are probably best characterized as follows:

1. The computer is imbedded in a system involving data receipt from many sources, information processing, information display, and information dissemination. The computer itself is a relatively small part of the operation.

2. Automatic data inputs from remote sources are part of the system. Data may be control data or data to be processed. The familiar digital data link involving wire communications or radio communications is employed and speeds up to 5500 bits per second are achieved.

3. Man-machine relationships are of the greatest importance. Equipments for the display of information are an integral part of the system as well as equipments to interrogate the memory of the computer. Virtually all the memory of the computer is available to analysts or operators, and many analysts can consult the memory or use the computer facility simultaneously.

4. Processing is characterized by the processing of independent requests for service. The ratio of tolerable delay to processing time is low as compared to most applications of computers for scientific and business purposes.

5. The requirements for reliability are very severe. Electronic failures which cause the complete outage of the system, are intolerable.

6. The information system must have the provision for growing as an integral system. This growth must take place in an orderly way and without lengthy periods of inoperability during change over.

7. The system must respond to a large variety of requests and it must adapt itself to the particular processing requirements of the instant.

Any information system may have these characteristics to varying degrees. In some cases, some of the characteristics are absent completely. For example, the requirement for absolute round-the-clock reliability may be absent. Another comment is in order: frequently these so-called information systems perform closed loop control of devices in addition to providing information to which humans react. In other words, the definitions for the various applications areas here are suggested to provide general guide lines and not absolute lines of demarcation.

Needless to say, information systems is the newest and most sophisticated applications area for electronic computers. Consider for example the use of interrogation devices and display devices in connection with computers. They are almost non-existent today. However, they are beginning to appear as standard pieces of computers and the possible uses of these devices are appearing in numerous places, frequently in unexpected areas.

Quite possibly the market for information systems will dwarf all other market areas eventually. The road ahead, however, is not without its serious obstacles. Programming techniques must be developed much further to allow the efficient communication between man and machine that is made possible, at least, by hardware inquiry stations and display devices. The automatic report generator techniques in computer programming are a beginning step in this direction. Also considerably more work must be done in the overall systems analysis to make good use of computers. It has been frequently stated that computers attached with automatic data input equipment, display and interrogation equipment can be used by corporate top management to conduct their daily business and more specifically, make computer-aided decisions. Operations analysis and decision theory must be developed considerably further before such uses of the equipment can be made.

## Computer Organization Trends

### Computer Adaptability

Information systems, however, give rise to a different set of requirements for the customer than has been previously seen. Almost all of the needs can be summed up in one short comment: There is a need for computers which can adapt to problems. Up to this time, computer customers have found it necessary to adapt their problems to the computers; the computer is purchased as an entire off-the-shelf system and the customer does as best he can to make his problem conform; gradually the problem fills the computer and the customer starts all over again worrying about the next phase.

In the never ending search for speed, many of the requirements of the customer -- many of which now seem obvious and fundamental -- which would make the computer a saleable item, were neglected. Everybody wants to sell computers but it seems that nobody wants to design computers attractive to buyers.

As stated above, the computer must be adaptable; it must be adaptable to the particular characteristics of the application, it must be adaptable to the requirements which may change from minute to minute or from millisecond to millisecond, and it must be adaptable to changes which occur throughout the years.

Adaptability can undoubtedly be best achieved through modularity; the two thoughts are almost synonomous. Having modular components which can be added (or deleted) as the application changes provides the long term adaptability. Adaptability on a microsecond-millisecond basis is provided by connecting the modules by high speed switching equipment.

## Instruction Repertoires

Adaptability to the requirements of the user can appear in many places. One of these places is in the instruction repertoire. Some advances in computer instruction repertoire have been made during the last few years as I noted previously. However, techniques do not exist today to determine in some quantitative way, the power of a set of instructions as applied to a given information processing problem. Perhaps in this connection one should borrow an idea from the mathematical physicist. Consider treating a given application as an abstract vector space which is "spanned", as the mathematical physicists say, by the set of computer instructions. The set of instructions can be considered to be "independent" according to some definition of "independence" and as is done in theoretical physics in the case of a set of vectors. The set of independent vectors is "rotated" to minimize the "energy" required to "span the (problem) space". In other words the set of instructions should be chosen to be minimal from the standpoint of the processing required and with respect to a particular problem application.

A possibility in instruction logic is the idea of building computers which can change their form through changing the set of instructions they provide. This technique is frequently referred to as "micro-programming" and has fallen into a state of considerable disrepute. However, there are many good reasons to exhume the remains. The idea will look promising with the following provisos: if the programming is not burdened unduly by a new set of details, and if the computer can operate at nearly the same speeds as its wired-in counterpart, and if criteria can be established for the intelligent choice of computer instructions such as is suggested in the above paragraph. A possibly attractive idea is to build a basic computer package for marketing, and design it so that the buyer or seller can "particularize" it for a given application.

In a session[1] similar to this one at the EJCC in late 1958, the fact was deplored that instruction repertoires were not oriented toward automatic programming and translation between computer languages. Unfortunately nothing has happened to make that indictment less justified.

## Information Display and Interrogation

Man-machine communication is undergoing important changes. These changes will probably result in most large scale computers being designed from the outset to include display and interrogation consoles. Our first computers, and most of our present ones for that matter, allow the display of perhaps 100 lights to signify the state of the computer or to provide information to the user-operator. In most cases, approximately the same number of on/off switches are available to communicate with the computer. Of course, these computers were not specifically designed with sophisticated man-machine communication in mind. Our modern computers designed for a tight man-machine communication loop involve the presentation of thousands of bits to the user-operator in assimilable form while allowing him communication switches in roughly equal numbers. With most systems currently in use, information in the form of printed copy comes from the machine; the user must usually wait hours -- and frequently days -- before the new set of information, which he requests as a result of what he has just previously learned, can be made available. Not so, of course, with the new display-interrogation operator consoles such as those found on the RW-400 computer.

## Computer Control Hierarchy

Figure 1 depicts a technique increasing in popularity in computer design and computer use -- that of establishing a hierarchy of computer control. A possible result of this "status seeking" of computer operations is shown in the figure. The Systems Management function is the highest level of control; it would be responsible for the systems and procedures of the operation and the problem priority requirements. If, for example, a certain class of problems were not getting the required service, this level of control would change the priority of that problem class. The Systems Management control level would perform utilization analyses which would provide information as to possible equipment changes, and it would provide summaries and records of the way in which the computer system was used. Under this hypothetical model, System Control would be the next control level and would be responsible for the assignment of equipments to do a specific problem and for executing the priorities which were imposed at the higher control level. It would react to stimuli from external sources such as digital data links and the depression of keys at an analyst's console, it would initiate much of the processing required when problems are interrupted, and it would terminate problems as required. At the Problem Control level, macro-instructions would be interpreted and the management of memory would be performed; that is, decisions would be made here as to where data is to be stored, and how and where it is to be transferred. At the Problem Control level, the reactions to internal stimuli would be handled such as the interruption of the problem due to lack of data or the onset of an overflow condition. In addition, at this level the management, programming logistics and subroutines would be performed. The last two levels of control, Programming Logistics and Data Processing and Equipment Control, are the usual kinds of computer control which we have been doing many years. It might be added that a still lower level of control could be identified, namely, "micro programmed control".

Whether the control as indicated here would be carried out by hardware or through programming is uncertain. Probably most of the control would be effected through programming although this control is made possible through hardware. For example, the interrupt signal enables the reaction of the computer to external asynchronous occurrences. It behooves the computer designer to seek ways to make this control hierarchy easy to achieve through programming.

The functions of this hypothetical control model could be carried out in a time-shared way by one serially operating computer or, on the other hand, by computing and control elements working simultaneously and to the extent indicated, independently. Probably the latter technique will be found to be more effective; there will be a trend toward physically separate units performing the indicated control functions.

The importance of a hierarchy in computer control is that it gives the computer a self-organizing character. With such a hierarchy the computer can be introspective much as the human is, and can continually monitor the course of its business to optimize its operations.

Modularity and Reliability

Some of the benefits of a modular-designed computer have been referred to above in discussing computer adaptability. Another advantage of modularity is that it brings about greater reliability at a lower cost.

In Figure 2 there is shown a diagram showing the relationship between modularity and reliability. On the abscissa is the degree of modularity of a computer system defined as 100 minus the percentage of the total represented by the largest module. If, for example, the computer was made up of 10 modules, each of which was 10% of the total, the degree of modularity in this case would be 90%. The ordinate here refers to the extra equipment which is needed to provide a level of reliability equal to that of duplicating an entire computer system. The solid curve shows the amount of extra equipment needed to provide the same degree of reliability as obtained by doubling up on the amount of equipment in the case of the computer with zero modularity. It is assumed that all modules have the same reliability.

There are many vagaries in interpreting a graph of the kind shown in Figure 2, and the results there are meant to be more qualitative than quantitative. However, it can be shown through simple mathematics that the computer 90 per cent modular requires only 30% extra equipment to get the same reliability as duplicating the entire single computer with no modularity, where the assumption is made that the modules are each no more reliable than the entire computer without modularity. The dashed lines showing the total equipment refer to a rough estimate of the equipment needed plus a rough estimate of the additional equipment needed for communication among the modules. It seems clear that as the modularity proceeds closer to 100% -- that is, closer to the transistor or diode level -- that the equipment needed for communication becomes very great, and possibly unbounded in the limit. Experience plus some analysis shows that there is probably a minimal point lying somewhere between 80 and 95% degree of modularity.

Computer reliability, modularity, and control hierarchy, incidentally, team together in an important way. One of the higher levels of control, probably System Control of Figure 1, monitors equipment failures. When failure occurs the module not in use is switched in. Or, if all modules are in use, the lower priority problems are temporarily put aside. In other words, catastrophic failure does not occur; the system automatically adapts itself to carry on the processing with only slightly degraded performance.

## Computer Memories

Progress in the system organization and use of high speed memories is particularly lacking. We have made memories bigger and faster, but that is about all. Some discussion has taken place of the virtues of small ultra-high speed memories for "scratch-pad" use, and it has been adopted in a few cases. A few possibilities for new uses have been advanced[1] but there has been no embodiment of anything organizationally or logically new and, more surprisingly, no analysis has been performed.

Referring to Figure 3, there is a remark that could be made about the duty cycle of large scale memories. Probably the curve of memory usage vs memory size for large scale memories is similar to that shown in the figure. It certainly is true that the absolute requirement for 30,000 words of storage occurs far less frequently than the requirement for 4,000 words of storage. It is realized of course, that this is a function of the problem and a function of the programmer's tolerance of red tape programming logistics. However, since 32,000 words of memory are on hand in the hypothetical example, there is an extra capacity as represented by the part of the rectangle above the curve. This extra capacity is roughly proportional to the excess capacity for the memory device in excess of the "theoretical" amount needed as delineated by the curve. Therefore, it seems that if the computer were designed to share this memory with other problems and if a system could be devised to switch memory assignments, the duty cycle of the memory could be greatly increased and efficiency would accrue. The curve and the large area above the curve suggest that a considerable fraction of the cost of the memory could be spent in the hardware to effect a sharing of the memory and still a saving could be achieved.

## Information Systems Analysis

There was reference above to the need of computers which adapt to problems. The question immediately arises, however, as to deciding what the needs of the problems are. It is clear that if information systems are to take their respected place in our scientific world, we must develop ways of analyzing them.

## Queuing Theory

One of the disciplines which might be applied is queuing theory. It was stated above that information systems are characterized by a low ratio of tolerable delay to processing time for a service request. It is this characteristic which makes a queuing theory approach possible. A non-mathematical statement of the problem would go something like this: service requests for the processing of data arrive on a random basis and it is desired to know the minimum equipment necessary to provide the servicing of these requests while meeting system performance specifications stated in terms of maximum tolerable delay or average delay.

Figure 4 shows a curve which relates the processing speed of a computer system to the average delay in processing a request. The curve is similar to that given by Ackley[2] but the variables have been normalized so as to make the results generally applicable. An important implication of this curve is the following: the average service demand rate in almost all applications goes up as the system develops. If the processing speed does not at the same time go up, the ratio given by the abscissa is reduced and the average delay increases as the curve shows. Obviously then, there is a requirement to have the computer designed so that its processing speed can change according to the demand for service so as to keep the average delay within tolerable bounds and still not have a period of time when the system has excess capability and excess cost. Modular computers provide exactly this ability to increase their processing speed. Computing elements can be added to the system to keep pace with the demand and to keep, consequently, the average delay bounded.

Information Systems Design Methodologies

In the design of information systems it is important to develop techniques for systematically fitting the computer requirements to the operation requirements of the system.

Figure 5 shows a method for carrying out such a design. The operational requirements give rise to the identification of certain data types, volumes of data, the processing required, and the distribution of requests for service. Together they define a problem. If an initial assumption is made as to the speed for processing each one of these types of data, the entire process can be analyzed. Probably the technique used here is simulation by large scale digital computer. This gives rise to information on the average length of delays which, when compared with the delay requirements imposed by the operational characteristics of the system, will determine whether the service is tolerable or not. If the service is intolerable, then assumptions must be changed as to the speed with which the processing is performed. Having determined that the service is tolerable, and having determined the speed requirements, an analysis can be made to determine the complement of equipment necessary. After this is done, the total system is analyzed in view of programming requirements to determine whether the design is in all respects acceptable. Possible imbalances in the service or in the amount of equipment can result in a further analysis and a change in the equipment complement.

An analysis was performed at Ramo-Wooldridge similar to this and was reported by Rothman[3]; the results are shown in Figure 6. The problem mix gives the characteristics of the problem. For example, 50% of the problem required servicing requests which had to be completed on an average of 0.5 minutes and with a total processing time of 1.5 minutes. Poisson distributions were assumed and the frequency of arrivals is given essentially by the abscissa in terms of the average number of requests arriving translated to the number of computers to handle this average load. The zero per cent curve shows that if the number of computers equals the average load, then there is a zero probability of servicing the problem mix. As an excess of computers is applied, service improves as the results indicate.

In many cases, however, the queuing theory approach is not applicable. There is a need for a more general approach to evaluating the applicability of a computer to a given problem. What is sorely needed then is a technique for developing the important and significant characteristics of problems and, likewise, the important and significant characteristics of computers.

The general thought is portrayed in Figure 7 where, starting with the general problem characteristics and the general computer characteristics, a problem model of general applicability and a computer model of general applicability, results. (The author is indebted to S. Rothman of Ramo-Wooldridge for stimulating discussions on the subject.) If a specific problem is described in terms of this general problem, and a specific computer is described in terms of this general computer module, the theoretical performance of that computer involving the particular problem would be obtained. If the model were deemed sufficiently valid, the results then at this point would be useful. In the development of the methodology however, it would probably be desirable to compare the theoretical performance with the actual performance, providing that the specific problem had been already solved by a specific computer. By making this comparison one determines what problem model changes and what computer model changes are necessary. These then can be sent back to improve the problem model and the computer model.

## Conclusion

That there has been little progress in computer system design or organization stems from the fact that computer technology is relatively new. That excuse, of course, ages rapidly. A contributing factor is that most commerical computer manufacturers are inherently conservative, and see increased circuit speed as the only solution. Unfortunately other groups which are in a position to promote progress, have not always been sufficiently perceptive. However, "right is might", and the fruits of these approaches will most likely become increasingly manifest.

## References

1. "Computer Design from the Programmer's Viewpoint", W. F. Bauer, Proceedings Eastern Joint Computer Conference, December 1958.

2. "The Multi-Sequence Computer as a Communications Tool", J. N. Ackley, Proceedings of the Eastern Joint Computer Conference, December 1959.

3. "The RW-400 Data Processing System", S. Rothman, Proceedings of the Auto-Math Conference, International Congress of Information Processing, Paris, June 1959.

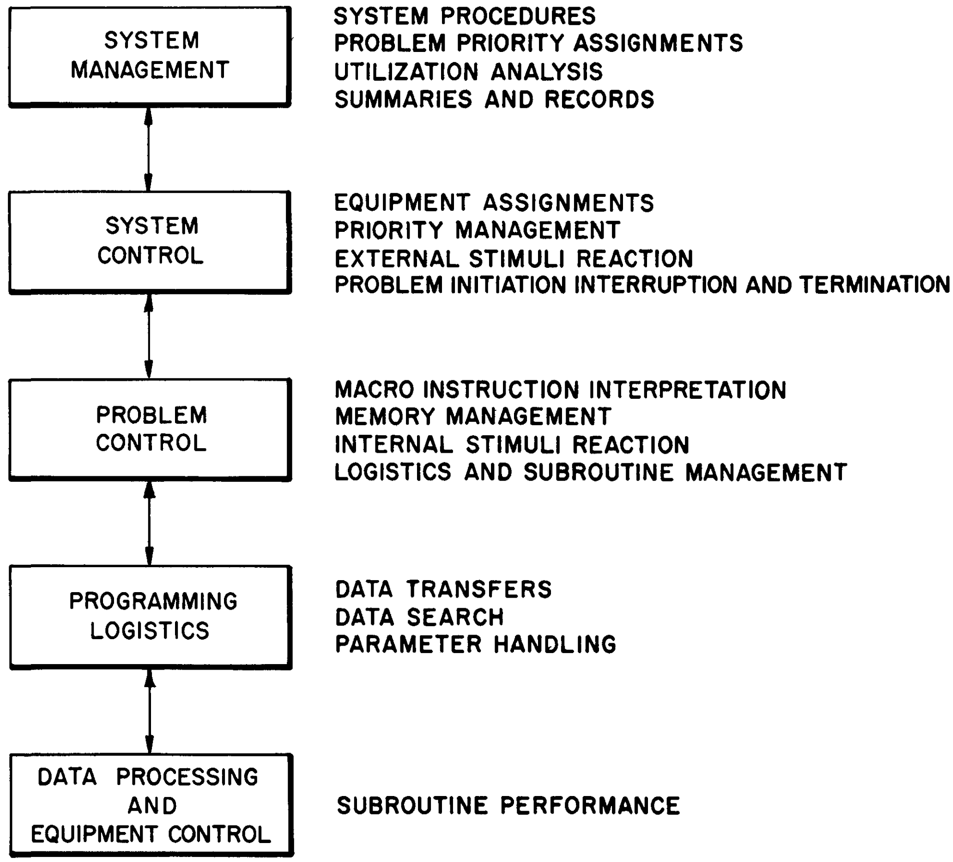


Figure 1. Computer Control Hierarchy

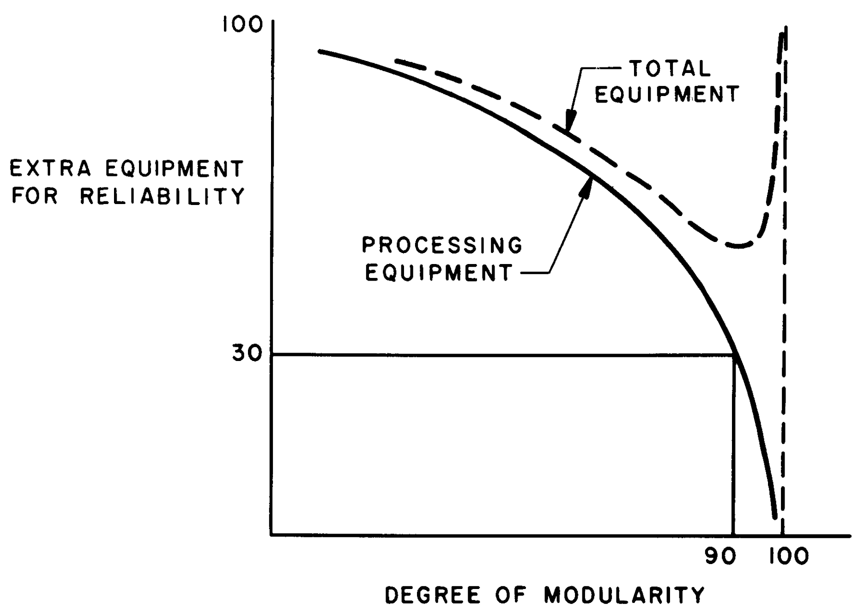


Figure 2. Reliability and Modularity

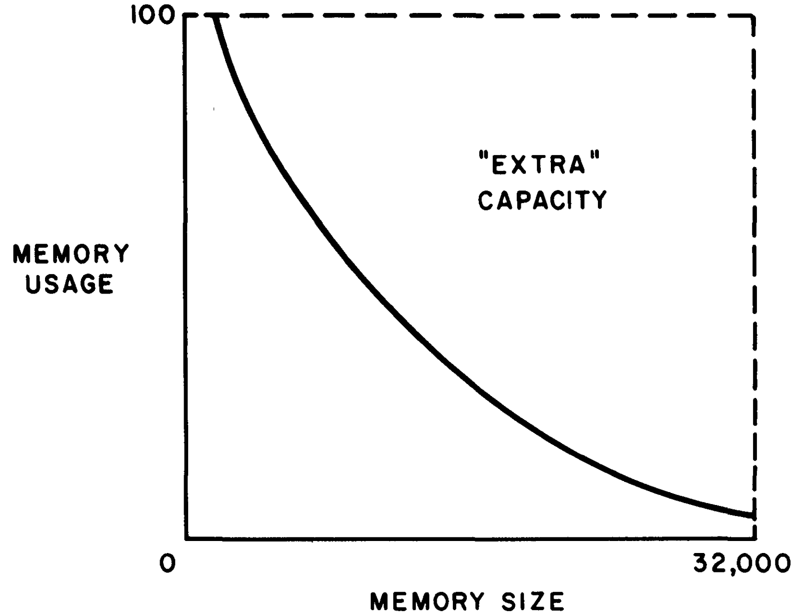


Figure 3. Memory Use

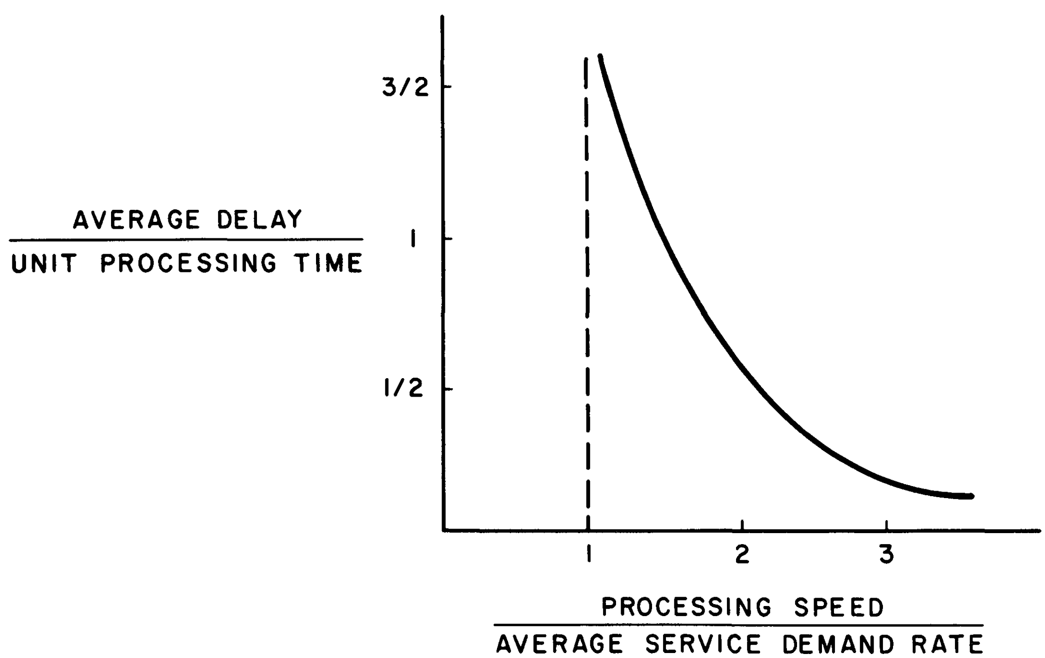


Figure 4. Queuing Theory Application

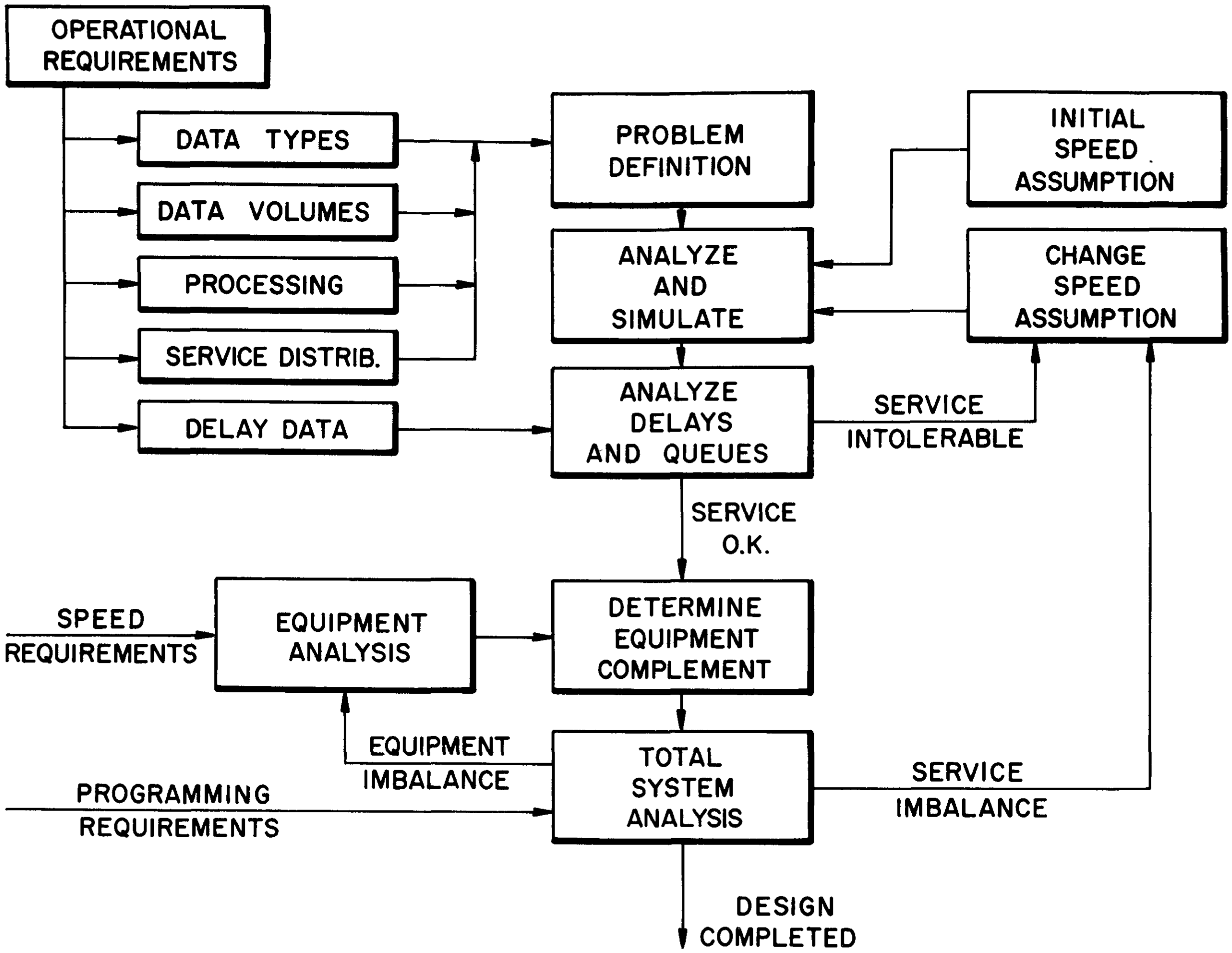


Figure 5. Information System Design Methodology

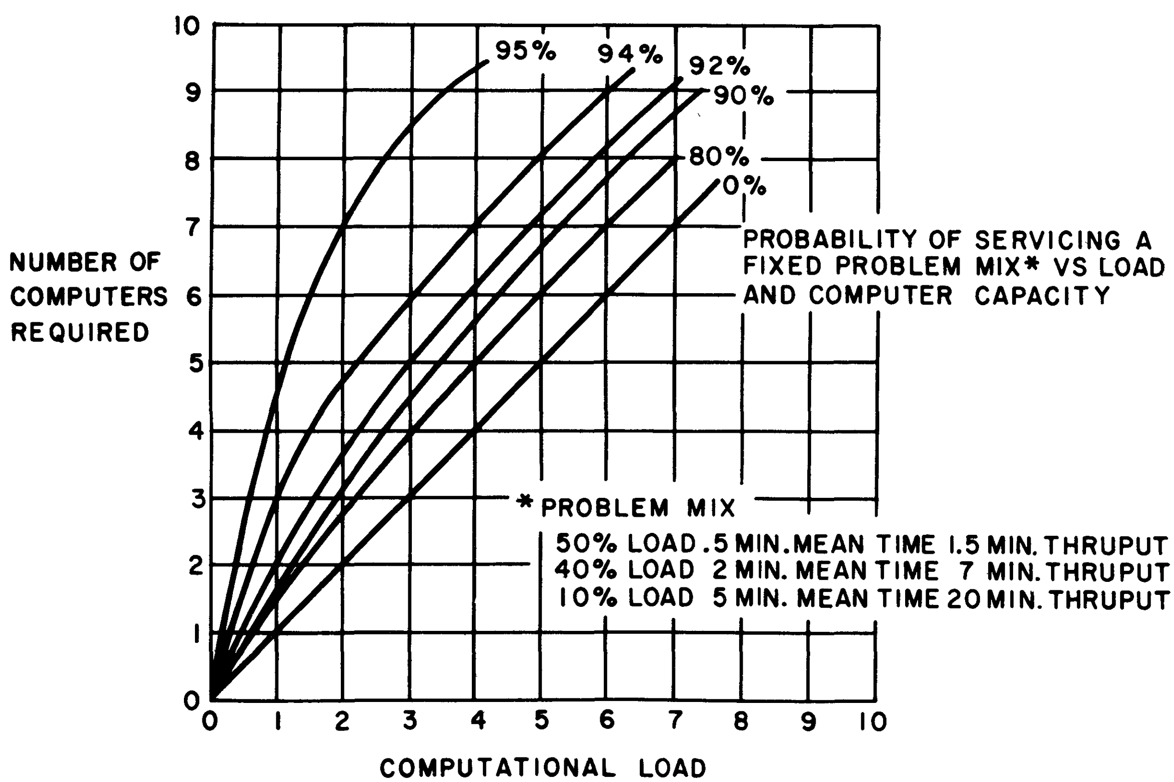


Figure 6. Queuing Analysis Example

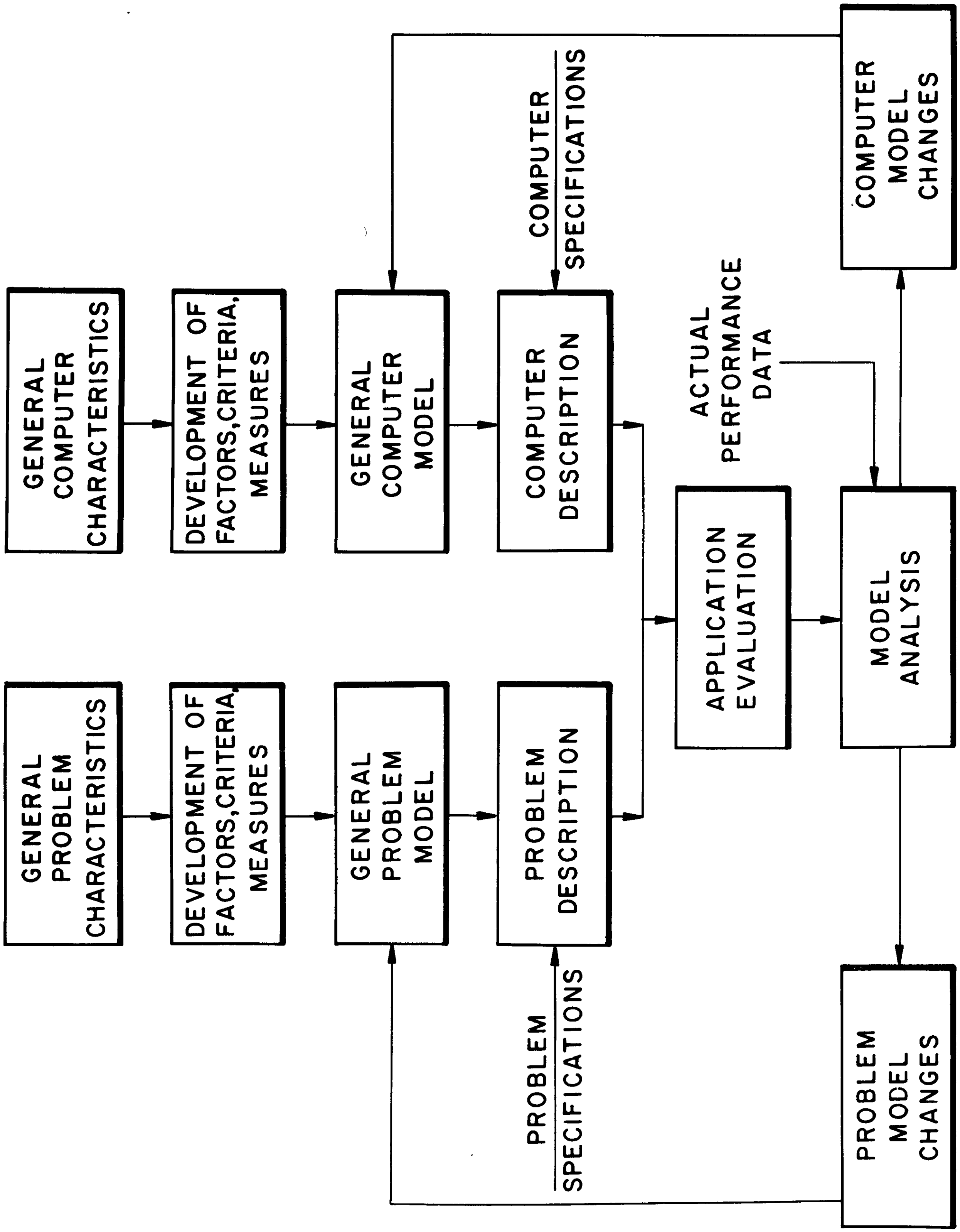


Figure 7. Application Evaluation Methodology