An overview of techniques available to address capacity planning in the production data processing environment is presented. The production data processing system is briefly described and its capacity is quantified. The measurement tools, reports, and data required to implement a capacity planning program are discussed. Modeling and prediction are placed in perspective with the overall objectives of the capacity planning process. Personnel (managerial and technical) and organization considerations are also discussed.

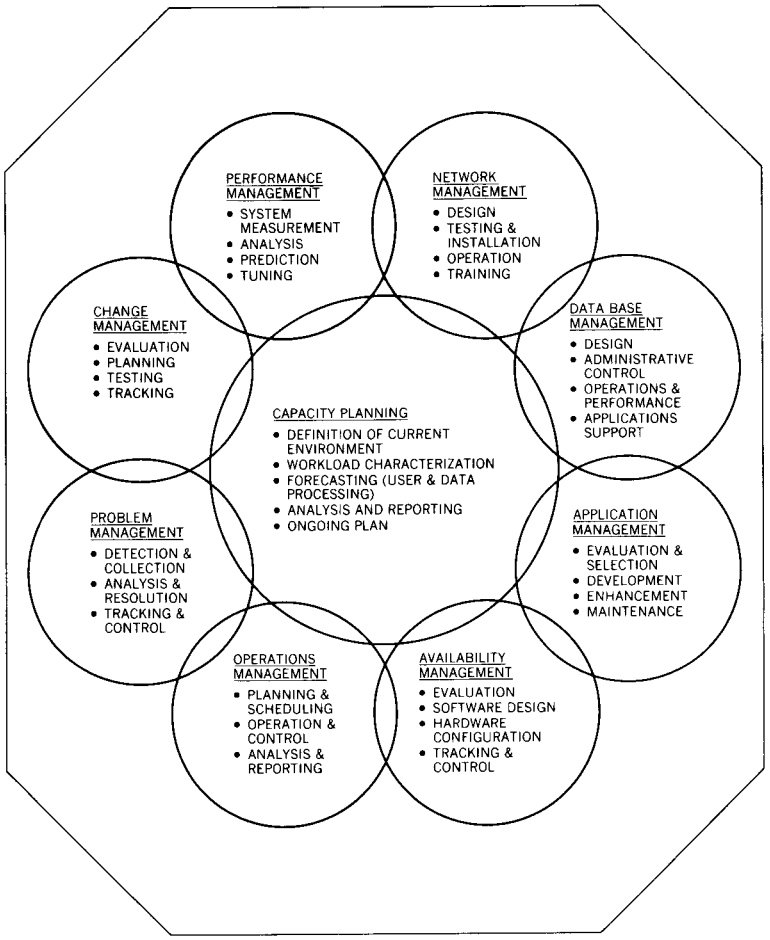
Overview of the capacity planning process for production data processing

by L. Bronner

Production data processing is an automated approach to taking as input information required to run a business enterprise and processing it in accordance with certain use specifications. In this environment, information volumes can be quite large, and user demands for service very stringent. Therefore, the planning problems to be addressed are those of forecasting user workloads, determining the required computer capacity, and effectively and efficiently managing resources (people, hardware, software) to meet user service objectives. Computer capacity planning[1,2] is a process developed to provide a systematic approach for understanding and predicting the capacity of production data processing systems.

The basic concepts underlying capacity planning are not new. Capacity planning, as discussed in this paper, is basically a systematic method of bringing together many of the past performance management ideas and integrating them with current performance management and measurement technology. Capacity planning addresses the problems involved in managing computer resources, namely:

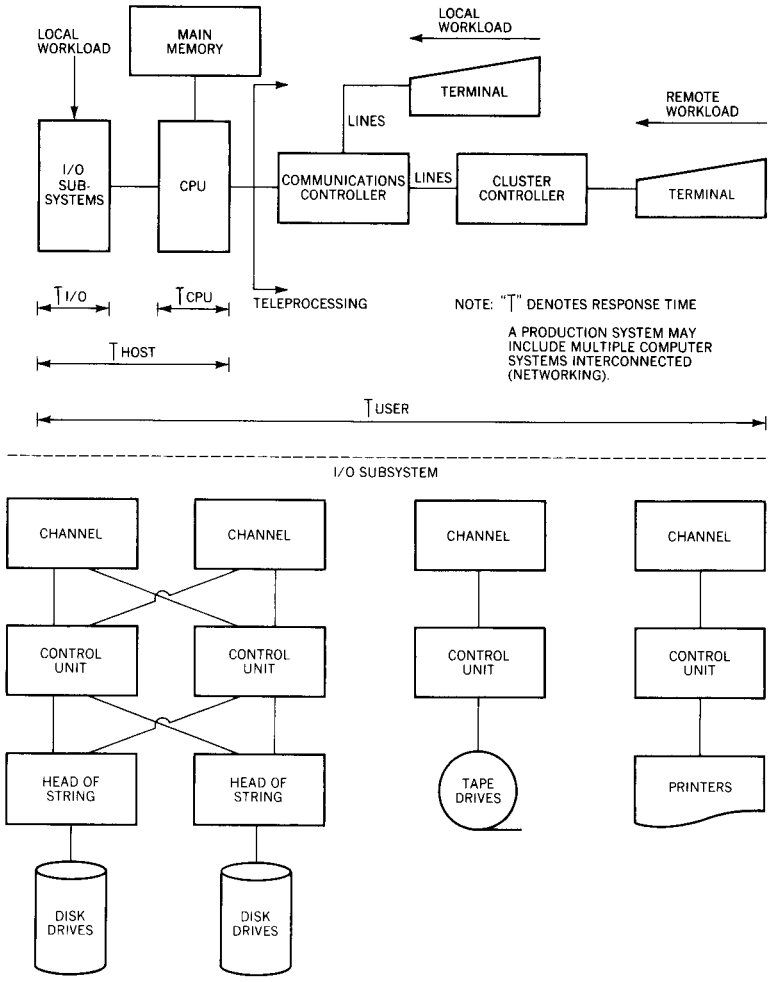
Figure 1 Capacity planning/installation management



- What parameters are to be collected to characterize the workload?
- What parameters are to be collected to characterize the software and hardware components?
- What parameters are required to forecast future workloads and system performance?
- What products are required to collect, analyze, and report the data items described above?
- How should the data processing executive manage his installation on a continuing basis using the data described above and the results of analysis (required reports, reporting formats, report flow, recipients, etc.)?

Capacity planning is basically a performance-oriented approach to data processing management. By this process, the loading, uti-

Figure 2 Production computer system



lization, and response of the various system resources are monitored and analyzed. Also, the flow of current and future work through the system is controlled to provide the best overall user satisfaction. Experience with many data processing installations over the last four years has shown that user satisfaction is the most critical factor in the capacity planning process.

In many instances, capacity planning is confused with the much broader area of installation management. Installation management (Figure 1) is concerned with the management of the following areas of a data processing installation:

- Performance
- Changes

- Problems
- Operations
- Availability
- Applications
- Data bases
- Networks

Generally, installation management is a process directed at understanding, correcting, and controlling anything that detracts from the normal operation of the computer system. Therefore, it is very natural for a capacity planning effort to overlap many installation management functions. Figure 1 depicts these interrelationships. Reference 2 contains a discussion on each of the installation management areas cited above and their relationships to the overall capacity planning effort.

In the remainder of this paper, various capacity planning techniques are discussed. The components used to characterize the computer system are outlined, and the parameters to be measured to establish the capacity of a computer system are discussed. Measurement tools, data requirements, and specific reports are defined. Performance model development and parameter predictions are placed in the proper perspective with respect to the overall effort. The data processing organization structure and its capacity planning relationships are discussed. Also, a bibliography of relevant capacity planning articles is provided.

The capacity planning process

production computer system

The capacity planning process discussed in this paper is developed for implementation on a production computer system as depicted in Figure 2. The basic hardware subsystems are: the CPU, main memory, I/O devices that include channels, control units, disk drives, tape drives, and printers, and teleprocessing equipment.

A function of capacity planning is to understand the utilization of each subsystem through measurement. Experience has shown that software measurement tools are preferred because subsystem use is required by some reasonable segmentation of the workload—by application (payroll, accounts receivable, inventory, etc.), by department (engineering, administration, etc.), and so forth.

capacity of a computer system

In discussing the capacity of a computer system it is necessary to differentiate between the factors that affect the capacity of a resource (Figure 3) and those that affect the capacity of the computer system (Figure 4). Although many installations indicate that they are unable to set or establish specific user service objectives

Figure 3 Resource capacity

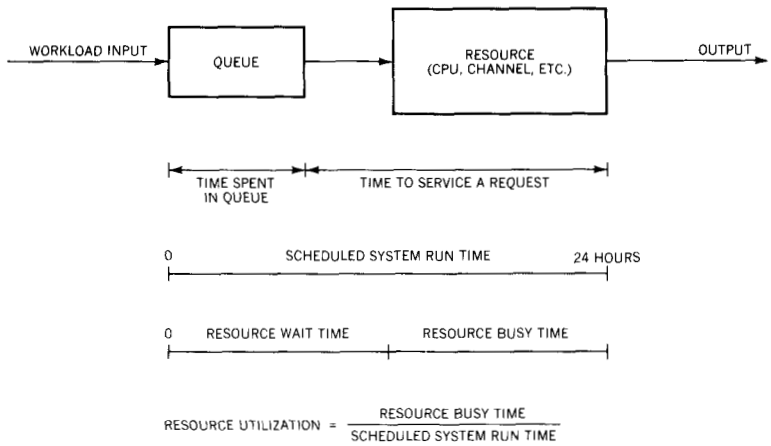
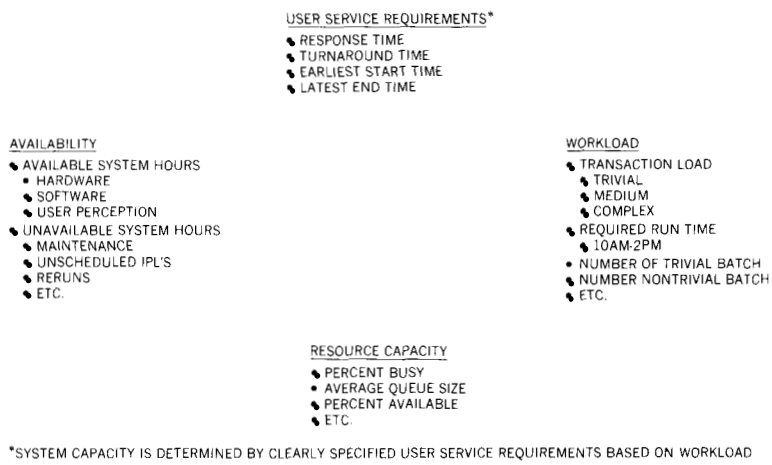


Figure 4 System capacity



or that such requirements are not practical in their environment, I submit that it will be very difficult to establish or understand the capacity of a computer system until user service objectives are defined.

The primary indication of the capacity of a resource is the time the resource requires to complete a request for service (Figure 3). Hence, in the scheduled operation of a resource, the summation of the times required to complete all requests for service is equated to the resource busy time. With respect to the measurement tools being used today, it is the resource busy time or resource utilization that is indicative of the overall capacity of a

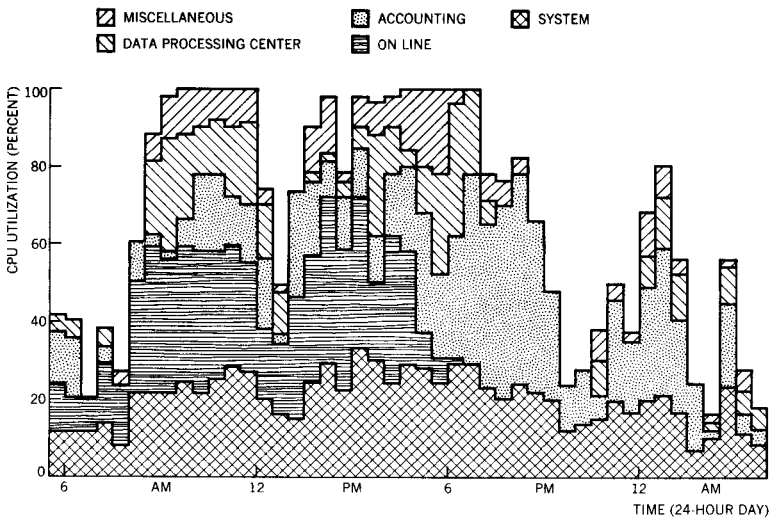
resource. When the busy time of a resource is equal to its scheduled run time, the total capacity of the resource has been consumed (no wait time component). In essence, every time a request for service has been completed, there is another request to be serviced. In queuing theory, it would be said that the resource is 100 percent utilized. This state of a resource normally gives rise to very large queue sizes (i.e., large numbers of requests waiting to be serviced).

The capacity of a resource as outlined in the previous paragraph might also be thought of as the independent or stand-alone capacity. If resources are looked at purely on a box by box basis, each can be monitored with 100 percent utilization as a capacity constraint. However, to understand the capacity of a resource as part of a computer system, it is necessary to understand capacity other than as an independent concept, because, in most instances, a resource does not sustain a continuous busy state over the scheduled period of operation. Normally, user service degrades to an unsatisfactory level before saturation (100 percent utilization) so that other alternatives (new hardware, off-load work, tuning, etc.) must be taken to improve system response.

The capacity of a resource may be viewed as having a potential of 100 percent utilization. But in most practical instances, a resource will not realize its full potential when it is constrained by installation service objectives. Hence, the capacity of a resource will vary among installations as well as within an installation, depending on the time of day. The upper limit on the capacity of a resource is the utilization (busy time divided by scheduled run time) above which the given resource becomes a bottleneck and degrades the response/turnaround time so that the user service objective can no longer be met. For example, experience at the IBM Washington Systems Center has shown that a channel within a computer system with an average utilization above 35 percent for a given Resource Measurement Facility (RMF) interval (30 or 60 minutes) of operation will elongate response time in an interactive environment.[3] Hence, interactive users may find their response time degrades to unacceptable limits.

For capacity planning purposes, a computer installation should be viewed as a system of resources. In other words, the capacity to be analyzed must be that of the total computer system. It is germane to know that a given CPU can execute "X" million instructions per second (MIPS) or that a channel is capable of transferring "Y" bytes per second, but the critical issue is: Will the combined performance of the resources provide satisfactory user service in terms of response/turnaround time? From the author's experience, the principal factors to consider in developing an understanding of the capacity of a computer system are outlined in Figure 4.

Figure 5 CPU consumption by application



workload characterization

Workload characterization[4-6] is a very important factor in understanding the capacity of a computer system. In many computer capacity analyses done today, the data requirements for transactions processed (i.e., transaction rates, paging rates, resource utilizations, response times, etc.) are normally not enough to adequately assess the capacity of a computer system. This is not to say that these are not important parameters, but there are other considerations (i.e., specific time windows, predecessor job requirements, number of tape mounts, etc.) which in many cases have been overlooked in capacity modeling efforts.

For example, one consideration is understanding the capacity of a system during specific time windows (e.g., 8:00 AM to 11:00 AM or 1:00 PM to 4:00 PM) versus using average daily parameters of performance as in the case of CPU utilization, which is shown plotted over a 24-hour period in Figure 5. There is obvious computer resource capacity available, as indicated by the many "valleys" on the graph. Assume that the resources of this installation are relatively well-tuned and no "bottlenecking" of resources is restricting the performance of the CPU. Also, assume that all user service requirements are being met during the peak periods from 8:00 AM to 12:00 noon and 2:00 PM to 7:00 PM where the CPU is sustaining 100 percent utilization. If it is known that the work being accomplished during the peak period cannot be shifted to other machines or different times of the day, then for all practical purposes the computer is out of capacity during these time windows regardless of what average values or modeling will indicate. Reasons would indicate that capacity is a function of the time of

the day, week, or month, and that scheduling of the workload bears very heavily upon understanding the capacity of a system. It is these kinds of considerations that begin to truly address the critical problem of system capacity and workload characterization. Until you understand how the workload of your installation is characterized, capacity planning will be a very difficult task.

availability

Understanding the availability of a computer system is a very important aspect of capacity planning.[2] From a system point of view, availability of the computer to do actual user problem program work is the key issue. As pointed out in Figure 4, there are three areas of availability to be addressed: (1) hardware, (2) software, and (3) user perception.

Of greatest importance to the capacity planning effort is the user's perception of his availability. It is of little consolation to a user that the hardware and software are up and running (available) when a critical data base is down or being reconstructed. This means that, for the applications requiring this data base, the system is unavailable and service commitments are not being met.

user service requirements

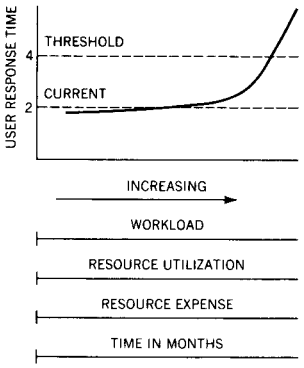
Although system availability, workload characterization, and resource utilization are very important factors in understanding computer system capacity, the key to establishing current and future capacities is the user service requirements. Without a firm fix on user service requirements, the capacity of a computer system will be very nebulous and in effect will float between many different values as system requirements change. For example, before moving from the capabilities of one computing system to another of greater capability (e.g., from a system driven by an IBM 3032 processor to one driven by a 3033 processor), the service (response/turnaround times) being provided to the critical batch and on-line applications on the current system should be established. In this light, the future capacity requirements are forecast with these performance parameters (response/turnaround times) as the base. It is usually decided what new applications are possible with the new configuration and what growth is to be accommodated in old applications.

One of the factors used in determining whether the new configuration will live up to its expectation is adherence to the old service requirements. Capacity is not normally allocated for users of current applications to move to a drastically improved service. However, user service is usually improved as a part of the migration to the new configuration. This improvement in user service tends to leave users with a false impression of their service requirement. The users may feel that their improved service must be maintained even at the expense of new planned applications. What this means is that users not aware of specific service objec-

tives established for their applications will reject any plan to return to some lesser service, even though that is the service they were provided with on the old system. The capacity of a computer system is caught up in the negotiations and agreements on user service requirements between data processing operations and the user community.

An important concern expressed by many users over the past several years is consistency of service rather than an improved service. They are requesting that once established, the service be maintained. This concern applies primarily to an on-line environment where certain work procedures are developed around a particular user service (response time). When the response time values change significantly (improve or degrade), procedures can be greatly impacted.

Figure 6 System capacity (theoretical)



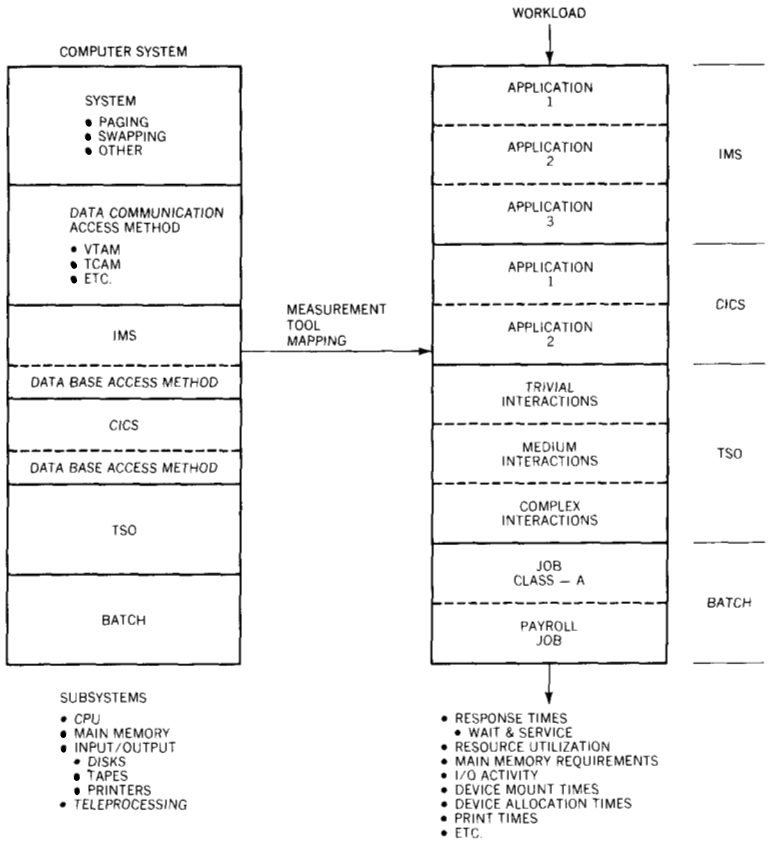
Another question arises concerning system capacity and user service objectives: "How is future capacity planned using response and turnaround times?" There are several methods available[7-9] in which the workload of a computer system (transactions per second, jobs per hour, etc.) is increased and the change in response or turnaround time is predicted. From a theoretical point of view, queuing analysis or discrete simulation may be used. A model is developed, and various known values of load are used as input. If the current user service being provided is known and some threshold (Figure 6) which cannot be exceeded is provided, the model workload is increased until the threshold is reached. At this load, which is indicative of a period of time in the future, resource utilizations may be noted from the model, and the expense of relieving any resource bottlenecks can be evaluated.

capacity planning is a process

A computer installation is a very dynamic environment where you have changing hardware, software, techniques, and people. In this paper, the examples will refer to the MVS (Multiple Virtual Storage) operating system, specific pieces of hardware (the IBM 3033 processor, IBM 3350 storage device, etc.) and such current software tools as the Resource Measurement Facility (RMF), System Measurement Facility (SMF), etc. However, because of the continually changing data processing environment, a methodology developed for capacity planning should be as independent as possible of any specific product.

Basically, all products (hardware and software) should be viewed as inputs to the capacity planning process. For example, a change in the overall capacity planning process should be minimal when it is necessary to change from one operating system to another, as from MVT (Multiprocessing with a Variable Number of Tasks) to MVS. Obviously, this is easier said than implemented. The author is well aware of the difficulties in moving to a new oper-

Figure 7 Measurement tool requirement



ating system when capacity planning techniques are relatively well-defined under another operating system.

Measurement tools, reports, and data requirements

Before specific measurement tools and some of the inherent difficulties in their use are described, a brief discussion is given on the type of measurement tool it would be desirable to have for capacity planning. A production data processing environment is driven by its users. The users create the work. They are responsible for any growth in the current load and for new application development. Therefore, as shown in Figure 7, it is desirable to have measurement tools to segment the consumption or utilization of each subsystem by user. Furthermore, it is desirable to have the ability to group user consumption by application (as shown in Figure 7), by department, or by some unique function. This means system services, data communication service, etc. are all

apportioned properly across users. Also, the workload and I/O activity are similarly apportioned. Then, as shown in Figure 7, response times, main memory utilization, etc., may be measured and associated with the appropriate users. In this context, a user might be a batch program as well as an interactive or on-line user.

An aim of the capacity planning process is to understand each user's (or group of users') demand for subsystem resources. This would be accomplished by segmenting and tracking the workload and associated resource consumption. Then models and predictions would be based on these data and data trends. It is also understood that although these might be desirable requests for a measurement tool, they may not be practical under current architectures. For example, it might be too costly in terms of CPU cycles to place all the "hooks" required in the operating system to accurately apportion all of the utilization among each user, although future architectures might adequately support these types of capacity planning measures.

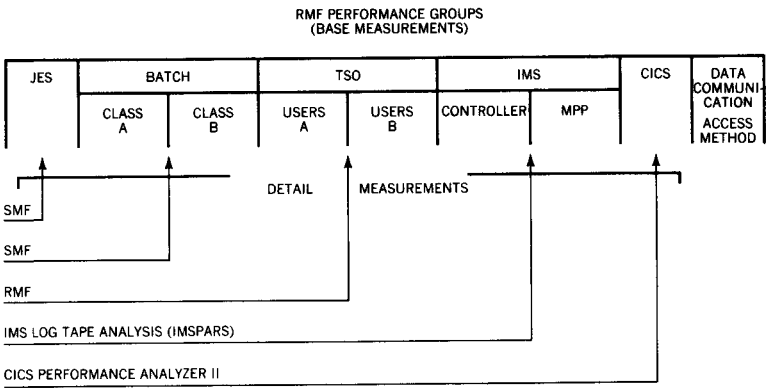
Table 1 Operating system environments

Operating systems	MVS, SVS	VS1
Software subsystems	Batch TSO CICS/VS IMS/VS	Batch CICS/VS IMS/VS

To be more specific and address some of the experiences in developing a capacity planning process, the MVS, SVS (Single Virtual Storage), and VS1 operating systems will be used as examples. The on-line program products used in these environments are the Customer Information Control System/Virtual Storage (CICS/VS) and Information Management System/Virtual Storage (IMS/VS). The interactive Time Sharing Option (TSO) is available under MVS or SVS. Batch processing is available with each operating system. The example environments are summarized in Table 1. The measurement tools recommended for capacity planning in the environments given in Table 1 are listed in Table 2.

These measurement tools are categorized as *base* or *detail*. The reason for this distinction is, as experience has shown, that correlation of many of the same measured parameters (e.g., CPU time, I/O activity, response time, etc.) between different measurement tools may require a detailed analysis of the method of data collection of each tool. The approach of the methodology being presented is to simplify the planning process as much as possible. Therefore, it is recommended that one measurement tool be selected as the base or controlling measurement. In MVS, RMF (Figure 8) is used as the base. As shown in Figure 8, each software subsystem is placed in a performance group, and various statistics are collected. The primary purpose of the base tool is to ensure that the parameters characterizing the various segments (i.e., JES (Job Entry Subsystem), TSO, etc.) add up to the total. For example, the sum over the CPU utilizations for all subsystems should equal the total utilization. If the parts do not sum to the total for the base tool, any required adjustments should be made and validated. For example, adjustments should be made to the performance group CPU time when RMF is used.

Figure 8 Measurement tool integration



NOTE: SVSPT AND VS1PT WOULD SUBSTITUTE FOR RMF WITH REQUIRED TRADE-OFFS.

Table 2 Recommended measurement tools

Type of tool	*Tool*
Base measurement tools	RMF (Resource Measurement Facility) SVSPT (Single Virtual System Performance Tool) VS1PT (Virtual System 1 Performance Tool)
Detail measurement tools	SMF (System Measurement Facility) CICS/VS Performance Analyzer II IMS/VS Log Tape Analysis IMSPARS (IMS Performance Analysis and Reporting System) GPAR (Generalized Performance Analysis Reporting) Print Load Analyzer
Manual data collection	Logs System parameters Etc.

This type of deficiency is the reason for developing the concept of "Capture Ratios."[10] A capture ratio is a factor indicating the amount of a parameter captured in relation to the true value of the parameter. For example, the RMF measurement tool captures a portion of the total CPU time used by a user program. Therefore, by using capture ratios, RMF performance group CPU time can be adjusted to a more complete value. Then, the parts should more closely sum to the aggregate. Thus, the base measurements are established.

The measurement tools designated as detail can be used to more finely segment the performance group values. For example, assume the CPU time for CICS/VS has been captured in a performance group and the capture ratio applied. Then, the CICS/VS Perform-

ance Analyzer II (PA-II) can be used to segment the overall RMF CICS/VS CPU time. If the sampling times of the measurement tools are properly synchronized, which is not always a trivial matter in a production environment, RMF and CICS/VS PA-II CPU time should correlate within reason. CICS/VS PA-II collects CPU time by transaction identification, which makes it possible to segment data by department, function, etc., if it is assumed that the transaction identifications are structured in this way. For the SVS and VS1 environments, SVSPT and VS1PT would be substituted for RMF with the required trade-offs (i.e., partition data versus performance group).

The kinds of data recommended for capacity planning are summarized below:

- Utilizations by CPU, software subsystem, application
- Utilizations and SIOs (start I/O instructions) by channel, control unit, head of string, I/O devices
- Transactions per time period
- Total number of transactions
- Response time by transaction
- Response time by transaction type or class
- Multiprogramming level
- EXCPs (execute channel program instructions) and SIOs by transaction
- EXCPs and SIOs by transaction type or class

These recommendations are the results of many different capacity planning analyses done over the past four years. Some examples of transactions are batch jobs, TSO interactions, and IMS or CICS transactions.

The utilization values, when possible, should be broken out by software subsystem (batch, TSO, IMS, CICS, etc.) and by application or department within subsystem. There is no clear approach for separating channel, control unit, head of string, and I/O device utilization by subsystem or application within subsystem. However, techniques are being developed that attempt to apportion these values.[7]

Workload and user service data are required. In this case, workload is the volume of transactions that serve as input and the associated I/O activity that is generated. After workload and resource consumption data have been collected, the other critical parameter is user service (response or turnaround time). There are other parameters of interest (e.g., paging, swapping, message lengths, etc.) not specifically noted here.

A primary recommendation in the development of a capacity planning process is to keep it as simple as possible and to keep the

Table 3 Recommended base measurement reports

RMF reports	*SVSPT reports*	*VS1PT reports*
CPU activity	CPU and channels	CPU utilization by partition
Workload activity	Job activity	Channel utilization
Channel activity	Real main storage	I/O device report
I/O device activity	Direct access devices	Real main storage
Paging activity	Nondirect access devices	Basic system profile
Page/swap data set activity	Basic system profile	Summary
	Overall activity	

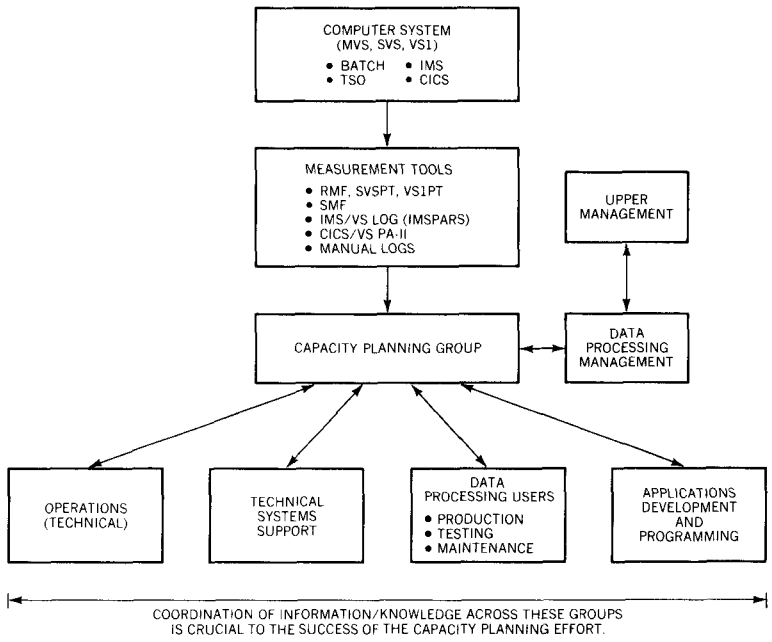
Table 4 Recommended detail measurement reports

CICS/VS Performance Analyzer II	*IMS Log Tape Analysis* (IMSPARS, GPAR)
Summary list report	CPU usage
Model report	Resource availability
Final totals report	Data base update activity
Summary DL/1 clocks	Management exception report
Summary DL/1 counters	Message queue utilization
SMF Record Types	*Print Load Analyzer*
4 —Step termination	VS2 print report
5 —Job termination	VS1 print report
6 —JES2/JES3 output writer	
26—JES2/JES3 job purge	
34—TS step termination	
35—Log off	
40—Dynamic DD	

amount of data to be analyzed to a minimum. This fact is so crucial to the success of capacity planning that in some instances accuracy may be sacrificed for manageability of data. For example, in a production shop having 150 to 200 disk drives, the problem becomes one of identifying the critical paths (channel, control unit, head of string) and disk drives so as to provide a reasonable base of data for analysis. In trying to accurately analyze this I/O structure, the amount of data one is faced with for all disk drives makes analysis almost impossible.

In Tables 3 and 4, the reports[11-17] outlined will allow a reasonable capacity planning effort to be developed. These reports would provide the data described above. This listing may not be inclusive of all reports required nor should it be assumed that each report must be used. The point is that a structure of reporting could be developed and expanded as required. The reporting structure would outline the required reports and data items to be captured, identify the appropriate people in the organization to receive the reports, define the functions each recipient is expected to perform with the reported data, etc.

Figure 9 Organizational reporting requirements



organizational reporting requirements

As shown in Figure 9, performance measurement tools are available to provide information for the following areas:

- Upper management
  - Corporate
  - User
  - Data processing
- Data processing management (below vice-president/director level)
- Capacity planning group
- Operations
- Technical systems support
- Applications development and programming
- Data processing users

Each area has certain unique data requirements, but a great deal of overlap has been noted.

Reporting the proper data to operations accomplishes two things. First, it provides personnel in operations with a better understanding of the user workload characteristics and the rate at which resources are being consumed. Second, it provides for validation of the measurement data and the conceptualization that the capacity planning group has of the overall computer operation (availability, resource utilization, workload, scheduling, etc.).

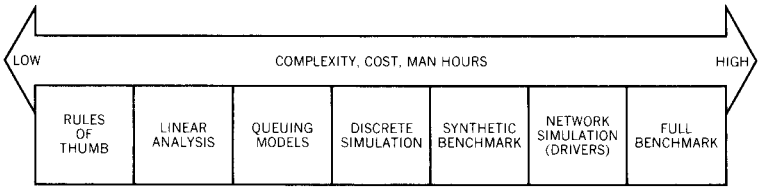
As part of the capacity planning process, it is very important that the system programmer receive the required system performance data that relates to certain fixed and changed parameters. It is not expected that a system programmer would predict various performance changes due to a change in a system parameter. However, systematically tracking and correlating performance results with parameter selection will provide invaluable insight into system tailoring for performance. Also, incorrect selection of system parameters may create major system bottlenecks. Therefore, it is important that the system programmer receive performance information on a continuing basis.

One of the most difficult problems in the capacity planning process is predicting computer system requirements for new applications. Experience has shown that the most readily used means of predicting new application requirements is comparing the proposed application to an existing application where certain performance parameters (CPU time required, response or elapsed time at various loadings, channel and device activity rates, etc.) are known. The author is unaware of any nice, neat equations or models from which various performance requirements result as output for a certain prescribed input. The best way to improve new application performance predictions is by measurement and maintenance of pertinent historical data files. For example, a new application can only be successfully compared to an existing application and its performance data used for prediction when such data has been measured and maintained. Also, validation of the method is possible by measuring the performance parameters of the new application after it is implemented and comparing it to the old base application. Therefore, to improve new application predictions, the application development group should receive certain application performance data on a continuing basis.

A primary problem to be addressed in capacity planning is characterization of the user workload. Characterization of a workload involves scheduling. Scheduling will answer such questions as when are the peak transaction loads and when must certain critical or heavy use batch jobs be run. Also, it may be necessary to define predecessor and feeder job requirements in a batch environment. These are only a few of the additional factors that may be required for user workload characterization. A characterization is a function of the industry (banking, retail, petroleum, etc.), user (clerk, technician, manager, etc.), and the actual customer within an industry. Workload characterization is a difficult task, and in most instances will require enhancement when greater knowledge of the user and his application is gained by ongoing tracking of the environment.

The data to be reported to upper management (data processing and corporate) must be clear and concise, and must represent the

Figure 10 Spectrum of performance analysis techniques



most pertinent factors (workload, user service, availability, etc.) on system performance. In most cases, when a data processing installation finds its system is out of capacity, upper management is the last to know. Then, the primary reason for informing upper management is to have them sign or approve the order for new equipment. One of the primary purposes of the capacity planning process is to keep upper management informed of the status of the available system capacity on a continuing basis. Therefore, equipment requests are not apt to be a surprise.

Analysis techniques used in capacity planning

As a part of the capacity planning effort, there is a requirement to provide data to develop models for prediction. Also, data is to be provided for input to these models and to validate their predictions. Models[7-10, 18] may take many different forms and have varying degrees of detail. Models may be very simple in that they are basically guidelines developed by monitoring the operation of the computer system over time. For example, when the available number of TSO users exceeds 40 on a system, operations will normally receive more user complaints. A workload of a certain number of transactions per second might be associated with this level of user discontent. This can be thought of as a model to be used as an aid in analyzing the TSO environment. For a more costly and detailed form of modeling, a given computer system might be structured with the actual hardware and software, where the actual application programs will be run for a period of time and certain performance data collected and analyzed. This procedure is termed a benchmark, and in many situations affords the most accurate modeling and predictive process. Accuracy would imply that the applications were well-defined and integrated on the hardware and software to be used in production. When a computer system has been analyzed and its useful life (i.e., satisfies the service objectives of the user community) predicted, it is normally assumed that the system is generally "well-tuned."[3]

A spectrum of performance analysis techniques that can be used for capacity planning are outlined in Figure 10. As indicated in the figure, as one moves across the spectrum, complexity, cost, and man hours increase with respect to the technique used. It might

be expected that accuracy would also increase as one moves across the spectrum, for example, referencing benchmarking at the high end which is sometimes felt to be the most accurate modeling technique. In benchmarking it is necessary to select that subset of applications and workload from the data processing environment that characterizes the total operation. Then this subset must be processed on a computer system that is as much as possible like the actual hardware and software being proposed. After the benchmark has been accomplished, it is necessary to analyze the results and to determine what the results of this subset environment mean in terms of the total. In most instances, only installations with a great deal of understanding of their data processing operations can adequately use the results of benchmarking as their only capacity planning tool. Also, in many cases, the level of understanding that a data processing installation has about their environment would dictate a much less costly and complex means of analysis. It has been seen in several account situations that data processing installations using simple guidelines, monitoring their system on a continuing basis, and using linear projections for future requirements are doing very credible capacity planning. Also included in this spectrum of analysis techniques are statistical processes using linear time series or regression analysis. Queuing analyses depicted here are open (single/multiserver type) and closed queuing models.[19]

In many instances, the accuracy of open queuing models of computer systems is questioned. But there are many analyses or environments where the model accuracy is not the primary question because many of the parameters required to develop the model are suspected of being grossly inaccurate and in some cases unknown. What is needed at this point is a simple modeling technique (more functionally adequate than theoretically correct) that provides the capability of stepping a workload through the computer system and basically being able to relate to the relationships between the model and computer system. This analysis technique is viewed as being much closer to "guidelines" or an "operational" type of analysis. In modeling production computer installations, it is clear that the dynamics (continuously changing environment) and interactions (people, hardware, software, etc.) present such a complexity that, regardless of the theoretical accuracy of a modeling technique, it is impossible to model what is not understood. Hence, the capacity planning process is an attempt at gaining a better understanding of the overall production data processing environment.

Concluding remarks

The concepts and ideas of capacity planning may best be summarized by discussing how the process might be initiated and what

the final product of a capacity planning effort might resemble. These concepts were developed from experiences with large data processing installations over the last four years.

Before a capacity planning effort is initiated, the people required to staff the project must be selected. The areas providing the principal input for the development of the capacity planning process are:

- Operations department
- System programming (MVS, SVS, VS1, etc.)
- Applications programming (batch, TSO, IMS, CICS, etc.)

Therefore, the people selected to initiate the project should have some experience in these areas because development of the capacity planning process requires close coordination with each area.

There are no hard and fast rules as to the number of people required to begin the process or even whether a specific group of people should be set aside to perform the function. In many cases to date, organizations have selected one or two people and established a new department called the Capacity Planning Department. One person is selected to head the project and assume primary responsibility for the department activities. The individuals chosen are experienced data processing professionals and have a strong background in at least two of the technical areas cited above. This means the capacity planning group will begin with knowledgeable people able to provide a good interface to operations, systems, and applications departments.

As to whether one or two people are required, it seems that operations and systems or operations and applications expertise is required, which would imply that two would be the most reasonable requirement. Also, two people would give protection to the project as well as continuity if either person had to leave. The size of the account and data processing staff might dictate that only one full-time person is available. With the proper consultation, having one person initiate the capacity planning efforts is not unreasonable. This part of the process is concerned with understanding the current capacity planning efforts and developing a plan for enhancement or a new development. As the plans begin to be implemented, the capacity planning department will probably require additional people. But during the time plans are being developed, personnel needs may be outlined.

To initiate capacity planning in a data processing environment, several key items (Table 5) must be considered. Implementation of the capacity planning process requires a major commitment on the part of upper management within the organization. The coop-

Table 5 Summary: Plan for initial implementation of capacity planning process

1. Secure upper management commitment to the capacity planning project.
2. Understand management organizational structure and its impact on the development of a capacity planning program.
3. Outline in detail current capacity planning functions and the reporting processes.
4. Account for as much CPU usage as possible by subsystem (batch, TSO, IMS, etc.) as well as by major application area within each subsystem.
5. Establish specific user service objectives (response and turnaround times).
6. Perform some type of initial performance analysis and forecasting.
7. Define parameters for quantifying natural load growth for each application area in terms directly related to the values established in 4 above.
8. Define parameters for quantifying new applications for each area in terms directly related to the values established in 4 above.
9. Define a procedure for tracking performance on a continuing basis.
10. Establish report formats and define the required tools for implementing items 4, 5, 6, 7, and 8 above.

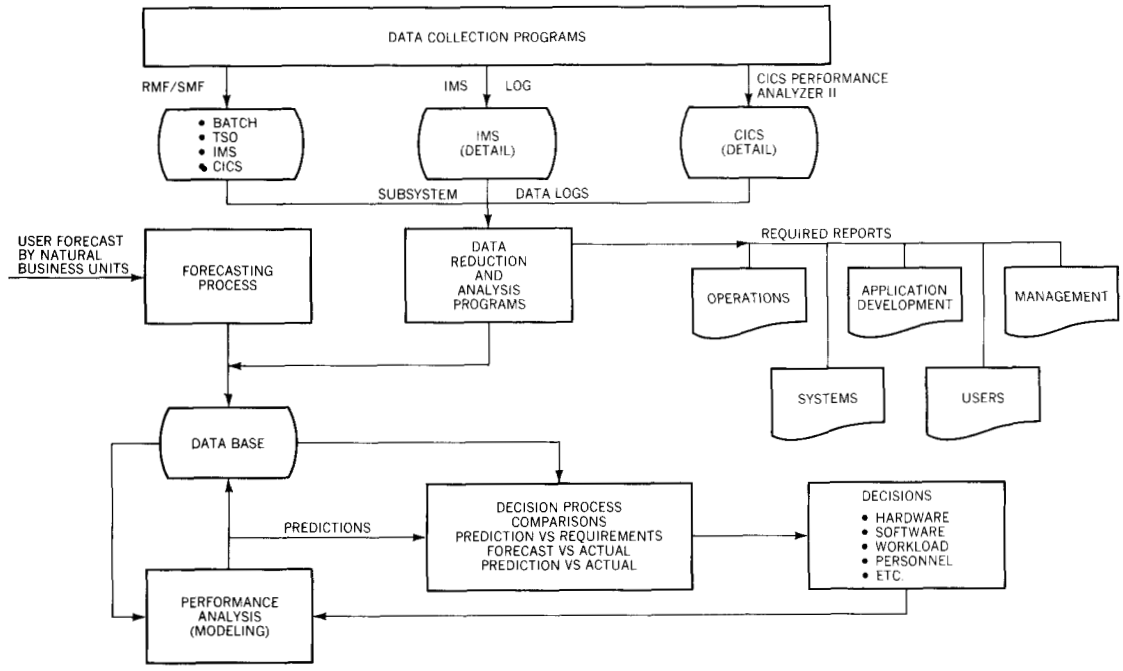
eration and coordination required among various departments (operations, systems, application development, users) to effectively develop a program will happen only if each department sees capacity planning as a major management commitment. Also, if these departments and management responsibilities are at separate locations or in different line organizations (divisional versus corporate), development of the capacity planning process may require a more complex structure than would be necessary if this separation was not present. Upper management must see that the right talent (people), hardware, and software are provided to implement the process. The capacity planning process must be installed and managed by the data processing installation. Consultation can be very helpful, but the task of implementation sits squarely on the shoulders of the *data processing organization.*

As a final product, capacity planning must be viewed as much more than a data gathering and performance prediction exercise. The process should be viewed as an integration of the following components:

- Data processing management
- Technical data processing personnel
- User community
- Computer hardware and software
- Measurement tools
- Data collection and reporting
- Workload characterization
- System modeling and performance prediction

These components should be systematically structured and controlled to provide an effective capacity planning program. One

Figure 11 Capacity planning process



example of a systematic approach to ongoing capacity planning is outlined in Figure 11. It is the user community that drives the data processing installation. As shown in Figure 11, the user workload would be forecast in Natural Business Units (NBUs), such as the number of new accounts, number of invoices, etc. This forecast is the input to a process designed to convert NBUs into Data Processing Units (DPUs), such as transactions per second, jobs per hour, earliest start time, latest end time, etc.

The process of selecting NBUs and converting them to DPUs and using the results for capacity planning is still very much an "art." For example, experience has shown the best approach for implementing such a process is by trial and error. By analysis and elimination, select those NBUs that appear to be the dominant ones (those NBUs that account for the major portion of the data processing workload). Then implement a plan for tracking the performance of the NBUs (current volumes as well as growth) against the DPU performance. If certain selected units do not appear to be dominant (DPUs are not tracking in any reasonable way with the NBUs), reassess your environment and make other selections. So much of the capacity planning process depends on "try it," "track it," and "change it." If, as shown in Figure 11, it is assumed that the forecast process is adequate and NBUs can be reasonably converted into DPUs, current data as well as growth factors over time are input to the data base.

Any capacity planning process requires measurement tools to collect, reduce, and report upon the required performance parameters. Specific requirements should be outlined for the timely collection and reporting of data (i.e., daily, weekly, monthly). Also, a customized set of reports must be defined for each area (operations, systems, application development, users, management). Certain data stored in the data base will be used for performance analysis of the computer system, namely, model development, model calibration (prediction of current versus observed), performance prediction (workload forecasts are inputs), and model validation (predictions versus future).

There are many different modeling techniques available, and several are discussed in this issue. A modeling effort may be viewed as having three phases—calibration, prediction, and validation.

The heart of the capacity planning process, as depicted in Figure 11, is the data base that contains the following data:

- Current and forecast workloads
- Current and historical performance data
- Performance predictions (calibration and validation data)
- Data for reports

When the capacity planning process has matured to the point characterized by Figure 11, the intent is for the process to be cyclic and ongoing. The process would be maintained and enhanced as management and technology dictate.

Experience gained by the author in a capacity planning role through involvement with many large data processing installations over the last four years has indicated that a good, operational capacity planning program is of paramount importance in understanding and managing today's complex data processing environments.

CITED REFERENCES

1. L. Bronner, *Capacity Planning, An Introduction*, Technical Bulletin GG22-9001, IBM Corporation (January 1977); available through the local IBM branch office.
2. L. Bronner, *Capacity Planning, Implementation*, Technical Bulletin GG22-9015, IBM Corporation (January 1979); available through the local IBM branch office.
3. R. M. Schardt, "An MVS tuning approach," *IBM Systems Journal* **19**, No. 1, 102–119 (1980, this issue).
4. H. P. Artis, "A technique for determining the capacity of a computer system," *Proceedings of CPEUG 76* (November 1976).
5. A. K. Agrawala, J. M. Mohr, and R. M. Bryant, "An approach to the workload characterization problem," *Computer* **9**, No. 6, 18–32 (June 1976).
6. D. Ferrari, *Computer Systems Performance Evaluation*, Prentice-Hall, Inc., Englewood Cliffs, NJ (1978), pp. 221–275.
7. D. C. Schiller, "System capacity and performance evaluation," *IBM Systems Journal* **19**, No. 1, 46–67 (1980, this issue).

8. P. H. Seaman, "Modeling considerations for predicting performance of CICS/VS systems," *IBM Systems Journal* **19,** No. 1, 68–80 (1980, this issue).
9. H. C. Nguyen, A. Ockene, R. Revell, and W. J. Skwish, "The role of detailed simulation in capacity planning," *IBM Systems Journal* **19,** No. 1, 81–101 (1980, this issue).
10. J. C. Cooper, "A capacity planning methodology," *IBM Systems Journal* **19,** No. 1, 28–45 (1980, this issue).
11. *OS/VS2 MVS Resource Measurement Facility (RMF), Version 2, Reference and User's Guide*, SC28-0922, IBM Corporation (September 1977); available through the local IBM branch office.
12. *SVS Performance Tool (SVSPT), Program Description and Operation Manual*, SH20-1838, IBM Corporation (November 1976); available through the local IBM branch office.
13. *VS1 Performance Tool (VS1PT), Program Description and Operation Manual*, SH20-1837, IBM Corporation (November 1976); available through the local IBM branch office.
14. *OS/VS2 MVS System Programming Library: System Management Facilities (SMF)*, GC28-0706, IBM Corporation (July 1977); available through the local IBM branch office.
15. *CICS/VS Performance Analyzer II*, SB21-1697, IBM Corporation (1978); available through the local IBM branch office.
16. *IMS Performance Analysis and Reporting System (IMSPARS)*, SB21-2140, IBM Corporation (1978); available through the local IBM branch office.
17. *Print Load Analyzer*, SB21-2356, IBM Corporation (1978); available through the local IBM branch office.
18. H. M. Stewart, "Performance analysis of complex communications systems," *IBM Systems Journal* **18,** No. 3, 356–373 (1979).
19. J. Buzen, "Analysis of system bottlenecks using a queuing network model," *Proceedings ACM SIGOPS Workshop on System Performance Evaluation*, Association for Computing Machinery, New York (1971).

GENERAL REFERENCES

1. A. H. Agajanian, "A bibliography on system performance evaluation," *Computer* **8,** No. 11, 63–74 (November 1975).
2. H. P. Artis, "Capacity planning for MVS computer systems," *Performance of Computer Installations*, North-Holland Publishing Co., Amsterdam (1976).
3. H. P. Artis, "Forecasting computer requirements: An analyst's dilemma," *Proceedings of the Symposium on Computer Resource Performance Management in Association With the Computer Society of South Africa* (April 1979).
4. Y. Bard, "Performance criteria and measurement for a time-sharing system," *IBM Systems Journal* **10,** No. 3, 193–216 (1971).
5. T. E. Bell, B. W. Boehm, and R. A. Watson, "Framework and initial phases for computer performance improvement," *AFIPS Conference Proceedings, Fall Joint Computer Conference* **41,** 1141–1154 (1972).
6. T. E. Bell, B. W. Boehm, and R. A. Watson, "How to get started on performance improvement," *Computer Decisions*, 30–34 (March 1973).
7. J. Boyce, R. Belhumeur, P. Raimer, and T. Shute, *IBM System/360, Tracking and Resolving Problems and Coordinating Changes*, Technical Bulletin GG22-9000, IBM Corporation (June 1975); available through the local IBM branch office.
8. G. Carlson, "Controlling reruns," *EDP Performance Review* **6,** No. 4 (April 1978).
9. *Proceedings of the CMG IX International Conference on Management and Evaluation of Computer Performance*, Computer Management Group (CMG), San Francisco, California (December 5–8, 1978).
10. *Proceedings of the Fourteenth Meeting*, Computer Performance Evaluation Users Group (CPEUG), Boston, Massachusetts (October 24–28, 1978).
11. R. F. Dunlavey, "Workload management," *EDP Performance Review* **6,** No. 4 (May 1978).

12. *Computer Performance Evaluation*, Conference Proceedings, European Computing Conference, London (September 1976).
13. C. F. Gibson and R. L. Nolan, "Managing the four states of EDP growth," *Harvard Business Review* **52,** No. 1, 76–99 (January-February 1974).
14. S. W. Heil, "One approach to the management of computer performance data," *EDP Performance Review* **7,** No. 1 (January 1979).
15. P. C. Howard, B. A. Stevens, and G. Carlson, "Evaluation and comparison of software monitors," *EDP Performance Review* **4,** No. 2 (February 1976).
16. P. C. Howard, B. A. Stevens, and G. Carlson, "A case study of turnaround and response time improvement," *EDP Performance Review* **5,** No. 2 (February 1977).
17. P. C. Howard, B. A. Stevens, and G. Carlson, "Bibliography of 1976 performance literature," *EDP Performance Review* **5,** No. 3 (March 1977).
18. P. C. Howard, B. A. Stevens, and G. Carlson, "How to get started in performance evaluation," *EDP Performance Review* **5,** No. 6 (June 1977).
19. P. C. Howard, B. A. Stevens, and G. Carlson, "Performance management information systems: State of the art," *EDP Performance Review* **5,** No. 7 (July 1977).
20. P. C. Howard, B. A. Stevens, and G. Carlson, "Bibliography of 1977 performance literature," *EDP Performance Review* **6,** No. 3 (March 1978).
21. P. C. Howard, B. A. Stevens, G. Carlson, and R. Dunlavey, "A data processing annual report," *EDP Performance Review* **6,** No. 10 (October 1978).
22. *Managing the Data Processing Organization*, GE19-5208, IBM Corporation (October 1976); available through the local IBM branch office.
23. J. M. Jenkings and P. C. Howard, "Measuring system capacity," *EDP Performance Review* **5,** No. 4 (April 1977).
24. G. M. King, *Graphic Throughput Analysis of Mixed Workloads*, Technical Bulletin GG22-9017, IBM Corporation (February 1978); available through the local IBM branch office.
25. P. J. Kiviat and M. F. Morris, "Getting started in computer performance evaluation," *Computer Measurement Group Transactions*, No. 10, 3.2–3.9 (December 1975).
26. R. Malick, "Systems performance/measurements—a quantitative base for management of computer systems," *Proceedings of the National Computer Conference* **43** (1974).
27. J. A. Morris, "Performance constraints in computer systems," *EDP Performance Review* **4,** No. 8 (August 1976).
28. J. D. Noe, "Acquiring and using a hardware monitor," *Datamation* **20,** No. 4, 89–95 (April 1974).
29. G. J. Nutt, "Computer system monitors," *Computer* **8,** No. 11, 51–61 (November 1975).
30. *Proceedings of the International Conference on the Performance of Computer Installations (ICPCI 78)*, Gardone Rivera, Lake Garda, Italy (June 22–23, 1978).
31. *Price, Waterhouse, and Co., Management Controls for Data Processing*, IBM Installation Management Manual GF20-0007, IBM Corporation (1976); available through the local IBM branch office.
32. *Proceedings Spring Technical Meeting*, Share European Association (SEAS), Berne, Switzerland (April 30, 1978).
33. *Computer Measurement and Evaluation*, Vol. III, Edited by J. Wixson, Share Inc., Chicago (December 1973–March 1975).
34. B. A. Stevens, "Audit and control of performance in data processing," *EDP Performance Review* **6,** No. 1 (January 1978).
35. C. E. Walston and C. P. Felix, "A method of programming measurement and estimation," *IBM Systems Journal* **16,** No. 1, 54–73 (1977).

The author is located at the IBM Washington Systems Center, 18100 Frederick Pike, Bldg. 2, Gaithersburg, MD 20760.

Reprint Order No. G321-5113.