RECORD OF SOCIETY OF ACTUARIES
1976 VOL. 2 NO. 4

EFFICIENT COMPUTER UTILIZATION

Moderator: ROBERT J. JOHANSEN. Panelists: WILLIAM O. BURNS, QUINTIN J. MALTBY, A. DOUGLAS MURCH.

1. What methods have been developed for maximizing return on development costs, including planning for computer utilization, and for evaluating the contributions of computers to increased productivity? Are there efficiencies in the purchase of application programs as compared with development in-house?

2. What has been the impact on efficiency of
   a. Specialized Operating Systems Software?
   b. Data Base Management Systems?

3. How are trade-offs evaluated in choosing between
   a. redoing existing programs for improved computer efficiency, and
   b. using the same resources to computerize operations being carried on manually?

4. How are charges allocated by line of business for
   a. unused computer capacity?
   b. planning and development for future hardware acquisitions and operating systems software?

MR. ROBERT J. JOHANSEN: Considering the hundreds of millions of dollars that have been projected for purchases of computers and software packages over the next decade, the millions currently invested in computers and software, and the additional millions spent on operating costs, it is imperative that stress be put on increasing the efficiency of computer operations. It is no longer possible simply to say that computers save money because it is cheaper to do a job on a computer than to do it manually. Computer operations in insurance companies are now of such size and account for such a large proportion of the company's expenses that an increase in the efficiency of its computer operations is more and more financially important.

Our speakers will give you the benefit of their experience with their own computer operations. All three are in key positions where they can keep track of computer costs and uses and can try to increase efficiency of their operations. Mr. Murch's comments in particular are pertinent with respect to cost control. Our hope is that you will go out of this session a little bit better equipped to increase the efficiency of your own operations.

MR. QUINTIN J. MALTBY: How does one maximize the return on development costs? The answer is simply to do the development quickly and correctly. Not as simple though is the process of deciding what is correct and then ensuring that it is done properly and on time. The key is to follow a comprehensive systems development methodology. This involves dividing the project into phases such as feasibility study, basic broad design, detailed design, programming and module testing, system testing, conversion and cutover, and finally the follow-up. The work should be so structured that a specific decision to go ahead, to recast, or to cancel is made at the end of each of the development phases. It

is important that this principle of "creeping commitment" be remembered and followed as it is all too easy to waste precious resources by acting too hastily. All of this is predicated upon intense user involvement throughout the process.

There are various "old" and "new" methods in use for organizing and performing the development work. One does not have to accept every facet of a method in order to materially benefit from its usage. One should become aware of how a new method solves problems that commonly arose from the use and abuse of older methods.

The planning for future computer utilization should be part of the overall corporate planning process. This computer planning must take into account new systems needed and the changing workloads of old ones. The methods used are highly dependent on the particular corporate planning process and the organizational realities in the various data processing functional areas.

Evaluating the contributions of computers to increased productivity is at best an inexact science. One has to cost out the alternative systems (whether actual or projected) as honestly as possible and then compare costs and benefits. In doing this it is wise to remember that the computer system which will best handle a function will have a rather different structure and a different set of flexibilities and rigidities than the manual system that best handles the same function.

Extensions and upgrades of existing computer systems can also have their relative productivities similarly evaluated. Any valid comparison must involve equivalent levels of service. Sources of data for these evaluations and comparisons include systems development costings, functional cost studies, time and motion studies, hardware costs, and operating costs. One often forgotten point is that the benefits of newly instituted computer systems come from two sources, namely the computer generated efficiencies and the reorganization of the application itself.

The decision to develop in-house or to purchase applications software is becoming more common as more good software is marketed and organizations increasingly scrutinize their overall software budget. By purchasing application software the development costs are in effect spread over many users, and a debugged and documented system may be available on short notice. However, before buying a good package, one should ask:

1. Does the package solve our problems - or just the vendor's?
2. Can the package readily interface with our connecting systems?
3. Can the package meet our needs with little or no modification? (In general, the more the modifications, the less sound is the decision to acquire.)
4. Are we prepared to make the necessary effort to learn how to use it properly - and then so use it?
5. Are we prepared to shoulder the maintenance load as if it were an in-house developed system, or to pay to have the maintenance done elsewhere?

Specialized Operating System Software can have an effect on efficiency. For example, I.B.M.'s Systems Management Facility, an option in some of the important operating systems, produces copious data relating to the jobs run, the job steps, all input/output activity, and so on. This data is used by the software support group to find out, among other things, the points of critical

loading of the system. This data is usable for tuning the system with respect to both operating system maintenance and production job setup. Tape management systems are another example. These systems schedule production tape usage, capture pertinent data about tapes when used, keep track of all tapes owned by the system, and protect taped data from unintentional destruction. The efficiencies resulting from tape management system usage show up most in the computer room itself, but the benefits of better control over and protection of the tape data are widespread.

An example of "application system tuning" may be of interest. Our individual policy file is maintained daily on a modified ALIS system running in a non-virtual machine. The file maintenance program structure is a dynamic overlay in which the high-level control logic loads the logic to process each transaction. The logic modules for many of the transactions are themselves structured as planned overlays. The monthly run in which all our Pre-Authorized Cheque (PAC) cases are "billed" took nine hours when running in a dedicated machine. Considerable modification of the list of transactions whose logic is made core resident and some of the planned overlay link edit structures reduced the time first to seven hours and then to five hours. After additional core was acquired and one of the more popular transaction modules was expanded in size so that it had no overlay activity, the PAC run timing was reduced to three hours despite a somewhat increased work load.

Data Base Management Systems have an impact on efficiency by allowing an organization's data to be independent from the application systems and programs that utilize it. The programs can then utilize the data according to the logic of the system at hand without being overly concerned about the format or file structure of the data. Data Base Management Systems impose an extra overhead burden on the hardware system. However, this burden is made worthwhile by the increased power, flexibility, and maintainability of the application systems that use the data base. The benefits of Data Base Management Systems do not come automatically. The Data Base Administrator function must be properly implemented and managed, the system itself must be maintained, and the application systems must use the Data Base Management System efficiently. It has been said that Data Base Management Systems will impact the Electronic Data Processing field in the 1980's to the same extent that operating systems have affected the 1970's. There are a number of different Data Base Management Systems in existence today. To choose one or to choose none at all is a significant commitment.

How does one evaluate the trade-off between redoing existing programs for increased efficiency and using the same resources to computerize manual operations? Although much technical input is usually required, the decision should be made by a manager who is at a level high enough to have a sufficient perspective of the issues involved.

The manager involved should ask:

1. With respect to proposals for rewriting an existing program or system:

    a. Does the system have to react to considerable maintenance activity? If it does, and if rewriting will improve the understandability, maintainability, and speed of the system, then rewriting is likely to be worthwhile.

    b. Will significant economies take place?

c. Does the system belong to an earlier generation of hardware? If so, it may well be a prime candidate for replacement and upgrading. In this case it is best to scrutinize the entire system so that the maximum benefit can be extracted from the current technology.

2. With respect to proposals for computerizing manual systems:

a. Does the manual system really merit computerization? The process of answering this question should involve a full dialogue between the application area and the data processing area.

b. If the manual system does not quite merit mechanization in its own right, does its inclusion as the completing part of a larger computer system result in overall economies? If so, it is a better candidate for action.

3. With respect to either type of proposal:

a. Is the system-supported function slated for expansion or extensive revision? If so, careful planning must be done to decide how to best handle the situation. This situation could well move the function out of the discretionary area and into scheduled development work.

b. Will the system in question be required for a long time? If not, it is best to leave it alone.

c. How does the suggested activity mesh with the other departmental priorities?

After the manager has answered the above questions, he can then analyze the comparative costs and benefits of each competing proposal, and decide what is to be done and when to do it.

MR. WILLIAM O. BURNS: My presentation will reflect, for the most part, my own company's approach and philosophy concerning efficient computer utilization. It is important that you understand our organizational structure so that my comments can be interpreted in the environment of that structure. Our Life Insurance operation is decentralized into 25 regional offices. Those offices are responsible for underwriting, issue, service, and claim activity for the policies assigned to the geographical region. The Home Office has staff functions and a central computer system that services all regional offices. All regional offices are connected to the central system by a teleprocessing network. All input to the system is generated in the regions and transmitted over leased lines, and most transactions update master records on line.

In regard to Topic 1, the following is a list of methods that we have used to maximize return on development costs with a brief comment about each method:

1. Design of Systems for User Efficiency - In designing a system, we have tried to make the input of data as simple and minimal as possible. Minimal amounts of data and simple format, of course, require more complicated programming. Simple formats and minimal amounts of input data lessen the chance for human error and place responsibility for editing on the system. People needs are reduced and efficiencies result.

2. Design of Integrated Systems - We have designed the system so that input of like or similar data is not duplicated for different uses. For example, data on a new business case would be entered once. From that one entry a Master Record would be established, data for the various agency runs would be developed, and data for various statistical reports would be determined. Rather than having numerous computer runs (30 or 40 as an example) in a daily system, the number of runs is held to a minimum, say fifteen. Obviously, better efficiency and better control result.

3. An Effective Organizational Structure - The company is decentralized into 25 administrative regional offices around the country. The computer system is located in the Home Office. In the Home Office we have a staff organizational structure which has been very effective. There is a Procedure Research and Development Group (PRAD) composed of an Office Systems Department which is responsible for writing office procedures, approaches, and methods; an Electronics Research Department which is responsible for developing specifications on how transactions and applications should be written and what they should do; and an Electronics Systems Department which is responsible for programming and implementing those specifications. PRAD Departments all report to the same Vice-President. The important point is that users are represented by the Office Systems Department and analysts from Office Systems work closely with those responsible for the programming activity. This approach has developed excellent rapport between the user department and programming activity. There is a team effort atmosphere. The result is a very effective and efficient system.

4. On Line Systems - Regional Offices in the company are connected to the central computer by leased lines. Records are updated on line. Paper shuffling has been greatly reduced. This approach provides fast response to error situations and tends to eliminate the compounding of errors that often occur in the batch-oriented type systems. Direct input without intermediate coding is also possible in many applications.

5. Involvement of User Departments - It is better for user departments to be involved in the beginning so that the new system provides the desired end result. Lack of involvement may mean wasted time in rethinking and redoing.

6. Modular Design of Programs - We develop modules or what I refer to as program segments which can be tied together in various conbinations when developing a new computer application. Take, for example, the module that would calculate a policy premium; once that module has been developed and tested, it will be plugged into any particular transaction, function, or application that requires a policy premium calculation. This approach minimizes programming duplication.

7. Use of Time-Sharing Terminals - The use of terminals in the debugging of application programs has increased efficiency tremendously. A programmer may condense as many as five to ten remote debugging sessions into one on a time-sharing terminal.

8. Establishment of Objectives and Priorities - Concrete objectives for the programming departments and a definite list of priorities are essential to an efficient and orderly operation. Do not jump from one effort to another. Avoid a "putting out fires" approach by planning and allocating resources ahead of time. Obviously, this approach takes cooperation from other disciplines and support from top management.

9. Project Coordinator Concept - On any sizable project, a project coordinator is assigned. A project team will be formed with all interested departments or parties represented on that project team; it is the project coordinator's responsibility to see that the project stays on time and all facets of the problem are covered. He does not have administrative control of the other project team members. He must go through his own administrative channels if the job is not being completed in a proper manner. This approach helps overcome the rigidity of the typical pyramid type of organization.

10. "Freeze" Concept - A control point is established in the implementation schedule after which modifications or changes in specifications for a particular application are nearly impossible to have approved. An extremely critical situation must exist before this rule would be ignored. This control point allows one to meet objectives and targets on time. Just as important, however, is the fact that users are forced to make decisions early in the game.

11. Use of a Consolidated Policy Master Record - All data concerning a particular policyholder is located in one record. Separate policyholder files are not maintained for different activities. This is a very important concept since neither data nor updating is duplicated. Control, therefore, is better and easier to maintain.

12. Involvement of People Knowledgeable in Both Data Processing and Life Insurance - One can do a data processing programming job in a better fashion and more efficient manner if people involved are knowledgeable not only in data processing, but also life insurance. Having an individual knowledgeable in both areas becomes more and more difficult as time goes along. An individual who is knowledgeable in both aspects is a very valuable asset.

13. Sticking to the Basic Requirements - An operation is much more efficient and economical if one uses only those parts of operating systems and other systems that are needed for what is wanted to be accomplished. Unnecessary frills should be avoided. Time is saved in development and less computer power is needed in operations. Systems programmers are not tied up with constant systems monitoring.

14. Use of Various Equipment Vendors - Efficiencies can often be realized by purchasing memory and other components from a vendor other than the vendor who supplies the basic computer. Costs may be lower and/or better equipment features may be available.

15. Purchase of Computer Equipment with Purchase Credits Available - When it makes economic sense, one can purchase equipment from a user who has purchase credits accrued. The user passes those purchase credits along to the company upon purchase and one can lease back to that user the computer system at more favorable rates than the user is currently paying. When the systems are needed by your organization, the systems are brought in at reduced rental rates to the company as compared to Vendor rental rates. Our company has two systems owned being used by original users and those systems will be moved into the company in the latter part of 1977 according to current plans and at a much lower rental charge than original equipment would carry.

16. Allowance in Planning for Sufficient Growth and Unexpected Requirements - Monitor where the company is going and what facilities will be needed to handle that growth in the future. Provide some slack for unexpected requirements. The point here is that even though you may have some excessive capacity you are not faced with having to interrupt important projects for a crisis situation, because you have room to function. The approach provides a more orderly and efficient approach to running the current system and developing new applications.

17. Minimizing Development Costs - Do a good job of analyzing the situation and seeing all its ramifications before you begin. A detailed set of programming specifications should be developed. One should try to anticipate all of the questions that will be asked by programmers. Less errors and less time to develop result.

    Try to keep major parts of the system independent. For example, in a teleprocessing system design the system so that either the central computer system, the teleprocessing network, or the terminal configuration at the end of the network can be changed without any great interruption of the other involved components.

18. Involvement of Top Management in the Decision Making - Top management must realize the ramifications of decisions or changes in priorities. Involvement is very important in keeping an orderly approach to development of systems.

Now, I will mention some methods we use for evaluating the contributions of computers to increased productivity. New business underwriting and issue are monitored in relation to the number of cases processed per employee. Likewise, the number of policies being serviced per service department employee is monitored and compared to previous experience. Data processing unit costs are monitored versus trends of other unit cost figures developed for the company. Finally, functional unit costs are developed annually and monitored against inflation, previous company experience, and experience of other companies.

Let us now consider the efficiencies and inefficiencies of purchased software. Generally, a software package can be obtained relatively fast compared to development of a similar package in-house. Programming anything takes a long time. Secondly, a software package can be obtained and implemented generally with less in-house resources devoted to the project. It seems obvious that costs should be less since a package supposedly spreads the development costs over many user organizations. The purchase of an application program also involves opportunity costs. If an application is developed in-house, the opportunity to do some other type of programming is lost. Perhaps an outside program can be purchased for considerably less money than what could be saved if the resources to develop the same program internally could be devoted to another application. Finally, there is an advantage in that the vendor generally has more technical expertise on the particular subject compared to the purchaser of the package. To develop in-house can mean a long and costly educational process.

On the other hand, there are disadvantages and inefficiencies. The software package purchased may not be exactly what is wanted or it may not yield the results one would demand if he were developing his own approach from scratch. The vendor may not be responsive to problems in the package or he may be slow in implementing changes that are required in the software package at some

future date. The vending company may not be around at some future date. The package may be developed on a generalized approach to be applicable to many situations and therefore may be considerably less efficient compared to one that could be developed in-house and tailored for the company's needs. Last, but very important, integration into the total system of the user company may be more difficult if not impossible with an outside package.

In regard to Topic 3, our method for determining what takes priority follows these lines. As long as we are providing good service to user departments and there is ample capacity in the system to service those departments, reprogramming takes a very low priority and using the same resources to computerize operations being carried on manually has a high priority. However, if running out of capacity and/or providing poor service to the user departments is facing us, the rewriting of existing programs or changing or reconfiguring the system to solve those probelems would move to the top of the priority list.

In setting priorities we also consider the expected rate of return on our investment of resources, the public relations aspect from the viewpoint of the user department, the agent, and the policyholder, and the requirements of any new products that need to be administered. When reprogramming is proposed, we also ask whether or not the reprogramming would provide better information to the policyholder, the agent, or the user department.

These considerations, of course, have to be weighed in the framework of available resources to do the job in the first place.

MR. JOHANSEN: I would like to interject a couple of comments at this point with regard to operating software. Back in the late '50's and early '60's, Metropolitan Life developed an English language compiler for its Univacs. Then the Company switched to Honeywell 800's which used transistors. The compiler was modified and the Company continued to use English language and the old programs. As we went from 800's to Honeywell 1800's to 8200's, our computer experts modified the Honeywell operating systems and the compiler Consequently the same programs could be used in going from one computer system to another. There was an obvious increase in efficiency in that higher powered machines could be used without redoing existing programs. It is important to note that all these machines were tape oriented.

This idea is very attractive and Metropolitan would like to be able at the present time to move programs from one manufacturer's computer to another's but without having to modify the programs because of the underlying software change. Our computer people have developed what is termed a DMI, a Data Management Interface, to be used between the computer's operating system and the data processing application program - the insurance program. The DMI would also be used between the insurance program and a Data Base Management System. The purpose is to avoid rewriting programs because of changes in equipment or a change in the Data Base Management System. It effectively isolates the programmer from being concerned with Input/Output or record fields.

Suppose you have a policy file with defined fields and have programs written to operate on it. Then suppose you add a new plan which requires that your policy file have a new field. Generally you would have to modify all your programs, but with the DMI this is not necessary. You can ignore the fact of the field change and just modify those programs that reference the new field. It has given Metropolitan a tremendous boost in programming efficiency.

Just to illustrate the DMI operation: a new Data Base Management System was installed a short time ago - IBM's Amigos. Modifications were made in the DMI and no program changes were necessary. We checked out Amigos on one program - a Check-o-Matic program - and found we achieved a 20 percent reduction in time. For a company the size of Metropolitan this amounted to $55,000 on an annual basis. Some of these efficiencies do add up.

MR. A. DOUGLAS MURCH: In the field of insurance administrative operations, improving productivity usually means handling more work with fewer people, or with much less than a proportionate increase in people. I think most of us would agree that, of all tools that exist for improving productivity, the computer is the most powerful.

Broadly speaking, there are two methods by which computers are used to improve productivity in administrative operations. I will refer to these as Method 1 and Method 2:

> Method 1 consists of developing and installing new or expanded computer applications, which result in transferring work from people to machines. This tends to reduce clerical staff and salary-related expenses, although it tends to increase program development and maintenance staff, and machine-related expenses.
>
> Method 2 consists of improving the efficiency of computer operations and systems. This tends to increase the amount of useful work performed by a given computer installation, and thus tends to reduce unit costs for work performed by computer.

Our topics for discussion this afternoon span both Method 1 and Method 2. I stress the distinction between the two methods because, for many years, there was a tendency in most companies towards almost exclusive concern with Method 1, on the assumption that equipment and software improvements by vendors, plus natural initiatives by the data processing staff, would take care of Method 2. However, as companies have developed large computer workloads, annual costs of computer operations have risen, and more and more companies have come to realize that substantial opportunities for productivity improvement exist through Method 2.

Maintaining proper emphasis on Method 2 efficiency improvements is sometimes difficult. Most outside pressures on computer systems staffs are for more projects of the Method 1 type. Most outside pressures on computer operations managers are for maintaining high standards of service and quality, rather than for making Method 2 efficiency improvements.

I will describe in brief, simplified terms a few mechanisms the Prudential uses to encourage efficiency improvements of both Method 1 and Method 2 types. These are very much integrated with our overall expense management and budgetary processes, and so I need a word of explanation about expense control before we proceed.

All budgeted expenses in Prudential are classified into what we call "Activities". Each Activity includes all types of budgeted expense - salaries, benefits, rent, telephone and telegraph, postage, machine expense, and so on - and it includes all expenses incurred for the Activity in Regional Home Offices as well as the Corporate Office. There is accountability for Activity expense results by Organization. We also name an "Activity Head" who is in charge of expense control for each Activity.

The Activities are defined either along major functional, or else branch of business, lines. While we have over 20 Activities, the two Activities most relevant to our discussion today are

first, the Individual Insurance Administration Activity, which includes home office administrative operations for individual life and health insurance, from underwriting through claim, for a 1976 budget of $134 million, and

second, Data Processing, which includes expenses for operating all of our computer centers throughout the Company, for a 1976 budget of $40 million.

I am the Activity Head for both these Activities. It is my responsibility to establish budgetary "guidelines" or "constraints" each year for each organization throughout the Company which incurs expense for these Activities, to follow for budget variances during the year, and in general to exercise Corporate responsibility for the expense performance of these Activities.

In the Data Processing Activity, we include all expenses incurred in operating our computer centers - equipment, manpower, supplies, space and other expenses. The Data Processing Activity does not include salary-related costs for development and maintenance of computer applications. This is charged to the various other Activities, depending on the purpose of the particular application.

For example, the Individual Insurance Administration Activity includes this salary-related systems development and maintenance expenses for systems related to Home Office administration of individual life and health insurance. The Group Insurance Activities include it for development and maintenance of Group Insurance systems. The Investment Activities include it for development and maintenance of Investment systems, and so on.

Thus, in Prudential, basic Corporate responsibility for initially authorizing, and subsequently following, the expenses for systems development and maintenance is in the hands of the Activity Head, who in each case has expense responsibility for all budgeted expenses for that Activity, both line and staff, and not just the systems development and maintenance expenses alone.

Assigning this expense responsibility to the Activity Head is an important way in which we both stimulate and control efficient computer utilization through Method 1 - that is, through developing and installing new or expanded computer systems. We feel that giving to the Activity Head, who in each case is a senior Corporate officer, the expense responsibility not only for current operations of that Activity, but also for development and installation of future computer systems for that Activity, leads to the best resolution of conflicting priorities, and the best overall return on development and installation costs.

Each Activity Head is interested in not just one, but all 3 of what I consider to be the most important facets of Method 1 efficiency improvements, namely:

(a) initiating new systems developments to enhance the expense performance of the Activity in future years,

(b) controlling and following on development and installation expenses to make sure that they are within budgeted limits,

(c) following through, after installation, to make sure that systems are working properly, that necessary systems improvements are made, and that maximum clerical savings are achieved.

As for Method 2 efficiency improvements, that is improving the efficiency of computer systems and operations, our methods are somewhat different. Responsibility for stimulating and following on these efficiency improvements falls, for the time being at least, primarily on the Activity Head for the Data Processing Activity, since that Activity by definition includes all expenses for operating our computer centers.

Over the years, it has been a common phenomenon for computer centers to be equipped with successively more and more powerful and expensive equipment. About 5 years ago we decided that, in view of the importance to the Company of its computer centers, we needed some realistic measure of the efficiency of operation of each center. Expenses for each center were already accounted for in the Data Processing Activity, but we had no reliable measure of the workload performed by the center, other than such things as job counts, elapsed running times, idle times, snapshots of CPU utilization, and so on.

What we set out to develop, and subsequently have implemented, is a measure of workload, expressed in terms of billings, that satisfies three objectives:

(1) It includes all useful work performed by the center, both production and testing, on line and batch, whether performed on large multiprogramming computers or smaller, stand-alone or "dedicated" computers.

(2) It excludes all work of a vendor software overhead nature, such as running IBM control programs like OS-MVT, ASP and VS.

(3) It bills work using a set of fixed computer center rates, called "standard rates", which are applied to the units of computer center capacity required by each job performed in the center. The sum of the billings for all jobs performed in a given period of time represent the "income" of the center for that period, and are compared against the expenses of the center to determine a "profit" or "loss" for the period.

We further have a structure of "standard rates" that we believe fairly reflect the demand of each job upon the resources of the center - machines, people and supplies. This point is important because we want to create incentives, not just for full utilization of computer center resources, but also for minimizing the amount of computer center resources required to perform a given job.

This system of job billings has been in effect in our computer centers for several years, and results in computer-produced reports of billings and expenses broken down by major application user, and by individual job. The existence of these reports has had a considerable beneficial influence in stimulating efficiency improvements of the Method 2 type.

For the manager of the computer center and his staff, it serves as an income, expense and "profitability" statement, with standard billings representing the income. One objective of the computer center manager is to increase his "profitability", which he can do either by reducing expenses or, more likely, by making sure that expenses increase less rapidly than do billings.

To aid him in pursuing this objective, he is encouraged to analyze aspects of this computer center operation where utilization is low, where equipment could be released, or where further efficiency improvements could free up capacity to handle additional work. Regular analyses of such areas include storage and utilization of data sets on tape and disk, blocksize factors, channel and control unit utilization, CPU utilization by time of day, job scheduling characteristics of printer utilization, number and frequency of tape and disk set-ups, reruns, restarts, etc. Improvements in such areas are often substantial. They usually lead to being able to absorb an increasing workload in the computer center without anything like a proportional increase in expense, or without major equipment upgrade.

For the director of each systems and programming project, the reports of computer center billings and expenses provide a somewhat different, but equally important, means to pursue Method 2 efficiency improvements. For him the reports show the standard billings incurred for each major system developed or maintained by his staff, with production and testing shown separately, in each computer center in the Company. In addition, unit costs are shown for production work. These unit costs are obtained by dividing work volume counts - usually transaction or in force counts - into standard billings, with billings factored down to a "zero profit", or actual expense, basis.

The director of each systems and programming project can, therefore, follow the expense and the unit cost being incurred in each computer center for the systems which are his responsibility. He can compare the efficiency of the various centers in running his applications, and he can follow, over time, the trend of expenses and of unit costs for operating his systems.

Needless to say, these billing, expense and unit cost figures form the basis of a Management By Objectives program aimed at reducing billings and improving efficiency and unit costs for applications where that action is in order.

For most of our larger systems in production, computer center costs, as measured by "zero profit" billings, exceed by a considerable margin the salary related costs of systems and programming. We have tried very hard to impress upon our project directors that they are responsible not just for developing systems that result in clerical savings, and for controlling their own staff costs, but also for controlling costs incurred in computer centers for their systems, both testing and production.

We have the usual hardware and software monitors available to indicate areas of programming inefficiency. Frequently, a modest amount of properly directed programming effort can yield a very high return in terms of reduced billings. Typical target areas for this effort are better overlay structures to reduce disk calling and channel contention, eliminating grossly inefficient coding indicated by high CPU utilization, pre-compiler screening and elimination of inefficient coding sequences, and improved testing techniques.

Finally, on the question of allocating costs for unused computer center capacity and for future hardware and operating system software planning, we basically distribute these costs by line of business in proportion to the actual use of computer center facilities. We use the distribution, by line of business, of actual computer billings during the period, to perform this allocation.

MR. JOHANSEN: Now I have a couple of comments on Topic 4. A few years ago I had the job of trying to coordinate the Company's various computer operations. We had a central EDP system for Personal insurance, we had another for Group, and we had various other "stand-alone" systems throughout the Company. As a result of the studies made at that time, we tried to bring all the independent systems into the one system. One of the problems with a merged system is that when you have to charge your computer operations over several different lines - Group Life, Group Health, Personal Life, Personal Health - you have the problem of charging for underutilization of the equipment.

It is a knotty problem. If you charge for use on a final basis at the end of the year, then you can spread the underutilization over all users on the basis of their use charges. You may however get complaints from some of your users. For example, your Group line has quoted on a particular contract on the basis of so much for each claim processed by EDP. Then they find at the end of the year that they are being charged a larger unit cost because another computer was purchased during the year.

We have been examining this problem to find an equitable way of handling it. One method is to operate the computer center as a profit center and overcharge everybody. If there is a profit at the end of the year, a way must be found to spread the profits. Another way might be to make a small initial charge to each user and accumulate a fund which is maintained from year to year. In some years it will increase and in some years it will decrease depending on overutilization credits or underutilization charges, but the benefit is that there is a predictability of charges in advance. People can budget their EPD costs. You could also say that the acquisition of a new computer is for the good of the company. In such case the underutilization cost would be spread as general company overhead. We haven't settled the problem - we are still working on it. Probably a lot of others are too.

MR. W. DUANE KIDWELL: In a small company, we get sales pressure for a wide variety of types of products. Since computer systems are generally less flexible than manual systems, it often becomes extremely expensive to accommodate the product design people. Do any of you get involved back at the product design stage in an effort to control the product design to fit your system and thus cut down the cost of subsequent administration?

MR. BURNS: Yes. For example, our company is currently redesigning to come out with a new line of individual policies. Right now new business can be entered into the system directly without intermediate coding; the entry of data is very easy for clerical people. I remind those involved that if this application is touched, a lot of expense will be incurred to reprogram that whole activity. They listen.

Top management needs to realize that something that looks very difficult to someone outside the EDP area may be very easy to accomplish while something that appears easy may actually be difficult. Thus, involvement with the people making new product decisions is very important.

MR. JOHANSEN: We have purchased some software to handle some new products in both the life and casualty areas. With the design of the software package in mind, the product can be designed so as to minimize modifications. On the other hand making a product more saleable is very important too.

MR. FRANCOIS VACHON: I have a theory (which is impossible to apply) for charging for unused computer capacity: the users who should be charged for unused capacity are those which have applications that are not on the computer but ought to be. The problem is that you cannot make a list of such applications.

MR. JOHANSEN: We have been thinking about that idea too. Perhaps that is one of the arguments behind spreading the cost of unused capacity on a corporate basis without regard to whether or not one uses a computer; eventually almost everything gets on the computer.

Suppose that a Product A is going to be computerized in 1976, and that an estimate of required time has been made. Then for some reason the job gets postponed to 1977 or 1978. Meanwhile, since one cannot buy a computer and get it installed at a moment's notice, we are left with the extra time. The problem is that someone has to pay for it. We have a similar problem with time-sharing, and we have not yet found a solution that is acceptable to very many people.

MR. MURCH: One of the results of our income, expense, and performance measures has been that for several years, in spite of a steady annual growth in our workload, we have not, at any point in time, had significant amounts of spare capacity. With the hardware options currently available, one can add capacity in small increments and get within reason almost any desired increment in capacity. Further, our method of distinguishing between dedicated and nondedicated charges is geared to discourage the situation of an entire computer being used for a single application. An application area which wants to "strike off on its own" in terms of a new type of computer or a new type of software will have to shoulder the full cost from the start. This is basically a deterrent for proliferating different varieties of software and hardware.

MR. JOHANSEN: On the question of allocating expenses, we have three large computer centers with a system of alliance between the centers such that if Center A has an overload, some of the work can be sent by wire to computer center B or C or both, processed there, and then sent back to the users. The user may not know at any time which computer is processing his work. However, it would seem unfair, if center A is overloaded and center B is underloaded, that the particular user whose job is being shipped from A to B should pay the full leased line cost. So the leased line cost for that kind of operation is spread over all users without regard to where their work is done.

MR. THOMAS TIERNEY: Currently, one of the biggest debates in the computer industry is distributive processing vs centralized processing or, if you will, mini-computers versus large scale computers. Are there any comments on this?

MR. MALTBY: I cannot say much about it except that I am a centralist. I have seen arguments for having a "mini" for this job, and a "maxi" for that job, and a "micro" for another job; but with the nature of our business one still has to tie everything together, and a large CPU computer does the job. Of course, this does not mean you should not have lines out to your various branches and centers.

MR. BURNS: We have just the "central system with terminal" arrangement in the administrative offices. I currently believe that the central system is the most efficient. With a number of computers in different places, there is the

tendency for people to want to write different programs with different modifications in different places. Control is lost, and maintenance can be a real problem.

MR. ROY ARTHUR SAUNDERS: We are thinking of going to distributive processing. For my company (in contrast to a large company with so much existing computer power that a relatively small percentage increment can be easily achieved) an increment of one good mini-computer is a substantial increment in computer power. We are using a ten year old 360-30 and now require enhanced capacity. One of the options we are considering is acquisition of one or two "minis", originally for dedicated operations. But they will be under control of the EDP department. It will just be a dedicated small system which will be integrated with the rest of the system with all operations and programming under the standard procedures of programming management. User programs will be run on this "mini", but if there is excess capacity, it will be available to other users. I essentially view the distribution as a functional distribution of actual physical computer power rather than a distribution of all the functions of EDP systems, programming, and operations.

MR. JOHANSEN: We are currently handling group dental claims in two ways: one is with a CRT on line to a large computer; the other is a mini-computer on site, which then connects with a large computer after having completed preliminary processing. We will compare the two systems; it would appear that the CRT has an edge.

John Hancock uses a system for programming for new applications called HIPO. Would someone from the John Hancock like to comment on that?

MR. TIERNEY: Machine costs have been decreasing relative to personnel costs. Since machine costs are more measurable, they have tended to be the focus of our interest, but we need to force ourselves to look toward people productivity. That leads us to systems development and system maintenance costs which in turn brings us to the product and how it is built. One of the methods of doing it better is the IBM HIPO (Hierarchical Input-Process-Output) system.

HIPO is really just systems theory applied to computer programming development. It involves structuring the development process through a rational deductive well-thought-out approach.

MR. JOHANSEN: It seems to solve the problem of making sure that both the programming group and the user group know, understand, and define precisely what will be done.

MR. TIERNEY: Yes. The users are forced by the nature of the HIPO system to get involved much earlier. This plays a very large part in controlling costs. Problems are precisely defined. Quality criteria are set - such as functional independence, interchangeability, and size. If the developed system then meets those criteria, it will work better, cost less money, and be more easily maintainable.

HIPO is really predocumentation. It involves the walkthrough, simply a process of writing a particular systems process on a blackboard and then going through it step by step, and, in effect, troubleshooting.

Another method in this area is egoless programming, which is a process of taking computer development from being one's own private art form to a kind of blueprinting, or public art form. The programmer displays his coding or ideas on a screen, and allows several people to have a benevolent, well-intentioned, "shot" at them. This usually results in a much better product.

It is important to remember that tools such as these are only means for developing a better product and that one must be careful not to let the means become an end in themselves. There are many situations where their use is either inappropriate or not necessary. HIPO, for instance, has proven to be very useful when working with monolithic second-generation languages such as COBOL or FORTRAN but it is only marginally useful with the more advanced, segmentable and self-documenting, third-generation languages like PL/1 or ALGOL 68.

The focus in the developmental process should not be on the means but on the end-product itself - i.e. (1) a system superstructure where modules are engineered together in the simplest (as determined by systems theory formulae) possible scheme, (2) individual modules that have a good internal coding structure and the proper external attributes for system linking, and (3) documentation that is unified and as brief as possible.

On the question of mini-computers, it is my opinion that they will become an increasingly important part of insurance company administrative procedures. Just how important, however, is something which I find very hard to guess at.