RUNNING A SMALL-SHOP INFORMATION SYSTEM
BASED ON THE USER/PROGRAMMER CONCEPT

Eric W. Roberts
Information Resource Manager
The Audichron Company

INTRODUCTION

The purpose of this paper is to show how an Information System can be set-up and run successfully at minimum cost in a small to medium-sized company. The ideas presented can be used either by a company with an existing DP shop, or by a company just starting one up. The user/programmer concept is based on use of the following resources:

- Purchased, integrated applications software.

- Fourth generation languages.

- User/programmers who write their own report programs.

- Departmental Computer Coordinators.

We will take a look at the three main ingredients of the system: software, hardware, and people, and then outline some helpful rules of the road that help to maintain control in this unique environment.

SOFTWARE

Start off by purchasing a good family of integrated applications software packages, all from the same vendor. Don't try to buy an MRP system from one vendor and try to glue it to someone else's order entry package. Take plenty of time to evaluate several packages, and make sure the evaluation includes at least one hands-on session. The most important criteria, once you get past the price tag, is ease of use. Other important factors are vendor technical support and the life expectancy of the vendor.

Next on your shopping list is a package of fourth generation language (4GL) products which should include as a minimum a report generator, a screen-oriented data entry and lookup tool, and a data dictionary. Shops with high volume transactions should consider a batch data entry product. Select your 4GL products from the same vendor to make it easy on yourself when it comes time to train your users.

There is no family of applications software that is going to meet ALL of your users' needs, so you will have to develop a few small applications in-house, preferably using IMAGE databases along with your 4GL toolkit. Rely on your users to develop these applications for you; more on that in the "PEOPLE" section later.

Round out your software inventory with a couple of purchased housekeeping utilities like ADAGER to pull maintenance on your databases, and MPEX to keep your discs tidy. Packages like these are inexpensive, and will pay for themselves quickly.

Finally, you should purchase software maintenance agreements from each of your vendors. You will not have the time nor the people required to maintain your purchased packages in-house.

Figure 1 illustrates how users in a typical small-shop company access the databases using a combination of 4GL products and off-the-shelf software. The example portrays the 4GL products of Quasar Systems, Ltd., and the MANMAN family of applications software from ASK Computer Systems, Inc.
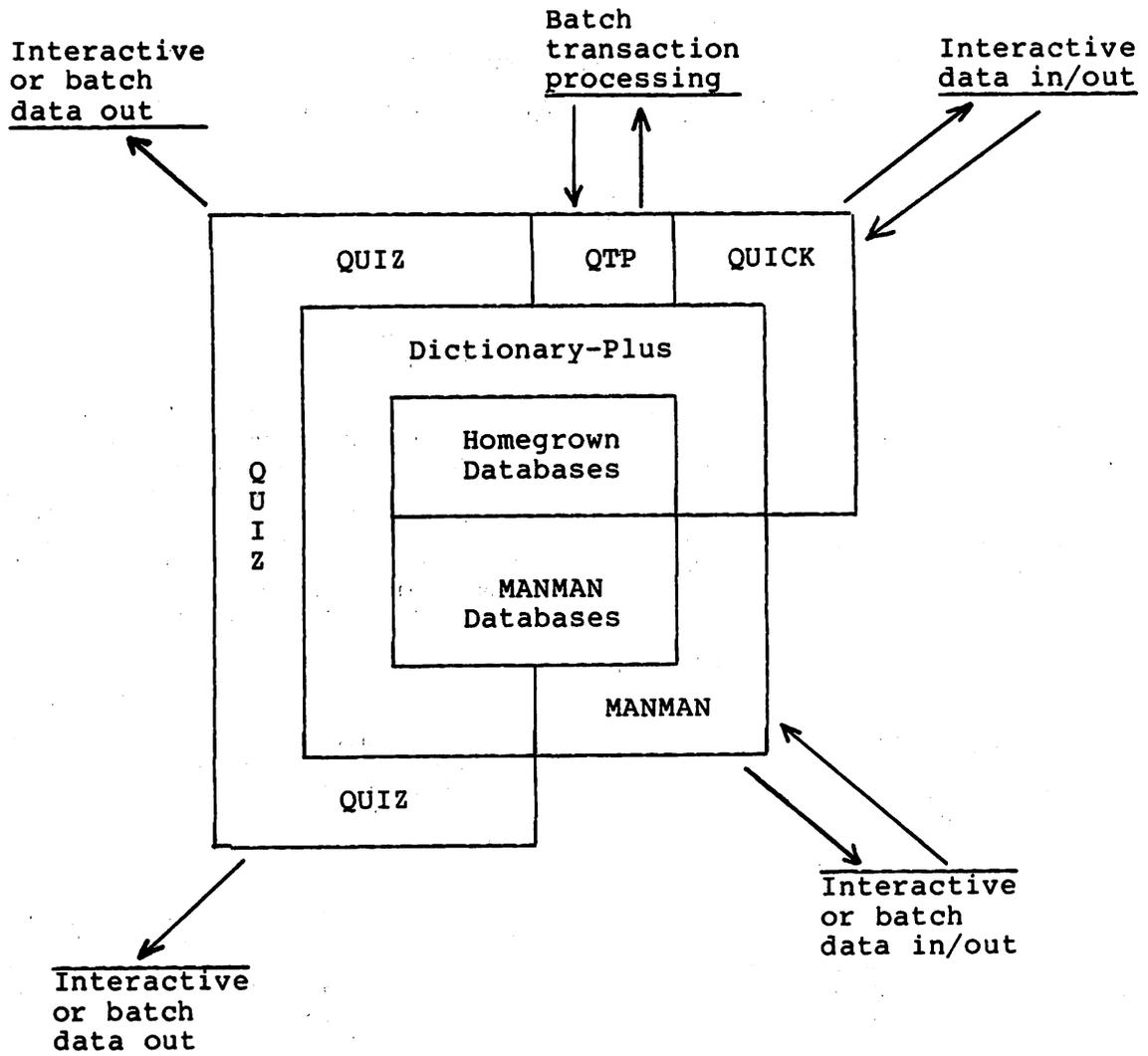
Batch
transaction
processing

Interactive
or batch
data out

Interactive
data in/out

QUIZ   QTP   QUICK

Dictionary-Plus

Q
U
I
Z

Homegrown
Databases

MANMAN
Databases

MANMAN

QUIZ

Interactive
or batch
data out

Interactive
or batch
data in/out

Figure 1
Example of Combination of Purchased and Homegrown Databases
and Methods of Access

HARDWARE

Buy only as much mainframe hardware as you need now, plus what you foresee within a horizon of one year. Don't buy a 68 now and wait three years only to find out that you could have gotten by with a 48. The architecture of the HP3000 lends itself very nicely to future expansion, since HP has pursued, and hopefully will continue to pursue, a policy of upward compatibility in both hardware and operating system.

The same principle applies to peripherals such as line printers, tape drives, and disc drives. Buy enough disc space to let you run at not more than 80% capacity over the next year. It's easy enough to add on extra capacity when the time comes. With a small-shop IS, whole idea is to keep every aspect of your system as simple as possible to minimize workload on your staff, and dollars in your budget.

If your users are far-flung and you have to get into communications gear such as multiplexers and modems, be sure to buy equipment that is reliable and guaranteed compatible with the HP3000. You will have enough problems with the computer itself and the people using it without having to be bothered with data communications failures.

In the area of terminals, purchase only high-reliability CRT's and printing terminals. The extra reliability will cost extra dollars, but you'll be thankful you made the additional outlay up front. Unreliable terminals cause user frustration and higher maintenance bills, not to mention the time it takes your tiny staff to do the initial troubleshooting.

Let each department purchase their own terminals, but give them a list of approved terminals from which to choose. If you buy their terminals for them, you'll spend lots of time chasing down promise dates from the vendor every time a user calls you to ask "Where's my terminal?". Also, if a department owns a terminal, they're much more likely to be conscientious about taking care of it. The System Manager must rule over the allocation of new ports with an iron fist. Don't let the idea take hold amongst your users that terminals are like telephones, and one belongs on every desk. Try to make sure that terminals are placed in common areas where multiple users can access them, not on a secretary's desk where she is a dedicated user. Dedicated terminals should be placed only at the desks of heavy-duty data entry people, such as stock clerks and order takers.

Some of your users will be quick to try to talk you into buying some microcomputers, since they are the fad now, and all the trade magazines have pictures of executives with their micros on their credenzas. Beware the micro trap! Install micros ONLY where they are needed for very specialized applications that the HP3000 doesn't handle well. If you do install micros, assure that they are 100% capable of communicating with the HP3000, to include uploading and downloading of files. The HP120, HP125, and HP150 micros are good examples of machines with total HP3000 compatibility.

PEOPLE

The first "people" thing that has to happen is the formation of a small IS staff. An existing company with a large IS staff will have to trim down some excess fat if they are to adopt the user/programmer concept. Large shops are going to tend to want to justify their continued existence, so they will probably demand that the purchased packages be modified so they'll more closely resemble the in-house system that is being phased out. Top management will have to be aware of this natural tendency of self-preservation, and act early to counter it when organizing for a small shop.

Figure 2 shows a typical placement of the IS function within a small to medium-sized business. It is important that the Information Resource (IR) department not answer to one of the using department managers, so as to assure objectivity of the services rendered by IR personnel. It is also important that the Information Resource Manager (IRM) be placed on the same level as other department managers to assure that he will have sufficient clout to implement IS plans and policies company-wide. In a medium-sized company there may even be a Vice-President for Information Resources answering directly to the President, which is the ideal situation in terms of implementing an integrated corporate Information System.

```
                          ┌─────────────┐
                          │  President  │
                          └──────┬──────┘
          ┌──────────────────────┼──────────────────────┐
   ┌──────┴──────┐        ┌──────┴──────┐        ┌──────┴──────┐
   │     VP      │        │     VP      │        │     VP      │
   │  Support    │        │  Marketing  │        │ Operations  │
   │  Services   │        │             │        │             │
   └──────┬──────┘        └─────────────┘        └──────┬──────┘
          │                                   ┌─────────┼─────────┐
          │                              Quality    Engineering  Manufacturing
          │                              Assurance
   ┌──────┼──────────┐
Personnel  Finance  Information
                     Resources
```
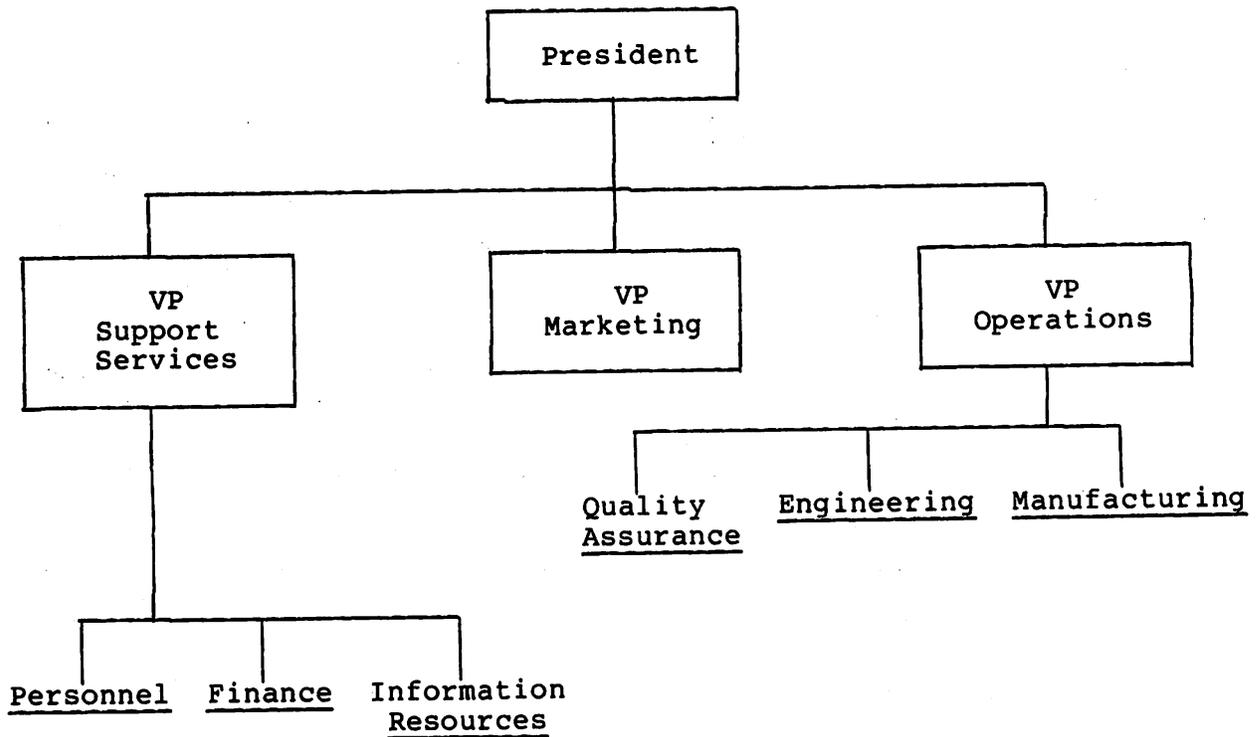
Figure 2
Typical Placement of Information Resources in the Organization Chart

Figure 3 shows the bare minimum IR staff required to get the job done with the user/programmer concept. Some larger companies may need to change the Programmer/Analyst slot to Systems Analyst, and put one or two Programmers under the Systems Analyst. It would be nice to have a full-time department secretary to help the System Administrator with documentation and correspondence. In any case, you get the point:

keep the staff trimmed down to the bare minimum required. Don't fall into the trap of thinking that a company with X employees should have an IR staff of size Y, replete with the traditional cubicles of Programmers and Analysts. That is the old in-house MIS way of thinking. The new Information Resource concept, based on the user/programmer, is a relatively new way of thinking in the field of data processing.
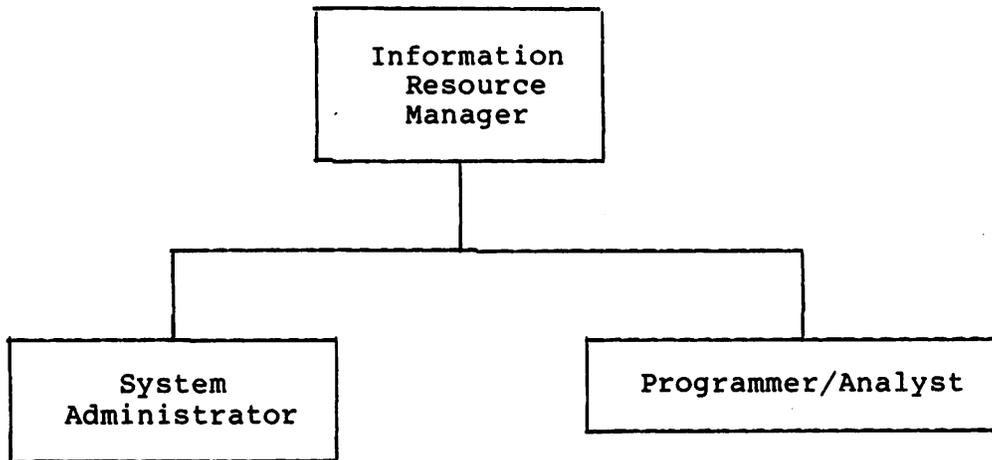
48-4

```
                    ┌─────────────┐
                    │ Information │
                    │  Resource   │
                    │   Manager   │
                    └──────┬──────┘
              ┌────────────┴────────────┐
    ┌─────────┴─────────┐     ┌─────────┴──────────┐
    │      System       │     │ Programmer/Analyst │
    │   Administrator   │     │                    │
    └───────────────────┘     └────────────────────┘
```

Figure 3
Orqanization of a Small Information Resource Department

Following are abbreviated job descriptions for the IR staff:

Information Resource Manager:

- Serve as System Manager for the HP3000.

- Coordinate all off-line information systems such as word processing, TWX/TELEX, and microcomputers.

- Publish and maintain the Company Procedures Manual.

- Conduct all in-house training for use of the HP3000.

System Administrator:

- Operate the HP3000 computer, with System Manager capability.

- Create and purge accounts, groups, and users.

- Maintain all departmental records.

- Manage the tape archives and Contributed Software Libraries.

- Publish usage reports and traffic studies.

- Place service calls to HP Customer Engineering.

Programmer/Analyst:

- Develop and maintain major in-house applications.

- Act on users' Requests for Service.

- Write system utilities as directed by the IRM.

- Install patches as directed by vendor technical personnel.

So let's assume that the staff shown in Figure 3 is installed and we're ready to proceed. The next "people" ingredient that is needed is a Steering Committee, composed of top management plus the Information Resource Manager. The President, or the person to whom the IRM reports, whichever is higher, should chair the committee. The IRM should NEVER chair the committee, so as to maintain a system of checks and balances. The committee should also contain at least one employee who is technically savvy regarding the HP3000, and who does not work in the Information Resource department. This accomplish two objectives. First, it gives the committee another angle of expertise when technical issues arise. Secondly, since the IR staff usually works in a tight concentration around the computer room, a major catastrophe could take out the entire pool of Information Systems expertise in one fell swoop. At least one person physically removed from the IR department should be kept continually in the know as to the daily operations of the HP3000, and should also sit on the Steering Committee.

The Steering Committee should meet monthly or quarterly and develop and maintain long-range plans and goals for the corporate Information System. The IRM should not report to the committee, since committees usually make for lousy bosses, but he should be receptive to the committee's input. Conflicts should be resolved by the chairman, who is usually the IRM's supervisor anyway. Since the IRM will be doing most of the reporting at the meetings, he should serve as the committee's secretary.

Training is a key facet of the user/programmer concept. A lot of people have to be trained in a lot of different areas. Before laying out the training plan, the idea of the Alternate System Manager and the Departmental Computer Coordinator has to be explained.

Once the staff is in place, the IRM must select certain individuals to serve as Alternate System Managers (ASM's). An ASM is one who would take over System Management duties in the event of the IRM's temporary or (heaven forbid) permanent absence. Top management must assure that the IRM is not allowed to be the only employee carrying all the information eggs in his basket. Some ASM's must be trained to take over in case the proverbial truck catches up with the IRM, so that the Information System doesn't ground to a halt within a few days of the unfortunate mishap. The IRM and anyone who is designated to be an ASM

should be sent either to Programmer's Intro or System Operator school at HP, and then to the System Manager course. If your applications software vendor offers a System Manager course, by all means send them to that course also. Then the IRM should conduct an in-house course to supplement the generic courses by teaching policies and procedures peculiar to the company. The System Administrator and the Programmer/Analyst plus the one non-IS type person mentioned above should all be trained as ASM's as a minimum. Three alternates should suffice in most companies. Any more or less would probably make your EDP auditors a little nervous.

Next, the IRM and all of the ASM's should attend training courses given by the vendors on applications software and 4GL products. These courses typically go into great detail on the structure of each package and how to use them for maximum gain. Armed with this knowledge, the IRM and his charges should return home and set-up training courses for all of the users. The IRM should plan on teaching all of these courses himself, perhaps with some help from the ASM's. A training room with a few terminals is essential to let the users get hands-on training. The user classes should not be intensive like the vendor's, but should be tailored down so as to give the users in your particular company only the knowledge needed to get their particular jobs done. All users should attend as a minimum classes on the applications software, and then classes on the 4GL products, if they show an interest in programming.

Now the IRM must put together a "staff" of Departmental Computer Coordinators (DCC's), all of whom should have gone through the appropriate set of user training classes. These DCC's will form the cornerstones of your small-shop IS concept. They will be the first line of contact for your users, answering questions, helping with terminal problems, and helping to write 4GL programs. You should train at least two DCC's in each using department, one of which should be the department manager. The department manager may never logon as a bonafide user, but at least you will know that he has been exposed to what the computer can and cannot do. The manager will also be better equipped to survive meetings at which computerese is being bandied about the table.

The second DCC for any given department should be a person who has some experience with computers, is comfortable around them, and wouldn't mind helping out his fellow

workers with computer applications. This person should be designated the primary DCC for the department, and the department manager should be the alternate DCC. The informal organization of DCC's and ASM's is shown in Figure 4. The user first contacts the primary DCC with the problem. If he can't solve it, then the DCC contacts the IRM. The IRM has his vendor technical personnel to fall back on if he can't answer the question. Note that there is no one with the job title Alternate System Manager or Departmental Computer Coordinator. These are considered to be additional duties on top of whatever primary duties the individuals have.

```
┌──────────────────┐   ┌──────────────────┐   ┌──────────────────┐
│     Software     │   │     Software     │   │     Hardware     │
│    Vendor #1     │   │    Vendor #2     │   │    Vendor #3     │
│   Tech Support   │   │   Tech Support   │   │   Tech Support   │
└──────────────────┘   └──────────────────┘   └──────────────────┘

    ┌──────────────┐   ┌──────────────┐   ┌──────────────┐
    │  Alternate   │   │ Information  │   │  Alternate   │
    │   System     │...│  Resource    │...│   System     │
    │  Manager #1  │   │  Manager     │   │  Manager #2  │
    └──────────────┘   └──────────────┘   └──────────────┘

 ┌──────────┐   ┌──────────┐        ┌──────────┐   ┌──────────┐
 │   DCC    │   │   DCC    │        │   DCC    │   │   DCC    │
 │ Dept. 01 │   │ Dept. 02 │        │ Dept. 03 │   │ Dept. 04 │
 └──────────┘   └──────────┘        └──────────┘   └──────────┘

   Users          Users              Users          Users
```
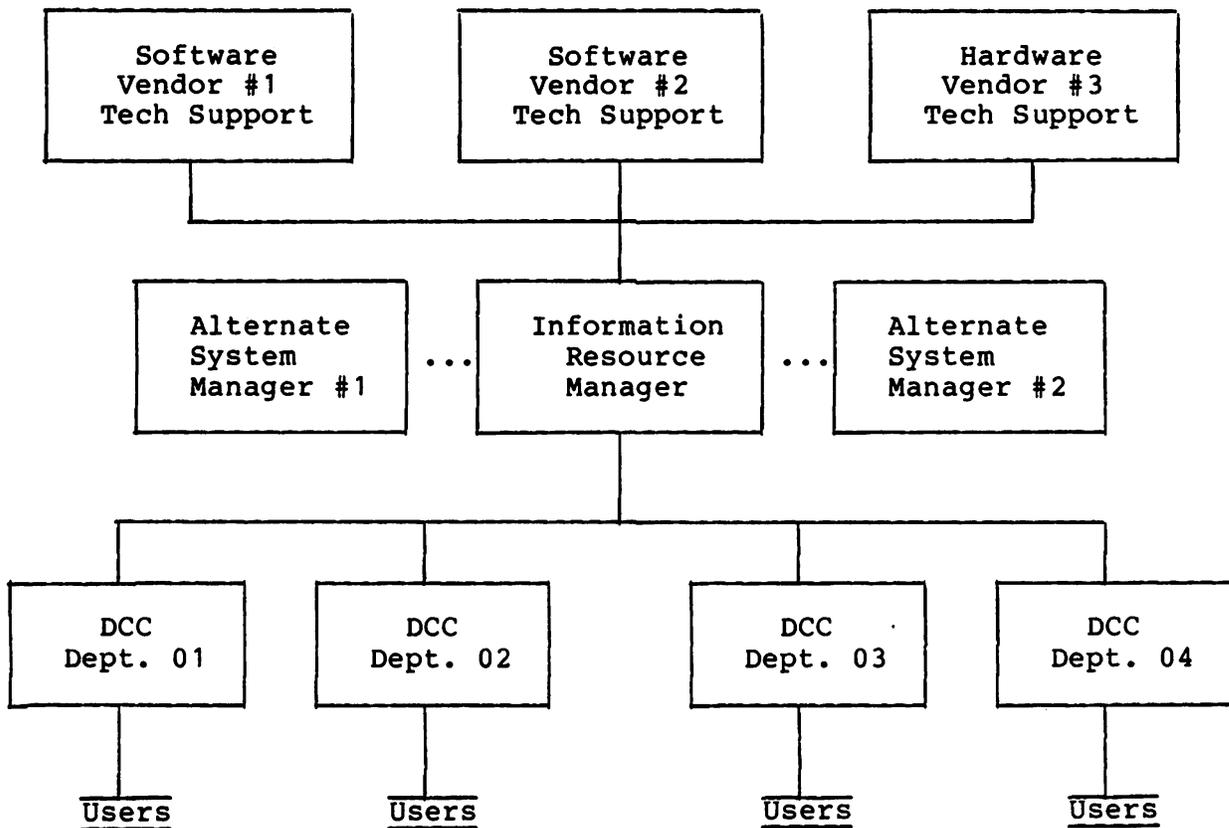
Figure 4
Lines of Communication for Problem Resolution

Finally we get to the concept of the user/programmer, which is a new word I've concocted in the style of the well-worn title programmer/analyst. A user/programmer is simply a company employee whom you've trained to use the applications software and to program with a fourth generation language. The user does not simply use; he is an active part of the total information system. The idea is to encourage user independence. Don't let your users think that the only way to get information out of the computer is to run a report written by your vendor, or a 4GL program written by another user. With just 3 or 4 hours of training on a 4GL product, most users with a basic aptitude for programming can be writing their own simple adhoc report programs. Granted, not all of your users will have that basic aptitude or even want to

program. It is encumbent on the IRM to at least give all users an opportunity to learn the language and try their hand at it. You'll be surprised how many users will pick up the ball and run with it given half a chance. A word of caution is in order here. Don't let your general users have the ability to create programs that write to the databases. Designate a few key DCC's to have this ability, and let your general users use their programs to input data. Of course the vast majority of data will be input using your vendor's programs, so this caveat applies only to your homegrown databases.

You will discover after a short time that a select few of your DCC's will take a strong liking to the 4GL products. They will undubtedly be the first to approach you and ask whether the HP3000 would handle this and

that application. If the DCC shows real aptitude with the computer, then take him under your wing and show him as many tricks of the development trade as you can, within security limitations. Help him to design the database for his application, and actually let him create the database. Some of the 4GL packages, such as Dictionary-Plus from Quasar, contain utilities that allow users to build a schema in editor, then create the database and the data dictionary directly from the schema with just a couple of commands. Spend time to continue to develop these high- performance DCC's. You'll find them to be an invaluable part of the total plan as your Information System matures.

Installing the user/programmer concept at your company will render the following benefits, as a minimum:

- With users writing their own ad-hoc reports, the workload on your Programmer/Analyst is reduced, and thus his backlog will be shorter.

- Users will get a psychological boost because they feel they are an active part of the system, rather than relying on other people's programs.

- Users get exactly the report they need, since they don't have to translate their needs to someone else.

- Users get the reports they need quickly, rather than waiting the days or weeks it will take for the project to make it through the Programmer/Analyst's backlog.

- You will have depth of programming expertise throughout the company. If only one or two people are doing all the company's programming in a central IS shop, you are vulnerable in the event of their absence, temporary or otherwise.

It's obvious that if you have a hundred or so users running around gleefully writing their own programs, then you will soon have a plethora of different reports giving different people different information on the same subject. It is imperative that the IRM design some way of controlling what I call "corporate programs". These programs should produce company standard reports, and they should only be altered by IR personnel. There are many ways to design such a control system, which I will not try to outline here, but one way would be to create a separate group called COMMON that is a repository for corporate programs. The Programmer/Analyst in the IR department could be designated as group

librarian, and only he could have write capability on all files in the COMMON group.

One final note on the subject of people. There is no need for a night computer operator with the small-shop concept. You can use a job scheduler like SLS or SLEEPER (which are programs in the HP3000IUG's Contributed Software Library) to run all your batch jobs overnight. The System Administrator can untangle all the output first thing the next morning and have it ready for user pickup. The only other thing a night operator might do if you had one is system backup. You can accomplish this task by doing an incremental partial SYSDUMP during the lunch hour Monday through Friday. You can pay one of your people overtime to come in during the weekend and do a weekly full SYSDUMP. Some companies do partials early in the morning, and the full backup on Friday afternoons, since most users in many companies cut out early on Fridays anyway.

RULES OF THE ROAD

The small-shop Information System based on the user/programmer concept is more than an idea; it's a reality that is working today in many companies with progressive Information System policies. Getting the system running is only half the job. You must lay out very clear Rules of the Road to guide all those concerned in order for your system to survive and prosper. The devotees of the traditional "big shop" will be swarming vultures waiting for you to make a fatal mistake if you don't follow these simple rules:

1) Do not, under any circumstances, modify any of the vendors' source code. Do not create any special programs within the domain of the applications packages.

2) Write all of your homegrown applications using 4GL products. Do not even think about using FORTRAN, COBOL, or any other third generation language.

3) Develop a solid working relationship with your vendors. They are in effect an extension of your IR department, allowing you to minimize your in-house staff.

4) Give update training to all your users, especially the ASM's and DCC's, on a regular basis, perhaps quarterly or semi-annually. Publish a user newsletter frequently between update training sessions. A well-trained, informed user is your

most precious asset. Protect your
investment.

Eric W. Roberts

Biographical Sketch

Eric W. Roberts has been Information Resource Manager at The Audichron Company in Atlanta, Georgia, for the past 1 1/2 years. Audichron manufactures systems which give the familiar time, weather, and temperature announcements over the phone. He received his Bachelor of Industrial Engineering degree from the Georgia Institute of Technology in 1972. Subsequently he served in Germany for three years as a Communications Officer in the U.S. Army. He joined Audichron in 1978 as a Field Engineer, then served as Quality Control Manager before assuming his current position in 1982. Eric is currently attending night school at Georgia State University in Atlanta in pursuit of an MBA degree, with a concentration in Information Systems. He is Chairman of the SIGASK Special Interest Group of the HP3000 IUG, a group of users with interest in the MANMAN family of applications software produced by ASK Computer Systems, Inc., Los Altos, California.

- - - - -