52

OPTIMIZING THE COMPUTATIONAL ENVIRONMENT FOR ECOLOGICAL RESEARCH*

S.G. Stafford, M.W. Klopsch,
K.L. Waddell, R.L. Slagle, and
P.B. Alaback

ABSTRACT

Computational tasks for ecological research--database management, statistical and graphical analysis, word processing, communications, and accounting--are diverse. Hardware and software options should be considered as a unit to create the most useful computational environment. Hardware includes mainframes, minicomputers (minis), microcomputers (micros), and supermicrocomputers (supermicros), and such attendant peripherals as printers, plotters, and tape drives. Ease of use, low cost, fast response time, and potential graphics capability make micros a popular choice for interactive computing (statistical analysis and word processing). Software, which must be reliable and easy to use, includes statistical analysis, graphics, data management, word processing, modeling, accounting, and communications. Graphics software can be a powerful medium for presenting research results. Restricting equipment use to specific tasks and appropriately scheduling machines, providing manuals and short courses, and carrying maintenance contracts are important to maximizing output. Resources should be shared through networking to make the best use of existing equipment. Hardware acquired should be adequate for the job intended, and choice of vendor and support level

*FRL 1951, Forest Research Laboratory, Oregon State University, Corvallis. The mention of trade names or products does not constitute endorsement or recommendation for use.

must be carefully evaluated. Developing in-house software should be avoided if commercial packages are already available; a table of statistical software for micros, minis, and mainframes is provided. Although computing systems have hidden costs, the potential gains remain high for all but the most modest research projects.

INTRODUCTION

Optimizing the computational environment means increasing the productivity and efficiency of researchers with existing hardware, acquiring compatible new hardware and software, and coordinating all hardware and software into a well-integrated, cost-effective system. Computational tasks in ecological research include database management, simulation modeling, statistical and graphical analysis, word processing, data entry, communications, automated data collection, and accounting. Each of these tasks places different demands on the available facilities, and some systems do the job more efficiently than others, saving time and money.

Database management generally requires considerable data storage capacity, computer memory, and speed. Large databases on mainframe computers and minicomputers commonly need substantial storage. Even small databases on microcomputers can quickly exceed the available floppy disk storage, requiring the addition of a hard disk. Data analyses, including modeling, matrix operations, graphics, and other statistical techniques, generally make heavy demands on computer memory, speed, and, to a lesser degree, storage. In production statistical analyses (in which a large number of predetermined analyses must be run), computational speed is crucial. Response time is critical for editing and word processing, but less so for analyses. For interactive statistics, simplicity of operation and integrated graphics are most important. Real-time tasks like data entry, laboratory instrumentation, digitizing, and communications are often more efficient on microcomputers than on large multi-user systems.

Staff at the Forest Science Data Bank, which serves the Department of Forest Science at Oregon State University, manage large quantities of data collected from ecological studies conducted throughout the Northwest, including data from the H.J. Andrews Experimental Forest, a Long-Term Ecological Research (LTER) site. The resulting data files and associated documentation are organized, stored, and manipulated within the Data Bank's database management system. In this paper, we describe hardware and software options for optimizing the computational environment on the basis of our experiences.

HARDWARE OPTIONS

Mainframe Computers

Data management and analysis have relied primarily on mainframe computers, which are fast, have large memory and random-access storage, and are very expensive. The peripheral devices (peripherals) for mainframes (tape drives, line printers, large plotters, high-resolution graphics terminals, and laser printers) add substantially to mainframe utility, but also to their cost. Maintenance is also expensive, and system support requires a large staff. The high cost has been justifiable to universities and other large research organizations because mainframes offer a uniform computational environment for a large number of users. To be cost efficient, however, these machines must carry heavy workloads, resulting in long waits for interactive applications such as word processing or data entry.

Minicomputers

Minicomputers recently have become important to ecological research. They have many of the characteristics of mainframes at a substantially lower price. Minis are sufficiently fast for most production statistics and modeling, but cannot serve as many users as a mainframe. Memory and storage generally are not limiting, and their cost is within reach of some large projects and departments. The end user has significantly more control over the software and peripherals than with a mainframe. Furthermore, most applications available on mainframes are also available on the major minicomputers. Minis might become a more cost-efficient alternative to mainframes as the need for computational capacity increases. For example, preliminary estimates suggest that the Forest Science Department could save at least half its annual computer budget if it bought and maintained a minicomputer rather than continuing to pay for the University mainframe.

Microcomputers

A revolution in computing has been brought about by microcomputers, which have proliferated in homes and offices, putting many people in direct contact with computers for the first time. Micros are fast enough for most single-user applications, performing best on tasks requiring fast response time and only moderate computational capability (e.g., data entry, word processing, communications, and automated data acquisition). Their moderate-resolution graphics are useful for interactive statistics and many other applications.

Microcomputers today range from 8-bit microprocessors with 64K (K is about 1000 bytes, or characters) of memory to 16- and 32-bit microprocessors with up to 16,000K of memory. Floppy disks with large storage capacities (e.g., 1.2Mb; Mb is a megabyte, or 1000K bytes) have made more applications possible. A hard disk of 10Mb or greater is a powerful peripheral when users are working with many databases or multitasking software. Networks significantly increase the utility of microcomputers by linking them to shared databases, faster computers, and expensive, shared peripherals. Micros are relatively easy to interface with a wide variety of devices, making them ideal for laboratory instrumentation and other real-time data collection devices.

Supermicrocomputers

Recently, a new class of computers based on the latest 16- and 32-bit microprocessors has been developed. These supermicrocomputers bridge the gap between micros and minis, representing our closest glimpse of what the microcomputers of the next 5 to 10 years will be. They now cost between $10,000 and $50,000 and should become less expensive within the next few years.

Supermicros provide minicomputer architecture and capabilities to a small group of users. They are fast enough for several users or small-scale production statistical analyses. Most machines have between 0.5 and 4Mb of memory, but have operating systems that allow virtual addressing of up to several megabytes of disk memory. Disk storage ranges from 40 to 320Mb, which is less than that of many minicomputers, but adequate for most applications. Many supermicros use a version of Bell Laboratories UNIX multi-user operating system; however, so far, only a moderate amount of UNIX software has become available for the supermicros (Darwin, 1984). With the introduction of AT&T computers into the supermicro market, software may not be long in coming (Hunter, 1984).

Peripheral Devices

Over the past 5 to 10 years, the performance of computer peripherals has improved significantly, and their cost has decreased. The hard-disk drive has become physically smaller, yet can store orders of magnitude more bytes of information. Laser disks, providing in excess of 1000Mb (10^9 bytes) of random-access storage, are currently being tested (Hecht, 1984). Graphics displays have attained higher resolution with more colors and faster response time. Memory chips have become cheaper, permitting larger computer memories and making it practical to allocate memory to simulate ultrafast disk storage. Printers have become faster, producing higher quality text and graphics output. Quiet laser printers, which provide near letter-quality printing at speeds approaching those of the fastest dot matrix printers, are now available (Bernard, 1984). Color graphics printers have become inexpensive enough to permit the quick reproduction of medium-resolution color graphics. Communications are being revolutionized by local area networks (LANs), which allow data storage and peripherals to be shared among computers. Intersite communications are becoming faster and more reliable with microwave satellite links and better phone modems; modems for communication at 2400 to 9600 baud are becoming more common, and AT&T is experimenting with 56K baud modems.

SOFTWARE OPTIONS

General Statistical Software

Statistical computing has come a long way when we consider that computer technology is only a few decades old (Boardman, 1982). The reduced costs of devices for processing and memory (*e.g.*, microchips) have added the personal computer to the computer environment and made it a practical research option (Chambers, 1980; see also Stafford *et al.*, "Data Management Procedures in Ecological Research," this volume). The development of computer systems and programs is central to statistical analysis because it expands the quantity of data that can be analyzed quickly and accurately (Chambers, 1980).

Statistical software has been classifed into three basic categories (Chambers, 1980). *Single programs* are task-specific programs reading data in specific formats, performing calculations, and outputting results. The disadvantages are that single programs are not easily modified to perform larger, more general analyses, are often inadequately documented, and may unknowingly be duplicated at different institutions to solve similar problems.

Algorithm collections (statistical libraries) are sets of subprograms designed to supplement a main program, which the user must write. A good algorithm should be easy to run on many different computers, be reliable, and adapt well to a variety of problems. It should not be limited by the size of problems handled or make restrictive assumptions. Some large collections of statistical algorithms have been developed (e.g., the International and Mathematical Statistical Library [IMSL] [IMSL, Inc., 1980] and Biomedical Computer Programs [BMDP] [Dixon, 1983]); using algorithms simplifies programming and usually improves the quality of results. However, the attributes that make an algorithm machine efficient and statistically rigorous are seldom going to make it easy for the casual user to understand or apply. Often, an interface such as user-friendly documentation is needed between user and algorithm.

Statistical systems are collections of programs that combine the advantages of both single programs and algorithm collections. Normally, they are thoroughly tested major packages (e.g., SAS [SAS Institute, Inc., 1982], SPSS [Nie et al., 1975], and MINITAB [Elkins, 1971]), which include a strict organization of commands through which the user specifies the analyses to be performed; the underlying design incorporates a collection of well-chosen algorithms. Statistical systems must be flexible to be able to handle a great diversity of data.

However, the power of statistical systems often has more to do with their data management and graphical flexibility than with their statistical routines. A common dilemma is that the most powerful statistical systems, which allow for flexible, complex data mergers and support presentation-quality graphics, are the most machine-dependent ("nonportable"). For example, until very recently, SAS (SAS Institute, Inc., 1982), one of the best engineered statistical packages, ran only on IBM mainframe computers, though it is now available on several minicomputers and will soon be available for 16-bit micros. In contrast, SPSS (Nie et al., 1975) is available for most mainframes, minis, and micros, but its data management capabilities are much more limited. Clearly, well-documented statistical systems integrated with spreadsheets, graphics, and word processing will increase user efficiency and capability. For most researchers, attaining a powerful yet easy to use statistical system can be the most critical element in the computational environment.

Microcomputer Statistical Software

Many attempts have been made to describe criteria that might be useful in evaluating statistical software on microcomputers (Carpenter et al., 1984). Key features of certain major software packages, current as of November 1984, are listed in Table 1. Software packages need to provide an integrated system of routines, allowing the user complete flexibility in performing major operations. Input routines must allow direct data entry as well as entry from external databases. Manipulation routines should include transformations, sorting, merging, recoding, and editing. Data management routines need to provide several methods of storing and cataloging data sets. It should be possible to select portions of a data set on the basis of prespecified criteria and pass those portions to one or more of the data exploratory or statistical analysis routines.

Microcomputers have three important features for conducting statistical analysis: ease of use, low cost, and potential graphics capability. The industry has been moving toward increased user friendliness, higher computational speeds, improved color graphics, faster and more versatile peripherals, and more modern programming languages, all on highly capable, inexpensive machines (Chambers, 1980). The impact of increased computing options on micros has already affected software development for data analysis and graphics. Analyses that were too time consuming and costly are now manageable on micros. However, obtaining reliable statistical software for microcomputers can be a problem because the statistical accuracy and long-term reliability of commercial packages recently entering the market have yet to be proven (Lachenbruch, 1980; Carpenter, et al., 1984).

Graphics Software

The way in which analytical results are presented can be as important as the validity of the results themselves. In fact, few statistical tools are as powerful as a well-chosen graph (Chambers et al., 1983). Graphs have been used primarily to show the overall data structure or pattern, though exploratory graphical procedures are most often used

Table 1. Statistical software for microcomputers, minicomputers, and mainframes (adapted from Carpenter et al., 1984), current as of November 1984.

Software package	Statistical capabilities[a]	Operating system	Minimum requirements RAM[b]	DD[b]	Software/(Other)
Microcomputers					
ABSTAT 3.04	D/R/A/G-low	CP/M-80	56	2	
		CP/M-86	128	2	
		MS,PC-DOS	128	2	
ABSTAT 4.00	D/R/A/G-low	MS,PC-DOS	196	2	
AIDA	D/R/A/MV/G-high	Apple-DOS	48	1	
ASTAT	D/R/A/MV/G-low	CP/M-80	48	1	
		Apple-DOS	48	1	
		MS,PC-DOS	48	1	
ASYST	D/R/A/C/G-high	MS,PC-DOS	320	2	
BMDPC	D/R/A/MV/G-low	PC-DOS	640	1	(5Mb hard disk)
DYNACOMP	R/A/G-low	Apple-DOS	48	1	
HSD-STATS PLUS	D/R/A/G-high	Apple-DOS	48	1	
HSD-ANOVA II	A/G-high	Apple-DOS	48	1	
HSD-REGRESS II	R/G-high	Apple-DOS	48	1	
HSD-PC STATISTICIAN	D/R/A/G-high	MS,PC-DOS	128	2	
INTROSTAT 2.2	R/A/G-high	Apple-DOS	48	1	
		MS,PC-DOS	64	1	
		Atari-DOS	48	1	
MICROSTAT 4.1	D/R/A/MV/G-low	CP/M-80	64	1	APC-BASIC
		CP/M-86	64	1	
		MS,PC-DOS	128	1	APC-BASIC
MICRO-TSP	D/R/G-high	Apple-DOS	48	1	
		CP/M-80	64	1	
		MS,PC-DOS	128	1	
MICRO-TSP 4.0	D/R/A/G-high	MS,PC-DOS	256	1	
MINITAB	D/R/A/G-low	PC-DOS	320	1	(10Mb hard disk)
MSTAT	D/R/A/G-low	CP/M-80	64	1	
		CP/M-86	64	1	
		MS,PC-DOS	128	2	
NUMBER CRUNCHER	D/R/A/MV/G-low	CP/M-80	64	2	M-BASIC
		MS,PC-DOS	196	2	

Table 1. Continued.

Software package	Statistical capabilities[a]	Operating system	Minimum requirements RAM[b]	DD[b]	Software/(Other)
Microcomputers (continued)					
NWA-Statpak	D/R/A/G-low	CP/M-80	64	2	M-BASIC
		CP/M-86	64	2	
		MS,PC-DOS	128	2	
RS/1	D/R/A/C/G-high	P-System[c]	128	2	(DEC PRO 300 series)
SAM	D/R/A/MV/G-low	Apple-DOS	48	2	
		CP/M-80	56	2	M-BASIC
		CP/M-86		2	
		MS,PC-DOS		2	
Speed STAT	D/R	Apple-DOS	48	2	
SPS	R/A/MV/G-high	Apple-DOS	48	1	
		CP/M-80	64	1	M-BASIC
		CP/M-86	64	1	
		MS,PC-DOS	64	1	
SPSS/PC	D/R/A/MV/G-high	PC-DOS	320	1	(10Mb hard disk)
STAN	D/R/G-high	P-System[c]	64	2	Apple-PASCAL
		P-System[c]	64	2	IBM-PASCAL
StatPac (Wallonick)	D/R/A	CP/M-80	64	2	
		MS,PC-DOS	128	2	
STATPRO	D/R/A/MV/G-high	P-System[c]	64	2	(Apple)
		P-System[c]	128	2	(IBM,DEC,Sage)
SYSTAT	D/R/A/MV/G-low	CP/M-80	56	1	
		MS,PC-DOS	256	2	
		UNIX	500	2	
TWG-ELF	D/R/A/MV/G-high	Apple-DOS	48	1	
		CP/M-80	64	2	
		CP/M-86	128	2	
		MS,PC-DOS	128	2	
Minicomputers					
BMDP	D/R/A/MV/G-low	UNIX, VMS			
MINITAB	D/R/A/G-low	UNIX, VMS			
PSTAT	D/R/A/MV/G-low	Most systems			
S	D/R/A/MV/G-high	UNIX			
SAS	D/R/A/MV/G-high	Most systems except UNIX			

Table 1. Continued.

Software package	Statistical capabilities[a]	Operating system	Minimum requirements RAM[b]	DD[b]	Software/(Other)
Minicomputers (continued)					
SPSS-X	D/R/A/MV/G-high	CMS,MVGS,VMS			
Mainframes					
BMDP	D/R/A/C/MV/G-low	Most systems			
S	D/R/A/MV/G-high	UNIX			
SAS	D/R/A/C/MV/G-high	IBM only			
SPSS	D/R/A/MV/G-high	Most systems			
SPSS-X	D/R/A/MV/G-high	Most systems			

[a] D = Descriptive statistics; R = Regression; A = ANOVA; C = Nonlinear curve fitting; MV = Multivariate statistics; G-low = Graphics, low resolution; G-high = Graphics, high resolution

[b] RAM = random access memory, in kilobytes (K); DD = minimum number of disk drives.

[c] Provided with software by manufacturer.

to show departures from an expected structure. Graphs are powerful diagnostic tools for confirming assumptions or, when assumptions are not met, for suggesting corrective action (Chambers, 1980). There has been significant development in graphics software for micros; most microcomputer statistics software also makes extensive use of graphics.

How graphical results are presented is important. Poorly arranged tables and figures can hinder the understanding of research results and lead to incorrect conclusions. Caution must be exercised, for instance, in using color on statistical graphs. Experimental psychologists have demonstrated that varying the intensity of colors can cause optical illusions in the apparent size of objects, thus affecting people's judgment of quantitative information (e.g., Cleveland & McGill, 1983). "Chernoff faces"--depicting clusters or natural groupings within a multivariate data set--also can bias interpretation of numerical data. Variables are assigned different facial features; if a relatively insignificant feature is assigned to a highly significant variable (or vice versa), the researcher's conclusion can be directly influenced. In summary, the researcher should be aware that eye-catching presentation-quality graphics, though attractive, must be used cautiously to ensure that data are presented objectively.

USING HARDWARE AND SOFTWARE TO THEIR BEST ADVANTAGE

Steps for Optimizing the Computational Environment

Despite the allure of new technological developments and their potential applications to research, the top priority for data managers, who often have limited budgets and tight deadlines, must be to optimize the efficiency of existing equipment. The first task should be to inventory all equipment and look for bottlenecks in data processing. Research plans and usage trends should be examined so that future demands on computational resources can be anticipated. We have found it effective to take the following steps:

- Restrict the use of equipment to specific tasks to prevent a computer with capabilities in high demand from being tied up on a task which could be executed on a less critical machine. A busy microcomputer should not be used for time-consuming "number crunching" if more powerful machines are available. Likewise, using a mainframe for data entry is not cost effective if micros are available.
- Schedule equipment to spread usage more uniformly over a larger portion of the day, reducing the "feast and famine" cycle typical of unscheduled equipment. Scheduling allows users to plan their time more efficiently and, in the case of mainframes, to take advantage of the lower traffic and reduced time-sharing rates available during certain shifts. Scheduling should be flexible enough to accommodate machine down time (time lost to repairs) and open time (time on unscheduled machines) for short-term "rush" jobs.
- Provide in-house handouts summarizing operating commands for each machine or analysis package; an easily accessible and well-stocked library of commercial user manuals; and, when possible, regularly scheduled short courses designed for specific users. Acquiring software that is easier to learn and more interactive

has also helped increase the productivity and efficiency of data bank personnel. Increased automation of routine data retrievals, analyses, and archivals using batch command languages on both mainframes and micros has contributed to this process.

- Maintain the balance between batch and interactive service modes; that is, use whichever mode is best suited for the given task. Batch mode conserves system resources (*e.g.*, allowing a user to submit a job which runs during less congested times), whereas interactive mode facilitates user involvement. This balance provides flexibility, which translates into increased productivity.
- Share resources to make the best use of all existing equipment. Computers not continuously used by one research team might be shared by others. A color graphics terminal and color printer/plotter are prime examples of expensive equipment that can be purchased cooperatively and shared among projects at a site.

  The ultimate goal is a network system linking all computers to shared software and peripherals (see Klopsch & Stafford, this volume). For instance, at Oregon State University, installation of a broadband LAN with several thousand ports linking microcomputers, terminals, and larger computers is under way. The LAN will use general-purpose communications software to allow any user to address any other willing user. The campus-wide cable network can carry more than 100 channels, of which the LAN will require only one; the remaining channels will be available for special dedicated uses (*e.g.*, TV, sub-LANs). When fully operational, the LAN will telecommunicate data, documents, and electronic mail between cooperating departments, including the University's Department of Printing.
- Carry maintenance agreements and regularly schedule preventative maintenance to avoid one of the biggest potential costs to researchers: loss of productivity due to computer down time. Machines must be protected from electrical power surges, dust, smoke, temperature extremes, and humidity to minimize damage and time lost to repairs. Software for local diagnostic testing of computer problems and for recovering partially destroyed disks also can minimize these costs. In our experience, most microcomputer down time has been attributable to defective equipment;

defects have usually surfaced within 6 months of acquisition. To guard against these initial failures, equipment should be purchased from reputable vendors that test their products before shipping; it may be especially advantageous to get equipment that is "burned in" (i.e., tested for at least 100 hours).

- Provide adequate security. Misuse, theft, hardware vandalism, software piracy, and data file security (including problems associated with both public and internal network access) must be considered in system planning. If system security is inadequate, user passwords or numbers might be misused or stolen, and account funds subsequently depleted or data file contents damaged.

Optimizing Acquisitions

Understanding the technical and operational aspects of hardware and software helps the staff make the best selection; however, this knowledge alone is not enough. Current information on availability, compatibility, and plans to update programs and documentation of commercial software packages should be examined thoroughly before a product is purchased.

Hardware and software must be considered together because one without the other is useless. Avoid developing major in-house hardware and software if commercial products are available. Commercial software packages are cheaper and usually better documented. However, when no reasonable alternative is available, the decision to develop an in-house package must be made with full recognition of all the costs involved, initial and ongoing.

Some caveats for acquisitions apply:

- Do not delay purchasing hardware or software just because something better is coming; something better is always coming. The best way to avoid obsolescence is to purchase systems that can be expanded as new products become available and to choose systems which follow industry standards, use standard disk formats, and have widely used operating systems.
- Avoid outdated equipment unless it is the only available option. Little software is developed or maintained for older machines or operating systems.
- Be realistic about delivery times. Delivery times on newly released hardware and software are always optimistic. Vendors want users to know that new or

updated products are coming, but often cannot project when. Newly released products frequently have bugs which take considerable time for users to find and the company to correct. There may be a substantial time lag between new hardware releases and software availability, so wait for software before purchasing hardware. Several instances of firms advertising products that do not actually exist have been documented. Sometimes advance orders fund the basic research and development for the product!

- Establish a list of available products which meet current needs, and consult with other users whose requirements are similar. Considerable time and money can be saved by sharing software as well as solutions to common problems.
- Assess compatibility between hardware and software. Whenever possible, select similar software for all machines at a site to simplify the user environment and optimize equipment usage.

Choosing Hardware

Hardware, whether a mainframe or micro, should be designed to provide a full range of analytical tools for all sizes of tasks. Be sure the equipment is suitable for the job. Many microcomputers with 8-bit processors are restricted by 64K or less of memory, which may be inadequate for working with large arrays. Supermicros often come with virtual memory, allowing each user to access 16Mb of memory. Disk storage must be adequate; purchase all the storage you can afford. Hardware durability must also be considered; it is unfair to expect a product designed for occasional home use to stand up to production pressures.

Buy from a well-established, financially sound company that will continue to upgrade its products. But remember that brand names are not always a good indication of support. Many larger computer and software companies do not like working with end users and rely on dealers. If dealer support is poor, purchasing a brand name product can be worse than purchasing the product from a less well-known company whose dealer can provide the necessary support. Furthermore, distance from vendors is not always a good indication of support level. Local support often consists merely of sales personnel with poor technical training.

Some mail-order companies have excellent service departments, but, in general, local support is preferable. However, support must be evaluated in light of staff capabilities; the support adequate for one group may not be adequate for another. Ask vendors for the names of people who have purchased similar hardware and software and use those people as references.

Our experiences with vendors have varied. In one instance we found that servicing a standard IBM monitor was far more costly and time consuming than repairing any of our terminals bought through mail-order houses. In buying some widely available computers through local distributors, we have had trouble getting technical help; in the case of one popular computer company, we were expected to call a single phone number nationally (which was nearly always busy). Our biggest disappointment, however, was with a state-of-the-art multi-user microcomputer bought from a mail-order house. The unit as shipped was inoperable due to improperly designed memory boards which had been inadequately burned in. After many phone calls, patching the operating system, switching boards, and mailing vital components back and forth, we found the system was too slow to be useful for its original purpose. Our experience taught us that minimizing a purchase price through mail-order houses may be false economy when you consider ease of repair and availability of replacement parts. Although all repairs were done under warranty, by the time the system was finally fully operational (over a year later), prices and technologies had changed to such a degree that for less money we could have purchased a far more useful, capable, and probably reliable system. The moral to the story is this: although something better will always be coming, it is important to act decisively when a need arises.

Choosing Software

Software has been changing considerably. Operating systems, typified by UNIX and VMS (Digital Equipment Corp.'s minicomputer operating system), have become more flexible, permitting virtual memory, multitasking, and input/output redirection. Software for mainframes and minicomputers has become more interactive. Standards are being developed to make graphics displays more portable from machine to machine (McCune, 1984; Wong, 1984). Relational database management systems are superseding the older, more restrictive hierarchical systems and network systems. As microcomputers become more powerful and memory increases, analogs of many software packages previously restricted to minis and mainframes are appearing on micros.

A thorough evaluation of software before it is purchased will reduce disappointments from incompatibilities or unrealistic expectations of the product. LTER researchers, for instance, have the advantage of consulting research groups within and among LTER sites to determine which software best meets their needs. However, discussing statistical software is a problematical process. It is important to remember that software is constantly being improved; that is, it is easy to wind up evaluating an obsolete program (Hamer, 1981). We can safely infer only two things from papers evaluating statistical software: 1) specifics about the programs themselves, and 2) information about the attitudes and skills of those who write the programs. While the latter has longevity, the former is likely to be short-lived.

When first considering the acquisition of any software, define its objective and intended function. For example, if the software is to be used by a large number of people, a well-documented, user-friendly package is probably desirable. The most important, obvious advantages are 1) reduced personnel time training novices, and 2) broadly increased computer literacy within a research group because the software is easy to learn. Disadvantages may be less apparent, but are real and of concern. Software that is touted for its user friendliness may not be machine efficient, cost effective, adequate for specialized tasks, flexible enough to handle general tasks, or rigorous enough for sophisticated analyses.

When we first evaluated database management software for the Andrews LTER site, a bewildering array of options was available for our microcomputers. Logically, we thought that the most powerful package would be best. We eventually purchased two packages, one strictly for replacing our keypunch data entry system, the other for its highly regarded database management capabilities. After intensive use, we found that the package purchased for data entry was also the better one for database management because it was so easy to learn and use. When limitations arose, it was more efficient to develop some simple software to supplement the data entry package, using languages our staff already knew, than to try to learn the system language of the more sophisticated database package.

It is useful to get a demonstration copy of the software (demo disk) before purchasing. Some manufacturers provide these disks, which permit a trial of a "crippled" system before final purchase. (A "crippled" version of a program uses only a small data set or is in some other way modified so that it cannot be applied to real analyses.) Where demo disks are unavailable, try to test the software on another system before purchasing. When copies of commercial programs are being considered for multiple machines, additional licenses, usually available from the manufacturer at a reduced cost, will be needed. Software vendors are already incorporating protection devices into their programs to prevent users from making unauthorized copies. Look for complete and available documentation, independent training manuals, ease of interfacing with other software, good access to consultants, and a high degree of vendor support.

A generous amount of public-domain software already exists for many operating systems. Unfortunately, documentation of public-domain software tends to be poor, and bugs are not uncommon. Current periodicals and other users should be able to point out both good and poor software packages.

IMPLICATIONS

Chambers (1980) suggested that computing systems could be arranged in three configurations to make less expensive computing power available to more statistical users. The first would be a <u>minicomputer center</u> ($100,000 in 1984 dollars) supporting about 10 users, with the same processing capacity, program size, and data storage as formerly provided by the large mainframe computer systems. This facility would serve a more specialized user community and therefore require a less diverse selection of hardware and software. The second would be a <u>personal computer</u> or microcomputer on every scientist's desk. And the third would be a compromise between the first and second, a <u>distributed system</u> in which each user has a processor, but mass storage is centralized. This is a forerunner of the previously noted LAN system, which combines advantages of inexpensive local processing and utilizes high-speed transfers from system to system in the network.

Half a decade later, the future is not that different from what Chambers envisioned; most sites currently use one or more of the configurations described. Regardless of the arrangement, each computational environment must be configured so that users can gain the most advantage for the least cost.

Although people buy computers to save time and money, there are many hidden costs, both initial and ongoing, which need to be carefully evaluated. Software and hardware require time for installation and customization; the hardware itself may require additional furniture or wiring. Program bugs must be found and corrected, and users need learning time on new machines and with new programs. The annual costs of maintenance and supplies (e.g., maintenance contracts as well as personnel time for system management and preventative maintenance) are roughly 5 to 25% of the initial purchase price, depending on the level of support required. Nevertheless, the potential gains remain high for all but the most modest research projects.

The proportion of resources allocated to computing will doubtless increase with time as will the possibilities for more sophisticated research and increased productivity. Computer applications that were prohibitive with timesharing on a mainframe are now highly affordable on micros or minis. Attempts in the past to conduct large-scale, integrated research suffered from the lack of computer flexibility and capability that we now routinely have at our disposal. We feel that the investment in computers has more than justified itself in terms of both education and analytical output. Clearly, the ability of the LTER program to realistically meet its goals of acquiring and managing data depends on the ability of data bank personnel to be innovative in applying the latest computer technologies to optimize the computational environment for ecological research.

LITERATURE CITED

Bernard, J. 1984. HP's new laser jet printer. Computers & Electronics 22(7): 36-39, 87.

Boardman, T.J. 1982. The future of statistical computing on desktop computers. The Amer. Statistician 36(1): 49-58.

Carpenter, J., D. Deloria, and D. Morganstein. 1984. Statistical software for microcomputers. Byte 9(4): 234-264.

Chambers, J.M. 1980. Statistical computing: History and trends. The Amer. Statistician 34(4): 238-243.

Chambers, J.M., W.S. Cleveland, B. Kleiner, and P.A. Tukey.

1983. Graphical Methods for Data Analysis. Wadsworth International Group, Duxbury Press, Boston, MA. 395 pp.

Cleveland, W.S. and R. McGill. 1983. A color-caused optical illusion on a statistical graph. The Amer. Statistician 37(2): 101-105.

Darwin, I. 1984. UNIX software directory. Microsystems 5(4): 45-58.

Dixon, W.J. (ed.). 1983. BMDP Statistical Software. University of California Press, Berkeley, CA. 733 pp.

Elkins, H. 1971. MINITAB Edit, MINITAB Frequencies and MINITAB Tables: A Set of Three Interrelated Statistical Programs for Small Computers. University of Chicago, IL. 100 pp.

Hamer, R.M. 1981. Papers that evaluate computer programs. The Amer. Statistician 35(1): 264.

Hecht, J. 1984. Store it with light. Computers & Electronics 22(7): 60-64, 94-96.

Hunter, B. 1984. AT&T enters the micro/mini market. Microsystems 5(6): 114-118.

IMSL, Inc. 1980. International Mathematical and Statistical Library Reference Manual. Eighth edition. Houston, TX.

Klopsch, M.W. and S.G. Stafford. 1985. The status and promise of intersite computer communication. pp. 115-123. In: Research Data Management in the Ecological Sciences. W. Michener (ed.). Belle W. Baruch Library in Marine Science, Number 16. University of South Carolina Press, Columbia.

Lachenbruch, P.A. 1983. Statistical programs for microcomputers. Byte 8(11): 560-570.

McCune, D. 1984. An introduction in NAPLPS. Microsystems 5(7): 54-65.

Nie, N.H., C.H. Hull, J.G. Jenkins, K. Steinbrenner, and D.H. Brent. 1975. SPSS: Statistical Package for the Social Sciences. Second edition. McGraw-Hill, NY. 675 pp.

SAS Institute, Inc. 1982. SAS Users Guide: Basics. 1982 edition. SAS Institute, Inc., Cary, NC. 923 pp.

Stafford, S.G., P.B. Alaback, K.L. Waddell, and R.L. Slagle. 1985. Data management procedures in ecological research. pp. 93-113. In: Research Data Management in the Ecological Sciences. W. Michener (ed.). Belle W. Baruch Library in Marine Science, Number 16. University of South Carolina Press, Columbia.

Wong, W.G. 1984. Graphics portability with Digital Research's GSX. Microsystems 5(7): 74-82.