



Lawrence Berkeley Laboratory

UNIVERSITY OF CALIFORNIA

Engineering & Technical Services Division

RECEIVED
LAWRENCE
BERKELEY LABORATORY

SEP 26 1983

LIBRARY AND
DOCUMENTS SECTION

To be submitted for publication

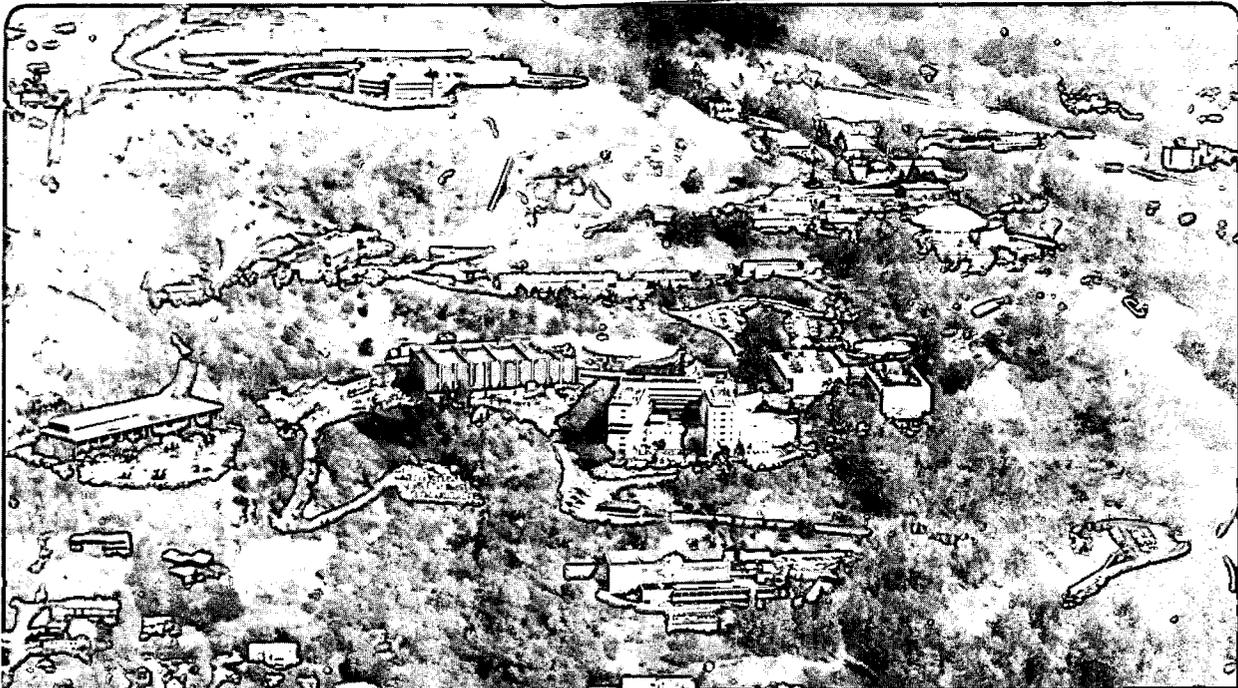
SUPPORTING THE LITERATE USER

D.F. Stevens

August 1983

TWO-WEEK LOAN COPY

*This is a Library Circulating Copy
which may be borrowed for two weeks.
For a personal retention copy, call
Tech. Info. Division, Ext. 6782.*



LBL-16512
c.2

DISCLAIMER

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor the Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or the Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or the Regents of the University of California.

SUPPORTING THE LITERATE USER*

David F. Stevens

Lawrence Berkeley Laboratory
University of California
Berkeley, California 94720

August 1983

* This work was supported by the U.S. Department of Energy under contract No. DE-AC03-76SF00098

SUPPORTING THE LITERATE USER¹

D. F. Stevens

University of California Lawrence Berkeley Laboratory

Introduction

Forbes and *Management Review* carry articles on office automation and data processing productivity. The computer was *Time's* "man" of the year for 1982. Computer courses are offered in elementary schools. Computer competence is required for graduation in college-bound tracks in high schools. Universities are announcing arrangements to provide personal computers for "every member of the University community". What do these events portend? How will these activities affect those charged with the effective management of their companies' computing resources? How will widespread computer literacy affect the user support function?

"Computer literacy" is a phrase that occurs in the popular (and technical) press with ever-increasing frequency. It is certainly evocative, but it is vague and ill-defined. The widespread use of this single term obscures the fact that the substance behind it is not a uniform concept: On the contrary, it is an amalgam of several related, but not totally consistent, ideas. It follows that any discussion on the topic is likely to produce confusion unless some care is taken to clarify the particular type(s) of "computer literacy" that one has in mind. As I have seen the phrase used in the last several months there seem to be three main themes emerging: three kinds of computer habituation that contribute to a generalized notion of "computer literacy". Not all of these movements are in the direction of anything that could properly be called "literacy", but they are all towards *familiarity* with computing, computers, or computer-driven tools and devices. We shall here consider these three movements, some of the characteristics and expectations of the people they encompass, and how information services management should respond to them.

Waves of Familiarity

The first thing to note about these movements is their magnitude. While neither of the first two will be as all-encompassing as the third, they all will achieve sufficient size and strength to qualify as waves of influence, fashion, and change. They will also achieve a wide distribution within the

¹ This work was supported by the U.S. Department of Energy under contract No. DE-AC03-76SF00098.

organization. Unlike traditional data processing, which has been very much a specialized activity whose practitioners have tended to be isolated from the corporate mainstream, these new waves of computer consciousness will penetrate all areas of most enterprises, including the field, the shop-floor, the administrative offices, and the boardroom. Secondly, although the three waves are independent, they will overlap in time. They will all develop over periods of several years, and the later ones will crest before their predecessors have fully ebbed. Responsible management will rarely have the good fortune to be able to concentrate on only one of these movements at a time. And thirdly, the temporal succession of the waves is strictly temporal; the order in which they will arrive is an accident of timing, and does not indicate any causal relationship from one to another. It is certain that some people will belong to two, or even all three of them in turn, but there is no natural progression from one to another, and it would be an error to try to force such a progression upon the user community.

All three waves are already showing signs of swelling. The first to flood will be the *office automation* wave; the second, close on its heels, the *video-arcade-and-personal-computer generation*; the third, the *public school generation*. The attitudes, computing styles and interface requirements that will characterize the individual movements are quite different from each other and from those encountered in a present-day conventional data processing operation. Suitable adjustment to these changes will require planning and preparation. There *may* be enough time to do adequate planning for the third wave (for those managers adept or lucky enough to survive the first two), but there is only time to *adapt* to the second wave, and barely time to *react* to the first, for it is upon us.

The first wave: Office Automation

The newly automated office is characterized by hands-on computer use by large numbers of first-time users. These users in general have little understanding of the processes they are controlling, and little desire to achieve such understanding. (In the ideal case, they should have little need for such understanding.) In a sense, the properly automated office will resemble the modern kitchen, in which the user is surrounded by equipment of considerable technological sophistication, all of which works reliably on demand and none of which requires formal training for effective use; indeed, most of it is quite self-evident, or at least self-teaching.

Many of the primary users of office automation systems will be technologically naive, and their naiveté will find expression in two general ways: *technophobia* and *utopiana*. The "technophobia" I have in mind is not a clinical psychosis, but a set of attitudes, which is more accurately described, not as *fear* of technology, but as a combination of *distrust*, *dislike*, and even *hate* of technology.

"Utopiana" is to some extent the opposite of technophobia; it is that state of happy anticipation indulged in by those who believe what the promotional literature has to say about the immense power, convenience, elegance, simplicity, speed, flexibility, intelligence and general all-round goodness of the system. While Utopiana is certainly unrealistic, it is understandable even to those of us who recognize the falsity of its premises, and computer professionals tend to treat its victims with a modicum of sympathy. By contrast, it is almost axiomatic that technophobia is quite foreign to computing professionals; consequently, it might be useful to dwell a bit upon its causes and manifestations.

Technophobia is founded in uncertainty, and has both political and emotional roots. Among its specific causes are the necessity to learn new skills and jargon, the substitution of electronic images for physical documents, uncertainty about the consequences of errors, and the potential for change in office power relationships. The new skills that must be learned are not necessarily very different mechanically from the traditional office skills, but they are packaged in new and different guises, they deal with objects that aren't really there, and they are described in terms of a strongly computer-oriented jargon. One result of introducing new skills into the work environment is that old expertise becomes obsolescent and old experts lose their claim to distinction. Those whose positions or images depend upon superior knowledge find themselves threatened by the possibility that they will play the fool in public before their learning is complete. As the office switches from dependency upon tangible, physical objects to dependency upon virtual, electronic objects, the power that goes with control passes from those in charge of the physical files to those in charge of the electronic files. And despite the claims of some of today's systems, control over the contents of electronic documents is still far more tenuous than control over physical ones. Related to this is the Silas Marner syndrome, in which comfort is derived from the ability to see and touch (fondle, if you will) the actual physical documents that are so important; if you can see it, you know it is there. It is hard for some people to see that the conversion to electronic form is not actually tantamount to shredding, and it is quite upsetting to some to find themselves responsible for the integrity of documents that are at the mercy of unknown and invisible forces that they do not understand, did not want, and don't think they need. They have no feeling for cause and effect in these new systems; they quickly learn that some trivial effects require sequences of actions that are both intricate and cumbersome, and they are certain that trivial actions can have devastating consequences for both equipment and data (and in this, unfortunately, they are generally correct). As a result, where once they were secure in their knowledge of policy and procedure, they now are prey to doubt and uncertainty. All of these influences tend, sometimes subtly, sometimes blatantly, both to generate change in office relationships (and in the political power

of position that derives therefrom) and to generate fear of those changes. And technology gets all the blame.

One aspect of first-wave technophobia is a reluctance to learn very much about the internals of new systems or to experiment with them: the introduction of new technology is not so much an adventure as an ordeal. This engenders a high degree of dependence upon expert assistance. Since this is often provided by the vendor, the usual result is that the first things learned are the flashiest, and that expectations soon soar far beyond reality. Thus, although they start from very different points, both the technophobes and the utopians arrive at a point of unrealistic expectations. Unfortunately, the reality is that these systems are underpowered and oversold, and the tools they provide, while unquestionably of great power and promise, are of correspondingly great immaturity. (One sees, for instance, the ability to combine line drawings and text in a system that has no provision to generate a table of contents; spelling checkers that won't detect an error in this frays; hyphenation by interactive scan, that must be redone if a phrase is inserted or deleted; sophisticated formatting systems that cannot show their results directly to the operator; etc.) The users' high expectations are instilled by the vendors and confirmed by the preliminary demonstrations and initial carefully choreographed applications, and they prove to be surprisingly resistant to the cold light of later experience.

Office automation will tend to have rather high corporate visibility. Many of these systems are being sold as elements of "executive information systems", and terminals are appearing in the offices of corporate management as well as in lower-level administrative offices. It is here, in the corporate offices, that one form of technophobia (fear of playing the fool) is most pronounced, for it is corporate management that has the most face to lose. It is also here that unwarranted expectations generate the most heat, and that the kitchen metaphor is most appropriate: the last thing a corporate executive wants to do is to waste his time learning a tricky new system. Even a well-designed self-teaching module is not good enough if it requires more than one hour to teach effective use of the system, or if the retention period is less than a week. Unless the system is self-evident it has no place in the executive suite.

The second wave: The Video-Arcade-and-Personal-Computer Generation

Two things will combine to make the video-arcade-and-personal-computer generation the most traumatic of the three waves. First, it will arrive in full force before the first wave has dissipated -- before it has even crested, in all probability. And second, it will be in many ways the antithesis of the first, so that the hard-won lessons we have just learned for dealing with the "new breed" of users will

turn out to be inappropriate (at best) and counter-productive (almost always) when applied to this second wave.

In contrast to the technophobia of the first wave, the video-arcade-and-personal-computer generation will be characterized by a kind of technomania. Instead of trepidation and reluctance to adopt new tools, the second wave will exhibit arrogance and eagerness. Where members of the first wave crept ahead with forebodings of disaster, members of the second will charge ahead with expectations of triumph. Where first wavers leaned heavily upon vendors and other properly constituted authority, second wavers will be supremely independent (to the point of anarchy, in many instances). Where the first wave hesitated to try new actions, even when procedures were carefully spelled out, the second will write its own rules (which it will then frequently disregard). Where the first wave saw its new systems as engines of great power and mystery, capable of amazing performance, but always fraught with the possibility of instant disaster, the second sees its new systems as delightful and exciting personal toys.

There should, at this point, be a rather familiar aura about this second wave, for it is typified by the grad-student hacker, who is the spiritual descendant of the deep systems programmers of the 50's and 60's. In the 50's and 60's these people were part of the priesthood; in the 80's they will be members of the congregation.

We have noted that the first wave will consist of computationally inexperienced people being introduced *en masse* to the wonders of automation; by contrast the second will consist of personal computer addicts who are largely self-motivated and self-taught. They will be the sort to whom cleverness appeals more than clarity. They may know a great deal about a very narrow range of computing, but very little about computing in the large. In particular they may be quite lacking in the good citizenship that is necessary for the effective use of shared resources.

The expectations of the second wave are likely to have a different focus from those of the first, being directed more towards performance than functionality, but -- surprisingly enough -- they may be just as unrealistic. This will be largely a result of experience with one-person, narrow-function, special-purpose, low-overhead systems; the high overhead that is invariably associated with multi-user, broad-function, general-purpose systems will be the subject of distaste, disparagement, denunciation, and derision by second-wavers. Their experience with their own machines will accustom them to total control of their working environment; they will seek such control in the corporate working environment as well. They will be impatient with limitations and devoid of consideration for others. One way in which this lack of consideration for the common good will become evident is in the nearly

frivolous abandon with which almost as many new *kinds* of systems are introduced as there are users: Each user will have his own reasons (or lack thereof) for selecting a particular system, and is unlikely to pay much attention to any formal corporate procedures, or even to corporate folklore if it runs counter to his inclinations. As the prices of systems continue to drop you will even find second wavers trying to influence corporate policy to conform to their choices of (home) personal computers.

Although the incidence of second wavers will tend to be greater among the young (personal computers are trendy) and in the technical areas of the enterprise, they will eventually be found throughout the corporation, in all departments, and at all levels. Much has been written about the difficulties of communicating with an upper management who could not understand the computing establishment because of our insistence on surrounding ourselves and our conversations with a cloud of technical arcana. I predict that there will be much greater problems in communicating with an upper management that has achieved an acquaintance with all the right buzzwords, but only insofar as they apply to one brand of personal computer.

The third wave: The Public School Generation

It is only with the arrival of the third wave that the term "computer literacy" begins to be meaningful. These are the people who will have had computer exposure throughout their lives. They will have used them in the home, at play, and in school from the earliest elementary grades all the way through graduate school. They will have both broad experience (the use of many different systems employing many different languages, interfaces, and styles) and appreciable understanding of the nature of computer-driven systems. The metaphor that is most apropos for this wave is that of the automotive society: everybody has seen a car, almost everybody has used one, most people can drive one, and most families own at least one. Untrained and self-taught people can do their own trouble-shooting, maintenance, and modular repair, and while most of us make do with standard models (of which there are a great many), large numbers of people modify both external and internal characteristics and features to suit themselves, and there is a considerable industry devoted to this customization. Finally, after a brief period during which it was a hard-to-manage luxury, the computer -- as did the automobile before it -- will become an easy-to-operate necessity of life.

The most obvious result of this demystification of computing will be the end of the priesthood. (It will not be completely eliminated, of course, any more than the spread of fundamental natural science completely eliminated the Church, but as the schools replaced the priests as the principal source of knowledge about natural phenomena, so will they replace IBM, DEC, and Bell Labs as the principal source of knowledge about informatic phenomena.) The priests will still be among us, but in smaller

numbers, and possessed of less awesome and autocratic authority. They will play a much smaller role in our lives: they will be *ministers* rather than *gurus*.

In time, the third wave will pervade the corporation. All of the white- and most of the blue-collar workforce will belong to it. Most corporate occupations -- not just those in the administrative area -- will have become, if not, indeed, computer-dependent, then at least computer-assisted. In the third wave, the computer will completely replace the tools of many trades, and it will be built into, or provide support for, the tools of most others. Most employees will be trained upon, and will expect to use, computer-driven apparatus, computer-enhanced instruments, and computer-run systems. (Indeed, it may well eventually come to pass that computers have become so much a part of everyone's background that most people will not be conscious of their presence -- at least, not so long as they work properly: rather like one is not conscious of the pervasive presence of electricity until one's child inserts a hairpin into a wall socket. But that will be, at last, the true "computer age" that has been heralded so often and so prematurely in the past decades; that is beyond the third wave.)

In order for the computer to achieve this ubiquity, of course, it must be most dependable. Thus you can be sure that this third wave, too, will have high expectations, and to the expectations of the first wave for functionality, and of the second for performance, this wave will add expectations of extreme reliability. To some extent the expectations of the third wave will be tempered by an understanding of the complexities of the underlying systems, but they will be fanned by knowledge -- and personal experience -- of other systems that performed well in the areas in question. The third wave will be the last and largest, it will have the most knowledge, experience, and understanding of the uses of computers, and it will therefore be the most demanding.

Interlude on "user support"

Before we grapple directly with the problem of supporting the three kinds of "literate" user we shall pause very briefly to consider in rather general terms the nature of user support. Broadly stated, effective user support depends upon knowing your users, both individually and in the aggregate; understanding their expectations and needs (perhaps better than they themselves do!); knowing how well the present systems and procedures satisfy those needs and expectations; and taking whatever steps are necessary to narrow the gaps between needs and expectations on the one hand, and current experience on the other.

Stated somewhat differently, users need

- *tools* that can do their work,

- *interfaces* that make the tools easy to use,
- *knowledge* of what tools exist and how to use them effectively, and
- *assistance* when they encounter difficulty or confusion.

We support them by seeing that all of these things are readily available. As we have seen, however, the three coming waves of increased computer consciousness will contain people with quite disparate attitudes about computers and computer-aided work elements. They will differ in the kinds of tools and interfaces they are comfortable with, in the knowledge they need and how it can most effectively be presented to them, and in the level and nature of assistance that is appropriate to them and to their situations. We shall consider how each of these aspects of user support can best be suited to each of the three waves. As we embark upon that journey, though, it is well to remind you that all three styles of user support may be needed at once in any given organization, for neither of the late waves will necessarily exhaust, replace, or supersede its predecessor(s). Archetypal first and second waves can be expected to linger on in every organization well after the initial rush of the third wave, and they will often be influential people who see no need to change *their* ways to fit the new majority. You must not let the dominance (numerical or political) of one wave blind you to the needs of the others.

Coping with the coming waves: Tools and interfaces

The users comprising the three waves will be very different in their interfacing with computer-driven systems, both in the tools they use and their general style of use. The first wave will be in some sense a surface phenomenon; its members will use very high-level tools, but only in very highly-constrained ways. They will have little direct contact with any computing system *per se*, but will deal with "document preparation", "decision support", and other packaged stand-alone application subsystems. Any tailoring to be done will be handled either by the vendor or by a system czar, and will tend to be at an institutional level rather than at an individual level.

The interface for these users should be totally task-oriented, and have nothing in it of the flavor of data processing. The packages should be menu- or query-driven, preferably making use of function keys, but in no way involving a command language. The results of every action should be immediately self-explanatory: if they cannot be shown directly on the screen (as is the case, for example, when causing a document to be printed at a remote location), an informative message in ordinary English should be presented. (Codes -- as in "Error 35794" -- are not informative, and octal and hexadecimal numbers do not occur in ordinary English.)

Since members of the first wave come from a background generally lacking in computer-related experience, they have little feeling for the kinds of actions that might destroy or damage their documents or data, and, as noted above, the absence of tangible, physical copies of their work product is a source of great uncertainty. As a result, first wavers are generally willing to be protected from themselves -- i.e., they accept the requirement that they reconfirm various actions that are considered to be dangerous or irreversible. They also have -- initially, at least -- little intuitive feeling for safe experimentation. Their training has been to work directly on the task at hand, not on methodological meta-tasks. First-wave users are ill-equipped by either interest or training to investigate systems in order to find out how they actually work; it is therefore extremely important that first-wave systems work *exactly* to specifications.

The members of this group are very good at doing things in the old ways; they are both fast and accurate, and they take pride in these qualities. The people for whom they work have come to expect this high performance, and will expect even more under the new, automated systems. They are thus under a great deal of pressure to do things faster and more accurately than before. They are unwilling -- and often discouraged from taking the time -- to experiment with new ways of doing things. They will tend to have very little patience with things that don't work the first time as they think they should. If it is still possible, when a deadline looms near they will tend to forsake the new ways for the old (they know what to do, how to do it, and how long it will take under the old system).

The second wave will tend to operate at depth: close to -- or directly with -- the underlying hardware. Whereas the first wavers will be more comfortable when restricted to clearly delineated, familiar paths, second wavers will demand control, and will be unwilling to accept any restrictions. While they will be relatively willing to accept *ideas* from the rest of the world, they will be less likely to accept external *implementation* of those ideas: they will tend to use only what they have themselves developed, or, at the very least, personalized. They want tools that will deliver the full power of the system to them. They are fond of secret languages, and they will prefer procedural, command-driven interfaces to the high-level constructs of the first wave. (This is not to say that they will shun high-level languages, however. The second wave has been foreshadowed by those APL programmers who today challenge each other to guess the function of a one-line program, or to shorten it by a character.)

In contrast to the first wave, which will abandon old procedures only slowly and reluctantly, the second wave will distort tasks, if necessary, to make them appropriate fodder for their personal systems.

The third wave will possess some of the characteristics of each its predecessors. They will want the easy access of first-wave interfaces, but with the tailorability of second-wave interfaces. They will use procedural languages if necessary and appropriate, but will avoid the game-playing and other anti-social behavior of the second wave. Control is not so important to them as is convenience. There will be times -- most of the time, in fact -- when they are content to be protected from themselves, but they will also demand the ability to leave the womb upon occasion. They will be more tolerant of system-protecting restrictions than is the second wave, and less tolerant of power-limiting restrictions than is the first.

The third wave will contain many large users of packages, but they will not be limited to the self-contained packages of the first wave. They will use application generators, query languages, document preparation systems, *etc.*, in long -- and nested -- sequences. They will not wish to learn whole new interfaces for every new package. Thus, effective support of the third wave demands great ease of interconnection of systems and packages.

Coping with the coming waves: Knowledge and education

The first of these problems -- how to tell a frustrated user what tools are available that address his specific problem -- is still unsolved. Various things have been tried, none with much success. The most common is an off-line catalogue; it is always unwieldy, usually out of date, rarely complete, and never written from the user's point of view. On-line *help* packages are becoming the norm, but access to them is rarely direct and the search vocabulary tends to be dominated by jargon, and to be extremely idiosyncratic and system dependent. One of the nicest things you could do for your users, of whatever wave, is to provide expanded indexes into such material, making very heavy use of synonyms. In the meantime, the best help is friendly, accessible experts who speak the users' language(s) and who know what is available.

One of the major sources of user dissatisfaction is service expectations that are not matched by present experience. Service expectations arise from a number of sources, including past experience, education, hope, fantasy, and rumor. Once instilled, they are often quite difficult to modify or dislodge, so it is of the utmost importance that the user support staff have a hand in their shaping. Among the ways in which they can do this are: replacing rumor with news, substituting fact for fantasy, tempering hope with reality, wielding what levers are to hand to improve experience, and, above all, providing education of the kind and quality needed by your users. Different classes of users need different kinds of education, on different topics, presented in different ways.

Members of the first wave will be extremely task-oriented. In the words of the old metaphor, they want to know what time it is, not how the clock works. There will be relatively little interest in system exploration and deliberate limit testing (but, because of the absence of preconceptions about what might or might not be possible there will be an enormous amount of incidental limit testing as side effects of naive use of the system). Manual-driven courses that are based upon reference documents will accomplish little in the way of effective training, but manual courses that provide hands-on training in all of the relevant tasks will be quite successful. Formal classes without hands-on practice will generally be a waste of time and effort; cookbooks, examples, and well-structured self-administered training, supplemented with convenient access to congenial expertise, are the training media best suited to the needs and desires of the first wave. (The Boss, and data processing experts, are generally not considered to be congenial, however convenient the access may be.)

To support the first wave properly you must have instructional materials that are very clear, and very closely allied with the actual tasks to be performed. Examples must be tailored to the style, forms, and format of their work, even if it means different sets for every department. You must know the limits of the system and how to translate them into rules of thumb that help your users rather than confuse them. And you must provide expert assistance in the form of consultants who can answer the many kinds of *detailed* questions on how to do quite specific things that will inevitably arise.

Members of the second wave will be more theoretically inclined, will be more interested in reference manuals than cookbooks, and will revel in limit testing. They will especially enjoy seeing what happens when the various prohibitions stated in the reference manuals are violated. (The best way to hold this moderately infantile behavior to reasonable levels (you cannot hope to eliminate it completely) is to make sure those manuals state quite clearly and *accurately* the consequences of any foolish act; after they have tried two or three and found the manual to be correct they will tend to believe it.) This group, however, will be much less tolerant of system limitations than the first wave, in the sense that instead of abandoning the system and doing things in the old ways, they will work very hard to circumvent or demolish the obstacles they encounter, no matter how much misuse or perversion of the system is required. (Don't forget that these are people who trained themselves to be absolute monarchs with respect to the systems they learned on; they will be reluctant to relinquish that role on the systems you support.) Thus one kind of education appropriate for this group consists of defining the full power of the system and how to achieve it.

Another sort of education appropriate for second wavers is system-theoretic. They are interested in implementation detail, particularly with respect to the limits they wish to remove and modules they

wish to tailor. They will welcome a certain amount of formal classroom instruction if it can be pitched at a sufficiently high level of detail. Attempts to keep them ignorant of system internals will generally fail, and will tend to have adverse consequences: They have a fairly active underground, and will readily exchange certain kinds of clever, system-beating tricks; defensive behavior on the part of system hierarchs will often inspire them to use this knowledge.

As always, the third wave will share some attributes of each of the first two. They will be more task-oriented than the second, but more technically inclined than the first. In particular, their instincts for exploration and limit testing will be rather well-developed, but they will be exercised for their own sake, not as means to unearth "forbidden" implementation detail. They will want reference manuals and reasonably high-level tutorials rather than either cookbooks or design notebooks. They will not wish to have to learn implementation detail, but will do so when necessary (to connect two non-compatible packages in series, for example). They would prefer systems that do not demand formal classes for effective use, but will accept such classes if they are necessary.

Coping with the coming waves: Assistance

One's first thought is that assistance is assistance, and, aside from knowledge of the different specific systems and packages in use, it need not take particular cognizance of the kinds of users involved. That, in fact, is not the case. The three waves differ in the kinds of problem they encounter, in the kinds of assistance that are useful to them, in their recognition of the need for assistance, and in the kinds of people from whom they will seek it. They all tend to share one failing, however, and it is something that your first-aid corps needs training in dealing with: Most users, of whatever inclination or stage of development, do not seek assistance for the problem that led them to seek assistance; they seek assistance in implementing their attempted resolution of that problem. The original problem is often buried several layers deep in partial solutions, so that one talent that your helpers and consultants must have is the ability to ferret out the real problem, without giving offence.

Those characteristics of the first wave to which an assistance program should be most responsive are its *task orientation* and its *technophobia*. The first of these determines in large measure the kinds of problems encountered and the way in which they are perceived -- and described -- by the users. Problems will almost invariably be described in terms of the tasks and objects under consideration, not in terms of system entities. For example, a first waver will tend to note merely that "addition doesn't always work", although some might investigate the problem in sufficient detail to be able to state that "addition starts to act funny in the neighborhood of 32,000"; a second-waver will recognize

-- and describe -- the same phenomenon as "the 16-bit problem". Technophobia is reflected in the kinds of assistance that will be sought and, most particularly, the kinds of people from whom it will be sought ... or rather, the kinds of people from whom it will *not* be sought: except in the last resort, it will not be sought from informaticians. And, as the problems are almost always couched in task-specific terms, so the assistance should be directed at the specific task in question.

The strong task orientation of the first wave can provide an excellent template for the preparation of training materials: a set of examples covering the full repertoire of tasks at hand will be very well-received. Explanations of *how* the system accomplishes its work are much less interesting to first-wavers than are the step-by-step recipes for setting it in motion. *Some* descriptive matter is wanted, of course, but it should avoid system internals and concentrate on the visible user/system interface: what actions to perform and the system response to each action, including any delays that might be expected under realistic system loading. Ideally, all system commentary, whether error-inspired or simply environmental or informational, is self-explanatory. Since that ideal has yet to be achieved, supplemental interpretive materials for all system messages are also necessary. These materials are often generated by those with training and experience in the data processing arena, so it is important to supervise their preparation with a highly critical first-wave eye.

Assistance for the second wave is both easier and more difficult to provide than for the first wave. It is easier, because second wavers will indulge in more investigative enterprise than first wavers, thus solving many problems for themselves, and also because they are more likely to be satisfied with clues or hints: after all, it's all just one more marvelous game. On the other hand, they are more difficult to deal with because they push the limits of the system in peculiar ways, so that solving their problems demands intimate knowledge of obscure system characteristics. In particular, it tends to demand encyclopedic knowledge of the behavior of the system as it is driven to its various extremes. Second wavers tend to demand documentation not only of the external characteristics of the system, but also of the internal limits and structures that generate those characteristics -- usually so that they can develop the means to circumvent some particularly galling constraint or restriction.

Second wavers are also probably the worst of the three groups with respect to the proposed-solution-equals-problem syndrome noted above. (Indeed, they sometimes don't want you to know what they are really trying to do.) Because of their penchant for experimentation they may be led to attempt any number of unlikely possibilities, and to lose sight of the the original problem in a merry chase down some interesting side alley. In dealing with the questions raised by second wavers, it is always important to keep your Why's peeled.

The third-wave characteristic that will provide the greatest challenge to the user assistance staff is *inter-package linkage*, the transmogrification of the output from one (perhaps proprietary) package into a form usable as input by another (equally, but differently, proprietary). This has been a particularly troublesome aspect of generalized system use in the past, but one portion of today's user community -- those who use UNIX¹ or UNIX-like systems -- finds it extremely straightforward.

Since it is likely² that a liberal sprinkling of third wavers will have UNIX or UNIX-like experience, they will be familiar with the UNIX "pipe" construct (the automatic chaining of output to input) and with its straightforward redirection of standard input and output; they will expect similar facilities in the systems you offer them, and they will expect such connections to go smoothly.

The other challenge posed by the third wave is breadth, for not only will the number of users expand dramatically with the arrival of this generation of "literacy", but also the average number of (sub)systems in regular use by the user community will explode. The number of possible system/system interfaces increases as the square of the number of systems, and you should assume that all will be tried. Your staff of experts needs to know which are possible (and how to do it) and which are not (and why not, and how to work around the barriers).

Conclusion

It should now be clear that "computer literacy" is not a simple trend, and that adequate preparation for it is not simply a matter of scaling up from whatever you are doing now. One helpful concomitant of the trend towards literacy is a heightened awareness on the part of the vendor community that

¹ UNIX is a trademark of the Bell Telephone Laboratories.

² Members of the third wave will have used computers throughout their lives, including their academic careers. If present trends are any indication, UNIX will be one of the most common computing environments to be found in academia during the development of the third wave. This prediction is not founded upon any great fondness for UNIX but upon the tremendous loyalty it inspires among its faithful, who are to be found in great numbers within the groves of academe. In this it is very like FORTRAN, which, despite its inefficiencies and inelegancies, has survived the scorn of its many detractors; like FORTRAN, the best efforts of the purists notwithstanding, UNIX will persist through many generations of computer users. Whatever you may think of it -- and I think, for instance that its user interface is excessively compact and cute, and particularly unfriendly to the novice -- you will ignore it at your peril.

"user support" means support for real people, not just for computer professionals, and that somehow there is a difference. It is not yet clear that their perception of that difference is either complete or correct, however. It is the job of the local user support staff to extend and modify the vendors' offerings in this area. I hope this note provides some insight into what is involved in that endeavor.

Good luck.

This report was done with support from the Department of Energy. Any conclusions or opinions expressed in this report represent solely those of the author(s) and not necessarily those of The Regents of the University of California, the Lawrence Berkeley Laboratory or the Department of Energy.

Reference to a company or product name does not imply approval or recommendation of the product by the University of California or the U.S. Department of Energy to the exclusion of others that may be suitable.

TECHNICAL INFORMATION DEPARTMENT
LAWRENCE BERKELEY LABORATORY
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA 94720