DOCUMENT RESUME

ED 297 687 HE 021 700

AUTHOR	Heterick, Robert C., Jr.
TITLE	A Single System Image: An Information Systems Strategy. Professional Paper Series, #1.
INSTITUTION	CAUSE, Boulder, Colo.
SPONS AGENCY	Digital Equipment Corp., Maynard, Mass.
PUB DATE	88
NOTE	30p.
AVAILABLE FROM	CAUSE Publications, 737 Twenty-Ninth Street, Boulder, CO 80303 ($16.00).
PUB TYPE	Viewpoints (120)
EDRS PRICE	MF01/PC02 Plus Postage.
DESCRIPTORS	Communications; Computer Managed Instruction; *Computer Uses in Education; *Dial Access Information Systems; Educational Technology; Higher Education; *Information Networks; *Information Systems; Information Technology; *Microcomputers; Online Systems; Users (Information)

ABSTRACT

Strategic planning for information systems and descriptions of the products needed to purvey an institution's information resources as though they were delivered from a single, integrated system are featured, emphasizing strategy and tactics. Various accepted premises for the current state of higher education information systems are set forth. Part I focuses on an information systems strategy (adopting a strategic view, identifying a strategic course, pluralism, and financing). Part II looks at a single system image in terms of the image; implementing the single system image; and issues for knowledge-based implementation. Part III discusses a new paradigm. It notes the paradigm for campus information systems turns from a mainframe-centered Ptolemaic model to a user-centered Copernican model, thus making it necessary to redefine responsibilities. An important goal is creating a single system image that provides the ubiquity and simplicity associated with the telephone system. This effort will require shifting many current burdens from the user to a knowledge-based network and rationalizing the way the user relates to services provided. (SM)

The Professional Association for Computing and Information Technology in Higher Education

A Single System Image: An Information Systems Strategy

By Robert C. Heterick, Jr.

Professional Paper Series, #1

A Single System Image: An Information Systems Strategy

By Robert C. Heterick, Jr.

The Professional Association
for Computing and Information Technology
in Higher Education

Professional Paper Series, #1

Copies of this paper are available to staff of CAUSE member institutions at $8 per copy, to non-members at $16 per copy. Orders should be pre-paid and sent to:

CAUSE Publications
737 Twenty-Ninth Street
Boulder, Colorado 80303
(303) 449-4430

Printed in the United States of America.

This paper was produced on an Apple Macintosh Plus computer and LaserWriter Plus printer using PageMaker software. Hardware and software were contributed to the CAUSE National Office by Apple Computer, Inc., and Aldus Corporation.

PREFACE

All men can see the tactics whereby I conquer, but what none can see is the strategy out of which victory is evolved.

Sun Tsu (3000 B.C.)

This paper addresses both strategy and tactics. It attempts to weave together a previously published article on the subject of strategic planning for information systems with a reasonably detailed description of the products needed to purvey an institution's information resources as though they were delivered from a single, integrated system. Several of the implicitly accepted premises for the current state of higher education information systems are set forth. The impact of technological change in the computer and communications fields is reviewed and its impact on these premises is assessed. The relationship of the institution's own strategic goals to the development of an information infrastructure is also investigated.

The need for a strategy, as opposed to a set of goals, is discussed in light of such rapidly changing technological parameters. A guiding strategy that will take the institution through the next five to ten years is developed by raising a series of typical questions that will need to be resolved during that time period. Specific solutions in hardware, software, and personnel are seen as tactical questions, to be resolved in the light of current technology as they are encountered.

A single system image is proposed as the vehicle within which such tactical questions are resolved. Increased pluralism of native computing environments is seen for the future, with the single system image as the principal strategic element whereby coherency in computing and communications is maintained.

The image provides a single view of electronic mail, data base access, print and plot service, and archival storage for all users—whether on personal, departmental, or institutional computing systems. The single system image offers a development prospectus that can be undertaken in small increments by highly disparate groups, allowing the total campus community to participate in its development. It may be viewed as completing the sixth and seventh layers of the ISO Reference Model.

A few additions and changes in terminology have been made in this current paper. I have corrected a significant omission from the previous version of this work by including more discussion of the role of the library in campus information systems and strategic planning. I am indebted to colleagues, too numerous to mention here, who have contributed significantly to the formation of these perceptions.

Robert C. Heterick, Jr.
May 1988

Acknowledgment

CAUSE appreciates the continued generous support of Digital Equipment Corporation, who funded the publication of this professional paper.

About the Author

Robert C. Heterick, Jr., is Vice President for Information Systems at Virginia Tech. Dr. Heterick has also served Virginia Tech as a professor of Management Science and Director of the Design Science Center, as well as Director of the Computing Center. He has lectured widely on computer systems management and distributed computing in networks of microprocessors, and served as a consultant on those areas to the IBM Corporation, the Governor of Virginia, the President's Privacy Protection Study Commission, the National Academy of Sciences, and numerous universities and private corporations. He currently serves on the CAUSE Board of Directors.

TABLE OF CONTENTS

1
AN INFORMATION SYSTEMS STRATEGY

Wherever we are, it is but a stage on the way to somewhere else, and whatever we do, however well we do it, it is only a preparation to do something else that shall be different.
Robert Louis Stevenson

Most institutions of higher education entered the world of computing machines in the late 1950s or early 1960s with the acquisition of first—or second—generation computers. Institutions continued to grow in their computing activities during the ensuing five to ten years with a succession of second-generation computers. During this time administrative and academic computing activities were established, generally as separate entities.

By the end of the 1960s most institutions made their second major commitment to computing by moving into third-generation computers and, frequently, consolidating all of their computing activities under a single organizational umbrella.

This second major commitment was accompanied by a significant increase in the institution's computing budget and appropriate changes in the institution's organizational structure to complement this new level of activity. From that time, institutional computing activities developed under the auspices of professional management—with the planning and development tools consistent with a major institutional activity.

We are now at another crossroad, marked technologically by the maturation of the fourth generation of computing. The last several years have demonstrated ir plicit recognition of this by several occurrences:

- Many institutions have created a vice presidency for the chief information officer.
- A number of colleges and departments have begun to require the purchase of personal computers by entering students, and many others have the subject under active consideration.
- A new organizational structure to consolidate the communications activities (voice, video, and data) of the institution has been created or is under active study.

The growth of computing and communications technology has merged into a new area that we refer to as information systems—incorporating activities in computing, communications, and data bases.

As we remap our strategic planning efforts, the time seems propitious to ask if we are not at the point of another major institutional commitment. It is generally agreed that the advent of the digital computer signaled the end of the "industrial revolution" and the beginning of the "information age." As computing science and technology have matured we have begun to recognize that we may have overvalued the obvious computational capabilities of computing machines and undervalued the informational capacities that they offer.

This new viewpoint suggests that it may be more than just sound management to rethink the institution's commitment to information systems technology. It may well be appropriate to set new directions and develop new initiatives in light of emerging technology and changing economic parameters.

ADOPTING A STRATEGIC VIEW

Goals are important, but it is the strategies employed that matter the most.
Sun Tsu (3000 B.C.)

Classical approaches to planning usually emphasize the establishment of goals. In a time where technology is growing and and changing so rapidly, such a static approach is clearly myopic. What seems more fruitful is a strategic view of the institution's computing and communications future—a view that attempts to articulate a growth philosophy that permits seizing opportunities when the state of technology is right. Some technological advances are clearly predictable; others are not so easily foreseen. Whatever strategic position the institution assumes *vis-à-vis* computers and communication, it must be predicated on foreseeable technological advances, and flexible enough to accommodate those that are not so easily discernible.

In attempting to develop a strategy that provides for the incorporation of new technology and the flexibility to adapt to unforeseen developments, it is useful first to outline several philosophic opinions about the future of higher education and computer and communications technology:

- Increasingly, institutions will perceive their constituency to be off campus as well as on—the need to reach out across traditional boundaries will assume an increasing importance.
- Interconnection—the capability for human and machine linkages—will become an important issue in higher education.
- The fifth generation notwithstanding, the next round of technological innovation will occur in the communications, not computer, arena.
- The capacity of the institutional budget to absorb continuing developments in computers and communications is significantly attenuated.

It is also useful to put into perspective the current institutional computing environment by describing it in terms of three levels: institutional, departmental, and personal. In many respects these levels may be equated to the "size" of computer: mainframe, minicomputer, and microcomputer.

Institutional computing has been with us the longest and has been subjected to the most extensive study in terms of delivery economies, operational strategies, etc. This is the primary base of computing on the campus, delivered generally on mainframes, and accessed by users via time-sharing terminals. The bulk of computing power is concentrated here, as is the predominant amount of electromagnetic storage and high quality printing and plotting equipment.

The primary source of departmental computing is special-purpose minicomputers. Where there is a relatively small number (10 to 50) of individuals working on closely allied problems, it has been found to be cost effective to create specialized computing environments.

Personal computing is a relatively new concept—one created by the confluence of VLSI and economies of scale in production and marketing. Personal computing permits the creation of highly-specialized, custom-tailored computing environments for the individual. The relative newness of personal computing may go a long way toward explaining why it is the least understood of the three computing environments.

We currently have a somewhat mixed approach to support of computing facilities. Institutional computing support is a central administration task. The situation is less clear when departmental and personal computing support is considered. Generally, departmental computing efforts may be supported centrally by a facilities management arrangement, or directly by the laboratory or department. There are exceptions to this principle as epitomized by central ownership and operation of minicomputers that belong in the departmental category. Personal computing is generally viewed as a college, departmental, or individual faculty or student responsibility. There are exceptions to this principle also, as evidenced by frequent computing center sponsorship of personal computer laboratories.

A further step in developing a comprehensive strategy for information systems is to consider a number of questions that currently present themselves:

- To what extent should computing requirements on campus be satisfied centrally?
- What will be the role of the ubiquitous personal computer?
- To what extent should voice and data communications be merged, and what are the roles of other subnetworks on campus?
- How should electronic mail and office automation be supported?
- Will access to the institutional data base become more widely required, and are current data management techniques appropriate for the next five or ten years?

Each of these questions raises, in turn, a whole host of issues and ancillary questions.

Central Computing Issues

We might begin by asking if central support of departmental and personal computing is appropriate. Should institutional computing be provided centrally, departmental computing by colleges and departments, and personal computing by departments, faculty, and students? A fundamental question is to what extent can a central organization be expected to understand and respond to the diverse computing needs of students, faculty, laboratories, and departments?

This raises the ancillary question of how computing funds should be distributed if some are to be allocated directly to college and departmental budgets. Clearly, financial support for computing is not unlimited. Just as clearly, the closer to the user the funds are spent the better the user's computing needs are likely to be satisfied (at least in the user's perception).

If computing needs are to be satisfied by a more decentralized philosophy, what percent of computing support should be supplied by the institutional budget and what percent by individual faculty, staff, and student purchase? Is it reasonable to expect students to acquire their own computing support? Or faculty and staff to acquire their own?

Finally, if a significant amount of computing resources is supplied by faculty, students, labs, departments, etc., what implications does this have for staff levels and skills in the central computing facility? Should individuals with computer skills be allocated directly to departments and colleges as is currently the case with many administrative departments?

Personal Computing Issues

The time has come to assess the role of personal computing in higher education. An implicit assessment has been made if one reads between the lines of computing policy changes over the last several years. Is it now time to make an explicit statement of policy?

The cost of providing both ends of the communications link for terminals is quite high. The terminal/port ratio is quite low. More widespread use of personal computers (and an operational policy that supports this environment) can significantly improve the terminal/port ratio, thereby reducing (or shifting to other sources) the increasing cost of providing connections to centrally supplied or supported computing and information resources. Strategies for supporting either a network of "dumb" terminals or a network of personal computers are significantly at variance. Is it time to adopt a strategy that favors connection of personal computers in the campus network? Can personal computers provide the bulk of computational support required by the average student and faculty member? Will they?

If personal computer requirements are not dictated centrally, and they probably shouldn't be, how will the institution avoid the chaos that might accompany undirected choice of systems? What should be the role of the central computing facility in dealing with personal computing: should consulting activities be redirected toward the personal computer, should the central negotiation of hardware and software contracts and site licenses for personal computer equipment be expanded?

Communications Issues

Communications issues are the major concerns of the next five years. Any strategy needs to consider to what extent, and how, voice and data networks will be merged. To this must also be appended the question of video networks and the extent to which they should be merged with either voice and/or data.

Perhaps the predominant question is whether or not communications networking should be a subsidized activity of the institution. That is, does the institution have an obligation to provide communications access to staff, students and faculty? Logically prior to that question is whether or not it is economically efficient or educationally necessary to provide such interconnect capabilities? In what ways are the missions of the institution enhanced by a more open set of communication paths? Is a major communications networking capability critical to the growth of the institution? If the institution perceives a need or commitment to service off-campus constituencies, is a relatively wide area communications network (voice, data, and video) necessary? important?

Are significant cost savings or avoidances available to the institution if it chooses to operate its own voice network? Are any potential cost savings offset by a new set of managerial or political problems? In the wide area domain, are there opportunities for microwave operation or band sharing that the institution should investigate? Are there opportunities in satellite communication we should investigate, either for ourselves or in concert with other agencies or institutions?

Will the local (or wide area) distribution of video (or combined voice, video, and data) see a significant increase? Should students have access to data networks from dormitories? Should they have access to video networks? If students in dormitories have such access, how will we handle students who live in town, or participate in off-campus instruction? Should the institution provide faculty with access to the campus data network from their homes?

Office Automation Issues

At most large institutions, current office automation and electronic mail support are provided with institutional mainframe resources via terminal access. Will those users (primarily administrative) continue this mode of operation for the next five years? Considering the number of terminals currently capable of connecting to the local area network, such "dumb" devices are probably in the minority and become an ever smaller minority every year. Should the communication network provide for intelligent device communication without passing through one of the network hosts?

On most campuses the commitment to a mainframe text formatting system is significant and pervasive. Can personal computers provide an equivalent level of support? Will the proliferation of personal computers cause a significant upsurge in the number of large files passing through the communications network? Is there an office automation facility available on personal computers to which we should encourage user migration?

Many campuses currently support many different electronic mail protocols, and several more are in use by individuals and groups that are not supported centrally. Few of these protocols can be used to reach gateways outside the local campus network. To what extent is off-campus electronic communication likely to be important in the future? What protocol should the institution settle upon, and how should we deal with the incompatibilities that this may cause with other heavily used software?

Institutional Data Base Issues

Many institutions made the decision to move to a data base environment in the late 1960s or early 1970s. At that time, the most advanced transaction processing systems were hierarchical or network model systems which provide a relatively high level of security features and are generally robust. They are also expensive and old technology. For the average administrator on campus, spreadsheet relational models are simpler to use and more consistent with what they do on personal computers. Is it appropriate to reconsider the DBMS decision?

Relational technology has certainly matured, but is it sufficiently proven in its operational efficiency to consider moving the institutional data base to a relational model? And if it is, which commercial product should we choose? Is it important to be able to download relational snapshots to data base users? This could be done via current as well as by a relational system: Are the benefits of a relational system sufficient to dictate a change?

In the more ubiquitous personal computer environment of down-loaded file snapshots, there is a host of operational, security, and privacy questions that do not have well-defined answers. In the longer run, can we expect the institutional data base to be distributed across multiple CPUs? Should the hardware and software resources needed to support the institutional data base be considered a departmental-like computing resource? To what extent can user departments develop their own reporting systems if the institution only maintains a central data base ar d the associated data base administration responsibilities?

IDENTIFYING A STRATEGIC COURSE

It is a misfortune, inseparable from human affairs, that public measures are rarely investigated with that spirit of moderation which is essential to a just estimate of their real tendency to advance or obstruct the public good; and that this spirit is more apt to be diminished than promoted by those occasions which require an unusual exercise of it.

James Madison
The Federalist

The foregoing litany of questions doesn't identify the strategic course we should set, but it does raise a sufficient set of problems to suggest that it is appropriate to forge a strategic plan. Whatever strategy finally evolves, it should be one that provides a philosophy in which these and other questions can be reconciled. In general, it is probably inappropriate to try to craft the strategy from specific answers to such questions. Rather, we should attempt to discern what role the institution plans to establish for itself and, within the context of how the institution views itself, set a compatible information systems strategy.

Underlying all the institution's aspirations is the need for an effective information systems infrastructure. This infrastructure can be divided into three major components:

- Communication networks (voice, video, and data)
- Computing support (institutional, departmental, and personal)
- Professional staff (communications, computing, and managerial)

Communications Networks

Most institutions have only recently begun to recognize the effort required to put into place a comprehensive communications network. To a large extent, this is due to the autonomous development of voice, video, and data communications. The major step of placing all these activities under a single management has yet to be taken for most institutions.

Voice service is probably the most broadly used of the categories. The two principal planning issues in this arena have to do with: (1) the merger of voice with video and data in the institution's network; and (2) the associated question of the appropriate division of responsibility between the institution and the local common carrier. For strategic purposes, it is probably useful to divide this issue on a geographical basis, with those voice applications required off campus generally falling within the purview of a local operating company, and those on campus that might be operated by the institution.

It is not likely that the institution can provide voice service levels that exceed those provided by the common carrier. The rationale for institutional operation of voice services arises from the significant concern that common carrier operation of the campus voice network leads to substantial sub-optimization. Put in the positive, by integrating voice with existing video and data networks, there may be significant cost savings available through optimization of a larger network — avoidance of duplicative cabling, bandwidth sharing, etc. While a digital switching facility permits a more rational integration of voice and data services, it also presents the opportunity for new services. Principal among these is voice mail, the ability to digitally store voice messages to be picked up later by the intended recipient.

At the moment, and for the next several years, data communication is the high growth area that is forcing decisions regarding the communication infrastructure of the institution. We can identify four major data communication areas: intraoffice networks; intrabuilding networks; interbuilding networks; and wide area networks.

The proliferation of personal computers is bringing with it a demand for intraoffice networking—the networking of several (perhaps up to six or eight) personal computers in a departmental or laboratory office. Most users perceive that there will be a dominant node in this network, probably providing large fixed disk storage, high quality printing, or some other sharable resource that is cost effective when distributed across a half dozen or so devices. The advent of 32-bit, time-shared micros will only accelerate this requirement.

Such networks can be created with logic cards for the personal computers and cable that is installed departmentally. There exists the potential for literally hundreds of such networks to develop—with a great diversity of products and protocols. This is not necessarily bad, provided the network contains a server that provides a gateway to the campus (or intrabuilding) network. The difficulty with such homegrown networks is that they may prove to be unsatisfactory because of user naivete and consequently represent a less than desirable expenditure of scarce resources.

While the development of intraoffice networks is debatably a central support issue, intrabuilding and interbuilding wiring and protocol standards must clearly be established centrally. There are many issues here that require some debate before standardization. As with most construction-related efforts, more cost-effective installation is likely if longer-term needs and technological developments are considered prior to initial installation.

The issue of wide-area networking is the most amorphous of all, related as it is to the strategic plan of the institution in terms of reaching off-campus constituencies. Clearly, the more serious the institution is in its plans to engage in significant off-campus efforts, the more critical is the aspect of the infrastructure. Some rather specific guidance on this issue is called for.

Computing Support

As defined previously, computing support can be looked at in terms of institutional, departmental, and personal effort. In identifying a strategy, it may be appropriate to start with the personal level, as this seems to be the current driving force in the computing arena.

From the strategic viewpoint, the paramount question is whether personal computing will, in the next five years, provide the principal base of computing support for the institution. If it will, then clearly we need to rethink our approach to institutional computing and set new directions.

One way to approach this issue might be to consider whether, and under what circumstances, we can expect a large percentage of students, faculty, and staff to have relatively immediate access to personal computing. Certainly by 1990 we can expect 32-bit, 20MHz (5-10 MIPS) processors that can be purchased at prices approximately equal to current-generation machines. This would imply that the power of today's minis would be available in desktop (probably lapsize, if desired), personal machines. If the prices of computers continue their current trend, then today's high-end machines will cost less than $1,000 in 1990, and such machines should be powerful enough and sufficiently inexpensive that access to one could be taken for granted. In brief, there seems to be no technological or financial impediment to assuming that by 1990, anyone who wants a personal computer will have one.

There are still a number of activities that should be provided centrally as part of the institutional base. These would include the institutional data base, printing service, graphics service, archiving, supercomputing, and specialized software. One might very well extend the list, but the characteristics of what might make an institutional service item are probably more important and interesting. An institutional service seems in order where a broad cross section of users must access the same data. This is certainly the case with the administrative records of the institution and access to the library catalogue.

An institutional service also seems in order for very high-priced peripheral devices. High-volume laser printing, high resolution graphics, microforms, photographic media, and typesetting are typical devices that would fall into this category. Another category of this type is high-volume storage devices for machine-readable data.

There will continue to be a need for special purpose software which either must run on a large mainframe or is too expensive to acquire unless its cost is amortized over a larger user community. Increasingly, we can expect a role for supercomputing related to such software packages. The supercomputer of the 1990s will be on the order of 10 gigaFLOPS, bringing a new set of problems within the realm of practical computability.

If personal computing is expected to provide the bulk of computing service for most users, then the institutional service should redirect its support efforts from "dumb" terminals to intelligent workstations. This transition will take time, perhaps five years. During this transition period there will likely be a number of temporary tasks that will also become candidates for institutional support—store and forward of electronic mail, gateways to external networks, and file transfer are examples.

The economics of computing technology suggests that departmental computing will become more important in the future. Our economy has always favored specialization, and value-added services in computing are becoming increasingly commonplace. If we accept the principle that departmental facilities service a limited base of users, we can identify a number of computing services that would likely be of the departmental category: office automation, CAD/CAM, CAI and CMI, library automation, and robotics.

Again, it is probably more interesting to discern the characteristics that make a computing service a candidate for the departmental category. Clearly, one such characteristic is the need for computing power beyond that likely to be available on the next several generations of personal computers. One characteristic of the preceding list is the intensiveness of computing for the activities mentioned.

Perhaps the thorniest question regarding departmental computing is the source of funding. Where total funding is from soft sources such as grants and contracts this is not an issue. More generally, we need to resolve to what extent funding for such facilities should be considered an institutional question. If departmental facilities are at least partially subsidized by the institution (and all are at least to the extent of certain overhead costs), we need to determine what portion of the cost should be subsidized and how the institutional facility plan for such acquisitions will provide for the disbursement of funds.

There is a second-order effect attributable to departmental facilities that may well raise issues more critical than those of cost sharing. Departmental facilities require systems programming support, site preparation, gateways or bridging into the campus communication network, and other labor intensive support activities. Replicating the skills available in the institutional facility ten or more times over is probably not possible, much less cost effective. To some extent, certain of these labor intensive skills will be provided by faculty and graduate students. There is a real question as to whether this is cost effective and whether the support level will be adequate.

Professional Staff

Some infrastructure questions relating to professional staff were raised earlier with respect to central computing issues. The primary strategic focus in this area must be on skills that will be required, the number of personnel necessary, and the distribution of those individuals. This concern is exacerbated by the shortage of skilled computer professionals, the even greater shortage of highly qualified technical management personnel and, as a consequence, the extraordinary salary levels commanded by individuals with these scarce talents.

Consider the skills likely to be of most value in the next several years. As the campus builds toward thousands of intelligent workstations, the first-level, and likely predominant, need of users for information will concern their personal computer hardware, software, and interface to various communication networks. When we consider the number of different personal computers on the marketplace (probably over one hundred) and the number of software packages (probably close to 10,000), we recognize that it is not possible to provide expert consulting for all of them. We might possibly consider specializing in one or two machines and perhaps a dozen pieces of software. Given that the half-life of micro hardware and software technology is about 18 months, and given that at any time it is likely that three generations of the technology will be in common use on campus, the problem of expert consulting still seems beyond reasonable grasp.

It seems inevitable that the burden of consulting will have to shift even further to faculty and students. Attempts to proscribe either hardware or software configurations seem antithetical to the purposes of a university and should probably be avoided. The role of central computing staff in the personal computer milieu needs to be addressed. There would seem to be several initiatives that will need to be maintained centrally: standards for electronic mail, institutional printing access, institutional file service, system maintenance of institutional software packages, and arrangements for quantity purchase of selected personal computer hardware and software.

The foregoing list is not all-inclusive, but is indicative of the types of activities that ought to be provided centrally. The common characteristic of these activities is to support devices, protocols or software, rather than direct support of the user. As the personal computer workstation becomes the norm on campus, the capacity and capability to provide user consulting is diminished, but the need to provide network access to special services is amplified.

As the institutional service becomes more directed to device support, it is reasonable to imagine direct user charges for items produced, rather than for computing equipment exercised in their production. For instance, we can imagine charges imposed for pages of output printed, slides or microforms produced, graphic frames plotted, etc. In this environment we can envision page printing to be a service of the Print Shop rather than the Computing Center. We can envision slide and graphic production to be a service of the Learning Resources Center rather than the Computing Center—the focus of institutional computing support shifting to provision of the network intelligence to provide the interconnect capability for the devices in the network.

PLURALISM

There is nothing more difficult to carry out, nor more doubtful of success, nor more dangerous to handle, than to initiate a new order of things. For the reformer has enemies in all who profit by the old order, and only luke-warm defenders in all those who would profit by the new order. This luke-warmness arises partly from fear of their adversaries, who have the law in their favor; and partly from the incredulity of mankind, who do not truly believe in anything new until they have had actual experience of it.

Machiavelli
The Prince (1513)

If there is one notable characteristic of this new generation of computing it would have to be pluralism. The age of centrally directed and supported computing has passed along with typing pools, keypunch and word processing centers, and the punched card itself. The economy of scale phenomenon applies only to a few hardware devices attached to computers and to the computer-communications infrastructure itself.

Whatever strategic plan we craft must recognize this as the paramount fact of modern computing and communications technology. The freedom to innovate is also the freedom to fail. As a consequence, we must be prepared to have groups propose, and implement, computing environments that have little apparent chance of providing successful, long-term solutions to the problems for which they have been devised. Hopefully, an occasional brilliant success that couldn't have been predicted in advance will help offset some of the naive failures.

Pluralism will be most evident in the personal and departmental computing areas. The tendency of users to want highly specialized computing environments that are mirror images of the latest technology being used in industry will accelerate. Most of these environments will be clearly ephemeral, seldom having an industrial life of more than a few years. The truly difficult problem will be to encourage the user community to find some way to teach principles devoid of their current technological trappings and packaging. As with any relatively new technology, it is difficult to ascertain the underlying principles, and tough to argue against academic programs that want to provide "hands on" experience for students. The major brake on unnecessary or unwise technological acquisitions will have to come from carefully designed funding strategies.

In the personal computer arena, the tendency to want to move almost annually to the latest technology will be difficult to blunt. Aside from engineering and computer science, it will be difficult to justify significant student expenditures for personal computers. An expenditure of more than a semester's tuition on a computer is too much. Most recent statistics suggest that the retail price of the software being acquired for the personal computer is about equal to the cost of the machine itself. Requiring specific hardware or software configurations on the part of students in particular programs will eventually isolate those academic programs from the institution at large. Unchecked, this kind of pluralism has the potential to destroy the institution as a community.

Along with the freedom to select various configurations of software and hardware must come an understanding of the need to make generic computing requirements. When a spreadsheet is required it should not be specified to be precisely Lotus, or VisiCalc, or .... At this juncture the faculty have done little to solve this problem, and until it is resolved the unchecked growth of personal computing must be viewed as a mixed blessing. Fundamental to our information systems strategy must be some mechanism to bring the issue to the table for consensual resolution. For anyone who is currently teaching courses in which the students frequently have their own personal computers, the Tower of Babel analogy is readily apparent.

This pluralistic computing environment will place increasing pressure on centrally provided services to have some cognizance of the multiplicity of software/hardware configurations that are attempting to access institutional services. It is clearly not possible or economically sensible to centrally design software that is capable of printing a file edited on any one of a hundred different what-you-see-is-what-you-get text formatters. A similar observation could be made regarding downloading file snapshots in a format suitable for any one of twenty commonly-used spreadsheet programs.

FINANCING

People who don't count, don't count.

Anatole France

In at least one respect financing presents the most difficult aspect of crafting a strategic plan. It can be argued that it is not the responsibility, or proper role, of Information Systems to locate and develop funding sources—it should simply manage, as effectively as possible, the funds allocated to it by the institution. Of course, in the push and shove of the budgetary process it can emerge with either more or less funding. The important aspect of the budgetary process is that it won't be much more, or much less.

However, if the perception that we are at the point of considering another major institutional commitment to information systems is correct, then funding cannot be ignored. For it is one thing to consider reallocating current funds and quite another to consider assuming initiatives beyond those already assumed. If the strategy is built around a relatively level (in constant dollars) budget, then questions of what current services to restrict or curtail, what current users to reduce or eliminate services to, become paramount issues. If new funds are available, then service changes can be phased in over longer periods of time, causing relatively minor dislocations of user service levels.

It is probably appropriate to discern more subtle shifts of funding responsibility that have occurred in the last few years.

- Some colleges and departments require students to acquire their own computing resources.

- Many faculty have acquired their own personal computers.

- Most colleges and departments provide access to personal computers or terminals for their faculty and staff.

- A significant percentage of departmental grant funds are expended on general purpose computing equipment.

- Colleges and departments have begun to assume the local area network connection cost in addition to connect time charges on mainframes.

All these charges, once covered out of the central computing budget, have shifted to local units (colleges and departments primarily), frequently with no commensurate shift of budgeted computing funds. As a general rule, institutional computing service is a subsidized commodity, paid for centrally in the computing budget, and rationed through a computer allocation scheme. Local and departmental computing resources are paid for from grant, college, or departmental funds accumulated through operating budget set-backs or expenditure reallocations. Interconnect charges (modems, line costs, multiplexors, etc.) have risen from a negligible item in most departmental budgets several years ago to, for many departments, the single largest charge item in their budget. Capital expenditures for personal computers have risen from almost nothing three years ago to, for many departments, the single largest capital outlay expenditure in their budget. The institution through equipment modernization and other supplemental programs, needs to contribute significantly, although indirectly, to these acquisitions.

As a general rule, inter-connect costs are direct costs paid by colleges and departments from operating budgets. Communication network developments are paced by revenues generated from current services. This situation is tempered, although only slightly, by the previous investment in communications equipment paid for from subsidized computing operations. It would take the wisdom of Solomon to untangle this Gordian knot of support for computing and communications. To some extent, the knot is severed with the establishment of a consolidated communications group, but an underlying rationale for funding disbursement seems not to have been found—or at least not uniformly applied at most institutions.

Funding is an extremely important issue because we seem to be trying to continue doing all that we were doing before (and absorbing natural and inflationary growth) and concurrently starting a major new program in voice, data, and video networking. The only way we can continue on this path and grow the communications networks in the time frames and cost efficiencies that seem reasonable is with the infusion of new funds. Without, at this time, attempting a detailed assessment of funds required, and ignoring any shortfall of funding that may currently be perceived to exist, this shortfall represents the infusion of several million dollars on a one-time basis for most campuses.

While it is reasonable to believe that the largest part of the communications network costs can be shown to be recoverable in a five- to ten-year period (via more cost effective operation by the institution as compared to a common carrier), such an argument is not one that should be made. A cost saving argument presupposes a "business as usual" condition that is significantly at odds with the perception that the institution is poised for a new spurt of activity which involves significant changes in its goals trajectory. This next set of aspirations of the institution requires the establishment of a sophisticated computer and communications infrastructure—the communications part of which is not in place, and the computer part of which is in need of considerable tuning.

Institutions are increasingly turning to non-traditional funding sources for needed infusions of capital and such a path could be pursued for the computer and communications infrastructure. Without making any judgments here as to the desirability or probability of success of alternate strategies, the following suggest themselves:

- Joint ventures with industrial partners in the computer and communications business;

- Major negotiation efforts with foundations that have particular interest in educational delivery systems, particularly non-traditional systems;

- The trading of industrial park space with a suitable industrial partner for a "named" project activity in computers and communications; and

- State- or institution-supported revenue bond issues.

2
A SINGLE SYSTEM IMAGE

Software integration is the holy grail of information resource management.
Daniel Appelton

The historic development of information systems has brought us through three readily definable stages:

- Transaction Processing
- Management information
- Decision Support

The first, and immediate need, was for a system to support high volume transaction processing in each of the functional areas (accounting, purchasing, personnel, student records, etc.) of the institution. The data base management system (DBMS) needed to provide a high level of reliability (backup and restore features) and data item level security and authorization protection as well as high volume transaction support.

The second stage was reached when several of the functional areas had been brought on-line and cross-area access could be used to produce management information. This management information was provided in the form of visual display screens providing pre-formatted responses to fixed queries, but on-demand, in user time (synchronously) rather than as scheduled, printed (asynchronous) reports. User requests for information were evaluated, and those with sufficiently high frequency were programmed into the management information facilities of the system.

The system eventually would need to respond to *ad hoc*, unstructured queries coming from a much broader user community. As a consequence of the proliferation of intelligent workstations, there has been a commensurate increase in distributed computing, native user environments (Lotus, BASIC, dBASE, etc.) and naive, first-time computer users.

This growth, and change in character, of the user community has brought, rather forcefully, to the attention of users that there is a significant difference in the requirements for transaction processing and decision support. Not only are there many more users desiring access to administrative information, but this access is increasingly less structured—both in the form of information requests and the format in which the response is desired. This growth has fostered an equivalent growth in the backlog of management information system components—a backlog that is paralleled throughout industry as well as higher education.

From the current vantage point, it appears that there are four types of systems that are reasonably differentiable in the University's information systems environment:

TPS—Transaction Processing Systems play a fundamental role in the initial capture of data and provide the major resource for functional offices.

MIS—Management Information Systems provide a broad cross section of secondary users with pre-formatted, easily accessible, query responses. These responses frequently require data from multiple functional areas.

DSS—Decision Support Systems are on the increase as users begin to develop significant computing capabilities on their own intelligent workstations. Unlike the MIS demand, this is primarily for access to information with the user supplying the processing environment.

KBS—Knowledge-Based Systems appear to offer significantly enhanced decision support for naive users. These types of systems are extremely difficult to construct and are currently at the forefront of software engineering research.

Viewed on the dimensions of "query structure" and "user capability" (in the sense of computer competency) these systems appear as shown in Figure 1. TPSs are sometimes referred to as "closed" systems, DSSs as "open" systems—the less structured the query, the more open a system is required. On the other dimension, the more naive the user, the less capable he or she will be of handling an open system.

There are several guiding principles of an institution's information systems that are important and appropriate.

Figure 1. Open-Closed System Spectrum

Query Structure		*User Capability*: naive	*User Capability*: skilled
	high	TPS	MIS
	low	KBS	DSS

- Source point data capture
- Value-added data handling
- Destination point documentation

These are principles associated with the transaction processing aspect of the institution's information systems. Nothing in any perception of the changing information systems environment on campus suggests that transaction processing is not still an important component.

What is equally evident, and superficially paradoxical, is that the most appropriate decision regarding transaction processing may well be most inappropriate when viewed from the domain of decision support. At least two courses of action suggest themselves:

- Find a DBMS that strikes the best balance between the apparent conflicts of the two needs.
- Find a way to separate the two activities so that choices for data base management and decision support may be made in restricted, non-overlapping domains.

It is the data custodian who must define in what way, how often, and by whom data may be captured, updated, accessed and deleted. In the process of this definition there will arise, at least implicitly, information upon which decisions regarding DBMSs may be made. The custodian might well desire data to be collected and stored at one level of detail, but accessed for decision purposes at another. In fact, a compelling case can be made that the typical decision maker might not even be able to navigate through the data dictionary for the data kept at its most primitive level. The custodian may well decide that there are some data not available for direct query by users. The custodian has responsibilities beyond the simple capture and storage of information. Those responsibilities include, at least, the availability of information that may be as unambiguously interpreted as possible.

This multi-level structure is frequently referred to as storage, conceptual, and user views of the data. Among the purposes of management information systems is the protection of the user from needing to know the storage and, frequently, the conceptual schema; that is, making the user interface as simple as possible.

It seems natural that this process of making transparent and simplifying interfaces be extended to the question earlier raised—should a single DBMS be chosen to address both transaction processing and decision support requirements?

In a properly segmented set of data views, the choice of transaction processing system can be made, and changed, independently of decision support requirements.

When we consider the likely capabilities of workstations that will be available within five years, both in terms of processing capacity and data storage, such workstations will be more than capable of supporting significantly large data bases. Such machines are likely to be 20MHz, 32 bit devices with several MBytes of internal storage and hundreds of MBytes of disk storage.[1] The availability of such workstations will accelerate current trends toward both distributed data and computing. In short, the administrative records of the institution are likely to be found, as well as used, in a heterogenous, distributed computing environment.

It also seems likely that access to many data bases, external as well as internal, will be desired for decision support purposes. Such a view requires that we treat the issue of administrative systems as only a subset of a larger problem. When a solution to the larger problem is found, ease of access to administrative data will naturally follow. It also seems likely that "dumb" terminals, while they will provide appropriate native computing environments for some, will become a distinct minority in the mix of workstations on campus.

The critical issue for users with intelligent workstations will be access, not processing. While there will continue to exist a class of user who desires some pre-defined processing of information from a data base (what we have referred to as the MIS function), the vast majority of users will wish to do their own processing. Users who desire pre-defined processing will continue to depend on some other organization to support their interface. As a general principle, it seems preferable for this support function to be located as close to the user (and the data custodian) as possible. This suggests

[1] John P. Crecine, "The Next Generation of Personal Computers," *Science*, 28 February 1986, pp. 935-943, contains some prognostications of interest.

that an increasing amount of support will come from functional offices (particularly the custodian's office) rather than from a systems development group.

These intermediaries (custodian and functional offices) will have the same characteristics as the bulk of the user population with intelligent workstations. Their major concern will also be ease of access. In addition, the custodian must define and share security responsibilities. In the distributed environment privacy will be considerably more difficult to insure. While it is true that the major change has come in the domain in which security is to be invoked—from paper, to centralized (primarily hard-wired) terminal systems, and now to distributed data and processing—the threshold of improper access may both be lower and not seem to require blatantly illegal action on the part of the trespasser. At least, a new round of user education is called for.

THE IMAGE

... a distributed system should look like a centralized system to the user. The problems of distributed systems are (or should be) system problems, not user problems.

C. J. Date

The proliferation of personal computers into the institution has occurred so rapidly that the central computing and communications organizations have not had the time or resources to adequately provide an appropriate computing environment for them. To date, most institutions have supported terminal emulation, aided by file transfer protocols, as the only means for the personal computer user to access the central organization's data base and peripherals. Electronic mail is usually accessed via terminal emulation on one of the institution's mainframe computing systems.

This situation raises the threshold of detailed knowledge required of the user—precisely the opposite of what the situation should be and the inverse of the reason most users acquired a personal computer in the first place. The lack of appropriate networking support has only exacerbated the situation.

The situation is similar for users of special purpose minicomputer systems who have chosen such a system for their native environment in order to have access to CAD/CAM, laboratory automation, specialized operating systems, etc. In fact, many mainframe users can be similarly classified. Some users have chosen a mainframe environment for access to sophisticated office automation facilities or access to specialized compilers or software not available on the current crop of personal computers.

We can use the term "native environment" to encompass this broad category of users who have selected a micro, mini, or mainframe subsystem for what it has to offer their individual computational and information needs. Even when registered on one of the host mainframes, these users may have an extremely limited knowledge of their complete environment, choosing to see themselves as office automation users for instance, not knowledgeable in the whole gamut of other operating system services available to them. Further, they are probably not the least bit interested in learning the protocols necessary to use them.

What these users need and want is an institutional environment in which they can use the native environment for the specific set of tasks for which it was selected, without forfeiting access to electronic mail, organizational data bases, or sophisticated peripherals found in the organization's computing network. Accompanying the proliferation of personal computers has been a commensurate proliferation of user native environments—Lotus, dBASE, Wordstar, BASIC, etc. The user needs a simple extension to the native environment that provides access to the aforementioned organizational resources. Further, that extension should appear the same to every user, irrespective of the particular native environment.

We can refer to this extension as a single system image (SSI). The user wishes to see the image of a single system—one which is a natural extension of his or her native environment. Such an image should be free of specific protocols associated with one or more of the institution's central computing systems. In fact, any association with one of those systems should be totally transparent.

It would appear that the future of intelligent workstations versus shared resource, interactive computing will parallel that of the private auto and mass transit. While the public has indicated that they believe in the economies of mass transit, they generally find it individually dysfunctional—or at least personally restrictive—to the point that they continue to prefer the private auto. Much of the same psychology enters into the decision of personal versus shared-resource computing. The first wave of the personal computing frenzy may have begun to recede, but as ever more sophisticated workstations come to the market, we can expect personal computing to proliferate even further.

The image could be implemented as an extended electronic mail system. In addition to a conventional mail facility that provided access to a half dozen or more national networks (BITNET, ARPAnet, CSNET, NSFNET, etc.), the mail system could provide access to organizational data bases, high speed laser printers, typesetting equipment, flat bed plotters, high capacity disk drives, etc. The plethora of native environments suggests that the single system image will be complex and its development will need to be shared between the central computing and communications organizations and the community of personal computer users.

Synchronous access to distributed data is still at the frontier of research in computer science. To be sure, there are several data base systems that purport to manage distributed data. In every case a similar software package resides on a similar hardware architecture. This is not the problem that many institutions will need to address. Commercially viable versions of a solution are not even likely in the next five years. On the other hand, an asynchronous solution such as electronic mail appears to offer most of the functionality of the solution we would hope to see—and it appears that it can be constructed in a relatively short time with limited resources.

Electronic Mail

A mail service would need to permit the user to compose the mail item in his or her native environment and communicate the item to the network and from there via network service machines and/or host mainframe facilities to the intended recipient. The actual transfer of the mail item could be via a personal computer file to a host file, but should not be restricted to such. In fact, many, if not most, personal computer users will opt to transfer an object which is not a DOS or UNIX file. The object might be one or more cells from a spreadsheet, a programming language variable, or an object created by some piece of "desk organizer" software.

From the native environment, the user would need to connect to the network. Having made such a connection, a message to a host or service machine would invoke some executive routine which would accept the mail item from the personal computer (this embodies some or all of the characteristics of a file transfer protocol such as Kermit or PCTRANS) and dispatch it via the host system or service machine electronic mail facilities. The user need know only the name of the executive routine to invoke and the user ID, node, and network of the intended recipient.

Many current workstations are limited to 19.2 Kbps and are configured with an inexpensive asynchronous interface. As many users will access the system in a dial-up mode, such asynchronous connectivity must be supported. However, to get maximum efficiency from the network and to avoid dedicating workstations during large file transfers, a high efficiency protocol should be provided for.[2] The next generation of workstations will likely be true multitasking machines with higher I/O port transfer rates.

The user should have the facility to request that a particular host or server system file be sent to the personal computer. The mail item would be displayed on the screen, captured by a native environment variable, or placed in a workstation file. Some users may prefer to archive mail items on a network server or host system, others will prefer saving mail items in a workstation file or as some type of object in their native environment.

[2] See, for instance, Andrew Tannenbaum, *Computer Networks* (New York, NY: Prentice-Hall,1988), for additional discussion of sliding window protocols.

Archive Service

The personal computer user would likely desire some form of archive service from the network. An archive would provide for disposition of mail histories, workstation file backups, or temporary high volume storage external to the personal computer. The economics of storage technology indicate that large volume storage on network hosts or service machines will continue to be the most inexpensive approach until multiple write laser disc technology is commercially available.

The archive could be implemented via electronic mail by means of an executive routine capable of storing and retrieving items using host system or service machine protocols. The user could mail an item to the archive for storage or mail a request to the archive to retrieve a previously stored item.

Print Service

A rudimentary print service could be quite easily implemented as an electronic mail facility. The major stumbling blocks involve the location (personal computer or host) where the file was formatted and the compatibility of the escape sequences for print control in the formatted object and the target printing device. For many of the what-you-see-is-what-you-get personal computer formatters this is far from a trivial problem.

In order to accommodate both formatting strategies it is probably appropriate to devise an institutional printing standard for word processing codes which would permit any formatted file to be "sifted" and altered to the standard. Translation from the standard to specific network devices could then be constructed (there should not be many of them needed) to map the file to a specific printing device.

Most special printers will require that the font combinations desired also be specified—probably in the request rather than in the file to be printed. There are also potential problems associated with specifying the type and/or size of the paper to be used. Neither of these issues appears to present significant problems that could not be easily accommodated in a mail format.

The definition of a standard set of word processing codes will be a difficult problem—one sure to raise heated debate from several quarters. There are several emerging "standards" for both layout and mark-up. A cursory review of them suggests that they will not accomplish what is required of the standard proposed here, although they may offer many good pointers for such a specification.

Office Automation

We can expect many users will, for some time, continue to prefer an office automation environment as a terminal user on a central computing facility. These users will desire to have the processing routines and interface to the single system image constructed for them. There seems little advantage in constructing similar interfaces which would operate on intelligent workstations. When data base access is added to the functionality already present in such systems we might expect a significant new demand for services to develop.

IMPLEMENTING THE SSI

Nothing will ever be accomplished if all possible objections must be first overcome.

Samuel Johnson

Access to distributed data bases is considerably complicated by questions of authorization, privacy, security, etc. Ignoring, for the moment, these very real and important questions, we could formulate a simple electronic mail environment for gaining access to a data base. It seems fair to note that this environment would dramatically affect both the number and size of messages flowing through the campus communication networks.

Access could be specified in terms of some real or some canonical language. For our purposes here we will take the Structured Query Language (SQL) as a model. SQL, as most other data base languages, has operators for file creation, record addition, modification and deletion, and generally a number of other useful functions. We will temporarily restrict our consideration to record selection from a relational model of the data. We will assume that some set of routines is capable of producing a relational view, irrespective of the actual data model implemented in the data base and the data base management system. Further, we will assume that some metadata definitions of actual or logical relations (user views) have been implemented to relieve the user of the necessity of having detailed knowledge of the primitive data base design and data dictionary.

Conceptually, access to data requires navigating through both the single system image and the DBMS of the target data. The DBMS is required to physically manipulate the data elements of each particular data base (although there may be participating "data bases" that are simple flat or sequential files with no DBMS) and there may be a large number of DBMSs involved. The single system image then is data about data (metadata) that provides information about the physical location of data bases, rules to decide if some canonical language request is properly formed, first-level security and authorization checks and translation rules.

Figure 2 shows a summary of the roles of application routines and the DBMS or SSI for different data types that may be found in the data bases participating in the SSI.

Figure 2. SSI Conceptual Model

DATA TYPE	DATABASE		METADATA	
	Application	DBMS	Operation	SSI
Numeric	arithmetic sort etc.	store	logical relational	DIRECTOR
Text	catenate substring etc.	retrieve	lexical	ADMINIS-TRATOR
Geometric	intersect union etc.	update	relations	ACCESSOR
Image (bitmaps)	clip composite	etc.		TRANSLA-TOR

A conceptual model of the image could be constructed of five functions:

COMMUNICATOR—the hardware, software and protocols required to physically transport data between source and target systems.

ACCESSOR—the data manipulation functions required to query (perhaps store) data anywhere in the single system image, and to route data (perhaps electronic mail, print files, etc.) to a target.

TRANSLATOR—the ability to translate data and/or ACCESSOR commands from source to target formats.

DIRECTOR—knowledge of where data may be found or how targets may be accessed, if data need to be translated or headers modified, how the target is addressed, etc.

ADMINISTRATOR—rules for access and update authority, invocation of consistency requirements, privacy and integrity requirements, etc.

We can consider the COMMUNICATOR role to be played by the local campus networks and discuss only some low-level protocol issues. The ACCESSOR, DIRECTOR, and ADMINISTRATOR functions of the image, however, must be defined beyond a brief discussion of low-level protocols. Finally, the role of the TRANSLATOR is deferred to the discussion of a knowledge-based representation of the image in the next section.

The COMMUNICATOR

At least initially, most of the workstations participating in the single system image would communicate with a service machine using existing asynchronous interfaces and networks. Asynchronous serial communication has traditionally been used for communication of dumb terminals with host computer systems. In this environment, data are typically transmitted in an unstructured format. Such transmission is not appropriate for the reliable exchange of large amounts of data between computer systems in the single system image. Electrical noise and marginal electronic components can cause transmitted data to be lost or corrupted.

A low-level protocol can be implemented on an asynchronous network to insure data integrity and prevent loss of data. Use of block sequence numbers, error detection codes, and retransmission of corrupted or lost blocks are the usual techniques for constructing protocols that ensure data integrity. In addition to insuring reliable data transmission, the protocol used for the single system image should be efficient in order to allow large amounts of data to be transmitted over networks that function at relatively low data rates. Protocols that take advantage of the full-duplex nature of modern asynchronous networks are prime candidates for the SSI low-level protocol. Sliding window and piggybacking protocols allow the sender to transmit several blocks before receiving an acknowledgement for the first block, thus significantly reducing the wait time between transmissions.

The protocol used for the single system image could be used to implement a limited security scheme. Messages exchanged by a workstation and network service machine could include a session password. This password could be assigned by the network server during an initial identification sequence during which time the user is validated by a user password. Following the initial sequence, all messages exchanged by the workstation and the network service machine would contain the session password. Since a workstation may connect to and disconnect from the network many times during an active session, the session password would allow a session to be identified without the need to invoke the full user validation exchange during each reconnection. The session password would also prevent session confusion if the connection between the service machine and a workstation were prematurely terminated, and a second workstation became substituted for the first. For security reasons, the session password should only be stored in volatile RAM memory.

The low-level protocol should be implemented on all workstations that communicate on unstructured asynchronous lines. (Some computers participating in the single system image, particularly hosts, might communicate using other, commercially available, reliable protocols.) The initial implementation would require protocol drivers on the network service machine and popular workstations.

One question of concern is network delay. That is, to what extent would naturally-occurring delays in the network cause a mail-based system to operate beyond the bounds of convenient user time? For our preliminary purposes here we might think of network response time of a few seconds for trivial actions (e.g. confirmation that a mail item has been routed to its destination locally) to a minute or so for responses to significant actions (e.g. return mail of a significant data base query). At this juncture we are not concerned with delays imparted by the software in handling the action, but communication network delays in actually passing the results.

Using a full duplex, asynchronous protocol for mail activities within the image (intelligent workstation to network server), there would be three major areas where delays might occur:

NETWORK—the effective communication speed (or bandwidth) of the underlying local area networks.

PROCESSOR—the effective processing speeds of the workstation and network servers.

DISK I/O—the effective data transport rates between CPU and disk on the workstations and network servers.

If the server and workstation were very fast (and they will be within the very near future), then it is clear that the network delay would be the major bottleneck for typical "trivial" queries. Network delay, of course, would vary with the interface to the network being utilized; speeds of up to 64 Kbs are possible on PBX systems, with much lower speeds on typical dial-up modems. However, even 64 Kbs will not be sufficient as large data queries and file transfers become commonplace. High-speed networking alternatives, such as Ethernet and Token Ring, supporting megabit speeds, will need to be investigated.

The ACCESSOR

The role of the ACCESSOR would be to provide a canonical language which permits and facilitates access to components of the image.[3] The canonical language need not actually be embodied in code, but rather would act as an interface standard which everyone would write to. The user, desiring elements from one or more data bases, would compose his or her request in the canonical language. The various data base administrators would provide "front ends" to their DBMSs which would translate canonical language requests to well formed calls in the language of the DBMS. This process would make the storage and conceptual views of each participating data base transparent and permit all data bases to be addressed with the same command structure.

---

[3] The word "neutral" is frequently used in place of "canonical," particularly in the context of IGES and MAP.

For the naive user, the interface to the canonical language might be mediated in several additional levels. For instance, an office automation system might call up Query-by-Example-type screens to provide an iconic interface to the canonical language. Similar types of "front-ends" will likely be developed by personal computer users for the more popular native environments such as Lotus, dBASE and BASIC. Minicomputer system administrators might well decide that similar facilities should be included in the operating environment for their systems.

The choice of canonical language should not be made without extensive study. In the image concept, DBMSs may come and go, but the canonical language used for access lives on forever. The canonical language might be selected from one of the popular fourth generation languages such as FOCUS or NOMAD. However, it is not clear that such products will ever garner sufficient market share to become *de facto* standards. It seem more likely that *de jure* standards will evolve from national standards sources. If we had to guess what might emerge, we would go with SQL.

If an existing definition such as SQL is chosen, it will still be necessary to consider non-standard extensions to provide for mail, archiving, printer access, etc. which are not part of the conceptual model language implementations.

The definition of user views falls to the purview of the data custodians. In the case of complex data bases such as those represented currently in most administrative systems, the custodian will need to define a set of data fields that is not necessarily a subset of those actually maintained in the data base. For instance, it may require a dozen fields and hundreds of computations to produce a normalized annual salary—the data element that the custodian deems most appropriate for query purposes.

In addition to defining the logical fields of the user view, the custodian will need to define a set of access rules—authorization to view all or a subset of the fields and all or a subset of the records. For many data bases this will present minimal problems as the entire data base will compose the user view and there will be no restrictions on fields or records. For the institution's administrative files this is clearly not the case. The "front-end" that prefaces access to this data base will need a great deal of thought. This general subject is surveyed more completely in regard to the ADMINISTRATOR.

For some data bases, particularly those administrative data bases of the institution, it should be possible to bypass the image altogether and access the data base directly with fourth generation products such as FOCUS or NOMAD. This may eventually be the access of choice for the functional offices and groups who are building MIS-like interfaces to one or more of these data bases.

The DIRECTOR

The DIRECTOR task can be envisioned as sharing responsibilities with the data dictionaries of the participating data bases. A user request, formulated in the canonical language, would first go to the DIRECTOR, who would have to parse the request to decide several issues:

- Is the request well formed syntactically?
- Are the target data bases referenced known to the Director?
- Are data paths and data formats defined for the target(s)?
- Is more than one target referenced in the request?
- Are there institutional administrative tasks that must be performed—first level security or authorization checks?
- Etcetera

The list is summarily terminated with an "etcetera" because these are typical of the issues that a system design would address. I will not attempt a design here, but rather will identify what seem to be the thorniest problems and their potential solutions. The goal will be to suggest that the design problems can be solved at reasonable cost, in reasonable time, with current technology.

To be maximally useful, the image should be able to handle a request that references more than one data source. The major problem here is to decide whether such requests would be serviced synchronously or asynchronously. If asynchronous responses are acceptable, the user would receive two or more responses to a request, one for each target accessed. The user would then have the responsibility of merging the responses to make a composite response. While this might be acceptable in many cases, it would prohibit cross data base queries where one data base must be first accessed to develop the access criteria for a second data base.

If the decision is for asynchronous treatment of composite queries, it should be seen as a short-term, temporary solution. More thought needs to be given to the frequency and necessity of sequential routing through multiple data bases.

Even if sequential routing is not required, the asynchronous arrival of responses to a composite request is still less than satisfactory—too much of the burden is allowed to rest with the user. A synchronous response to a composite request would require that the DIRECTOR assume many of the characteristics of a DBMS itself, in fact, to display some characteristics that are not currently found in DBMSs. The DIRECTOR would need to have a memory and a temporary storage, and be able to hold partial responses until all have cleared, at which time it would recompose a composite response to the user.

There is also the question of how much about the data base structures should be known to the DIRECTOR. Should the DIRECTOR have knowledge of the domains and/or fields of the participating data bases? If it did, it could do additional content checking of requests, over and above verifying targets. This would permit a reduction in network traffic, which would be exceptional in a distributed environment. It seems that a much "cleaner" set of interfaces would emerge from a design that does not attempt to do anticipatory content checking—one that checks content only when levels are reached where content is already defined.

It may be appropriate at this point to observe that not all the data sources participating in the image need be on campus, or even under the control of institutional employees. It is easy to imagine a pseudo data base that mimics some commercially available data base. The role of the pseudo data base is to be the "front-end" for the commercial data base. In such cases, the "front-end" would not be the responsibility of the data administrator for that data base, but some intermediary on campus who assumes this surrogate role.

The DIRECTOR would also need to be able to communicate with the ADMINISTRATOR when security or authorization checks are required at the image level. Such communication should probably be synchronous. If it is, the prospect of deadlocks would need to be contended with.

DIRECTOR services would also require some additions to the canonical language—at least in the sense of an error message methodology. A minimal set of error messages would include:

SYNTAX—the original (and any derivative) request returned with some indication of the point in the request where it became syntactically unparsable.

VALUE—the original (and any derivative) request returned with the offending name.

DOMAIN—the original (and any derivative) request returned with the name of the offending operator where data were not in the domain of the operation requested.

These messages should be constructed so that they could be received and operated upon by an automated process.

Handling update requests would place a number of additional burdens on the DIRECTOR, among them questions of recovery and what to do with composite updates when one of the data bases is not "up" (a form of the concurrency problem). It is certainly possible to update files and data bases asynchronously via electronic mail. Major difficulties might arise relative to edit checks and decisions as to whether to do partial updates when some of the data fail to pass an edit check. It is also likely the case that many updates would most naturally be expressed in a format other than that supported by the canonical language. This would require the development (or modification) of a significant amount of software to convert these more natural formats (e.g., reporting class grades directly from faculty) into the canonical form. The update problem may indicate the need for some shared file capability within the image functions.

The DIRECTOR would need some means of communicating with users to allow them to inform themselves of institutional services available in the image. There would also likely need to be some form of HELP services for syntax of the ACCESSOR. There is a distinctly difficult question about whether the DIRECTOR should know about fields that exist in the user views of the participating data bases. If it does, and is capable of responding to user requests about information at the field level which is available, it would likely also have to know about authorization rules for those fields. This might be seen as an ADMINISTRATOR question, but the real problem has to do with the distribution of authorization security between the image and the participating DBMSs. Further consideration of this question is provided in the section dealing with the ADMINISTRATOR.

The DIRECTOR is most easily envisioned as a single "machine" maintaining its own directly accessed storage devices (DASD). The DIRECTOR would need the support of gateway "machines" to facilitate access to other networks or to users attached to multi-access machines. It is likely that a single DIRECTOR "machine" would experience significant performance degradation when there were any substantial number of image users. It might well be appropriate to partition the DIRECTOR services functionally among a mail director, a print director and an archive director. This might have the unwelcome effect of requiring users to know to which of the directors a specific task should be directed. An alternate strategy might be to provide a shared file system among the directors.

The ADMINISTRATOR

The ADMINISTRATOR function presents the most difficulty in the implementation of the image. Authorization and security are presumed to be ADMINISTRATOR functions. There is an overriding question of network security in general which, while not directly an ADMINISTRATOR function, might best be discussed at this point.

The image would necessarily increase significantly the volume of traffic flowing through the institution's communication networks. If we ignore the need for physical and electromagnetic radiation security of the actual physical links (something we might not be able to do in the long run) there still exist the network security questions related to whether this message is actually from whom the packet envelope says it is, and whether its contents have been altered. These are non-trivial questions without obvious answers. At least initially, something such as a session password would provide a level of security commensurate with that currently available in most campus networks. In the longer term it would be necessary to add data encryption to the network.

The only authorization checking that should be done by the ADMINISTRATOR is at the global level. That is, if the DIRECTOR is limited to knowing only of the existence of data bases and nothing at a lower level (fields and domains), then it is reasonable to have the ADMINISTRATOR know only of authorization at that same level. For instance, is this user (or representative of a particular user class) permitted to know of the existence of a certain data base? If he is, then further requests for field-level information would be passed on to the front end of the requested data base; if not the request would be terminated with some appropriate response.[4]

Having moved nearly all the difficult questions from the domain of the image to the front ends of the participating data bases, it is probably appropriate to consider the development problems engendered.

Most of our ADMINISTRATOR discussion has taken place in the context of either DBMS front ends or data base director machines. It may not prove useful to continue to try to separate the administrative from directory functions. We will do so here on the principle that maximum segregation of function would make the image easier to maintain and enhance. Automated data base administration would serve two functions:

DICTIONARY—the ADMINISTRATOR should provide minimal data dictionary functions for those data server machines that otherwise would not have a data dictionary. For data servers with a formal DBMS it would need to point to the appropriate data dictionary, avoiding the duplication of that information.

AUTHORIZATION—it would need to handle authorization for those data server machines that did not have an authorization facility. Similarly, for data servers with a formal DBMS it would point to that facility.

[4] This might be grounds for adding another "error" message to our short list.

ISSUES FOR A KNOWLEDGE-BASED IMPLEMENTATION

Any sufficiently advanced technology is indistinguishable from magic.

Arthur Clarke
Profiles of the Future

Ultimately, the image evolves into a major knowledge-based system providing a standard specification of all systems participating in the single system image. It incorporates some of the functions of data administration, user services and network services into an "intelligent" communication interface.

As a strategy, the single system image provides a development prospectus that can be undertaken in very small increments by highly disparate groups. It has utility at any stage of its development; the more fully developed, the more utility. Further, it is transparent to the user, requiring the user to know only the ACCESSOR commands in addition to the manipulation requirements of his or her own native environment. Ultimately, it makes the entire computer and communications infrastructure transparent to the user, permitting data maintainers to be concerned primarily with the collection and storage of data and only secondarily with user access to it. Hardware, software and communications technology may be changed without impacting users. In this context, the image can be seen to be a transparent interface between the user's native environment and the data storage and manipulation environment of the target system (another user's mail files, a system printing device, an IMS data base, etc.).

There are a number of design issues that could be dealt with more or less independently, suggesting that the project could be fast-tracked. Among these issues are:

- Design of the low level, high efficiency, communications protocol
- Design of a standard electronic mail format
- Design of gateway "machines" for off-campus electronic mail
- Design of a "neutral" text formatting standard
- Design of a shared file archival service
- Design of a security system, perhaps including data encryption

Perhaps the greatest advantage of the single system image would be the necessity to publish formal documentation and interface standards. Perhaps the best way to disseminate the system standards would be via a bulletin board service. This would permit users to contribute their own interface code for use by others. The quality of such unsolicited code would likely be quite variable (witness the code for various implementations of Kermit maintained by Columbia University). It might be desirable and/or necessary to enforce some sort of quality control over user-submitted code but, even so, it would have the advantage of permitting widespread participation in the development of the image.

The TRANSLATOR

The long-term goal for the TRANSLATOR (See SSI Conceptual Model, Figure 2, page 13) should be to support the user's native computing environment by permitting the user to transmit and receive in data formats of that environment. Because there would likely be a very large number of native environments the TRANSLATOR will likely evolve into a knowledge-based system.

In the early phases of the image it should be possible to support several of the more broadly used data formats. For instance, several DOS-based environments (SuperCalc and dBASE) support data transport using Comma Separated Format (CSF) ASCII files. Many popular packages can generate CSF files from their internal storage schema and can read CSF files and reformat them into their internal storage schema. It would be useful to create TRANSLATOR functions to convert columnar data into CSF. Such functions should be relatively simple as all that is required is to quote strings and separate fields with commas. Headings and titles, particularly in spreadsheets, would significantly complicate the task. The inverse function of creating a columnar format from CSF would also be useful. One obvious use of such functions would be to put a "flat file" obtained from a data base query into a spreadsheet.

This type of translation is only adequate when the computed cell values of a spreadsheet or the data items from a data base are required. Such a translation would not preserve the structure of the data for data bases (e.g., the size and type of fields). Neither would it preserve the actual contents of cells that contain computational formulas. In the early phase of the image some relatively simple TRANSLATOR functions could be produced to aid in translation of the most popular spreadsheets and data bases. These functions would be based upon low-level code, would not be general, and would require maintenance as versions of the software changed.

There is an important issue relative to the location of the TRANSLATOR functions. They might either reside on a network service machine, or be distributed as workstation programs. Maintaining the functions on a server has the advantage of single-point maintenance and update with only one version for each translation. The major disadvantage would be the additional time required to transmit files to and from the server.

Ultimately, the image would require a knowledge-based implementation that would promulgate a "data description language." At the lowest level, the internal representation of primitive data elements would need to be described. The TRANSLATOR would need to know that a particular native environment spreadsheet stored floating-point data in 32-bit, excess-64, normalized format with a hidden leading mantissa bit. A particular description for strings might state that characters are stored in 8-bit bytes using 7-bit ASCII with high order bits cleared and a NULL delimiting the end of string. The TRANSLATOR would need to contain descriptions for most of the popular file storage schema.

The user would be required to describe input data and target formats. Given these descriptions and a file matching the input data description, the TRANSLATOR would produce the corresponding file in the target format. These descriptions would probably be registered with the image and could be changed by the user or data custodian by as simple a mechanism as changing their registration parameters. These parameters might even be changed from one session to another as the user invoked multiple native environments.

Using the primitive descriptions, aggregate data types could be described. For example, an N-element, floating-point vector could be described as the concatenation of N floating-point representations. A data base record could be described as a structure of primitive representations.

Queries might be returned as simple relations, or "flat files" with a standard format such as number of records, number of fields, field names, and the row major order string of the query content. The "data description language" represents yet another canonical or neutral language that protects users from needing to know anything other than the characteristics of their own native environment and the Accessor commands to communicate with other environments. It is likely that current research on knowledge-based systems and automatic program generators will shed more light on how this might best be accomplished. Synchronous access is a problem of significant dimensions without an obvious answer. The problem arises whenever compound queries require accessing one data base for pointers to another, and in relation to any form of update request. The confounding issue is the need to keep several processes concurrent and synchronized.

DBMSs devote a considerable amount of code to the synchronization issue in an attempt to avoid lockouts, to maintain data integrity and to provide a consistent set of user views. The same set of issues promoted from homogeneous to heterogeneous systems are all the more difficult. The difficult nature of the problem can be seen in so simply stated a problem as creating a relational join from two or more data sources managed by two or more DBMSs. I am not convinced that such problems are intractable, only extremely difficult.

The solution may be found in some form of shared file system within the image or intermediary (auxiliary) processors that permit processes to share the same variable (a data item for instance). There are models of such systems, but the ones I am aware of maintain homogeneity at least at the level of the operating system.[5]

High Performance Computing

It is not difficult to imagine that there are many researchers who could profitably use the power of a supercomputer but who have never considered the option or have rejected it believing the threshold to use one is too high. One approach to lowering the threshold for such potential users would involve making access to the supercomputer appear a simple extension of their native computing environment. Such an extension could be implemented via electronic mail combined with sufficient intelligent routines available at some supercomputer site or hub. As many potential users might not be FORTRAN programmers in their native computing environment; significant lowering of the access threshold would need to include a problem description "language" which did ·t involve mastery of FORTRAN or its nuances for any particular supercomputer system

Such a strategy would require attention to current candidate native computing environments as well as such environments that might develop in the future. For instance, we could imagine a spreadsheet user describing an extremely computationally-intensive mathematical programming or heuristic search problem in some spreadsheet format. The construction of an "interpreter" to restate the problem implicit in the spreadsheet format can be theoretically envisioned, but seems hardly worth the effort given the amount of additional information the user would need to supply in (probably) some format other than the spreadsheet. When extended to the other (perhaps hundreds) candidate native environments the task appears formidable indeed.

An alternative strategy would be to develop some form of canonical language that requires far less time and effort to master than supercomputer FORTRANs, which can itself reference computational data maintained in the user's native environment. Efforts designed to produce similar types of canonical language environments are exemplified by the work on the Initial Graphics Exchange Specification (IGES) and the Manufacturing Automation Protocol (MAP). The canonical language envisioned here, while serving a similar type of exchange function, is not directed toward data, but rather toward metadata and process—that is, toward descriptions of data and how those data are to be processed. It is also dissimilar in that the exchange moves only in one direction, from the user in his or her native computing environment toward the supercomputer that is the target processing system.

A canonical language would have an additional benefit in the avoidance of the user's need to know in advance the target supercomputer. In this way the nuances of problem decomposition to effectively use the form of parallelism or vectorization implemented in a particular system and compiler would be avoided. The user would provide a "generic" description of the problem to be solved and the canonical language would be translated to a "proper" set of FORTRAN code once the target machine was chosen. This would permit a departmental hub to provide value-added service on several (perhaps radically dissimilar) supercomputers. One can imagine several scenarios of least cost, least wait, etc. path algorithms used in selecting the target supercomputer.

The forms such a canonical language might take are many and varied. Several candidate approaches come immediately to mind.

- A natural language interpreter
- Adaptation of a highly parallel processing language such as APL
- A semi-graphical block flow language that described data transformations desired
- A FORTRAN-like pre-processor that invoked the names of generic FORTRAN subroutines

The enterprising linguist could probably add several more.

The discussion in this section has necessarily been tentative and not prescriptive. Implementation of the initial phase of the image might well change the nature of many of the issues referred to here. I am convinced that a knowledge-based system will ultimately arise in response to the difficulties implicit in a heterogeneous, distributed computing environment. Its exact form is far from obvious to me at this juncture. I have attempted to survey some of the longer-term issues only to give a flavor of where I see the single system heading.

[5] James H. Morris et al, "Andrew: A Distributed Personal Computing Environment," *Communications of the ACM*, March 1986, pp. 184-201, is an innovative example under development at Carnegie-Mellon University.

3
A NEW PARADIGM

We have found a strange footprint on the shores of the unknown. We have devised profound theories, one after another, to account for its origin. At last we have succeeded in constructing the creature that made the footprint. And lo, it is our own.

Arthur Eddington

As the paradigm for campus information systems turns from a mainframe-centered, Ptolemaic model to a user-centered, Copernican model, it will be necessary to redefine responsibilities.

As the strategy evolves, we should find the central computing operation less and less involved in the user's native environment and increasingly concerned with image-associated activities. This suggests that it should be withdrawing from a number of its current user service activities and moving into new areas. For instance, classes and consulting activities related to native user environments should be decreasing with emphasis being switched to development of the single system image. For the near future, terminal-based office automation will probably continue to be a popular native environment for casual users, but it should be supported as a departmental computing activity. Institutional consulting relative to personal computers should be moving in the direction of assistance in hardware and software selection and configuration. The institution should be phasing out of any involvement in operating personal computer laboratories as student-owned machines proliferate. This type of analysis could be extended almost indefinitely, but consideration of the following figure will suggest the general nature of the shift of activities that should be occurring at the institutional level and suggest how these changing responsibilities might develop.

Accompanying such a shift in emphasis should be a commensurate shift in staff expertise. Systems programming skills will become increasingly important, and user consulting skills less needed. At the same time, systems programming skills will be required for an even greater diversity of hardware and software configurations.

A similar analysis of skills within software development could be made. We can expect decreasing need for constructing DBMS procedures and screen interfaces and increased emphasis on the conceptual model of data supported within the institutional systems. Interfaces to the image in the form of TRANSLATOR and ADMINISTRATOR rules will become increasingly important. Software development organizations should be able to expend more effort on metadata issues and the development of meta-operations on data dispersed throughout the institutional data base.

Figure 3. Responsibility Matrix

	Institution	College	Department
Institutional	Image E-Mail Archives Printing Computer Network		
Departmental	Office Automation Supercomputing LANs Spec. Software Library Automation	Minicomputers CAD/CAM Laboratory Automation Faculty Consul	
Personal	Device Connection Maintenance Contract Site Licenses PC Demos	PC Networks PC Labs	Terminals Faculty PCs PC Software Student Instruction

In a similar vein, those organizations concerned with communications—voice, data, and video—will need to come together to develop a common planning framework. A set of individual foci on telephones, on data communications, and on video will not be found cost effective or sufficient. The current push for all digital networks, particularly ISDN and digital television, will help focus an interest in seeing the network physical medium as the same for all three technologies. The result is likely to be increased use of fiber optics and much higher bandwidths to accommodate a burgeoning set of high speed departmental LANS.

Institutional operation of a digital PBX or acquisition of similar services from the local common carrier will be seen as increasingly attractive. However, even with ISDN such systems will only address low bandwidth data traffic and will not address video issues. The rationalization of high speed (megabit) LANs will be one of the major problems facing institutional communications organizations.

Libraries face the most difficult challenge presented by full realization of the information age. Libraries have over a 2,000-year history that, with the exception of the invention of printing, has changed little. Electronic publishing presents a set of challenges that will be difficult for libraries to assimilate. The change agents are computers, communications technology, and a plethora of new storage media. The problems for libraries will be exacerbated by continuing legislative confusion over copyright law and the need for publishers to decide how they intend to deal with the information age themselves

However, it is the integration of the library's stock of information that represents the fundamental step for the single systems image. Clearly the one addition to the campus network required to make it truly ubiquitous is bibliographic and full text information from the campus library and national data sources. In the longer term, the library might well account for the vast bulk of traffic in the network and the major load on the image.

In one sense the library represents only a specific instance of another data base participating in the image. In another sense, it is a very special problem because of its size and relative isolation (at least at present) from most campus communication networks. It is not difficult to envision that, once the major communication problems are addressed by the image, the library will become the major focus of computing and communications interest.

The library is likely to be relatively unique on the basis of data types as well as volume. As the library becomes a repository for machine readable data, the text, graphics, and bit-map forms of data are certain to become significant components of the data types flowing through the image. Transmission of these data types will require significantly higher bandwidths than electronic mail or even data base queries.

An additional confounding factor will be the high use of library resources by off-campus and, frequently, non-university users. Many of these complications are obvious. One that is sure to present conceptual difficulties is the question of cost recovery. While it is entirely appropriate for the university to treat the image (and its associated computational and communication resources) as an operational overhead item, substantial usage by non-university community users will generate the need for cost recovery.

To this point, all questions of managerial accounting in the image have been ignored. As the strategy is implemented, it will be necessary to make a number of adjustments in both funding (amounts and distribution) and chargeback policies. A user acquiring a personal computer can expect to pay about $3,000 for his or her native computing environment—charges for access to the communication network (primarily some form of modem) and CPU resources are added to this. A user acquiring a terminal can expect to pay about $700, not including communication connectivity and CPU resources. Current pricing policies at most institutions favor users acquiring terminals in a rather significant way. Not only is the out-of-pocket expense several thousand dollars less, but CPU resources are generally available in effectively unlimited amounts at no cost to the user. Such a policy favors user actions that are exactly the opposite of what is generally desired.

It seems an appropriate time to reconsider the chargeback policy for computing resources. Dividing the issue into two categories, subsidized and cost-recovered, suggests that those services shown in the upper left box of Figure 3 should be subsidized, all others cost-recovered.

SUBSIDIZED—resources and services associated with the single system image and information center-like activities.

RECOVERED—resources and services provided (or facilities management costs) to support mainframe- or mini-based native environments.

For instance, users of mainframe-based systems should be expected to pay for services received. On the other hand, personal computer or College-owned minicomputer users should not have to pay for services provided by the image, although when the service results in some "hard" form of output, it might be reasonable to charge for the "pieces" produced. As a general principle, access to the communications network (beyond the local modem) is subsidized; general purpose computing resources are cost recovered. As with any rule, there will likely be compelling reasons for exceptions, but they should be clearly recognized as such. Putting principle into action, say for office automation users, this would mean that the cost to provide the service should be determined and the cost recovered from users on usage basis. If office automation services can be produced more cheaply (or the service level is significantly higher) via mainframe, then users desiring an office automation-like native environment will opt for mainframe-based service. If not, less expensive (or higher service level) approaches on personal computers will be chosen. A more equitable chargeback policy will allow computing services to be more nearly market driven, rather than subsidy directed as is presently the case.

With such a view, it would appear that current subsidy vs. recovery policies for most institutions are nearly reversed from where they should be. Development of the communication portion of the infrastructure should, as a general principle, be subsidized. Institutionally maintained native computing environments, as a general rule, should be cost recovered. One way of seeing the issue is to recognize that there is frequently more computing power purchased by individuals or acquired by grant, considerably less power provided as a subsidized commodity by the institution. This imbalance can be expected to grow significantly, even without changes in cost recovery policies.

An immediate tactical question should be a study of current chargeback policies with the intent of defining a policy more consistent with desired user behavior and real costs. A principal issue in such a policy study should be the distribution of chargebacks between subsidized and "real" dollars for the various services of the computing center, communications network, and systems development activities.

An ancillary question to be answered would involve the distribution of "real" dollars—between the computing center, communications network, systems development, and colleges and departments. While the historical record at some universities generates some misgivings regarding the placing of "computing" dollars in [illegible]t budgets with the ability to use them for travel, paper clips, etc., there is potentially some merit in allowing "computing" dollars to be spent for personal and/or regional computing services in lieu of global services. Such an arrangement would have to be well planned and slowly phased into, but might serve to much more accurately register user demand.

Our goal should be the creation of a single system image that provides the ubiquity and simplicity that we associate with the telephone system. Such an effort will require shifting many current burdens from the user to a knowledge-based network and rationalizing the way the user relates to the services provided.

CAUSE is a nonprofit professional association whose mission is to promote effective planning, management, development, and evaluation of computing and information technologies in colleges and universities, and to help individual member representatives develop as professionals in the field of information technology management in higher education. Incorporated in 1971, the association serves its membership of more than 740 campuses and 2,000 individuals from the CAUSE national headquarters at 737 Twenty-Ninth Street, Boulder, Colorado 80303. For further information phone (303) 449-4430 or send electronic mail to CAUSE@COLORADO.BITNET.