Andrew Is Alive and Well and Living in the Computer Network

by Matthew Maguire

Students preparing research papers endure the same painful and painstaking process that has tormented scholars for generations. They scribble notes on index cards. As the wad thickens, they shuffle and reshuffle their notes and thoughts, hoping that an outline will materialize. As they work, they suffer.

Soon the process may become less onerous at Carnegie-Mellon. Using an innovative, new computer program, students will keep computerized "cards,"

"Andrew" It named for university founder Andrew Carnegie and Andrew Mellon, U.S. secretary of the treasury in the "20s, ambassador to Britain, head of the Pittsburgh banking firm and Industrial magnate, who endowed the National Gailery of Art In Washington, D.C.

index them and write from them - on a large screen that lets them' simultaneously display and work with the cards, the index and the paper-in-progress. Suffering may subside as the drudgery diminishes.

This program is one of dozens designed for and made practical by the powerful, new personal computers in use on campus. These advanced function workstations and the network that will link them are major elements of Carnegie-Mel Ion's much ballyhooed effort to computerize itself as no institution has ever done.

That effort is nearing fruition. Countless headaches and headlines after the university unveiled its vision of a computer environment unprecedented in its breadth and ambition, some of the powerful personal computers have been deployed and cable for the new network is being laid. Not every deadline has been met. Not all lofty goals have yet been realized. Nonetheless, Carnegie-Mellon has already taken many steps towards achieving its dream of conceiving and creating a computer network which, when completed, may very well be peerless.

The faculty and administrators who conceived this vision and then began turning it into reality eschew lofty claims when describing the project and predicting its effects on education at Carnegie-Mellon. Many of them balk when their efforts are characterized as "a revolution in education" and say they are creating only the potential for revolution. Indeed, they say that, in the beginning at least, changes wrought by the advanced computing environment will most often resemble those made possible by the note-card software. People will do the same chores, practice the same research procedures and learn the same things. They'll just do so quicker and easier.

The possibility of networking the campus and putting powerful personal workstations onto'that network had been bandied about for some time before the university formally began contemplating the idea in the early 1980s. The campus at that time relied heavily on time-sharing systems for most of its computing. (With time-sharing, users gain access to a single powerful "mainframe" computer through remote terminals that have no computing power of their own.

This works fine until the number of users demanding computing power exceeds the capabilities of the computer. As the number of system users increases, so do chances that the system will falter.)

By the late 1970s, the limitations of time-sharing at Carnegie-Mellon became increasingly evident as the demand for computing time grew.

"It became apparent that the continued expanding usage of mainframes wasn't going to yield enough computing and that was not where the resources should be invested, that the field in general was moving toward ensuring control over one's own computing cycles, rather than competing for cycles," says John Stuckey, director of computing for the College of Humanities and Social Sciences.

Fortuitously, the university's Computer Science Department was already researching a prototype for an alternative. Computer scientists working on the SPICE project (Scientific Personal Integrated Computing Environments) were studying potential problems in such networked personal computing environments and developing hardware and software for the department's own local network.

With knowledge gained from this research in hand, the university decided to consider a network of personal computers on a campus-wide scale. University President Richard M. Cyert created a Task Force for the Future of Computing, chaired by Allen Newell (IA'57), University Professor in computer science. In his charge to the task force, Cyert asked the participants "to develop a comprehensive model for the role of computing at CMU during the decade of the '80s and guidelines for realizing that role."

The task force returned its report to Cyert early in 1982 after an exhaustive effort to solicit comments and criticisms from everyone on campus who would be affected by the network.

"The most remarkable thing about that task force," Newell recalls, "is that it focused entirely upon policy and general effect. It did not attend to any of the details. It devoted itself to the general character of the arrangement that would be on campus and general conditions that ought to hold to make it successful at the social and organizational level, not at the hardware level."

The vision first articulated by the task

James Morris (S'63), director of the Information Technology Center, some 35 Carnegie-Mellon and IBM scientists and technologists who developed the "Andrew" system. The system Is owned by IBM.

force was refined into a plan that was grand and ambitious:

- The personal workstations on the network would have computer power and graphics capabilities well beyond those of the best personal computers of the day.
- Technological advances would drive down the price of personal workstations of this caliber — from around $30,000 in 1982 to around $3,000 by this year.
- Cable for the new network would be installed in every room on campus, and all students would have a machine by 1986. With some 7,000 of the advanced function workstations on campus — one for every student, faculty member and administrator — they would be literally ubiquitous.
- The design of the whole system would emphasize flexibility so that any vendor's workstation that met performance specifications could function on the network. And as a result of insistence on flexibility, dramati provements in computing and i. working technologies could easily added to the system, instead rendering it obsolete

The university began pursuing dream in earnest in October 1982 signing a $35 million contract with II to create the Information Technolo Center (ITC). ITC's charter is to devel' a software system - based on it: hardware but portable to other bardwa - that will support the university's p\- to integrate computing into its educ tional program.

That mission, says James Moi (S'63), director of the ITC. represer. "a rather daunting technical challen» A system like this hasn't worked bcfo anywhere, at least not this big. The sea we're aiming at is unprecedented."

Not all elements of that vision will \ realized this year. The project has ri' escaped the unforeseeable setback delays and changes that accompany sue. ambitious research-and-development ei forts. Parts of the plan will fall into plac late. Others will be ready on time b won't be perfect and will require refini ment and expansion.

For example, the sudden and histori explosion of affordable personal corr. puting, made possible by the success r the Apple Macintosh and the IBM Pet sonal Computer, forced a major — am difficult — change in the project. Thi personal computer revolution helper ease the burden on the university's time sharing systems and brought personal computing to most students on campus. But it also complicated the computerization project because it introduced an intermediate technology that soon pervaded the campus — and forced the ITC to adapt the plan so that these smaller machines could plug into the network along with the advanced function workstations.

But if short-term deadlines and goals have changed, the vision has not, according to William Arms, Carnegie-Mellon's vice president for computing and information services. Arms and John P. Crecine, senior vice president for academic affairs, are overseeing completion of the project.

"I think the most important statement we can make about the project is that the vision which the Task Force on Computing enunciated back in 1982 is the vision we're still working for," Arms says. "What President Cyert was after and what the task force was after and what we're alter now are the same. In fact, I'll go further: What we're aiming lor now is probably more ambitious than wh:il they had in mind."

lor one iliing, (lie overseers ol the quest learned early that they needed to expand **their** dream beyond ihe confines of **the** campus to include similar projects at other pioneering universities.

"When the project began, we weren't sure if we were competing with other

John P. Crecine (IM'61, IA'63,'66), Carnegie-Mellon senior vice president for academic affairs.

universities creating networks or were in partnership with them," Arms says. "So there was a reluctance to share contractual or other information with, say, Brown or MIT. We didn't know if a dollar we got was a dollar they didn't get. But we [the universities] are realizing how stupid it is to see if we can all invent our own wheels."

(To foster this cooperation among universities, Carnegie-Mellon founded the Inter-University Consortium for Educational Computing (ICEC) with support from the Carnegie Corp. Besides Carnegie-Mellon, ICEC members include Berkeley, Brown, Dartmouth, MIT, Michigan and Stanford. Crecine is chairman of ICEC's governing board.)

It also became apparent early in the venture that the grand experiment had to include other hardware vendors that wanted to participate.

"We feel that the kind of powerful personal workstation being developed has the potential to revolutionize the way education is being delivered in higher education — which hasn't happened since the printing press and the book came into being," Crecine says. "The key to any such revolution would be the software that delivers the educational applications. Like any other teaching material — textbooks, for example — it must be widely available, in great variety, and portable from one discipline to another and from one university to another.

"No computerized university will be able to use only its own software, just as no university today can teach only with textbooks written by its own faculty," Crecine adds. "Because universities will inevitably buy hardware from different vendors, the educational software will have to be portable from one brand of computer to another. Only with the same portability that textbooks have from one university to the next can software revolutionize education as the textbook did."

The backbone of Carnegie-Mellon's effort to meet this challenge is Andrew — a simple name for a complex series of interrelated components which constitute, in the computer professional's jargon, an integrated personal computing environment. Andrew (so dubbed in honor of Andrew Carnegie and Andrew Mellon) is being continually improved, but it has already attracted attention in computer science circles.

Andrew was designed to run on IBM hardware as well as other vendors' machines — a planned software "portability" which is a major part of the so-called "multi-vendor strategy." Andrew comprises six components: a communications network, a distributed file system, an operating system, a user environment, hardware and an applications programs. Some of these were developed at Carnegie-Mellon; others were developed elsewhere and modified for Andrew.

The Communications Network

"The idea is to break down physical barriers, in the broadest sense," Arms says. The network includes not only the physical wires in the walls but also the services it carries: the file servers (the computers which run the distributed file system and provide remote storage of files), the ability to connect cleanly with different versions of the advanced workstations, and the ability to send data to other machines on the network and, via other networks, to the outside world. The cable that constitutes the physical part of the network should be laid by the end of the year, with connection outlets in every office, dorm room and classroom on campus. Connections from homes off campus are also possible, although at first they won't be as elegant as connections on campus, Arms says.

The Distributed File System

The distributed file system, called

William I. Arms, Carnegie-Mellon vice president for computing and Information services.

Vice, is a university-supported, centralized system of computers for storing data outside the user's personal workstation. The data is accessible from any workstation on the network.

Vice, says Arms, is an important step toward a long-held dream: the paperless campus.

"The ultimate objective is the day when I stop using paper for anything I do, but I merely put data on my machine. It is stored someplace in the university that I don't know and don't care about. Wherever I go in the university, I can switch on any machine and that data is at my fingertips," Arms says. "Moreover, the data I used 10 years ago and the data that you wish to transfer to me is also available.

"Vice is one of the definitive achievements of the Information Technology Center," Arms continues. "In many ways, it is the fundamental building block of computing at Carnegie-Mellon over the next five years. I think it's even

Andrew

Edward E. Uicente (E'60), president, Information **Products** Division, **IBM, who spoke at the** formal Introduction **of "Andrew," Jan. 21 at** the university.

more important than the workstations. It's less visible. You're not conscious of it. But it's the basic underpinning. It's like roads or sewers are to a city. The file system and the network are the infrastructure that makes everything else possible."

Vice is continually being improved. This year's version is less reliable, responsive and functional than the ultimate system. "New products are like that," Arms notes. But the kinks should be hammered out in another six to 12 months.

The Operating System

A computer comprises hardware and software. The hardware is the actual machine and its computing power. Software is the set of instructions that tells the machine what to do with its power. Software includes two components. One is the applications programs. The other is called the operating system- i is that part of the software which links the hardware to the applications-oriented part of the software. The operating system handles "housekeeping" chores — moving data around within the computer, telecommunicating it outside the computer, managing what the user sees on the screen, and so on.

Andrew is based on a commonly used version of AT&T's Unix operating system. Unix has several advantages. It was developed in and is tailored for a research environment. It is widely available, which makes possible the flexibility of multiple vendors. And it is portable from one brand of hardware to another, which further ensures long-term flexibility.

The User Environment

What the user sees when he or she turns on the machine is the most visible part of computing. In Andrew, the user environment is good and will be made even better, says Arms.

"The polishing process is a continuing one," he says. "If you compare this user interface with a traditional time-shared system or, say, an IBM Personal Computer, our user environment is clearly superior already. But if you look at an Apple Macintosh, you can see it doesn't yet have the Apple's polish." Users who are accustomed to the extraordinarily slick user interface of the Apple machine may be slightly disappointed by the initial version of the interface. Improvements will be made in subsequent versions, Arms says.

The Hardware

Companies such as Apollo and Sun have turned out advanced function workstations designed for engineers for several years. Carnegie-Mellon recognized that such machines could be of immense value to students and researchers in all disciplines.

This type of workstation has been dubbed the "3-M Machine" because it has at least a million each of three critical elements:

- It has several million bytes of memory (roughly twice that in the current IBM Personal Computer).
- It executes at least a million instructions per second (roughly four times the speed of the current IBM PC).
- It has a million pixels in its display (roughly six times the number of display elements in a PC). This means the screen is much larger than the screen on a conventional PC and can be sub-divided into several "windows" for different documents or tasks and can show much more sophisticated graphic displays.

The larger screen and its better display are essential to educational applications. With the ability to divide the large screen into several smaller windows and simultaneously run different programs in these windows, the screen becomes more than just a substitute for a sheet of paper; it is an electronic blackboard.

Nerds are not the only students attracted to "Andrew." Punks, painting, playwrltlng students also find the system useful.

"Most educational applications have to provide a lot of information to the user," says Jill Larkin, an associate professor of psychology and director of the Center for the Design of Educational Computing (CDEC). Faculty who write educational software for the smaller screens of personal computers "spend an enormous amount of time trying to decide what information to leave out that won't leave the user hopelessly at sea." The larger screen finesses this issue and saves valuable time for the faculty programmer.

Because the university has adopted its multi-vendor strategy, the network is designed to support almost any computer. And Andrew has the potential to run on a variety of manufacturers' products. By the end of 1985, there were already two: a Sun computer and the Microvax by Digital Equipment Corp.

An IBM computer that will run Andrew - IBM RT PC - was announced in January.

The price of this class of machine is falling rapidly. By the end of 1985, they were costing Carnegie-Mellon around $10,000. Arms said he expects the price to drop (o around $5,000 by fall, which will enable the university to make them widely available to some individuals and to place more than 100 of the workstations in public clusters. By the fall of 1987, the price should be around $3,000. When that happens, Arms says, the university will encourage students and faculty to buy their own workstations.

The Applications Programs

Applications programs are the software that enable the computer to conduct specific educational tasks. The note-card program, developed by David Kaufer and colleagues in the English Department, is an example. Students in all disciplines will use these programs. Art majors will paint; engineering students will create and study dynamic systems that evolve in time; history students will analyze American demographic data; physics students will solve differential equations graphically; and chemistry students will work out complex chemical calculations.

This element of the system will evolve forever, Arms says.

"The biggest long-term challenge is to build up a base of really good applications programs," he says. For one thing, Andrew must be refined so that software developed elsewhere will run on it. At present there is no single system which will run all the leading applications programs. . . . The best drawing programs, MacPaint and MacDraw, run only on Apple Macintoshes. Getting programs of this quality to run under Andrew is the sort of problem that is difficult to solve."

One way to build this base of applications programs is to encourage faculty members to develop their own. Therein lies a challenge. Andrew is a system with vast power, but as it stands now "it's a real hacker's environment — which means it's not an amateur's environment by any means," says CDEC's Larkin. That challenge has already been addressed through development of CMU-Tutor — a language that "sits on top of the Andrew environment and makes it dramatically easier for non-experts to write educational software. CMU-Tutor was implemented by Bruce Sherwood, associate director of both CDEC and ITC, and Judith Sherwood, director of educational computing for the Mellon College of Science. The effort to tame Andrew's power will be led by CDEC, which helps software developers from all disciplines share information, learn new design techniques and demonstrate their work to the university community.

Even as faculty programmers begin developing software to exploit Andrew's potential, precisely how the network will affect life and education at Carnegie-Mellon remains unclear.

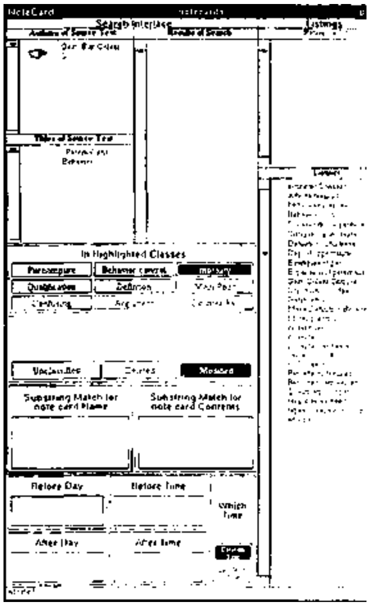
"The basic goal of the effort," says Arms, "is to design a computing set-up which will give a tool to individuals — students, faculty, administrators — which will make their time at Carnegie-Mellon more productive. We're particularly interested in improving the quality of the life that we give to our students and helping them learn — in the broadest sense of the word." Initially, that means development of the straightforward tools that will cut the drudgery associated with all facets of academic life, creation of tools for teaching, and facilitation of administrative computing.

But beyond those general goals, no one will predict exactly what this means.

"The truth is that we don't know how this will be used," says Howard Wactlar, vice provost for research computing and a member of the 1982 computing task force. He likens the effort to the introduction of telephones. Everyone recognized the potential of the omnipresent utility, but no one could predict exactly how it would be used.

"Too much of what people heard when they heard about us going to all the computers is that it's a new form of education, that we're going to be teaching everything in a computer-based way. That's not the case at all," Wactlar says. "What we're doing is delivering a technology to the users, making computing ubiquitous, so you don't have to stand in line to use it."

If the full potential of the new utility is exploited and the potential for a true revolution is realized, it will be a "bottom-up" revolution, says Crecine. "It will be led by individual faculty and students who, in independent ways, learn how to exploit the new technology for their own educational and research programs. The really exciting things will be done by individuals driven by their own academic goals and creative energies."



Andrew can provide users with a screen that will divide Into as many as 16 windows. This gives students an opportunity to have a term paper In one window, note cards and other materials In additional windows. Andrew's windows are unusual because they do not overlap and are consistent In where they will appear on the screen.

Matthew Maguire is account executive for science and technology in Carnegie-Mel Ion's Public Relations Department.