Title
Ideas for the Arcturus personal workstation

Author
Standish, Thomas A.

Publication Date
1981-03-23

Peer reviewed

Ideas for the Arcturus Personal Workstation

OK

Thomas A. Standish

Irvine
Programming Environment Research Center

Technical Report 172

Computer Science Department
University of California
Irvine, California 92717

March 23, 1981

A b s t r a c t

In order to achieve more effective use of computers for interpersonal collaboration, management of software lifecycle activities, and dynamic retrieval of information, as well as for ordinary programming chores, it may be useful to take a fresh look at the devices we use to help us conduct our transactions with computers.

This paper presents some new ideas for using large flatscreen displays and dedicated computers as personal workstations.

Table of Contents

I n t r o d u c t i o n

In programming environments of the future, the name of the game is to improve both software quality and programmer productivity simultaneously.

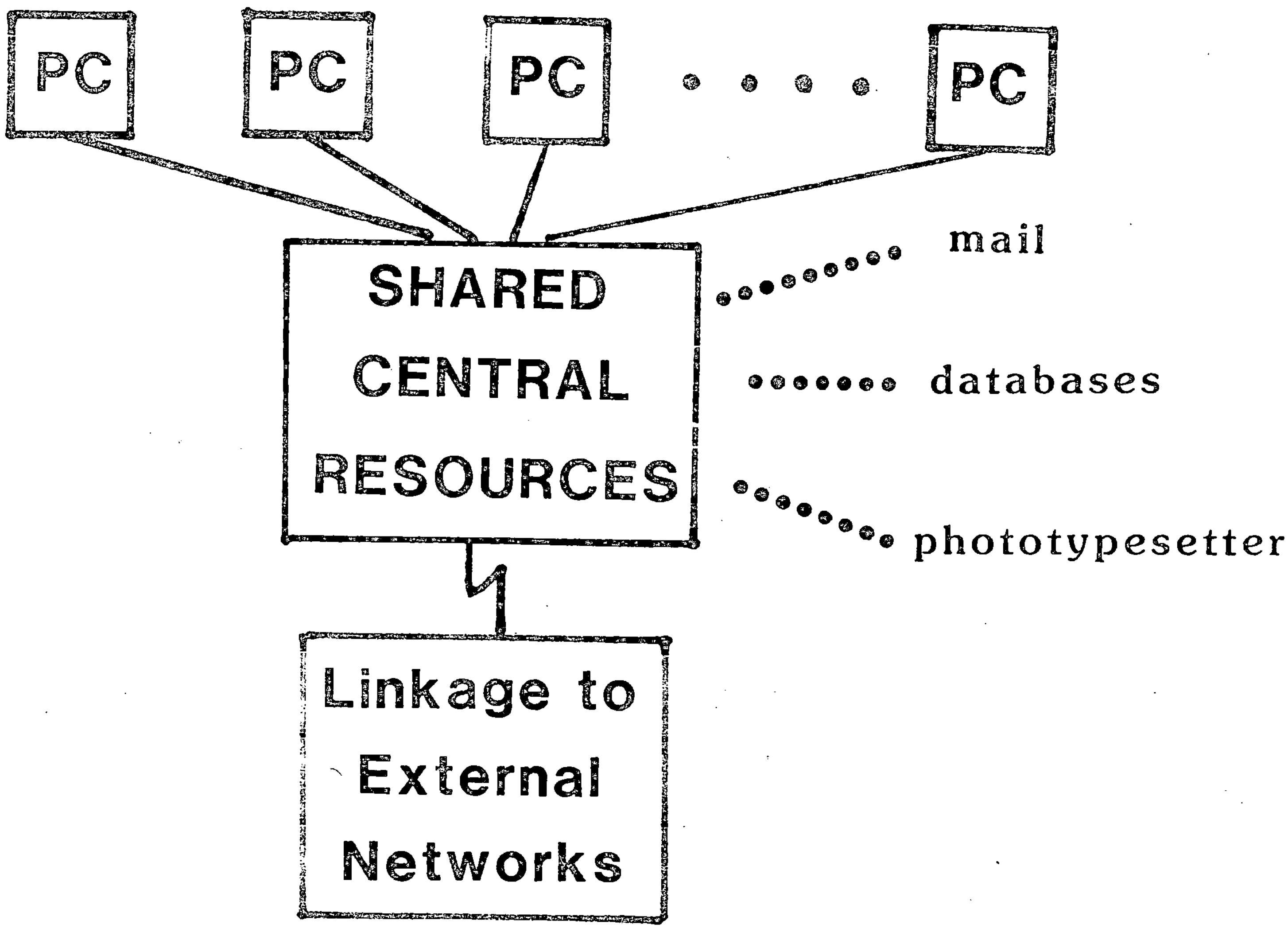
While many kinds of improvements in programming environments may influence quality and productivity, one of the possible areas of improvement we may wish to examine is the effectiveness of the devices we use to conduct our transactions with the computer.

Of course, we use computers not only to perform programming chores, we also use them to communicate with other people, to collaborate with teammates on joint projects, and to manage software activities.

From a broad perspective then, we should view computers as general communication media, as vehicles that promote effective collaboration between project personnel, as devices to help us (both individually and collectively) to schedule and organize work and to monitor progress and adjust effort incrementally, and as information retrieval devices giving us dynamic access to information.

Figure 1 portrays our view of the computing arrangements that appear to be emerging for the mid-1980s. Here, the declining cost of processors, coupled with the generally non-declining cost of peripherals creates economic

Computing Arrangements of the Mid-80's



Arcturus

FIGURE 1

incentives that are driving computing arrangements into a configuration in which each programmer (or manager or librarian or whatever) will have a <u>dedicated personal computer</u> linked to a central facility providing access to expensive shared peripherals, archival databases, inter-personal communication services, and (perhaps) access to external networks.

The advantage of a dedicated personal computer is that it is yours alone to command, and because of that, it delivers a dependable, predictable level of service that is not degraded by shared usage. Computations too big for a dedicated machine may still have to be exported (as in large number-crunching applications), but most tasks would be done locally.

Styles of Interaction

Figure 2 portrays abstractly a set of observations relating granularity and nesting of programming activities as a function of terminal type.

In batch environments, where, say, we have a remote job entry terminal for queuing a job request, or where we use off-line preparation of a job specification, we tend to invoke tool activities sequentially and in large grain sizes. For example, we may first call a text editor to create or change the entire text of a program, then call a

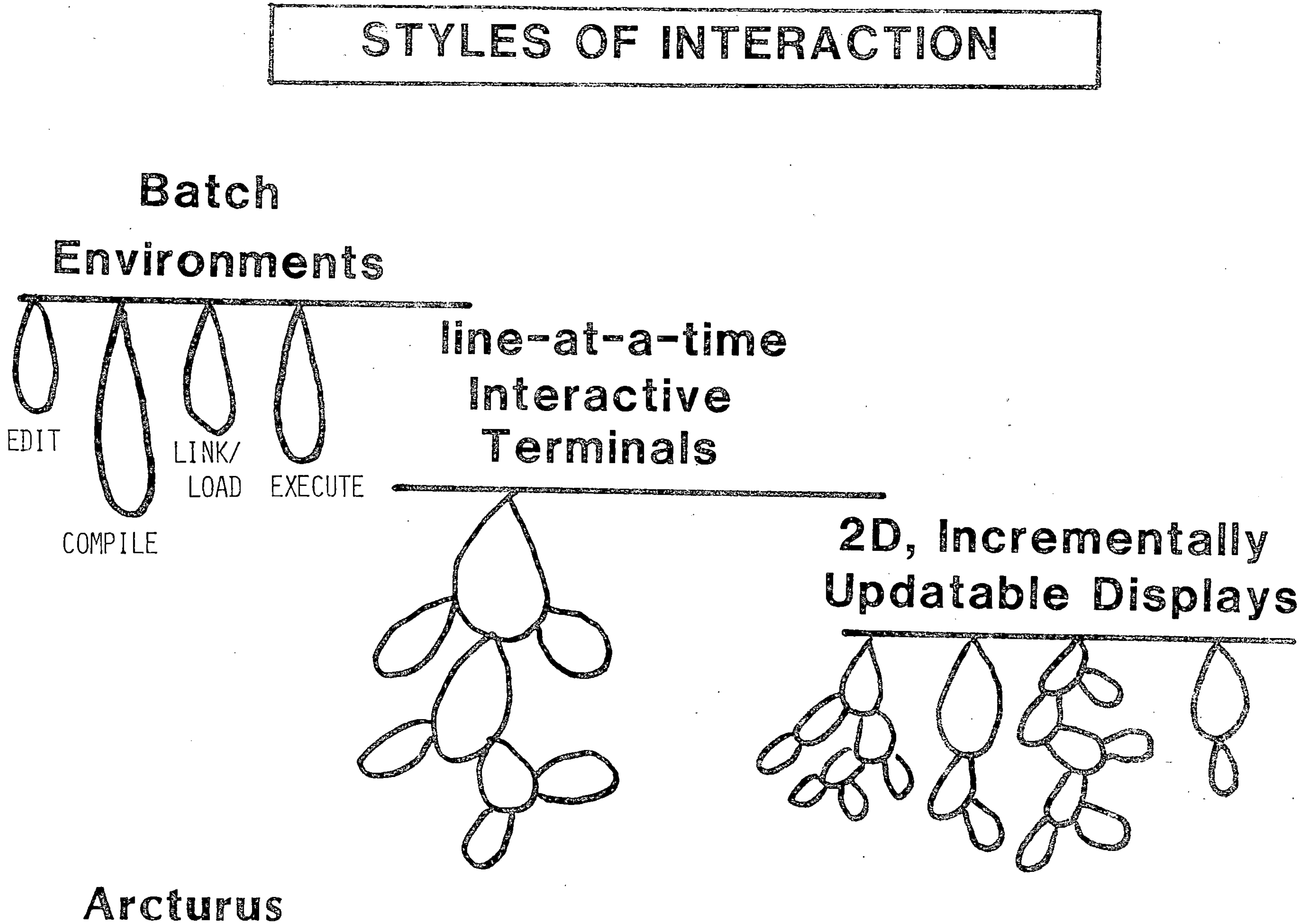

STYLES OF INTERACTION
Batch
Environments
EDIT
COMPILE
LINK/
LOAD
EXECUTE
line-at-a-time
Interactive
Terminals
2D, Incrementally
Updatable Displays
Arcturus

FIGURE 2

compiler to translate the entire program text into relocatable code, then link/load, and finally execute. If we want to change a single semi-colon in the program, we must iterate the entire cycle. The processors communicate with each other by depositing large granule intermediate results in a database. The tools do not call each other in a nested fashion.

If we have a line-at-a-time interactive terminal, the style of interaction with the machine often changes to one that proceeds in small grain sizes and in which the tools are frequently nestable. For instance, we may call an interactive language processor (such as APL or Lisp). We might then use a text editor within APL or Lisp to define the text of several functions. Then we might call a function, which might halt upon reaching an error. At the point of suspension, we may evaluate expressions, inspect the contents of variables, incrementally edit text, and even send and receive mail or call the system help facility for information. At the conclusion of a nested activity, we may be able to resume the background activity that was immediately interrupted beforehand. Thus the tool activities can be nested and our requests for service are incremental and frequently small. The response time tends to be proportional to the size of the change requested (unlike the situation in batch systems).

When we have a two-dimensional, incrementally updatable display, in which we can redraw the contents of a portion of the screen without having to redraw the whole screen, yet another style of interaction becomes available to us and has received increased usage. We are able to manage more than one nest of tool activities simultaneously. Some nests may be suspended and awaiting our subsequent attention and direction. We may be pursuing a current focus of attention actively. At the same time, some background activities may be executing but we are not attending to them. For example, we may have a text editing activity currently suspended with a partially completed text file open. We may have an active background process monitoring for the arrival of mail, with instructions to notify us of any change of status. And we may be currently interacting with a program that is executing.

In this style of interaction, a given nest of activities may be associated with a window on the screen, and there may be an inexpensive way to command the terminal to track the change in our attention from its current active focus to some new focus. For instance, we may be able to change the position of a cursor from one window to another by moving a mouse and by depressing a button to signify to the computer that we have selected a new active focus for interaction. Menuing, windowing, and zooming also help us choose and adjust tool activities.

The two-dimensional, incrementally updatable display thus allows us to have a number of simultaneous activities underway, and allows us to shift our attention between them --- much as we can do with our desk-tops, when we have, say, a manual open to a particular page, a half-written program in front of us, a phone to interrupt us to bring incoming conversations, and various information displays, such as calendars and task lists on our walls.

However, current two-dimensional, incrementally-updatable displays have one serious drawback --- they are too small! This forces us to do all of our work as if we were manipulating many small slips of paper inside a shoe-box, and we spend too much time moving these slips of paper around inside the shoe-box and enlarging the ones we are interested in transacting with at the moment. If we had more space in which to spread things around, as we do on our desk-tops and walls, we could acquire the freedom we are used to having in the physical media that surround us. Thus, the shoe-box isn't good enough.

Properties of Desktops, Walls, and Paper worth Preserving

Let's think about our normal workplaces and make some observations. What do we see? We see walls covered with useful items such as calendars, lists of phone numbers, notices of upcoming events, reminders, plans and schedules, and other useful information that usually changes on a not-too-volatile basis. (We may also see decorations designed to increase the aesthetic pleasure of the workplace.) There may also be writing surfaces (blackboards, flip charts) that serve as collaborative communication media, designed for volatile change and/or short term storage of pictures and information.

Our desk-tops are used for more highly volatile activities where we actively manipulate the information context, but we typically spread out our working materials in space and alternate our attention (look up a reference in a reference volume, incrementally add new text or drawings to a document, cut and paste pieces of previously composed material, etc.). Since we have lived in three-space all our lives, we have highly developed capacities for storing inert objects in the space around us, for remembering where we left a suspended activity in the local geometry, and for alternating our attention between active and suspended activities by directing our gaze and using our advanced pattern-recognition capacities.

Our desk-tops begin to become ineffectual when we overload the space they provide with too much material, and when we begin serially stacking contexts in piles which we have to search by inefficient retrieval methods (such as serial enumeration). When this happens, there are incentives to utilize indexed collections of information --- labeled file folders, alphabetized lists (of, say, addresses and phone numbers, etc.), so we can retrieve desired information rapidly.

Properties of Books and Paper Documents

Books are really quite remarkable as information containers. In addition to being cheap, portable, and never down (properties enjoyed by few computers), books are often indexed doubly (by a table of contents and by an alphabetical index of contained items akin to a partially inverted file). We can browse through books, turning the pages rapidly in search of some remembered item of information that may be easily recognizable pictorially (as in searching for "that histogram of military spending versus industrial productivity increase rates" somewhere near the end of Chapter 3). We can also do interpolation searches if items in a book are presented in lexicographic order (e.g. section numbers, alphabetized headers, or the like).

If we could combine these properties of books with capabilities computers perform well, such as string searching, we might gain some leverage.

Paper documents also have properties worth attempting to duplicate in our Arcturus Personal Workstation. When we write papers, we frequently take drafts, scribble on them, mark them for insertion, deletion, and text motion, and trade them with collaborators.

Using inking (a technique in which we sample the location of the pen point, say, every ten milliseconds, and in which we deposit a dot on the display in the sampled position), we can imitate electronically the effect of scribbling on paper. We can also use hand-written character recognizers to convert from handwritten input to printed input. If we add the capabilities of "cut-and-paste" text editors, we can begin to approximate the convenience of real paper with electronic paper. (Portability and the property that paper has of always being "up" may not be so easy to duplicate, however.)

Properties Beyond Reach

We can't duplicate every quality of the physical world in our electronic media cheaply. When people interact, they do so in three-dimensions, in color, and with voice. While it may be possible to add voice cheaply to an electronic medium that manipulates electronic paper, we can't duplicate real-time, 3D, and color cheaply. This probably means we will have to do without the richness of interpersonal meetings (to the extent that gestures, facial expressions, and prosodics contribute to communication). Thus, what we propose will probably not be a complete replacement for actual interpersonal communication (and we are not recommending it as a device by which we could "compute to work" instead of "commute to work").

The Personal Workstation

The Personal Workstation of our current dreams consists of a system based on a powerful, cheap desktop computer and cheap flatscreen displays. We hope large, cheap flatscreen displays will soon make their appearance and that they will come in large rolls. Given this assumption, we would like to unroll some and cover a large desktop, and then unroll more and fasten it to a large area of a wall, say, the size of a blackboard. An independent portable keyboard and a portable stylus would connect (say, via radio) to the system.

We would like the flatscreen display to have the quality of liquid crystal --- that is, we want printed text, drawings, and figures to appear in it as if a black printed transparency had appeared inside a sheet of plastic or a pane of glass. We want the resolution to be sufficiently fine-grained to support high-quality pictures, line-drawings, and type-fonts of many sizes and styles. We want the flatscreen display to have independently switchable (x,y) coordinates so that subpictures and contents of windows can be incrementally changed, and we want the switching speed to be sufficiently fast that a page of text or picture can appear in a few hundred milliseconds.

the ARCTURUS Personal Workstation

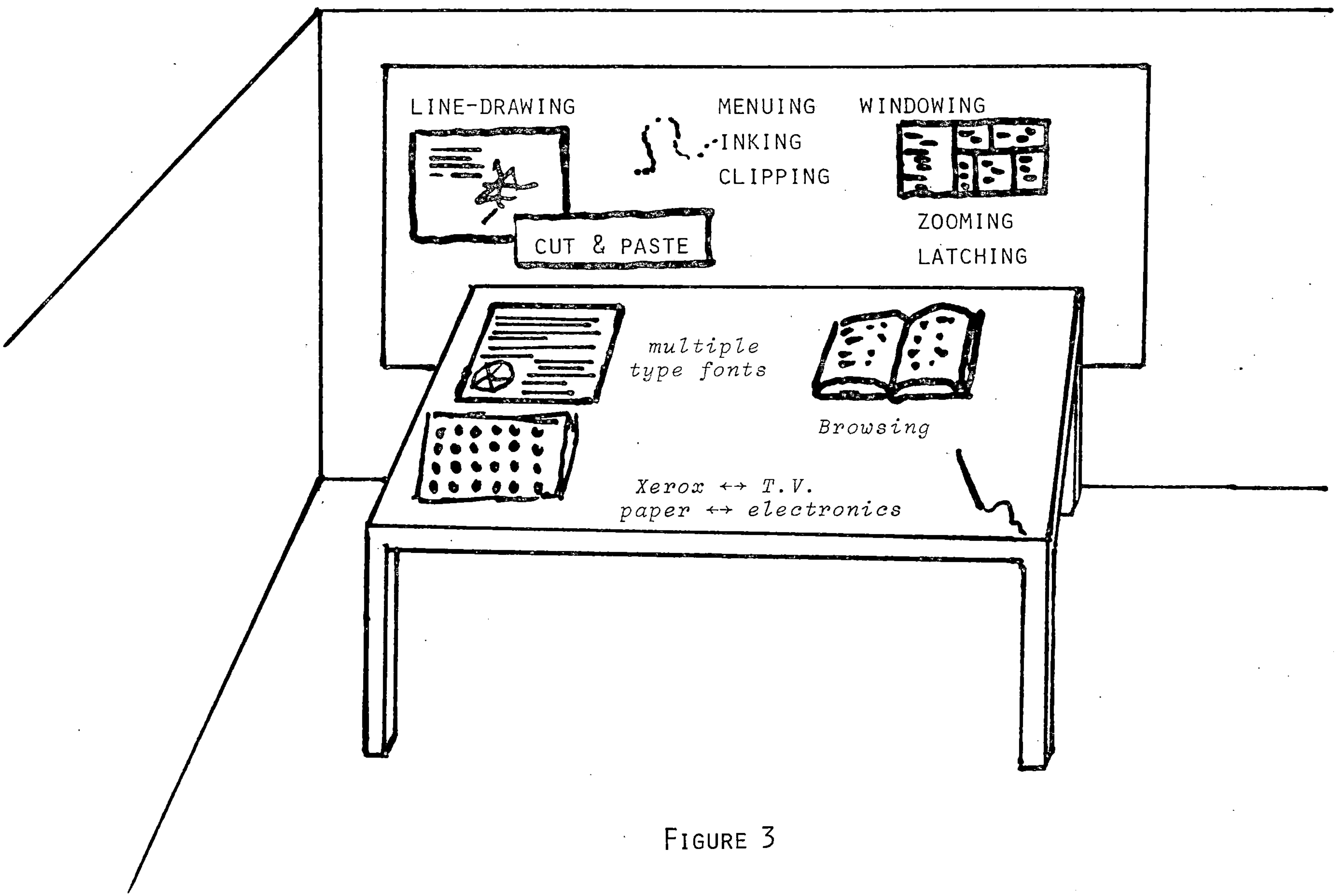


FIGURE 3

The technology for driving such flatscreen displays has been known for some time, and involves windowing, clipping, inking, menuing, latching, rubber-banding, and computation of simultaneous views in "panes" of a window. This technology has been developed progressively on the Lincoln Labs TX-2, on the Harvard PDP-1, on many Evans & Sutherland systems, and recently on the Xerox PARC Altos used in, e.g. Smalltalk and Interlisp. Cut-and-paste text editors, animation sequences (producing dynamic books), and a uniform command language using menuing, windowing, and simultaneous contexts (one of which is usually active and the others suspended), are well-known techniques of proven effectiveness for using such a medium.

As we noted in the introduction, we feel that the current technology, which uses a small display, is a disadvantage, causing users to operate as if they were manipulating many small slips of paper in stacks inside a shoe-box. Large flatscreen displays on desktops and walls should cure this ailment, making it possible to use normal size pieces of "electronic paper" and to have many of them displayed simultaneously on a desktop or wall, more like the properties of real paper.

In our dream workstation, we would like to see a graceful marriage of the worlds of paper and electronics. A very high resolution TV camera could shoot pictures of printed text, drawings, (or anything, for that matter), yielding images that could be shipped electronically. In addition, we envisage character recognizers that could process such TV images of text, producing descriptions of the characters and their fonts in "pre-runoff" form, suitable for text editing. We also could cut and paste pieces of electronic images resulting from TV capture of real paper images for inclusion in electronic documents. We would be able to use inking to include handwritten remarks in electronic documents to be shipped over computer networks. To go from the world of electronics back to the world of paper, we envisage good-quality, two-sided Xerographic printers (such as the Penguin variety of printer).

Given electronic books and manuals, we envisage programming the workstation not only to search via normal text editor search, but also to simulate browsing, via analog controls, starting at interpolated search points (as in searching for the name "Vickers" in a telephone book starting "somewhere near the end").

When we latch onto a piece of electronic paper lying inside our desktop and drag it along with a stylus, we envisage drawing a blinking frame along with the stylus as the stylus moves. When the stylus is depressed to unlatch and deposit the paper, the paper would get redrawn at the point of deposit as an overlay on top of whatever else was there below it.

It may be possible to program the workstation to do fancy things such as placing the text of a book on an imaginary football field somewhere to the left of the Moon, and flying out to view it at Warp 6, applying a spelling corrector to the football field, and having lights turn on everywhere there is an error (to get some portrayal of the error density, viewed from afar), and then zooming on selected areas for further text manipulation by normal means. However, we are not convinced that rapid, real-time, color displays (as in the E&S flight simulators) are necessary for our task. Real-time, color, simulated motion with peripheral detail, is probably unnecessary. We include this remark in the discussion to illustrate the idea that we are not being as grandiose as we could imagine, and that we are not playing the game of, "whatever you have, I'll imagine something more general."

The inclusion of color is a point of debate. Some, who are in a position to know (such as S.L. Squires), argue forcefully that color adds an important orthogonal dimension of freedom of expression that is too important to be ignored.

An important consideration for us, which we do not see strongly reflected in present work (though perhaps we are in ignorance), is the availability of prefabricated means for smoothly extending the user-interface language, using windowing, menuing, simultaneous computed views in panes of a window, command completion, and the like. We think it should be made easy to add a new class of menuing and display commands in exactly the same style that the original system supports. This encourages users to add tools whose command languages obey the same conventions as the original tools in the system --- providing a strong incentive to keep the user interface uniform in user created tools.

Alternative Technologies

What happens if the large flatscreen displays don't materialize because the supporting technology does not emerge (at an affordable cost)?

Currently, there are no prospects for large flatscreen displays that come in rolls and have high-speed, LCD, incrementally switchable quality. However, there are small tiles with this property under development in research laboratories and (it is rumored that) Toshiba has developed a television set with a small display with the properties we envisage.

But what if we cannot compose the tiles into large cheap displays?

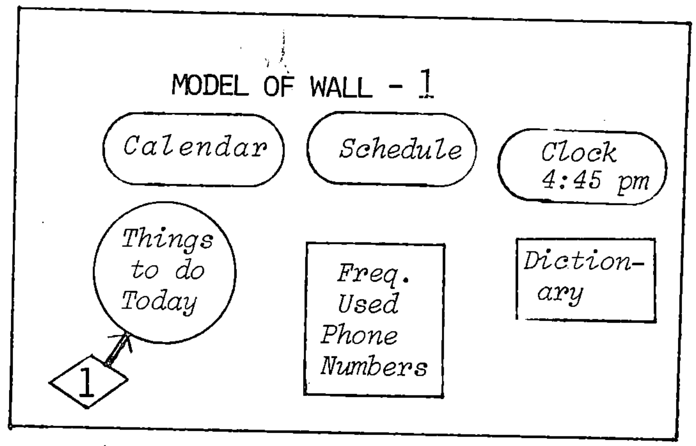
One alternative is to use a projection technology and to magnify a small precise display to cover a wall or a desktop. But this may have the disadvantage of being usable only in darkened rooms.

Another possibility is to use two (or more) normal size displays, as follows. One display would portray the global spatial arrangement of current tasks. This would provide a pictorial model of the imaginary location of current "information contexts" as, say, a picture of where named items are located on a desktop or table.

Local displays would provide views of each "selected information context." To change the context under view by a local display, one could issue a command to call up a named context pictured on the global display. The result, would be that the called-up context would instantly replace the former local context on the local display. Figure 4, below illustrates this notion.

Thus, by buying a box-car full of cheap TV sets, we may be able to achieve a partial approximation to the Arcturus Personal Workstation in case an appropriate cheap flatscreen display technology does not materialize, on time.

Video-disc technology may provide very interesting information retrieval possibilities, especially since programs occupy few bits in relation to the bits needed for, say, five minutes of color TV.



GLOBAL MODEL SCOPE

THINGS TO DO TODAY

1. *Reply to Nelson*
2. *Mail J2 Paper to Int. Conf. on TDMS*
3. *Read intro. to K's Book*
4. *Prep. briefing for Jones*
5. *Confirm travel arrangements to DCA*
6. *Ask George to Lunch*

LOCAL VIEWING SCOPE - 1

MACRO TEN CODE

```
ENTERO: HLLZ A2(i5)
        PUSHJ PLOT4
        CAMIE 4(i5)
        JRST  NONE
        WAIT  2
        POPJ

        ;Normalize T12
        ;First
```

LOCAL VIEWING SCOPE -2

THE USER HAS DESIGNATED THAT THE LIST OF "THINGS TO DO TODAY" SHOULD BE DISPLAYED ON LOCAL VIEWING SCOPE - 1.

FIGURE 4