SINGLE USER SYSTEMS

Ian Willers
CERN, Geneva, Switzerland

Abstract: The first part of the talk will be devoted to establishing concepts and trends in interactive computing technology and is intended to provide a framework. I will then discuss personal computing architectures of today and planned architectures of the 1990s. I will present current personal computer environments for the programmer and for the user. Scenarios for future computing environments will be developed. Finally, I will open a discussion on the social implications of personal computers. Many of the ideas in this paper are covered to greater detail in the proceedings of the Capri conference held this year [1].

1. Introduction

The computing world has been dominated during the last two decades by the multi-user, time-shared machine. The reason for this was an economical one of sharing expensive items between a large number of users. Since many of the components are no longer expensive we have seen the introduction of single user systems in the form of personal computers and personal workstations.

Network technology has evolved so that single user systems may access server machines which offer services which are to be shared amongst the users. The servers will give access to expensive devices which are normally attached to multi-user, time-shared machines. Economics is still the driving force behind the design of such systems.

What is it that makes the single user system so interesting? The dedication of a single processor with a fast graphics interface to a large bit map display has given users a new level of user interaction. Due to the high speed of the graphics the user interface is no longer restricted to being verbal but can include images that can be moved on the screen by the user or by the computer to give animation. Graphical applications now run consistently at high speed no longer being swapped out as another user's program runs.

Adding a new server to a system involves interfacing a device to the network rather than incorporating it into an already complex system making both installation and maintenance an easier task.

At CERN we have begun the introduction of Apollo personal workstations and we have tested a number of other types. The highest priority has been to integrate the Apollo with networks at CERN. The work on the machine itself is based on Graphics and Fortran development. We have developed 2-D graphics using GKS, a 3-D graphics system, PIONS, and a graphics editor, GUI. The kernal of the CERN Fortran library has been ported.

2. Evolution of Technology

If we look at the second half of this century, it began with expensive single user systems allocated to privileged individuals. After three decades of large multi-user, time-sharing systems we can see a trend back to single user systems. Only now more or less any individual can afford to have his own personal computer and professionals are purchasing personal workstations which retain the advantages associated with multi-user, time-sharing systems!

Let us now look at developments during the period 1950-1999.

1950's stand alone, single user systems: This often required booking schemes with computers in operation 24 hours each day. By the end of the decade batch services were in operation. A computer ran a job to completion before taking the next job from the input device. Programs were numerical, data was transient and the first mnemonic assemblers and compilers were conceived.

1960's multi-user, time-sharing machines: Computers were still very expensive and the emphasis was on finding efficient ways of exploiting the hardware. Time-sharing systems such as Multics [2] were developed. The complexity of the systems grew as they were used to manage banks and airline reservation systems. Complex databases were constructed.

1970's big mainframes and software engineering: New machines were developed to handle larger and larger problems. Building software to handle such problems often with large teams of programmers became analogous to a large engineering project. This led to new programming methodologies, abstraction mechanisms and the life cycle concept of software.

1980's component technology. Both hardware and software experts are now concentrating on constructing systems from modular components. Single user systems are linked via networks to each other and to services normally supplied by a time-sharing system. Personal computers have hard interfaces enabling users to tailor their systems. Languages such as ADA [3] and Modula-2 [4] support modules and check the software interfaces between modules. There is also an enormous development of user interfaces using powerful graphics systems on single user machines.

1990's knowledge engineering: Future computers will support expert systems. All types of disciplines will be utilising computers giving them a knowledge support environment.

3. Personal Workstation Genealogy

The early work on personal workstations began in the 1970's at the Xerox Palo Alto Research Center, commonly called Xerox PARC. Their dream was a machine which was the size of a book and resembled in very many ways a sketch-pad. The result of this was a personal workstation. The Alto [5] was begun in 1973 and included a bit-mapped screen and a graphical input device. Xerox has since constructed two research machines, which are called the Dolphin and the Dorado, and an office workstation which is called the Xerox Star.

The Massachusets Institute of Technology, MIT, developed Lisp [6] machines called CONS and CADR. From these a commercial machine, the Symbolics 3600, was manufactured primarily to run the large algebra program MACSYMA [7].

In Europe, ETH in Zurich produced the Lillith [8], with a microcode tailored to the language Modula-2, and Inria in Paris has an office project using the Buroviseur [9].

The American computer manufacturers have now produced a number of personal workstations, the SUN, Lisa, Corvus, PERQ, Apollo etc.

4. Recent Events

In order to get necessary performance the control processor of a personal workstation was constructed out of bit slices which were microprogrammed. In 1979 the Motorola 68000 microprocessor was incorporated instead. The Motorola 68000 is a 16 bit microprocessor which allows 32 bit operations and a performance of 1 MIPS.

As personal workstations form one component of a system, the system was connected using networks. In 1980, Ethernet [10] became commercially available so that personal workstations now either connect directly to Ethernet or will gateway from its own network to Ethernet. Ethernet was another Xerox product.

In 1982, the Motorola 68010 microprocessor was introduced giving a virtual memory of 16 Megabytes to the user. This gives the personal workstation the power to run programs which previously could only run on mainframes. It also gave Motorola 68010 based machines an immediate advantage over bit-sliced machines where microcoding of virtual memory has proved to be difficult.

This year has seen the introduction of very high quality bit-map displays both in black and white and in colour. The improvement in quality has kept within the present price structure.

5. The Personal Workstation as Part of a System

Since the personal workstation is a component of a networking system the operating system should be prepared to aid the user in accessing different components of the systems. A number of solutions to this problem have been proposed and are outlined below.

1. Enhance the operating system by adding remote login and file transfer. This is the solution adopted by UNIX[1] [11] and UNIX-like systems which support the TCP/IP protocols [12]. The personal workstation remains isolated but files can be transferred easily from one system to another using the FTP program. The user can investigate the state of or use another computer by logging in using TELNET program. Computers attached to the network are recognized by their network address.

2. The kernal of the operating system can be written to use message passing whenever the service being requested could be on another computer. The other computer must run a kernal that understands these messages. However the message may be passed locally or via the network. The address space of objects has to be extended so that each object in the network can be addressed. In the case of Apollo the messages are passed over the Domain [13] token ring network and each object has a unique 96 bit address. Programs and files are paged across the network.

3. The third approach is that of the Newcastle Connection [14] as applied to UNIX and UNIX-like systems. In this case calls to the operating system kernal are intercepted and the code determines whether the call is local or not. In the case of a non-local call, a message is passed to the other computer using a remote procedure call protocol. The results are then returned to the caller via another message. Local calls are passed on to the local kernal giving some inefficiency. The address space of objects in the network is extended by using device files referring to other computers.

4. Terminal emulation is the slowest and least powerful technique of integrating system components. It is however the most used method of communicating between single user systems and other computers. The terminal emulator mimics the operation of a terminal to the

---

[1] UNIX is a trademark of Bell Laboratories

computer. It is capable therefore of remote login and a slow form of file transfer. By listing a file, the file's contents are passed over the terminal line and are copied to the disk of the single user system. This transfer is usually limited to the speed of the terminal line which is normally between 300 and 9,600 bits per second. This technique is the most commonly used due to the high availability of terminal lines and the generality of the method.

6. Building Blocks of Personal Workstation Hardware

6.1 Physical Description

The personal workstation is normally constructed out of four items: a processing unit, a screen, a keyboard and a mouse. The processing unit stands on the floor but fits under a table. It will contain the central processing unit, the Winchester disk, the network connection and a variety of connectors for printers, terminal ports etc. This unit normally generates heat and incorporates a number of fans which make noise. These are often stored remotely from the user.

The screen is physically large, in order to support good windowing, with a very fine bit-map display. The windows are black and white and stand out from a grey, alternate black and white pixels, background. The keyboard incorporates a relatively large number of programmable function keys but otherwise is a normal layout. Beside the keyboard is a pointing device called a mouse used for graphical input or pointing.

6.2 The Seven M's

The performance of present personal workstations is judged by such things as processor power, memory size, screen size, disk size and network speed.

6.2.1 1 Mips.

The standard processor today is a Motorola 68010 or like a Motorola 68000 in performance. Personal workstations do exist with three times this performance. Future offerings include the Motorola 68020 with full 32 bit capabilities and the National Semiconductor 32032 which promises to be very fast.

6.2.2 1 Megabyte of memory

The memory is arranged to give about 1 Mbyte of real memory to the user. The resident part of the operating system will take up part of the real memory so it may be necessary to buy a 2 Mbyte machine in order to get such space for the user.

6.2.3 16 Megabytes of Virtual Memory

This is a function of the number of bits reserved by the microprocessor for addressing. New personal workstations exist with 256 Mbytes of virtual memory. It is expected that the limiting factor on the use of virtual memory will be the space allocated on the Winchester disk for page swapping.

6.2.4 Mega-pixel Screen

The bit mapped screens are physically larger than an A4 sheet of paper and have at least 1024x800 pixels. Each pixel is mapped onto memory taking part of the memory away from the user's program unless he directly wants to manipulate the screen. Fast hardware, called raster-op or bit-blt, is used to manipulate the screen.

6.2.5 10 Mbytes of Disk

This again is the amount of disk space allocated to the user. The system with its swap space can take up to 15 Mbytes of space making a 25 Mbyte disk necessary. Economically large disks are a better buy than smaller disks. This has led to the concept of a diskless workstation with a file server supplying disk space on the network. In this type of system multiple accesses to a single disk becomes a bottle-neck.

6.2.6 10 Megabit/second network.

The hardware for the 10 Mbit/second Ethernet is in place and most personal workstations work on networks of comparable speed. Token ring or slotted ring networks working in the 60 to 100 Mbit/second range are being produced and will be marketed in the near future.

6.2.7 Mouse

A mouse is a pointing device used for graphical input. The movements of a metal ball under the mouse can be sensed and the movements passed to the machine. A number of buttons on the mouse enable the pointing to be enhanced by clicking one of the buttons.

7. Building blocks of Personal Workstation Software

7.1 User Interface

The presence of fast hardware for handling the bit-mapped display and a graphical input device has made new user interfaces simple to construct. The main innovation is the presence of a display manager that directs output to windows and allows input via menu selection.

A window is a rectangular area of the screen reserved for input and output of a program. A program need not be restricted to one window, graphics programs often separate text and graphics into separate windows. In a multi-tasking system a number of programs may run concurrently each updating or accepting input from its own windows This may be used for trivial things such as displaying the time or for complicated applications such as program debugging. In debugging the user would typically have a window open for the program, another for the debugger and at least one window viewing the original text of the program. Editing and recompilation can also take place in another window.

The menu system displays a menu of choices in a small window and the user may select options by pointing using a graphical input device such as a mouse. With a good menu system a user of a program need remember very little in order to run quite sophisticated operations. Much research is now being done into user interfaces using menus. The Macintosh represents a good implementation of a menu system where the names of the main menus are permanently on display across the top of the screen and the depth of menus is small. Menus that lead to other menus can lead a user through a labyrinth where he can easily become lost.

7.2 The Synthetic Camera

The ability to run many applications on one screen using good graphics has lead to new thinking in program design, especially, where graphics are directly concerned. The graphical representation of an object on the screen is a picture of some data in the computer that represents that object. This interpretation of the role of a program is to call the program a synthetic camera.

Traditionally, a graphics program would input data and by calls to a standard graphics package such as GKS [15] or CORE [16] represent that data graphically. Unfortunately while GKS and CORE can be used with bit-mapped displays they do not take advantage of their speed. Programs would therefore bypass these packages and directly manipulate the bit map themselves. Graphically the only common graphical representation of an object was the bit-map itself. Unfortunately a bit-map contains no information on the source of the bits. For example, is this a straight line, part of circle or even text? This makes scaling or manipulation of graphical objects impossible.

There has been a move therefore to concentrate on standard ways of representing data and then constructing programs as synthetic cameras to view this data. Object oriented programming is the name given to this type of programming. The objects or data representing these objects are stored in the system and are characterised by their long life. The objects are stored separately from the programs used to view this data. Finally, programs may use other programs to generate views of objects.

For example a text processing program may call upon one program to represent an object by a table, and a second program to represent the same object as a graph. Both the graph and the table are incorporated into a document. Changing the object will enable us to simply produce an updated document. In some systems this is done dynamically so that changes are immediately viewed on the screen within the document.

An example of such a system is 'Office by Example' [17] where data is represented in a standard way as tables in a relational database. The user has a small number of commands whereby he can select data from the database and display it in a variety of ways. A text processor and mail system is available to distribute the information.

Another example is Smalltalk [18] where objects are stored with the procedures and methods of manipulating them. A message to the object will perform one of the defined operations.

8. Recommendations on Personal Workstations in Europe

CERN has conducted some research into the personal workstations most suited to CERN's needs. We were able, along the way, to identify other areas where different personal workstations excel. For CERN the main requirement is for a powerful Fortran environment with a fast compiler, good symbolic debugger, fast editor for large files and a good performance analysis program. The table below shows a number of areas and one machine that excels in that area.

Table 1: Personal Workstations

Application Area	Machine
FORTRAN	APOLLO
CAD	APOLLO
UNIX	SUN
REAL-TIME	MASCOMP
ALGEBRA/LISP	SYMBOLICS 3600
CHEAP (BUT LIMITED)	MACINTOSH
IBM CULTURE	IBM-PC (and look-alikes)

Apollo has concentrated on CAD and has captured a large part of that market. Its machines are tailored for the large Fortran, graphical, programs typical of CAD. Their relatively wide range of personal workstations include integrated colour often required in CAD.

The SUN personal workstation runs the same version of the UNIX operating system as runs on the majority of VAX computers, a popular UNIX host machine. The communications between the systems are over Ethernet and use 'standard' TCP/IP protocols. Other personal workstations such as the Cadmus and the PERQ have implemented the Newcastle Connection.

Mascomp have modified their UNIX system for real-time work and their hardware to accept high data rates. Their version of UNIX quickly reacts to interrupts and tasks may be locked in memory. They also have a lot of experience in interfacing to equipment.

The Symbolics 3600 was developed at MIT to run the Algebra system MACSYMA which is a large program written in Lisp. The machine is adapted to the demands for high processing speed and large virtual memory required by Algebra systems.

The Macintosh is viewed by the personal workstation community as a big advance in personal computers. It is affordable by the individual and contains many of the essentials of a personal workstation. It is a rugged closed system. The closed nature of the system make it unsuitable for many applications.

Finally, the IBM-PC has few properties of a personal workstation. The open nature of the machine and the IBM culture that surrounds it has given it enormous popularity. The recent announcement of the IBM-PC AT introduces windowing in a general way showing that IBM is following the trends.

9. Recommendations on Buying a Personal Computer

Since most people will one day own a personal computer, the market place can only become more competitive at this low end of the range of single user systems. At this stage, the purchase of such a device can cost between $150 and $8000. A Macintosh would normally cost about $2,500 and is made available to students at some universities for as little as $1,100. While prices vary considerably it is necessary to judge the market place carefully before buying.

Some guidelines:

a. Do not acquire a personal computer before you know what to use it for. This forms the basis of your judgement on the suitability of the personal computer.

b. Look for clear definitions of what it can do in the areas that interest you. Good documentation based on clearly defined concepts is essential if you are to make effective user of the machine.

c. Pay attention to the tools supplied for you to enter programs that you have either written yourself or copied from books or magazines. The speed at which you can do this will depend on the efficiency of the tools and the ease of program debugging.

d. Avoid any system that is more complex than you can easily handle. At purchase time you should emphasize that you need a useful tool and not a fascinating toy.

e. Be prepared for the fact that you will need educating in the use of personal computers. However intelligent the salesman says it is, it cannot replace your intelligence on its usage. Evaluate the educational material.

f. Concentrate on the effectiveness of the personal computer's software and hardware and then take your decision.

10. Social Implications

There are three main areas of social importance. One is purely social and the other two are legal. Firstly, there is now a requirement for computer literacy which has created a new set of 'haves' and 'have-nots'. Secondly the copyright laws are proving as ineffective with software as they are with books, music and video. Finally, politicians must control databases.

10.1 'Have' and 'Have-nots'

The fist clear distinction is that some people have personal computers and some do not. Further more, not everyone will be educated in their use. Computer networks give people access to a world of information which is denied others.

10.2 Copyright

The attempt to protect software pirates copying protected software has always failed since the pirates are as clever as the law abiding citizens. IBM software may be copied once and is 'impossible' to copy a second time, at least, using the IBM copy procedure. However a program is sold, legally, that overcomes this restriction. The sale is legal but the use is illegal. The author of this program has also protected his floppy against being copied by his own and by IBM's programs.

10.3 Databases

While politicians believe that little can be done about the first two problems they do believe that politicians must control databases.

In 1974 Sweden passed a law that all databases must be registered. France and Germany more recently applied similar laws. The law was conceived when the number of databases were small. The French body is reportedly overwhelmed with work and consequently can fulfil none of the useful tasks conceived for it. Sweden recently changed the law to exclude all databases that are for personal hobby use.

The U.S.A. government claims to control all financial databases. A small error in a financial database can cause personal problems. For example, a bank may refuse you a loan, a credit card company may refuse to give you a card, both, without the necessity to reveal the exact source of the information.

10.4 Conclusion

The industrial revolution enhanced our physical abilities. The computer revolution will enhance our mental capabilities. The English Luddites wrecked machines, cars kill, the third world is poor and the industrial nations rich. What will the computer revolution bring?

REFERENCES

[1] Proceedings of the International Conference on Advanced Personal Computing Technology. CRAI, Italy. May 1984.

[2] Corbato, F.J. and Vyssotsky, V.A. Introduction and Overview of the Multics System. AFIPS Conf. Proc., Fall Joint Computer Conf., vol. 27, pp 185-196, 1965.

[3] Ada Reference Manual, U.S. Department of Defense, Jan 1983. ANSI/MIL-STD 1815A.

[4] Wirth, Niklaus. Programming in Modula-2. Springer-Verlag. 1982.

[5] C.P. Thacker, E.M. McCreight, B.W. Lampson, R.F. Sproull, D.R. Boggs. Alto: A personal computer. Xerox Palo Alto Research Center. Report CSL-79-11 (1979).

[6] Levin, M., et al. The Lisp 1.5 Programmer's Manual, M.I.T., Cambridge, Mass. 1965.

[7] MACSYMA Reference Manual. Mathlab Group, Laboratory for Computer Science, Mass. Inst. of Tech. 1977.

[8] Wirth, Niklaus. The Personal Computer Lilith. ETH, Zurich.

[9] B. Scheurer and N.Naffah. Local Networks and Individual Workstation in the Office of the Future and in the KAYAK project. French Swedish Symp. Data Proc. Telematics, Automation. Stockholm November 1980.

[10] R. M. Metcalfe and D. R. Boggs, Ethernet: distributed packet switching for local computer networks, CACM (July 1976).

[11] D.M. Ritchie and K. Thomson. The UNIX Time-Sharing System. Communications of the ACM, 17, No, 8 (July 1974), pp 365-375. The original UNIX paper.

[12] Internet Protocol Transition Workbook. SRI International. March 1982.

[13] Nelson, David L. and Leach, Paul J. The Evolution of the Apollo Domain. Hawaii International Conference on Systems Science, 1984.

[14] D.R. Brownbridge, L.F. Marshall and B. Randell. The Newcastle Connection, or, UNIXes of the World Unite. Software Practice and Experience, Dec. 1982, pp 1147-1162.

[15] Graphical Kernel System (GKS), Version 6.6. International Standards Organization. May 1981.

[16] Status Report of the Graphics Standards Committee. Computer Graphics 13(3), August 1979.

[17] Moshe M. Zloof. QBE/OBE: A Language for Office and Business Automation. IEEE Computer, May 1981.

[18] Goldberg, Adele and David Robson. Smalltalk-80: The Language and its Implementation, Addison Wesley, 1983.