Exchange

Exchange

July/August 1986

Software

Getting Started

Random Data

Departments

Exchange of IBM PC Information
IBM

Exchange of IBM PC Information is a monthly publication of the National Distribution Division, International Business Machines Corporation, Boca Raton, Florida, USA.

Managing Editor	Michael Engelberg
Technical Editor	Bernard Penney
Associate Editor/ Design Director	Karen Porterfield
Writer/Automation Consultant	John Warnock
Editorial Assistant	Lowell Blackham
User Group Editor	Steve Mahlum
Automation Consultant	Sherry Reardon
Illustrator	Jeff Jamison
User Group Support Manager	Gene Barlow

Exchange of IBM PC Information is distributed at no charge to registered PC user groups. To register with us, please write to:

IBM PC User Group Support
IBM Corporation (4704)
P.O. Box 3022
Boca Raton, FL 33431-0922

To correspond with *Exchange*, please write to:

Editor, *Exchange*
IBM Corporation (4704)
P.O. Box 3022
Boca Raton, FL 33431-0922

POSTMASTER: send address changes to *Exchange of IBM PC Information*, IBM Corporation (4704), P.O. Box 3022, Boca Raton FL 33431-0922

IBM cannot be responsible for the security of material considered by other firms to be of a confidential or proprietary nature. Such information should not be made available to IBM.

IBM has tested the programs contained in this publication. However, IBM does not guarantee that the programs contain no errors.

IBM hereby disclaims all warranties as to materials and workmanship, either expressed or implied including without limitation, any implied warranty of merchantability or fitness for a particular purpose. In no event will IBM be liable to you for any damages, including any lost profits, lost savings or other incidental or consequential damage arising out of the use or inability to use any information provided through this service even if IBM has been advised of the possibility of such damages, or for any claim by any other party.

Some states do not allow the limitation or exclusion of liability for incidental or consequential damages so the above limitation or exclusion may not apply to you.

It is possible that the material in this publication may contain reference to, or information about, IBM products, programming or services that are not announced in your country. Such references or information must not be construed to mean that IBM intends to announce such IBM products, programming or services in your country.

User Group Support
(305) 998-PCUG
User Group Locator System
(305) 998-1575
Electronic Bulletin Board System
(305) 998-EBBS

What's New in BASIC 3.20

John Warnock
IBM Corporation

The BASIC language interpreter has been evolving and improving along with the Disk Operating System (DOS) to support new members and features of the IBM Personal Computer family. With the recent announcement of the IBM PC Convertible and 3.5-inch diskette drives, IBM also announced DOS 3.20. BASIC 3.20 is part of that offering.

BASIC 3.20 is a logical extension of BASIC 3.10, which added device driver support, environment table support, device error support, and the ability to run DOS commands from within a BASIC program. New capabilities in BASIC 3.20 include support for the IBM PC Network and the IBM Graphics Adapter.

New commands are added and others are enhanced:

SCREEN statement supports four new screen modes.

PALETTE and PALETTE USING statements give you more flexible use of colors.

PCOPY statement lets you copy between pages in screen memory.

OPEN statement supports new file access features.

LOCK and UNLOCK statements control file access.

ON KEY(n), KEY(n), and KEY statements let you trap additional keys.

EXTERR(n) function accesses DOS extended error information.

Documentation

While BASIC 3.20 comes on the same diskette as DOS 3.20, the *BASIC Reference Manual* is ordered separately. (This is because many people use programs written in BASIC, but not everyone writes BASIC programs.) The manual comes with a *Handbook of General Programming Information* and a *Quick Reference* to supplement the *Reference Manual.*

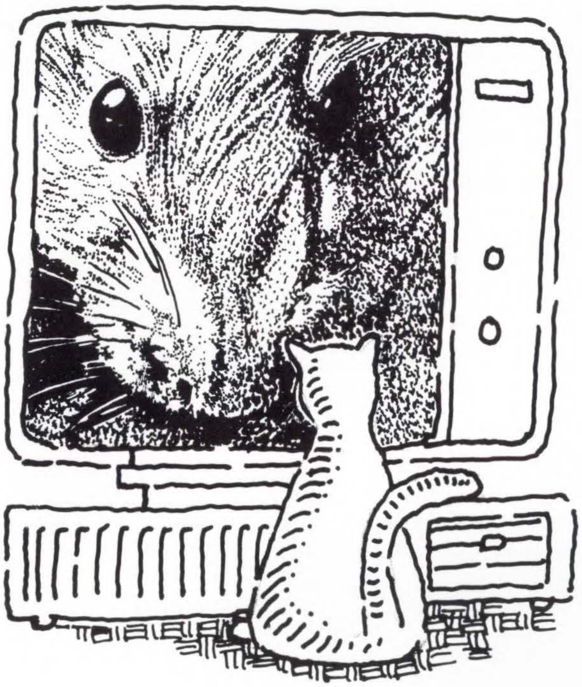
The *Handbook of General Programming Information* provides general rules and guidelines for programming in BASIC. The *Quick Reference* summarizes each BASIC statement and command, and gives examples. The *Reference Manual* discusses statements, commands, and variables in depth.

SCREEN

The SCREEN statement has four new modes to support the Enhanced Graphics Adapter. The additional modes are:

7 medium-resolution graphics (320 x 200) with up to 16 colors.

8 high-resolution graphics (640 x 200) with up to 16 colors.



9 enhanced high-resolution graphics (640 x 350) on the Enhanced Color Display only.

10 enhanced high-resolution graphics (640 x 350) on the IBM Monochrome Display only.

For example, to get high-resolution graphics on the Enhanced Color Display, you would use the statement:

```
SCREEN 9
```

GET

The GET statement for graphics now supports SCREEN modes 7, 8, 9, and 10. SCREENs 7 and 8 support 16 attributes and require four bits per pixel. SCREEN 9 supports either four attributes with two bits per pixel or 16 attributes with four bits per pixel; however, the latter requires more than 64KB of memory on the Enhanced Graphics Adapter. SCREEN 10 supports four attributes with two bits per pixel.

COLOR

The COLOR statement now lets you define your own palette (see PALETTE and PALETTE USING below), and select from up to 64 colors depending on the type of monitor and the screen mode.

You can specify background and palette colors with SCREEN 1, and foreground and background colors for SCREENs 7, 8, 9, and 10. (The COLOR statement is illegal with SCREEN 2.)

PALETTE and PALETTE USING

PALETTE and PALETTE USING are new statements in BASIC 3.20. They let you control the hardware of the Enhanced Graphics Adapter. The PALETTE statement takes the form:

```
PALETTE [attribute,color]
```

where

attribute is an integer within the range allowed by the amount of memory on your EGA, the SCREEN mode in effect, and the type of display you are using.

color is an integer expression within the range allowed by the SCREEN mode in effect and the display you are using.

For example, the statement:

```
PALETTE 3,6
```

assigns the color 6 to the third palette attribute.

PALETTE USING lets you assign all palette attributes in one statement. Like the PALETTE statement, this statement is valid only with the Enhanced Graphics Adapter. This statement appears as:

```
PALETTE USING arrayname(starting)
```

where

arrayname is an integer array containing color values.

starting is the number of the first array element from which to start assigning color values.

PCOPY

PCOPY lets you copy one page of video memory into another. This statement is valid for SCREEN modes 0, 7, 8, 9, and 10 with the Color/Graphics Monitor Adapter or Enhanced Graphics Adapter. The format is

```
PCOPY source page,destination page
```

where

source page is an integer expression within the range allowed by your configuration.

destination is an integer expression within the range allowed by your configuration.

OPEN

The OPEN statement lets you access a file or device. BASIC 3.20 lets you control access and locking as well. To control access and locking, use the form:

```
OPEN filespec [FOR mode][ACCESS access][locking]
      AS [#]filenum[LEN=recl]
```

where

filespec is a string expression for the drive, path, and filename.

mode specifies the file type and access method. It can be

OUTPUT opens the file for sequential output.

INPUT opens an existing file for sequential input.
APPEND opens an existing sequential file for adding records to the end.
RANDOM opens a random access file for input or output.

access specifies types of operations allowed on random access files. It can be

READ opens the file as read-only.
WRITE opens the file for write operations only.
READ WRITE opens the file for both read and write access.

locking restricts access to other processes or programs. It can be

SHARED anyone may use the file.
LOCK READ no one else may read the file.
LOCK WRITE no one else may write to the file.
LOCK READ WRITE no one else can have read/write access to the file.

filenum is an integer between 1 and the maximum number of files allowed open.

recl is an integer between 1 and 32,767 that sets the record length for a random file.

To open a random-access file as read-only and let everyone share it, you would say:

```
OPEN "testfile.dat" FOR RANDOM
     ACCESS READ SHARED AS #1
```

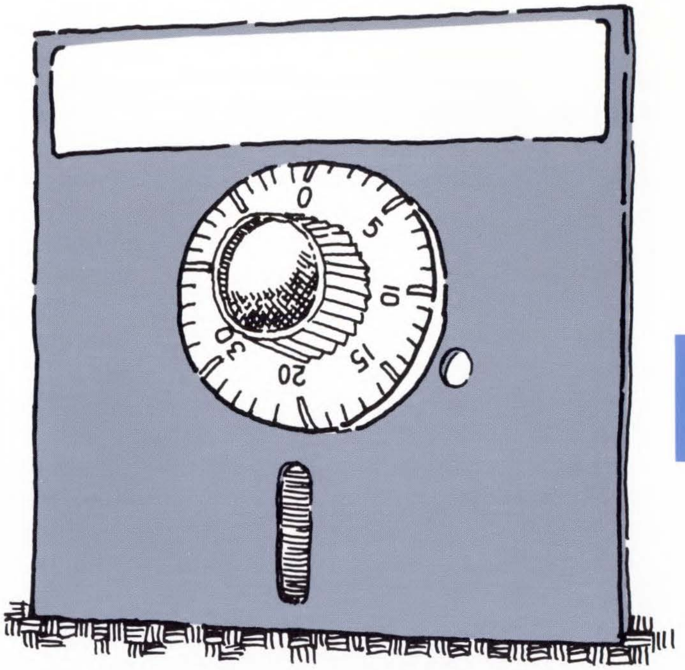
LOCK and UNLOCK

The LOCK and UNLOCK statements are new in BASIC 3.20.

The LOCK statement restricts file access by other processes or programs. It takes the form:

```
LOCK [#]n[,[recnum] [TO recnum]]
```

where



n is the number of the opened file.

recnum is the starting or ending number of a range of records to be locked.

Note: If you fail to unlock a file before ending your program, unexpected results may occur.

If you were modifying the fifth record in a file, and wanted to prevent others from tampering with it, you would use the statement:

LOCK #1, 5

ON KEY(n), KEY(n), and KEY

The ON KEY(n) statement lets you branch to other BASIC statements when you press the F11 and F12 keys on the IBM Personal Computer XT and Personal Computer AT Enhanced Keyboards and keys 21-25 (which are the Ctrl, Shift, or super-shift keys defined by the KEY statement).

The KEY(n) statement lets you activate trapping for the F11 and F12 keys on the IBM PC XT and AT Enhanced Keyboards and keys 21-25.

The KEY statement lets you assign a string of characters to a function key. With BASIC 3.20 you can assign character strings to keys 30 and 31 as well as the original 1-10. Keys 30 and 31 are valid only for the IBM PC XT and AT Enhanced Keyboards (function keys F11 and F12).

EXTERR(n)

The EXTERR(n) function is new in BASIC 3.20. It returns DOS extended error information when your program performs DOS functions. Its format is:

EXTERR(*n*)

where

n	Value returned
0	Extended error code
1	Extended error class
2	Extended error suggested action
3	Extended error location

BLOAD and BSAVE

If you used BSAVE in earlier versions of BASIC to create a screen file from a Monochrome Display and Printer Adapter, and if the screen had a background value of 1 to 6, you will get different results when you BLOAD the screen on an IBM Monochrome Display attached to the Enhanced Graphics Adapter. You should either use BASIC 3.20 to rerun the program that created the file, or change the background value to 0. BLOAD and BSAVE support the new graphic modes, SCREEN 7, 8, 9, and 10.

In summary, BASIC 3.20 provides greater file control capabilities in a networking environment. It also greatly enhances display color possibilities when used with the Enhanced Graphics Adapter. Improved keyboard handling and error tracking capabilities round out the improvements in BASIC 3.20.

EZ-VU Enhancements

Kim Krostue
IBM Corporation

The Interactive System Productivity Facility for the IBM Personal Computer, Version 2, EZ-VU II Development Facility and EZ-VU II Runtime Facility is a strategic productivity aid for the IBM Personal Computer. It is a dialog manager designed to increase the productivity of the application programmer. Now developers can create more sophisticated applications with greater ease. EZ-VU combines easy-to-use screen design facilities with professional level functions and services.

The ISPF/PC family consists of two main parts:

- The EZ-VU Development Facility — part number 6317026
- The EZ-VU Runtime Facility — part number 6317025

The Development Facility prepares the screens that are displayed by the Runtime Facility (as directed by the application program). The supported IBM languages are BASIC, Compiled BASIC, Macro Assembler, Pascal, C, COBOL, and FORTRAN. For other languages, a generalized program interface has been published.

EZ-VU is designed to shorten the application development cycle. It saves the programmer from complex and time-consuming tasks such as coding display formats, range and error checking, and program linkage. In addition, EZ-VU dialogs can be written to install and maintain software.

EZ-VU II provides all the function of EZ-VU I plus much more. All dialogs developed with EZ-VU I are fully upward compatible with EZ-VU II.

IBM products which require EZ-VU are ISPF/PDF PC Editor (EZ-VU Editor), PROFS/PC2, PC

Organizer, EZ-Prep, PC-Videotex Graphics Aid Facility, and PC Network Analysis. EZ-VU is the PC version of ISPF, IBM's 370 host dialog manager.

Customer Developed EZ-VU Applications

The following types of applications have been developed using EZ-VU:

- A major insurance company has developed a commercial lines rating application using EZ-VU. The total number of panels developed exceeded 900.
- Another major insurance company is using EZ-VU as the "glue" for a developer's workbench.
- Yet another major insurance company used EZ-VU to offload an IMS application.
- A major package delivery service used EZ-VU to implement an application to support international service.
- Another user used EZ-VU to make a PC resemble a cash register.

The EZ-VU II Development Facility (DF)

The Development Facility (DF) is a tool to simplify panel design and dialog development for the application developer. The components are the Screen Design Facility, the Screen Development Utility, the Language Interface Modules, the Profile Edit Utility, the Configuration File Utility, the EZ-VU II tutorial, and the Panel Library Utility.

Highlights of the Development Facility are:

Multiple Panels

The Development Facility lets developers create multiple active panels for use by end users. Panel location may be specified at screen design time, in the application, or as a runtime parameter. This gives the programmer more flexibility in choosing how the applications are presented to the user.

Panel Conversion Utility

This utility converts PC panels to host ISPF panels so that developers can port their panels to the IBM 370 host.

Panel Test Facility

DF allows programmers to test panel designs, including multiple panels, without developing a dialog.

Pop-Up Panels

DF allows a first panel to be overlaid by a second panel without erasing the data on the first one. Thus a user can call up a HELP, MENU, or other desired panel while running an application, read it, and clear it.

The Development Facility lets developers create multiple active panels for use by end users.

Fast Editing

There are no complicated panel definition commands. DF provides a WYSIWYG (what-you-see-is-what-you-get) editor that shows input as it is keyed in. Input data can be moved, copied, changed or deleted with a few keystrokes.

Picture Edit

This allows representation of phone numbers, social security numbers, dollars and cents, serial numbers, and other such data items. Using this feature replaces time-consuming developer-generated editing for panel fields.

Undo

This "oops" function enables instant correction of panel design errors.

Panel Library

Working with the EZ-VU II Runtime Facility, this function permits better developer control of the set of panels required for a specific application.

Setpoint Visual Indicators

These indicators make panel, area, and field design easier and faster.

Levels of Emphasis

Developers can now define a panel using logical attributes. Thirty-two levels of emphasis are supported by DF. This permits the programmer to design panels to be executed on both monochrome and color displays; the colors and highlight attributes are easily altered by the user without modifying the panel.

Configuration Utility

This utility allows the developer to specify runtime variables such as variable pool sizes.

The Dialog Manager

The program that oversees dialog processing in EZ-VU is known as the Dialog Manager (EZ-VU II Runtime Facility). This program sets up and controls the environment in which interactive applications run. An application is first invoked under EZ-VU, then makes calls to the Dialog Manager for various services.

Highlights of the Dialog Manager are:

Select Service

EZ-VU II allows developers to create a series of selection panels, including specially tailored messages, without lengthy coding.

Execution Control

The EZ-VU II Runtime Facility is reentrant. This lets the developer construct nested programs through use of the Select function.

Selection Fields

Developers can create selection panels using selection fields. With this format, users can select from the displayed lists with a simple point-and-poke interaction, using the cursor and a function key.

Selection Field Prompting

With this feature, the application developer can include a comment, description or instruction that will be displayed automatically when the user chooses a particular selection field.

Floating Point

EZ-VU II provides a convenient way to work with dollars and cents and other floating point variables.

Performance

Two features improve panel display performance:

- In-memory panel storage allows frequently used panels to be loaded into storage.
- Direct buffer write is an option that the developer can select.

Application Programming Interface Extension

This feature applies to both the EZ-VU II Development Facility and the EZ-VU II Runtime Facility. The design of EZ-VU II is open-ended. That is, developer provided extensions can be used to expand the set of services provided by EZ-VU. For example, an extension could be written to interface to the voice card and the extension could be called like any other EZ-VU service. Both floating point support and picture edit are implemented as extensions to EZ-VU.

EZ-VU II Operating Environment

The EZ-VU II Development Facility and the EZ-VU II Runtime Facility require DOS 2.00 or higher. EZ-VU II operates properly under TopView but does not require it. In the TopView 1.10 environment, multiple EZ-VU II dialogs require only one resident copy of the EZ-VU II Runtime Facility.

EZ-VU II is designed to provide additional capabilities in the application development environment rather than duplicate the function of TopView. EZ-VU II operates properly on all PC family hardware. EZ-VU II developed applications will run on Personal Computers with as little as 192KB of memory. The Development Facility requires 256KB.

Summary

In summary, EZ-VU is a strategic productivity aid for the IBM Personal Computer. EZ-VU was designed to be a dialog manager for the PC. EZ-VU operates in the constrained storage environment of the PC while taking advantage of its unique hardware functions.

EZ-VU is functionally similar to host ISPF. For the professional user, application development with ISPF/PC -- EZ-VU II is the same as for the host version of ISPF.

EZ-VU enhances even the most skilled PC user's ability to generate new applications. EZ-VU lets them integrate several applications, linking them together with menus and dialogs. It also allows applications to be installed correctly.

IBM PC XENIX Version 2.00

Henry H. Thomas
IBM Corporation

(Editor's note: The following article is adapted from the IBM Personal Computer Seminar Proceedings.*)*

IBM Personal Computer XENIX Version 2.00 is an operating system for the IBM Personal Computer AT that utilizes the advanced features of the 80286 processor. The IBM PC XENIX system brings many of the features normally found on mainframe computers to the IBM Personal Computer AT.

The IBM XENIX Operating System Version 2.00 is a multitasking, multiuser system derived from AT&T UNIX System V Operating System, Release 1. On the PC AT, multiuser means that up to three users can use the system at the same time. Two users can access PC XENIX through ASCII terminals or PCs and the third user can access PC XENIX using the system console. The multiuser capability also allows multiple usage of the same file or program. This capability makes possible the immediate updating and sharing of critical data from several sources.

PC XENIX file management controls individual file execution permissions, read permissions and write permissions for file security and integrity.

Multitasking means the concurrent running of several programs, including the programs of one or more users. The PC XENIX system allows programs and tasks to be initiated as background processes, freeing the terminal to do other work while a time-consuming task is being completed in the background. This can increase productivity, because tasks run simultaneously rather than in sequence.

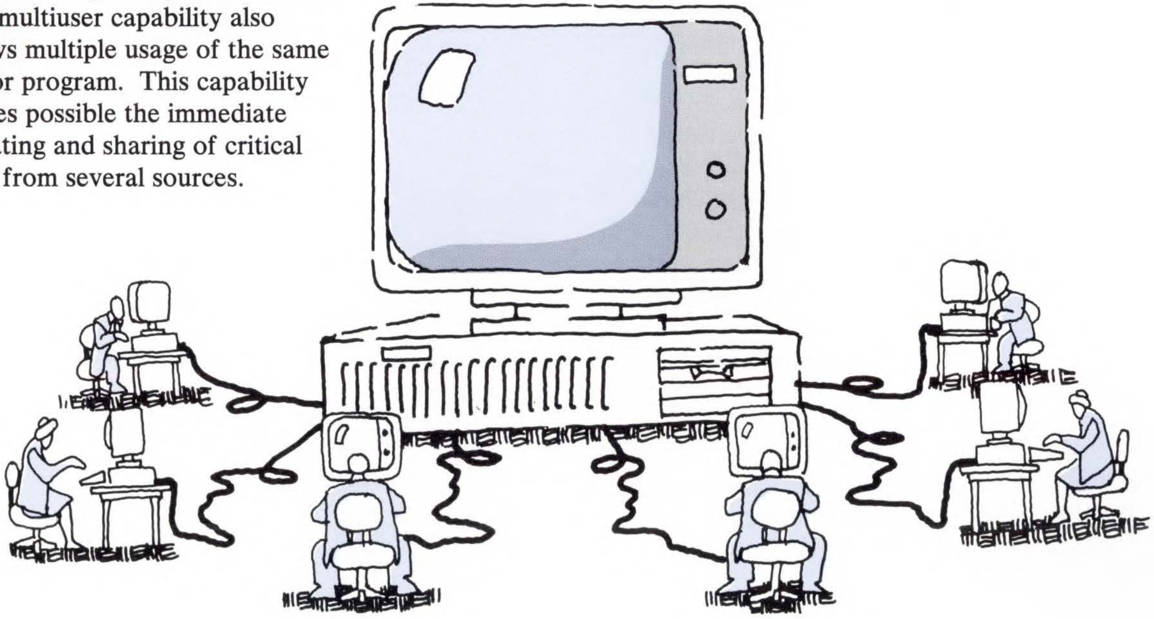
The IBM PC XENIX Version 2.00 system is comprised of four separate program products. The Operating System is the base package. It requires an IBM Personal Computer AT with 512KB of memory, 20MB fixed disk drive, 1.2MB diskette drive and a display. The Operating System is a prerequisite for the other three products, the Software Development System, Text Formatting System and Operating System Extensions.

The Software Development System contains a large set of application software development tools, including the C programming language.

Together, the C compiler, 80286 Macro Assembler, interactive debugger, productivity tools and other programming utilities provide a complete environment for the programmer.

The Text Formatting System provides high-level formatting macros for document preparation in conjunction with NROFF (a formatter for printers) and TROFF (a formatter for phototypesetters). Additional formatting facilities also are provided.

The Operating System Extensions provides additional functions in the IBM PC XENIX environment which enhance connectivity and compatibility with other IBM UNIX derivative products. It includes a full-screen editor with windowing, a user interface driven by function keys, UNIX System V accounting functions and versatile intersystem communication facilities.



The IBM Personal Computer XENIX Operating System, Software Development System and Text Formatting System were developed for IBM by Microsoft Corporation, Bellvue, Washington. The IBM Personal Computer XENIX Operating System Extensions was developed for IBM by INTERACTIVE Systems Corporation, Santa Monica, California.

The Kernel

The Operating System is composed of many parts. One of these parts is the PC XENIX kernel. The kernel is the central part of the operating system. It controls all the resources of the computer and allocates them to the active processes. The kernel manages the file system, automatically assigning disk space to files and directories as they are created or enlarged, and returning space no longer used to the free list.

The kernel takes advantage of the protect mode of the IBM Personal Computer AT 80286 processor to protect itself from the users and to protect the users from each other. While operating in protect mode, the processor is architecturally designed to address up to 16 megabytes of real memory and considerably more logical memory. Utilizing this mode, PC XENIX is capable of managing as much memory as can be installed in the IBM Personal Computer AT.

The kernel also controls access to the system and to the files through password protection and file permissions. Each user is granted access to only those files and directories that he or she is authorized to use.

Reconfigurable Kernel

The IBM PC XENIX kernel can be reconfigured in various ways to more specifically address the needs of the user. The following partial list of configurable items serves as an indication of the level of configurability designed into the PC XENIX product:

- Size of kernel buffers
- Number of concurrent processes
- Number of open files per process

PC XENIX is designed to let users quickly and easily modify the parameters that define each of these kernel features.

Device Drivers

The most flexible feature of PC XENIX is its ability to conform to a wide range of user-specific hardware. This is accomplished through a kernel module called a device driver. A PC XENIX device driver is a software interface that communicates with a hardware device and provides a means by which PC XENIX can control the device to perform input and output operations.

PC XENIX is shipped with the following device drivers for the PC AT:

- Keyboard device driver
- Display device driver
- Asynchronous communications device driver
- Diskette drive device driver
- Fixed disk device driver

Device drivers may be written to provide an enhanced level of function over that shipped with the product. User-supplied device drivers must be incorporated into the PC XENIX kernel.

PC XENIX supports linkable device drivers and installable device drivers. The PC XENIX system administrator can link linkable device drivers into the kernel by remaking the kernel. Installable device drivers can be loaded by the operating system during system boot. The installable device driver feature allows much simpler device installation procedures.

Once loaded, the installable device driver is installed into the kernel image in order to provide access to the device it is designed to control. The creation of an installable device driver requires only a small additional effort over that required for ordinary drivers linked into the kernel image during a configuration process.

To create an installable device driver, the following data structures must be added to an existing device driver, or these data structures must be included when developing a new device driver:

struct iddsw This data structure contains control information for the program that performs installation during bootstrap.

struct bdevsw This data structure is included if the device driver is a block device driver (e.g., tape drive or disk drive).

struct iddsw This data structure is included if the device driver is a character device driver (e.g., terminal or line printer).

After the installable device driver is written for the system, one can then prepare the device driver object module to be an installable image that the boot program can dynamically link to the kernel at bootstrap time. This produces an installable device driver. Linkable device drivers are those that can be linked directly into the PC XENIX kernel. The PC XENIX Software Development System must be installed on the system to produce a linkable device driver. To build a linkable device driver, do the following steps:

1. Move the device driver source code to the /usr/sys/io directory.

2. Add the name of the device driver file objects to the OBJS list in the makefile in the /usr/sys/io directory.

3. Type ***make*** in the /usr/sys/io directory to create the device driver object files.

4. Modify the master file in the /usr/sys/conf directory to include information about the new driver.

5. Modify the xenixconf file in the /usr/sys/conf directory to include information about the new driver.

6. Build a version of the kernel containing the new drivers by typing ***make*** in the /usr/sys/conf directory.

7. Copy the new version of the kernel to the root (/) and enter it as the executable image to the boot program.

Device drivers are the link between the PC XENIX kernel and the external world. The capability to alter or enhance this connection provides the user with a high degree of flexibility. Guidelines for the development and installation of device drivers can be found in the *IBM Personal Computer XENIX System Administration Guide* and the *IBM Personal Computer XENIX Application Development Guide.*

Over 150 commands and programs are available with the PC XENIX Operating System.

Commands and Usage

Over 150 commands and programs are available with the PC XENIX Operating System. They include two editors, Vi and Ed. Vi is a full-screen editor that has many commands to insert, replace, move and search for text. Ed is a line editor available in most UNIX-derived systems. Other commands are available to create and maintain files and directories, create file systems, mount and unmount file systems, link files across directories, and communicate with other users. The capabilities of linking and combining these commands with shell scripts, redirection and pipelines are among PC XENIX's most powerful features.

IBM PC XENIX offers you a choice of three shell command interfaces to the system. You can select the shell that best suits your needs and experience.

Bourne Shell

The Bourne shell (sh) is a command language interpreter that serves as the interface between the user and the rest of the system. It interprets commands, calls the corresponding programs into memory, and executes them. The shell serves as both an interpreter and a programming language. This permits the creation of new commands simply by writing shell scripts (also called procedures). The shell language contains flow control constructs such as WHILE, IF, ELSE, FOR, DO and CASE. It also allows the definition of variables, parameter passing and the processing of software interrupts.

A shell script is a file that contains a collection of PC XENIX commands, pipelines, shell conditional and flow control constructs, and parameter names. The procedure is invoked simply by typing its name, optionally followed by one or more arguments.

Visual Shell

The PC XENIX Visual Shell (vsh) is a full-screen shell interface that optionally replaces the Bourne shell as the command interpreter. It is recommended for casual users of the system and users with specialized needs. The Visual Shell replaces the operating system command line with a visual menu of the most commonly executed applications and utilities, and provides help information for each selection. The user can select a file for use with a command simply by pointing to it via cursor movement keys.

Another important feature of the Visual Shell, especially valuable to software developers, is the ability to customize the shell easily. By developing appropriate shell scripts or programs and changing the words on the menu, application developers can make a simple application interface for the occasional system user. This flexibility enables complicated data processing tasks to be performed with minimal computer knowledge.

C Shell

The C shell (csh) is a command language interpreter for more sophisticated users. Like the Bourne shell (sh), the C shell is an interface between the user and the operating system, translating command lines typed at the terminal into corresponding system actions.

The C shell offers features above those provided by the Bourne shell. The C shell can maintain a history list, into which it places the text of previously entered commands. Using a special notation, one can reuse commands or words from previously entered commands to form new commands. This mechanism is useful for repeating previous commands or correcting typing mistakes.

The alias facility simplifies commands, supplies default arguments, and performs transformations on commands and their arguments. This mechanism enables the user to substitute a personally preferred name for a system command, for a system command and its arguments, or even to represent multiple commands or pipelines.

Also provided by the C shell are language structures similar to those of the C language, including WHILE, FOR EACH, IF, ELSE and SWITCH.

Input/Output Redirection

The PC XENIX shell structure provides a flexible means of redirecting standard input and output. The standard input is the keyboard and the standard output is the display. By using this feature, each physical I/O device — including the keyboard, the display and other peripherals — can be treated as any normal file. Thus, via a shell, the input to a command can come from the keyboard or from a file stored on a disk, and still look like the standard input. Similarly, the output from a command can be redirected from the display to the printer or a disk file. With these capabilities a single PC XENIX command can often perform several distinct tasks. For example:

```
cat x y > z
```

concatenates files x and y and puts the result in file z.

Pipelines

PC XENIX has the ability to pipe data from one command to another. Two or more connected programs are called a pipeline. The shell can execute a sequence of commands, with the output of each command becoming the input to the next command in sequence. This ability to pipe commands often eliminates the need to develop new programs for new applications. For example, if an application requires sorting a file of words, removing all duplicate entries, counting the number of remaining words, and printing the results, four existing PC XENIX programs can be piped together to perform this new action:

```
sort words | uniq | wc | lpr
```

The vertical bar is the pipe symbol. The meaning of this pipeline is (1) sort the file named words, (2) run the sorted list through the Uniq program, which removes duplicate entries, (3) run the output through the WC program, which counts the number of words, and then (4) print the output of WC on the printer using the LPR command.

The PC XENIX File System

The PC XENIX file system consists of files and directories arranged in a tree-like, hierarchical structure. It allows filenames up to fourteen alphanumeric characters in length, and differentiates between uppercase and lowercase. The PC XENIX file system consists of many levels or hierarchies, i.e., directories within directories.

The file system has a unique path for each file in the system. This unique path is the name of the file in relation to the root of the file system. The root of the file system is the parent of all other members of the file system and is designated by the slash symbol (/). Within the file system, some directories are standard and contain PC XENIX command files. The directories /bin and usr/bin contain most of the common commands. The directory /etc has the system administrator's commands. There are other standard directories such as /usr, where user directories and files are stored.

Each user of the PC XENIX system is assigned a login or home directory. When a system user logs in using the assigned login name, that user is automatically in his own directory, which is a directory with the same name as the login name in the /user directory. Starting at the home directory, a user can create and delete files and directories as needed.

Editing Facilities

Two editors are included in the PC XENIX Operating System. A third editor, provided in the Operating System Extensions, will be discussed later. Together, these editors provide varied approaches to editing because they are designed to suit the needs of two groups — the programmer who prefers line editing and the user who prefers a full-screen editor. Any ASCII text file can be created and modified using one of two text editors, Ed and Vi, available with the PC XENIX Operating system.

Communications

Among the several ways to communicate with other users on the PC XENIX system are the two basic facilities, WRITE and MAIL.

The WRITE command allows communication with any user currently logged on the system. One user can write to another user, then wait for the other user to write back. Following the execution of the write command, whatever is typed on the first user's terminal appears on the other user's terminal, and vice versa. The users take turns reading and writing messages until one decides to end the conversation.

The other basic way to communicate with another user is with the MAIL command. MAIL sends a message or a previously created file to the other user. The receiving user does not have to be logged on or even be a user on the same system to receive mail. Mail is delivered to the user's mailbox directory, and when the receiving user logs in, a message indicates that mail has arrived. The receiving user can then read, delete, save or transfer the received mail to other users.

Networks

To communicate with a user who is not on the same system (e.g., the login or home directory is in another system), additional communication facilities may be needed. First, the two systems must be able to communicate with one another. If the systems are not far apart, perhaps across the room, RS232 lines and the serial ports on the IBM Personal Computer AT can be used to form a network. If the other system is across town or across the country, a modem and the UUCP communication system can be used to complete the network.

PC XENIX is designed so that part of the fixed disk may be reserved for other operating systems... such as DOS.

In PC XENIX, the UUCP command retains its UNIX meaning of UNIX-to-UNIX copy. IBM PC XENIX also has Micnet, a limited local network that connects personal computers via the serial ports. After a form of communication has been established between the two systems, the MAIL command is used to communicate with users in the other system. Commands within Micnet allow copying or transferring files and accomplishing tasks such as executing commands on a remote PC XENIX system. The UUCP communication system can combine the usefulness of local Micnet systems by connecting them via modems.

Co-Residence of Multiple Operating Systems

Some PC XENIX users may wish to use another PC operating system. PC XENIX is designed so that part of the fixed disk may be reserved for another operating system, such as IBM Personal Computer DOS. During installation, PC XENIX permits specifying a disk partition for PC XENIX that is less than the full size of the disk, so that subsequent installation of another operating system on the remainder of the disk is possible. PC XENIX requires a minimum of three partitions.

If you intend to install multiple PC XENIX software products on your PC AT, you should limit the amount of disk space you allocate for an operating system partition accordingly. For example, with the Operating System, Software Development System, Text Formatting System and Operating Systems Extensions all installed on a 20M byte fixed disk, approximately 17.5M bytes are utilized. This leaves approximately 2.5M bytes for an additional operating system.

PC DOS Utilities

PC XENIX provides the capability to transfer files between IBM Personal Computer DOS diskettes or DOS fixed disk partition and the PC XENIX file system. There are several utilities to access DOS files and directories on DOS diskettes or fixed disk. The commands have a DOS prefix. They include support to copy files, list files and directories, erase files, and create and delete directories.

One of the PC XENIX commands to access DOS files is DOSCP. It copies files between a DOS diskette and the PC XENIX file system. DOSCP takes two arguments, file1 and file2, and copies the contents of file1 to file2. Either filename can be a DOS file by using the following form:

```
device: file
```

where ***device*** is the path name of the device containing the DOS file.

A simplified way to specify the DOS device is provided in PC XENIX via a set of letters that represent the DOS devices. For example, to copy a DOS file named file1 from a DOS diskette in drive A to the PC XENIX file system and keep the same filename, the syntax would be:

```
doscp A: file1 file1
```

To copy file1 from the PC XENIX file system to a DOS diskette in drive B and rename file1 to file2, the syntax would be:

```
doscp file1 B: file2
```

Terminal Support

Termcap is a data base describing terminals. It is used by programs, such as Vi, to interface with a user's terminal. IBM PC XENIX Termcap supports many ASCII terminals, including those manufactured by vendors other than IBM.

Some of the terminal features described in a Termcap data base entry include the number of lines, the number of columns on each line, how the screen is cleared, backspace capability, cursor motions, highlighting, underlining and insert/delete character keys. Termcap support also can be easily added for new terminals as they become available.

As a complement to Termcap, Curses is a screen updating and cursor movement set of library functions that are packaged as part of the IBM Personal Computer XENIX Software Development System. Curses relies on the information in the Termcap data base to tailor terminal commands that allow the user to write terminal-independent programs. These functions provide a simple and efficient way to use the capabilities of the terminal attached to the program's standard input and output files.

Summary

The IBM PC XENIX Version 2.00 products support the maximum memory available in the IBM PC AT and provide multiuser, multitasking capabilities that are especially useful in an environment where several people work together on a common project. The PC XENIX system, with existing IBM PC AT hardware, enables up to three users to work concurrently on a single IBM PC AT. It also provides electronic mail and communications capabilities to local and remote systems. IBM PC XENIX is licensed for eight users and exploits the capabilities of the advanced 80286 architecture.

Book Excerpt: C Wizard's Programming Reference

David Schwaderer
IBM Corporation

(Editor's note: This article consists of excerpts from C Wizard's Programming Reference, *by David Schwaderer, published by John Wiley & Sons, Inc. Reprinted with permission of the publisher.)*

C is an interesting language. At its best, it provides remarkable access, portability, and freedom to programmers. At its worst, it provides the unwary programmer memorable experiences in program and, perhaps, operating system annihilation. Successful C compilation and linking hardly implies successful program execution. Furthermore, because of C's permissiveness, apparent successful execution hardly implies program portability or data portability. Yet, for increasing numbers of thoughtful programmers, it is the best language, period. C is emerging as the language of the 1980's.

Section 1.

The Compilation Process and Its Implications

A C program consists of one or more functions with attendant declarations and definitions. Every program must have a function named main() which marks the initial point of code execution.

A program can span several files allowing partial compilation on a file basis. While extremely useful, the process of separate compilation introduces the opportunity for various object declaration errors. Figure 1.32 shows a conceptual model of the C compilation process.

Lint

Lint is a source code program that is useful because of the permissive nature of C compilers. It is not part of the C compiler. It checks:

- The type and number of arguments passed in programs against the type and number of arguments declared in called functions (inconsistent typing).
- Declaration vs. usage of objects.

- Whether a function returns a value and, if so, whether or not it is able to return a value from every exit point in the function.
- Whether a calling function uses a returned value of a called function.
- Whether there are unused declared objects and/or code that cannot be executed.
- Whether automatic and register storage-class variables are initialized before use.
- Whether a loop can ever execute or terminate.
- For code portability in the use of bit fields, characters, and assignment of long variables and char variables.
- For suspect construction.
- For correct use of structure pointer variables and operators.
- For correct use of structure member selection operator.
- For archaic syntax.

Several of the checks can be modified by the command line flags or the use of appropriate comments in the source code.

The C Preprocessor

Like Lint, the C preprocessor is not part of the C compiler. It allows source-code file inclusions, and modest macro definitions that allow dynamic, in-line, source-code replacement or modification in preparation for the compile phase. The preprocessor can receive a variety of compiler flags on the preprocessor command line, and, together with a file's preprocessor directives, produces a preprocessed source code. This achieves the effect of conditional compilation without change to the original source code statements.

Compile Phase

The compile phase reads the resulting preprocessor source-code output and compiles the code, producing an object-code file. C allows a program to consist of one or more source-statement files. This introduces object resolution problems when an object in one file must by accessed or, in the case of a function, called. The compiler generates an object-code file with a map of identifiers and the location of every externally defined object with program scope. The identifiers may be altered(truncated and/or case-shifted) to conform to normal system naming conventions. The object file may optionally be placed in an object library with other object files.

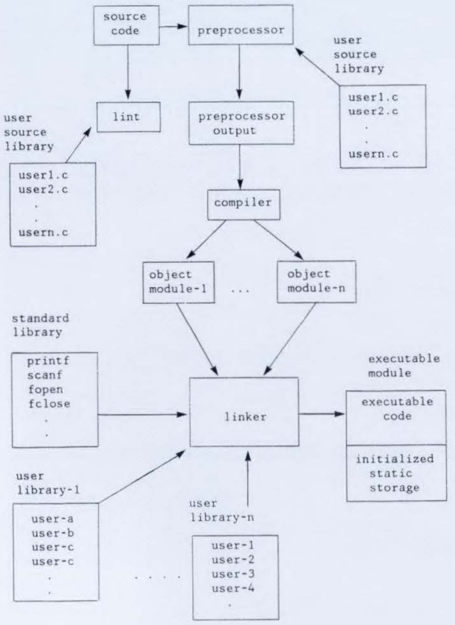


Figure 1.32 Conceptual model of the C compilation process

Link Phase

The linker is usually a system function and not part of the C compiler. It combines all object modules to produce a single, executable run module. Various system and private libraries are searched for various object modules in the order indicated on the linker command line. The search order in which they are examined is important in that it allows a specific version of a library to be found instead of another version that is intended.

The C Non-Standard Library

One of the current impediments for C source-code portability is the incompatibility of the so-called "standard" libraries that are not a part of, but accompany, every C compiler. These incompatible libraries have functions with the same name that behave differently, functions with the same name that behave identically, functions with the same name that behave identically, as well as unique functions found in no other compiler library.

Linkage Implication on Names

The linker is a standard system function and not part of the C compiler. It therefore requires conformance to system naming conventions to accomplish its tasks. As a result, all object file names as well as identifiers of objects having program scope must observe the system conventions as well. External compliance is achieved by the compiler by possible identifier truncation and case shifting as it produces the individual object modules.

Section 2.

Primary Expression Operators

Primary expression operators are operators that are applied to primary expressions such as identifiers, strings, constants, etc. They associate from left to right. Expressions in parentheses are primary expressions also.

Function Call ()

When a primary expression of type **function returning** is followed by the function call operator that contains zero or more expressions separated by commas, the corresponding function is invoked and the enclosed arguments are passed by value.

Note that the function call operator should not be confused with the parentheses separators used, for example, in arithmetic expressions.

Subscript []

When a primary expression of type **array of type** is followed by one or more subscript operators containing integer expressions, the corresponding pointer arithmetic is invoked to produce the appropriate object.

Note that a[b] is always exactly equivalent to (*a+b)).

Struct/Union Pointer >

When a primary expression of type **pointer to struct** or **pointer to union** is followed by the struct pointer operator, which is in turn followed by a member of the type **structure** or **union**, a corresponding instance of the structure/union variable is produced.

The struct pointer can be preceded by an integer constant that is cast to a valid type as indicated:

```
pop = ((struc city_ptr) 0x2F34) ->
      population; /*non-portable */
```

In this case, the integer value points at a memory address. The portability of this technique is nonexistent.

Note that **a->b** is equivalent to **(*a).b.** Moreover, since ***a** is always an **Ivalue**, **a->** is always an **Ivalue**.

Struct/Union Member (Dot Operator)

When a primary expression of type **struct** or **union** is followed by the struct member operator, which is in turn followed by a member of the type structure or union, a corresponding instance of the structure/union member is produced.

The .operator yields an Ivalue if its left operand is an Ivalue.

Section 3

C Preprocessor Directives

Technically, the C preprocessor is not part of the C language. All preprocessor directives are preceded by a #, which should be the first non-whitespace character on the line. For maximum portability, the # should appear in column 1 with no intervening space(s) between it and the preprocessor directive.

As a direct consequence, even though conditional preprocessor expressions can be nested, they should not be indented for portability considerations.

Finally prior to S3J11, the /**/ symbol could be used to concatenate two arguments. As an example:

```
#define smash(a,b) a/**/b
  /* a and b become a single token */
```

previously combined two tokens into one. This practice is now invalid and is replaced with the approach of

```
#define smash(a,b) a#b
    /* a and b become a single token */
```

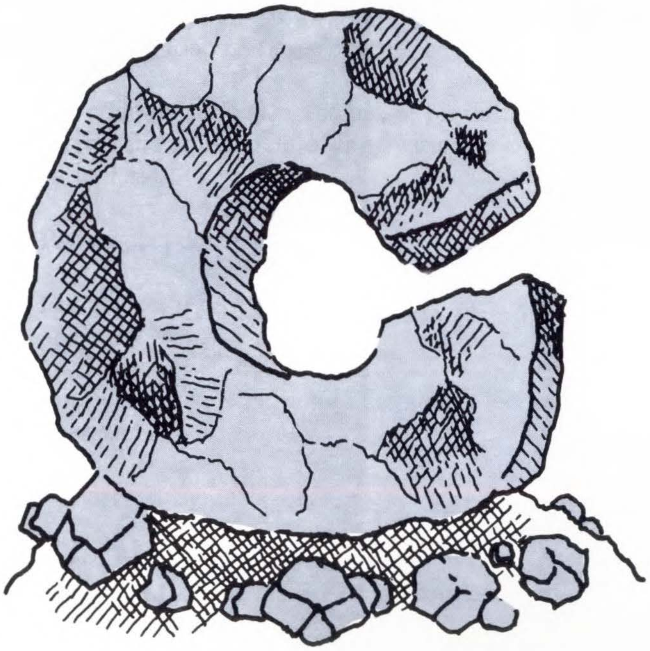
#define

Use

Creates preprocessor macros and preprocessor identifiers with optional, associated, text-replacement strings.

Examples

```
#define DEBUG
#define MSG      "This is a message"
#define macro(a, b, c) ( (a) / ((b) + (c)) )
```



Comments

The #define preprocessor directive is used in three contexts:

1. Defining the existence of a preprocessor identifier.

2. Defining the existence of a preprocessor identifier and associating a character string that the preprocessor substitutes for each subsequent occurrence of the identifier.

3. Defining a macro that the preprocessor expands into C source code.

While the first two uses are subsets of the third, they will be discussed separately for clarity. In all instances, the identifiers are case-sensitive, just as C identifiers are. However, prevailing convention dictates they be completely capitalized to distinguish them from other program identifiers. Such identifiers are generally referred to as "manifest constants". Finally, some preprocessors consider it an error to define a preprocessor identifier that is already defined, while others allow the previous definition(s) to be stacked and reinstated by subsequent corresponding #undef directives.

Defining Preprocessor Identifiers

Preprocessor identifiers are created by a #define directive followed by one or more whitespace characters followed by a unique identifier. The identifier should not be followed by a semi-colon. As an example:

```
/*no semicolon or text follows identifier;
avoid comments on same line */
#define DEBUG
```

defines the existence of a preprocessor identifier DEBUG. You may use identifiers created in this manner in subsequent #ifdef, #ifndef, defined tests and #undef preprocessor directives. In such instances, these identifiers are replaced with a value that evaluates as true (non-zero). Such identifiers are also useful for creating synonyms for basic data types but lack the generality of the typedef keyword. Finally, one may use this type identifier as indicated in the following example:

```
#define THEN  /* no value provided */

   if (i==2) THEN exit ();
 /*THEN is replaced by a NULL
  string & ignored */
```

On can usually create this type preprocessor identifier by invoking the compiler with a -D flag. As an example; this would be equivalent to:

```
cc prog1.c -DDEBUG
```

Note that some compilers force all identifiers created in this manner to upper case. In this case:

```
cc prog1.c -DDEBUG
```

would be equivalent to:

```
cc prog1.c -Ddebug
```

Section 4.

(Non-) Standard Library File I/O Function Portability

File I/O is not part of the C language. File I/O is achieved via library function that accompany each compiler. C compilers were written by a variety of dispersed groups for a spectrum of machines with a variety of hardware architectures, peripheral devices, and operating systems. Therefore, portability problems typically exist for C programs involving file I/O.

C compilers were written ...
for a variety of hardware architectures,
peripheral devices, and operating
systems.

The portability problems also appear to have arisen partly as a result of having compiler run-time libraries attempt to mimic UNIX's low-level I/O functions when the libraries native operating system had different file I/O notions than UNIX, primarily in the area of file structure. Traditionally, the implementers achieved varying degrees of success. The areas that could not be addressed introduced a variety of portability situations.

Symptoms of the portability problem appear as identically named functions providing different services, functions having different names providing identical services, and functions that are unique for given systems. Finally, functions can have identical names, and provide identical services, but still require different formal parameters.

In summary, compiler compatibility in the area of file I/O functions is best described as modest but sharing a UNIX flavor. The following discussion must therefore be viewed as conceptual rather than definitive, though most of it will often apply to any given environment. Your compiler's library is your definitive guide to file I/O.

Standard I/O Concepts

The prevailing perspective of file processing is heavily influenced by UNIX. Consequently, two levels of processing are associated with C library functions, referred to as low-level file I/O and high-level file I/O. In either level, a file must be opened before it can be processed. The manner in which a file is opened determines the types of services and level of processing (high or low) that can be used on the file. Regardless of the level, you should explicitly close all explicitly opened files when their processing is completed. Finally, be sure to include stdio.h in files that use standard I/O functions.

Low-Level (UNIX) File I/O

In low-level file I/O, the associated library functions directly invoke various operating system file services. This level of processing necessarily observes the operating system's view of data files and forfeits a corresponding level of operating system independence in the process. In low-level file I/O, an open file is individually identifiable from other open files by a unique, unsigned integer value, called a file descriptor, assigned to the file by the operating system during the open process. A given file may be opened multiple times and at any time. Each instance of the open file would have a unique file descriptor.

Code written for low-level processing often experiences the least degree of portability and higher degrees of complexity, though it can enjoy a greater level of file processing efficiency and flexibility. As an example, both text and binary files can be processed with this level.

High-Level I/O

In high-level file I/O, library functions invoke the more primitive low-level file I/O services. In high-level file I/O, a given file is individually identifiable from other open files by a unique pointer value called a file pointer. The file pointer points at a unique file-structure variable associated with the file during the file-opening process. This approach realizes a corresponding increase in code portability because of the greater isolation from the operating system environment. It achieves this portability at the expense of file-processing flexibility. As an example, binary files cannot be processed with this approach in some environments. Finally, to use high-level file I/O, a program must include stdio.h that contains the preprocessor definition of EOF, FILE, and NULL which are necessary for this level of processing.

Low-Level Read Write Position Pointers

In high-level processing, a file is regarded as a continuous sequence of bytes, called a stream. Read/write operations typically occur from/to buffers, which are maintained by the high-level file I/O functions. Thus, the routines keep track of where the next character will be fetched from or replaced in the buffer for reads and writes respectively.

In low-level file I/O, the position of the next character(s) within a file to be processed (read or written) is determined by a value, called the file's read/write position pointer, sometimes referred to as a file's character pointer.

The value of a file's read/write position pointer is updated after each file input/output request and reflects the position where the next file read or write will occur. A file's read/write position pointer is conceptually maintained by the operating system. In some instances, in both high and low-level I/O processing, the value is program resettable, allowing a program to provide processing as opposed to what normally occurs when the read/write position pointer is not explicitly repositioned by the program.

Your compiler's library is your definitive guide to file I/O.

Buffered vs. Unbuffered I/O

Low-level file I/O interfaces directly with the operating system services that maintain a file's read/write position pointer. Each low-level, read/write request is accompanied by information for the particular, desired activity as well as an associated data area that receives/holds the corresponding data. One read/write call conceptually results in an immediate physical read/write to the file if the operating system is not buffering the file.

In contrast, high-level I/O services queue data in buffers in anticipation of future requests. In the case of input files, the various requests are applied against the waiting, buffered data until a new request cannot be completely serviced with the data remaining in the buffer. The high-level I/O routines then refresh the exhausted buffer via a low-level I/O request and service the outstanding request with the previously remaining data plus an appropriate amount of new data. In the case of output data, the data are placed in a buffer until it is filled. The high-level routines then invoke low-level I/O routines to write the buffer out to the file.

Note that some systems allow programs to set the size and location of the buffers used in high-level I/O via a setbuf function call. A program may indicate that a buffer size of zero is to be used, in which case no buffering will be done even through the file is processed using high-level processing. If the setbuf function is used with an automatic buffer variable, make sure to close the file and flush the buffer before returning from the owning function. This prevents the system from reusing the buffer area for other processing before the data has been flushed. As an example, consider:

```
#include "stdio.h"
main()
{
    char buf[BUFSIZE];
  /* DANGER ! should be static for safety sake! */
    setbuf(stdout,buf);
            . . . .
}
```

As indicated, buffers should be declared static to avoid random, system-defined failures.

Mixing Levels of I/O on a File

When a file is opened for high-level processing, a unique file pointer is associated with the file as are one or more file buffers. The file pointer points to a unique structure variable used by the high-level services. Because the high-level file-processing support invokes low-level file-function calls, one of the members of the file structure variable is often the file's file descriptor required for low-level processing. Thus, it is sometimes possible to deduce the file descriptor for a file opened for high-level processing.

Because of buffering, individual, high-level I/O requests for sequential file processing do not usually result in immediate, physical reads/writes to a file and subsequent updating of the file read/write pointer. Indiscreet low-level function requests using a deduced file descriptor are invisible to high-level services and can cause the operating system to modify a file's read/write position pointer. If the file position pointer has been disturbed by renegade, low-level calls, data that are eventually read from (or written to) a file by a later, high-level I/O request will not sequentially follow data previously read (or written). Do not mix high-level and low-level service calls on a single open file. Use of deduced file descriptors in explicit, low-level, file-service calls is guaranteed to corrupt subsequent, high-level calls. The results can be ghastly, though well-deserved.

The New STACKS Configuration Command

Ken Allen
Houston Area League of PC Users, Inc.

DOS 3.20 defaults to a maximum of 9 simultaneous hardware interrupts. Many PCs will exceed this limit in the normal course of operation. Exceeding the limit is especially onerous — DOS displays the message "FATAL: Internal Stack Failure. System Halted."

Unless you look up this message on page A-48 in the *DOS 3.20 Reference Manual*, you won't find out about the new STACKS command, which corrects this error. Page A-48 directs you to Appendix B, where the STACKS command is explained.

The new system configuration command STACKS is available for you to include in your CONFIG.SYS file. You enter the command in the following format:

```
STACKS=n,s
```

where

n is the number of stack frames. The default is 9, and the range of values is 8 to 64.
s is the size (in bytes) of each stack frame. The default is 128, and the range of values is 32 to 512.

The manual states that each time an interrupt occurs, DOS allocates one frame from the stack pool. After the interrupt has been processed, DOS returns the frame to the stack pool.

I am using STACKS=16,128 and it seems sufficient for my use, but there are no guarantees. You should be aware that increasing the number of stack frames under DOS decreases the available amount of random access memory.

IBM advises that you not increase the setting until you get the error message. You just have to hope that what you're working on at the time the error occurs can be easily replaced.

IBM User Group Services Update

Gene Barlow
Manager, IBM User Group Support

IBM PC User Group Locater Service

In IBM PC User Group support, we frequently receive requests from individuals who want to join a user group, but don't know the location of user groups in their area. Also, we often get calls from persons who want to communicate information to user groups, but don't know how to find them. Finally, we talk to user groups who are trying to contact other groups and need information about locating them. In all these cases, we are happy to provide information about locating IBM PC User Groups.

We are announcing an IBM PC User Group Locater System that uses the recently announced Voice Communications Adapter attached to an IBM Personal Computer. Anyone with a tone pushbutton telephone can call the User Group Locater System at any time on 305-998-1575 to retrieve information about PC user groups.

When you use the locater system, the computer will guide you to enter requests for user group information. You may search for user groups by entering a telephone area code or postal zip code. The system will respond with the name, city, and phone number of all user groups in that area.

Feel free to try our new user group locater system. If you belong to a PC user group, you may want to verify that your group is included correctly. If the information about your group is missing or incorrect, inform us and we will update our locater database as soon as possible.

Electronic Bulletin Board System

The IBM PC User Group Support Electronic Bulletin Board System (EBBS), located in Boca Raton, Florida, contains hundreds of items of information about IBM PC hardware and software products. It also contains a limited amount of PC software examples and programs.

The system is available to users 24 hours a day, 7 days a week. To allow multiple users to access the system at the same time, the Electronic Bulletin Board System consists of nine IBM Personal Computer XTs connected to a Personal Computer AT using the IBM PC Network.

Everyone is welcome to access the system through the normal long-distance phone line, 305-998-EBBS.

The IBM PC User Group Support Electronic Bulletin Board System can be accessed using most communications programs (such as IBM's Personal Communications Manager, PC-TALK, QMODEM, PC-DIAL, etc.). To use the EBBS:

1. Set your program with the following parameters:
   - 300 or 1200 baud
   - 7 data bits, even parity, 1 stop bit
   - or 8 data bits, no parity, one stop bit (for binary data transmission)
2. Dial 305-998-EBBS
3. Enter your name as prompted.

Once your session has begun, you will be guided through a series of command menus. Each command in a menu may be selected by pressing one of the keys shown in brackets. (Do not press Enter after you press letters shown in brackets.)

From the main menu, you may choose the <I>nfo section or the <F>iles section. The <I>nfo section contains nearly 50 topics, each with files that discuss pertinent information about that topic. For example, after pressing "I", you will be asked to enter a topic. If you are interested in information about the IBM PC Convertible, you would enter CONVRTBL. The CONVRTBL topic contains several information files which you may browse through, selecting those you wish to read.

The <F>iles section contains program files available for you to download. There are several PC Storyboard presentations you may download and run. To use these, download UNPACK.HOW and ST.EXE.

All menus include on-line help (pressing the H key displays explanations of the functions available in a menu). You can terminate the help text by pressing Ctrl-C.

During the next few months we will be significantly updating the database to contain the latest available information about the PC. We will be extracting information from other PC databases available internally within IBM, as well as retrieving information from other user group bulletin board systems across the country. We intend to make the Electronic Bulletin Board System the primary source of IBM PC product information for PC users.

What Print Spoolers Can Do For You

Rich Chambers
Idaho IBM-PC Users Group

If you are new to computing, you may be asking "What is a print spooler, and why should I be interested in it?" Simply stated, a print spooler lets you use your computer for other things while you are printing documents on your printer.

An excellent print spooler comes with IBM's DOS package. It is called PRINT.COM, and it is on the main DOS diskette. It is a very flexible utility, allowing you the option of printing one or many documents at a single sitting. Because I use DOS 2.10. my reference will be to that version. (There are some enhancements to this program in later versions of DOS.)

A typical use of the PRINT.COM utility is this:

```
A> PRINT newsltr.doc
```

This command sends the file newsltr.doc to the printer. Once your document begins printing, you can simultaneously do other things on your computer.

Suppose you have a series of files that you want to print while working on something else. If all the files had the same extension, you could type:

```
A> PRINT *.doc
```

This routes all the files in the current directory having the extension .doc to the printer. The wildcard characters * and ? may be used freely.

If you want to print the files in a particular sequence, you can specify this too:

```
A> PRINT file1 file2 file3
```

where "file1 file2 file3" are the filenames and extensions of the files you wish to print. Up to ten filenames can be selected in this manner, and the files are "queued" to the printer.

Note that if you use diskettes rather than fixed disks, you may not remove the diskette that holds queued files until all selected files have been printed. But if you are using a hard disk, you can move to a different directory after you have selected the files to print. The utility returns to the right directory to find the files you have queued.

To see which files are presently in the print queue, type PRINT.

To add more files to a previous print queue, type:

```
A> PRINT newfile1 newfile2
```

where "newfile1 newfile2" are the filenames and extensions of the files to be added.

As with other DOS utilities, PRINT lets you include several parameters when you access it. Using these parameters, you can cancel any file previously placed in the queue; place more files in the queue; or terminate all print jobs.

The parameters are /P, /C, and /T. They are easy to use. For example, to terminate all printing type:

```
A> PRINT /T
```

A termination message is printed to the screen, the printing stops, and the paper is advanced to the next top-of-form.

To cancel one previously issued print job, type:

```
A> PRINT file2/C
```

This deletes file2 from the print queue. Other files in the queue are not affected.

You can use more than one parameter in a single print statement. For example:

```
A> PRINT file2/C file5/P
```

This cancels file2 from the print queue and adds file5 to the queue at the last position.

Things to remember:

- DOS 2.10's PRINT command does not allow you to use PATH commands to select files for the print queue. You must change to the directory where the files to be printed reside, and select them at that time. Once you have made your selections, you can proceed to a different directory.
- Many processors save their files in formats other than ASCII, and their print utilities accomodate the different formats. PRINT.COM knows nothing of this, and attempts to print out the files in exactly the format they have on the disk. However, many word processors can print a file to a disk instead of to the printer. If you first print a file to disk, PRINT.COM should print the output you expect.
- If you have a dot-matrix printer, and want to print your files in a particular font, you must select the printer font prior to running PRINT.COM.
- Don't expect the speed of printing you normally see. PRINT.COM uses just a small portion (or slice) of your computer's operating cycles to send characters from the files you have selected to the printer. Normally, only about 512 characters are sent to the printer during this slice. While you will notice a considerable degradation in your printer's normal throughput time, the program running in the foreground will be minimally affected. When you finish using the application in the foreground, you can run another application while PRINT.COM continues in the background. Thus, printing numerous documents need not keep you from being productive.

The DOS PATH Command

Ethan Winer
Connecticut IBM PC User Group

When you enter a program name at the DOS prompt, DOS normally looks for the program file in the default directory on the default drive. If DOS does not find the file in the default directory, you are given an error message: "Bad command or filename."

By using the PATH command, you can tell DOS ahead of time, "If you can't find the specified program file (*.COM or *.EXE) or BATch file (*.BAT) in the default directory, here's a list of places I want you to look."

At the DOS prompt (or from within a BATch file), enter the PATH command and a series of drive and directory specifiers, each separated by a semicolon. Be sure not to include any blank spaces.

```
PATH=A:\;B:\
```

Then, whenever DOS receives a call for a program or BATch file, it will look for it first in the default drive, then in the A drive, and finally in the B drive. If DOS cannot find your specified program file in the default directory or in the directories specified in the PATH command, you will still get the error message, "Bad command or filename."

The PATH command is extremely useful on a fixed disk system that is organized into many separate directories and subdirectories.

I organize my own hard disk with each application in a separate directory. This makes it easier for me to find the files I'm looking for.

When I type DIR on a fixed disk, the computer displays this:

```
Volume in drive C is WINER
Directory of C:\

COMMAND  COM       17792    10-20-83  12:00p
CONFIG   SYS          29    08-04-85   4:10p
AUTOEXEC BAT          15    08-04-85   4:00p
DOS          <DIR>          08-04-85  11:02a
LOTUS        <DIR>          08-06-85  10:23a
WRITE        <DIR>          08-12-85   9:19a
BASIC        <DIR>          09-03-85   3:08p
FINANCE      <DIR>          09-17-85  11:17p
UTIL         <DIR>          09-22-85  10:46a
BATCH        <DIR>          10-05-85  12:14p

   10 File(s)      4823072 bytes free
```

This list does not display many of the programs and files on this disk, but I can find them easily. Each entry marked <DIR> is actually a complete directory that contains its own group of programs and files.

Without BATch files and the PATH command, the only way to load a program would be to remember which directory it is in, switch to that directory, and enter the command at the DOS prompt. The more files I have, the more involved those directories can become.

Notice the last directory called BATCH. I have several BATch files in that directory, one for each of the programs I use most frequently. The BATch files contain all the necessary commands to switch to the appropriate directory and call the desired program files. I simply set a PATH to the BATCH directory, and whenever I enter the BATch file name at the DOS prompt from any drive or directory, DOS finds the correct BATch file and executes the commands specified in it.

For example, if I happen to be in the LOTUS directory, and wish to load my word processor, I type WRITE. My path is set in the AUTOEXEC.BAT file to C:\BATCH;C:\DOS;C:\UTIL. DOS will search the default directory (LOTUS) first, and, not finding any program or BATch file named WRITE, it will search the BATch directory. There it will find a BATch file named WRITE.BAT that contains the following lines:

```
ECHO OFF
CD C:\ROC
WRITE
```

By using the PATH command, you can tell DOS ahead of time... "Here's a list of places I want you to look"

Even though I entered the WRITE command from within the LOTUS directory, DOS executes these commands, properly changing to the specified directory and loading pfs:write.

Notice I have my PATH set to the DOS and UTIL directories as well as the BATch directory. This enables me to have all my DOS files in a separate directory, yet I can call any DOS command from within any other directory or drive without having to specify a drive and path when I enter the DOS command. This is also true of the utilities programs in the UTIL directory.

There is only one limitation to the PATH command — it will not search for data files, only program (*.COM and *.EXE) files and BATch files (*.BAT). If I tell pfs:write to load my latest letter to Mom, and it isn't in the current directory, DOS will not search the directories specified in the PATH to find the letter.

But it's actually better that way. Suppose I could retrieve data files using a PATH. When I was done editing and went to save my letter, it would be saved in the current directory. (It's just too much to expect DOS to remember where it came from originally.) This would create two versions of mom's letter in two different places, and only one of them would be the latest version.

However, some programs, like older versions of Wordstar, make calls to overlay files that are not actually programs. If you are not in the directory that contains both the program file and the overlay files, DOS will not go out and find the overlay files when the program calls to them. The solution is to have one directory that contains the program and overlay files as well as your data files. Then use a BATch file to change to that specific directory whenever you wish to load the program.

Should you ever wish to reset the PATH command to search in only the default directory, just enter the PATH command followed by a semicolon:

```
PATH;
```

The PATH command is simple, but it allows you to organize your programs and files into several directories while still being able to access your programs from anywhere on the disk.

Adding to Your PATH

Mark Chance
IBM Corporation

(Editor's note: This method of adding to your Path is **not documented** *in the DOS Reference manual and is* **not supported** *by IBM.)*

When processing a BATch file, DOS (version 2.00 and higher) has an interesting way of accessing the information you have placed in the environment space using the DOS SET command. If you use the SET command from within a BATch file to set XXX to YYY:

```
SET XXX=YYY
```

and then you enter the XXX variable from with another BATch command as shown below:

```
ECHO %XXX%
```

then DOS returns YYY.

Using this information, you can create a BATch file that will add a single directory to your existing PATH statement without your having to re-enter each directory currently in the PATH. First, you create a BATch file (I named mine ADDPATH.BAT) that contains the following command:

```
PATH %1;%PATH%
```

If you create this BATch file using the COPY CON: command, type the lines shown below from the keyboard. The word "enter" in brackets means you press the Enter key at the end of the line; the F6 means you press Function Key 6.

```
COPY CON: ADDPATH.BAT  <Enter>
PATH %1;%PATH%  <Enter>
F6  <Enter>
```

Suppose your current PATH contains the following directories:

```
PATH C:\;C\DOS;C:\UTIL;
```

And you wish to add the directory C:\APPS to the path. You simply enter the following command at the DOS prompt:

```
ADDPATH C:\APPS
```

The PATH would become

```
PATH C:\APPS;C:\;C:\DOS;C:\UTIL;
```

If you wish to add the directory to your path temporarily, you might consider loading COMMAND.COM as a second level command processor before you enter the ADDPATH statement:

```
COMMAND
ADDPATH C:\APPS
```

The new path will be valid as long as you operate under the second level command processor. When you enter "EXIT" and return to the first level command processor (loaded when you booted DOS), the original PATH would again be valid.

Taking Photos of Your Screen

(Editor's note: Two authors contibuted separately to this article.)

Larry Harvey
San Diego Personal Computer User Group

It is relatively easy to take pictures of a monitor screen. If you take a few precautions when setting up your camera, special lenses, and monitor, you can produce excellent print negatives and slide transparencies, but very good results can be obtained without special lenses, extenders, or auxiliary equipment designed to make prints or slides from the screen.

Although I use several types of cameras, including an ancient but venerable Speed Graphic 4x5 and an equally old Zeiss Ikon 120 roll film camera, my principal camera is a 35mm Nikon F2. I find it helpful to use a telephoto lens and a small extender, because they permit five feet of working space between the screen and the lens.

For black-and-white slides, I use Tri-X film rated at 1000, an F-stop of f5.6, and an exposure time from 1/25 to 1/100 second. I develop the negative as a reversed slide in ACU-1 Acufine working solution of one-to-five for 8 minutes at 75 degrees F. I have found a 1/50 second exposure best, with 1/100 second passable. Longer exposures produce bleary text in the slide, and shorter exposures (less than 1/25 second) produce very light lines in the lettering.

For color slides, I use Kodak Ektachrome 64 (rated at 64), an F-stop of f5.6, and an exposure time of 1/4 to 1 second. My subject for one such shot was the color bar screen from DOS 1.10. All the slides were passable, with the 1/2 second exposure appearing to be best. I develop the Ektachrome film commercially, however, you can process your own color film with Kodak E6 chemicals if you wish.

For light, I use only the monitor screen itself, and I take the shots at night with room lights off. Street light coming through the window is not a problem. However, the adjustment of the screen intensity is somewhat critical. I prefer a white foreground with black background. Using the intensity control, I adjust the background until it begins to change to gray. I then rotate the intensity control counterclockwise one quarter turn. The screen adjustment when displaying color is not as critical. The program's normal intensity control settings are adequate.

In setting up the screen's content to be photographed, I find a 40-character line works best for presentation purposes. Commercially available graphics programs also have worked well in shooting presentations. I have shot a program that displays various color drawings from its data disks for making presentation slides. Colored foregrounds with black backgrounds work well when you use double exposures to place text inside the color presentation "frames" provided by the drawing data disks.

If you generally follow the steps outlined above, you will be able to produce slides at a relative low cost. And if you are handy with photochemicals, you can design, display, shoot, and process your slides in less than one day.

Larry D. Goss
North East Indiana Users Group

As an amateur photographer, I would like to contribute my list of rules for photographing images from either the monochrome or color monitors.

- Use a single-lens reflex camera. You want to frame the video screen as accurately as possible to make the slides appear as professional as you can. A single-lens reflex camera avoids the parallax problems of viewfinder and twin-lens reflex styles, so you come closer to a "what you see is what you get" situation. For most computer enthusiasts, this means that you will be using a 35mm camera.

  Just using a 35mm camera does not solve all your framing problems, though. The problem is that the aspect ratio (height: width ratio) of the camera is 2:3 and the aspect ratio of the video screen is 3:4. This makes it tough to frame a full screen on the computer without getting some of the bezel of the monitor in the picture. The solution is to plan your screen displays with this in mind. Leave plenty of border on the display so you can frame the picture without including the bezel. Some of the older color monitors have a nice wide border that makes framing a full screen of text relatively easy.

- Use a tripod. The exposure times and framing are going to be critical enough that handheld shots will not be adequate.

- Use a medium telephoto lens (if you have one). A 135mm lens is about right. This is the length of lens that is often used for portraits. A lens any longer than this has problems focusing at a close enough distance, and a normal (50mm) lens starts to show a lot of barrel distortion because of the curvature of the display screen.

- Photograph in a darkened room. The reflections on the screen that you normally ignore when you're using your computer become very distracting when they show up in a photograph.

- Set the display for normal viewing. Turning the contrast and brightness way up or down only confuses the issue of determining the exposure for the picture. Look at the pixels with a magnifying glass if you need to, but set the controls so that the pixels don't bloom or bleed into each other. If your monitor is already blurring the letters on the display, you won't be able to correct the situation in the photograph.

- Aim the camera straight at the center of the viewing screen. This will eliminate "keystoning" of the image. Do this with the lights on in the room. The reflection of the camera that you see in the video screen can be used as a guide for determining when the camera is centered and set up perpendicular to the glass surface.

- Get a reading for exposure with the screen full of text and with the color of background that you will be using for the ensuing shots. Keep the shutter speed slow; never faster then 1/30 of a second. Any speed shorter than this can result in streaked images because of the refresh/flicker frequency of the video display.

- Close down the aperture (f/stop) by one number from the "normal" exposure setting that the camera or light meter tells you to use. This compensates for the averaging effect that normally occurs with through-the-lens metering, and keeps lettering and raster linework from "blooming". It also gives a better depth of field to keep all portions of the video screen in better focus.

- Once you have set the camera exposure, leave it alone regardless of how much text is on the screen being photographed. Check the exposure if you change background colors, but don't change it on the basis of how full the screen is.

  If you have a camera with automatic exposure, turn off that function for this type of photography.

Hardware

IBM 3117 Scanner, Adapter/Extension

The 3117 Scanner is a desk-top, flat-bed image scanner that has a small footprint, low noise and heat generation, and is lightweight. It scans images up to letter size. Its flat-bed design facilitates scanning a page in a bound document, part of a large page, or paper of odd size, shape or paper weight. The scanner scans a letter size (8-1/2" x 11") page in approximately 30 seconds.

Partial page scan is provided under program control. One or more rectangular areas can be defined for selective scanning within a page. Through use of this function, unnecessary information can be eliminated, reducing the image data volume. You can control darkness, contrast, and halftones for each rectangular area.

The scanner has a resolution of up to 240 x 240 pels per inch. Horizontal and vertical scanning resolution are program selectable and can be:

- 240 x 240 pels per inch
- 240 x 120 pels per inch
- 120 x 120 pels per inch

You specify darkness control through selection of an appropriate threshold level via program control, enabling the scanner to scan a darker copy, a lighter copy, or a colored original document.

Digital halftones is provided to improve the quality of scanned pictures by emulating grey levels. Up to 16 levels of digital halftones can be supported. Digital halftones makes use of a matrix of 4 x 4 pels to determine the grey level of the scanned image.

The 3117 Extension Unit makes use of an Intel 80186 microprocessor to provide additional image processing power in the scanner. The extension unit provides additional capabilities to the 3117, including:

- Attachment to an IBM PC via various general purpose communication adapters (as described below)
- Image compression in the scanner
- Contrast control

Contrast control is achieved using dynamic thresholding and edge enhancement functions of the Extension Unit. These functions are useful for capturing images with various quality levels, especially for scanning an image where there is low contrast between the image and its background. The functions also make the edges of characters or lines look sharper and crisper. The edge enhancement is especially useful when the image consists of pictures and characters using digital halftone. Contrast level is selectable under program control.

The 3117 Scanner attaches to the IBM Personal Computer, IBM Personal Computer XT, IBM Personal Computer AT, IBM 3270 PC, or IBM 3270 PC AT, and operates with the Image Support Facility 2 licensed program. Hardware requirements vary by attachment method:

- Attachment via the 3117 PC Adapter card
  - An IBM PC with a minimum of 256KB main storage (512KB for 3270 PC family member)
  - A 3117 PC Adapter on the PC (requires full card slot)
  - A 3117 PC Cable
- Attachment via the 3117 Extension Unit
  - An Extension Unit on the 3117
  - An Asynchronous Communications Adapter, Serial/Parallel Adapter, or High Speed Adapter on the IBM PC
  - A Communications Adapter Cable for the IBM PC, PC XT, or 3270 PC or a PC AT Serial Adapter Cable for the IBM PC AT or 3270 PC AT.

Only one 3117 can be attached to an IBM Personal Computer.

IBM 3118 Scanner/High Speed Adapter

The IBM 3118 Scanner is a high-quality, compact, desktop, medium-speed, feed-through page scanner with an average scanning speed of 12 seconds per page and up to 240 x 240 pels per inch resolution.

The High Speed Adapter provides a 1 million bit per second RS-422 interface. It is designed to operate with the 3118 Scanner and to maximize the 3118's performance.

It scans pages of various sizes, from widths of 2.6 inches to 8.5 inches, and from lengths ranging between 3.9 inches and 14 inches, with an adjustable paper guide. It scans paper that weighs between 13 and 50 pounds.

Horizontal and vertical scanning resolution is program-selectable, and may be any of the following:

- 240 x 240 pels per inch
- 240 x 120 pels per inch
- 120 x 120 pels per inch

The scanner can scan a letter-size page in an average of 12 seconds at 240 x 240 pels per inch resolution and with compression. The speed may vary depending on the paper size, characteristics of the image, the application programs, or communication speed. Scanning speed will be slower when communicating with an IBM Personal Computer via the Asynchronous Communications Adapter or the Serial/Parallel Adapter, since these adapters are limited to 9600 bits per second.

The scanner takes advantage of the increased processing power of the high performance Intel 80186 microprocessor to handle such functions as compression, digital halftoning, darkness and contrast control, dynamic thresholding, edge enhancement, partial page scan, and others.

Compression of the image data is performed in the 3118 using the Modified Modified Read (MMR) compression method. This technique results in an average compression ratio of 27 to 1 at 240 x 240 pels per inch resolution.

Note: Compression ratio will vary and depends on image content characteristics.

The 3118 Scanner gives you partial page scanning capability. This function lets you define one or more rectangular areas to be scanned within a page, and helps eliminate unnecessary image data volume transmission and storage requirements. You can control darkness, contrast, and halftoning for each rectangular area individually.

The High Speed Adapter can be installed in an IBM PC, PC XT, PC AT, 3270 PC, or 3270 PC AT. It provides a 1 Mbps RS-422 interface which maximizes the scanning performance of the attached 3118 Scanner.

A 3118 Scanner attaching to an IBM Personal Computer is designed to operate with the following hardware:

- An IBM Personal Computer, PC XT, PC AT, 3270 PC, or 3270 PC AT System Unit
- An Asynchronous Communications Adapter, Serial/Parallel Adapter, or High Speed Adapter in the PC system unit.
- A Communications Adapter Cable, for the IBM PC, PC XT, and 3270 PC, or a Serial Adapter Cable for the PC AT, and 3270 PC AT.

Note: When attached to a 3270 PC family member, use of the High Speed Adapter is strongly recommended.

The High Speed Adapter is designed to operate with:

- A full-size slot in the PC system unit.
- An IBM Monochrome Display, IBM Color Display, IBM Enhanced Color Display, or IBM 5272 Color Display on which to run diagnostics.

A 3118 Scanner attached to an IBM Personal Computer will be supported by the Image Support Facility 2 licensed program.

- Only one 3118 Scanner can attach to an IBM Personal Computer.
- The Expanded Memory Adapter (XMA) for the IBM 3270 Personal Computer family is not supported.
- For System/36 or System/38 attached PCs the Asynchronous Communication Adapter speed should not exceed 9600 bps.

IBM InfoWindow Display

The InfoWindow Display is a 13-inch dual frequency display that is capable of accepting information in both RGB and NTSC composite video formats. This lets you mix text and graphic information from the personal computer with still or motion video and sound from up to two OEM videodisc players. The information can be presented separately or intermixed (text and graphics over video) on a single screen.

The display provides two frequencies switchable under program control. A 640 x 350 mode provides higher quality for viewing of text and graphics. Video images or text and graphics or a combination of text and graphics and video can be viewed in 640 x 200 mode.

To control the presentation, you use the touch-sensitive display screen, capable of allowing up to 60 programmable touch areas. The PC keyboard also can be used for input if desired. The screen can display a full range of colors for video presentations and a choice of up to 16 colors selectable from a palette of 64 colors for any text and graphics screen.

Two stereo speakers imbedded in the display provide audio output from either a synthetic speech card containing 218 prerecorded words and sounds in the display, the audio tracks from the videodisc player, or an external audio source. Additional words can be read from a PC diskette or fixed disk to extend the vocabulary. You must record, digitize and store words to be used for this external speech function. An audio jack on the front of the display allows the option of using a customer-supplied headphone. When the headphone is used, the imbedded stereo speakers are disabled. In addition, you can supply an external stereo amplifier and speakers, an auxiliary composite video display and/or a projection screen to be attached to augment the capabilities of the InfoWindow Display.

The IBM InfoWindow Display requires one copy of the IBM InfoWindow Control Program that includes an Application Program Interface operating under DOS 3.20 on an IBM Personal Computer.

Laser videodisc players that may be attached to the InfoWindow Display are available from original equipment manufacturers (OEMs). The following videodisc players or their functional and interface equivalent have attachment interfaces capable of providing the audio and video input to the system:

- Pioneer LD-V6000 (RS-232C serial interface)
- Pioneer LD-V6000 with SWSD Processor SS-D1 (sound over still)
- Pioneer LD-V6200 (IEEE-488 parallel interface)

Attachment of a compatible videodisc player can be through either the RS232 serial interface port or the IEEE 488 parallel interface port. Both ports are available at the rear of the display. However, only one port can be used at a time. If two videodisc players are required, both must be attached via the IEEE 488 port. Both players must be of the same make and model.

Attachment cables for videodisc players are not available from IBM and should be obtained from the videodisc player manufacturer.

A nine-foot power cord, and a Display Tilt and Swivel Adapter are available for the IBM 4055 InfoWindow Display.

Enhanced Graphics Adapter Jumper Card

The Enhanced Graphics Adapter Jumper Card supports the IBM InfoWindow Display. It provides an external synchronization from the display and enables the graphics stored on an IBM Personal Computer to overlay the video stored on a videodisc player. The IBM Enhanced Graphics Adapter Jumper Card is installed on the feature connector of the IBM PC Enhanced Graphics Adapter and connects to the IBM PC General Purpose Interface Bus Adapter.

Software

PC Image Document Utility (PC/IDU)

Personal Computer Image Document Utility (PC/IDU) provides image handling facilities for documents captured by the IBM 3117 Scanner, IBM 3118 Scanner and IBM Scanmaster 1 (8815) when they are attached to the IBM Personal Computer family of products.

PC/IDU provides image print support for the IBM Proprinter, the IBM Graphics Printer, the IBM Quietwriter 2, the IBM Color Printers and the IBM Scanmaster 1 (8815).

Image documents can be displayed on the Personal Computer family of products using the Color/Graphics Adapter, the Enhanced Graphics Adapter, and the 3270 PC equipped with PC graphics capability. Image documents can be filed on PC diskettes or hard files.

- Display functions provide scrolling, rotating and magnifying of image pages.
- Limited editing of image pages (clip, extract, erase, reverse, annotate) is supported.
- Image documents which are filed in compressed MMR format or binary raster format can be displayed.
- Compressed image documents created by the host based Image View Facility Program Offering (5785-ECX) and transferred to the PC are supported.
- Compressed image documents created on host systems with the new IBM 3117 and 3118 scanners and transferred to the PC are supported.
- Transfer of PC image documents to/from IBM System 370 processors, IBM System 36/38 processors and other Personal Computers is supported using announced PC file transfer licensed programs.

PC Image Document Utility operates on the IBM PC, PC XT, PC AT, 3270 PC and 3270 PC AT under DOS 2.10 or later. PC/IDU requires Image Support Facility 2. Minimum configurations require 384KB or storage, one double-sided diskette drive, and one 10MB fixed disk.

Image Support Facility 2

Image Support Facility 2 assists programmers in developing image application programs running on the IBM Personal Computer, IBM Personal Computer XT, IBM Personal Computer AT, IBM 3270 Personal Computer, and the IBM 3270 Personal Computer AT by providing an Application Program Interface (API). Image Support Facility 2 provides basic image functions and device support which are prerequisites for image application programs. Programmers can write application programs that handle image information on the IBM Personal Computers without concern for details of image processing algorithms, or for the techniques and operating characteristics of image input/output devices. Image Support Facility 2 provides callable functions, and is a prerequisite for end users using an image application program developed using Image Support Facility 2.

The Image Support Facility 2 supports image manipulation functions in the IBM Personal Computer such as move, copy, rotate, scale, reverse, extract, erase, mirror, and paint image data. It also supports image conversion which allows the scanned and compressed image to be stored or sent to other devices with different resolution.

Image Support Facility 2 provides the means of capturing image information via the IBM 3118 Scanner or the IBM 3117 Scanner attached to an appropriately configured IBM Personal Computer.

The Image Support Facility 2 supports the following IBM Personal Computer displays: IBM Monochrome Display with Extended Graphics Adapter (EGA), IBM Color Display, IBM Enhanced Color Display, and IBM 5272 Color Display (3270 Personal Computer).

The Image Support Facility 2 supports image printing on the following printers: IBM Graphics Printer, IBM Color Printer, IBM Proprinter, IBM Quietwriter Model 2, and the IBM Color Jetprinter.

Image Support Facility 2 supports application programming in the following programming languages and provides language bindings for:

- IBM BASIC Compiler 2.00 (operating with DOS 3.10 or higher)

- IBM Macro Assembler 2.00
- IBM C Compiler 1.00
- IBM Pascal Compiler 2.00

Image Support Facility 2 is designed to operate on one of the following systems with minimum memory of 256KB (512KB for an IBM 3270 Personal Computer family member):

- IBM Personal Computer
- IBM Personal Computer XT
- IBM Personal Computer AT
- IBM 3270 Personal Computer
- IBM 3270 Personal Computer AT

The 3270 PC and 3270 PC AT require 512KB of memory and do not support the Expanded Memory Adapter (XMA).

At least one double-density diskette drive or high-capacity diskette drive is required to run Image Support Facility 2. An additional diskette drive or a fixed disk drive, that can hold image data is required for image manipulation.

Image Support Facility 2 is designed to operate with IBM Personal Computer DOS 2.10, 3.10, or 3.20.

Image Support Facility 2 supports the following input/output devices:

- Scanners
  - IBM 3117 Scanner
  - IBM 3118 Scanner
- Printers
  - IBM Graphics Printer
  - IBM Color Printer
  - IBM Proprinter
  - IBM Quietwriter Model 2
  - IBM Color Jetprinter
- Displays
  - IBM Monochrome Display with EGA
  - IBM Color Display with Color Graphics Adapter (CGA)
  - IBM Enhanced Color Display with EGA
  - IBM 5272 Color Display (3270 Personal Computer)

IBM InfoWindow Control Program

The IBM InfoWindow Control Program licensed program product provides an interface that enables the user to access a wide range of multimedia functions available through the IBM InfoWindow system. The IBM InfoWindow Control Program executes in the IBM Personal Computer (PC) and consists of two parts: the Application Program Interface, which handles commands from an application, transmits the commands to the IBM InfoWindow system, and provides response messages to the application; and the Language Interface, which enables the user to communicate with InfoWindow in different programming languages. Three languages are currently supported: IBM Compiled BASIC Version 2.00, IBM PC C Version 1.00, and Lattice C Version 2.15. Interfaces to other languages can be added by the user. User applications can include authoring, administration and presentation systems, as well as courses, product information programs and other applications.

With the Control Program's Application Program Interface (API), an application can access the multimedia function of InfoWindow through a command-oriented language. Authoring systems using this interface can simplify the task of creating presentations by providing prompting screens to solicit specific course and application design input.

Through the API, users can control up to two players and perform functions such as showing still frames, showing motion sequences, obtaining videodisc status information, controlling two channels of audio, and playing stored audio for a single frame (still frame with sound). A pass-through mode is provided for direct programming of player-unique commands.

Users can define up to 60 touch areas for each screen; the dimensions of the areas and numbers of areas can vary for each screen. The system provides touch data output as row-column coordinates, X/Y coordinates, and key numbers (indicating the user-defined area touched). Two tones of confirmation beeps are available.

The IBM InfoWindow Control Program requires an IBM InfoWindow Display and associated cables, and an IBM PC. The personal computer must be the IBM Personal Computer, IBM Personal Computer XT, or IBM Personal Computer AT. The PC must contain a minimum of 256 KB of memory, and must contain at least one 5.25" diskette drive. Required feature cards for the PC are the Enhanced Graphics Adapter, Enhanced Graphics Adapter Graphics Memory Expansion Card, General Purpose Interface Bus, and Enhanced Graphics Adapter Jumper Card.

The IBM InfoWindow Control Program requires DOS 3.20. Authoring systems, courses and other applications using the Control Program may have additional software requirements.

The IBM InfoWindow Control Program is compatible with DOS 3.20. The API is also compatible with the following applications provided by IBM:

- Video/PAssage Author and Video/PAssage Presentor program products
- InfoWindow PILOT - Authoring and InfoWindow PILOT Presentation programs
- Exploring the IBM InfoWindow System, an interactive videodisc with diskettes.

See the section of the IBM InfoWindow display for Laser videodisc players, which may be attached to the Display.

IBM InfoWindow PILOT

IBM InfoWindow PILOT is a command-type procedural authoring language for writing and presenting interactive video instruction on the IBM InfoWindow Display. Developed exclusively for the education and training environments, InfoWindow PILOT enables teachers, industry trainers or CAI authors to write a variety of interactive lesson modules and simulations. The program includes a text editor and a full-screen graphics editor and fully supports videodisc and touch-screen interface.

- Provides extended answer analysis for matching free-form answers up to 32 characters in length or for matching a series of anticipated responses
- Includes a graphics editor for all-points-addressable graphics, full color control, and combinations of text and graphic images
- Permits multiple text windows, graphics overlays, pull-downs, and pop-ups
- Allows animation and movement of text on the screen
- Includes a text editor for 80- and 40-column screen display
- Accepts answers in the form of equations
- Permits storage of student input and maintenance of student record files
- Supports synthetic ROM speech
- Provides sound generator control for music instruction or composition of simple melodies
- Lets you include assembler language programs
- Consists of easy-to-learn commands
- Provides full control of videodisc
- Supports touch-screen of IBM InfoWindow Display with 60 touch-sensitive areas
- Has the facility for saving student responses and creating a scorekeeping module
- Provides student "notepad" capabilities

PC DOS 3.20 and the IBM InfoWindow PILOT Program runs on either a PC XT or PC AT, connected to the IBM 4055 InfoWindow Display and optionally to a videodisc player. Additional requirements are Enhanced Graphics Adapter (EGA), Graphics Memory Expansion Card, General Purpose Interface Bus (GPIB) Card, GPIB Cable and EGA Jumper Card. Minimum configurations require 256K of memory, one double-sided 5.25" diskette drive, and one fixed disk drive.

Video/PAssage Authoring/Presentation System

The Video/PAssage Authoring and Presentation System consists of two programs forming an environment that lets you create a presentation composed of media elements and information. The Video/PAssage Authoring and Presentation System, combined with a new IBM touch sensitive intelligent display called InfoWindow, lets you define, test and run presentations consisting of various media elements. The presentations authored with Video/PAssage run on an IBM Personal Computer configured with an InfoWindow Display and up to two optional original equipment manufacturer (OEM) videodisc players.

Video/PAssage Author lets you define a presentation that may consist of video, audio, graphics, text, and synthetic speech elements. It uses a time-line concept to control and synchronize these elements. From the author's input, Video/PAssage Author produces an output file containing the control information for the execution of the presentation.

Video/PAssage Author supports the concept of team authoring by facilitating the organization of a presentation in an outline, and by providing the capability to integrate the various segments of the presentation through the presentation management function. In this way, a team of authors can work together in the creation of large and complex presentations. The product contains several authoring aids designed for ease-of-use to enhance the productivity of the author. The authoring system is menu driven with step-by-step prompts and pop-up windows. The author can choose from a number of IBM supported text and graphics editors.

The author has the capability of presenting video segments, video stills, audio, graphic frames, text and synthetic speech. With the flexibility of the PC and the attachable hardware devices, it is possible for the author to use media mixing to produce a variety of visual and audio effects. Examples of media mixing are:

- Graphic frame(s) over motion video with stereo sound
- Graphic frame(s) with audio from videodisc
- Graphic frame(s) over video still(s) with sound from speech chip
- Motion video and sound from the videodisc
- Video from one player, audio from second player
- Sound over video still frame.

Video/PAssage Presentor requires:

- IBM Personal Computer XT or IBM Personal Computer AT
- At least 512KB of memory for DOS, InfoWindow Control Program, and program execution
- Additional 128KB of memory if using PC/VTXACCESS for NAPLPS decoder support
- An IBM InfoWindow Display
- An IBM Enhanced Graphics Adapter and Graphics Memory Expansion Card with 128KB of graphic storage
- An IBM Enhanced Graphics Adapter Jumper Card and a GPIB adapter
- One 10MB fixed disk drive or larger
- One double-sided 5.25" diskette drive or one 3.5" diskette drive.
- IBM DOS 3.20
- IBM InfoWindow Control Program
- IBM PC/VTXACCESS 1.0 (optional) for NAPLPS decoder support

In addition to the above, Video/PAssage Author requires:

- At least 512K bytes of main storage for DOS, ISPF/PC EZ-VU Runtime Facility, InfoWindow Control Program, and program execution
- IBM ISPF/PC EZ-VU 2.0 Runtime Facility

See section on the IBM InfoWindow Display for laser videodisc players that are supported.

Video PAssage is compatible with the IBM InfoWindow Display and the InfoWindow Control Program. In addition, it is compatible with the following products:

- IBM PC Palette 1.00
- IBM PC Storyboard 1.10
- IBM PC Paintbrush 2.80
- IBM VTXGRAF 1.00
- IBM Macro Assembler 2.00
- Lattice 8086/8088 C Compiler 2.15
- IBM Personal Editor 1.00
- IBM Professional Editor 1.00

DisplayWrite 4

DisplayWrite 4 provides full-function text processing capabilities in a streamlined interface and reference format. It features enhanced document creation, editing, and revision capabilities, and includes an extensive Help facility. Data import, voice/audio support, document interchange and print support are included.

DisplayWrite 4 supports the creation, revision, pagination, merging, storing, and printing of general correspondence, multi-page documents, technical documents, and statistical tables. DisplayWrite 4 utilizes a simplified menu which includes small pop-down menus that occupy only part of the screen, allowing text to be visible at the same time.

Users new to word processing can easily become familiar with DisplayWrite 4 through the simple menu format and the on-line Help facilities. Help includes information about keys, menus, commands, and program and printer functions.

Experienced users of DisplayWrite 2 and 3 can readily adapt to the simplified interaction. Commonly used functions are in the main path; the less frequent, more complex functions can be easily reached.

DisplayWrite 4 provides a wide range of capabilities, including interchanging documents with other IBM programs or systems. DisplayWrite 4 can create, assemble, revise, and interchange documents containing text and voice/audio annotations with both standalone PCs and PCs connected to network, department systems, and host systems. Text may be brought in from IBM DisplayWrite Assistant, and 7-bit and 8-bit ASCII can be brought in using Get. Utilities covered include:

- Copy, Erase, Rename, Compress, Recover, and Convert
- DOS commands from within DisplayWrite 4
- Spelling verification for words, pages, or the entire document.

The Cursor Draw function allows the user to create line drawings and simple bar charts and graphs. Cursor Draw uses horizontal/vertical cursor movement and any keyable graphic or alphanumeric character as a draw character.

Voice/Audio Annotation enables the user to incorporate verbal comments, instructions, or attention-getting statements as an integral part of a document. Audio play capabilities include pause, resume, and volume control. A voice note is stored in the document, and may be listened to while concurrently viewing the text information. A speaker is required to listen to the voice notes.

DisplayWrite 4 requires DOS 2.10 or DOS 3.20 and runs on the IBM Personal Computer, Personal Computer XT, Personal Computer AT, *Portable* Personal Computer, and the PC Convertible. DisplayWrite 4 requires two double-sided diskette drives (with a minimum application memory requirement of 341KB), or one diskette drive and a fixed-disk drive (with a minimum of 310KB for application program memory), and a Monochrome Display, Color Display, or Enhanced Color Display. Printers are optional.

Micro CADAM

Micro CADAM is a 2-1/2 dimensional interactive computer-aided design system that runs on the IBM Personal Computer AT. Micro CADAM functions are a subset of those found in CADAM Interactive Design Version 2 and Professional CADAM.

Micro CADAM provides interactive 2-1/2 dimensional design and drafting capability in a standalone environment. It offers a cost-effective way to place a CADAM workstation on the desktops of managers, manufacturing engineers, and others not involved in production drafting but who would like access to CADAM functions. Micro CADAM lets you:

- Create, view, or modify complete production drawings with an IBM Personal Computer AT system
- Generate points, lines, circles, ellipses, and splines
- Juxtapose, separate, or otherwise locate views of a drawing
- Separate drawings into details
- Move, scale, rotate, and edit drawing elements
- Obtain hardcopy of models on IBM Color Pen Plotters

Micro CADAM is available as a licensed program, or in a Convenience Kit that includes input devices and their interfaces. Additional programs are available to allow certain user-written programs to interface to Micro CADAM and to transfer CADAM drawing files between Micro CADAM and host-based CADAM systems.

Micro CADAM requires an IBM Personal Computer AT with 640KB of memory, 1.2MB diskette drive, 20MB fixed disk drive, and a Math Coprocessor. For graphics, an IBM Professional Graphics Adapter, IBM Professional Color Display, or IBM Enhanced Graphics Adapter with memory extension is required.

Also needed are a Mouse Model M3, Mouse/Keypad Interface Board Number MCD06, and the Lighted Program Function Keypad Model

MCD05 (all manufactured by CADAM, Inc.).

The system requires DOS 3.00 or later. An IBM PC Emulation Program is required if files are to be exchanged with a host CADAM system; one of the 3270-PC File Transfer programs are required if files are to be transferred between Micro CADAM and the CADAM host system.

IBM PC PicturePak

The IBM PC PicturePak is a comprehensive series of computer graphics libraries. PicturePak images are used in conjunction with IBM PC Storyboard to produce effective presentations on your IBM PC's screen.

This addition to IBM PC Storyboard extends customization and flexibility in your presentations. You can display all PC PicturePak images in a variety of colors with the touch of a key. Also, you can display figures and images that run, dance, tumble, fly, and spin. These features can be used to highlight important points while they entertain and maintain interest.

Each PC PicturePak contains 80 to 100 professionally-drawn images and animation sequences, and sample programs on specific subject matter that can be inserted into a PC Storyboard presentation. This enables non-artists to include high-quality graphics in their presentations.

There are four PicturePaks:

Eye Openers

The Eye Openers PicturePak contains images designed for the business and professional environment. It enhances management, executive, financial and sales presentations. Categories include:

- Highlights and Introductions
- Travel and Entertainment
- Time and Money
- Communications and Technology
- Goals and Motivation
- Words and Symbols
- Productivity and Performance
- Borders and Backgrounds
- People and Places

TypeStyles

The TypeStyles PicturePak adds 18 new typefaces in 7 different sizes to the standard fonts currently available with PC Storyboard. It also includes an image library of fancy type characters and a series of letter and number animation sequences. Typeface selections include:

- Serif:
  - PC Bengal
  - PC Elite
  - PC Styles
  - PC Campus
  - PC Quill
  - PC News
- Sans Serif:
  - PC Block
  - PC Paris
  - PC Media
  - PC Basic
  - PC Show
  - PC Gothic
- Decorative:
  - PC Olde
  - PC Script
  - PC Stamp
  - PC West
  - PC Diode
  - PC School

PC Family

The PC Family is a source of hardware- and software-related images. It is designed to help in computer-oriented marketing, sales promotions, training classes, and employee orientations. Categories include:

- IBM Personal Computers
- Networks
- Internal Components
- Printers
- Keyboards
- Software
- Monitors
- People and Computers

Holidays

The Holidays PicturePak enhances everything from season announcements to holiday memos. It is excellent for advertising and internal company use. It also can be used at retail points of purchase and window displays. Eighteen holidays are included in this PicturePak.

Because PC PicturePak is an enhancement to the IBM PC Storyboard, each PC PicturePak is packaged similarly to and sold side-by-side the IBM PC Storyboard. Included in the package are the instruction diskette and four to five program and image diskettes.

The IBM PC PicturePaks run on the IBM Personal Computer, Personal Computer XT, Personal Computer AT, *Portable* Personal Computer, PC*jr* and the PC Convertible. The computers require 256KB of memory (except the PC*jr* requires 384KB of memory). DOS version 2.00 and either an IBM Color Display, Enhanced Graphics Adapter, Professional Graphics Display and Adapter, or IBM PC*jr* Color Display. PicturePaks also support TV/composite monitors and RGB video projectors. Printing a PicturePak screen requires a Proprinter or a Color Jetprinter.

IBM PC PicturePaks can be ordered only by calling 1-800-327-5711. Orders may be placed in any quantity.

IBM Personal Computer/Classroom LAN Administration System (IBM PC/CLASS) Version 1.00

The IBM Personal Computer/Classroom LAN Administration System provides installation assistance and operational support for the effective use of IBM Personal Computer education software in a local area network (LAN) environment.

The program provides a menu-oriented user interface for students, teachers and administrators to control student access to courseware. PC/CLASS keeps and prints bookmark records and progress data for students using:

- IBM Reading for Meaning Series

- IBM Reading for Information Series
- IBM Math Concepts Series
- IBM Math Practice Series
- IBM Vocabulary Series
- IBM Spelling Series
- IBM Parts of Speech Series
- IBM Touch Typing for Beginners

The program provides security for teacher and student records and all program information on the server, and includes simplified instructions for installing network software. PC/CLASS includes documentation to meet the needs of three audience groups:

- Installer
- Administrator
- Teacher

PC/CLASS supports IBM Cluster, IBM Personal Computer Network and IBM Token-Ring network hardware with Advanced NetWare/86 Version 2.0a network software, or equivalent with up to 50 student workstations per server.

Server requirements are specified by Advanced NetWare/86 Version 2.0a. A minimum (dedicated) server configuration consists of:

- An IBM Personal Computer AT or IBM Personal Computer XT
- A NetWare-compatible fixed disk
- At least 512 KB of memory
- One double-sided diskette drive
- An IBM Monochrome Display and Adapter
- An IBM Cluster Adapter, or IBM Personal Computer Network Adapter, or IBM Token-Ring Adapter and related hardware.

Up to 50 workstations (nodes) can be active at a given time on the server machine.

The workstation requirements depend on the courseware used with IBM PC/CLASS, except that:

- Diskette-less workstations may be used on IBM Cluster networks
- Workstations must have at least 256 KB of memory
- An IBM Cluster adapter, or IBM Personal Computer Network adapter, or IBM Token-Ring adapter is required
- IBM PC/CLASS administrative functions are supported on the IBM Personal Computer, IBM Personal Computer XT, IBM Personal Computer AT, IBM *Portable* Personal Computer, and the IBM PC*jr*.
- Advanced NetWare/86 Version 2.0a, or equivalent, for the server
- IBM Personal Computer Disk Operating System (DOS) Version 3.10 or later is required for the server
- IBM Personal Computer Disk Operating System (DOS) Version 2.10 or later is required for each workstation (some personal computers or networks require a level of DOS higher than 2.10).

Scientific Reasoning Series

Recent additions to the Scientific Reasoning Series are designed to help junior high school and high school students to think and reason using scientific methods. They complement the IBM series for Biology, Physics, and Earth Science. The programs can also be used in scientific literacy courses at the college level.

Each product in the series contains two separate sets of highly interactive dialogs. The programs ask a series of questions, leading students through lines of reasoning to insights and conclusions. Although emphasis is placed on teaching reasoning and problem-solving skills, some dialogues also help students learn content.

The programs in the Scientific Reasoning Series follow a learning cycle of exploration, conceptualization, and application of concepts. This discovery approach to acquiring problem-solving and reasoning skills makes the series a valuable supplement to any science classroom or laboratory.

Scientific Models: Batteries and Bulbs, and Families Version 1.00

Scientific Models helps students develop scientific models of physical phenomena through experiments and observation. "Batteries and Bulbs" investigates electric circuits and the construction of a scientific model. "Families" emphasizes constructing scientific explanations. Students descend onto a new planet to observe the inheritance of individual characteristics of the planet's inhabitants. The rules of Mendelian genetics are followed.

Ratio Reasoning: Crystals and Speed Version 1.00

Helps develop ratio reasoning skills essential to scientific activity. "Crystals" helps students find which crystals in a group are the energy crystals. Students are guided through experimental measurement of mass, volume and density as they solve the problem. In "Speed", experimental situations allow students to investigate motion and to develop the concepts of distance, time and speed.

Concept Development: Heat and Temperature, and Graphs Version 1.00

Concept Development emphasizes how concepts might be derived by procedures rather than stated definitions. In "Heat and Temperature", students introduce a new concept and develop new terms as a scientist would. The concept of heat is developed with familiar experiments using hot and cold water, vacuum bottles, ice, and water on stoves. In "Graphs", students learn to represent data using histograms and linear graphs.

The Scientific Reasoning Series requires one 5.25-inch double-sided diskette drive and 128KB of memory on an IBM Personal Computer, *PCjr*, Personal Computer XT, Personal

Computer AT, or *Portable* Personal Computer.

The programs also need one of the following: An IBM Color Display with IBM Color/Graphics Display Adapter, an IBM PC*jr* Color Display with Cable Adapter, an IBM Enhanced Graphics Adapter and IBM Enhanced Color Display. DOS is not required.

IBM Assistant Series Version 2.00

Five IBM Assistant Series programs, Filing Assistant, Reporting Assistant, Planning Assistant, Graphing Assistant, and Writing Assistant, have been enhanced and two new programs, DisplayWrite Assistant and Project Assistant, have been added. These programs provide additional functions such as pop-up "help" windows and more powerful text editing.

Version 2.00 programs display context-related help text in windows on the screen so the user can read the help information and, at the same time, usually see the portion of the screen about which there was a question. Other enhancements include more consistent back-up and install procedures and a common menu that can be used to quickly invoke each Assistant Series product.

All programs work with DOS 3.2 and can be used on the entire PC family. Both 3.5-in. and 5.25-in. diskettes are included with each Version 2 program.

IBM Filing Assistant Version 2.00

IBM Filing Assistant Version 2.00 allows users to store, retrieve, and use information normally kept in files or on index cards. Users can create customized forms for entering information, using a single page or multiple pages. As forms are filled, records are created and stored permanently.

When retrieving records, users can specify the search criteria for an exact match or a partial match on one or multiple fields. IBM Filing Assistant Version 2.00 allows users to retrieve information, print labels for mass mailings, and prepare forms such as invoices.

Files created with IBM Project Assistant can be used by IBM Filing Assistant Version 2.00. Data received in a Filing Assistant Version 2.00 data format from a host system or another IBM Personal Computer, using IBM Mainframe Communication Assistant, can also be used by IBM Filing Assistant Version 2.00.

If users want to present or summarize information stored in computer files, IBM Reporting Assistant Version 2.00 and IBM Graphing Assistant Version 2.00 will create reports and graphs using the IBM Filing Assistant Version 2.00 data diskette. IBM Project Assistant can accept files from IBM Filing Assistant Version 2.00. In addition, IBM Writing Assistant Version 2.00 or IBM DisplayWrite Assistant can use the information from IBM Filing Assistant Version 2.00 in letters and documents.

IBM Reporting Assistant Version 2.00

IBM Reporting Assistant Version 2.00 allows users to organize IBM Filing Assistant Version 2.00 information in detailed or summarized reports. Reports can consist of one or many pages and are organized by specifying columns to be displayed, sort order, and column format. Information in reports can be sorted alphabetically or numerically and up to three additional calculated columns can be created from information in the records. A report can be defined to provide needed information and the report definition can be saved to be used repeatedly. Report formatting options include titling and page numbering.

IBM Reporting Assistant Version 2.00 will search on one or multiple fields. When selecting records, the search criteria can be specified for an exact or a partial match.

Reports created with IBM Reporting Assistant Version 2.00 can be inserted into letters and documents created with IBM Writing Assistant Version 2.00 or IBM DisplayWrite Assistant.

IBM Planning Assistant Version 2.00

IBM Planning Assistant Version 2.00 is an electronic spreadsheet designed for numerical planning involved in budgeting, planning, predicting, projecting and forecasting. Customized formulas can be used for extensive calculations on the data in the spreadsheet. IBM Planning Assistant Version 2.00 provides a quick and easy way to analyze "What If" situations.

With IBM Planning Assistant Version 2.00, users can consolidate spreadsheets and use other data sources by accepting numerical data from files saved in the DIF and SYLK file format.

IBM Graphing Assistant Version 2.00

IBM Graphing Assistant Version 2.00 allows users to summarize information in bar charts, pie charts, and line charts. Users can switch between the forms or mix bar and line charts.

Graphs can be created using an IBM Graphing Assistant Version 2.00 input screen or by retrieving the information created by IBM Planning Assistant Version 2.00 or IBM Filing Assistant Version 2.00. IBM Graphing Assistant Version 2.00 can summarize IBM Project Assistant data and graph the totals or create charts using information saved in the DIF or SYLK file format.

Information can be entered by designating the X and Y axis values and filling in the data table. IBM Graphing Assistant Version 2.00 allows the user to specify legends, titles, and X and Y axis descriptions. A graph can consist of a single set of charts or up to four sets of data for bar and line charts. This allows comparisons of information, such as budget versus actual.

IBM Graphing Assistant Version 2.00 supports the IBM 7371 two pen plotter and the IBM 7372 six pen plotter. IBM Writing Assistant Version 2.00 or IBM DisplayWrite Assistant can print the graphs created with IBM Graphing Assistant Version 2.00 to make letters more meaningful. IBM Drawing Assistant Version 2.00 will accept a graph created with IBM Graphing Assistant Version 2.00 making it possible to give a special touch to the graph by adding a logo or special font or drawing the bar graph in three dimensions.

IBM Writing Assistant Version 2.00

IBM Writing Assistant Version 2.00is a word processor for creating, storing, displaying and editing documents such as office memos, letters, articles, financial tables, reports and books.

IBM Writing Assistant Version 2.00 provides for underlining and highlighting text, center, right and left justification, justifying only portions of a document, titling pages, page numbering, color selection and changes within text, handling Enhanced Graphics files, and placement of headers and footers. A 125,000 word Spell Checker and a synonym dictionary are included.

When preparing a document, an image appears on the screen showing the top and side boundaries of a piece of paper. The margins are preset for a standard page width and length that can be altered. The image can also be displayed with a graph or drawing when merged with the document, aiding in layout planning.

By saving a document as Revisable Form Text Document Content Architect (RFTDCA), a document can be sent to a host system for use by other compatible word processors.

IBM Writing Assistant Version 2.00 works with all members of the IBM Assistant Series.

IBM DisplayWrite Assistant

IBM DisplayWrite Assistant is a new text editor that provides IBM Writing Assistant Version 2.00 functions plus selected IBM Displaywrite functions including additional editing capabilities. It provides RFTDCA support for document interchange with the office systems family of products.

IBM DisplayWrite Assistant's text processing capabilities include Cursor Draw to create line drawings, bold and underline text, headers and footers, subscripts and superscripts, automatic page numbering, merge graphics or files with document at print time, global search and replace, and multiple color selection for text.

IBM DisplayWrite Assistant includes a 125,000 word Spell Checker, the ability to access a 6,000 character user-created supplemental dictionary, and a synonym dictionary. Other features of this program include online help facility, block editing, file merge, join file, join graph, directory support, document interchange, and a user printer support via table generation facility.

IBM DisplayWrite Assistant requires a minimum of 170KB of memory, and a maximum of 186KB. Additional memory is required (over 186 KB) for the Merge File with document task when the merged file is in the dBASE II, DIF, PDS or SYLK format. The Printer Description Utility Program requires a 97 KB partition size.

Note: Synonym information is based on the American Heritage Dictionaries Data Bases, Roget's II: The New Thesaurus owned by Houghton Mifflin Company and used with permission. Copyright (c) 1982 by Houghton Mifflin Company.

IBM Project Assistant

IBM Project Assistant is a project planning and management program that shows a project's critical path in the form of a horizontal Gantt bar chart. It allows users to link multiple projects together and view project subsets. IBM Project Assistant helps to plan and report project progress, costs and resource allocations and provides instant answers to "What If" questions.

IBM Project Assistant supports projects on an hourly, daily, weekly or monthly basis up to five years in length. Multiple work calendars can be defined and can contain multiyear holiday definitions. Multiple resource files may be defined, each containing up to 30 labor and/or other variable cost definitions. Calendar and resource files associated with a project may be changed at any time to gauge the effect of changes in the work schedule and/or variable costs.

This program has a sophisticated reporting system that allows the user to design his own report formats or use those supplied. Any or all of the data can be selected, sorted and printed and reports can be printed sideways.

IBM Project Assistant can be used with IBM Planning Assistant Version 2.00, IBM Graphing Assistant Version 2.00, IBM Filing Assistant Version 2.00, IBM Writing Assistant Version 2.00, and IBM DisplayWrite Assistant.

Copyrights, Service Marks and Trademarks

ADM is a trademark of Lear Siegler, Inc.

AIX is a trademark of AT&T Bell Laboratories.

Applix is a trademark of Applix Incorporated.

dBASE is a registered trademark of Ashton-Tate.

CADAM is a trademark of CADAM, Inc.

ColorPlus is a trademark of Plantronics Corporation

CompuServe is a trademark of CompuServe, Incorporated.

Corporate MBA is a trademark of Context Management Systems.

CP/M is a registered trademark of Digital Research, Incorporated.

CP/M-86 is a trademark of Digital Research, Incorporated.

Data Encoder and its associated documentation are under the U.S. Department of State Munitions list, Category XIII(b) and, as such, must be licensed by the U.S. Department of State prior to export from the United States.

DIF is a trademark of Software Arts, Incorporated.

Dow Jones News/Retrieval Service is a registered trademark and Dow Jones is a trademark of Dow Jones & Company, Incorporated.

EasyWriter is a trademark of Information Unlimited Software, Incorporated.

Fact Track is a trademark of Science Research Associates, Incorporated.

Framework is a trademark of Ashton-Tate

HomeWord is a trademark of Sierra On-Line, Incorporated.

IBM is a registered trademark of International Business Machines Corp.

IMSL is a trademark of IMSL Incorporated.

TEN/PLUS, Connect, IN/ix, INmail, INnet, and INed are registered trademarks of Interactive Systems Corporation.

INTERACTIVE and IS/5 are trademarks of Interactive Systems Corporation.

Interleaf is a trademark of Interleaf, Inc.

Lattice is a registered trademark of Lattice, Inc.

Logo is a trademark of Logo Computer Systems Incorporated.

Lotus and 1-2-3 are trademarks of Lotus Development Corporation.

Managing Your Money is a trademark of MECA (TM).

MECA is a trademark of Micro Education Corporation of America, Incorporated.

Microsoft and the Microsoft logo are registered trademarks of Microsoft Corporation.

Multiplan is a U.S. trademark of Microsoft Corporation.

NEC is a trademark of Nippon Electric Co., Ltd.

PC*jr* is a trademark of International Business Machines Corp.

PC Mouse is a trademark fo Metagraphics/Mouse Systems.

Peachtext is a trademark of Peachtree Software Incorporated, an MSA company.

Perfect Writer is a trademark of Perfect Software Incorporated.

Personal Computer AT is a trademark of International Business Machines Corp.

Personal Computer XT is a trademark of International Business Machines Corp.

pfs: is a registered trademark of Software Publishing Corporation.

PlannerCalc is a trademark of Comshare.

Professional CADAM is a registered trademark of CADAM, Inc.

RS/1 is a trademark of Bolt, Beranek, and Newman, Inc.

SAMNA is a trademark of SAMNA Corporation.

SMARTMODEM is a trademark of Hayes MicroComputer Products, Inc.

SOLOMON III is a trademark of TLB Incorporated

Supercalc is a trademark of Sorcim Corporation.

Symphony is a trademark of Lotus Development Corporation.

Synonym information in PCWriter and Word Proof is based on the American Heritage Dictionary Data Base, Roget's II, The New Thesaurus, owned by Houghton Mifflin Company and used with permission. Copyright 1982 by Houghton Mifflin Company.

Tektronix is a trademark of Tektronix, Inc.

Teletype is a trademark of Teletype Corporation.

The Learning Company reserves all rights in the Rocky, Bumble, Juggles and Gertrude characters and their names as trademarks under copyright law. Rocky's Boots, Bumble Games, Bumble Plot, Juggles' Butterfly, Gertrude's Puzzles, Gertrude's Secrets and The Learning Company are trademarks of The Learning Company.

THE SOURCE is a service mark of Source Telecomputing Corporation, a subsidiary of The Reader's Digest Association, Incorporated.

Time Manager is a trademark of The Image Producers, Incorporated.

TopView is a trademark of International Business Machines, Incorporated.

UCSD, UCSD p-System and UCSD Pascal are trademarks of the Regents of the University of California.

UNIRAS is a trademark of UNIRAS Incorporated.

UNIX is a trademark of AT&T Bell Laboratories.

Visi On is a trademark of VisiCorp.

VisiCalc is a trademark of VisiCorp.

Volkswriter is a trademark of Lifetree Software Incorporated.

Word Perfect is a trademark of Satellite Software International.

WordStar is a trademark of MicroPro International Corporation.

Workstation Publishing Software is a trademark of Interleaf, Inc.

Xenix is a trademark of Microsoft Corporation.

“ C compilers were written...
for a variety of hardware architectures, peripheral devices, and operating systems.(page 16)

“ The Development Facility lets developers create multiple active panels for use by end users.(page 5)

“ Your compiler’s library is your definitive guide to file I/O.(page 17)

“ Over 150 Commands and
programs are available
with the PC XENIX
Operating System.(page 9)

“ By using the Path command, you can
tell DOS ahead of time...
“Here’s a list of places I want you to look”.
(page 22)

“ PC Xenix is designed so that part
of the fixed disk may be reserved
for other operating systems....
such as DOS.(page 11)