

Comparative anatomy

It's no secret that I am not now (nor will I be) an unqualified supporter of the Apple Macintosh. My bias does not lie as much in the technology of the Mac versus competing systems as it does in the overwhelming way in which the Mac tries to lay the smothering blanket of its user interface over the user. For neophytes, this is fine and desirable. As your familiarity with the system increases, the Apple Desktop paradigms break down; how many of us have gone crazy trying to copy files from a folder deep within one set of folders to another deep within another set? About the fifth time you have to move and resize the windows, MS-DOS starts to look almost enviable. What's great about the Apple II is that we can run *both* the Mac-like desktop interface and more powerful alternatives such as *ProSel*.

Most of my comments regarding System 7 prior to this month have focused on its place in demonstrating that Apple has not treated the Apple II and Mac lines equitably in terms of promotion. Apple revealed most of its system software plans for the Mac well over a year in advance of System 7's release while at the same time refusing to allow any official discussion of upcoming features for Apple II systems. Given the uncertainty Apple had already generated in the Apple II market, Apple II users saw the System 7 uproar to be another nail Apple was driving into the coffin.

Now that System 7 is here, it's clear that many changes are cosmetic, with a few (but significant) "golly-gee" improvements. The cosmetic improvements do a great deal to ease some of the limitations of the original Desktop, using sensible extensions of the Desktop model. A few changes indicate ways the IIGS Desktop could be improved, and one particular change is of special interest to Apple II users.

Many of the improvements have to do with revising the "catch all" nature of the Macintosh's System Folder, which makes it more like the GS/OS System Folder. Like the System folder on a IIGS GS/OS boot disk, the Mac System Folder was intended to contain most of the system software core and enhancements such as desk accessories, initialization files, drivers, and so on.

Unlike GS/OS (and ProDOS 16 before it), the Mac System Folder did not try to group related items within subdirectories; every enhancement was dropped into the same single folder. For power users, this resulted in a System Folder so overloaded with files that locating a specific file was like locating a needle in a haystack.

System 7 solves this by grouping related files into functional folders within the System Folder; the folders identify Enhancements (programs that add features to the Macintosh environment), Apple Menu Items (Desk Accessories or other items that will appear in the Apple Menu listing), Control Panels (or CDEVs), Startup items (initialization files or programs to be executed when the system is started), Preferences (a place for programs to store custom information), and Help (to store help files). IIGS owners will recognize some of these folders as correlating roughly to the organization of the GS/OS System folder; apparently the Mac had to learn a few things from the IIGS.

The System Folder also includes a System file that contains most of the core of System 7. Like the previous Mac operating systems, the System file contains system fonts and sounds (Desk Accessories have been moved to the Apple Menu Items folder). Unlike previous systems, the System file in System 7 can be "opened" to reveal the sounds and fonts it contains, and these resources can be copied in or

out of the System file as if it were another folder.

Most of the improvements in the new organization affect the installation of and access to system software features. Mac users now no longer have to endure the use of a separate "installer" program (Font/DA Mover) to add fonts or desk accessories to their System; *but that's an ordeal IIGS users have never had to endure.* Another enhancement in System 7 is that when you drop files into the System Folder, the Macintosh Finder will try to sort out where to put them and drop them in the correct "subfolders" for you (after getting your approval).

There are also some improvements over the way the current IIGS Finder environment works. Some actions are now more intuitive, such as clicking a header category (such as "Size") in a Finder window listing to sort the list by that category.

Both the IIGS and the older Mac desktop interface used a special Control Panel desk accessory to select certain system functions and configurations. Under System 7 you can now access a Control Panel device by clicking on it directly rather than having to first open the Control Panel itself. System 7 doesn't need a Control Panel desk accessory because it allows multiple applications, including the Finder itself, to be executed and coexist in memory at one time. This effectively makes *every program* a desk accessory to every other program.

Under System 7, a new mechanism allows you to create an *alias* for a file (a special copy of the original file's icon) that relays messages to the original file when acted upon. To add a "Control Panel" item to the Apple menu, you create an alias for the Control Panel folder and move the alias to the Apple Menu Items folder (actually, System 7 comes with this already done for you). When you select "Control Panel" from the Apple menu, you are automatically switched to Finder (if you're within another application) and the Control Panel folder is opened to display its contents.

Aliases also allow the Mac to emulate a feature of the IIGS Finder. On the IIGS, you can drag an application out of its folder onto the



desktop, but the original program filename still resides in the original directory and the program can use that directory path to locate its files. On the Mac (which uses a different directory mechanism), a program on the desktop is presumed to reside in the volume directory. Many Mac programs can't find their auxiliary files, which are still in the subdirectory. To make sure the Mac application can find its other files, you can now create an alias of the main program file and drag the alias onto the desktop; double-clicking it causes the same action as clicking the main program file, and the program runs from within its folder. You can even add an application to the Apple menu by dragging its alias into the Apple Menu Items folder, or you can cause the application to automatically launch on startup by putting the alias into the Startup Items folder.

The use of aliases isn't confined to the Finder; they appear in file dialogs and other locations like "normal" files and can be used in the same way.

The other major Finder enhancement is the accommodation of the file hierarchy of a hard disk. The original Finder was very awkward with a hard disk because each folder had to display its contents in a separate window. Under System 7, when listing the contents of a folder by Name, Size, Kind, or Date (options other than the Icon or Small Icon views) a small triangle icon to the left of each folder points to the folder's name. If you click on the triangle, it rotates to point downward and the files within that folder are displayed in the same window, indented beneath the enclosing folder. You can repeat this for folders within folders, extending the display to show the entire hierarchy for the folder (or volume) in a single window. Best of all, you can select files from different levels of the hierarchy at one time; no more opening multiple windows to copy several files from different locations on a disk.

The portions of these features that extend beyond their Ilgs counterparts would be welcome in the Ilgs System Software if feasible. Our favorites are the alias concept and the hierarchical folder display; these changes make the Desktop more functional for hard disk users.

However, the lack of a MultiFinder for the Ilgs may make enhancements such as the new Control Panel implementation difficult to implement (maybe the Ilgs wizards at Apple will think of other, possibly better, mechanisms). The ability to drop an item into the System Folder and have it automatically directed to the correct subfolder would also be a nice Finder enhancement. But realize that these changes incurred a heavy penalty on the Mac: Apple recommends at least *two megabytes of memory and a hard disk* to run System 7. Our own experience is that almost nothing but the Finder can fit in a two meg Mac. My System file, which includes several extra fonts and extensions we use for desktop publishing, takes up 2.25 megabytes when loaded into memory. We've upgraded the Macs we have to a minimum of four megabytes of memory.

Another interesting feature is *virtual memory*. On a Macintosh equipped with a 68030 processor (current models are the Mac SE/30, IIsi, IIfx, and IIfx) or a system enhanced with a PMMU (paged memory management unit), it is possible to allocate memory from a hard disk to act as if it were extended (albeit slower) RAM memory for Macintosh programs. Since System 7 now supports memory sizes greater than 8 megabytes, virtual memory is a practical way to simulate large amounts of RAM with cheap hard disk space until you can afford to buy more RAM. Whether such a feature could be added to the Ilgs is also unknown to us but we anticipate it would be difficult since there is no hardware memory management unit for the Ilgs. (Apple's direction toward multimedia makes more RAM attractive; sound and picture files eat up a *lot* of memory quickly.)—DJJ

A Mac as Apple II peripheral

At least one of System 7's features actually benefits Apple Ilgs users who work in a multiple-computer environment. Apple has added what amounts to AppleShare to every Macintosh (see "AppleShare and the Apple II", September 1989, pp. 5.59-62). "Personal AppleShare" does not support the printer server or Apple II startup options of a dedicated server and has a practical limit of 5 or 6 users, but it also does not require that a Mac be exclusively dedicated for use as a server. It means, for example, that if you have both a Ilgs

and a Mac in your home or office, you can now easily connect the two so that they can access the same files on the Mac's hard disk.

To do this you need 'network connectors' for each machine and some telephone wire to hook the connectors together. Apple's LocalTalk connectors are quite expensive, but alternatives are available from a variety of sources, including our own catalog. The connectors plug into either the printer or modem port on the back of your computers. On the Ilgs you have to install the GS/OS AppleShare software (use the Installer from your system disks) and you have to turn on AppleTalk in your control panel. On ROM 1 machines this will use up two slots (7 and 1 or 2). On ROM 3 machines you lose only the slot related to the port you plugged the connector into (1 or 2).

On the Mac end, file sharing under System 7 is enabled by highlighting a folder and selecting "Sharing..." from the "File" menu. A dialog box appears similar to those used for setting AppleShare file access attributes from a Ilgs or on a Mac AppleShare workstation (System 7's "Sharing..." replaces the "Get Privileges" option of earlier versions). From the dialog you can enable the options you want, including an option to give any new folders created within the shared folder the same attributes as the one they're in (on an AppleShare server, new folders are always initially "private" to the creator).

Next, you have to enable the use of your Macintosh as a server. Opening the "Sharing Setup" control panel enables you to enter your own name and password and a "Macintosh Name" (this corresponds to a server name when another workstation polls the network). Two buttons allow you to select whether "File Sharing" or "Program Sharing" are active or inactive. Your user name also becomes an identification you can use to access your Macintosh hard disk remotely from a workstation with all privileges.

File sharing is the familiar AppleShare concept where the contents of volumes and folders can be accessed by others on the network, limited by the access privileges assigned to the folder or volume by its owner. Even though the shared item may be a folder on your Mac hard disk (it can also be an entire disk volume, though CD-ROM volumes do not appear to work), it will appear as a "server volume" to a network workstation. You can designate up to eight shared folders and volumes for your "server"; all folders within the shared item will appear as folders on the "server volume".

Program sharing is a new capability. Programs within the hierarchy of the shared folder will be available to other users based on the accesses set for the network. You can also designate a launchable application as "shared" (using the same "File" menu option) to allow other users on the network to launch and use it. Such programs should be written to expect simultaneous use by several individuals; such use may also be constrained by the use license for the program in question (most programs are distributed with at least a minimum restriction that only one copy of the purchased program is to be in use at any time).

The server metaphor extends to a "Users & Groups" control panel that allows creating a "New User" or "New Group". These options appear in the "File" when you open the CDEV and a window displaying icons for the groupings appears.

Selecting "New User" causes a "New User" icon to appear in the window, which can then be renamed to the user name you desire. Opening the icon brings up a dialog where user options can be entered. You can enter the user's password (and enable their ability to alter it) and enable their ability to connect. There is also a list in the window where the user's group assignments can be viewed, and a checkbox to enable or disable program linking for the user.

Creating a "New Group" brings up an icon that can be renamed to signify a grouping of users. To assign users to the named group, drop their user icon into the group icon. Opening a group icon displays a window with the icons for the members of the group.

There is also a "File Sharing Monitor" control panel that will display connected users and the total activity; if you want to temporarily "bump" a user from your system, you can highlight their name in the monitor list and click a "Disconnect" button. To keep them off, you'll need to disable their access to your "server".

Like other Mac files, users and groups can be completely deleted by dragging their icons into the Trash icon and selecting "Empty

Trash" from the Finder's "Special" menu (the System 7 Trash can must always be emptied *manually*).

The reason Personal AppleShare is so significant to Apple II users is that it provides a simple, inexpensive network environment for users of multiple systems. A Mac Classic with two megabytes of RAM and a 40 megabyte hard disk (suggested retail under \$1500) and System 7 software can become a network server; you no longer need to purchase a separate Macintosh and expensive server software, all you need is the appropriate network cabling. You can also have use of the Macintosh with two minor annoyances; some of the Mac's performance will be absorbed in server duties, and a workstation will lose the server connection if a program crashes the "server" Mac in an unrecoverable manner.

Practical limitations of such a server are the inability to configure the Personal AppleShare "server" as an Apple II boot volume or as a print server. Given that we've had a network for some time and long ago stopped trying to boot over the network — we use our network primarily to allow us to get at the same files from machines of different types in different rooms — neither of these omissions has been problematical. One bug has also been discovered: copying a file to an Apple II results in an incorrect auxiliary type due to a bug in the Personal AppleShare software.—DJD

The Apple II as Mac peripheral

Apple has also completed the announced Apple IIe Card that contains Apple II hardware and firmware on a card that fits within the Macintosh LC.

The \$199 Apple IIe Card can *only* be used in the Mac LC; our system used the new (512 by 384 resolution) color monitor. The IIe Card occupies the Mac LC's single 020 Direct Slot. Instructions for installation are not provided (installation is intended to be performed by a dealer) but basically it's a matter of removing an insert at the left rear of the Mac LC to provide access to a connector on the IIe Card and inserting the card into the 020 Direct Slot connector along the left edge of the LC motherboard.

Also provided for installation is a Y-shaped cable with three connectors. One attaches to a connector on the back of the IIe Card, it splits into a 19-pin connector for attachment of Apple II disk drives, and a 9-pin "game" connector similar to that on the back of a real IIe.

A Macintosh disk is provided that includes startup software. To launch into IIe mode, you double-click the IIe application from the Mac Finder. Without a IIe disk inserted you will find yourself sitting at a facsimile of the IIe power-up screen (with "Apple //e" at the top) and, after a pause while the disk drives are "scanned", the Applesoft prompt (")") will appear. If a bootable disk is present in a boot volume, it will be used to start up the virtual IIe.

You configure the IIe Card through the use of a control panel entered by the same control-open-apple-escape sequence used to enter the Classic Control Panel on the IIgs. The screen that pops up looks like the IIgs or (pre-System 7) Macintosh Desktop Control Panel; a series of icons along the left side symbolize various aspects of the card's environment that can be configured: general defaults, SmartPort (3.5 drive assignments), slots, network printer, memory card allocation (from the Mac's available RAM), and so on.

General setup. The IIe Card can be operated at one of two speeds: "Normal" (one megahertz; typical of a standard IIe) and "Fast" (two megahertz; twice as fast as a normal IIe but slower than most accelerated IIe systems). An actual change in speed will only occur when the IIe is "restarted" via a button option in the control panel.

Other options which can be set are the system beep sound (one of the Mac beeps, or a provided "Apple IIe Beep"), keyboard delay (before key repeat), key repeat rate, activation of a type-ahead buffer, selection of color or monochrome display modes (the manual is careful to point out color is only available if you have a color monitor and the use of color is also enabled in the Mac's Control Panel), and the use of Normal or Inverse characters.

Buttons at the bottom of the Control Panel allow you to eject a disk from the internal Mac floppy drive (or drives), request help on the current Control Panel device settings, quit the IIe (back to the Mac environment), restart (cold start) the IIe, or continue with the current (IIe) program.

More RAM, please. As provided, the card is designed to appear in the standard IIe configuration with 128K memory (64K of "main" memory, and 64K of "auxiliary" memory). One "General" option that is available is the allocation of a second bank of 64K "auxiliary" memory following the protocol originated by Applied Engineering's *Ram-Works* expansion card. If you enable this option, you double the amount of auxiliary memory, but we see no real sense in this since the amount is not sufficient for any real purpose.

The better way to proceed is to instead enable the use of a virtual Apple Memory Expansion Card, which is assigned to slot 7 by default. You can allocate up to one megabyte of memory to this virtual card and use it just as you would a memory card in a IIe. One megabyte may not satisfy some "power" IIe users, but it is more than enough to do credible work.

If you do enable the use of the "slot-based" memory card, you will need to disable the use of the extra 64K "auxiliary slot" memory expansion or AppleWorks 3.0 will not use the "slot-based" memory.

What's this talk of "slots"? In the IIe Card Control Panel, there is an option for slot configuration that allows you to mix and match devices from those already installed and a few "spares" provided. As supplied, the configuration is:

| | |
|--------|-----------------------------|
| Slot 1 | <empty> |
| Slot 2 | clock |
| Slot 3 | 80-column card |
| Slot 4 | mouse |
| Slot 5 | SmartPort (3.5 disks) |
| Slot 6 | Disk 5.25 controller |
| Slot 7 | Apple Memory Expansion Card |

The spare devices are a modem port, printer port, and network printer.

You can re-assign slots by dragging the device of your choice to the slot of your desire, either from another slot or from the "spares" stored in an array on the lower half of the screen. If the destination slot is already occupied, dragging a new device into it will cause the two devices to exchange places (saving the extra step of removing the first device). Unlike a real IIe, but like the IIgs, the desired startup slot can also be set.

Software compatibility does seem to be very good to excellent. Copy-protected 3.5 disks designed to expect a UniDisk 3.5 didn't operate on the internal Mac 3.5 (which does work for unprotected disks), but did work on an external UniDisk 3.5.

Most 5.25 disk-based copy-protected programs worked on an Apple 5.25 attached to the external connector; two exceptions out of a series we tried were Sirius Software's *Bandits* and Brøderbund Software's *AE* which use complicated protection techniques. All of the "problem" programs (except the 3.5 software expecting the UniDisk) worked fine on a IIgs using the same 5.25 drive.

Some communications programs had problems with the modem port. *Proterm v2.2* managed well, Applied Engineering's *ReadyLink* lost characters as if experiencing a buffer overrun, and *America Online* didn't seem to be able to cope with the serial port at all.

Some problems that crop up may remain obscure. Most of us know that the preferred interleave for formatting a 3.5 disk for a IIe (even with a *Universal Disk Controller*) or IIc is four to one (for the IIgs and an Apple 3.5 drive, it's two to one). Yet the internal drive on the Mac *always* formats a diskette to a two to one interleave. If students are moving 3.5 disks from one system to another, they may take a performance hit when using the two to one disks on UniDisk 3.5 drives.

And for you old-timers: yes, AppleVision did run.

CPU performance was generally on a par with that of a IIgs, with one salient exception. Comparing the IIe Card at 1.0 and 2.0 MHz with the IIgs at 1.0 and 2.8 MHz respectively. Given Tom Smith's comments regarding the nature of benchmarks ("Apples and oranges", March 1991, pp. 7.12-15) we paid particular attention to some of the subsystems, such as (3.5) floppy access and video speed.

We used a program that wrote and read back a 64K text file, wrote 64K of text to the 40- and 80-column text display, and filled the high-resolution graphics display line-by-line (proceeding from top to

bottom) ten times. Times were roughly comparable until it came to video performance; the IIgs at "fast" speed took 50 and 85 seconds to write 64K of text to the 40- and 80-column screen respectively versus 99 and 189 seconds for the IIe Card at 2.0 MHz, and the IIgs took 24 seconds to complete the graphics benchmark versus 39 for the IIe Card. Given that educational use of Apple II systems is often heavily graphics based, the penalty in text and graphics display could be noticeable in normal use.

What constitutes "no compromise"? When John Sculley first revealed plans for this card last spring, the operative phrase in his characterization was that the card would represent a "no compromise" Apple II. Obviously, this card is being offered to try and appease schools and IIe owners in an attempt to get them to accept a Mac platform with IIe emulation as the "next step". So how does the card measure up against the competition?

We perceive the competition to be the Apple IIe (although the IIc serves many well, the slot concept of the IIe Card seems intended more to favor comparison with the IIe), the Diamond *TrackStar-E* (see "Running with the enemy", January 1991, pp. 6.93-95) in a PC compatible host (a combination being proffered to some schools), and the competition Apple seems to fear the most: the IIgs. At a "complete environment" level, the Mac LC in combination with the IIe Card would have to be judged against the latter two systems in terms of upgrade paths.

We would classify "no compromise" to mean that the IIe Card is expected to run *all* software that a IIe could be expected to run, use *all* common peripherals that a IIe can use, and perform at least as well as a IIe.

The biggest compromise is with the slot configuration options. They are *not* as flexible as a IIe, and you may notice obvious peripherals such as a hard disk are not supported (yet; Apple included a note indicating support for AppleShare file servers is being worked on). Of the three peripheral interfaces Apple has announced for the Apple II in the last three years (Apple II WorkStation Card, Apple High Speed SCSI Card, Apple Video Overlay Card), **not one** is supported on the IIe Card. If these cards are considered valuable to the Apple II community, why is Apple foisting off an incomplete implementation of a IIe on customers as "no compromise"?

I decided I wanted to add a printer (our network LaserWriter, using the ImageWriter emulation) to slot 1; no problem. When I decided I needed the modem port in slot 2 I decided I would move the clock. After determining that the 5.25 drive was the most expendable option (I hardly use one these days), I drug the clock to slot 6; I was told it couldn't be put there (this was never a problem on my IIe). So I decided the slot 7 RAM card was expendable; I couldn't put the clock there, either. So I ditched the mouse and put the clock in slot 4. I shudder to think where the hard disk or server volume is going to go when Apple releases the appropriate support.

Comparing systems. Given that the IIe Card also requires the Mac LC, something has to be said about this combination versus the two alternatives: a IIgs that emulates a IIe but has Mac-like features as well as some of its own, and the Diamond *TrackStar* that operates within various PC compatible hosts.

In terms of Apple II compatibility, the IIgs is the best option. Some of the "limitations" of the IIgs are an artifact of the care taken to preserve compatibility, and users appear willing to deal with those minor limitations. The IIe Card's software emulation is close, but the slot configuration is considerably more restrictive (on the IIgs, you can use the internal ports *or* the slots; surely the best of both worlds).

The *TrackStar* has a few disadvantages versus the Apple IIe Card. It does not support Apple's 3.5 drive formats, and its software compatibility, while very good, is not quite as good as the IIe Card. Its price is over double that of the IIe Card. However, it does support the use of hard disk space (including network volumes), and it works in a host that can be considerably cheaper than the Mac LC. In fact, you can select a range of hosts, from a basic PC to an 80x86 monster.

Regarding the hosts themselves, that comparison is tougher. Apple has funded a study of productivity of their Mac models versus "comparable" MS-DOS systems; the results favor Apple. But not all customers buy first- (IBM, Compaq) or even second-tier PC compatibles. Many will buy consumer-priced models at substantial discounts, and we have not seen a price/performance study from Apple for

those circumstances.

As a Mac, the LC is my *least* favorite of the three new models (Classic, LC, and IIsi). The LC costs considerably more than the Classic but does not have correspondingly better performance. It *also* lacks features that make the IIsi attractive: speed (the LC is noticeably slower), memory expansion (10 megabytes for the LC versus 17 for the IIsi), and the presence of a 68030 processor (availability of virtual memory under System 7). My advice generally has been that, if you're going to buy one of the new Macs, buy a IIsi if you can. If you can't, buy a Classic and wait for the price on IIsi performance to fall (as many Mac prices have). If you drop the extra \$1500 for a Mac LC, you may not have enough money to take the real plunge in a year or two.

Another major issue is one of confidence in the company, which we have beaten to death elsewhere. The most pertinent question is whether an Apple II customer will see the Mac LC with IIe Card as representing a logical progression of Apple's technology (the "Swiss Army Knife" approach), or an investment in "new" technology that may have been better spent in making IIgs systems more attractive and cost-competitive for schools.

In our opinion the IIe Card represents a superior implementation of an Apple II emulation designed to go into a less than optimal CPU proffered by a company with less than exemplary support for its Apple II customers. The combination results in something that is more than a separate IIe or a Mac, but for which each of the individual components leaves something to be desired in price or performance.

It is also a combination from a company that has exhibited no real product support for the Apple II series in recent memory, but that felt "obligated" to supply Apple II compatibility to make its Mac products more attractive. Schizophrenia as a marketing direction does not imbue great faith.

Should Apple do something dramatically visible to alter the equation, the outlook may change. But given the limitations of the LC as a host, the erratic commitment of the parent company to support its installed base of users, and the availability of a competitive solution that may be adequate for many users, there are problems in supporting the IIe Card as a "solution" for current Apple II owners.—DJJ

Miscellanea

Apple's new Apple II Evangelist, Rob Barnes, has forwarded a letter to Apple II developers outlining the mission of the Apple II Business Unit. The three main objectives are to continue to fulfill the demand for affordable color computers such as the Apple IIe and IIgs, to enhance and improve the user experience of those who have already invested in the Apple II family, and to shift the key focus to consumers while maintaining the support for education.

Specifically, the letter says "Apple II evolution has *not* stopped." (emphasis theirs). The letter *does* seem to be directed toward the purpose of keeping Apple II developers "in the fold" as Apple's new mission is unveiled.

Barney Stone's *II at Work* newsletter has pointed out the inability or reluctance of Apple's Evangelism group to work with Apple II developers. The presence of an Evangelist that doesn't have to think "Mac first" is welcomed (Rob is grouped into the Apple II Business Unit, not with the existing Evangelist group). Also welcome would be the positioning of the Apple II as a computer for the home market that has been dominated by MS-DOS and Amiga systems sold through mass-market channels.

The most welcome development would be that Apple would actually get something moving out the door to indicate they want to market the Apple II. It has been over a year since John Sculley started making his announcements to appease Apple II users, yet Apple's marketing campaign to this point has continued to focus on Mac products, which Apple II users can only perceive as being accomplished at their expense. Given the amount of magazine ad space Apple has purchased for the new Mac models, printers, and so on, the absence of any separate advertisements for the Apple II series or new products

like HyperCard IIgs only emphasizes that Mr. Sculley is apparently willing to allow Apple II customers to flounder while forcibly slinging Mac products at the market.

Apple announces more company restructuring. Over the next year, Apple plans to change the company to focus on achieving a better market share and on cutting expenses.

This comes in the wake of reports of large numbers of unit sales for the new Macintosh models, especially the Mac Classic, but disappointing financial returns on those sales. The lower prices of the new units reflect lower profits per Macintosh sold, and Apple's strategy now has to shift toward raising those profits by further increasing sales and by cutting overhead. Apple is reported to be holding conversations with IBM regarding a technology exchange to the benefit of both companies.

Apple has announced authorization of CompUSA and selected CompuCom Systems, Inc., locations to market Apple Systems. The only systems mentioned in Apple's press release were Macintosh models. Apple revealed that the number of authorized dealer locations had fallen by 20 per cent (from 2000 to 1600) over the past two years.

Among the cost-cutting measures will be a reduction in Apple's workforce by layoffs and attrition. Apple currently employs about 15,600 including contract employees. Reductions (which may affect as many as 2000 employees) will affect all classifications.

Given estimates of John Sculley's salary, some wags have already suggested a way to recover a large sum of money by cutting a single position.

Apple's Developer CD Series Volume VII has a munged Apple II partition, much to the chagrin of Apple. Apparently the format of the Apple II partition went awry during mastering so that duplication replicated the error.

If you have access to a Macintosh with CD-ROM drive you can read files (with the exception of a munged Read Me file) from the Apple II content which was duplicated in a folder on the Mac HFS partition (Apple duplicates the Apple II material on the Mac partition for access as a file server volume). Some of the developer disc material is duplicated from issue to issue, but much new and revised information is also included.

SuperConvert (\$39.95) has arrived with the capability to import and export several graphics formats using the Apple

IIgs. Classic Apple II formats supported include normal and double high resolution, *Print Shop* (black and white) graphics, and *The Desktop Manager* (On-Three) text screen captures. IIgs formats include Super High Resolution (\$C0 and \$C1), *PaintWorks Plus*, *PaintWorks Gold*, *Print Shop IIgs* color and built-in graphics, Finder icon file, and 3200 color formats.

In addition, *SuperConvert* will import and convert several other formats, for the Atari ST, Amiga (IFF FORM ILBM display structures), Commodore 64, Macintosh, and IBM PC compatibles. Several computer-independent formats (GIF, RLE, IPI, and QRT) can also be loaded.

Once loaded and converted, the graphics can be saved in several IIgs formats as well as in the computer-independent GIF and TIFF formats. One of the options allows you to create a file that can be dropped into your boot disks's System/System.Setup folder so that the image will be displayed as a "backdrop" for Finder and other Desktop applications (those that don't overwrite the default desktop pattern).

SuperConvert's user interface conforms to the three-window design of its predecessor, *SHRConvert*. One window displays a list of available formats in the upper half; selecting a format class causes a list of specific formats supported to appear in the lower half of the window for selection. Once you have selected the format to load and convert, you can click one button in the window to get information on the format, or click a "Load" button to actually select and load a graphic.

SuperConvert is available from Seven Hills Software, 2310 Oxford Road, Tallahassee, Fla. 32304-3930, 904-575-0566. Those who have paid shareware fees for Jason Harper's earlier *SHRConvert* program should contact Seven Hills about a special upgrade rate.

If you're looking for an alternative IIgs monitor, the May 14, 1991, issue of PC Magazine ranks a large number of "Presentation Monitors" (pp. 347-377). A summary chart lists IIgs compatibility among other features, and only three Mitsubishi models are not listed as compatible.

The catch? Well, these are all high-end large-tube monitors (varying from 26 inch to 37 inch diagonal screen sizes) and the lowest-priced monitor in the group is \$3795.—DJD



Ask (or tell) Uncle DOS

Three errors regarding the newsletter for *Apple II Sysops* ("Miscellanea" last month): Douglas's last name is "Granzow", not "Gram-zow", the rate for one issue should have been \$2.50, and the ZIP code should be 21784, not 21874.—DJD

Revenge?

Like most Apple II lovers I have watched our discouragement at Apple Computer's treatment of us and our beloved machines. But I have finally come to accept the facts: Apple Inc. is indeed going to kill the IIgs and the Apple II line. Our alternatives for future computer investment are the Mac or MS-DOS clones. There are two good reasons to go MS-DOS:

price and revenge.

First, MS-DOS clone prices are generally very low. Secondly, Apple Inc. has put the proverbial screws to us. They have lied to us over and over about their "commitment" to the Apple II line. The "promise" of a burst of IIgs goodies to be announced in October did not take place. And Apple is not above dumping its old Macs on school teachers; I thought only IBM would stoop to this trick. Apple Inc. is just another corporation making a lot of bucks at our expense and is as manipulative as any company can get.

And, frankly, I don't think Apple Inc. is going to survive against the clones. MS-DOS and *Windows* will win. Eventually.

There is one serious problem of switching computer systems: loss of *AppleWorks* and other programs that are simply not available in MS-DOS, like *HyperStudio* and *DB Master*.

I would like to make a modest proposal. I would like to suggest that we Apple II users support the translation of our most important programs to MS-DOS. Fund it. Become stockholders in a corporation that does the translation and marketing.

Frankly I think it is a good risk because *AppleWorks* will sell itself, as will the other outstanding Apple II programs. (Beagle Bros, Barney, Roger, and all the other big Apple companies, are you reading this?)

If we could get an *AppleWorks* equivalent and if we Apple II users left the Apple fold en-

mass, with much publicity (an essential ingredient) we could give Apple a parting shot they deeply deserve.

Robert R. Hall
Nantucket, Maine

Windows may not be the only MS-DOS GUI (*Graphical User Interface*) *Apple has to worry about. We've seen the MS-DOS version of the America Online software which runs under Berkely Softworks GeoWorks interface; GeoWorks is not only attractive on a VGA display-based system, it runs acceptably on even the basic 10 MHz 8088 system we tried it on. (One criticism of Windows has been that it is quite greedy about system speed and memory requirements.)*

The overall problem with MS-DOS GUI interfaces has been that they suffer from two disadvantages versus Apple's integrated products. The MS-DOS environments do not have the inter-application consistency of the *Apple Desktop*; formal environments like *Windows* and *GeoWorks* may help standardize specific environments, but there is still the problem of whose standard(s) you're going to adopt when writing software.

Also, the number of standard features a programmer can reasonably expect to have access to on the Mac and IIgs is greater; all Mac and IIgs systems have extended sound

capabilities standard, for example, MS-DOS programmers can't fully rely on these capabilities being present.

Apple has to exploit their proprietary advantages by fostering commercial software (for both the Mac and IIgs) that uses these features to outperform their competition in form and function. All is not assured for Apple; although toolbox support for programmers on the Mac and IIgs is robust, MS-DOS machines may entice programmers simply by having the large installed base that will make many programmers "jump the hurdles" to write software for them. So far, Apple has apparently been more concerned about displacing Apple II sales with Mac sales to worry about their competitors.

We don't find the MS-DOS GUI's to be quite up to the par of Apple's GUI for the Mac or IIgs, but it's obvious that Apple is going to have to reckon with competition from these fronts. Especially in the case of Apple II users wanting to upgrade to more capability; these customers may find the Mac's "GUI-only" philosophy too confining, and although the IIgs has the dual text/graphics capabilities that these users may prefer Apple hasn't exactly inspired great confidence in its commitment to that system. The IIgs is Apple's competition to the low-end "GUI clone" market, and Apple needs to start exploiting their advantages there.

Regarding the MS-DOS AppleWorks "clone": check the following...—DJJ

MS-DOS AppleWorks?

Have you heard of a product for MS-DOS called **SuperWorks**? It's supposed to be an AppleWorks clone aimed specifically at Apple II users who are switching or who have both types of computers. I've heard it's even file compatible with AppleWorks. I'd sure be interested in something like this. I really like my supercharged '386 but haven't found anything that even comes close to AppleWorks in terms of functionality and ease of use. Microsoft Works has a long way to go.

Rick Pedley
Kingston, Ont.

We've received promotional literature for the product **SuperWorks** from Remarkable Technologies, 245 Pegasus Avenue, Northvale, N.J. 07647, 201-784-0900 (800-782-1955 for orders only). The product's normal suggested retail price is \$399, but Remarkable is offering \$200 off in its introductory offer.

SuperWorks's selling points for Apple II users are the importation and use of current AppleWorks data files and a sufficiently AppleWorks-like interface to allow AppleWorks users to adapt to **SuperWorks** without a long learning curve. In addition, **SuperWorks** is supposed to work with **Lotus 1-2-3**, **dBASE**, and other files. In addition to AppleWorks compatible modules, **SuperWorks** supports 3-D spreadsheets, a macro language, communications, and a print spooler. To seal the deal, Remarkable is offering a 60-day trial period; if you don't like the program, return it for a refund.

The capability it doesn't list is two-way data transfer (right now, **CrossWorks** will allow transferring AppleWorks files to suitable formats for MS-DOS programs and back). If this is a one-way road, it will only interest someone who has already decided to make the move totally to the MS-DOS environment. And many AppleWorks users have made substantial

enhancements in their copy of AppleWorks (through **TimeOut** modules and other add-ons) that probably won't have counterparts on the MS-DOS side. (It's the integration problem again; you may find MS-DOS programs that duplicate some of **TimeOut**'s capabilities, but they likely won't work seamlessly within another program like **TimeOut**'s utilities do.)

There's no doubt that given the amount Apple hasn't done to maintain product loyalty among Apple II customers that products like **SuperWorks** will find an eager customer base; AppleWorks is the premiere product that defined the Apple II as a workhorse system for several million customers. Changing horses isn't the ideal solution for users, though; a better solution would be for Apple to quit trying to put the Apple II out to pasture and to integrate it into their marketing strategy.—DJJ

GS/OS prototype?

ProSel-16 has a command called **GS.OS.VERS** whose sole function is to identify which version of the operating system is currently installed. It identified System 5.0.2 as a "prototype" whereas System 5.0.3 is identified as version \$0303 and is a "final" version. What does this mean?

Ross Barrell
Warwick, Qld.

The version number returned is the version for the GS/OS component, which explains why System 5.0.3 returns "\$0303" (for GS/OS 3.03). The tool sets, drivers, and so on all have their own distinct version numbers (that's why it's hard to determine the version just by checking the version number on a file, see "GS/OS versions", p. 7.32, May 1991).

The return value for the **GetVersion** call (page 169 of the Apple IIgs GS/OS Reference) breaks down as:

MMMMMMMM

where "MMMMMMMM" represents a seven-bit major release number, "mmmmmmmm" represents an 8-bit minor release number, and "p" represents a bit that is set to indicate a prototype (as opposed to final release) version. Apparently System 5.0.2 made it out the door with **GetVersion** returning the prototype bit set; this was fixed in 5.0.3 (and 5.0.4).—DJJ

Color printing and GS/OS

GS/OS 5.0.4 does not work when I try to print in color using AppleWorks GS (the chroma selection for "Color" remains dimmed). Is there a bug? (I have an Apple IIgs ROM 01 and an ImageWriter II.)

Carmela Anna Vitti
Santhia, Italy

There's a "gotcha" that caught us the first couple of times we tried to print in color: you need to have the color ribbon inserted **before** you set the "Print" option to bring up the printer dialog box. Apparently the new ImageWriter II driver checks for a color ribbon when the driver is "opened" and won't enable the "color" option unless the color ribbon is already present. (If you used to select "Print" and then insert the color ribbon, you'll need to reverse the procedure).—DJJ

Quick reboot

Just a tip I recently learned about exiting the ProDOS "quit" trap we've all encountered. When the messages prompt you to give a ProDOS

application name and you simply want to start up a new disk but you can't remember for your life what the pathname is, simply input the word "PRODOS" and the newly inserted (ProDOS) disk will usually reboot.

I have enjoyed receiving the **A2-Central** magazine. It's full of helpful hints, some of which seem a little too technical, but nonetheless there are golden nuggets within.

Kerry Jacobs
Valles, Mexico

This technique isn't one that will work for all circumstances but is one we forget to tell people to try. Thanks for the reminder.—DJJ

Sound impressions

I am working on converting a short story from straight text to a set of **HyperStudio** stacks. With a little rewriting I've developed a number of logical text links and I've added full screen graphics to illustrate some parts of the story. I would like to add the capability to play back some of the dialog from sound files but I have run into a problem: I'm the only actor available, and I don't do impressions. I have a number of characters, both male and female, and would like to give them different voices. Do you know of any software or hardware (or combination) that would allow me to change the pitch and timbre of my digitized voices independently of altering the playback speed? (When I try to change voices with the playback speed all I get are munchkins and trolls, not real people.)

James L. Gibson
Indio, Calif.

We're not aware of any software on the IIgs (or, really, any personal computer) that would facilitate this. The easiest solution would be to draft a few friends to help with the voices. As for why a digitized voice would be hard to modify...

A sound can be represented by a two-dimensional **waveform** (this graph is for the word "Calypso", digitized with **HyperStudio**'s digitizer and **Sound Shop** program).



The horizontal axis ("time") represents the passage of time, and the vertical axis ("A" to "+A") represents the strength of the sound's signal at that specific instant in time. When you digitize a waveform, you measure the vertical deflection of the wave at very short intervals of time, creating a discrete table of values representing the approximate (within the limits of error allowed by the sampling intervals) shape of the wave.

A constant, uniform, sound is described by the power of the sound, the number of times the waveform repeats within a unit of time, and by the shape of the repeating waveform creating the sound. The power of the sound is defined by the "height" (amplitude) of the waveform. The number of times the waveform repeats determines the sound's pitch (or frequency), whether the sound appears to be a deep bass (lower frequency) or a shrill high (higher frequency). The shape of the waveform

determines the timbre of the sound; timbre is the "textural" characteristic of a sound that allows sounds of the same pitch to sound distinct, such as a raspy saxophone versus a brassy trumpet.

Any sound comprised of a specific amplitude, pitch, and waveform shape will sound the same. Just to make things interesting, though, all of these qualities can be changed as a function of time. This is familiar to us; when you strike a piano key hard and fast, the sound it generates starts with a sharp, loud tone. If you keep the key depressed, it will gradually (over several seconds) lose volume, and also change somewhat in timbre. Eventually, the sound fades to silence. If you depress the key more slowly, the initial sound will start more gently. If you release the key more quickly (allowing the sound to be damped by the piano's mechanisms), the sound will abruptly stop.

The changing of a characteristic of a sound as time progresses is referred to as the envelope (probably so named because it describes an "enclosing" pattern for the sound). Electronic synthesizers designed to imitate other instruments usually allow the adjustment of the envelope for a sound. The simplest envelope in common use on these devices is a four-stage "ADSR" envelope to control the volume of a sound over its lifetime: "ADSR" is an acronym for "Attack" (the initial change in volume as the sound starts), "Decay" (the rate at which the volume is lowered after the initial attack), "Sustain" (a volume to be sustained after the decay until the synthesizer is told to stop the sound), and "Release" (the final rate at which the volume trails off to zero once the synthesizer has been told to stop the sound). Some current synthesizers use more complex envelopes; synthesizers can usually also change other aspects of the sound as time progresses.

The human voice varies many qualities of the sound it is producing in order to generate speech in a near-continuous fashion. You can see that there is very little "regularity" in our waveform for "calypso". With some subjective analysis (and rough placement of the syllables in the diagram), we can almost see the "hard" attack of the "c" in the first syllable where the amplitude rises sharply, then trails into the flatter (and quieter) hissing sound of the "s". Tracing the gross structure of the sample in this wave gives us an envelope for the volume characteristics of the pronunciation of "calypso".

Within this envelope and even within each of the syllables the fine structure of the sample (the changes in amplitude occurring within syllables) indicates that the waveform is constantly changing. In the first syllable, we change from the "k" sound of the "c" to the "ahh" sound of the "a". This will in turn blend into the "l" sound starting the next syllable ("lyp") and so on. Here the timbre and the pitch of the sounds is more of a factor.

The human speech mechanism makes these changes by varying the tension of the vocal chords, the shape of the throat and mouth, the rate of air flow through the vocal chords, and so on. If you sat down to determine exactly how to duplicate these qualities, you get an idea of the complexity of speech synthesis.

Compressing or expanding the time frame of the entire sound gives the distortion you mention. What we would need to make this

sample sound like another person saying "calypso" would be the ability to separate the overall envelope for the pronunciation of the word from the relative pitches for the various sounds (whether the "ah" the same, higher, or lower in pitch than the "oh", for example), and the timbre of the sounds. Then the word could be reconstructed by applying the envelope and relative pitch sequence to a series of tones with the timbre of a new voice.

It's even more complicated than this, because certain characteristics of sound make it necessary to "fine tune" some of the interactions for the new "voice" we are creating. This is why computerized speech has been so difficult to create in a natural-sounding form. Digitizing a sound is easy; creating a new sound (or revising an existing one) is still difficult.

It is possible to use the computer to assist in these transformations, though the use of such software is specialized and may not be available "off the shelf" (it's certainly beyond us). Even locating software to allow editing waveforms is difficult; most "synthesizers" these days seem to be sampling existing sounds and making the limited alterations allowed by effects processing techniques rather than designing new waveforms from scratch.

There is a device used for roughly this purpose; it's a synthesizer called a **vocoder**. A vocoder is a real-time sound processor that takes an input sound, breaks it into frequency and envelope components, modifies these according to settings on the device, and then re-assembles them into a new version of the sound. Adjusting the vocoder to shift the pitch of the input sound and adjust its timbre while leaving the envelope essentially unchanged alters the "voice"; this is one of the device's more common uses. In most cases, the result sounds "robotic", but it may be worth some investigation. If you want to experiment, you may be able to find a vocoder at a professional musician's supply store in your area; rental may be a desirable option since prices run from several hundred to several thousand dollars.—DJD

Another SCSI possibility

In your discussion of SCSI standards, you mentioned a couple of things which might be meant by phrases like "Macintosh SCSI". There's (at least) one more. It turns out that at least one common SCSI control chip found in SCSI devices puts the SCSI bus into an illegal state for a very short time. The typical MessyDOS machine SCSI interface doesn't notice the illegal state, but the Macintosh does, and very grumpily reports the bus error. So the "Macintosh SCSI" phrase could mean simply that the device does not have this particular problem. In that case, there should be no problem using that device with the Apple II SCSI cards.

John W. Baxter
Port Ludlow, Wash.

Jerry Kindall at Quality Computers also sent a note that it may be an indication of whether the drive supplies terminator power since some Macs apparently don't require the drive to supply power (Jerry said the IIx was an exception; it doesn't supply terminator power). We've been confused about this since we get conflicting reports. We checked the AppleLink technical library and the Mac Plus and Mac Portable were specifically listed as not supplying terminator power.—DJD

Key resources

I am looking for a GS/OS resource editor for my IIGs. On the Macintosh a program called ResEdit is available and lets one change the keyboard layout (re-mapping of the keyboard) as one wishes. Can the Genesys software do the same thing? There has been a passing mention of this software over the past year but really nothing much has been said about the software capabilities itself.

Dr. S. S. Datye
Akureyri, Iceland

We think the answer is "yes", but we're a little confused, as we will explain...

Mac files have always been designed to contain two collections, or "forks", of information. The data fork contains information that can be organized in any way your program cares to access it. The resource fork contains packages of information, which can be data or commands, that have specific defined structures.

A resource is loaded by requesting it via a specific toolset called the Resource Manager; the Resource Manager will take care of locating the resource without your program having to know painful details of the resource fork's structure. Resources provide a consistent means of organization for system elements (icons, windows, menu bar contents, and so on) that may need to be altered, transferred, or shared between programs.

The Mac operating systems is based on resources as a fundamental component. Mac programs often contain only resource forks since the program code must reside in a CODE resource.

ResEdit is a Macintosh software tool developed by Apple engineers that allows viewing the characteristics of Mac resources and modifying them, or even copying entire resources between files. For example, if you want to modify the contents of a menu list in English so that the contents are listed in German, the menu list resource can be "opened" and the items edited so that they are replaced with their German equivalents. Or if you have already prepared a new copy of the menu bar in a new file, you can "cut and paste" the copy into the file you want to modify. Macintosh programmers (and well-informed users; you have to know enough about the structure of the items you are modifying to change them into a properly recognizable form) can use these techniques to change the visual appearance of a program to "localize" it for various languages without having to re-write and re-compile the entire program.

Probably the most profound change in System Software version 5 for the IIGs was the addition of a Resource Manager and the enhancement of several toolsets to allow specifying the use of resources for required data structures. As a result, some system files and newer programs use resources for their data structures. Formerly, these structures were stored in a "non-forked" file along with the rest of the program code and data.

However, since resources were not inherently part of the IIGs operating system from the beginning their use is not as ubiquitous as on the Mac. Some types of IIGs data are not kept in resources as they are on the Macintosh. Also, resources are not available from ProDOS

8 applications.

Here's where we get confused. There is a keyboard translation table resource defined for the IIGS System Software. We checked the USA version of System Software v5.0.4 and there wasn't a resource of the proper type (rkTransTable, \$8021) defined in the Sys.Resources file (in the System/System.Setup folder). But there is a French version of v5.0.2 supplied on Apple's Developer CD-ROMs, and its Sys.Resources file does have an rkTransTable resource installed. Our confusion rests in whether adding an rkTransTable resource to the Sys.Resources file will cause it to be available.

Therein lies another tale. **Genesys** currently does not have a specialized editor for the rkTransTable resource, but its generic Hex/ASCII editor could be used to create or modify a table. The rkTransTable format is given in **Apple IIGS Toolbox Reference Volume 3**, which also contains information on the Resource Manager and the new Event Manager calls that support the translation routines. (The Event Manager has to be active to use the translation routines.) Here's what we ran into: according to the **Toolbox Reference**, a resource ID of the form \$0FFF06xx is required for an rkTransTable item (the French 5.0.2 used \$0FFF0610), and **Genesys** wouldn't accept an ID larger than \$07FFFFFF. We've been told that \$07FFFFFF should indeed be the maximum value for a resource ID, so we're going to need to spend a little more time sorting this out. (Maybe we'll have it cleared up by

next month.)

In terms of paralleling **ResEdit**, **Genesys** for the IIGS is able to act as an editor to modify the resource fork of a new or existing file. For example, Beagle Bros's **Platinum Paint** has a very large list of items in its Edit menu (the menu "scrolls" vertically to allow access to all of the items); some users have modified the list to change the order of items to bring more commonly used items closer to the top of the list.

In addition to being a resource editor **Genesys** can also serve as a design tool for the user interface of a program. **Genesys** will display the element it is editing so that you can get a visual impression of its appearance in your finished program. It can also save the instructions needed to generate the resource in your program in several forms: most common IIGS languages (Pascal, C, assembler, MPW IIGS, and so on), in REZ source code (REZ is Apple's resource compiler supplied as part of the **APW Tools and Interfaces** package from APDA), or in raw binary form.

Genesys includes its own language for writing your own Source Code Generator (SCG), and some third-party SCG's are appearing. One from Doni Grande (included with recent versions of **Genesys**) even builds **ORCA/C** source code for a skeletal application program: after you define your "New Application" elements using **Genesys**'s resource editor, you select the options to save the source code and equates and the SCG spits out an **ORCA/C** program and "include" file ready to compile and run. (Of course, all that's been defined is the interface; you still have to write the program logic to make it do something useful.)

Resources on the IIGS are maturing as part of the operating system. In addition to the basic types described in **Apple IIGS Toolbox Reference Vol. 3**, newer types have been listed in the Apple II Technical Notes for the IIGS (IIGS Technical Note #76, "Miscellaneous Resource Formats").—DJD

Resource forks and ProDOS 8

What is the ProDOS format for storing resource files? This would be the actual information in the catalog blocks that shows where the blocks are for the program and resource fork. The old format included the file type, date, name, name length, first block, file length, etc. And why can't an old fashioned copy program copy the file? (Though that would probably be explained by looking at the block information.) I haven't found this detailed information in any of Apple's books, though it is probably in their tech notes somewhere.

James Grey
Anchorage, Alaska

There is no ProDOS 8 format for storing information within the resource fork because there is not (and is unlikely to be) a ProDOS 8 Resource Manager. Resources are accessed by calls to the IIGS (GS/OS) Resource Manager, and applications don't know (and don't need to know) anything about the way the individual resources are stored within the resource fork. (There is probably a reason for this: if the Resource Manager needs to be altered or improved it can occur without breaking previous applications.)

The physical way an extended file (one with a resource fork) is stored is relatively simple. As for other ProDOS files, the ProDOS directory

entry contains a two-byte "key" pointer and a storage type for the file.

The "key" pointer points to different things depending on the storage type of the file. For a file that uses one block or less of information (a "seedling" file), the pointer points to the lone data block. For a file that contains 2 to 256 blocks of data (a "sapling" file) the pointer indicates the single index block for the file. For a file that contains over 256 blocks of data (a "tree" file that requires more than one index block) the pointer indicates a block which in turn contains pointers to the index blocks for the file.

For an extended file, we have in effect two parts: the data fork and the resource fork. In this case the key pointer points to a block that contains information for the data fork at offset \$0000 (byte 0) and for the resource fork at offset \$0100 (byte 256).

The information for each fork uses the same format. The first byte specifies the storage type for that fork (normally one would expect only one of the first three):

```
$01 seedling
$02 sapling
$03 tree
$04 Pascal storage area
$05 extended file
```

The next two bytes contain the number of the key block for the file (this would correspond to the proper data for the fork's storage type). The following two bytes specify the size of the fork in blocks. The next three bytes specify the number of bytes in the fork. All remaining bytes must be set to "\$00".

(Incidentally, this is all delineated in **Apple II ProDOS 8 Technical Note #25: "Non-Standard Storage Types"**. Apple's not trying to hide the information; technical notes are where any updates to information in the manuals, which are always going to lag the real world by at least their publication lead time, are likely to be found.)

ProDOS 8 utilities can't copy the files because ProDOS 8 itself is not designed to manipulate extended files. Some utilities don't ignore files of an unexpected storage type and try to manipulate the file anyway; in the case of **Copy II Plus** version 8, we found it thought the file was a seedling file and so copied only the extended file's key block as if it were a data block.

Notice that the format of the "index" blocks that are not actually part of the data portion of the file are only good for ProDOS physical volume structures; GS/OS works with other types of volume and directory structures. If you have a IIGS, you should be using "GS/OS friendly" utilities which have several attendant advantages.—DJD

Coincidence?

A random thought per the movie 2001: A Space Odyssey: HAL transposes to IBM; MAC can transpose to CRT. What's the significance?

V. R. Reading
Virginia Beach, Va.

We think it means you miscounted; we got "DRT" ("dirty?") instead.

Incidentally, Author C. Clarke (who wrote the short story "The Sentinel" that inspired 2001) has denied the HAL to IBM transposition was intentional.—DJD

A2-Central™

© Copyright 1991 by
Resource-Central, Inc.

Most rights reserved. All programs published in **A2-Central** are public domain and may be copied and distributed without charge. Apple user groups and significant others may obtain permission to reprint articles from time to time by specific written request.

Publisher:
Tom Weishaar

Editor:
Dennis Doms

with help from:

| | | |
|----------------|----------------|-----------------|
| Sally Dwyer | Dean Esmay | Joyce Hammond |
| Jay Jennings | Jeff Neuer | Ellen Rosenberg |
| Denise Shaffer | Tom Vanderpool | Jean Weishaar |

A2-Central—fired **Open-Apple** through January, 1989—has been published monthly since January 1985. World-wide prices (in U.S. dollars; airmail delivery included at no additional charge): \$28 for 1 year; \$54 for 2 years; \$78 for 3 years. All back issues are currently available for \$2 each; bound, indexed editions of our first six volumes are \$14.95 each. Volumes end with the January issue; an index for the prior volume is included with the February issue.

The full text of each issue of **A2-Central** is available on 3.5 disks, along with a selection of the best new public domain and shareware files and programs, for \$84 a year (newsletter and disk combined). Single disks are \$10. Please send all correspondence to:

A2-Central
P.O. Box 11250
Overland Park, Kansas 66207 U.S.A.

A2-Central is sold in an unprotected format for your convenience. You are encouraged to make back-up archival copies or easy-to-read enlarged copies for your own use without charge. You may also copy **A2-Central** for distribution to others. The distribution fee is 15 cents per page per copy distributed.

WARRANTY AND LIMITATION OF LIABILITY. We warrant that most of the information in **A2-Central** is useful and correct, although drift and mistakes are included from time to time, usually unintentionally. Unsatisfied subscribers may cancel their subscription at any time and receive a full refund of their last subscription payment. The unfiled portion of any paid subscription will be refunded even to satisfied subscribers upon request. OUR LIABILITY FOR ERRORS AND OMISSIONS IS LIMITED TO THIS PUBLICATION'S PURCHASE PRICE. In no case shall our company or our contributors be liable for any incidental or consequential damages, nor for ANY damages in excess of the fees paid by a subscriber.

ISSN 0885-4017

GENIE mail: A2-CENTRAL
Voice: 913-469-6502
Fax: 913-469-6507

Printed in the U.S.A.