

***A journal and exchange of Apple II discoveries***

## System 6.0 AppleShare startup

By Dennis Doms

We aren't ready for a full review of Apple's networking solution for Apple IIs, *AppleShare 3.0*, since most users won't ever have a need for it. At \$1,199 retail plus a spare Mac to run it on, you either need to be an institution or a wealthy individual to afford it. Even we haven't installed it yet; our current *AppleShare 2.0* installation works fine and we've been timid about changing. But for our school readers who have to know these things, here's how to set up GS/OS System 6.0 to boot from the server.

*AppleShare 2.0* required a separately supplied Apple II Setup Disk to install startup files on the server. These files handled the bootstrap operations of letting an Apple II log onto a server volume and, if the proper files were present, start up from it. The Apple II Setup Disk was occasionally updated and supplied with the IIGs System 5.x package. But the IIGs System 6.0 package doesn't include a new Apple II Setup disk of its own, though the System 6.0 CD-ROM does include a newer version (*Apple II Setup v2.2*).

Instead, *AppleShare 3.0* now includes the Apple II Setup disk (version 2.1.3), which can be used to configure the server for use with either Apple IIGs System 5.0.4 or 6.0. What the manual doesn't make clear is that if you use the *AppleShare 3.0* Installer disk to put *AppleShare* on the server, the "Easy Install" option doesn't automatically add the Apple II setup information. To do this first install *AppleShare* itself, then from the Installer menu pick "Customize," and select the Apple II Setup software installation. This adds all the files necessary to start up a IIe using *ProDOS 8* to the server.

Next, if you want to boot IIGs System Software, you need to install it on the server using the IIGs System Disk. The procedure is basically the same as described in the *AppleShare 3.0 Administrator's Guide* (pages 76-78) with a few exceptions for 6.0. Here's the routine:

Initialize a blank 3.5 disk. Use the System 6 Installer to put "AppleShare on a 3.5 disk" on your blank disk. Restart your IIGs using the disk you've just created as the boot disk. You'll be asked for your user name. Give your server's Administration name and password.

When you get to the Finder, choose Control Panel from the Apple menu and select the *AppleShare* icon. Log yourself onto the network startup volume using the Administrator's name and password. Close the control panel.

Run the System 6 Installer again. Click the disk button in the Installer until the name of the server's startup volume appears above the window on the right side on the Installer screen. Pick Custom Installation and choose the script called "Network: Server Startup".

**If you need other IIGs system resources installed** (say the Video Overlay Card drivers), you'll also need to install them from a IIGs using the 6.0 Installer. You might as well go ahead and do this now.

Quit the installer. From the Finder, select the System and Icon folders that have just been placed on the server and change the access privileges to "Everyone". The trick here is to click on the folder to

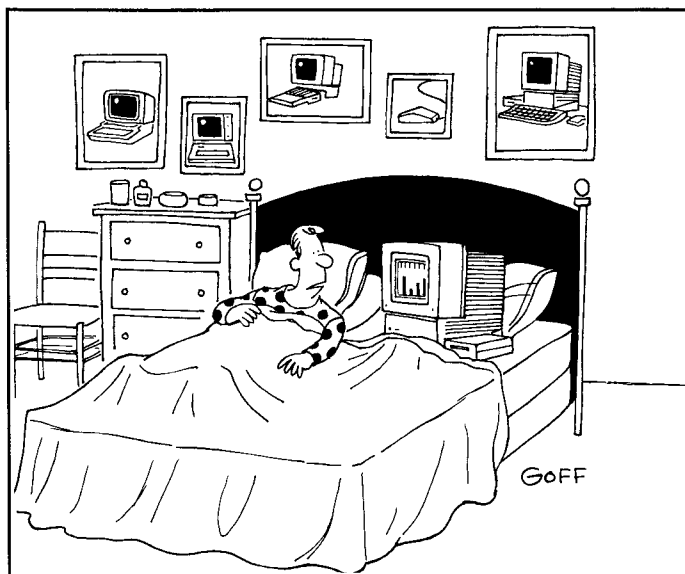
highlight it, select "Icon Info" from the "Special" menu, click on the "Access" tab at the bottom of the window that pops up, set the access to "Public", and press the button that says "Apply to Enclosed Folders." Don't forget to also do this to the Icons folder and any other folders you want your workstations to use. If you don't make both the System folder and the Icons folder public, workstations can't boot from the server.

**Preparing the server for Apple II startup.** Once the System Software is installed, wander back to the server Mac and get into *AppleShare Admin*. Choose "File Server Preferences" from the Server menu and click on the box that says "Enable Apple II Startup Service." Now close down the server and restart it.

Now turn on all your IIGs network workstations and *OPEN-APPLE/CONTROL/ESCAPE* into the control panel. On ROM 1 machines both slot 1 (or 2, depending on where your AppleTalk connector is plugged in) must be set to "Your Card" and slot 7 must be set to AppleTalk, and Startup should be set to slot 7. On ROM 3 machines either slot 1 (or slot 2, if that's what your network is plugged in to) or slot 7, as well as Startup, must be set to AppleTalk. When the control panel is set correctly you should be able to startup over the network.

**You can provide a little more fine-tuning** by returning to the Mac server software, opening the user list and the group list, selecting one or more users or groups, and choosing "Set Apple II Startup Info" from the Startup menu. This allows you to assign each user a startup application, a startup folder, and a default printer.

*AppleShare* also provides a user "Setup" folder. Theoretically, startup files (such as items from the "System.Setup" or the "Desk.Accts")



"AND THEN THEY SAID I WAS TAKING MY  
WORK TOO SERIOUSLY."

folders from the IIgs startup disk) that only need to load for a certain user should go into the "Setup" folder associated with that user on the server. You'll find a Setup folder inside each user's personal folder. And you'll find those in the server's "Users" folder.

I say "theoretically" because if a file's programmer assumed the file would always be in a specific folder on the boot volume, such as the "System.Desk.Accts" folder, the program won't work when put into an isolated "Setup" folder on the server. You may have to experiment with these; this isn't an exact science.

**The ZipGSX: start off slow.** If you're using a ZipGSX accelerator and booting from the server, you need to set switch 1-6 on each card to off. That allows the system to start up without acceleration. If you don't do this, the AppleShare interface in the IIgs won't sync up properly with the network and the IIgs will never see the server.

To bump the ZipGSX back up to speed, you have to install the file "Ziplnit" on the server. This is a bit tricky, since you need to configure Ziplnit before you install it on the server. My solution was to put Ziplnit in the "System.Setup" folder of my local startup disk, configure it using the Zip Utility Program ("ZipGS.Sys16"), then copy it to the server (either into the "System.Setup" folder for all users or into the appropriate user's "Setup" folder).

When you configure the Zip, you must set the AppleTalk delay to on. If you don't do this, when you start up from the server (with the init installed) AppleShare will gag as the init loads.

AppleShare seems to work fine under System 6.0 with version 1.2 or later of the Zip Utilities Disk and with the init set to 100 percent of the ZipGSX's speed (with older versions under System 5.0.4 we had to set our 8 MHz ZipGSX back to 81 to 87 percent or we couldn't use the network).

**New features for Apple IIgs networking.** A new Apple II Setup version 2.2 is available; but the only source we know of at the moment is the System 6.0 CD-ROM. We're working with Apple to see if we can add it to our APDA materials. The primary advantage of this version over the one supplied with AppleShare 3.0 is that it allows selecting whether you want your IIgs to boot into ProDOS 8 or GS/OS. Booting into ProDOS 8 is much faster (a few seconds versus a few minutes), but you can't run IIgs programs without loading GS/OS first.

Schools that use IIgs systems primarily as fast IIes may only want and need ProDOS 8 most of the time; after installing the new Apple II Setup files on the server, they can use the IIgs System 6 "Network" icon in the Control Panel to select ProDOS 8 as the startup operating system.

**But the Catch 22 is that** Apple hasn't provided any way to set this parameter back to GS/OS from ProDOS 8. Once you've selected booting into ProDOS 8 over the server there's no way to change your mind without booting up each workstation you want to change with a System Disk. Let's fix this.

The network startup selection is identified by the state of battery RAM parameter \$62 in the IIgs, so the selection is as "permanent" as the contents of battery RAM (that is, if you select ProDOS 8 the IIgs will continue to boot into ProDOS 8 from the server until the setting is changed). Bits 0-1 of parameter \$62 determine the network startup operating system (this has no effect on the operating system when you startup from a drive attached directly to a IIgs rather than over the network!); a value of "0" indicates the parameter hasn't been set, "1" indicates the desired network startup system is GS/OS, and "2" indicates it is ProDOS 8. Our little program will also check for a value of "3" though it's apparently not used by System 6.0, just in case the choices are expanded in the future (boot up OS/2?).

Here's our program:

```
1000 REM == toggle AppleShare OS boot on IIgs from ProDOS 8==
1010 TEXT : HOME
1020 GOSUB 1240: REM install ML stuff
```

```
1030 IF PEEK ( - 1103) = 6 THEN CALL 841: IF PEEK (856) < > 0 THEN PRINT "This
      program is for a IIgs only.": STOP
1040 CALL 768: REM read BRAM parm $62
1050 EC = PEEK (837) + PEEK (838) * 256: IF EC THEN PRINT "Error ";EC;" occurred on
      ReadBParam.": STOP

1060 PRINT "The current AppleShare startup system"
1070 OS = ( PEEK (839) + PEEK (840) * 256) - (4 * INT ( PEEK (839) / 4))
1080 IF OS = 0 THEN PRINT "has not been selected yet (with the": PRINT "Network
      CDev). The default is GS/OS."
1090 IF OS = 1 THEN PRINT "is GS/OS."
1100 IF OS = 2 THEN PRINT "is ProDOS 8."
1110 IF OS = 3 THEN PRINT "Uh, I'm confused (unrecognized value).": STOP
1120 PRINT : PRINT "Pick <P>roDOS, <G>S/OS, or <N>o change:";

1130 GET AS: IF (ASC (AS) > 96) THEN AS = CHR$ (ASC (AS) - 32)
1140 IF AS < > "O" AND AS < > "P" AND AS < > "G" AND AS < > "N" THEN GOTO 1130
1150 IF AS = "N" THEN GOTO 1220
1160 IF AS = "O" THEN POKE 839,4 * INT ( PEEK (839) / 4)
1170 IF AS = "G" THEN POKE 839,1 + (4 * INT ( PEEK (839) / 4))
1180 IF AS = "P" THEN POKE 839,2 + (4 * INT ( PEEK (839) / 4))
1190 CALL 803: REM write to BRAM
1200 EC = PEEK (837) + PEEK (838) * 256: IF EC THEN PRINT "Error ";EC;" occurred on
      WriteBParam.": STOP
1210 HOME : GOTO 1040: REM loop and check again

1220 VTAB 22
1230 END

1240 REM -- subroutine for TB call... --
1250 FOR I = 768 TO 856: READ J: POKE I,J: NEXT I: RETURN
1260 REM ML data for read/write BRAM
1270 DATA 8,24,251,194,48,169,0,0,141,69,3,72,244,98,0,162,3,12,34,0,0,
      225,144,3,141,69,3,104,141,71,3,56,251,40,96,8,24,251,194,48,169,0,0,
      141,69,3,173,71,3,72,244,98,0,162,3,11,34,0,0,225,144,3,141,69,3,56,251,
      40,96,0,0,0,0
1280 REM ...and IIgs ID via $FE1F
1290 DATA 8,169,0,141,88,3,56,32,31,254,110,88,3,40,96,0

If you look closely, you'll notice that "0" will "zero" the selection
value if requested (we added this for testing, but it's not listed in the
menu selections).

Here's the machine language loaded by lines 1240-1290:

* for CALL 768 (read parm value from BRAM)
RDBRAM      php          save status
            clc          switch to native mode
            xce
            rep          $30    tell processor
            LONGA      ON      and tell assembler
            LONGI      ON
            lda          $0000  clear result buffer
            sta          ERRRET  and error return
            pha          and save result space
            pea          $0062  save parameter #
            _ReadBParam  read parameter
            bcc          RDOk   skip if no error
            sta          ERRRET  save code if error
            pla          get result value
            sta          PARMVAL  and save it
            sec          back to emulation
            xce
            LONGA      OFF      tell assembler
            LONGI      OFF
            plp          restore status
            rts          back to BASIC
```

\* for CALL 803 (write parm value to BRAM)

```
WRBRAM    php                (almost the same)
           clc
           xce
           rep    $30        tell processor
           LONGA    ON
           LONGI    ON
           lda    $0000
           sta    ERRRET
           lda    PARMVAL    get new value
           pha
           pea    $0062
           _WriteBParam      write to parm 62
           bcc    WROK        skip if no error
           sta    ERRRET      otherwise, save
           WROK    sec
           xce
           LONGA    OFF
           LONGI    OFF
           plp
           rts
```

\* PEEK(837 and 838) for error return value

```
ERRRET    dc    i2'$0000' word for error code
```

\* PEEK or POKE (839 and 840) for parm value

```
PARMVAL    dc    i2'$0000' word for parm value
```

\* for CALL 841 (identify IIgs or not)

```
CHKIIGS    php                check for IIgs
           lda    $00          clear check byte
           sta    IIEFLAG      (becomes non-0 if IIE)
           sec                  carry is flag
           jsr    IDROUTINE    returns system info
           ror    IIEFLAG      move flag to result
           plp                  (recover status)
           rts
```

\* PEEK(856) for result (zero if IIgs)

```
IIEFLAG    dc    i1'$00' byte for result
```

**AppleShare or System 7 File Sharing?** With personal file sharing available in Macintosh System 7.0, the primary advantages of using AppleShare 3.0 on a 4-Meg, dedicated Mac are additional network speed, additional users, and the ability to startup Apple IIs over the network. (A note about the words "dedicated Mac." Under System 7, you can run other applications besides AppleShare on the "server" Mac. But I've tried it on a PowerBook 170, which is one of the faster Macs available, and I can guarantee that you wouldn't want to use the server Mac as a workstation. File access by another network client will often slow the server Mac to a crawl even with AppleShare limited to only part of the Mac's resources.)

If you can give up network booting and you have fewer than a dozen or so workstations hooked to your Mac, System 7's built-in file sharing maybe be all you need.

A disadvantage of booting over the network is that it's slow unless you boot into ProDOS 8, but that kills the advantages of the IIgs System Software. The advantage of booting over the network is that you have the network volume as a hard disk boot device; although it's slow, it's big — it will hold a completely configured system that wouldn't begin to fit on a floppy.

At the administrative level, booting over the network trades speed for low-cost mass storage. Assume a server cost of \$4,000. With a handful of IIgs systems you'd be better off (in both finances and performance) buying a hard disk subsystem for each computer; five 50-meg drives and controllers might cost in the \$3,000 range, leaving money for memory expansion (more memory may be needed to use

AppleShare no matter how you start up the computers). With ten IIgs systems the cost of individual hard disks exceeds the cost of the server plus extra hard disk space; adding a 500 megabyte hard disk to the server for about \$1,500 would give each user an average of 50 megs of disk space whereas buying individual drives and interfaces might cost \$6,000. If some "high performance" systems are needed a few IIgs's could be configured with local hard disks.

In educational institutions with an existing base of Apple II systems the above issues need to be looked at carefully before buying completely new (non-Apple II) systems that can't start up from a network and thus require the expense of a local hard disk.

## Apple Writer: built-in functions

by Ron Evry

While even novice users can be up and running *Apple Writer* in minutes, the program has a host of powerful built-in features that deserve taking the time to learn. Since, as we mentioned here last month, this best-selling word processor is now freeware, let's take a look at how to tame this beast.

The built-in commands that are included in *Apple Writer* can be divided into a few groups: the cursor movement commands, which we touched on in last month's article; the control character commands, which are used to access the ProDOS command menu, the Additional Functions menu, and lots more; and a vast array of printer and page formatting commands.

The Control Character command set has a number of handy items. Last month, we mentioned that CONTROL-B and CONTROL-E will move the cursor to the beginning and end of your document. And we mentioned that CONTROL-W and CONTROL-X will delete or retrieve a word or paragraph respectively, depending upon the direction of the arrow in the top bar (set by the toggle command CONTROL-D). CONTROL-R was mentioned as the command for toggling Replace mode on and off. We also barely touched upon the command CONTROL-P, which actually opens up an entire array of instructions to your printer.

**Save and load tricks.** One feature we began to explore was the CONTROL-L and CONTROL-S commands, for loading and saving files from a disk. But suppose you wanted to only save, say, just part of a document as a file on disk? Nothing could be easier. First of all, go to the end of the part you want to save and take note of the last few words. Then place your cursor at the beginning of the section and press CONTROL-S. When prompted for the name of the file to save, use the standard filename procedure you have been using, but at the end of the filename put an exclamation point, the ending words of the section, and another exclamation point. This grouping of words is called a marker since it marks the boundary of the file segment you wish to save. Do be careful to use enough words in the phrase so as not to end your file at another place where the same words can be found.

The exclamation points used in this procedure are called delimiters. The standard delimiter used is the exclamation mark, but most of the characters that can't be used in a standard ProDOS filename will work, including @, #, \$, %, &, \*, (, ), <, and >. The slash (/), on the other hand, won't work, since it is used in filenames.

As an example, if you were saving part of a letter you wrote so that you could use that part in another letter to somebody else, you might put your cursor at the point from which you wanted to save and press CONTROL-S. Then at the prompt, you might enter: /DATADISK/LETTER.1!Yours Truly,! then press RETURN.

If you wanted to mark the end of your document at an exclamation point, then you would need a different delimiter (ex: /DATADISK/LETTER.1&big savings!& ). Notice that the marker is case sensitive, unlike the rest of a standard ProDOS filename.

Another trick you can do is append either all or part of a document onto another document already on disk without going to the bother of loading the first file. To save the entire document you have in memory on to the end of an existing file, press **CONTROL-S**, enter the filename from the disk, and add a plus sign (+) to the end of it (ex: /DATADISK/MINUTES+ ). You can use the same procedure as before to add part of a document in memory to the end of an existing file by putting the plus sign after the last delimiter (ex: /DATADISK/LETTER.1!Yours Truly.1+ ).

**Conversely, any file on disk can be inserted at any point within a document in memory.** Simply put the cursor exactly where you wish to insert the file and load it in. *Apple Writer* always adds what you load to what's already there. If you do load a file with a document already in memory, the filename on the top bar will not change, since you are simply adding to your present document. This feature can be useful for cutting and pasting very large segments of documents together.

The standard memory available for an *Apple Writer* document on a 128k or larger machine is limited to about 46k (there are patches available to make use of extended memory — more about that another time). If you wish to work with a larger sized file, it is possible to load it in parts and save each segment as a separate file.

To load a large file from the beginning, simply follow the normal procedure. When the machine's memory fills up, an error message will appear on the screen, but the file loaded up to that point will remain. Before working with this part of the file, you may wish to delete some of it from the end of the document. This will give you room to work and set off a logical starting place to load the next part in later. Remember to save the file with a different name than the original document you loaded, or you could wind up erasing the original large file from your disk!

Once you've saved the first segment of the large file, you will need to use delimiters to load the second part. After clearing the computer's memory, press **CONTROL-L** and type in the filename followed by delimiters and marker words for the point you want to start loading from, for example, /DATADISK/LARGE.FILE!Chapter Two! You can also use a second marker to define the end of the section you want to load, for example: /DATADISK/LARGE.FILE!Chapter Two!Chapter Three! If you don't specify an ending marker, then *Apple Writer* will load to the end of the file or the end of available memory. And if you wish to load a file from its beginning to a specific marker, then put nothing in the first set of delimiters: /DATADISK/LARGE.FILE!Chapter Two!

**Duplicating part of a file in memory is simple.** Press **CONTROL-L**, then type the pound symbol ("#"), followed by your starting and ending delimiters and **RETURN** (ex: #!Crops!Kentucky! ). This works with any size block of text and, coupled with the **CONTROL-X** data deletion option, enables you to move large blocks of text easily, without resorting to using the disk as a clipboard. If you press the pound sign without any delimiters following it, you will duplicate your entire document. With short letters and a small mailing list, you can copy your letter a few times and insert the salutations at the beginning of each, followed by a form feed command ("FF" at the beginning of a line) after each copy of the letter.

There is another loading trick that enables you to put special markers in a document that you can choose not to load in with the rest of the file. This works great in setting up mailing lists, a topic that will have to be covered in its entirety in a future article.

One more fascinating loading command enables you to peek at any size text file on disk without disturbing the contents of a file you already have in memory! To do this, press **CONTROL-L**, type the name of the file followed by a backwards slash ("\"), then press **RETURN** (ex: /DATADISK/LARGE.FILE\ ). To halt the file while it is scrolling by press **CONTROL-S**. Do the same to start the scroll up again. When the file reaches the end, press **RETURN** to get back your original document in memory. This technique works especially well with the split screen

option (coming next!) for comparing two documents, and it is useful for picking out beginning and ending loading markers. **WARNING:** this will not work if you eliminate the backslash as an underline token (which you will learn how to next time around). If you have changed the underline token to something else, then use that particular character at the end of your filename.

**Other control character options.** A unique and useful feature of *Apple Writer* is the split-screen option. This command enables you to look at two different parts of a document at once and jump back and forth between them. To use this feature, press **CONTROL-Y**. You will be given a prompt that asks: (Y) SplitScreen (Yes/No/RETURN=Switch)

If you wish to use the split screen option, select "Y" and press **RETURN**. Your display will divide into two parts. You can move the cursor from one window to the other by pressing **CONTROL-Y** again and pressing **RETURN**. Any changes you make in one window will be changed in the other as well — you're not working on two documents here but you have two windows onto the same document. If you answer "N" at the prompt, you will return to the single screen. As mentioned above, you can compare a document in memory with one on disk by "peek-loading" a document in this mode. You can also use **CONTROL-X** and **CONTROL-W** to delete text in one section, jump over to the other section, and retrieve your text over there.

Would you like to know how far along you are at any point in a document? Just press **CONTROL-** or **CONTROL\_** (either the minus or bottom line will work). You will be given a page and line count wherever you are and this feature will remain active until you toggle that **CONTROL** key again.

Another toggle feature is the word wrap around. That "Z" in the top bar does not mean that Zorro was there, but that the word wrap around feature is in effect. If for some reason you don't want this feature active, press **CONTROL-Z**. This will affect anything you already have in memory and it will break up words strangely. If you toggle it back on, your words will go back together, so you can toggle back and forth without fear.

**CONTROL-C** puts you in what is called the Case Change mode. Say you've just typed out a sentence, you look up at the screen and see that you left your **CAPS LOCK** on! You do not need to erase the whole sentence and type it over. Just press **CONTROL-C**, and check the far left of the top bar. If it says "L", then you are about to convert anything you pass over with the left or right arrow key to lower case. If it says "U", then you are about to capitalize everything you pass over. Toggle back and forth between the "U" and the "L" by pressing **CONTROL-C**. To speed things up you can hold down the **SOLID APPLE/OPTION** key as you go. Press any other key besides the left or right arrow to leave Case Change mode.

**[F]inding your way around.** The Find command is among the most versatile of all *Apple Writer* tools. On the simplest level, you may need it to actually look for the occurrence of a word or name. Before using the Find command, check the direction arrow on the top bar and make sure it is pointing in the direction you wish to search. Of course pressing **CONTROL-B** or **CONTROL-E** and searching through the entire document from the beginning or the end is often easier than pointing the arrow from the middle and hoping you are going the right way.

The next step is to press **CONTROL-F** and enter the exact word or phrase you are looking for between two delimiters (ex: (F)ind:/Apple II/ ), then press **RETURN**. The Find command is case-sensitive, so if you're looking for "Apple Writer" don't enter "Apple writer". Once your word has been found, the cursor will settle on the first letter of it and *Apple Writer* will ask you if you wish to proceed looking for the next occurrence of the word. If you do, just press **RETURN**, as you are prompted to do. If not, then press the space bar (or any other key except **RETURN**).

If you want to pick up the search for the same word again later, press **CONTROL-F** and the prompt will show the last string you entered.

Run your arrow key through it and press RETURN, or enter a new string if you want.

The FIND command's search and replace option is an important tool. To use it, press CONTROL-F, write the word or words you want to replace between delimiters, followed by the text you wish to substitute, also delimited, followed by either an upper or lower case "a" (ex: (F)ind:/whethur/weather/a). This will eliminate all instances of the first word and replace them with the second. If you wish to do this one word at a time, just leave out the "a" at the end ("a" for "all"), and you will be prompted if you wish to change or leave alone each occurrence of the word. Once again, remember that the search is case-sensitive. Also watch out for places where all of the characters in a word show up as part of another word. If you change all occurrences of "in" to "out", you may wind up changing words like "grin" to "grout". To avoid this, put a space before both words in your command.

If you want to replace a word with nothing, then put nothing between the last two delimiters (ex: (F)ind:/nifty//a). A nifty trick you can do with this feature is to get rid of empty spaces in a document by entering "/ //a".

There may be occasions when you will need to Find a string that includes a carriage return. Yet if you press RETURN, all you will do is enter in what you have typed up till then. In this particular case you will need to use different delimiters. The usual method is to use a "less than" sign ("<") as your delimiter and a "more than" sign (">") as the symbol for a carriage return (ex: (F)ind:<>Apple II< would find the phrase "Apple II" only when it was preceded by a carriage return.). The ability to Find carriage returns becomes especially important when using mail merge or setting up special WPL programs.

Sometimes you may wish to put "wildcard" characters in your search string. These are characters that can stand for any other character. When your delimiter is the "less than" sign, you can do this by entering a question mark for each wildcard character. For example, (F)<car????< will highlight "cartoon," "carport," "carrier," "carpets," and "car mat." Yes, wildcards can even be blank spaces!

One other useful character to include in strings is the "any length" character. With the "less than" delimiter, this character is the "equals" sign ("="). A use for it would be to fill in the blanks between two words (ex: (F)ind:<sales=month< would highlight "sales for the month," "sales below last month" and "sales goals for the upcoming month.")

If you think you may need to use any of these characters as an actual part of your search screen, Apple Writer gives you other options. Here are some delimiters and the optional characters that will work with each of them:

| Delimiter | Carriage Return | Wildcard | Any Length |
|-----------|-----------------|----------|------------|
| /         | none            | none     | none       |
| !         | none            | none     | none       |
| <         | >               | ?        | =          |
| #         | %               | &        | \$         |
| &         | (               | )        | '          |
| *         | ,               | -        | +          |

The optional characters will not work in SAVE or LOAD commands as any function but delimiters.

Next time around we will explore the wealth of options available with the CONTROL-V, CONTROL-O, CONTROL-Q and CONTROL-P commands. In the meantime play around with these and see if they don't make word processing a little bit more fun.

## Miscellanea

**Apple II purists won't be happy to learn that as of the February 1993 issue, Incider/A+, as we know it anyway, will cease to exist.** The name will change as will its focus. According to senior edi-

tor, Paul Staat, the ten-year-old magazine whose subscribership is over 95,000, will focus its attention on the Macintosh line while not entirely abandoning the Apple II. Well, we all somehow knew it was coming, didn't we?

**The new program Universe Master by Econ Systems shows a lot of promise.** This file management system for the Apple IIGs sports a very pleasing interface; the manual is well written, easy to understand and to read; and it does a whole lot more than just back up your hard drive. The software comes complete with a step-by-step tutorial that teaches a lot along the way, just as a tutorial should.

The major area of the *Universe Master* window is made up of a map view area. When a volume is selected from the online volume display area, a hierarchy map of that volume is displayed in the map view area. The Command buttons are to the left of this area. They allow you to Copy, Delete, Backup, Restore, Catalog, or Recover deleted files. The File Info button displays specific attributes of files and allows you to change some of the attributes. Finally, the Print button allows you to print the contents of selected folders using several options to customize your printout.

The manual contains an invaluable section on Data Loss Survival and an appendix that will be of exceptional value, especially for those who are newcomers to the world of large storage devices. Pages are devoted to organizing and partitioning your drives, shortcuts, file system errors, and file type assignments.

**The only problem I encountered came after I flew through the tutorial and devoured the manual.** When it came time to actually back up my own hard drive, I had a hard time finding an easy way to do a full backup to floppy disks. The program as it stands now is very well suited to perform backups to other large storage devices such as removable media drives, floppicals, or an empty partition on your hard drive. Direct support of tape drives is planned for future versions.

According to D Proni, president and major honcho at Econ, a volume spanning feature that will allow backups to 3.5 disks, will be available in version 1.1, due out in about 6 weeks. Version 1.1 will also include an optimizer. The upgrade will be free to registered owners. For more information contact Econ Technologies, P.O. Box 195356, Winter Springs, Florida 32719, 407-365-4209.

**The postings on GEnie from the people who attended the Apple Expo East made me yearn for Boston this past weekend.** There was a lot of talk about *Twilight II*, an updated screen saver for the IIGs. Econ's *AutoArc* should be out as you read this but their SoundMeister card is still in the works. The biggest surprise to me was *KinderGraphix*, soon to be released by DreamWorld Software. Apparently it has incorporated a lot of *KidPix* (Mac/MS-DOS) features. I had no idea that this was in the making but sure am glad to hear it. *Out of This World* by Bill Heineman, published by Interplay, is supposed to be out by the end of the month. Speaking of Bill, he's working on a typing program for the IIGs that involves good ole Mario of Nintendo fame. Evidently its already out for MS-DOS users. And thanks to subscriber Linda White for the informative, concise report uploaded to GEnie from which most of this information was taken!

**Westcode Software was not at the Apple Expo East but did announce via GEnie *HardPressed*,** a transparent, completely automatic way to compress your files thereby increasing your storage capacity on the Apple IIGs, due out this fall. For more information contact them at 15050 Avenue of Science Suite 112, San Diego, Calif. 92128, 800-448-4250.

**Another new product worth mentioning is *DeskTop Enhancer* by Simplicity Software.** It's an Apple IIGs desk accessory whose most interesting feature for me is a desktop alarm clock with three independent alarm settings. Each can be set for one, two, three or continuous sounds and an onscreen reminder message box. Other features include a screen saver that works in the desktop mode or with ProDOS 8 applications, a palette option to change the back

ground of the Finder and a sound option that lets you customize sounds produced by your IIGs. For further information contact Simplicity Software, 13045 Chapman Avenue, Suite 302, Orange, Calif. 92668, 714-283-3957.

**We keep hearing that the Apple II is dead but we keep getting new products as noted above.** Another good sign is the fact that Quality Computers is looking for a new member of their Apple II



## Ask (or tell) Uncle DOS

### Amplifications and corrections

When thinking of IIGs languages, I usually leave out BASIC altogether. Not because it's a bad language — it's great for poking around in — but because there aren't any good ones for the IIGs available. TML BASIC, AC/BASIC and Apple's old GS BASIC are all discontinued and unsupported, and ORCA/Integer BASIC is more a compiler tutorial than something you want to write programs in.

That leaves Micol Advanced BASIC for the IIGs, and I specifically can't recommend it. No high-level language can do **everything**, so to some degree they rely on their ability to communicate with other languages — imagine AppleSoft without PEEK, POKE or CALL. Micol Advanced BASIC for the IIGs can't talk to anything but Micol's own assembler, which virtually no one uses (in favor of Merlin or ORCA/M).

That makes Micol actually **limit** your IIGs poking around. If you play around with graphics and want to change your program into a new desk accessory, you can't with MAB GS. There's no way. Other implementations all let you talk to standard (OMF) routines in other languages, so you can borrow code from someone else to do what you don't want to. In Micol you have no choice but to use their assembler or rewrite in some other language. I've repeatedly hoped, for over four years, that Micol would change this. They've shown no interest in doing so. Until they do, you can't do anything in Micol except applications and classic desk accessories, which is really unfortunate because otherwise it's a fine BASIC.

What **was** a problem with my September article was that leaving out IIGs BASICs made me ignore 8-bit BASIC languages other than Applesoft. There are two good ones out there — Micol Advanced BASIC IIe (the limitations make little difference in the more limited 8-bit environment) and MD-BASIC from Morgan Davis, which lets you use advanced programming structures (like some in C) to write Applesoft BASIC code — adding new features while not leaving what you know behind. Both have

been reviewed in **A2-Central** in the past; both are worth your money if eight-bit programming is what you like. I should have originally mentioned both; for this I apologize.

In any case, though, we're hardly talking "cutting-edge" — BASIC was invented in Dartmouth in 1964, and Pascal and C came from different sources in the early 1970s. I'll bet you have programs on your computer written by people younger than that. :)

Matt Deatherage  
Cupertino, Calif.

### Basic emotions

Hurrah for Stephen Brunier! I am weary of computer magazines acting as if BASIC doesn't exist, or shouldn't if it does. The similarity between people "in the business" and those outside of it with current politicians and the voters is remarkable. In both cases, it seems as if the elite is totally out of touch with the rest of their society. One has the feeling that people who write computer magazines talk only to other people who either write computer magazines or sell commercial software.

My first Apple was a II-Plus. Since then I have kept II's, IIe's, and now a IIGs in constant service. Commercial software (AppleWorks with TimeOut, *Time is Money*, CADAPPLE, *Publislit!*, HowardSoft Tax Preparer) has taken care of almost all my needs and when it didn't, I wrote my own little BASIC programs.

There are many of us who choose not to learn assembly language, or Pascal, or C or anything else. We have BASIC, we like it, it does the job and we fail completely to see any reason to change. When I saw Micol's ad I figured I would give it a try. When they came out with Version 4, I upgraded and was totally satisfied. They have just announced yet another update and I expect to take advantage of it. The language is easy, instinctive and it works for me.

I hope your readers respond sufficiently to encourage Brunier to write an article.

Calvin Kirk  
Shrewsbury, Maine

I most heartily endorse Brunier's letter on page 8.71.

I'll bet you've lost readers who want to continue using BASIC. In emphasizing the far out cutting edge of all new developments, you lose those of us who are not interested in all the bells and whistles. I've mentioned this in previous letters to no avail.

By all means, print the article Brunier offers to write. If not—so long!

Frederick Assman  
Pennington, N.J.

I feel that Matt Deatherage needs a little defending from the torrid critique in Stephen

Research and Development team. They want someone with an Apple II background, solid programming and writing skills. Experience with other platforms (especially the Mac) would be an advantage. If you're willing to move to the Detroit area and feel you qualify contact (by mail only): Quality Computers, Attn: Jerry Kindall, 20200 Nine Mile Rd., PO Box 665, St. Clair Shores, MI 48080. — edr

Brunier's letter to Uncle Dos in this October issue. It appears to me that Brunier's real reason for feeling that the article is so "worthless" was only because it did not mention his product! Besides the fact that he used the word article fourteen times, the response was in poor taste.

Thanks, Matt for the great new System 6.0 software and the informative.....piece on programming.

Jeff Pulgase  
Burundi, Africa

*This is clearly a hot issue among Apple II computists. I've offered Mr. Brunier space to do an article on BASIC programming and hope that he, or one of his staff, finds the time to write it. — edr*

### HyperC update

I would like to clear up some things in your last **A2-Central** concerning HyperC. The patch for HyperC in A2Pro (file #1741) does not work very well on the ROM 01 IIGs and from what I understood before HC.System, did not work on the ROM 03. There is a file in the A2 library (file #17406) that follows all the proper tool calls and patches to make HyperC run on any IIGs. There is a Hires library that can be used with HyperC also in the A2 library (file #17407). The file is called HyperC (File #16126) and contains a better set up (from what I am told) than the one in the A2Pro library.

Gary Desrochers  
Golden, CO

### X-10 Updates

Thanks for answering my questions and printing Mr. Coughlin's very helpful reply in the August issue, page 8.55. I got a letter from Heath recently stating that they were going out of the kit business so I don't know if they would still have the X-10 unit. Now all I've got to do is find an X-10 computer interface and a 10 MHz 65816. I used to have a lot of electronics catalogs around here but that was an auto accident and two brain surgeries ago.

As far as the Floating Point Engine goes, I had problems with it after installing AE's Transwarp 32k cache upgrade. Wrote to Innovative Systems about it and got a free upgrade, one or two chips, can't remember which, installed it or them and haven't had any trouble since.

Hope Mr. Coughlin announces his updated X-10 NDA so modem-free people can buy it on a disk or a listing of it.

Frank Gizinski  
Racine, Wisc.

I have to correct your reply on the X-10 (**A2-Central**, August 1992, page 8.55). Roughly one year after I upgraded to my Apple IIGs (around 1987) I purchased both software and cable for my X-10 from David Hill, MaineFrame Software. Since then they have released one major update.

Robert Ashton  
Kelowna, B.C.

David Hill and Mainframe Software can be reached at Box 315A Cousin's Island, Yarmouth, Maine, 04096.—edr

## Sat Siri Akal

One of my Midapple User Group members wants to use a font editor to create Punjabi fonts. I have suggested to him that he should not try to reinvent the wheel and that somewhere out there in **A2-Central** land is a potential contact who may be able to help him.

Any chance you would publish this and see what happens? He is an Apple IIGs owner who wants to start a new business and wants to distribute handbills to the Punjabi community in which he lives. His name and address are Jasbinder Dhariwal, 12 Mill Hill, Smethwick, England B67 6HR.

William Watson  
Kingsford, West Midlands  
England

Hmm, any chance I'll publish your letter William? Keep watching this section of the newsletter, I'll try to find room for it. :)

Your friend's wants struck a font of wisdom over here. Ironically, Tom spent two years of his life as a Peace Corps Volunteer in Punjab. He knows what a Punjabi font would look like, but he doesn't know where to find one.—edr

## Phil's piles

On page 8.28 of **A2-Central**, Phil Shapiro refers to a file to download from GEnie called "A2.Editor.bxy." I attempted to find this file with a search of "All Libraries," and was told that no such file exists.

What gives? Is it no longer being offered? Can I get the file from you somehow?

Martin Sandry, Ph.D.  
Chicago, Ill.

I don't know what keywords you used to search, Martin, but I found A2.Editors.bxy, file # 17158, by searching for all files uploaded by Phil Shapiro. Of course, I had the advantage of knowing Phil's GEnie address (P.Shapiro1). FYI, there is a mail directory at page 200:8 where it is relatively easy to locate email addresses.

Incidentally, Martin recently sent me a copy of a program he wrote with Kenneth Peiser, a fellow psychotherapist, entitled *The Rational Person*. The basic concept of the program is to respond to given emotion-evoking situations in, hopefully, the most rational manner. The player is prompted to convey not only how they would feel but also how they would act under the given circumstances. The program potentially would teach you to make the connection between what you think and how that

thinking makes you feel. It's based on Albert Ellis' Rational-Emotive therapy. I found it to be enlightening and educational. A demo of the program can be found in GEnie's A2 library, file # 11763. For further information, contact Dr. Sandry at PsychoSoft, 2503 N. Halstead St., Chicago, Ill., 60614, 312-871-2626.—edr

## Chainability

I recently ordered a Resource Central 100 meg drive and would like advice about installation. My present system includes a Resource Central 50 meg hard drive connected to a RamFAST SCSI card in Slot 7. I'd like to know if I can daisy-chain the new 100 meg drive with the other drive. Is there a problem with the terminator?

Stephane Levy  
Chalette, France

Look at the cables that came with your hard drives. The end that connects to the RamFAST is different from the end that connects to the drive. What you need is a SCSI Peripheral Cable that has both ends like the hard drive end. Simply remove the terminator from the 50 meg drive and put it on the 100. Then use the new cable to hook the 50 and the 100 together. You can repartition the 100 meg however you like. Apple's Advanced Disk Utility on System 6 will do this for you.

If you can't find the cable you need in France, Apple dealers should have them or we can get one for you. They cost about \$25.00.

## AppleTalkin'

I enjoyed reading the article by Mr. Dennis Doms on AppleShare connectivity published in the May 1992 issue. I would like to point out one small section in his article that appears to be incorrect. Mr. Doms states that Sequential Systems in Lafayette, Colorado makes interfaces to allow connecting Epson compatible parallel printers to an AppleTalk network. I have talked to them on the telephone and was told that their hardware will connect the printers to the network, but a printer driver for the IIGs or the IIE is not available. It means the printers can't be seen over the network from the Apple IIs. They also state that they were not planning on writing such a printer driver.

Stuart F. Quan, M.D.  
Tucson, Ariz.

A call to Sequential Systems confirmed Dr. Quan's information for the moment. However, maybe Dr. Quan was not the only one requesting this setup, I was told that Sequential was working on a solution.—edr

## A Ram in Ewe's Clothing

On page 8.3 of the February 1992 issue of **A2-Central**, Tim Swihart states that "FST's make it simple to move files formerly trapped on DOS 3.3 or Pascal floppies onto ProDOS, AppleShare or HFS disks."

I have a DOS 3.3 postal chess program that I use regularly. If I could transfer this to ProDOS, I would seldom need to use my 5.25 drive. That

would be a great convenience. I would love to know how to make such a transfer!

I tried it and managed to copy all the files using the Finder. I had to change a couple of file names because they were not acceptable to ProDOS but then when I attempted to boot the new disk, I got the "Syntax Error, break in 20" error message.

I next tried loading it through Wings choosing "cold boot 5." This time I got the "This is a data disk, not a startup disk" message. I would really like further information on this process.

Phyllis R. Kuehn  
Saginaw, Mich.

FST's allow you to move files from DOS 3.3, Pascal or Macintosh disks to ProDOS; but if the files you move are programs it doesn't modify them so they can be run under the destination operating system. Lots of people are having trouble understanding this distinction.

Just as a Macintosh program won't run properly under ProDOS (or MS-DOS, for that matter), neither will a DOS 3.3 program. The FSTs allow you to see the files on a DOS 3.3 disk, and they allow you to move the files to a ProDOS disk, but they don't modify the DOS 3.3 programs.

The FST's are more useful for moving data files between operating systems than for moving program files. Even with data files, you have to be careful, however. For example, you can move a BeagleWorks document from a Mac disk to ProDOS, but there aren't any Apple II programs that can read it.—TW

Sometimes timing is everything. Phyllis. Joe Kohn recently uploaded a shareware program to GEnie called DOS 3.3 Launcher written by John MacLean. It's a simply elegant program that allows you to copy DOS 3.3 files or disks to your hard drive, convert them, and run them. It includes an option to slow the computer down to 1 mhz upon launching and return it to your normal speed upon quitting. It's straightforward and easy to use. The main thing to remember is that the disk you're working with needs to be unprotected. That part wasn't included in the docs. The file number on GEnie is 19484 and I'll bet that if Dean isn't asleep (don't worry, word has it he never sleeps) it will magically appear on this month's **A2-Central** on disk! The program is worth much more than the piddly \$10.00 shareware fee that MacLean requests for its use. To contact John MacLean directly, write to him at 2 Grant Place, St. Ives NSW 2075, Australia or on the Internet: johnmac@fawltly.towers@munari.oz.au.—edr

## Keys to Diversi-Key

My reason for writing this letter is to respond to something I read on that long file you included on the July issue of **A2-Central-on-disk** taken from the System 6 category on GEnie.

My response is to Randy Chevrier about *Diversi-Key*, which I have used since the early days of the Apple IIGs. Even before System 5, I

was encountering weird problems that led me to devise a 'creative' way of getting "Key" into memory. When System 6 came along I knew there would be trouble so I took Key out of the boot process before I even installed System 6. I have indeed had a lot of trouble with System 6, even without Key. With great trepidation, I tried putting Key back in the boot chain and lo, and behold, the only new difficulty seemed to be a system hang when I tried to disable Key from the numerical keypad on the fly while under GS/OS ((10.)) There's no problem disabling it under ProDOS 8. Since I rarely do this, it's sort of a non-problem for me.

My solution to installing Key is basically to boot into ProDOS 8, then BRUN *Diversi-Key*, then run the GS/OS boot file that is usually named ProDOS. This, of course, continues with the regular GS/OS system boot. It adds a few seconds to the boot time but not enough to bother me. In detail:

1. Rename 'ProDOS' to 'BOOT16.SYSTEM.' Any name will do as long as it ends in ".SYSTEM."
2. COPY YOUR \*/SYSTEM/P8 FILE INTO THE ROOT DIRECTORY AND RENAME IT 'PRODOS.'
3. Put a copy of a neat little utility 'System.Setup' (by Sean Nolan) in the root directory of the boot disk.
4. Create a subdirectory in the root directory named 'SETUPS.'
5. Copy Diversi-Key into this directory.

6. With *Cat Doctor*, *Prosel 16 Utilities* or *Copy II Plus* sort the directory so that SETUP.SYSTEM is the first .SYSTEM file in the directory and BOOT16.SYSTEM is the second.

7. Reboot.

I found *Setup.System* in the November 1987 issue of *Call-A.P.P.L.E.* If it's not on GEnie already, I'll be glad to upload it if anyone is interested. Actually, I think with only *DiversiKey* to install, it would be easier to follow steps one and two, then write a small BASIC program to BRUN *DiversiKey*. I came up with this gig back when I was still without a hard disk and using *DiversiCache*. I just haven't bothered to change things.

As for my System 6 troubles, I've come to the conclusion that there may be interrupt problems and memory management problems somewhere around. I have seen numerous and very frustrating FATAL SYSTEM ERRORS-911 as well as the FINDER NEEDS 300K... errors, not to mention a few ERROR 201's when launching P8 applications. (I have 4.25 megs main memory.) In my experience the latter means that some GS/OS application has left a protected direct page in zero bank but I haven't been able to relate it to any application except maybe the Finder. However, if the Finder was not cleaning out all of bank 00 after itself, the message should come up every time you leave the Finder and it doesn't.

I've tried many different combinations including buying a power supply with more muscle. The power supply did seem to help some. Getting rid of the NDA *Menutime* has seemed to help although I don't think this is the culprit alone. Additionally the CDA *Disk Witch* seemed to cause problems. I've taken out most of the frilly inits. Is it possible that certain combinations of DA's and inits make the fuses blow, rather than any specific ones on their own—perhaps those that rely heavily on interrupts? I hope that you can keep the dialog on this going; much as I'd like, I can't afford the A2 BBS's.

Bill Robbins  
Hiroshima, Japan

Two short fixes for System 6.0 surfaced some time ago on GEnie that may alleviate some of your problems. *FixFindFile.bxy* (file #18369) fixes a bug that caused ProDOS 8 applications to crash giving a \$201 error after using *FindFile* NDA. There is also *SoundPatch.bxy* for the Sound Control panel (#18218) that fixes a problem with Shutdown sounds. And while we're on the subject, two of my recent very favorite inits for System 6 are *File #19428, Making Copies* and *File #19335, Oow.bxy*. -*Making Copies*, (two guesses when this one is used) by Bryan Pietrazak, is a sound from a Saturday Night Live skit and *Oow* is just a silly little sound and action programmed by Mike Nuzzi that occurs when you launch a file from the Finder. --edr

## Last month in A2-Central-on-disk:

Directory: /A2.ON.DISK.9210/

| Filename        | Blocks | Description                             |
|-----------------|--------|-----------------------------------------|
| GENERAL.STUFF1  | 1      | Stuff for all Apple II systems          |
| APPLEWRITER.2.1 | 153    | Apple Writer version 2.1                |
| AMWR.GS.PATCH   | 25     | Patches Apple Writer to print on IIgs   |
| AMWR.MISC.HELP  | 36     | Apple Writer help files & cheat sheet   |
| CREATURE.FIX    | 10     | Fix for bug in Creature Feature game    |
| HYPERC.LANGUAGE | 192    | Hyper C, an 8-bit Apple II C compiler   |
| HYPERC.DOX      | 13     | Basic documentation for Hyper C         |
| HYPERC.MAKESTS  | 64     | Make Hyper C programs ProDOS STS files  |
| HYPERC.GS.PATCH | 4      | Patches Hyper C compiler to work w/IIgs |
| IIGS.STUFF      | 1      | Stuff for Apple IIgs computers in here  |
| BOUNCFERNO.DOX  | 12     | Documentation for Bouncin' Ferno        |
| BOUNCIN.FERNO   | 900    | Exceptional IIgs marble game            |
| FILE.A.TRIX.1.0 | 53     | Exceptional file utility in a CDA       |

## Last month in Studio City:

Directory: /Studio.City.17/

|                        |     |                                                                             |
|------------------------|-----|-----------------------------------------------------------------------------|
| 1st Impressions2       | 270 | 2nd of B.Lynn's series. Adding sound & animation                            |
| Stupid Button Tricks 5 | 342 | 5th installment of Bill Lynn's ultra cool anim. buttons                     |
| Flight School          |     | Informative beginners stack                                                 |
| Master stack           | 15  |                                                                             |
| Slave stack            | 18  |                                                                             |
| HyperArt               |     | Use in your stacks                                                          |
| Animals.1              | 41  | by M.A. Trzyna                                                              |
| Central America        | 23  | by Michael Giltzow                                                          |
| North America          | 30  | by Michael Giltzow                                                          |
| South America          | 26  | by Michael Giltzow                                                          |
| Trains.92              | 35  | by M.A. Trzyna                                                              |
| HyperSounds            |     | For use in your own stacks                                                  |
| Car.Skid               | 60  |                                                                             |
| Car.Start              | 67  |                                                                             |
| Close.Drawer           | 10  |                                                                             |
| Gurgle.Click           | 31  |                                                                             |
| Kickoff                | 3   |                                                                             |
| Whistle.Ref            | 117 |                                                                             |
| NFL.Quiz               | 654 | NFL trivia with lots of Simple Script use                                   |
| Painting.Stack         | 391 | Excellent tutorial by M.A. Trzyna for creating your own graphics for stacks |

## A2-Central™

© Copyright 1992 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:

Ellen Rosenberg

with help from:

Denise Cameron  
Dean Esmay  
Jeff Neuer

Dennis Doms  
Richard Ginter  
Jean Weishaar

Sally Dwyer  
Mary Johnson

*A2-Central*, titled *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): \$34 for 1 year; \$60 for 2 years; \$84 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 and 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 \$90 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 20 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 drivel 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 unfilled 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