

The Best of Ottalini #7: Bonus Content

(ORIGINAL DISK NAME: THE BEST OF OTTALINI PLUS!)

////////////////////////////////////

/// SIG, Washington Apple Pi

12022 Parklawn Drive

Rockville, MD. 20852

(301)-984-0300

WELCOME!

WAP /// SIG Public Domain Library Disk

Category/Number : WAP Articles/3WAP-08

Disk 3WAP-08 is the second of 1992's "Best of Ottalini" disks. There was so much material available in 1992, that we were able to fill a second 5.25" disk for you.

Wow! We had so many articles from 1992 that we had to spill them over into a second disk. 3WAP-08 includes the /// SIG Public Domain Library articles published in the WAP Journal during 1992 along with any and all articles contributed by other members - including Paul Campell and John Lomartire. Lots of great stuff to keep you going into the new year!

WAP /// SIG PUBLIC DOMAIN LIBRARY

PDS NAME: THE BEST OF OTTALINI PLUS!

DISK ID : 3WAP-08

ARTICLES

Mainframe Batch Printing on the Apple ///

LaserJets and legal terms: The Apple /// at work

IBM: Should You Make the Switch?

Updating the Pascal MENU.MAKER Program

Apple /// Systems

On Disk Drives and Apples

Dealing with Driver Space Limitations

An Open Letter to John C. Dvorak

Apple Hardware Lubrication

Designer Disks for the Apple ///

Some Tricks of the Trade

New SARA Adventures

SOS vs. MS DOS

A Brief Review of StemWriter

The Apple StyleWriter and the ///

The Apple /// System Configuration Program (SCP)

Little-Known Utilities for the System Utilities

Apple // Family Used Computer Guide (May Prices)

Apple // Family Used Computer Guide (July Prices)

PD.LIBRARY

/// SIG PD Library additions by various months in 1992

Created by: ED GOODING on: 05/22/1992 07:36:07

It's late at night and you've been working long and hard on a Three Easy Pieces database used to manage records for a client. You know that your client is expecting 12 different reports, all of them longer than 30 pages, all of them sorted differently... and all of them done by morning.

If you are like me, you do not look forward to punching out a report, then waiting until the printer catches up before sending another. A print buffer helps in situations like this, but 12 reports that are 35 pages long add up to a total of 420 pages.

A page of text on the Apple /// averages about 2K in size, so a 256K print buffer will only hold about a fourth of the reports, leaving you to wait for the rest. Although the buffer would certainly make things go faster, wouldn't it be nice to do all the reports in 10 minutes, then give a simple command to print all of them while you sleep?

Printing from the wonderful world of SOS

The Apple /// is the most device-independent computer I have ever used (and I've used a lot!). All devices connected to the /// are controlled by device drivers which are loaded when the computer is turned on or rebooted. This feature of the SOS operating system allows your computer to print on virtually any printer or disk drive.

What does this have to do with your 420 pages worth of reports? Glad you asked! I don't know how fast your printer is, but a typical page on my Panasonic printer takes between 20 and 30 seconds in draft mode. This translates into a 2.9 HOUR print job for 420 pages in draft mode, and over 6 hours in letter quality.

The solution to this wait is using a subdirectory as a print queue, then printing to this subdirectory on your disk drive. This subdirectory could be called .PROFILE/PQ, if you have one of the new hard drives from On Three, you might want to place this on one of the different partitions. Three Easy Pieces (E3P) will allow you print a report to the printer, to an ASCII file, or to a disk drive as a formatted ASCII file.

Printing to disk takes place at the rate of 1.1 seconds per page for a time of 7.7 minutes for all 420 pages, much better than the 6 hours required for the printer. The formatted ASCII file output is the key. When Three Easy Pieces prints to disk in this mode, it also prints the file on disk with all of the margins, indents, headers, footers, and page numbers intact.

If you do not have your copy of Three Easy Pieces setup to print to disk, follow the instructions in the manual for adding a new printer, choose your printer type, then choose "To Disk". Set everything else like you would for your actual printer.

Now, you have made a subdirectory, sorted and printed all those reports to that subdirectory on the disk drive, and you are ready to give that simple command so you can go to bed and wakeup to a neat stack of reports. If I could only remember what it was, I would tell you...

Just kidding!

Apple /// System Utilities: Still Doing What MS-DOS Can't

System Utilities can do all the neat things it is supposed to do, like delete, copy, lock, rename, or do most anything with files and volumes. It also has a few undocumented functions, one of which will let us print those reports that we created in Three Easy Pieces.

To print your reports from System Utilities, go to the file handling menu, choose the copy file command and type the pathname of your subdirectory followed by a slash and an equal sign, then press return. The program will then ask you for the file destination, you should respond with the device name of your printer, press return, and make sure your printer is ready with plenty of paper!

System Utilities will ask one more question before printing all those reports; do you want to add a page of directory information before each file? If you answer no, it won't. If you answer yes, it will print 10 lines containing the date, time, and the filename before each file.

The Apple /// System Utilities will now begin printing every file in in your new "Print Queue" subdirectory regardless of size, and you can go to sleep. One thing to think about as you curl up under the covers, is that this method will also work for word processing and spreadsheet files that have been printed to disk.

It is possible for an author to create a book 8,000 pages long, hook the Apple up to a high capacity printer, and print the whole book in one shot. A word of caution; you might want to experiment with some smaller files to get the hang of this method.

Once the printing starts, it can only be interrupted if the Utilities program cannot find the source file. You can cause this to happen by opening the floppy disk door, or by using a Desk Accessory to alter the pathname. You do not have to worry if the paper runs out or the printer is taken off-line, System Utilities will wait patiently until the problem is corrected.

In using my print queue, I have never had to interrupt a job, so you will probably never have to use the emergency procedure information just mentioned, but like the saying goes, "It is better to have it and not need it, than to need it and not have it". That about wraps up this report, my SARA has a long night ahead of her, but as for me, well... sweet dreams!

Paul Campbell

An attorney and I were deep in conversation one day discussing his plans to open a new law firm in Detroit Michigan. "One of the most important things is my word processor, without it, all document production stops, and so does the office" he said.

We began to talk about computer hardware and he made it clear that one of the biggest problems with his former Wang/WordPerfect system was the destruction and loss of important documents. I thought for a moment then said "I think I know a system you might want to try".

The decision was made to purchase an Apple /// with 256K of RAM, a clock, a fan, a monochrome monitor, and a Profile hard disk. The owner went looking for software after the system was installed and came back to me looking disappointed.

It seemed that everyone was recommending AppleWorks for his needs, but nobody had a version for the Apple ///. I smiled and told him that /// Easy Pieces was basically the same as AppleWorks. Before long, the office system was up and running with /// Easy Pieces and the Desktop Manager.

The next project was hooking up a Hewlett Packard (HP) LaserJet II to the system. Believe it or not, there is actually a section in the LaserJet manual devoted to the Apple ///. It uses the standard RS232 device driver and gives the Driver Configuration Block (DCB) information that is needed.

The printer was previously owned and the switches for setting the serial port had been changed, so the device driver did not work the first few times around.

A few modifications to the DCB of the device driver solved that problem, and the driver was renamed .LASERJET so that any communication programs would not mistake it for the modem device driver.

In a strictly business environment, the Apple /// performed better than most people would expect. It was, after all, originally marketed for this type of use. However, there were a couple of limitations.

The first problem was the way EZP handled large directories. With all of the company files kept in the same subdirectory, EZP seemed to take forever to list them on screen because it must scan for EZP files then sort them by type, and also sort the filenames alphabetically.

Bob Consorti of On Three remarked that EZP should have been written so that it would keep a list of pointers (information about where files are located) to the files. These pointers would be updated when files were added or deleted.

This would eliminate all the extra processing every time you went to add a file to the desktop. In order to compensate for this problem, the company files were split into categories with one subdirectory for memos, one for letters, one for finances, etc.

This worked well for a few weeks, but after a while each subdirectory accumulated enough files to make the wait for directory listings too long again. Remember, this is a business that creates four to seven new files per day, some of which are thirty to forty pages long.

We tried copying old files to another directory then deleting them, but this was troublesome, and worst

of all, it was what I call a "Band-Aid Solution"; a temporary fix to a continuing problem.

The solution was to start off each month with a fresh subdirectory. An example would be /PROFILE/DATA/JAN91. Files like mailing lists and spreadsheets that remained in constant use were placed in another extension (a sub-directory added to the current directory) of /PROFILE/DATA/.

The numeric keypad was configured to change subdirectories with a single keypress using the Macro Manager from On Three (which was very easy to adjust to), and looked very impressive as the Apple /// flashed around the EZP menus faster than anyone could follow!

A printer change came following the demise of the LaserJet's photosensitive drum. The price of a replacement drum sent the owner shopping for a new printer. Our choice was the Epson Action Laser printer.

The new printer hook up went smoothly enough, until it came time to set the serial port on the printer. The owner's manual for the printer did not make it clear that you had to turn the power off after every time you change the serial port configuration of the printer.

The required printer cable was 25 feet long, which was far too long for a parallel port. A parallel port is only usable with cables shorter than 10 feet. This meant that we had to use the serial port or move the printer next to the computer.

The same RS232 device driver that worked for the HP LaserJet was used with further adjustments to the DCB and we were off and running.

For you experienced Apple /// users, the DCB for the Epson Action Laser is as follows:

```
0=0E, 1=00, 2=00, 3=00, 4=FF, 5=80, 6=13, 7=11, 8=00, 9=00, A=00, B=00.
```

The serial port on the printer itself was set to 9600 baud, 8 data bits, no parity, X-ON/X-OFF. Don't forget that changes to the printer itself will not take effect until the next time the printer is turned on.

Laser printers normally cannot print all the way to the edges of the paper. This non-printable margin is smaller on the Epson than it was on the older Hewlett Packard, but still enough to require compensation in order to have accurate text margins.

We found it less confusing to change the platen width within EZP and maintain margin settings that made sense, as opposed to adjusting the margins to odd settings to compensate for the non-printable areas.

Because mailing label output was needed from the EZP databases, an Apple ImageWriter DMP printer was also configured into the system.

The standard ImageWriter device driver was used to send output to the same built-in serial port as the laser printer. The output is switched between the two printers as desired.

How is the Apple /// doing a year after installation? The owner is still very pleased with the system and recently purchased another Apple /// for doing business at home.

This home/office combination would be the perfect setting for a pair of On Three's 45 Megabyte removable SCSI hard disks. These new drives make an incredible change in the character of the Apple

///.

I have identical test files on my Apple /// at home and an IBM PS/2 used at work. Even though the PS/2 is a well-respected computer, it could not keep pace with the /// in loading and saving files, and got hopelessly left behind.

Imagine having all the above mentioned storage and speed, then being able to pop the hard disk cartridge in your pocket and take it home!

The Apple /// hardware had already been chosen before the new hard disks were announced, so the office system uses a 20 Meg Sider, and the system at the owner's home runs on a 5 Meg Profile.

The Sider drive is split into two partitions, with the applications ("Apple-cations?") and The Desktop Manager on the first partition, and all data files on the second partition.

Partitions separate data storage areas on the hard disk, and makes the computer think that each partition is a different disk.

Why the partition?.. Glad you asked. The Apple /// can actually handle volume sizes of 32 megabytes, but the Pascal language system (The programming language that a lot of programs are written in) is limited to volume sizes of 16 megabytes

It would not make sense having a hard disk that cannot be accessed by System Utilities, CustomFont, Draw On Three, or any other Pascal software.

As we bring this article to an end, the company owner remarks that the only thing he misses is the ability to print in the background while working on another document in EZP. Perhaps a print buffer would solve that problem.

Another word processing package called StemWriter /// does printing from the background, and nearly everything else too! (Except wash dishes...)

This program can also spool files from a disk in the background and allow you to edit 8 other documents at the same time using 3 windows.

The output produced by the pair of Apples is professional looking, and with the cost of software so low it's a hard system to beat.

The Apple /// meets all their professional needs for hundreds of files. The computers are easy to use, and they have not lost a single file to date... something that could not be said of the former system with WordPerfect.

Paul Campbell

Should You Make The Switch?

*By Paul Campbell
WAP /// SIG*

There has been some talk on the local boards (in Detroit) about people thinking of jumping ship to IBM. The main reason is the cost of upgrading Apple][s. It's downright expensive! One of my replies to such a person is below:

I know what you mean about aftermarket equipment prices for Apples. I saw an ad for a 386SX with an 85 Meg hard drive, 1 Meg RAM, etc., etc... for less than \$1,300. The 70 Meg drive for my system cost over \$1,400 new. Another problem I have with Apple is user abandonment - I find it very hard to recommend to anybody a system that will probably be an orphan soon. It's sort of like buying a new car, then when the next model comes along you cannot get parts, service, or advice from the company that made the darn thing, they won't even talk to you! Despite all this, I still love the way even an old

Apple (in the right hands) will walk all over most IBMs in a lot of situations.

Before anyone jumps to conclusions, let me state that I am a full-time computer operator and work not only with a really sweet IBM mainframe and others like DEC (Digital), VAX, and Data General, but also with dozens of PCs.

I also do some computer repair on the side and have run countless tests with different systems back-to-back. I know that there is no way my old 6502B can come close to a 386SX in raw speed, but if a system can generate and kick out reports and documents just as fast, with three times the ease of the average MS-DOS system, why change? It took MS-DOS until 1991 to evolve into an operating system that came close to the level of Apple /// SOS, an operating system that introduced "Device Independence" before Unix was even created (I currently run SOS 1.3(S)).

I have run an Apple printer, a Panasonic printer, and an IBM printer all at the same time. Another reason I stick with my /// is the track record for system integrity. For 1991; zero system failures, zero CPU downtime, zero disk errors, and despite the fact that the Apple /// is used to manage over a thousand separate files, zero lost files, and zero corrupted files.

As long as a computer can launch an application with a single keypress, perform sequential batch downloading from a mainframe, sort a database faster than a PS/2, do macros that can jump from program to program and back again, output 8,000 pages of text with a simple command, and keep up with an "I can't see her fingers" speed post, there is not really a good reason to toss it in the closet just yet."

Updating the Pascal MENU.MAKER Program

*By C.M. Davidson
Burlington, Ontario Canada*

It has always been a disappointment to me that the /// SIG's excellent Pascal Menumaker program (on disk 3UTL-31) pays no attention to FOTOfiles, so ably handled by the Basic Menumaker. The reason, I suspect, is because Pascal knows nothing of FOTOfiles, handling its graphics instead as PasData files.

So I have tampered with the Pascal Menu program to the extent of having it display FOTOfiles until the <esc> key is pressed, then returning to the menu.

The modified program is called NUFOT.MENU.CODE, and the source is NUFOT.MENU.TEXT. I have had to modify the original CONSOLE.IO library file to CUNSOLE.IO, and both .CODE and .TEXT files are also there. The other library file, LIST.CAT, needs no changing. Some further details are given as comments on the TEXT files.

Although one can always use the old Pascal Filer to Alter a PasData filetype to FOTO, this has become something I find tiresome. I have tampered with the Basic Filetype.inv module to have it behave like a Pascal assembly language procedure (FTYPE.ASM), and round it I have built a rudimentary program (TYPESET) to change filetypes from within Pascal.

No doubt somebody with more skill than I have could polish it up, perhaps to show a list of filetypes on request. An advantage of this is that one can incorporate the procedure in a Pascal program such as the Chaos ones I sent earlier and thus force the Pascal system to give birth to a FOTO file which can then be readily examined using the new menu.

/// SIG Note: Mr. Davidson has been a long-time supporter and contributor to our /// SIG. You'll find his ALLDUMP Invokable that allows you to "dump" a graphics file to your printer, along with a Business Basic paint program on disk 3MSC-02. His Downfont program on 3MSC-04 allows font downloads to Panasonic printers along with FONTFLIP, a modified version of the Jeppson character (font) design program. The CHAOS programs in Pascal mentioned above are on disk 3PCL-09. Finally, you'll find his modified version of Tom Bartkiewitz's Pascal Menu.Maker program on disk 3PCL-17. //er Barbara Dudmanhas also modified Pascal Menu.Maker, as well as the ALLDUMP.INV Invokable.

They are on disk 3MSC-15.

*By Dave Ottalini
WAP /// SIG Co-Chairman*

Ever wondered how other ///ers put their systems together? We've heard a lot from our good friend Paul Campbell in Detroit about his system (and seen it as well at SIG meetings). And I've told you a bit about mine – which includes a 512K ///+, 800K disk drive, three 5.25" drives and a lowly 5MB Profile.

Recently, C.M.Davidson, a long-time ///er in Canada sent me a disk of information about his SARA and what he had attached to it. I thought you'd enjoy hearing how he's configured his Apple ///:

EQUIPMENT

Apple /// 256K, with 2 external Disk ///s and Corvus Hard Drive (20 Meg, subdivided into 1 Meg device .C1 as system controller, and two approx. 10 Meg devices .C5 and .C6, each accessed by a separate Catalyst Boot Disk, but capable, when running, of accessing the other (e.g. my 3EZP, AppleWriter, VisiCalc and CustomFont systems are on .C6, but all my EZP, AppleWriter, VisiCalc files and Fonts are on .C5).

The contents of .C1 (file Catlist.C1) may be of interest. All my SOS files (KERNEL, INTERP and DRIVER) for my .C6 Catalyst boot-up reside there; my Boot disk contains only the Block 0 code and VCSWAPPATH.DIF for VisiCalc (There must be SOME way to make Visicalc look to my Hard Disk for THAT file, but it still eludes me). This makes for a reasonably fast (18 sec.) boot-up. All this is made possible by a modified SOS.KERNEL which Scott Stinson prepared for me.

Installed peripheral cards are: UPIC and Corvus. I have a CP/M card, but took it out (no obvious use for it). I also have an Apple //e Mouse half-card, but my system refuses to boot when it is installed (some voltages drop too low at boot-up, I think). The system supports a 2400 baud Modem and a Panasonic KX-P1091i dot matrix printer.

One of the interesting things I learned about just by looking at the files he sent concerned a great way to set up a /// EZ Pieces Data Base file (and which, of course, applies to AppleWorks or other data base programs as well). If you always wished there was an easy way to pull a catalog or even a document into a DB program for faster sorting, all you have to do is make a new data base from a text (ASCII) file with just one field. Each line of your text will be loaded and then you can do finds or sorts as you wish. I think this may be (as Mr. Davidson proved with catalogs of your disks. Mr. Davidson used Power Cat III (disk 3UTL-15) to first create a catalog of his hard disk and then pull it into 3EZPs for easy searching.

In any case, we'd love to hear more about your system and how you have it configured. Or any tips or ideas you've tried and found to be successful on your /// would also be welcome. Write them down and send them to me care of the office and we'll share the wealth!

*By Paul Campbell
WAP /// SIG*

There was some conversation on the local networks about disk drives and Apples. One person was claiming that the][GS could not use the extra speed of any disk drive faster than 28Ms access time. After a post from me stating that this was not entirely true, the person also made derogatory remarks about our SARA, which made it difficult for me to remain silent (of course!). Here is my response:

Since I happen to have both a SCSI Turbo Drive and a RAM disk, I thought it would be interesting to run a few tests to confirm the information you apparently got from someone else. The first test was to take a large file, break out the stopwatch, and load that file from an 800k floppy which resulted in a total loading time of 47 seconds. 3.5 inch drives are relatively slow which is why I don't use mine except for backups and other infrequent stuff.

The next test was to load that same file from my D7T turbo hard drive which has an average access time of 28ms, resulting in a time of 19.7 seconds for the same 108.5k file. This translates into 5.5076142131979695 kilobytes per second. According to the information you received, this is the maximum speed for transferring data into the computer's memory, no other device will exceed this rate because this function is directly dependent on CPU clock speed.

The third test was to load the file from the Apple /// RAM disk. This device is NOT SCSI, and is also limited by CPU clock speed while it transfers data bit-by-bit. The test file I used happens to be the entire CustomFont user manual including the Introduction with 11 chapters, appendix A,B,C,D,E,F, and charts. The loading time from the RAM disk was 15.8 seconds.

I realize that you may have never seen/used an Apple /// or][GS. To give you an idea of the speed difference between one of these and an Apple compatible, the same search/replace function that takes my own Apple compatible 27.41 seconds to complete, is finished by my Apple /// in LESS than 1 second (0.82 sec). Another example is a back-to-back spreadsheet sorting test between an IBM XT and an Apple ///, the IBM was supposed to have the newer technology, but the times were; IBM - 11 minutes, Apple - 14seconds.

As mentioned before, the data transfer speed of a SCSI drive is dependent upon CPU clock speed (and interface card in some cases), the Apple /// can easily take advantage of access times faster than 28ms, and the][GS has a much faster clock speed than the ///.

Me thinks you need to double check the rumors that people off the street are spreading before making assumptions!

Dealing with Driver Space Limitations

*By Stan Shabasson
Apple Three Users of Northern California (ATUNC)*

*Edited By Dave Ottalini
Wap /// SIG*

When I first bought my /// I had a driver that was just large enough to do certain jobs. Each program has a set of particular drivers in order for to operate. Since purchasing my ///, I have tried to have a driver file as large as possible in order to handle certain jobs.

Later, I bought a PROFILE and was told that if I wanted to place my MASTER Programs on the hard disk, I would have to use Catalyst to dynamically load drivers needed by a particular program. This was "BSE" - Before Selector (which does not dynamically load driver files).

As I loaded my .PROFILE, I found that certain programs needed a particular driver and decided that I should have those drivers on the boot disk.

I then decided it would be easier to have a floppy disk dedicated only to drivers. Here's how you can put together your own similar disk:

Using System Utilities, format a disk in .D2 (the external disk drive) as /DRIVERS.ALL.

Using the System Configuration Program (in Sys Utils) do the following:

- (1) PRESS (R) Read a Driver to list all drivers configured in the SOS.Driver file in .D1. (the System Utilities driver file or other program Driver file of your choice.)
- (2) PRESS Escape
- (3) PRESS (D) Delete a Driver... Delete all drivers except one.
- (4) PRESS Escape
- (5) PRESS (G) Generate a New System...A System Generation will take place (if there's no .Console, you'll get an error message but keep going.)
- (6) Press (S) Save each driver as .d2/Console,.d2/Audio, etc.
Repeat steps (1) thru (6) until .d2/DRIVERS.ALL is loaded with all the drivers required for your programs
- (7) PRESS Escape.

All DRIVERS should be saved on .d2/DRIVERS.ALL as driver files. The list of files should look something like this:

```
/DRIVERS.ALL  
    /Console  
    /Audio
```

```
/Grafix  
/Printer, etc....
```

Again, disregard all 'warning flags'...

Now that all drivers have been copied to /DRIVERS.ALL, A NEW SOS.DRIVER file can be generated for any boot disk configured with the proper driver, by doing the following:

Again use SCP and:

- (1) PRESS Read a Driver and change .d1/SOS.DRIVER to .d2/Console. Hit RETURN and it will load the driver, showing:

```
.Console
```

- (2) Press Escape
- (3) Press Read a Driver

Again change .d1/SOS.DRIVER to .d2/AUDIO as you add a driver. This way the READ A DRIVER FILE will start to fill up in the order you read them in.

```
1. .CONSOLE or 1. .MOUSE  
2. .AUDIO or 2. .CONSOLE  
3. .GRAFIX or 3. .AUDIO  
etc...
```

Each SOS.DRIVER can be configured in any way you want.

The size of the SOS.DRIVER file normally is limited to 51K when using SOS Version 1.3; however, if you have the ON THREE Desktop Manager, it will increase beyond this limit. You cannot use the SCP normally without de-installing the Desktop Manager driver from your SOS.Driver file (it is possible to load it if you have a 512k machine but not always).

Using my other Stand Alone starter disks, SOS.DRIVER has been built up to 51K, without creating a memory problem.

Note that WAP /// SIG Disk **3UTL-04** contains two sides of Apple /// drivers in the format indicated above (that is each is in a separate file you can use to create your own SOS.DRIVER file).

*By David Ottalini
WAP /// SIG Co-Chairman*

Dear Mr. Dvorak;

I'm writing to say that I enjoyed your "What Ever Happened To..." article about the Apple /// in the July Tech Section of Computer Shopper.

But I would like to contribute some of my own comments to your column, as well as clear up some inconsistencies. Basically, I just want to bring you up to date about what's happening with the Apple /// in 1992.

I have no real arguments with your history of the ///. Apple made major mistakes by thinking the Apple // was on its way out and that the /// would be its replacement. Bottom line here was that a very young company intent on building its first product as Apple Computer didn't bother to do any real marketing or find out what its customers and dealers thought. They decided to design a product without any research.

SARA was rushed into its introduction at the National Computer Conference in May 1980 before all the bugs were worked out. Everyone wanted to see the /// and its introduction was a huge success (helped along with free tickets to Disneyland). But Apple soon discovered that it had, in effect, a vaporware product that was not ready for production.

Steve Jobs refused to let designer Wendell Sander put a fan in the ///. That meant the case had to be designed, in effect, as a giant heat sink, cast by a company that made engine blocks somewhere in the midwest. VLSI (Very Large Scale Integration) was unknown at that time and the ///'s motherboard has got a lot of chips on it. That resulted in lots of heat, something not even the ///'s die-cast frame deals with well. Add poor manufacturing to start (as you mentioned) and that spelled major troubles and a loss of valuable time to the competition.

I frankly don't think you gave Apple enough credit for replacing every single defective mother board in that initial run of 14,000. I don't know of any other company that would have done that. IBM certainly would not. I also think you give too much weight to the detachable keyboard and its contribution to the ///'s problems. It would have been nice, but most ///ers don't miss it that much.

You touched on the real issues in your column: the IBM PC and Apple's own marketing problems. For example, Lotus 1-2-3 was being developed on the /// but dropped when the PC was introduced. Why didn't Apple work with Lotus to make sure that killer program was finished for the ///? As for marketing, Apple neglected to sell the /// to its own sales force. They knew little about it and less about how to operate it. Don Williams finally brought them all together into a room and made each salesman put together his own /// to force the issue.

Further, Apple did not support its dealers like it should have. The /// was the most powerful personal computer on the market when it was introduced. But its operating system was so flexible that it could be confusing to use. Apple provided plenty of written materials, but little real training so that dealers

could themselves feel comfortable selling SARA and setting her up properly for customers who wanted to purchase turnkey systems. When I bought my new ///+ in fact, I knew more about how to operate it than the saleswoman did.

As for SOS, it is, in fact, a powerful assembly language program written and assembled using Pascal. It is not a P-Code program or a DOS-like shell. You mention that SOS used a menu system that turned out to cause more problems than it solved. What I think you mean is that the early versions of programs like System Utilities (which was written in Pascal 1.0) quit to the Pascal command line but could not actually "quit".

SOS itself had nothing to do with that. Later versions of Pascal-based programs simply quit to a "Reboot" message on screen (the Pascal Menu being an option). Unfortunately, Apple did not (as it did with ProDOS) include a way to get to another program after quitting except by physically rebooting with another disk in the internal disk drive. That was taken care of by Quark's excellent hard disk management system - Catalyst and later, Selector ///. They both allow users to load all their programs on a hard disk and then select them from a menu. Once finished, the user can return to the menu and continue working without having to reboot.

SOS remains one of the most powerful operating systems ever written for an 8 bit computer. Apple thought so much of it that they rewrote it into ProDOS for the //e and ProDOS 16/GS-OS for the //GS. Files are completely transferable between those machines.

The Lisa Operating System drew heavily on SOS and Apple just recently refused to place SOS into the public domain because "The Macintosh Operating System was developed from SOS." Frankly, SOS still provides more flexibility than any of the versions of ProDOS. Its unique bank switching technology allows use of up to 512K of memory (which many ///ers have installed thanks to a third party vendor and is still available.) It allows complete control of all devices by pathname or device name, something few other computers can do. It uses interrupts that make possible background utility programs like On Three's Desktop Manager which, unlike on the //, can be used from within any program at any time.

Had Apple designed the /// motherboard better, it could make use of almost any Apple // peripheral card on the market. The case/motherboard is simply too small to accept some cards. But for those that are small enough, all that's needed is a device driver to access them. Today, ///ers routinely use 800K (and soon 1.4MB) drives, mice and trackballs, SCSI hard disks, 9600 baud modems, laser printers and more.

One thing Apple did get right was the ///'s keyboard. It is STILL the best keyboard available for any Apple computer. Period.

Apple did make a major mistake by not upgrading SARA's emulation mode. Interesting isn't it that Apple believed the /// would succeed the //, yet cripple it by not providing a 100% upgrade path. Apple never did provide more than a 48K //+ emulation, even for its final ///+ machines. It was left up to another third party vendor, Titan Technologies, to develop first a single card to allow up to 128K for //+ programs and later, a 128K //e emulation using two cards. Those cards are still available today and combined with a 65c02 CPU replacement on the ///'s motherboard allows the use of ANY Apple //e program, including Publish It! 4, the New Print Shop, etc. The RAM on the Titan cards is available as a RAM disk in /// native mode.

Software continues to be written and sold commercially for the /// and there are active /// user groups

on both coasts. In fact, in Detroit, we have a member who sells ///s to lawyers and other professionals and regularly delights in running rings around MS DOS machines with his 12 year old orphan.

Was the /// the wrong machine at the wrong time? Apple did not do its research, failed to have a real vision for the /// in light of competing demands made by the Lisa and Mac lines, rushed the machine into production and had to waste a great deal of time making good on damaged goods. Apple could have simply improved the ///s hardware while providing ProDOS as an upgraded Operating System. Instead they decided to bring out a more powerful machine that came *this* close to being the right machine at the right time. I think it was simply inexperience that pushed things the other way.

You must know Mr. Dvorak, that by the time production on our SARA ended, Apple actually had a winner on its hands. All the bugs were gone and the machine worked like a champ. And the /// was turning a profit. But the computer world had by that point bypassed SARA. Those of us still using her don't care, though, because we still know what a great machine we have. It can still do many of the basics we need done. Maybe not as fancy as the Mac or a PC with Windows. But then, we can get along just fine with 256 or 512K and a 5MB hard disk. Our software has been elegantly written to get the most out of a small amount of memory and doesn't cost all that much either. We have a tremendous Public Domain software library from which to draw as well.

There are many of us, Mr. Dvorak, who still serve proudly aboard the Good Ship Apple ///. If you need a picture, just let us know!

*By Tom Linders
ATUNC (Apple Three Users of Northern California)*

This article is about lubrication and what I recommend for computers. The following are in my stable of things made for cleaning and lubrication:

- Tuf Oil
- Tuf Grease
- Turbine Oil
- WD 40
- 99% Pure Isopropyl Alcohol
- Platen Restorer

I use the 99% pure Isopropyl Alcohol as a cleaner for the following:

- Cleaning Disk Drive heads (always wipe dry to remove the film)
- Cleaning Disk Drive Rails
- Cleaning Disk Drive Pullys
- Printer Rails
- Keys on the keyboard that are sticky

There are other alcohols that I do not like to use. Rubbing Alcohol has about 30% water, which is bad for metal parts. Denatured Alcohol may use something as the denaturing agent that is bad for the part you are cleaning, like MEK or Trichloroethane. Unless you are aware of the denaturing agent, don't use this type of alcohol.

I use Tuf Oil as a super lubricator for hard metal parts. Always remove any excess oil, as it attracts dirt and dust:

- Disk Drive rails: one drop
- Printer rails: one drop
- Roller or ball bearings

I use Turbine oil as a lube for porous bearings:

- Disk drive motors and
- Furnace Motor bearings

I use WD 40 for the following:

- Cleaning and lubricating printer heads
- Re-inking printer ribbons

I also use it for general cleaning up including the disk drive and printer rails in lieu of alcohol as a cleaner. But I always clean it off and then use the Tuf Oil. I never use oil or WD 40 on sticky keyboard keys as it attracts dirt by the basket full.

I use Platen Restorer for the following:

- Platen Restoring
- Disk Drive belts
- VCR Drive Wheels and belts.

(Tom Linders is a long-time ATUNCO and SARAsaur. He repairs Apple ///s and edits the ATUNC Newsletter in his spare time...)

By John Lomartire

If you have an Apple /// and do not have Bob Consorti's Desktop Manager Utility, then you don't know what you are missing. Here is a software adjunct to make the Apple /// do all sorts of things quickly and easily. I can't begin to list all the large number of built-in modules supported by DTM, but among them are a calculator, appointment reminder, note retainer, macro capabilities, graphics manager, file management, communications software, and on and on. To top it all off, these added modules perform their task and then the user is returned to the software program he was using before calling up the DTM unit!!! If this world of capability is unknown to you, contact Bob Consorti immediately!

In its original and very worthwhile, conceptual form, DTM was designed to sit within a Profile (or other hard disk) as a complete set when any one of a multitude of modules would be available at the touch of a key or two. Used in this way, DTM makes an Apple /// the envy of many other computers, even some of the newer models.

As important and worthwhile as DTM is in its originally conceived utility, what I want to present is an exciting offshoot from the main concept.

It is possible to utilize only selected modules of the entire DTM series to prepare MINI-DTM's (as I choose to call these modified disks) to build your own "designer disks" for specific purposes. The advantage here, is that a particular software's capabilities are enhanced with characteristics that the original software lacked and these "designer disks" are portable from one computer system to another while a whole DTM system, although all encompassing, is resident only on the hard disk on which it is installed.

Adding individual modules can augment a particular piece of software, while groups of modules can, as a set, perform certain functions beyond the capabilities of the original software program.

For example, one of my pet peeves with VisiCalc or Advanced VisiCalc, is the lack of a cataloging function. The only way to find out what files are on a disk is to scroll through the list one file at a time. The MINI-DTM approach can be used to add cataloging and a few other functions to VisiCalc so that with a press of a function key, you enter the cataloging module, tell it which directory to list and you get the entire disk contents displayed. Press ESCAPE a couple of times and you are right back in VisiCalc!

Another example: I have a program that displays FOTO files. I also do a lot of charting with Apple /// Business Graphics. Unfortunately, screen files are saved by Business Graphics are not FOTO files, but Pascal data files, so my FOTOVIEW program will not display them. Business Graphics screen files need to have their File Type changed to FOTO (\$08), and it would be nice to be able to change the file name, since BG screen files have ".S.DATA" appended to the filename. This can be done by using one program to change the file type (block editor, or Communications Manager), and some other program to change file name, before FOTOVIEW to display the charts. Instead, the FOTOVIEW disk was modified as a MINI-DTM to incorporate a Change.File module, along with Copy.File, Rename.File, and Lock.Unlock modules.

It is all in one package now. The BG screen files can be modified quickly and easily without booting a

variety of other program and when the various changes have been made, you are back in the FOTOVIEW program ready to view the charts. (Note: although FOTOVIEW is written in Basic and therefore it could use this language to rename files and lock/unlock them, you would have to leave the FOTOVIEW program and go into the underlying BASIC language to do this. The advantage of the DTM approach is that the module functions can be carried out at any time and when done, you are returned to the part of the program that you were in when you first called up the module.

If this seems interesting, here are some tips on how to proceed to prepare a MINI-DTM.

This concept will work with any program that allows booting of a program followed by running the program WITHOUT the need for a disk having to stay in Drive 1. The DTM program can be installed in any drive except Drive 1 and the disk with the DTM files must be in that same drive when the boot disk is activated and whenever DTM modules are called into service during the program activity. (If a program disk is required to remain in Drive 1, then the MINI-DTM approach can still be used, but it will require either a third drive, or a bit more handling of disks during the process when modules are being accessed.)

Try using the built in drive (D1) and add an external drive (for example, D2).

Use the procedure described in the DTM manual to install DTM, but install it on a disk in drive 2, not the Profile.

With System Utilities, delete the unwanted modules from the MINI-DTM disk. This is an interesting finding. DTM requires that 220 blocks be available during installation. **BUT IT DOES NOT REQUIRE THAT THIS NUMBER OF BLOCKS BE MAINTAINED ONCE INSTALLATION IS COMPLETE.** Therefore, the large DTM files originally installed can be pared down to exclude all unneeded modules.

Also, you can copy in a new module to replace an existing unneeded one. For example, OPTIONE can be copied to OPTIONA and will replace it. The advantage here is that the modules in OPTIONS A through E can be accessed immediately with a simple keystroke.

Depending on what and how many modules have been selected during the booting operation, the process may stop and display an error message. This may occur more than once during the boot. Press RETURN at these stoppages until the final program is fully loaded. Everything should work OK after this.

I have barely scratched the surface of possibilities in this presentation. With a DTM library that has modules for checking machine status, going to and from a clipboard, printer controls, font controls, all sorts of file handling, accessing of the monitor, and many, many more, the potential for tailoring disks for specific purposes is endless. I encourage you all to give it a try. The final outcome is elating.

By John Lomartire

Sometimes one gets so used to doing things a certain way that he fails to see that other options exist. In this regard, it may come as a surprise for some that a spreadsheet program can be quite a respectable word processor. Admittedly, it does not have the power of conventional dedicated software for word processing, but it can do quite a bit.

For example, it sometimes happens that while working on a spreadsheet, I find a need for dashing off a note, or even a short letter. No need to leave the spreadsheet. Just move to a clear column, set the width to 78, and type away. The prompt line at the top of the screen serves as a convenient guide to regulate the line input length. Centering a line is a snap. Corrections are very easy. If you need to edit a line after it has been entered, you can either use the editing mode and the prompt line to make any necessary correction, or if it is more complicated, move one column over, set its width to 78, make the corrections on any number of lines, then copy these corrected lines back to the original paragraph. Whole paragraphs can be rewritten in this fashion. It gives you a chance to retain the original text while making the revision.

Even underlining can be done. Pick out the words to be underlined. Count how many letters and spaces to underline (or you could even skip the spaces) and then count how many spaces there are before the start of the underline. Now move one column over, enter the number of spaces to reach the start of an underline, enter the underline. Now, when you get to the print-out, print up to and including the line containing the text to be underlined. When this has been printed, move back one line on the printed page, move the cursor to the next column containing the underline and print only this line. Then continue printing as usual, starting (of course) after the underlined text.

Finally, and aside from word processing, one annoyance when using VisiCalc (and I suppose many other spreadsheets) is that the conditional @IF statement will not accept text as one of the alternates. It only takes value entries. To get around this, I use a wide column (e.g. 40 char) in one section of the spreadsheet. The top left hand corner is often a good place. Then next to the cell to be monitored I enter a prompt such as SEE NOTE: and in the cell next to this is the conditional statement where the number that it generates refers back to the note area.

It would look something like this:

In the upper left hand corner put:

NOTES

1. This value is too large, check cell B17
2. Any value less than 50 requires multiplication by 1.5

At another part of the spreadsheet, say G10, there might be a number that needs to be monitored. So at H10 you could enter SEE NOTE: and at I10 would be the formula:

`(@IF(G10>1000,1,@IF(G!)<50,2,0)`

resulting in a reference to NOTE 1 if G10 is greater than 1000 and a reference to NOTE 2 if G10 is less than 50. Otherwise, the entry is 0.

(In case you were wondering, this article was originally generated on a spreadsheet.)

By Paul Campbell

The three most important things in real estate: location, location, location.

The three most important things in computing: results, results, results.

Remember those times when I was a mainframe operator and turning in reports created on my Apple ///? Those reports would get looked over, then a few days later I would come across someone's attempt at duplicating one of those Apple /// reports. I say "attempt" because they could never get it as good as our SARA. Sometimes another person would manually type out a word processing document just to get it looking similar, other times they would just shake their heads and say "we can't do that".

Well, history repeats itself again and my new position is no exception. The group that I work with has always kept equipment devices records on an IBM computer. When my first database report using Three Easy Pieces was turned in to the Client Support department, they loved it! They even jokingly claimed that one of the people in the department wanted to marry the report. Other co-workers had similar comments about the reports being very good and "Just what we need".

What was the outcome of all this? In the middle of a corporation that has hundreds of IBM and clone computers, dozens of terminals, countless computer hardware devices, and multiple mainframe systems, now resides a newly installed Apple ///.

MS-DOS: The nightmare continues... Long live Apple /// SOS!

*By Paul Campbell
WAP /// SIG*

In my spare time (ha ha...) I've been studying operating systems and the information only confirms my opinion that MS-DOS still has a way to go before it reaches the level of SOS. IBMs operate in two modes: the "Real" mode in which programs can grab any chunk of memory, even if it's not supposed to and even if something else is using it; and the "Protected" mode where sections of the memory are roped-off and protected.

8088 and 8086 processors are incapable of operating in the protected mode. An IBM cannot access DOS in the protected mode and must downshift to real mode, and almost all of the time they must come out of the protected mode to access any disk drives.

On the other hand, Apple /// SOS has memory management built in. If an application tries to grab a pre-determined chunk of memory, SOS will allow it if nothing else has it occupied. If the space is not available then SOS protects it and won't let something else grab it. Business Graphics tries to load within a certain range and Write On (which is not worth running anyway) tries to get to the exact same place every time.

Aside from these two programs, everything else runs normally. /// Easy Pieces loads fine if it's the only thing around, but also runs fine under Selector, The Desktop Manager, huge driver files, a ram disk, TWO ram disks, or any combination. SOS will load it where ever it will fit.

As a side note, I'm amused at the interest in Virtual Memory. This is a great feature, but certainly not a breakthrough discovery. Try this test; load everything you can find until your SARA runs completely out of memory. Now open a fill-sized application from the Desktop Manager like Communications Manager or Graphics Manager. How in the world can you load another application with no memory? You got it... some of the RAM gets swapped to disk until the Desktop Manager is done, then swapped back. Not exactly the same behavior as Virtual Memory, but the same in concept and operation.

DRIVERS

One of the beautiful features of SOS is that you can build a driver file from scratch for a completely different system, then just send your client a disk! I have a subdirectory for device drivers on my 70meg Sider which allows me to scan the driver files and click on the driver I want.

If we were to get really fancy, we could get a hardware description over the phone, build the SOS.DRIVER file, then end the file to the other system over the phone lines. This is just one of the things that shows how MS-DOS is playing catch-up. For those who don't believe this, take two disks with forty or fifty files, give one to an Apple /// user and the other to an MS-DOS user, then tell them to copy HALF of the files to another disk. The MS-DOS user may not even attempt a task that is a such a breeze for SARA.

SOS IMPROVEMENTS

One thing that I think would be a real breakthrough and would make SARA a real serious machine in terms of software, would be a Pascal to Assembly Language Translator. What in the world would this accomplish? 1) Global speed increases. 2) Elimination of the 6 volume Pascal limit. 3) Elimination of the 16 Megabyte volume size, allowing hard drive partitions of 32 Megabytes. 4) Reduction or elimination "Lost" device drivers.

This application must be able to take a Pascal source code (or object code for real head-turning) and compile an Assembly Language object code. I'm not a programmer (hardware is more my specialty), but when I get ahold of some resources, this will be near the top of my list.

By Paul Campbell

A newly acquired addition for my SARA is StemWriter (On Three), a word processing program from Australia. If you want features, StemWriter has them!

For starters:

- Footnotes.
- 1,2,3, or 4 column text formatting.
- Automatic hyphenation.
- Multiple document loading.
- Viewing 2 different documents on the same screen.
- Background printing while you work on something else.
- Background printing of files on disk.
- Onscreen Bold, underline, sub/superscript.
- Password protection/file encryption.
- Automatic backup.
- Mind boggling printer options.

The list goes on and on. StemWriter is not as easy to use as Three Easy Pieces because of the many options. The main advantages of StemWriter are the multiple column formats and background printing. The disadvantages are; no exit to Selector and the formatted files are not compatible with Three Easy Pieces or AppleWriter.

The text pickup that is a part of The Desktop Manager will not work with StemWriter because the screen you use is really a graphics screen. My main problem is that I cannot get the StemWriter utilities to run (it was supplied on a non-original disk - the company no longer exists). I have the UTILS.CODE file, but according to Selector it is not an interpreter or a codefile. If anyone has some clues, let me know... until I can run these utilities to configure my printer, StemWriter is useless.

(/// SIG NOTE: You can find more information about STEMWRITER in the February and August, 1988 issues of On Three magazine.)

*By Paul Campbell
WAP /// SIG*

For those of us ///rs (and][rs) who would like to use the StyleWriter because of its size, price, or quietness, the answer is... YES! There is a different version of the StyleWriter available that has a standard Centronics parallel port and is available with Epson emulation. It works with /// Easy Pieces, AppleWriter, Word Juggler, etc.

This printer is identical to the original StyleWriter in regards to its print engine and mechanicals; the ink cartridge and head assembly can even be swapped between the two printers.

The printer is called the Cannon BubbleJet. What am I talking about? I did some hunting and discovered that the guts of the Apple StyleWriter are manufactured by Cannon, the StyleWriter even says "Made in Japan" on the bottom! Cannon uses the same parts in their BubbleJet which gives it identical quality and performance.

A new Apple /// user bought a StyleWriter and left it for me to play with. I got the driver set up along with the hardware cables, but that was the end of it. I spent hours on the phone trying to get additional info from Apple to no avail. The most concise answer I got from Apple Computer was a definite "I don't know". So much for support...

I would have continued my pursuit but was disappointed by the StyleWriter's lack of both speed and quality, the clincher was finding a normal printer in a similar price range that was the same except for the shell and paper feeder.

It seems strange that even Macintosh graphics need to be converted to a special format for the StyleWriter.

Well, enough about the StyleWronger. For anyone else looking for an ink jet type printer might be interested in the Hewlett-Packard DeskJet 500. It is much faster than the StyleWriter/BubbleJet and has larger capacity paper trays, the price is about \$365.

For TRUE laser quality, take a look at the Okidata LED PagePrinter (about \$575) or the Epson Acton Laser II (\$699). The Okidata (4 pages per minute) is slower than the Epson (6 pages per minute), but the print quality is real close. In favor of the Epson is a new technology that allows the electrostatic image to be transferred to the paper without the paper touching the print drum. This increases drum life to 30,000 pages.

If anyone is looking for a used printer for their SARA and comes across an IBM QuietWriter2, GRAB IT (if the price is reasonable). Although the QuietWriter uses a lot of ribbon, the quality is head and shoulders above any ink jet I've seen. The Apple /// graphics are outstanding on this printer. I will have samples at the next Apple /// SIG meeting. If you ARE going to be doing graphics on this printer, test it first. Some of them will not line feed accurately in the graphics mode which leaves hair-lines between each pass of the print head.

*By Joe Dumbrowolski
Apple Users Group International*

One of the most important things any /// user MUST understand is how to use the System Configuration Program (SCP) and how to build a SOS.DRIVER. Only you, the user, know your system's configuration and how you'll be using any piece of software. With this in mind we bring our first tutorial, one on SOS.DRIVER building.

RAM, ROM AND TURNING IT ON

Between the time you turn your computer on and the time you can do something useful with it, an awful lot of things happen - if everything goes well. This discussion concerns what should, or might, go on.

The first thing that occurs when you /// is turned on is that it checks its own hardware based on the instructions (diagnostics) in the Read Only Memory (ROM) chip. If any of these tests are failed you get one of the error messages listed on pg. 113-114 of the Owner's Guide. These indicate very serious problems which will require servicing, so quit right there if you get one.

If these tests are passed, the machine then looks to the internal disk drive to read its next set of orders (booting). More exactly, it is looking for a file called SOS.KERNEL which contains the fundamental instructions the machine needs to function internally. Without these instructions the /// can do nothing.

Next the /// looks to the same drive for the file SOS.INTERP (interpreter) which contains the instructions for interfacing with you, the user.

Finally, it looks on the disk in the internal drive for a file called SOS.DRIVER which contains the instructions for the machine to manage all its other devices such as the console, printers, hard disk, etc. This file also must be present and must contain at least the CONSOLE.DRIVER. As you may have learned, the absence of any of these files prevents a successful boot and you get an error message telling you what is missing. These errors (pg. 114-115), for the most part, do not indicate anything wrong with your machine, rather there is usually something wrong with the disk you attempted to boot and the message tries to tell you what.

While the 3 SOS files must be there, you do have some control over 2 of them. (The one you can't control is SOS.KERNEL which is the same for all boot disks.) You actually select the Interpreter. This file is an assembly language program which contains the language you are using such as BASIC or PASCAL, or the actual application program you are using such as VisiCalc or AppleWriter.

Hence you choose the interpreter when you choose the program. This also means that SOS.INTERP files are not freely interchangeable. The BASIC interpreter will not allow you to run Pascal programs even

though it will boot. If you are building a boot disk, be sure you have copied the correct interpreter for what you want to do.

The one you can really manipulate to your advantage is SOS.DRIVER and you don't even need to know a thing about programming. SOS.DRIVER is actually the directory name of the file which contains the individual device drivers. A device driver is just an assembly language program that tells the computer how to read from and/or write to the device.

Using the System Configuration Program (which is only a fancy file mover) on the Utilities disk you can add device drivers to the file as you add devices. **YOU CAN ALSO DELETE THEM.** Why would you delete drivers? Usable memory. When you boot your system you read all the individual drivers in SOS.DRIVER into your RAM memory. (See table of contents for list of drivers and their approximate memory requirements in K bytes.)

Thus SOS.DRIVER with several drivers could take up 40K or more of memory. That is 40K you cannot use for a VisiCalc template or an AppleWriter document; and, particularly if you have only 128K, you will probably miss it sometimes. The secret is, **AFTER MAKING A COMPLETE BACKUP COPY OF THE COMPLETE SOS.DRIVER FILE**, to reconfigure the system using only those drivers you really need.

If you don't have a QUME or Silentype printer you don't need those drivers and pick up 8K. You only need .FMTDX on your Utilities disk (Backup ///, 3 EZ pieces etc.) since it is used only for formatting. Very few application packages use .GRAFIX, etc. A word of caution: Read the configuration section of an application program carefully to determine the drivers it requires. For example, VisiCalc Advanced Version requires an .AUDIO driver. Still, the SOS.DRIVER that comes with it needs about 20K. If you use it with just a second floppy drive and a printer, you can cut that to about 14K by deleting .PROFILE and .QUME.

If you are wondering about the disk drive driver, don't. It is included in SOS.KERNEL because every /// has a disk drive. Admittedly there are some more things that may automatically happen when you boot a disk. The interpreter program, e.g. VisiCalc, may automatically start. Also, your HELLO (BASIC) or System.Startup (Pascal) program may automatically start (Turnkey disk) or you may get a message telling you to insert another disk (Two-stage boot). However, these events are not necessary for an Apple /// to get started. The sequence of SOS.KERNEL, SOS.INTERP and SOS.DRIVER is necessary and under those exact file names on the same boot disk. This is true even if you are using a "Hard Disk Boot" such as Catalyst or if you have PMOved Pascal to Profile.

WHAT HAPPENS WHEN YOU BOOT YOUR DISK?

When you boot up an Apple /// application, the system loads the Operating System (SOS.KERNEL), the Interpreter or application (SOS.INTERP), and the Drivers (SOS.DRIVER) into memory from the top down. In downloading the drivers, the system **WILL NOT SPLIT A DRIVER BETWEEN TWO 32K REGIONS OF MEMORY.**

If the driver to be loaded will not fit in the current 32K bank the loader skips the rest of that bank and loads into the next lower 32K segment. That skipped space is NOT available to any application and is wasted.

With larger configurations of drivers like print spoolers, it becomes more likely that this will occur. With a 256K or 512K machine you can probably chalk this up to overhead, but on a 128K machine you will probably wish to optimize the situation if possible and experiment with progressively larger configurations.

The reason we can't give you firm guidelines on which drivers to load in what order is that it depends primarily upon factors which vary completely from system to system, user to user and application to application. See the next section for a review.

TECHNICAL NOTE ON SYSTEM LOADING OF DRIVERS

The three variable system parameters which affect the loading of any particular driver file, and thus the 32K (more or less depending on revision) boundary problem are:

- the size of the Interpreter loading below SOS but above the drivers.
- the number and size of each driver configured, and,
- the order of those drivers in your SOS.DRIVER file (which, by the way is the opposite of the order the SCP list them in).

One example should get you on the track.....

- start with a SOS.INTERP that is 25K long
- using SCP we build a SOS.DRIVER file by loading an existing file (normally a SOS.DRIVER, however in our case individual files) which contain .CONSOLE driver 5K and .PROFILE driver 3K.
- then add a 24K print spooler, let's use D A Datasystems .PPRINT (a parallel printer spooler) in our example.
- Since .PPRINT is added last it will be loaded before .CONSOLE or.PROFILE.
- Now save the new SOS.DRIVER on our new boot disk and boot it...

Let's discuss what this means:

SOS.KERNEL loads from FFF to 8800 of the topmost reserved System Bank leaving 14K free in that bank (8000-B7FF). SOS.INTERP loads below that extending down 11K into the top User Bank, leaving 21K free in that bank. The 24K .PPRINT WILL NOT FIT in this remaining space so it is skipped and .PPRINT followed by the 8K of .CONSOLE and .PROFILE load into the next lower bank.

The net result is 21K wasted space.

HOW TO MAXIMIZE MEMORY

1. Select the minimum number of drivers necessary to utilize your application in the manner you wish.
2. Group drivers in near 32K groups (on a separate piece of paper).
3. Load driver, first the smallest group, then the larger (one nearest 32K - not over).
4. Edit driver if necessary.
5. Set the System Parameters.
6. Generate the new system.

*(Originally published in the **News and Views Newsletter**, November 1985)*

By Hal Edelstein

Because the Apple /// manual was not the world's best, several convenient techniques that are available in the System Utilities often go unnoticed. For example:

1. You don't ever have to type the names of existing files that are to be copied, deleted, etc. When the file name is requested in the lower left corner, type the drive (say .D2) or the volume name or the subdirectory pathname, then hit the up-arrow key. A slash and an equal sign (/=) now appear after your entry. A list of the files within that directory will appear in the upper right of the screen. Move the cursor to the desired file name, and hit the right-arrow key. More than one file may be selected at a time in this way. If you change your mind, hit the left-arrow key while the cursor is on the file name. A right-arrow alongside the file name indicates those that have been selected; the arrow disappears when you un-select it.

When you're through selecting, press Return. If you're copying files, a default name will appear on the next prompt line. If that has to be changed, make sure that '/=' follows the pathname.

If '.D2/=' appears as a default on the first line, DON'T press Return unless you want ALL the files in the directory. Instead type in or edit the directory name without the equal character (the end slash is optional), and hit the up-arrow as described above.

2. There is a character insert/delete feature built in to the System Utilities and available for most (but not all) activities in the Utilities! This is handy for editing long pathnames, etc.

To enter the Insert mode, press Open-Apple-I. A vertical bar will appear between two characters. This bar is a cursor indicating where characters will be inserted if typed. It can be moved with the left- and right-arrows, and toggled off and on with the same Open-Apple-I command.

To delete a character under the cursor, press Open-Apple-Right-Arrow. To delete one to the left of the cursor, press Open-Apple-Left-Arrow.

(Originally published in the ATUNC Newsletter, April 1985 (Vol. 2, No. 4).)

Apple // Family Used Computer Guide (May 1992)

By David Ottalini
Apple /// SIG Co-Chairman

Prices for used Apple // family computers continue to go down. In just the last two months, Sun Remarketing has reduced their prices from \$20 to \$50 dollars on their //e's and GS's. Biggest drop was the bare-boned GS - \$50.00 cheaper and now the SAME PRICE as an Apple /// (whose price did NOT drop). Micro Systems has also dropped their price for a used 1.25 MB GS by \$50.00. Note that prices can change quickly and this column is written far in advance, so be sure to call and double check prices.

CPU = Computer ; MON = Monitor; DD = Disk Drive; PR = Printer; SW = Software
 RGB = RGB Color Monitor; U=Unenhanced; PCT = PC Transporter; HD = Hard Disk;
 MM = Much More; +/- = Increase/decrease from last month's listing. "Micro Systems" advertises in A+ and Computer Shopper's Classifieds section. Wood Chuck Industries was also in Computer Shopper.

<u>COMPUTER/PRINTER</u>	<u>SOURCE</u>	<u>SYSTEM?</u>	<u>PRICE</u>	<u>+/-</u>
Apple /// 128K	Sun	CPU	\$295.00	
Apple /// 256K	Sun	CPU	\$345.00	
Apple /// 256K	Sun	CPU/MON	\$349.00	
Apple][+	Sun	CPU	\$49.00	
Apple //e 128K	Sun	CPU	\$195.00	- \$34.00
Apple //e 1MB	WAP Journal	CPU/MON/DD/PR/RGB/SW/MM	\$900.00	
Apple //e	Micro Systems	"System From"	\$274.99	- \$74.01
Apple //e 128K	Post	CPU/RGB/MOD/PR/SW/MM	\$950.00	
Apple //e 256K	TCS	CPU/2DD/MON/PR/MM	\$375.00	
Apple //e 192K (U)	TCS	CPU/2DD/MON/MM	\$325.00	
Laser 128	Computer Works	CPU	\$199.00	
Apple //e 128K	Sun	CPU (5.25 DD)	\$229.00	- \$20.00
Apple //c	Post	CPU/DD/MON/PR/MM	\$425.00	
Apple //c Plus	Sun	CPU (3.5 DD)	\$295.00	
Apple IIgs	Sun	CPU	\$349.00	- \$50.00
Apple IIgs 1.25 MB	Micro Systems	CPU	\$549.00	- \$49.01
Apple IIgs 1.0 MB	Wood Chuck	CPU	\$450.00	

Apple IIgs	Post	CPU/RGB/2DD/MOD/MM	\$950.00	
Apple IIgs	TCS	CPU/2DD/MOD/MM	\$550.00	
ImageWriter II	Sun	Printer	\$249.00	- \$80.00
ImageWriter I & II	Micro Systems	"Printers From"	\$225.00	
ImageWriter II	Post	Printer	\$225.00	
ImageWriter II	Post	Printer	\$340.00	
ImageWriter II	Post	Printer	\$275.00/obo	
ImageWriter II	Post	Printer	\$125.00	
ImageWriter II	TCS	Printer	\$325.00	

Apple // Family Used Computer Guide (July 1992)

By David Ottalini
Apple /// SIG Co-Chairman

After a one month delay, I'm happy to bring you another edition of my used computer price guide for the Apple // family. Please note there was a typo in the price for a used //e from Sun, which has been corrected this time around. We continue to find there are plenty of bargains out there. You may have found your own at the WAP Garage Sale last month. If you missed it, take a look at what's being offered and use these prices to help guide you in your negotiations. If you buy or sell an Apple // family system, please let me know and I'll include it in my next listing.

CPU = Computer ; MON = Monitor; DD = Disk Drive; PR = Printer; WC = Wide Carriage; SW = Software RGB = RGB Color Monitor; U=Unenhanced; PCT = PC Transporter; HD = Hard Disk; MOD=Modem; MM = Much More; +/- = Increase/decrease from our last listing. //c comes with an internal 5.25 DD. //c+ with a 3.5 DD. Apple /// and ///+ with internal 5.25 DD.

COMPUTER/PRINTER	SOURCE	SYSTEM?	PRICE	+/-
Apple /// 128K	Sun	CPU	\$295.00	
Apple /// 256K	Sun	CPU/MON	\$345.00	
Apple /// 256K	Sun	CPU/MON	\$349.00	
Apple ///	Tom Linders	CPU	\$50.00	
Apple][Plus	Sun	CPU	\$49.00	
Apple][Plus	Post	CPU/DDs/RGB/MON/PR/MM	\$280.00	
Apple //e 128K	Sun	CPU	\$349.00	
Apple //e 128K	Sun	CPU/MON	\$429.00	
Apple //e	Post	CPU/MOD/MM	\$500.00	
Apple //e	Post	CPU/HD/DDs/MON/PR/MM	\$1,500.00	
			obo	
Apple //e	Post	CPU/PR/RGB/DDs/MM	\$900.00	obo
Apple //c 128K	Sun	CPU	\$195.00	- \$34.00
Apple //c 128K	Sun	CPU/DD/MON	\$249.00	
Apple //e New	Sun	CPU	\$245.00	
Apple //c	Post	CPU/RGB/MON/DD/PR/MM	\$700.00	
Apple //c	Post	CPU/DD/MON/PR	\$495.00	
Apple //c	Post	CPU/MON/MM	\$150.00	
Apple //c Plus	Sun	CPU	\$295.00	
Apple //c Plus	Sun	CPU	\$345.00	
New				

Apple IIgs 256K	Sun	CPU	\$349.00
Apple IIgs 2MB	Post	CPU/3 DDs/RGB/MM	\$1,050.00
			obo
Apple IIgs 2MB	Post	CPU/DDs/RGB/PR/MM	\$925.00
Apple IIgs New	Sun	CPU	\$399.00
Apple IIgs 1.2 MB	Post	CPU/DD/RGB/MM	\$800.00
Apple IIgs	Post	CPU/DDs/RGB/MM	\$1,100.00
ImageWriter I	Sun	Printer	\$279.00
ImageWriter II	Sun	Printer	\$279.00
ImageWriter II	Sun	Printer	\$329.00
New			
ImageWriter II	Post	Printer	\$250.00
ImageWriter II	Post	Printer/MOD	\$250.00
ImageWriter II	Post	Printer	\$300.00
WC			

By David Ottalini
WAP /// SIG

As promised, I am happy to announce this month that our /// SIG PD now includes ThreeWorks, my "unique bibliographies and data bases for the Apple /// Computer."

Developed over many years, I've spent literally hundreds of hours gathering a tremendous amount of information about the /// that you can use quickly and easily with Three EZ Pieces. In recognition of the time spent on this project, I must make this a shareware offering. If you purchase ThreeWorks, I am asking that you send me an additional \$5.00 (thank you!)

This is the 1.1.92 version of ThreeWorks and I've updated the bibliographies so that they include listings of all /// articles from 1991. I've updated wherever possible, including the vendor and repair lists. You will be able to get the most out of ThreeWorks with a 256k Apple /// or ///+ because some of the files are quite large. If you own a // and have any version of AppleWorks or AppleWorks GS, you can read these files without any problems.

/III.INF.31 (Side One)

BIB.80.84

An Apple /// Bibliography of articles beginning in 1980 and ending in 1984.

FATAL.CODES

A list of fatal codes which the user may, on occasion find on his/her screen due to a program failure. Original information from Sun Remarketing.

NON.FATAL.CODES

A list of non-fatal codes which the user may, on occasion find on his/her screen and what it means to the operation of the user's program. Original information from Sun Remarketing.

PEEK.POKE.CALL

For the Basic programmer. A list of all known Peek, Poke and Call statements with Apple /// Business Basic translations where possible. Useful for those trying to convert AppleSoft Basic

programs.

REPRINTS

How to order reprints of articles you find in ThreeWorks.

THREE.WORKS

The information file you are reading right now.

/III.INF.31 (Side Two)

APPLE3.BIB.85

An Apple /// Bibliography of articles for the year 1985.

A3.GLOSSARY

A glossary of terms related to the Apple /// and many of its programs.

/III.INF.32 (Side One)

APPLE3.BIB.86

An Apple /// Bibliography of articles for the year 1986.

A3.PGM.VERSIONS

A list of many Apple /// programs and their latest version numbers.

A3.REPAIRS

A list of companies or individuals who can repair your Apple /// or its peripheral hardware.

A3.CHIPS

A list of the chips found inside your Apple ///, their respective location and sources of replacements. Original information from Ed Gooding's ///s Company BBS.

/III.INF. 32 (Side Two)

APPLE3.BIB.87

An Apple /// bibliography of Apple /// articles for the year 1987.

APPLE3.BIB.88

An Apple /// bibliography of Apple /// articles for the year 1988.

FILE.TYPES

A list of the various file types used by the Apple // computer family.

MON.COMMANDS

A complete set of the Apple ///'s Monitor Commands.

/III.INF.33 (Side One)

APPLE3.BIB.89

An Apple /// Bibliography of articles for the year 1989.

APPLE3.BIB.90s

An Apple /// Bibliography of articles for the years 1990 & 91.

DL13.DB

A data base listing the offerings on the DL (Data Library) 13 section of MAUG (Micro-networked

Apple Users Group) on CompuServe.

/III.INF.33 (Side Two)

PD.3SOFTWARE

A list of all known public domain offerings by vendors and clubs around the nation. Includes prices and descriptions.

* Note WAP PD listing is located on a separate disk: 3INF.01 is ASCII version; 3INF.23 is 3EZP Version.

/III.INF.34 (Side One)

APPLE3.BBS

A listing of Apple /// Bulletin Boards around the nation.

APPLE3.BOOKS

A listing of books about the Apple /// or that mention the Apple /// either in a technical or historical setting.

APPLE3.CABLES

An Apple cable compatibility guide for the entire Apple family of computers, including stock and engineering numbers.

APPLE3.CLUBS

A current listing of organizations that are active Apple ///-specific clubs or have Apple /// SIGS (Special Interest Groups).

APPLE3.CLUBS2

This file contains a list of groups that were Apple /// specific or had /// SIGs (Special Interest Groups). In many cases, the contact person may still have a /// and can provide support if you live in the area.

APPLE3.CONSULTS

A list of individuals known as consultants for the Apple ///. The compiler makes no claims nor can he attest to the ability of those listed in this data base (except, that is, for himself!). Please call or write for information.

APPLE3.DRIVERS

A listing of all known Apple /// device drivers, their latest version numbers, size, etc.

APPLE3.PUBLCTNS

A list of publications dealing specifically with the Apple /// or that include articles about it.

APPLE3.TIPS

A listing of Apple /// tips from numerous newsletters and magazines. Brief description of tips provided in most cases.

/III.INF.34 (Side Two)

APPLE3.VENDORS

A list of all vendors who support or supported the Apple /// Computer. The compiler has checked some of them, but would recommend the user call or write a specific vendor in this list before sending any money for a particular product.

BASIC.KEYS

A complete listing of the Keywords used in Business Basic. Includes descriptions, examples and notes on each Keyword.

Also going into the PD this month is another program from Dr. Al Bloom. Disk 3UTL.36 is called

ASCIUPLD, or ASCII UPLOAD. As Dr. Bloom explains it:

ASCIUPLD converts an ASCII text file (one line per record field) from one data base or spread sheet program into the upload format needed by an importing program. A special command file tells ASCIUPLD (1) which fields of the input ASCII file to be imported and (2) in which order to import the fields. It even lets you create import fields that aren't in the original file.

Full documentation is included on side one, along with the source code. The actual program is on side two.

Finally, we're adding disk **3TEL.09**, the long-awaited Infonet BBS software. Thanks to Sun Remarking back in 1987, this BBS was placed into the PD. It's been extensively upgraded by Ed Gooding and Walt Pawley. Ed uses it on his ///s Company BBS. The User Manual and source code are located on side one while the actual program is on side two.

This disk is NOT self-booting and some modifications may be required to tailor Infonet to your needs. That means you should have some knowledge of Pascal if that's what you want to do. The documentation does explain how to do that, however.

Coming up in the future: More Best of ATUNC disks, more programs from Dr. Bloom and as always, a few surprises. Stay tuned!

*By David Ottalini
WAP /// SIG CO-Chairman*

We offer two disks this month, /// SIGers.

The first disk, **3UTL-35**, is called Font Generator /// and was originally published by Apollo Software. It's a great font editing program you should find fun and easy to use. It's written in Business Basic. Information files are on side one and the program itself is on side two (including some font files for you to try out.)

Font Generator /// not only allows you to design character fonts, but it also lets you check out how your fonts would look in different text modes (even with foreground and background colors.) As many as ten character fonts may be loaded at the same time, and the program can copy characters from one font into another one.

The second disk, **3INF-35** is another in our long-line of "Best of Ottalini" offerings. This time yours truly republishes all his articles from 1990 onto one disk.

They're fun. They're educational. And most importantly, they're cheap. So grab a copy while they're still available (John Ruffatto will be working overtime to keep up with the expected demand.) As an additional added incentive, I've thrown in some article from a few other authors as well.

By David Ottalini
WAP /// SIG C-Chairman

"If April Showers should come your way...they bring the /// PD disks that bloom in Bethesda"

- Long Lost Cousin of John Ruffatto

All right. So John Ruffatto doesn't have a long lost cousin. But he has been a busy Apple // librarian these past weeks copying all the Apple /// PD disks going into our library.

I actually took some time and counted the disks we now have in the PD and they're up to some 180+ (mostly) double-sided disks. 180 times 5 copies each is... well you get the idea of what it takes.

Of course that doesn't include all the fun times I have putting together these disks for you as well. I stopped counting the hours a long time ago!

This month, I'm happy to report that we're adding another Graphics disk to the pile already on your table. **3GRX.37** is our fifth disk full of GIF graphics (from the Apple // side of the family) that have been converted to the ///.

Thanks to our new 5.0 graphics version of Menu.Maker, you'll be able to both view and print any of these disks that you like. They are presented on screen in INVERSE mode due to the quirks of the invokable we're using to print the graphics. On paper, the graphics are in NORMAL mode.

The pics on this disk include EAGLE.1; WIZARDS; MANHATTAN SKYLN; BAT LOGO; X.24; ROGER.RABBIT; ROBOTARM; MICKY.MOUSE; COKE.CAN AND MT.HOOD.

We're also releasing **3INF.26**, The Best of ATUNC #4. As with the other disks in this series, this one contains a wealth of information on a wide range of subjects:

On Side One:

MISC.A3.INFO (Subdirectory)

WHAT.U.NEED: What the new Apple /// user really needs.

MONITORS (Subdirectory)

A3.TO.COLOR.TV: Connecting an Apple /// to a color TV set.

COLOR.MON.HKUP: Hooking a color monitor to the /// from composite port.

PANASONIC.MON: Hooking up a /// to a Panasonic Monitor.

SCREEN.BLANK: Inserting <CONTROL 5> (screen blanking) into a program.

SCREENBLNK.BEEP: In Visicalc (or other program) use <CONTROL 5> to blank screen
then <CONTROL G> and /// will beep when done calculating.

VIDEOTAPE.A3: Hooking up a /// to a VCR.

On Side Two:

ONTHREE.PRODUCT

DRAW.ON.3.CMDS: Draw On Three commands.

DRAW.ON.3.REVW: A review of Draw On Three.

GRAPHIC.MGR.DMO: Report on a demonstration of Graphics Manager.

GRAPHICS.MANAGR: Report on Graphics Manager and printer compatibility.

ON.THREE.DEMOS: Bob Consorti visits ATUNC for demos of his 512K bd, etc.

ON.THREE.PRODS: Reviews of Selector, Desktop Mgr and other products.

UPGRADE.512K: Upgrading your /// to 512K.

WHY.512K.UPGRAD: Why upgrade to 512K.

OTHER.COMPUTERS (Subdirectory)

A3.A2.MAC.ETC: An evaluation of the /// vs. other computers.

COMPUTER.TO.BUY: Discussion about what your second computer should be.

IBM.PS2.DISADVA: Excerpt of an article "PS/2 is for Suckers."

IBM.RUN.BY.A3: How to control an MS-DOS machine remotely with a ///.

LASER.128: Discussion about this Apple // compatible.

LASER.128.AS.A3: Apple /// w/Titan Cards vs. Laser 128.

PRINTERS (Subdirectory)

CABLE.SER.PNTR: How to build a serial printer cable for the ///.

DIABLO.PRT.PROB: Diablo 630 printer problem/solution (in software).

GEMINI.PRINTR: Getting a Gemini printer to print enhanced/bold print.

HP.LASERJET: How to hook up a Laserjet Plus to a ///.

HP.LASERJET.PNT: Configuring the Laserjet to work on a ///.

IMAGEWRITER.PTR: How to hook up an Imagewriter to a ///.

IMAGEWRITR.PROB: Correcting a print problem with the null modem cable.

PLOTTING: Plotting and graphing with the ///.

POWERPRT.SIZE: Discussion of the Powerprint drivers (D A Datasystems).

PRINT.SPOOLER: Discussion of the Quark Discourse print Spooler.

SERIAL.DRIVER: How to know you have the right serial driver.

SOFTWARE.MISC (Subdirectory)

AMORTIZATION.PG: Discussion of a PD real estate amortization program.

BPI.GEN.LEDGER: Demo and discussion of BPI General Ledger.

BUS.GRAF.FAIL: Fixing Stack Overflow problems with Business Graphics.

BUS.GRAF.PRINT: Business Graphics drives a /// user to the Mac.

DATABAS.CONVERT: Converting Apple /// files to another computer.

DISCOURSE.PROB: Installing Catalyst Discourse on a Profile.

DISK.LABELS: Sources for PD software-including a disk label utility

FINANCE.DISKS: Announcement about new Advanced Visicalc financial modeling disks
for the ATUNC PD.

Again remember that these disks and our entire PD library will be available during our meeting on Saturday, April 18 at the WAP Office.

By David Ottalini
WAP /// SIG CO-Chairman

Our PD Library continues to grow for your enjoyment and the (slight) enrichment of the WAP coffers. This month we offer SIX disks. As usual, there's a little something for everyone:

3ZAW06 is our first "Contributions Disk" for Appleworks/3EZ Pieces templates. You'll find templates that will help you keep track of your portfolio, music and video libraries and your ancestors. There are templates to print your address on an envelope, do labels, keep an address book, keep track of your credit cards and much, much more. There is some documentation, but most of the templates are easy to follow and use.

3INF-27 and **28** are the final two installments in our "Best of ATUNC" series. **3INF-27** has some interesting articles geared towards anyone interested in programming, tons of information about all kinds of software, telecommunications, testing and repairs, and the Titan cards. **3INF-28** completes things with information related to ATUNC itself, various utilities, Visicorp products (like Visicalc) and Word Juggler.

The disks in this series are a tremendous resource for all Apple ///ers and a must for any Apple /// library. Again our thanks to Robert Howe for compiling all this in a form that makes it really easy to find information about our beloved SARA.

You know I couldn't forget graphics, so we offer 3GRX-38 and 39 this time around. 3GRX-38 is only for those of you who like dinosaurs. Lots of them. Print them big or small, they'll entertain little ones with crayons or scissors for a long time.

3GRX-39 is a disk full of bookplates. Print them out and use in your own books or let your kids have some fun as well. Something for everyone as we always like to say.

All these graphics were scanned using the Quickie on a //gs and then transferred to the /// using Graphics Manager. Menu.Maker will print them one at a time or use the Print.Show program to print them all large or small.

Upcoming: more disks for you!

Appleworks/3EZP fans, another How Do I disk, a third Best of ///s Company and more graphics disks. Stay Tuned!

By David Ottalini
WAP /// SIG Co-Chairman

I'm happy to announce a number of new PD disks this month, /// \SIGers. They run the whole range of our PD library so you have no excuses by not coming over to the WAP office right away and purchasing the whole slew of them!

3UTL.38 is called A3.Reboot and is offered with permission of On Three. It's a special version of Selector ///, a demonstration of sorts that will give you a taste for what the program can do while itself being something you'll find useful IF you don't own a hard disk yet.

Basically, A3.Reboot is a "preboot." You boot it first and it gives you the Selector menu. It will allow you to do one thing: Boot any disk from .D1. When you're finished with that program, quit and you'll be back to your Selector menu. No more rebooting. Just put in your new disk, press RETURN and your new program will run automatically.

You'll want to configure the SOS.DRIVER file on A3.Reboot to your system. That's because it loads all three SOS files on bootup. After that, it will only run the SOS.INTERP file of your program disk.

To make this disk more useful, we've included information about Selector, a comparison of Selector and Catalyst and more on side one. A3.Reboot is on side 2.

Thanks to Dr. Al Bloom, we've been able to update disk **3MSC-19** with the latest version of the Mail List Manager hard disk version.

3PCL-17 is C.M. Davidson's modified version of Tom Bartkiewicz's Pascal Menu.Maker program. This version will allow the importation of Apple /// FOTOfiles. Knowledge of Pascal required to use this disk. Side One is bootable from Menu.Maker and contains information about the program. The modified Pascal MM is on side two. We've also thrown in a couple of FOTOfiles and two of Mr. Davidson's Madelbrot programs (in Pascal).

3UTL-39 is "Conpath by Bloom." As our own Dr. Al Bloom describes it:

CONPATH tells about ways to convert one file format to another. The premise is simple. There are a bunch of "bridges" and a bunch of file formats that can be "stepping stones" between the file format you have and the one you want. Most commercial data base and spread sheet programs can "import" or "export" data in one or more formats. Those files are stepping stones between systems.

There are plenty of "bridges" around. Getting from one file format to another can be easy. However, the path from "here" to "there" may not be obvious, it may be downright devious, and there may be several paths. CONPATH is designed to help you find those pathways.

As with most of our disks, the user manual for Conpath is on side one and can be read with

Menu.Maker. Side two contains the program and is also self-booting. The source code is included.

Dr. Bloom has contributed a number of other programs to our PD, and they'll be making their way into the library over the next few months, including two DIF format related disks.

Moving into my favorite area (notice I saved it for last...) we add two graphics disks this month. **3GRX-40** is "Animals #1" and includes a number of scanned graphics of... you guessed it... animals. These FOTOfiles include fish, cats and bunnies this time around. As I have time, we'll scan and produce more disks. Suggestions welcome. This disk is self-booting and you can look at or print any of these from Menu.Maker.

3GRX-41 is called Artworks. It's a Pascal-based graphics program you might enjoy. It is primarily keyboard-based, by the way. Side one includes information about the program and how to use it. There's a text file version of the Artworks source code and a demo FOTOfile created with Artworks. Side Two includes the self-booting program and Pascal text file version of the source code.

*By David Ottalini
WAP /// SIG Co-Chairman*

Disks, Disks and more Disks for you this month, my faithful SARAsaurs. A lot of disks that I've been working on for quite a while finally got off the ol' disk drive for you.

To start off with, we've updated the New Member Disk, **3INF-02** so that it reflects new addresses, new phone numbers and new realities. **3UTL-29**, disk 3 of Dr. Al Bloom's Mail List Manager Utilites has been updated. And finally, I'm happy to report that **3BSB-09**, our Menu Maker disk has been updated to version 6.1 thanks to some suggestions by our Canadian Friend C.M. Davidson. The primary new feature lets you back out of a subdirectory to the next level (rather than having to return to the root and working your way down again.)

As for new disks, here's the skinny this month.

PDS NAME: BBQ Recipies and Much More!

DISK ID#: 3MSC-20

SUBJECT: Miscellaneous

BOOTABLE?: Bootable

DISK SIZE: DS 5.25

OS: SOS

DESCRIPTION: Due to popular demand of the /// SIG Co-Chairman, this PD disk includes a treasure-trove of recipes sure to entice any and all BBQ lovers. As an added bonus, we're also including the latest (1992) information about the best restaurants in the Washington DC area - courtesy of the Cooks board on the WAP TCS. The BBQ recipes all come from the Cooks Forum on CompuServe.

On Side One:

BBQ.Fish (Subdirectory)

Calamari; Squid.Grill; BBQ.Singapore; Mahi.Mahi\Grill; Frog.Legs.Grill;

Shrimp.Napalm; Albacore.Grill; Avodaco.Tuna; Basic.Brine.

Hundred.Best.DC: 3EZ Pieces/Appleworks DB of 100 Best DC Restaurants (ASCII version on Side Two)

On Side Two:

BBQ.Meats (Subdirectory)

Lamb.Garlic; Ribs.French; Moroccan.Lamb; Pork.Tender; Beef.Broch(ette);

Venison.Roast; Smoked.Beef; Hams; Smoked.Beef.II; Burned.Grill; Lamb; Sesame.Beef; Sesame.Ribs; Flank.Pinwheel; Beef.Kabobs; Sear.Serloin.

BBQ.Sauces (Subdirectory)

Dry.Rub; RNY.Day.Sauce; BBQ.Sauces;BBQ.Sauces.II; Red.Pep.Paste; Jalapeno.Pste;

Spice.Mix; Florida.Sauce; Bushwhacker.Sce.

BBQ.VEGS (Subdirectory)

Potato.Barby; Tomato.Sp.Grill

Richmans.Fav: Phyllis C. Richman's favorite DC Restaurants

Best.Bargains: The Washingtonian's 100 Best Restaurant Bargains (1992)

Hundred.Best: The Washingtonian's 100 Best DC area Restaurants (1992)

Cooks.Online: The cooks on CompuServe talk BBQ.

PDS NAME: **Bankcalc #1**

DISK ID#: **3ZAW-07**

SUBJECT: **3EZP/AW Templates**

BOOTABLE?: **Non-Bootable**

DISK SIZE: **DS 5.25**

OS: **SOS**

DESCRIPTION: Bankcalc 1, 2 and 3 are collections of Visicalc (DIF) models for the banking industry. These models were originally written and released in 1983 by community bankers or Banking Industry Software Houses and contributed to Bancalc. The original purpose was to provide Bank personnel with VC models which could be used immediately. As a home user, you may find some of these models useful as well. As DIF files, you can pull these models into Visicalc, Advanced Visicalc as well as 3EZ Pieces. You should also be able to use them with AppleWorks or Appleworks GS.

On Side One:

Model 1: Variance Analysis of Yields

Model 2: Yield Summary

Model 3: Rate Sensitive Statement of Condition

Model 4: Simple Loan Amortization Schedule

Model 5: Rate of 78 Amortization Model

Model 6: Effective Rate Model

Model 7: Compound Yield

Model 8: Fixed Assets Model

Model 9: Accrue Expenses

On Side Two:

Model 10: Charitable Contributions

Model 11: IRA Calculations

Model 12: IRA Vs. CD

Model 13: All Saver CD Yields

Model 14: Credit Analysis

PDS NAME: **Bankcalc #2**

DISK ID#: **3ZAW-07**

SUBJECT: **3EZP/AW Templates**

BOOTABLE?: **Non-Bootable**

DISK SIZE: **DS 5.25**

OS: **SOS**

On Side One:

Model 1: Commercial Loan Yields

Model 2: Installment Loan Yields

Model 3: Real Estate Loan Yields

Model 4: Money Market Yields

Model 5: Investment Yields

Model 6: Deposit Costs

Model 7: Asset Yields

Model 8: Liability Cost

Model 9: Interest Margin

Model 10: Earnings Projections

On Side Two:

Model 11: Accelerated Cost Recovery

Model 12: Budget Income Expense

Model 13: Budget Sheet Budget to Actual

Model 14: Monthly Rates

Model 15: Income Tax

Model 16: Mortgage Amortization

PDS NAME: **Bankcalc #3**

DISK ID#: **3ZAW-09**

SUBJECT: **3EZP/AW Templates**

BOOTABLE?: **Non-Bootable**

DISK SIZE: **DS 5.25**

OS: **SOS**

On Side One:

Model 1: Balloon Payment Loan Amortization

Model 2: Interest Only Loan Amortization

Model 3: Daily Investment Rates

Model 4: Daily Teller Transaction Count by Hour

Model 5: Account Analysis

On Side Two:

Model 6: Director's Report

Model 7: Banker's Acceptance

Model 8: TEFRA calculation

PDS NAME: **Fortran Manual: Disk One**

DISK ID#: **3PCL-18**

SUBJECT: **Pascal**

BOOTABLE?: **Non-Bootable**

DISK SIZE: **DS 5.25**

OS: **SOS**

DESCRIPTION: Disk One of Two disks containing the manual for the Fortran Language. Pascal required to read/print these files (which are Pascal Text Files). Use Pascal Menu.Maker, disk 3UTL-31 to read or System Utilities to print (disk 3UTL-01 - use "Copy Files" and copy to .Printer).

PDS NAME: **Fortran Manual: Disk Two**

DISK ID#: **3PCL-19**

SUBJECT: **Pascal**

BOOTABLE?: **Non-Bootable**

DISK SIZE: **DS 5.25**

OS: **SOS**

DESCRIPTION: Disk Two of Two disks containing the manual for the Fortran Language. Pascal required to read/print these files (which are Pascal Text Files). Use Pascal Menu.Maker, disk 3UTL-31 to read or System Utilities to print (disk 3UTL-01 - use "Copy Files" and copy to .Printer).

PDS NAME: **Fortran Language**

DISK ID#: **3PCL-20**

SUBJECT: **Pascal**

BOOTABLE?: **Bootable**

DISK SIZE: **DS 5.25**

OS: **SOS**

DESCRIPTION: Fortran Language. Requires Pascal. See Manuals on Disks **3PCL-18** and **19**.

PDS NAME: **Conpath by Bloom**

DISK ID#: **3UTL-39**

SUBJECT: **Utilities**

BOOTABLE?: **Bootable**

DISK SIZE: **DS 5.25**

OS: **SOS**

DESCRIPTION: Conpath by Bloom is another in a series of great programs by Dr. Al Bloom. This disk describes bridges - ways to get from one file format to another. User manual is on side one. Program is on side two.

PDS NAME: **BLOCREAD By Bloom**

DISK ID#: **3UTL-40**

SUBJECT: **Utilities**

BOOTABLE?: **Bootable**

DISK SIZE: **DS 5.25**

OS: **SOS**

DESCRIPTION: BLOCREAD is a utility program that lets you look into the guts of a disk file. It shows you a file's 512-byte data blocks, in both character and hexadecimal format. You may look at any block in a file and choose to either print it or copy it to another disk file. BLOCREAD is invaluable for seeing what a file really looks like. Another great program from Dr. Al Bloom.

Side One includes BLOCREAD documentation and Pascal Source Code.

Side Two is the BLOCREAD program.

Both sides are self-booting.

PDS NAME: **NVASCI By Bloom**

DISK ID#: **3UTL-41**

SUBJECT: **Utilities**

BOOTABLE?: **Bootable**

DISK SIZE: **DS 5.25**

OS: **SOS**

DESCRIPTION: NVASCI is another great program from our good friend, Dr. Al Bloom. NVASCI will convert a NAME:VALUE format text file into an ASCII text file with one line per record field - required for importing into many Data Base programs.

Side One includes a complete tutorial and the source code to NVASCI.

Side Two contains the Pascal-based program.

*By Dave Ottalini
WAP /// SIG*

We're back on track with a great Holiday lineup of PD disks for you to consider! How about a 3EZ Pieces/Appleworks Video Catalog Data Base - it's already got a lot of movies listed. Add your favorites as you like.

Omnis 3 was one of the best Apple /// Data Bases ever produced. We don't have the program itself to offer (yet!) but we do have a disk full of helpful tips, reviews and articles you'll find very useful.

Dr. Al Bloom adds his LISTIT program - which provides formatted listings of Pascal programs. And then there are two new telecommunications programs - hpTerm and Access 3270 (by request of Paul Campbell). As always, I've enhanced these disks to make them as useful as possible - despite the lack of manuals.

Epson Screen Printer gives Epson and compatible printer users a great deal of flexibility while printing graphics or text. You can, for example, print text sideways if you like. Instructions are provided on disk.

Finally, DIRECTRANS by DROB protects the root directory of your hard disk while Disk Speed will check the speed of your internal disk drive. Information for both programs also provided on disk.

In light of our Macintosh friends' great offer, I too have chosen 25 of our best Apple /// disks that you can have through the month of December for just \$25.00. Happy Holidays - and Happy New Year!!

PDS NAME: Video Catalog: Disk 1

DISK ID#: 3AWZ-10

SUBJECT: 3EZP/AW Templates

BOOTABLE?: Nonbootable

DESCRIPTION: Video catalog of movies that you may find useful. Each record includes categories for Title, Star, Director, Category, Year, Store, Stars and Code.

Side One: A-G; H-K

Side Two: L-P

PDS NAME: **Video Catalog:Disk 2**

DISK ID#: **3AWZ-11**

SUBJECT: **3EZP/AW Templates**

BOOTABLE?: **Non-bootable**

DESCRIPTION: Video catalog of movies that you may find useful. Each record includes categories for Title, Star, Director, Category, Year, Store, Stars and Code.

Side One: Q-T

Side Two: U-Z

PDS NAME: **Omnis 3 Technotes Plus!**

DISK ID#: **3INF-36**

SUBJECT: **Information**

BOOTABLE?: **Bootable**

DESCRIPTION: Omnis 3 is a data base program originally offered by Blyth Systems. It was later sold by D A Datasystems. This disk does NOT contain the program, but rather a set of tech notes released by D A Datasystems for Omnis. We've added a bibliography of all known articles about Omnis for the /// as well as some reviews and informational articles.

On Side One:

Omnis.Review1: Review from TauTales

Omnis.Review2: The Omnis Man, Michael Reilly writes about Omnis 3.

Omnis.Man: From The Three Magazine, a short article about Reilly.

Omnis.BIB: 3EZ Pieces/AppleWorks Data Base of Omnis Articles.

Omnis.DB.Ascii: Text version of Omnis.DB.

Omnis.Review3: On Three article on Omnis 3 by Earl Brelje.

On Side Two:

- Read.Me.First:** Introducing Omnis 3 for the Apple ///
- Omnis.Tech.1:** Pathname Lengths and Hard Disk Directories
- Omnis.Tech.2:** Mass Change
- Omnis.Tech.3:** Paging Through Multiple Screens (Entry Formats)
- Omnis.Tech.4:** Screen Dump Command
- Omnis.Tech.5:** A note on formulas
- Omnis.Tech.6:** Testing Search and Mass Change Formats
- Omnis.Tech.7:** A good first use for menus
- Omnis.Tech.8:** Inserting Lines and Characters
- Omnis.Tech.9:** Using Omnis with Catalyst
- Omnis.Tech.10:** Omnis Disk Utilities
- Omnis.Tech.11:** Backing Up Omnis Data Files
- Omnis.Tech.12:** Omnis-3EZ Pieces (AppleWorks) Interfacing
- Omnis.Tech.13:** Imbedding Printer Control Characters in Reports
-

PDS NAME: **LISTIT By Bloom**

DISK ID#: **3UTL-42**

SUBJECT: **Utilities**

BOOTABLE?: **Bootable**

DESCRIPTION: LISTIT prints formatted listings of Pascal programs or other text files. LISTIT prints an identifying title on each page, provides margin and line length control, and optionally prints line sequence numbers. The listing may be printed a page at a time or continuously. Additionally, LISTIT can print selected pages.

PDS NAME: **hpTerm ///**

DISK ID#: **3TEL-10**

SUBJECT: **Telecom**

BOOTABLE?: **Bootable**

DESCRIPTION: hpTerm ///
is an emulation of the HP (Hewlett Packard) HP262x terminal. Information about the program is on side one. The program, written in Pascal, is on side two. Any additional information about this program would be welcome.

PDS NAME: **Access 3270**

DISK ID#: **3TEL.11**

SUBJECT: **Telecom**

BOOTABLE?: **Bootable**

DESCRIPTION: Access 3270 is an Access ///
-like program that allows the Apple ///
to access any computer that supports IBM 3270 communications. Emulates operation on an IBM 3270 keyboard and provides reconfiguration options.

On Side One:

The.Menus: All the menus you'll find while using the program.

Read.Me: Disk description and information.

Review: Review/Tutorial on using the program by Weber Baker.

On Side Two:

Access 3270 Program. Self-Booting and copyable. It can be placed on a hard disk and used under Selector or Catalyst.

PDS NAME: **Epson Screen Printer**

DISK ID#: **3UTL-46**

SUBJECT: **Utilities**

BOOTABLE?: **Bootable**

DESCRIPTION: EPSON SCREEN PRINTER allows the printing of Apple /// text and graphics screens in several different modes. Usable under Basic or Pascal, it is easy to use and will work with any Epson or Epson-compatible printer.

On Side One:

Manual: How to use the Epson Screen Printer

Read.Me: Information about this disk

Epson.Printers: An AppleWorks/3EZP DB bibliography on Epson Printers

On Side Two:

Self-booting Epson Screen Printer Program.

PDS NAME: **DIRECTRANS By DROB/Disk Speed Test**

DISK ID#: **3UTL-47**

SUBJECT: **Utilities**

BOOTABLE?: **Bootable**

DESCRIPTION (Side One): DIRECTRANS will protect the root directory of your hard disk. Loss or unreadability of same for any reason is a major catastrophe as it renders the entire disk useless. DIRECTRANS by DROB transfers the root directory blocks of the hard disc to another location (on a floppy disk). If needed, these blocks can then be transferred to the proper location on the hard disk, restoring the root directory.

DESCRIPTION (Side Two): DISK SPEED TEST is a DOS 3.3 program that is self-booting and will automatically check the speed of your internal disk drive. There is a small potentiometer on the small circuit card at the back of the drive that can be turned to adjust the speed so it is within tolerances. Copy-able with System Utilities but not list-able on the /// (use Copy //+).