/u/fortune* *news*. . .

The Newsletter for Users of the Fortune 32:16 Computer **January 1986/Volume 3 Number 1**

Bad Blocks

And What To Do With Them

This article was contributed by Guy Washburn of Walksoft Corporation, Needham, MA.

All Fortune users at one time or another run into problems with bad blocks on the surfaces of our hard disks.

```
hd error: during transfer of block xxx
on drive 0 in state move
no more spares
```

or

```
Drive 0 has gone bad will not accept any
more requests hd drive status selected
seek complete flushing queue
```

Such messages at the wrong time (is there a right one?) put fear into anyone's heart. But it is not as bad as all that. It is possible to correct the problem, avoiding the inconvenience of a cold boot, and if the error is fixed in time the data in the bad block can often be saved.

Survey the Damage

The first step in the recovery strategy is to find out where the bad block is and whether it is part of an important file.

The error message (or diagnostic results) would give us the disk's physical block number [DPB] which is the number that the hardware uses to refer to the bad block. For example, the error statement reproduced above which begins "hd error:..." gives the number of the bad block.

To find out where the bad block is located on your disk, we have to take into consideration how the hard disk is arranged. Briefly the hard disk (on most Fortune systems) is divided into three partitions (0, 1 and 2). The **zero** partition contains the configuration block, the boot program and 46 spare blocks (which we will mention later). The **one** partition is used for swap space. The **two** partition contains the file system where all of the programs and data files are stored. The configuration block in the zero partition contains all the information about the sizes and beginning locations of the partitions. This information can be seen with the unix command

```
# /etc/rdconf /dev/hd00 <CR>
```

See **Bad Blocks**, page 6

Featured in this Issue. . .

Bad Blocks -- How to use some UNIX commands and the information from your conf block to save bad blocks . . . **Page 1**

MULTIPLAN Hints -- How to change directories or change computers . . . **Page 9**

A Safe *rm* -- A shell script designed to protect you from inadvertently deleting files . . . **Page 2**

Articles and Topics Index -- We include an index of the various articles and topics we've covered in Volumes 1 and 2 of /u/fortune *news* . . . **Page 12**

Master Dealer Program -- Fortune explains their Master Dealer Program and how it might affect you . . . **Page 8**

The BASIC Advisor -- Ray Wannal explains some of intricacies of making reports with IDOL . . . **Page 1**

/u/HELP! -- How to remove all files in a directory *except* certain special files . . . **Page 4**

The BASIC Advisor

Ray Wannall is president of BaSiC Software Corporation in Baltimore and is contributing to this publication independently.

Question: When I first purchased my BAS Business Applications over 2 years ago my dealer (who is long since out of business) informed me that if I purchased IDOL I could write my own software. Is this true? I find the instructions very hard to follow, and I don't have the time to study them. I am not (nor do I wish to be) a programmer, but if you could tell me how to get ANY use from IDOL, I would be eternally grateful.

Answer: Ah, my first potential fan! (Sure hope I don't blow it!) Yes, you can use the Report Definition feature of **IDOL** as long as you don't try to get too fancy. Please have a seat at your terminal and follow along as we create a simple report.

See **The Basic Advisor**, page 10

*/u/fortune *news* is published by The Cambridge Consortium, Inc. It is not affiliated with Fortune Systems Corporation.

IMPORTANT RENEWEL INFO

SPECIAL OFFER CONCERNING SUBSCRIPTION RATES

We are planning a subscription rate increase for **/u/fortune** *news* which will take effect within the next month. We value you as a subscriber and want to give you the special opportunity to renew your subscription to **/u/fortune** *news* before the rate increase goes into effect. So, if we receive your renewal order, with payment, and postmarked by February 28th, we will extend your subscription for another 12 issues at the old price of $36.00. (By the way, you can always know how many issues you have left in your subscription because we print this number on the mailing label on your newsletter.)

We think you will agree that **/u/fortune** *news* is an exceptional bargain at that price - especially since we've been improving it. Don't miss this opportunity to ensure a good thing at a really great price. **Renew your subscription by February 28th and you'll get another 12 issues for $36.00**

/u/fortune *news*. . .

Editors Josh Lobel, Mark Palmerino
Associate Editor Benzion Chanowitz
Advertising Manager Mark Palmerino
Technical Editor Scott Bradner

/u/fortune *news*. . . is published 12 times per year by The Cambridge Consortium, Inc. for /u/fortune, a users' group for users of Fortune 32:16 computers. Neither **/u/fortune** *news* nor The Cambridge Consortium are affiliated with Fortune Systems Corp. Contact us about subscription rates. Membership in /u/fortune is available for $12/year. Quantity discounts are also available.

Back issues are offered at the following prices:

Vol 1, Numbers 1-3	$1/each
Vol 1, Number 4	$.50/each
Vol 1, Number 5-8	$3/each
Vol 2, Number 1-7	$3/each

Although every attempt is made to insure that our contents are error-free and without risk, The Cambridge Consortium is not responsible for losses that occur due to correct or incorrect interpretation of **/u/fortune** *news*.

For more information about us, please call or write:

/u/fortune *news*, 20a Prescott Street, Suite 28, Cambridge, MA 02138. Telephone 617 876-4763.

The text for this newsletter was prepared using the **vi** editor and **Fortune:Word** on a FORTUNE 32:16 computer. It was then transferred using Kermit to a PDP 11/44 computer running UNIX and prepared for typesetting with the **troff** program.

UNIX™ is a trademark of Bell Laboratories, Inc.
Fortune 32:16™, Fortune:Word™ and FOR:PRO™ are trademarks of Fortune Systems, Corp. Multiplan is a trademark of Microsoft. IBM-PC is a trademark of IBM. MS-DOS is a trademark of Microsoft.

©1985 by The Cambridge Consortium, Inc. 1/86

The UNIX Directory

Shellscript -- A Safe *rm*

Our shellscript this month provides a simple way of protecting oneself from accidently removing a file. If you have used the computer for any length of time, then you have accidentally deleted a file. If you have that file backed up, then this accident represents a minor inconvenience (proportional to the difficulty involved in restoring the file). However, we don't always have our files backed up so an accidentally deleted file could be lost forever!

The UNIX **rm** command deletes files! With the **rm** command, once a file is deleted it is impossible to reverse your actions. The creators of **rm** realized how irreversible the use of their command could be and programmed some safeguards into its structure - if you want to use them. For example, if you use **rm -i**, **rm** will interactively inquire if you want to delete the specified file. To really delete the file, you will have to type a **y** to **rm**'s query.

This is not a foolproof safeguard, however. Once you answer **y**, the file is irreversibly erased from your disk. The only way to prevent a file from being irretrievably deleted is to not delete it in the first place. The shellscript we present here shows the basic technique that many UNIX installations use in their ***safe*** **rm** command.

The idea is a very simple one and has two basic components. First, create a shellscript which moves (using the **mv** command) the files to a temporary directory instead of deleting them. Second, periodically clean out this temporary directory so that the files are eventually deleted.

Thus, this command provides a "safety buffer" in that the "deleted" file is really stored on the disk for 24 hours.

The following shellscript accomplishes these two tasks. The first thing this shellscript does is checks to see if your temporary directory exists. We've chosen to set up a subdirectory called **.rm** inside of the user's home directory. If this directory does not exist, our command creates it. Next, we use the UNIX **mv** command to move the files to our temporary directory. Finally, we use the **find** command to check through the temporary directory to delete any files that are older than 1 day.

Thus, our command provides a "safety buffer" in that the "deleted" file is really stored on the disk for 24 hours. If you decide, within that amount of time, that you really didn't want to delete that file, you can get it back by copying it out of the temporary directory.

Continued on next page

The UNIX Directory, Cont'd from page 2

What follows is a first attempt at the shellscript: (Some Notes: First, the pound (#) signs, in a Bourne shell script, indicate that the line is a comment. Thus, only the lines that are not preceded by a pound sign are shell commands. Second, we use **/dev/null** a couple of times. What is this? It is an illusory device which leads to nowhere and is used in shell scripts when we want to eliminate unsightly output. A good example of this is the use of /dev/null on the **dd** command.)

Installation

If you are interested in using this shellscript, you should use an editor to input the above lines into a file called **rm** (you should be logged in as **root**). After this, you will need to make the file executable. To do that, use the **chmod** command as in:

```
$ chmod 775 rm
```

Now, move the file to the **/usr/local/bin** directory. If you don't have a directory called /usr/local/bin, we highly suggest that you create one. This is how it is done:

```
$ mkdir /usr/local/bin
```

Why should you put it in this directory? The answer involves some of the philosophy behind UNIX and the standard ways that UNIX system managers deal with the kind of problem we are presenting here. Scott Bradner, the system manager for the Computer Based Labs at Harvard University, explained that for a command like this *safe rm*, it is best to keep it named **rm**. Why is this? Because, if it is named something else, experience has taught him that users will not use it. And we really *do* want users to use this command since it keeps them from accidentally deleting some file - plus, it keeps the system manager from having to restore accidentally deleted files every 10 minutes.

But wait a minute, what about the *real* **rm** command? Suppose we really want to use it. And what about all the shellscripts on the system that now use the **rm** command, won't they get confused, or worse, bomb out?

That might happen, if we stopped here! However, there is one more thing we need to do when installing this new **rm** command. We need to modify the **paths** of all the users for which we want this new command to be the default (see **The UNIX Directory** in volume 2 number 5 for in-depth information about paths and how to modify them). What we do, is modify the user's paths so that **/usr/local/bin** is the first element. This ensures that the new version is found when the user issues an **rm** command.

As for system shellscripts that should access the real **rm**, we don't have to worry because these are run from the

Continued on next page

```
#
#  Create the temporary directory if it doesn't exist.
#
#  chmod 775... indicates that we want the directory publically
#  readable and executable, but not publically writeable
#
if test ! -d $HOME/.rm
then
        mkdir $HOME/.rm
        chmod 775 $HOME/.rm
fi

#
# Move the files to the temporary directory and
# "touch" each file so that the access time on the file
# will be today.  The mv command does not change the access time.
# We use the dd command to send 1 record of the file to /dev/null
#

for i in $*
do
      mv $i $HOME/.rm
      dd if=$HOME/.rm/$i of=/dev/null count=1 2>&1
done

#
# Put this find command into the background - it will
# delete any files older than 1 day that are in the user's
# temporary .rm directory. The "nice -20" part of this command
# affects the "priority" level of the find command. In
# this case, we've given the find command a low priority
# which means that other commands will take precedence if
# there is competition for the computer's resources
#

nice -20 find $HOME/.rm -atime +1 -exec /bin/rm -f '{}' ';' 2> /dev/null &
```

The UNIX Directory, Cont'd from page 3

root or manager account where we have not modified the path. If you, as a system manager, want some existing system shellscripts to use this new **rm** command, you will have to edit the shellscript and modify it so that it specifically accesses /usr/local/bin/rm.

Version One Has Problems

The above version is simple and under some circumstances will work just fine. However, there are some significant oversights in the simplified design presented above. Here's just one possible problem, what would happen if a user used our program to remove a file called **fred** and then, within the same day, created another file called **fred** and deleted it? The second version of **fred** will overwrite the first one! One final note, when you delete a file from Fortune:Word or Multiplan, the file is really deleted.

We've programmed a more comprehensive safe rm but have decided to postpone publishing it for two months. Why? We'd like to solicit your solutions to this problem. More formally, if you would like, you can submit to us your version of this safe rm. We will examine each of these and judge which one, in our opinion, is the most comprehensive. We will then publish it in the March /u/fortune news along with our version. We will also send the winner his or her choice of any of our FREE Software diskettes. In order to have your version considered it must be postmarked by February 28th. Good Luck!

Mark Palmerino

BACKUP AND RESTORE UTILITY

PROTECTION MADE EASY!

- o Back up all or selected user directories
- o Restore all or selected user directories or files
- o Format diskettes
- o Review backup volumes
- o Menu-driven, easily installed
- o Complete documentation

Also copies the files necessary to preserve your system configuration--a tremendous help if your system ever needs to be cold booted!

$150.00 Backup/Restore Utility
$200.00 Utility + 2 boxes 5 1/4" 96 TPI high-quality diskettes

For more information, call CTD at (313) 540-1282.

To order, make check payable to *Computer Trade Development Corporation.* Send to:

BACKUP/RESTORE UTILITY
Computer Trade Development Corporation
Suite 1386
Birmingham, MI 48010

Company:_____
Contact:_____
Address:_____
City:_____
State:_____ **Zip:**_____

/u/HELP!

Ray Esposito, of The Brass Ring Society (Tulsa, Oklahoma), writes with the following problem. He has a special directory which is used exclusively for floppy backup operations. He has in this directory two special shellscript files that are used to do the backup. These files are called BKUP and BKUP2. The problem is this: how can he delete everything in this directory EXCEPT these two files. In his words, "*What I would like someone to offer (and mostly for my education) is a rewrite of the* **rm *** *command. Is there a way to use this command and still retain a named file?*".

Well, as we have often said, UNIX is a powerful system. It is composed of many versatile commands. Our **UNIX Directory** column is dedicated to presenting and explaining some of these commands. In the past issue (Volume 2 Number 7) the **UNIX Directory** explained a little bit about the **find** command. To answer Ray's question we can say that *there is no need to rewrite* **rm** because **find** is already equipped to do the job.

Let's see how this is done. Recall that we said that one of the basic functions of the **find** command is to find files with a certain name. Thus, the following **find** command would find the files called "BKUP" and "BKUP2" in the **/u/storage** directory:

```
$ find /u/storage -name BKUP \
-name BKUP2 -print
```

Also recall that the following command would delete the above two files:

```
$ find /u/storage -name BKUP \
-name BKUP2 -exec /bin/rm -f '{}' ';'
```

This is exactly the opposite of what we want, which is to delete everything *except* BKUP and BKUP2. This is accomplished by combining the **negation**, **!**, flag with the **or**, **-o**, flag.

```
$ find /u/storage \! \( -name BKUP -o \
-name BKUP2 \) -exec /bin/rm -f '{}' ';'
```

What this command does is finds all the files **not** named BKUP **or** BKUP2 and sends them to the **rm** command. (Note: The backslashes (\) which precede the **!**, (and) are necessary. They are called **escape** characters and tell the shell to interpret the next character differently than if they weren't preceded by the backslashes).

Dwight Rudisdill writes,

Your recent article on correcting a configuration block was most helpful. I can recall four or five hard disks which were discarded ... when the error message stated the configuration block was messed up. I suspect that these current instructions might have saved some lost time and money had our repair section known this option was available.

See **/u/HELP!**, page 14

JFM Business Systems, America's Largest Fortune System Dealer, wishes you a happy and prosperous New Year.

As an authorized Master Dealer, JFM has a nation-wide support and service program to meet your needs.

Additionally, we are offering introductory specials on all Fortune Systems Products.

As a new JFM client, you will receive the following:

- Complete upgrade on your hardware including all Engineering Changes (EC's).
- Complete update of your software from operating system through program products.
- Continuing warranty on your system while under any of our maintenance/service plans.
- Notices and technical letters applicable to your system.

JFM is number one because our clients are number one to us.

Talk to us, then talk to our clients

The Merchandise Mart
Lobby Suite 144
Chicago, Illinois 60654
Call (312) 644-0870

Authorized Fortune Systems Master Dealer

Bad Blocks, Cont'd from page 1

(You need to be logged in as root to do this. Also, you should keep a hard copy of the output from this command since when there is a problem with your system, you may not be able to issue the above command. See **/u/HELP!** for more details.)

To determine the impact of the bad block we need information from the rdconf command including the starting block numbers of partitions one and two.

Now compare the DPB number with the starting blocks of partitions one and two. If the DPB is in partition one (ie. less than the starting block of two and larger than the starting block of one) you can feel lucky and skip to the section on **sparing** blocks. If you are less lucky the DPB will be in partition two (DPB greater then the starting block of partition two). If the DPB is in partition zero (less than the starting block of partition one) the good news is that your data is safe, the bad news is that it is time to do a full system backup and cold boot because things are quite scrambled.

The configuration block in the zero partition contains all the information about the sizes and beginning locations of the partitions.

Assuming that we are dealing with a bad block that falls in partition two, we will now proceed to find out what file is affected by the bad block and then try to save the data in the block. Since this is going to be a software fix we are going to need the two block numbers that UNIX uses to refer to the block. The first is called the file system physical block number [**FSPB**], the second is the file system logical block number [**FSLB**]. We calculate the FSPB by subtracting the starting block of partition two from the DPB. The FSLB is calculated by dividing the FSPB by 2.

Now armed with the FSLB, we can find out what file is involved with the bad block using two unix commands (again, you should be logged in as root).

```
# icheck -b [FSLB] /dev/hd02 <CR>
```

This gives you the inode number of the affected file. *(Note.* **icheck** *is normally distributed with the development utilities but will be available on our fifth FREE Software Diskette to be announced next month.)* With the inode number we can use **find** to give us the file name.

```
# find / -inum [inode number] -print <CR>
```

This gives the full path name of the affected file. Which is also the information we need to retrieve only the damaged file from backups (you do have a good backup don't you?).

Saving The Block

If for some reason there is a problem with the backup (not having one is definitely a problem), there is still a good chance of saving the information in the block with the next procedure. With the FSPB in hand we do the following

```
# dd if=/dev/rhd02 of=/tmp/[FSPB] count=1
bs=512 iseek=[FSPB] <CR>
```

You should see:

```
1+0 records in
1+0 records out
```

If it does not print this, or if a hd error message comes up, try again. It may take several attempts to get it to work, we are dealing with a flaky block, remember. If, after several tries you can't seem to salvage the information in the block, you will have to continue with the steps below. It can be anything from a minor inconvenience to a full blown disaster, but sometimes the information can not be recovered.

Sparing The Block

Once the dd works we can **spare** the block. The process of sparing a block has the result of causing the system to use an alternate good block (located in partition zero) as a replacement for the spared bad block. The kernel performs this operation automatically based on the spares list kept in the configuration block which is read when the system boots up. (Editors note: the real guts of UNIX is often referred to as the **kernel**). To spare a block, then, we must make a new entry in the spares list for the block that has turned bad. To do this we use the following command.

```
# mkconf -i /dev/hd00 /dev/hd00 <CR>
```

At this point, the first entry in the configuration block will be displayed with '**?**' as a prompt. Press `<CR>` to accept the default on each of the questions until you reach this question:

```
Number of Spare Blocks = 46 ?
```

Enter `46 <CR>`. At this point any blocks that have been spared will appear after the spare entry number (they start at three, the first are used for diagnostics). Press `<CR>` to accept each one until you get to one that says '`Bad?`'. On the first '`Bad?`' entry, enter the DPB number from the error message or diagnostic. On the next '`Bad?`' entry type `done <CR>`.

(Note. If the spare entry number comes up with 'Free?'

Continued on next page

Bad Blocks, Cont'd from page 6

next to it you should patiently go through the remainder of the spare blocks and enter `bad <CR>` for each one. Otherwise the kernel will try to spare blocks for us, randomly blowing away blocks that may not even be going bad without saving the information the block contains. Not a good idea.)

The process of sparing a block has the result of causing the system to use an alternate good block (located in partition zero) as a replacement for the spared bad block.

By typing `done`, the configuration block is automatically written back out to the hard disk. We can then check the results of our efforts by using rdconf to read the conf block which should now show our changes.

At this point we shut down the system to allow the change to take effect. You can use the normal shutdown procedure, or we sometimes use this shortcut method:

```
# cat > /tmp/.powerdown <CR>
^D
```

(this turns off the fsck on powerup)

```
# sync <CR>
# sync <CR>
```

Now press the reset button to restart the system and when it is up login as root.

Now we have this copy of the contents of the flaky block hanging out in the **/tmp** directory. By using the following command, we can put the data, that is in this copy in the **/tmp** directory, into the spare block we just defined.

```
# dd if=/tmp/[FSPB] of=/dev/rhd02 \
count=1 bs=512 oseek=[FSPB] <CR>
```

You should see:

```
1+0 records in
1+0 records out
```

And now if everything has gone correctly the data should be restored. It is a good idea to run a file system check at this point to see if everything is ok.

Epilog

Now that we are finished with this digression (salvation?) it does not mean that backups are outmoded. This procedure will work if the error is a "PSOFT" error, for those who have diagnostics. If you don't have the diagnostic, but still feel like taking a swing, feel free. But please note:

THIS PROCEDURE IS NOT FOOLPROOF.
IT IS NOT GUARANTEED TO WORK.
AND IT MAY NOT SAVE YOUR DATA.

But then again, it just might save the day. It is also a good excuse to learn a bit about UNIX and how the system works, which is never a bad idea.

This article is based on a piece in the Fortune hardware reference guide but has been corrected, adjusted and translated into english for the average system operator.

Bulletin Board News

We've recently added the contents of the entire *Fortune Software Catalog* to the bulletin board, along with a great blackjack game. To log in and try it out, dial 617 648-1263 (1200 baud, 24 hours).

Each month or so we will choose a one of topics on the bulletin board to pay special attention to. You'll always know which topic this is because it will be highlighted (if you are signing in with a Fortune terminal).

Have you got some favorite Multiplan shortcuts? Well you're in luck because this month is Multiplan month on the bulletin board. Call in and write them in Notefile 11, Multiplan Notes. We will publish them in **/u/fortune** *news* as we have room.

ENJOY THE PERFORMANCE ADVANTAGE OF

HANDSHAKE™ V

VT100 TERMINAL EMULATOR

This Data Communications Software Package from SST, Inc. permits computer networking between your Fortune 32:16 and any system, database or information service capable of supporting VT100 terminals. You will appreciate the performance advantages far exceeding other VT100 products especially in the graphics mode of display.

HANDSHAKE V allows your Fortune terminal to function as if it were a VT100 terminal, using either ANSI 3.64 or VT52 terminal mode.

- **Copy Files**
- **Transfer Data**
- **Share Resources**
- **Run Programs**
- **Send Messages**
- **Access Databases**

SST INC.

4701 W. Schroeder Dr., Milwaukee, WI 53223

Phone (414)355-6990 Telex 590597 EasyLink 62792652

Ask about our HANDSHAKE ITE, Asynchronous TTY33 Protocol emulator; HANDSHAKE II, 327x emulation for IBM mainframe access; HANDSHAKE III, 5251 emulation for IBM 34/36/38 Systems access; HANDSHAKE DG, Data General 410/460 emulation; HANDSHAKE Prime, Fortune/Prime interface;HANDSHAKE COLT, COmputer Linked Telex.

Fortune's Master Dealer Program

What their new sales structure means

(editor's note: We asked Fortune Systems to explain their Master Dealer distribution network to us. The following article is their description of the program, and of its benefits to end users.)

During the second half of 1985, Fortune announced and began implementing its new "Master Dealer" program, which significantly improved the way the company does business with its dealers and end users.

Since that time, the company has received many positive testimonials from customers and dealers alike, including New York Information Systems President Walter Pacyga, who said, "Our firm gets a faster response with delivery and the resolution of problems now that there is an East Coast contact."

The Master Dealer program is designed for improved service and support of Fortune's dealers and end users. "The Master Dealer program provides several benefits for Fortune users," said Bert Lucas, Fortune's national sales manager. "The first, and foremost, is improved support. Prior to this program, there was really no local service and support mechanism for many of our customers. It was vital that our Master Dealers be able to provide that function to both Affiliates and end users.

"Another real benefit is improved product availability," Lucas added. "Our Master Dealers provide a regional source for new products when a customer orders them," Lucas said. "That means, in most cases, immediate delivery of those products to our end users."

The dealer organization is now structured so that smaller volume dealers order product and receive technical and marketing expertise from large volume dealers, allowing the small dealers to have local expertise and attention.

Altogether, there are now four categories of Fortune dealers: Distributors, Master Dealers, Affiliate Dealers, and Value Added Resellers (VARs).

Master Dealers are large dealers who have the resources to provide full hardware and software service and support for the Fortune product line. There are currently six Master Dealers nationwide.

Affiliate Dealers order their product and receive service, support and marketing expertise from a Master Dealer. The majority of dealers which previously dealt directly with Fortune are now Affiliates of a Master Dealer.

VARs are resellers who obtain product and support directly from Fortune, and have made annual purchase commitments of sufficient size to warrant that relationship. Their way of doing business is basically unchanged by the new program.

If you are in the market for a dealer, the following information should give you some basic guidelines:

QUESTIONS AND ANSWERS

1. What advice would you give end users on the best way to go about purchasing a Fortune System?

Know your requirements and needs, and don't be afraid to talk to several Fortune dealers to see who can give you the most comprehensive support. Look for a dealer that can offer you local service and support. Make sure there's a good level of technical expertise. Make sure the dealer has got good product availability and the best price.

2. What should an end user do when his dealer has gone out of business?

Contact Fortune or a Master Dealer for the names of other dealers in your geographic area. Again, shop around, but remember, good support is not free.

The following is a list of the addresses and phone numbers of the six Fortune Master Dealers:

Western Blue Corp. Sacramento, California (916) 971-1239

J.F.M. Business Systems Chicago, Illinois (312) 644-0870

Computer Trade Development Detroit, Michigan (612) 831-2300

G. T. Hawes Valley Stream, New York (516) 568-0044

J.J.J. Inc. Charlotte, North Carolina (704) 525-0582

Virginia Information Systems Arlington, Virginia (703) 276-7166

Fortune Press Release

The following are excerpts from a press release we recently received from Fortune Systems Corporation.

FORTUNE SYSTEMS ANNOUNCES RESTRUCTURING OF SENIOR MANAGEMENT, REDUCTION IN FORCE

BELMONT, Calif. (January 15, 1986) -- Fortune Systems Corporation today announced a series of actions designed to reduce the company's expenses. Those moves include a reduction in force of 41 employees, approximately 12 percent of the company's workforce of 331, and the restructuring of its senior management team.

See **Press Release**, page 14

Multiplan Shortcuts

A number of readers have passed along some shortcuts for Multiplan which we thought you might be interested in. Here they are:

Use **CMD chl** *to change libraries from the Multiplan Menu.*

Although most of the shortcut commands from Fortune:Word don't work within Multiplan, hitting the **CMD** key followed by the letters **chl <CR>** will let you switch to another directory.

Use the TRANSFER OPTIONS SETUP command sequence to change libraries from within the spreadsheet.

While you are editing a spreadsheet, if you invoke the commands **Transfer Options** you will see several choices on the left, and the word **setup** on the right. If you tab over to this field and type in a new library (e.g. /u/joe) it will change your working directory. When you go to load or save a spreadsheet, it will use the new directory.

Save a spreadsheet in the SYMBOLIC mode to determine NAME assignments or to move your spreadsheet to a different kind of computer.

When you invoke the commands **Transfer Options**, you will see **Normal, Symbolic,** and **Other** displayed on the left. These options tell Multiplan which format to use when loading or saving spreadsheets. **Normal** is the default format that is usually used because it is the fastest, however the **Symbolic** format has several uses.

The **Symbolic** format stores your spreadsheet in ASCII characters. We have explained in past issues that ASCII is a standard method for storing information. It's benefit is that it can be used by many different kinds of computers. If you store your spreadsheet in the **Symolic** mode, you can transfer the information over to an IBM-PC, for example, and then load it into IBM-PC Multiplan in the **Symbolic** mode. You will then be able to use it on the IBM. Note that you still need the appropriate Multiplan program for both the Fortune and the IBM (or whatever other computer you plan to use), but that the spreadsheet information can be transferred without being rekeyed.

The **Symbolic** format has another benefit if you use a lot of Name definitions in your spreadsheet. Once you've got several Named cells, it's difficult to remember what each one refers to. Because the **Symbolic** format is straight ASCII, you can look at the spreadsheet on your terminal, or print it out, and it will list what all of the Named cells refer to. The following is a sample printout:

```
ID;PMP
F;DGOG10
B;Y8;X4
NN;Nrent;ER3C4
NN;Nmonths;ER3C3
NN;Nall;ER3:8C4
C;Y1;X1;K"Annual Expenses"
C;Y3;K"Rent"
C;X2;K500
C;X3;K3
C;X4;ERC[-2]*RC[-1];D;K1500
C;Y4;S;R3;C4;K300
```

You may find it a little difficult to translate, but you will notice the lines that begin **NN;N**, e.g. **NN;Nrent;ER3C4**. What this means is that the name **rent** is assigned to the cell Row 3, Column 4. **all** is the name assigned to Rows 3 through 8, Column 4 (R3:8C4).

Use names to define formulas or lines so that they can easily be copied to another location.

In our spreadsheets, we often have complicated formulas that stretch for several columns within a single row. Below that we have several rows that are the same, but they are separated by blank lines. If we use the **Copy Down** selection, we end up copying into the blank lines, which we would then have to go and clear out. Another option is to **Name** the cells on the row that contain the formulas, e.g. **form1** will refer to columns 5-8 in row 3. Then we can use the **Copy From** command to copy those formulas anywhere we'd like. For instance, if you go to Row 10, column 5 and type **C** for Copy and **F** for From, you will be prompted to enter **From Cells:** at which point just type in **form1**. Then tab over to the **To Cells:** field and hit **EXECUTE**. This will copy the whole string of formulas into Row 10, Columns 5-8.

We also use names to define things like a row of --------- which produce lines on the spreadsheet across several

See **Multiplan**, page 14

KISS YOUR TELEX GOOD-BYE!

UNIX-BASED SOFTWARE NOW OPERATIONAL ON AT&T AND FORTUNE!

HANDSHAKE™ C.O.L.T. (Computer Linked Telex) eliminates all of the Telex hassles. No more Telex equipment fees, no more maintenance downtime, no more timing problems!

Handshake C.O.L.T. links UNIX-based micros with the full range of Western Union EasyLink Instant Mail Services: Telex, Telegram, Cablegram, computer letters and Mailgram Messages.

You can take full advantage of your information management applications while all EasyLink Services take place in the background.

ADDITIONAL FEATURES
- Automatic Send/Receive at preferred times.
- Text-Editor
- Message Log/Audit Trail
- Data base—Telex Directory

For more information contact:

SST INC.
9434 North 107th Street
Milwaukee, Wisconsin 53224
Telephone (414) 355-6990
Telex: 590597 (SST INC)
EasyLink 62792652
(SST INC)

Ask about our other communications software programs available on the AT&T and Fortune 32:16. • Handshake I.T.E. (Interactive Terminal Emulator) links the UNIX and non-UNIX worlds. • Handshake II emulates IBM bisync or SNA/SDLC 3274/76/78. • Handshake III emulates IBM 34/36/38 terminals.

Handshake I, II, Handshake II and Handshake C.O.L.T. are descriptive trademarks of SST Inc. UNIX is a trademark of Bell Laboratories. AT&T is a trademark of AT&T Technologies, Inc. and Fortune 32:16 is a trademark of Fortune Systems Corporation. Telex is a trademark of Western Union.

The Basic Advisor, Cont'd from page 1

First you will need a copy of the file from which the new report will be generated. From the Global Menu enter s4 (**IDOL**) and log in. From the menu, enter **S4** and <CR> to get to the **IDOL DOCUMENTATION FUNCTIONS** selector. Enter 12 (for Record Layouts) and <CR>. (Be sure there is a printer available and turned on). The computer says, "**PRINT ALL FILE LAYOUTS (Y/N)**", you press <F2>. "**PRINT BY DATA BASE (Y/N)**", press <F2>. "**ENTER STARTING FILE NUMBER** ", enter 163 and <CR>. "**ENTER ENDING FILE NUMBER**", enter 163 and <CR>. "**PRINT DATA ELEMENT DOCUMENTATION (Y/N)**", heck no! Press <F2>. "**ENTER PAGE PREFIX**", hit <CR>. "**ENTER STARTING PAGE NUMBER**", hit <CR>. Once the File Layout is printed you will return to selector 4.

Now exit **IDOL** to the Global Menu and go into Business Applications. From the main menu enter 797 and press <CR>. You will be asked to enter the name or number of a file. Type in **CGLMS** (which is the technical name of the General Ledger Master Chart of Accounts) and press <CR>. The file maintenance screen for **CGLMS** will appear and you will have your list of options (ADD, CHANGE, etc.) at the top. Number 5, **REPORT**, allows you to create your custom report using **IDOL**, so enter 5 and <CR>. Watch as the field definitions are rearranged into an almost unreadable order (I hate this, but I guess it is done to free up screen space). Then you will be asked the following questions (remember that <F1> means "yes" and <F2> means "no"):

DEFINE NEW REPORT/INQUIRY (Y/N) - Of course we want to define a new report! Press <F1>.

DEFINE NEW FIELDS (Y/N) - Oh why not. Let us assume that we are able to identify our General Ledger Accounts by looking at the first 20 characters of the account description. Therefore, in the interest of saving space, we will eliminate half of the 40-character Account Description which is held in the file. Press <F1>.

ENTER NEW FIELD LENGTH PRECISION FOR FIELD 33 - Your cursor is behind the word LENGTH expecting a number. Since we want to see the first 20 characters of the description, enter 20 and <CR>. Now the computer wants something called **PRECISION**. This has to do with numeric fields (how many decimal places). Press the <CR> to tell the computer that this is not a numeric field.

ENTER NEW FIELD DEFINITION - This is when we need our printed record layout. Notice that on your printout there is a column titled "**POS IN E$ (IDOL)**". Next to the **ACCT DESCRIPTN** (field 4) in this column, the printout tells us that the "address" of this field in the record is E$(15,40). Since we wish to see only the first half (20 characters) of the description on our report, we should now enter E$(15,20) and press <CR>. (This is entered at the top left portion of the screen. If you make a typing error, you will receive the message "**SYNTAX ERROR**" and be required to reenter.) The numbers in parentheses behind E$ refer to the starting position of the field within the record (15) and the length of the field (40, which we have reduced to 20).

ENTER NEW HEADING - Whatever you enter here will be printed on each page of the report we are defining over the column containing the 20 characters of description just defined. Enter "Account Description" (uppers and lowers are okay) and press <CR>. Then you will be asked for another **ITEM#**. Press <CR> to exit.

DEFINE SPECIAL LOGIC (Y/N) - We won't even go into this. Press <F2>.

SORT REQUIRED (Y/N) - Just to see how it is done, let's have our report sort by Account Type (field 5). Press <F1>.

ITEM# START CHR OR 'RETURN' NUMBER CHRS - For the **ITEM#** enter 5 (the field number for Account Type) and <CR>. Press the <CR> key when asked for START CHR and again for **NUMBER CHRS**. You will then be asked for another **ITEM#**. Press <CR> to exit.

LOGICAL RETRIEVAL REQUIRED (Y/N) - Since you have no desire to be a programmer, we'll skip this question by pressing <F2>.

TOTALS REQUIRED (Y/N) - Let's try it. Press <F1>.

ITEM# - For which field do you wish to see report totals? Enter 8 (for **BAL FWD LAST YR**) and <CR>. True, this will give us a useless report (because debit and credit accounts will be totaled together as positive numbers), but we are merely demonstrating. Bear with us. When you are again asked for an **ITEM#**, press <CR> to exit to the next question. (If you wish to see totals from any of the fields after **BAL FWD LAST YR**, enter a desired field number, between 9 and 32, each time you are asked to enter **ITEM#**.)

SUB TOTALS REQUIRED (Y/N) - Let's say yes. Press <F1>.

ITEM # START CHR OR 'RETURN' NUMBER CHRS - For **ITEM#** enter 5 (Account Type) and <CR>. This will cause the report to print a sub total line each time an account type is completed printing. Press the <CR> for **START CHR** and **NUMBER CHRS**, and again when you are asked for the next **ITEM#**.

Continued on next page

The Basic Advisor, Cont'd from page 10

DEFINE NEW STACKED FIELDS (Y/N) - This allows you to "stack" information when you do not have enough room on one printed line to display all of the fields you have selected. For example, you may wish to put the **ACCOUNT NUMBER** and the **SUB ACCOUNT #** (fields 2 and 3) in the same column. You can read about this in your **IDOL** book if you are really interested. For our purposes here, press <F2>.

ENTER LINE LENGTH (30-131) - One horizontal line of print on a hard copy report contains 131 characters (numbered zero through 131). A CRT screen normally has 80 characters (zero through 79). For a report on the printer, enter 131 and <CR>.

ENTER REPORT HEADING - This will be the title of the report to be printed at the top of each page. May I suggest **"GENERAL LEDGER REPORT SORTED BY ACCOUNT TYPE"** and <CR>?

ITEM# OR 'RETURN' - Now we start selecting fields to report. Start with Company Code (1 <CR>) and select as many as you wish. Be sure to include the fields which will be totaled and sub totaled as well as the Account Type (field 5). When you are asked to enter **NEW COL. HEADING OR 'RETURN'** and **ENTER NEW MASK**, press <CR>. When you are finished selecting (or when you are told **TOO LONG**), press <CR> to end.

END OF SELECTIONS (Y/N) - Press <F1>. (When you press <F2> you go back to **ITEM# OR 'RETURN'**).

SINGLE SPACE (Y/N) NBR LINES TO SPACE? - Press <F1>. If you say <F2> you will be asked for the number of lines to space between each detail line printed. In this case, an entry of 1 means double space, 2 means triple space, etc.

HARD COPY (Y/N) - Press <F1> to see the report on the printer. Select a printer number if asked to do so after this question.

SUMMARY REPORT ONLY (Y/N) - If you want to print total and sub total lines only, you must answer yes. For our sample, press <F2>.

SAMPLE PAGE (Y/N) - We'll skip this with <F2>.

KEY RANGE SELECT (Y/N) - Press <F2> to stay out of trouble for now.

SAVE REPORT PARAMETERS (Y/N) - We may discuss this in another article. For now, press <F2>.

If all went well, the computer will now "write" the coding for the report and print out the report for you. Congratulations. You are now officially a pseudo-programmer, whether you wanted to be one or not.

Another Compressor's Delight Note

We spent many hours working on a comprehensive install script for the **Compressor's Delight** Diskette that many of you have ordered. The install script is a moderately complicated program and as such had a couple of bugs. The first few recipients have called us and we have fixed them. However, you may have received a **Compressor's Delight** with the "buggy" install script. If so, you can always access any of the programs on the disk by mounting it and using the UNIX **cp** command to copy the desired program from the diskette to your hard disk. For example, if you wanted to copy **grep** from the diskette and put it into **/usr/ucb** on your hard disk, do the following:

```
mount /dev/fd02 /f
cp -t /f/bin/grep /usr/ucb
umount /dev/fd02
```

Yes, that's all you have to do. Any other program or file on the diskette can be accessed in the same way. Please accept our apologies for the inconvenience. If you would like an updated copy of the **Compressor's Delight** diskette, simply send back the old one and we'll send you a new diskette immediately.

DO YOU KNOW...

there is an organization that has a thorough knowledge of

FORTUNE BAS
MULTIPLAN IDOL

We have been providing widespread support to our Fortune clients for more than 2 years.

If you need:

- training
- support
- maximum computer performance
- peace of mind relating to processing
- customization
- application results
- minimum errors and confusion
- better ideas on how to increase the power of your Fortune

Call Dan Borrowman at (314) 837-7111

SUPERIOR COMPUTER SYSTEMS, INC.

917 N Hwy 67, Suite 105
P.O. Drawer T
Florissant, MO 63031

Making computers work for you

Index of Articles in /u/fortune news

Many of our new readers have wondered what they've missed in back issues of **/u/fortune news**. What follows is an index of sorts of the articles and topics we have covered in the newsletter from Volume 1 Number 1 to Volume 2 Number 7. We hope this will be helpful to all of our readers. The numbers to the right of the topic is of the form **Volume.Number.Page** - for example, **2.2.5** means Volume 2 Number 2 page 5. For information about ordering back issues, please see the gray box on page 2. (Note: F:W stands for Fortune:Word.)

FORTUNE HARDWARE

32:16 PS and XP	1.1.2
XP 45	1.7.1
Graphics Controller	1.7.4
FIS 1000	1.8.1
FIS 1000	2.2.5
Product Enhancements From Fortune	2.4.1
Hard Disk Preventive Maintenance	2.5.10
Fortune Demonstrates New Products at COMDEX	2.7.1

OTHER HARDWARE

Communications - Modems	1.3.3
Power Supply	1.6.1
Uninterruptible Power Supplies	2.1.1
Uninterruptible Power Supplies	2.2.3
Electronic Bulletin Board Arrives	2.6.1
Using a Modem on Your Fortune	2.7.10

FORTUNE SOFTWARE

Fortune:Word	1.1.3
Fortune Software Updates	1.2.3
F:W Advanced Features	1.5.8
Formatting Chapters with F:W	1.6.1
Subscription Update Services	1.6.4
F:W Glossaries	1.7.1
For:Pro Upgrade	1.7.4
F:W - Making Glossaries and Exception Dictionaries	2.1.6
Extended F:W	2.2.1
New Updates	2.2.1
Fortune Software Index (Part I)	2.3.1
F:W Glossary Entry to Underline Words and Sentences	2.3.2
F:W Courseware	2.3.8
What are Glossary Entries	2.4.1
Fortune Software Index (Part II)	2.4.1
Glossary Entry: Formatting Adresses for Letters and Envelopes	2.4.4
Glossary Entry: Additions to Glossary that Formats Addresses	2.5.1
Looking at Fortune:Windows - Review	2.6.1
Fancy Printing with F:W - The Special Wheels File	2.7.12

OTHER SOFTWARE

Multiplan	1.1.4
KERMIT	1.2.1
PROGRESS	1.3.1
KERMIT, ITE and UUCP	1.3.2
KERMIT and ITE	1.5.5
SC - The Screen Editor	1.6.7
Handshake	1.7.4
Accounting Programs	1.7.4
Training Programs	1.7.4
KERMIT Basics	1.7.10
Link BAS/IDOL/BASIC to Multiplan	1.7.11
PC:WORKS - IBM meets Fortune	1.8.1
Installing *at* and *cron*	1.8.3
A Database Overview	2.2.4
Review about Progress	2.3.1
The BASIC Advisor Televideo Terminals Business Basic Guides Accounting Systems	2.5.1
Electronic Bulletin Board Arrives	2.6.1
The BASIC Advisor Bypassing menu selection process	2.6.8
Review about MINITAB	2.6.12
Softrak Microtrak Project Scheduling	2.7.2
StatusM - A database Search tool	2.7.2
The BASIC Advisor CCNVZ Graphics ERROR messages	2.7.4

Continued on next page

Fortune Users:

JUST POSSIBLY
THE CHEAPEST PHONE CALL
YOU'VE EVER MADE!

- 70 MB disk drives
- 10 MB REMOVABLE disk drives
- Expansion cabinets

- Laser Printers
  8 thru 26 Page Per Minute
  (no extra software required!)

- Genicom Impact printers
- Multi-tech 2400 Baud modems

Shop around... then call us.

Robbins & Associates
(206) 771-2286

* * * * * * * * * * *

Call us **COLLECT** today for a quote

Index, Cont'd from page 12

THE UNIX DIRECTORY

Entering and Leaving UNIX	1.2.2
Global Menu vs. UNIX	1.3.4
File Permissions (chmod)	1.5.6
Installing Programs on the Global Menu	1.6.2
The Password File	1.7.6
Automating Your Computer (*at* and *cron*)	1.8.4
UNIX Do's and Don'ts	2.1.4
Electronic Reminders	2.1.4
Reconfiguring without a Shutdown	2.2.3
Customizing Your UNIX Environment	2.5.2
Managing Disk Usage - Part I (*df* and *du*)	2.6.2
Managing Disk Usage - Part II (*find* and *compress*)	2.7.1

SHELL SCRIPTS

What is a shell script	1.5.2
Print floppy directory	1.5.3
Find lost files	1.3.5
Add/Remove from Global Menu	1.6.6
Dump from Satellite Workstation	1.7.8
Installing *at* and *cron* on your system	1.8.3
Install reminder messages	2.1.5

/u/HELP!

Hooking up printers printcap handshaking baudrate cabling	1.5.4
Large Capacity Hard Disks	1.6.1
Double Bin Sheet Feeders	1.7.5
Hardware Hazards	1.7.5
Basic and Expansion Cabinets	2.1.7
Modifying the /etc/passwd file	2.1.7
F:W and Multiplan Tip	2.2.3
Setting up Expansion Cabinets	2.2.3

BOOK REVIEWS

The UNIX Programming Environment	1.7.7
Multiplan Models for Business	2.1.1
Introducing the UNIX System	2.5.5

IMPORTANT CORRECTION

In the article in last month's newsletter (December, Vol 2, No 1) about modems, we inadvertently said to put the **stty02.rc** file into the **/etc/rc** directory. The **rc** file referred to on page 11 should go in the **/m/rc** directory, e.g. **/m/rc/stty02.rc**. We apologize for the error.

☞SOFTWARE FOR THE FORTUNE 32:16☜

FINANCIAL LINK - stores BAS Income Statement and Balance Sheet data in files. Once stored it can be sent to Multiplan or instantly printed without the need for subsequent sorts or calculations. **$195.00**

MENU LINK - provides software and documentation for adding items to the Global Menu as well as a sub-menu for your use. Includes backup scripts for user directories and BAS/IDOL/BASIC programs and data. **$95.00**

CALCULATION LINK - provides programs for amortization, depreciation, loan repayment, averaging (with graph), linear correlation and breakeven analysis. **$95.00**

KOMPACT PERSONNEL ACCOUNTING - provides for data capture of personal, wage/salary, job, education, salary/wage, appraisal, training, dental, medical and life insurance data. Includes over 80 pre-defined reports. **$495.00**

MODEM LINK - automatically installs software and changes to allow a Hayes 1200 Smartmodem to be used for incoming *and* outgoing calls on one port. **$45.00**

MULTIPLAN LINK & RECORDS PROCESSING LINK - pass data between BAS/IDOL/BASIC and Multiplan, Records Processing, respectively. **$295.00**

Other products are also available. National and International sales.

UNI-KOMP, 9803 Rocktree, Houston, Texas 77040 (713) 937-7782

/u/HELP!, Cont'd from page 6

One thing which did escape me, however, is how does one go about reconfiguring the block when they find that it has become scrambled? What are the commands which are used? The answer is probably embarrassingly simple but in all honesty, I don't know...

We're glad it escaped you because in the article you are referring to (in Volume 2 Number 5, page 10) we did not explain how to reconfigure the block if something should scramble it. Let us re-emphasize to our readers that they should take the time to make a printout of the **configuration block** as it could save them much time and money.

At the UNIX shell, type the following:

```
$ /etc/rdconf /dev/hd00 | lpr <CR>
```

Ok, so now you have the information. How can you use that information? The answer to this question is not "embarrassingly simple," and is beyond the scope of this **/u/HELP!** column. However, in this issue of the newsletter, you will find an article called "Bad Blocks, and what to do with them" that shows one way in which this information can be used to restore bad blocks. Future issues of **/u/fortune** *news* will explore other ways in which the information in the conf block can be used. In general, however, you will be mighty glad if you have saved it when there is a problem with your hard disk. You know what they say, "An ounce of prevention..."

Press Release, Cont'd from page 8

Assuming the office of chief financial officer is Gregg A. Anderson, currently serving as corporate controller. Anderson has been with Fortune since 1982, and has twice served as acting chief financial officer. Roy L. Good has been promoted from director of product development to vice president of engineering, and will continue to oversee development of the company's next generation of hardware and software products.

LeRoy C. Cochran, the company's senior vice president of finance and chief financial officer, Robert J. Ruebel, senior vice president of corporate development, and James F. Ferenz, senior vice president of operations, have resigned. James S. Campbell, Fortune's president and chief executive officer, has assumed responsibility for strategic planning and corporate development.

"The removal of the senior vice president management layer is part of Fortune's expense reduction planning," said Campbell. "These actions accompany an overall one-third cut in expenses, all of which address the need for Fortune to be profitable in 1986."

Cochran added, "Bob, Jim and I are leaving Fortune amid good feelings on all sides. We participated in the planning process for this action and, in fact, spent the past several months grooming our successors and preparing for the transition. In addition, each of us has offered to work with Fortune on future projects, if and as needed."

Although all areas of the company except sales are affected by the reductions, a significant portion of the actions impact manufacturing and reflect the company's recent move to offshore terminal manufacturing. All other hardware and software will continue to be manufactured at the company's Belmont, Calif. plant.

Multiplan, Cont'd from page 9

columns. Then we just name the row and call it **line** referring to, for example, Row 7 Columns 1-15. Then use the **Copy From** technique described above to copy the line anywhere on the spreadsheet.

Use split windows when defining formulas over large portions of a spreadsheet.

Let's say you are defining a grand total, which is the sum of rows 1 through 125, and the total goes in row 128. If you use the cursor to define the cell addresses, you keep paging back through 125 rows, each time you need to define the formula. If you split the sheet horizontally, putting rows 125 to 128 on the bottom half of the screen, and rows 1-15 on the top half, all you need to do is hit the **Change Window** key (f8) to pop back and forth from the beginning to the end of the spreadsheet. For example, put the cursor at row 128. Hit **V** for value, then type **sum(**. Now press **f8**, and the cursor will pop all the way to the top. Put the cursor on the starting cell for the sum. Notice in the formula it says R[-127]C. Then hit the **:** to define the end of the range and the cursor instantly goes back down to row 128, where the formula is. Simply move the cursor up 3 rows, type **)** **EXECUTE**, and your formula is complete.

Use the Multiplan DELETE key (f5) to remove cells that have been externally copied.

The **Xternal Copy** command is used to copy values from one spreadsheet into another. This is a feature that is unique to Multiplan. However, once a cell or cells have been linked to another spreadsheet, they are locked and cannot be blanked or altered without first deleting them. To delete them you have to follow these steps. First, put the cursor on the offending cell. Next type **X** for Xternal and **C** for Copy. You will get the following display at the bottom of the screen:

```
EXTERNAL COPY  from sheet: name:
               to: R1C1    linked: (Yes) No
```

You will be prompted to enter the name of the spreadsheet the value was originally copied from, and the name of the cells that were copied. Next tab over to the field that says **To:** and hit **f5**. The cell addresses will be removed. Now hit **EXECUTE** and the cell contents will be erased.

Josh Lobel

Fortune Users:

G.T. HAWES & CO. Bought this ad to tell you our prices are:

CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP
CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP
CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP
CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP
CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP
CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP
CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP

And our delivery is:

FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST
FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST
FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST
FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST
FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST
FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST
FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST

Fortune VARS, OEM's, System Integrators, Affiliate dealers and all resellers:

Our prices are cheaper than cheap -- our delivery is faster than fast!!

"If you don't see my name on your computer shipment
you probably paid more than you should have."

G.T. Hawes & Co., Inc.

475 W. Merrick Road
Valley Stream, NY 11580
516 568-0044

Free Software

"It may be free, but what is it?!" That's a good question. As we have announced in the past, we presently have 4 disks of free software and are just about to start a fifth. There is likely to be something very useful or entertaining for everyone. What follows is a very brief description of the 4 available disks.

Disk 1: Fortune Utilities

This valuable disk is also known as the Marketing Support Utilities and is from Fortune Systems Corp., although *it is not supported by Fortune Systems.* It includes:

1) Function key programs
2) at & cron
3) sc (screen)
4) hexed
5) autodial programs
6) SWITCH
7) many games

Disk 2: Fortune: Word Tutorial

This disk is also from Fortune Systems, and contains a basic demonstration script of about 30 pages for Fortune:Word and the Extended Fortune:Word modules. The disk also contains numerous files that contain the examples that are used in the script. It's a great way to introduce someone to the power of Fortune:Word (and it might even help you find a feature that you've overlooked).

Disk 3: D.C. Grab Bag

Called the **D.C. Grab Bag** in honor of the Washington D.C. users' group that provided many of the original programs. We have added some exciting programs of our own as well as several that our readers have submitted. Here's a quick rundown of this one:

1) sctutorial
2) sc (screen)
3) ksc
4) FTfix
5) clock
6) Fortune:Word to ASCII
7) kermit
8) Floppy Backup
9) Okidata printcap
10) many shellscripts
11) many more games

Disk 4: Compressor's Delight

Our newest diskette has several very useful utility programs along with many games. Two of the highlights are the programs **compress** and **grep**. The games on this disk include cribbage, craps, hack, several banner programs and a new big clock program.

If you are interested in any, or all, of these diskettes you should send **$10, per diskette**, which includes the cost of the disk and mailing. Please make your check payable to **The Cambridge Consortium, Inc.** and mail it to the return address listed on this page.

/u/fortune *news*
20A Prescott Street
Suite 28
Cambridge, MA 02138

/u/fortune* *news. . .*

The Newsletter for Users of the Fortune 32:16 Computer **February 1986/Volume 3 Number 2**

The BASIC Advisor

Ray Wannall is president of BaSiC Software Corporation in Baltimore and is contributing to this publication independently.

To /u/fortune news readers: In my article for December, 1985, (Volume 2 Number 7) I asked for calls from **IDOL/BAS** *support companies. I depart now from my usual question and answer format to report on the results of that survey. We'll get back to questions in a moment.*

The calls began as soon as the issue hit the streets. I heard from the very large companies with several technicians on staff as well as the small one-person operations. There were a total of nine calls. I asked each caller for the name of his or her company, name of the person to contact, the company's home city and the method of billing. To be completely fair, I present them here in random order with no editorial comments. I will leave it up to the individual end-users to contact whichever companies they wish; but I do suggest that they contact several before making any decisions, because many offer services beyond telephone support.

SUPERIOR COMPUTER SYSTEMS, St. Louis, MO. Contact Dan Borrowman at (314) 837-7111. Telephone support is sold on a subscription basis. Total fee amounts depend upon applications installed.

HANDS COMPUTER SERVICES, Wheaton, IL. Contact Russell Hands at (312) 510-0088. Telephone support is $45.00 per hour with a $10.00 minimum.

DAY PROM COMPUTER, INC., Dayton, OH. Contact Judy Potter at (513) 299-8555. Telephone support is $60.00 per hour, billed by the minute.

BaSiC SOFTWARE CORPORATION, Baltimore, MD. Contact Lynne Crawford at (301) 448-9460. Telephone support is $60.00 per hour billed in fifteen minute blocks.

BEACON SYSTEMS, Palm Springs, CA. Contact Mike Eisen at (619) 323-4555. Telephone support is $75.00 per hour billed in tenths of an hour.

TIM WAGONER (Independent), Columbus, OH. Contact Tim Wagoner at (614) 291-3337. Telephone support is $50.00 per hour billed in fifteen minute blocks.

See **The Basic Advisor**, page 3

Featured in this Issue. . .

The sort Command -- How to use this powerful UNIX command -- in The UNIX Directory . . . **Page 5**

The Bulletin Board Sequencer -- How to use our Bulletin Board more efficiently . . . **Page 12**

The Glossary Entry -- Using Fortune:Word glossaries to make a useful calculator . . . **Page 1**

UNIX Book Review -- A comprehensive book on UNIX that is useful for the beginner . . . **Page 10**

The BASIC Advisor -- Ray Wannal tells us where to find professional help for BASIC problems . . . **Page 1**

/u/HELP! -- The power up sequence, what do those numbers mean? . . . **Page 14**

The Fortune:Word Glossary

It's always struck me as odd that with a $8,000 computer sitting on the desk, I still need my $22 calculator. I guess in the age of specialization that will always be true, but it should be possible to lessen my reliance on the adding machine. This month's column will deal with some simple math functions that can be executed using a glossary entry. The source for this entry is the *Fortune:Word Advanced Glossary* manual which is published by Fortune Systems.

What we have done is to embellish the calculator explained on page 5-66 of the above manual. This entry makes it possible to enter two numbers and add, subtract, multiply, or divide them. This doesn't create an adding machine that will deal with a whole list of numbers, but it is handy when you just want to perform a quick calculation.

The basic principle of its operation is that the numbers are entered and stored as variables. This means that they have a name and can be used later in the program. The operator (e.g. +, -, *, and /) is also entered and stored. The entry then just tests to see which operator has been entered and then performs the desired calculation. Most of the lines are included to jazz up the prompts for data entry and to print the results of the calculation on the bottom of the screen.

Here's the entry. It appears just as it would on your screen. The line numbers on the left are there for reference -- they shouldn't be typed in.

See **The Glossary Entry**, page 8

*/u/fortune *news* is published by The Cambridge Consortium, Inc. It is not affiliated with Fortune Systems Corporation.

☞SOFTWARE FOR THE FORTUNE 32:16☜

FINANCIAL LINK - stores BAS Income Statement and Balance Sheet data in files. Once stored it can be sent to Multiplan or instantly printed without the need for subsequent sorts or calculations. **$195.00**

MENU LINK - provides software and documentation for adding items to the Global Menu as well as a sub-menu for your use. Includes backup scripts for user directories and BAS/IDOL/BASIC programs and data. **$95.00**

CALCULATION LINK - provides programs for amortization, depreciation, loan repayment, averaging (with graph), linear correlation and breakeven analysis. **$95.00**

KOMPACT PERSONNEL ACCOUNTING - provides for data capture of personal, wage/salary, job, education, salary/wage, appraisal, training, dental, medical and life insurance data. Includes over 80 pre-defined reports. **$495.00**

MODEM LINK - automatically installs software and changes to allow a Hayes 1200 Smartmodem to be used for incoming *and* outgoing calls on one port. **$45.00**

MULTIPLAN LINK & RECORDS PROCESSING LINK - pass data between BAS/IDOL/BASIC and Multiplan, Records Processing, respectively.**$295.00**

Other products are also available. National and International sales.

UNI-KOMP, 9803 Rocktree, Houston, Texas 77040 (713) 937-7782

/u/fortune *news. . .*

Editors Josh Lobel, Mark Palmerino
Associate Editor Benzion Chanowitz
Technical Editor Scott Bradner
Advertising Manager Mark Palmerino

/u/fortune *news. . .* is published 12 times per year by The Cambridge Consortium, Inc. Neither **/u/fortune** *news* nor The Cambridge Consortium are affiliated with Fortune Systems Corp. Contact us regarding subscription rates. Quantity discounts are also available.

Back issues are offered at the following prices:

Vol 1, Numbers 1-3	$1/each
Vol 1, Number 4	$.50/each
Vol 1, Number 5-8	$3/each
Vol 2, Number 1-7	$3/each
Vol 3, Number 1-	$4/each

Although every attempt is made to insure that our contents are error-free and without risk, The Cambridge Consortium is not responsible for losses that occur due to correct or incorrect interpretation of **/u/fortune** *news*.

For more information about us, please call or write:

/u/fortune *news*, 20a Prescott Street, Suite 28, Cambridge, MA 02138. Telephone 617 876-4763.

The text for this newsletter was prepared using the **vi** editor and **Fortune:Word** on a FORTUNE 32:16 computer. It was then transferred using Kermit to a PDP 11/44 computer running UNIX and prepared for typesetting on a laserprinter with the **ditroff** program.

UNIX™ is a trademark of Bell Laboratories, Inc.
Fortune 32:16™, Fortune:Word™ and FOR:PRO™ are trademarks of Fortune Systems, Corp. Multiplan is a trademark of Microsoft. IBM-PC is a trademark of IBM. MS-DOS is a trademark of Microsoft.

©1986 by The Cambridge Consortium, Inc. 2/86

FORTUNE SYSTEMS REPORTS FOURTH QUARTER, YEAR END RESULTS

(editor's note: The following text is excerpted from a press release from Fortune Systems Corporation. We've set it in small type to fit most of it in this issue.)

BELMONT, Calif. (February 24, 1986) -- Fortune Systems Corporation today announced that revenues for the quarter ended December 31, 1985 totaled $14,885,000, an increase of 82% over the third quarter. Comparable 1984 revenues were $18,003,000.

During the fourth quarter, reserves and adjustments of approximately $14 million were made for costs associated with down-sizing operations, which included severance costs for the November 1985 and January 1986 reductions in force, costs associated with excess facilities, various reserves against inventory including those associated with streamlining product plans for 1986, and reserves and writedowns of receivables, fixed assets and other assets to estimated realizable value.

These charges increased the Company's quarterly loss to $15,611,000 or $.74 per share.

For the year ended December 31, 1985, the Company reported revenues of $47,510,000 and a loss of $23,561,000 or $1.12 per share.

Fortune's President and Chief Executive Officer, James S. Campbell, stated that the strong down-sizing actions were required to substantially reduce the Company's break-even point and to reestablish Fortune's financial credibility with the investment community and the Company's customers.

"From a product standpoint, our customer satisfaction remains high," Campbell said. "The 25 hardware and software products announced last August strengthened our product performance and are now shipping in volume."

Industry spokesperson Jean Yates recently commented, "Fortune is in a leadership position in office automation systems for UNIX. With the largest installed base for OA-related UNIX supermicros, commercial users continue to purchase Fortune and its excellent Wang-like OA software."

See **Fourth Quarter**, page 9

The Basic Advisor, Cont'd from page 1

KRETA TECHNOLOGIES, INC., So. Bound Brook, NJ. Contact Joseph Kreta at (201) 469-4347. Telephone support is $60.00 per hour billed in fifteen minute blocks.

INTERNATIONAL SYSTEMS CORPORATION, Milwaukee, WI. Contact Dave Meister at (414) 327-5809. Telephone support is $89.00 per hour plus expenses billed in fifteen minute blocks.

CONCEPT OMEGA CORPORATION, Somerville, NJ. Contact Terry Druckman at (201) 722-9565. Telephone support is for non-Fortune users only and is billed on a pre-paid contract basis.

The prices contained in this list are the current billing rates and are subject to change. My thanks to everyone who responded to our survey. Maybe we can do it again sometime. Now back to business.

Question: I seem to be having numerous errors in my **BAS** *programs. When I try to run an Aged Trial Balance in Accounts Receivable I get an error 11. The 4-up mailing labels are feeding an extra line after four labels print, I get an error 13 after Payroll updates the General Ledger, I am getting an error 1 in Accounts Payable when I am trying to run a register and I have a headache. What do you suggest?*

Answer: I may be able to help you with some of your problems, but certainly not all of them. Let's take them one at a time and in the following order.

First, get rid of your headache. I will leave you to your own methods to solve that one.

Feel better? Good. Now let's look at the error 13 in Payroll. The error 13 in **CGLUX0** at statement number 9000 indicates that you have a damaged file. This problem will be eliminated if you reinstall the Payroll Application through the Global Menu. You do not need to worry about overwriting your data files as they are protected in the installation program.

Once that is done you can concentrate on the 4-up label printing problem. This is corrected by the Fortune Systems Technical Tip number 56.1 dated 03/31/83. It is also corrected by the **IDOL/ BAS** update diskette dated May 5, 1983 (**BAS** Revision 1.1 and **IDOL** Revision 1.3). I strongly suggest you find a copy of this disk, because it will solve more problems than you have found yet. It installs very easily through the **s5** Global Menu selection.

Let's move onto your error 11 in the Aged Trial Balance. Somewhere along the line you have managed to delete (remove) a customer from your Customer Master File. Since there is still detail in the form of open invoices, memos or cash payments, the program bombs when it tries to read the file **CCSMS** (Customer Master File). To solve the problem simply add the customer back into the Customer Master File under the same number that was previously assigned to that customer. The next time you run an Aged Trial Balance it should go without error (unless you have deleted more than one of your Customers).

Finally, let's discuss the error 1 in Accounts Payable. Unfortunately I will not be able to handle this one through a newsletter. There is a bad record in the Purchases Header File (**CPYRH**) that needs to be removed, and the job needs to be done by a technician. I suggest you contact one of the support companies listed in this article for assistance.

Question: I just finished reading your article on **CCNVZ** *and feel compelled to comment. You failed to mention that the Company Information Records contain such daily-changing fields as Next Invoice Number and Next Purchase Order Number. Considering this, I believe you should have more strongly emphasized the need to back up the data files on a daily basis.*

Answer: You're right, I goofed. Thank you for bringing this to the attention of our readers.

Question: When programming on the Fortune in Business **BASIC** *I find that I frequently run into an error 33 (Memory Capacity). How do I avoid this?*

Answer: It is entirely possible that you are not keeping a close eye on your program size. You run out of memory when you continue to add lines of coding to an already huge program. If you wish to see how big your program is, type **PRINT PSZ**. This will cause the computer to respond with a number which represents the number of bytes in the program currently in memory. I like to use **PRINT PSZ/256** to get the number of Pages. This corresponds to the number of pages in the ***OPSD** utility when you are defining a new program. In general, I like to keep my programs under 30 pages (although the Fortune can usually handle 40 or more pages of program code). If you find that you are running too high in program bytes, try cutting back by using more subroutines to handle repeating lines of code or by breaking the program into two or three different programs. Also, many programmers tend to include alot of **REM** statements when they are writing code (after all, it beats keeping notes on paper). If this is the case, kill the extraneous **REM's**, swallow your pride and use a pencil and paper for your notes.

ATTENTION All Fortune Systems Users!

Do you want to
- UPDATE
- UPGRADE
- EXPAND
- TRADE

your Fortune System and/or software?

JFM Business Systems, America's Largest Fortune Systems Dealer, probably has the best solution.

As a new JFM client, you will receive the following:

- Complete upgrade on your hardware including all Engineering Changes (EC's).
- Complete update of your software from operating system through program products.
- Continuing warranty on your system while under any of our maintenance/service plans.
- Notices and technical letters applicable to your system.

Talk to us, then talk to our clients

JFM BUSINESS SYSTEMS
The Merchandise Mart
Lobby Suite 144
Chicago, Illinois 60654
Call (312) 644-0870

Authorized Fortune Systems Master Dealer

The UNIX Directory

The *sort* Command

UNIX has many virtues. That is why it is quickly becoming one of the industry's standards as far as operating systems go. One of its main virtues is the many programs that are considered part of UNIX's backbone. One such program is **sort**. In this month's Unix Directory we tackle this powerful UNIX tool.

This program, as its name suggests, is used to sort the contents of a file. Its simplest invocation is:

```
$ sort filename
```

The dollar sign is the system prompt. When used like this, the contents of that file would be alphabetically sorted on a line by line basis. By default, sorting is done with each line considered as a whole unit and the output is sent to the screen. The ordering of the lines is lexicographic and this order is dependent upon the [illegible]

such a file for demonstration purposes (the names have been changed to protect the innocent) and if I display the contents of that file on my terminal it looks like this:

```
$ cat numbers

John Smith 617-354-8956
Mary Jones 203-878-6219
Sheila Fredericks 415-395-5537
George Morton 617-439-3848
Lida Norris 617-372-9466
Lilly Smith 914-869-0032
Robert Hanson 617-645-1328
fred Smith 301-744-0708
Xavier Smith 203-668-7241
```

The file is quite simple and is composed of a first name, a space, a last name, a space, and then the telephone number that includes an area code, an exchange code, and four digits.. Now, if I were to sort the file, I would see:

```
$ sort numbers

George Morton 617-439-3848
```

The UNIX Directory, Cont'd from page 5

Smiths are all together as they should be but *within* the Smiths the lines are sorted on the basis of the phone number (which, in this case, is the "rest of the line" after the last name).

One question at this point is, "What constitutes a field?". The default definition of a field is any continuous "non-white" space. This basically means that spaces and tabs represent the end of one field and the beginning of another field. (In this case, a hyphen would not serve to separate fields and each 10 digit phone number, as they are now shown in the file, would be considered a single field. Later we will show you how to specify a hyphen as the field delimiter.) Thus, the first field in my **numbers** file is the first name. This field is always followed by a space which demarcates the end of the first field. The second field is the last name which is also ended by a space and the last field is the phone number. Then, "+1" has the effect of skipping over the first name and beginning the sort at the last name. Now suppose I want to sort by the last name and then within last names I want to sort by first name instead of the phone number. I would type:

```
$ sort +1 -2 +0 -1 numbers

Sheila Fredericks 415-395-5537
Robert Hanson 617-645-1328
Mary Jones 203-878-6219
George Morton 617-439-3848
Lida Norris 617-372-9466
John Smith 617-354-8956
Lilly Smith 914-869-0032
Xavier Smith 203-668-7241
fred Smith 301-744-0708
```

The above output shows that the last names are indeed in the proper order but also, within Smith, we have the list sorted by first name.

Now, let's sort by phone number:

```
$ sort +2 numbers

Xavier Smith 203-668-7241
Mary Jones 203-878-6219
fred Smith 301-744-0708
Sheila Fredericks 415-395-5537
John Smith 617-354-8956
Lida Norris 617-372-9466
George Morton 617-439-3848
Robert Hanson 617-645-1328
Lilly Smith 914-869-0032
```

This list starts at the lowest number (203) and goes to the highest number (914). We can reverse the order of the sort by using the "r" flag, or argument, like:

```
$ sort -r +2 numbers
Lilly Smith 914-869-0032
etc....
Xavier Smith 203-668-7241
```

(Note: the "r" is preceded by a hyphen. Most commands in UNIX use the convention of preceding flags, or argument, by a hyphen.)

The **sort** command has a number of other flags which control how it works and now we will briefly describe some of them.

The "n" flag is used to direct **sort** to consider the fields that are sorted numerically. This is often important because the lexicographic sort order of digits (considered as a string of characters) is different than the numeric sort order. For example, suppose we had a file called odd.numbers which had the following numbers in it: 11 3 1 7 5 9 13 such that each number was on a different line. Now, if we type:

```
$ sort odd.numbers

1
11
13
3
5
7
9
```

we will get the above output. This is not in numerical order! However, if we type

```
sort -n odd.numbers
```

we would get the proper output. (Question: What would you get if you type `sort -rn odd.numbers`? Note: in this case of using more than one flag the set of flags (i.e. "rn") was preceded by a hyphen. We'll leave for you to discover what the differences are, if any, of `sort -r -n odd.numbers` or even `sort -nr ...`).

The *sort* command is very flexible and uses flags to control its operation.

Another important flag is the "o" flag which controls where the output goes. So far, our examples have simply sent the output to the terminal. That is, we have sorted the contents of a file (e.g., **odd.numbers**) and sent the newly sorted order to the terminal but we have not disturbed the original contents of the file. There are basically two ways to get the output (that would normally come to the terminal) to a file. The first is through the use of redirection. Thus, if you were to type:

```
$ sort -n odd.numbers > odd.sort
```

a new file in your directory would be created and it would contain the sorted contents of the file "odd.numbers".

The other way is to use the "o" flag as in:

Continued on next page

The UNIX Directory, Cont'd from page 6

```
$ sort odd.numbers -o odd.sort
```

So, what's the difference? In most applications there is no difference. However, if you ever want to replace the current contents of a file with those sorted contents you might be tempted to try something like:

```
$ sort -n odd.numbers > odd.numbers
```

The only problem is you will end up with an empty file and the original contents will be nowhere to be found (unless you've been diligent in backing up your system). This is because of the way UNIX deals with indirection. When UNIX encounters a command with a redirection symbol (e.g. ">") it either creates an empty destination file or, if the destination file exists, it "nulls" the destination file, which is the same as saying that UNIX erases the contents of that file. Thus, when you want to write the sorted order of some file over the original order you had better use the "o" flag as in:

```
$ sort -n odd.numbers -o odd.numbers
```

(Would "sort -n -o odd.numbers odd.numbers" have the same effect?)

The sort command can be used to merge two already sorted files together. To do this you would use the "m" flag. Suppose I had a file which had odd numbers (**odd.numbers**) in numerically sorted order and another file which had even numbers (**even.numbers**) in numerically sorted order and wanted to merge those two files together. To do so, I would type:

```
$ sort -mn odd.numbers even.numbers -o all
```

(Note: For the merge option to work properly, each separate file must already be sorted.)

The final flag we will talk about in this article is the "t" flag. The "t" flag is used to control what is considered as the field delimiter. Remember we said that the default field separator is "white space" (i.e., spaces or tabs). If we have a file that uses some other character to separate fields we can still use the sort command. To demonstrate how this works, let's return to our phone **numbers** file. Let's suppose that we wanted to sort that file by the exchanges in the phone numbers. These are the three digits after the area code and the first dash in the phone number. When we dealt with the numbers file previously we had three fields which were demarcated by spaces. If we stay with this way of demarcating fields there is no easy way to sort the lines in the numbers file by the phone exchanges. However, if we call a hyphen (-) the field separator, we can get at the exchange and do the sorting we want. Thus, our **sort** command would look like:

```
$ sort -n -t- +1 -2 numbers

John Smith 617-354-8956
Lida Norris 617-372-9466
etc...
Mary Jones 203-878-6219
```

The syntax of the "t" flag is like any other flag in that it requires a hyphen before it. The field delimiter symbol directly follows it. In the above example we had `-t-`. If we had a file with fields separated by equal signs, we would need to type `-t=`.

There are other flags that we haven't discussed in this article that further allow one to use the flexibility that has been built into the **sort** command. For further information about these flags, and the **sort** command in general, you can consult the **Introduction to FOR:PRO** book. Pages 2-27 to 2-33 describe the **sort** command in some detail and pages 1-80 and 1-81 in the second part of the book give the manual page for the **sort** command.

Mark Palmerino

Stolen System

A Fortune computer was stolen from the Experimental Vehicles Building of the Ford Motor Company over the Christmas Holiday. The serial numbers of the equipment are as follows:

CPU -- 16565; Console -- 12447; Keyboard -- 29482

If you have any information about this equipment, please call Mark Hallman at 313 594-1332 or Anne Fitzgerald at 313 845-3815.

YOU SPENT $$THOUSANDS FOR A FORTUNE

MAXIMIZE YOUR INVESTMENT

with HANDSHAKE™ I.T.E.

SST's INTERACTIVE TERMINAL EMULATOR that allows your Fortune to communicate with many other systems including:

*DEC *AT&T *NCR *Prime
*Honeywell *Burroughs *Qantel
Plus *OCR *Typesetters *N/C Machines

With HANDSHAKE ITE, you can:

* Set and store parameters
* Execute directly from Global Menu
* Automatically log on to remote system with a single keystroke

Use HANDSHAKE ITE to access a broad spectrum of database services including:

*MCI Mail *W.U. EasyLink & FYI Services
*Source *Compuserve *Westlaw *Dow Jones

For additional information, contact:

SST INC.

(414) 355-6990
4701 W. Schroeder Dr., Milwaukee, WI 53223

The Glossary Entry, Cont'd from page 1

```
1  entry C /* PERFORMS A CALCULATION  Page 5-66
2  */
3  {
4      operand1 = 0
5      operator = 0
6      operand2 = 0
7      result = 0
8      formula = "                                    "

10     call posmsg(25,20,formula)
11     formula = "\034HDCalculation:\034Id "

13     call posmsg(1,42,"\034HD Enter Number & Press Execute:
14     \034Id\007")
15     call posmsg(2,50,"")
16     operand1 = keys
17     call clrpos(1,42,38)
18     call clrpos(2,50,30)
19     formula = cat(formula,pic(operand1,","))
20     call posmsg(25,20,formula)

22     call posmsg(1,42,"\034HD Enter Operator (+,-,*,/,%):
23     \034Id\007")
24     call posmsg(2,50,"")
25     operator = key
26     formula = cat(formula,operator)
27     call posmsg(25,20,formula)

29     call clrpos(1,42,38)
30     call clrpos(2,50,30)
31     call posmsg(1,42,"\034HD Enter Number & Press Execute:
32     \034Id\007")
33     call posmsg(2,50,"")
34     operand2 = keys
35     formula = cat(formula,pic(operand2,","))
36     call posmsg(25,20,formula)
37     call clrpos(1,42,38)
38     call clrpos(2,50,30)

40         if(operator == "+")
41         {
42             result = operand1 + operand2
43         }
44         else if(operator =="-")
45         {
46             result = operand1 - operand2
47         }
48         else if(operator == "*")
49         {
50             result = operand1 * operand2
51         }
52         else if(operator == "/")
53         {
54             result = operand1 / operand2
55         }
56         else if(operator == "%")
57         {
58             result = operand1 % operand2
59         }
60     formula = cat(formula," = ")
61     formula = cat(formula,pic(round(result,2),","))
62     call posmsg(25,20,formula)
63 }
```

How it Works

It's probably easiest to see what is happening if you try the entry out once to see how the input and display work. Here's an explanation of exactly what is happening:

Lines 1 - 3 labels the entry with the letter **C**. In order to access this entry, you hit the blue **LF/GL** key followed by the letter C. The notes included between the **/*** and ***/** are comments. Each entry always begins with an open bracket ({).

Lines 4 - 8 initialize all of the variables that will be used. This insures that the variables start with a known value. **Formula** is initialized to all spaces, because it will be used to blank out part of the screen.

Lines 10 - 11 display the blank formula string at the bottom of the screen (row 25, column 20). This clears any previous calculation. Note that this information does not become part of your document. Then it assigns some text to the variable formula which will be used later for display. The numbers like **\034** are octal (base 8) numbers which have special functions. **\034HD** turns on reverse video and **\034Id** turns it off. This is only used to make the output look fancy. They need to be entered in octal because they are not ordinary characters. (For a complete listing of octal codes and their impact, see Appendix C in the *Fortune:Word Advanced Glossary* manual.)

Lines 13 - 16 prompt for the first number and assign your entry to the variable operand1. The prompt appears at the top right of the screen. **operand1 = keys** assigns your key input to **operand1**.

Lines 17 - 18 clear the message area at the top right of the screen. You will see these lines repeated again.

Lines 19 - 20 print out the number you typed at the bottom of the screen. Line 19 does several things. Let's look at it from right to left. The expression **pic(operand1,",")** reformats the number you have typed in, and adds commas after each three digits to make it easier to read, e.g. 3238232 becomes 2,238,232. The expression **cat(formula,pic. . .)** tacks the result of pic onto the end of the variable formula, so it now says **Calculation: 2,238,232**. After it is formatted, line 20 uses **posmsg** to display it at the bottom of the screen.

Lines 22 - 27 prompts for the entry of the **operator** and assigns it to the variable operator. Note that because we use the **operator = key** command

Continued on next page

The Glossary Entry, Cont'd from page 8

rather than **keys**, the computer knows we just want to enter one character, so you don't even need to type a **<CR>** after entering the operator. Lines 26 and 27 are similar to 19 and 20 above. They add the operator onto the end of formula, and display it all on the bottom of the screen. (Formula now equals **Calculation: 3,238,323+**)

Lines 29 - 38 clear the message area, prompt for the second operand, add it to the formula, and display the whole thing on the bottom of the screen.

Lines 40 - 59 do the actual calculation. The expression **if(operator** == "+") compares the operator you've entered to the character +. If they are identical (that's what **two** equals signs mean), then the if statement is true, and the commands between the two brackets will be executed. If the operator does not equal +, then it is tested to see if it equals anything else that seems useful. Note that in each case the math is performed and the answer is assigned to the variable result. All of this is necessary, because although the glossary entry is powerful, it's not smart enough to be able to use the variable operator as an actual command. For instance, **result = operand1 operator operand2** will not work. Note that when we are assigning a value to a variable, a single equals sign is used. When two things are being compared to see if they are the same, two equals signs are used.

Lines 60 - 62 finish off the entry by displaying the result along with the rest of the formula. Line 60 tacks an equals sign onto the string to make it look nice. Line 61 is similar to Lines 19 and 35, with the exception that the **round** command is added. This rounds the result off to two decimal places, e.g. 495.63. Note that several expressions can be combined into one.

Line 63 finishes the entry with a curly bracket, which is always necessary to tell the computer to stop.

This entry will leave the results of your calculation on the bottom of the screen until another calculation erases it, or some word processing command writes over it.

As is noted in Fortune's *Advanced Glossary* manual, there are many improvements that can be made to this calculator, including giving it the ability to read input numbers from a document. One improvement I've been thinking about is making the calculator into an adding machine that would add a column of numbers and display them down the right hand side of the screen as they're entered. If you've created a calculator and want to share it, please let us know.

Josh Lobel

Fourth Quarter, Cont'd from page 2

The Company's cash balances at year-end were $21,725,000. "A major priority established for 1985 was controlling the use of our cash assets," Campbell said. "Ending the year with nearly $22 million in cash shows significant improvement. We were able to drop our use of cash from $28 million in 1984 to $5 million in 1985."

"The microcomputer industry turmoil in 1985 taught everyone that revenue and profitability assumptions needed to be re-thought," Campbell continued. "It became clear that radical long-term structural adjustments rather than short-term fixes were required to position the Company for 1986 and beyond.

"Since the beginning of 1985, headcount has been reduced from 499 to 276, facilities square footage has been decreased 34 percent, and the manufacture of our highest volume product has been moved offshore. In addition, through writedowns and other actions taken, we have reduced net inventory from $19.9 million to $11.8 million, decreased fixed assets from $12.8 million to 8.1 million, and reduced net receivables from $22.0 million to $15.3 million. We expect these actions to improve margins and further reduce operating expenses for 1986 and to establish the foundation for Fortune's long-term growth in the microcomputer industry."

	1985	1984
Quarter Ended December 31:		
Revenue	$ 14,885,000	$ 18,003,000
Loss	$ (15,611,000)	$ (14,933,000)
Loss Per Share	$ (.74)	$ (.70)
Average Shares	21,188,000	21,171,000
Year Ended December 31:		
Revenue	$ 47,510,000	$ 70,190,000
Loss	$ (23,561,000)	$ (21,964,000)
Loss Per Share	$ (1.12)	$ (1.02)
Average Shares	21,082,000	21,598,000

KISS YOUR TELEX GOOD-BYE!

UNIX-BASED SOFTWARE NOW OPERATIONAL ON AT&T AND FORTUNE!

HANDSHAKE™ C.O.L.T. (Computer Linked Telex) eliminates all of the Telex hassles. No more Telex equipment fees, no more maintenance downtime, no more timing problems!

Handshake C.O.L.T. links UNIX-based micros with the full range of Western Union EasyLink Instant Mail Services: Telex, Telegram, Cablegram, computer letters and Mailgram Messages.

You can take full advantage of your information management applications while all EasyLink Services take place in the background.

ADDITIONAL FEATURES
- Automatic Send/Receive at preferred times.
- Text-Editor
- Message Log/Audit Trail
- Data base—Telex Directory

For more information contact:

SST INC.
9434 North 107th Street
Milwaukee, Wisconsin 53224
Telephone (414) 355-6990
Telex: 590597 (SST INC)
EasyLink 62792652
(SST INC)

Ask about our other communications software programs available on the AT&T and Fortune 32:16. • Handshake I.T.E. (Interactive Terminal Emulator) links the UNIX and non-UNIX worlds. • Handshake II emulates IBM bisync or SNA/SDLC 3274/76/78. • Handshake III emulates IBM 34/36/38 terminals.

Handshake I.T.E., Handshake II, Handshake III and Handshake C.O.L.T. are descriptive trademarks of SST Inc. UNIX is a trademark of Bell Laboratories. AT&T is a trademark of AT&T Technologies, Inc. and Fortune 32:16 is a trademark of Fortune Systems Corporation. EasyLink is a trademark of Western Union.

Fortune Users:

JUST POSSIBLY
THE CHEAPEST PHONE CALL
YOU'VE EVER MADE!

- 70 MB disk drives
- 10 MB REMOVABLE disk drives
- Expansion cabinets

- Laser Printers
  8 thru 26 Page Per Minute
  (no extra software required!)

- Genicom Impact printers
- Multi-tech 2400 Baud modems

Shop around... then call us.

Robbins & Associates
(206) 771-2286

* * * * * * * * * * *

Call us **COLLECT** today for a quote

Book Review

A User Guide to the UNIX System (Second Edition) by Dr. Rebecca Thomas and Jean Yates. Berkeley, CA: Osborne McGraw-Hill, 1985. Paperback, 716 pages.

In the past, we have reviewed a number of books devoted to teaching UNIX to the novice user. Each of these was uneven either in the material that was covered or in the clarity of presentation. At long last, we have found a book that we can recommend to the beginning UNIX user. The book exhaustively covers the core of information needed to use UNIX; and it presents this material in clear English that will be comprehensible to the beginner. And, joy of joys, it gives readers a clear idea, at the beginning of each lesson, of what it is they are about to learn. However, this book will also be very handy as a reference tool for the experienced UNIX user. It uses a tutorial approach to review 75 common UNIX commands and it closely examines 44 of these commands. The book also addresses many comments to the person who uses UNIX in business operations. In one of its many useful appendices, the process of System Administration is explained. It discusses all the major releases of the UNIX system and lists all the major computer systems on which the UNIX system is implemented. (In fact, we learned that the Fortune 32:16 system is based on Bell Labs Version 7 and UNIX System III with Berkeley enhancements. Yes! The book does mention the Fortune machine!). The differences between the various releases are discussed. After reading this book, you will have a good overview of the entire UNIX system. And there is a very good index.

Here is the table of contents:

Chapter Headings

1) Understanding the UNIX System
2) UNIX in the Workplace
3) Fundamentals of Using the UNIX System
4) Mastering the Special Features of the UNIX System
5) Text Processing
6) Commonly used UNIX System Commands

Appendices:

A) Resources for Users of the UNIX System
B) Summary of UNIX System V
C) Interacting with your UNIX System
D) Essentials of System Administration
E) Octal Equivalents of ASCII
F) Quick Reference to UNIX System Commands
G) Selected Bibliography

Chapter 1 provides an overview of the UNIX system and a short, helpful history of the system's development. Chapter 2 talks about how the UNIX operating system is an especially congenial environment for running applications; and it outlines the sorts of applications software currently available. Chapter 3 contains 5 tutorial sessions on the basics of using UNIX. Chapter 4 teaches you about the Bourne shell and the C shell. Chapter 5 explains the ed, ex and vi test editors and the nroff text formatter. Chapter 6 discusses 44 UNIX commands in detail with very helpful and clear tutorial examples. The available command options for the various versions of UNIX are also detailed. The first appendix presents information on computers that run UNIX, software that runs under UNIX, and user support of the UNIX system. The second appendix provides extensive lists of the commands available in the different versions of UNIX. The third appendix discusses the use of UNIX as an element in a communication system including how to set up a terminal or modem to a UNIX computer and how to use communications software to access and transfer files from other computers. The fourth appendix provides a helpful primer for the prospective system administrator. The last appendix consists of a bibliography for additional reading.

In short this is the book for beginners who want to learn how to use UNIX. You won't know everything after reading this book. But, you will be in a position to know what it is that you don't know.

Benzion Chanowitz

JONES BUSINESS AUTOMATION CENTER

a FORTUNE SYSTEMS MASTER DEALER

and the JBAC AFFILIATE DEALER NETWORK

are

ORGANIZED TO SUPPORT FORTUNE END-USERS

with

OUTSTANDING PROFESSIONAL SYSTEMS SUPPORT

at REASONABLE PRICES

---

* BAS/IDOL SUPPORT
* FORTUNE:WORD
* MULTIPLAN
* CUSTOM DEVELOPMENT
* COMMUNICATIONS
* HARDWARE UPGRADES
* MAINTENANCE
* LASER PRINTERS
* OCR PAGE READERS
* COMPUTER NETWORKS

Inquire about: **The JBAC "PERFORMANCE SOLUTION"**
Upgrading your present system to
a new SUPER PERFORMANCE SX-70
the Price Performance Leader

The JBAC Affiliate Dealer Network		
ADMS - Advertising Data Mgmt 4173 First Avenue Tucker, GA 30084 (404) 936-8100	**ATOM - Automated Time/Office** 126 Fieldstone Drive Ringwood, NJ 07456 (201) 962-6497	**MicroAge Computer Store** 1939 Poplar Avenue Memphis, TN 38104 (901) 722-8280
BRINC Business Systems 1662 Savannah Highway Charleston, SC 29407 (803) 763-0599	**CompuSystems of Georgia** 2200 Northlake Parkway Tucker, GA 30084 (404) 491-8004	**DataLogix** 3000 Dundee Road Northbrook, IL (312) 291-9225
DataLogix 200 Central Park Avenue Hartsdale, NY 10530 (914) 997-1627	**Future Systems** 2525 Washington Road Augusta, GA 30904 (404) 737-8313	**Advanced Information Systems** 2719 Leo Drive Greensboro, NC 27405 (919) 375-6935
DRS Enterprises 1912 Alyssum Place Winston-Salem, NC 27107 (919) 724-9122	**Synergistic Business Systems** 4300 Paper Mill Road Knoxville, TN 37909 (615) 584-0211	**MultiUser Systems, Inc** 10818 N. W. 6th Court Miami, FL 33168 (305) 758-2007

JONES BUSINESS AUTOMATION CENTER

834 Tyvola Road, Suite 112
Charlotte, NC 28210
(704) 525-0582

The *SEQUENCER* Subsystem on The Bulletin Board

We're very excited about our new electronic bulletin board for Fortune users. We feel that this is a resource that is invaluable to anyone who uses the Fortune computer and products. We hope that this bulletin board will provide you with a number of services that complements what we provide in **/u/fortune** *news*. For example, the bulletin board is expected to become a central node in the exchange of information that can help you, the individual Fortune owner, solve problems. Do you have a problem with Multiplan? Why not describe that problem in the Multiplan notefile. We suspect that within a very short period of time you will find several clever solutions to your problem. Of course, not everyone uses, or has problems with, Multiplan. However, have you ever had a problem with Fortune:Word, Glossaries, any of the BASIC systems, databases, other products, etc. Chances are you would find a solution by posting the problem on our bulletin board.

The bulletin board is easy to use. Anyone can call the number and by signing in as user (or your personal user name, if you have contacted us) you can be connected. After answering a couple of questions regarding your terminal characteristics (such as the number of lines and columns your terminal supports) you will see the first part of the main menu which lists the first 15 notefiles. If you hit a <CR>, the next 8 notefile topics will be displayed. At the moment we have a total of 23 notefiles. Figure 1 shows what you would see if you logged into the Bulletin Board and saw the main menu.

Continued on next page

Figure 1. The Main Menu on the **/u/fortune** Bulletin Board

```
Copyright 1985 by Duncan Communications
   NOTEFILE       SELECTION      INDEX
--------------------
```

Sequencer, Cont'd from page 12

As you can see, we presently have 23 different notefiles. On the one hand, that is good because there are sufficiently fine subdivisions to allow one to place a meaningful question and expect others with similar interests to view one's note. On the other hand, if you wanted to keep abreast of the new problems and issues that are currently being discussed, you would need to look through all the notefiles, patiently (or not so patiently) scanning the topics and responses. Well, this could be tedious, to say the least. On top of that, this problem will only be exacerbated as the number of responses and notefiles grows. That is, there will be more and more responses which will require more and more scanning of old messages which would tax even the most patient of us.

There is a way out of this dilemma which takes advantage of a useful and powerful feature of the bulletin board known as the **SEQUENCER** Subsystem. The **SEQUENCER** subsystem allows you to quickly scan through previously selected notefiles and read only the notes and responses that are newer than the last time you read through the notes.

How to Use the **SEQUENCER** Subsystem

The first thing you must do in order to take advantage of the **SEQUENCER** Subsystem is to create a list of the topics that should be scanned. As an example, let's say you use your Fortune mainly for keeping records for your business. In addition, you and your staff rely heavily on Fortune:Word. Finally, you personally use Multiplan for a variety of purposes. Thus, you would probably be most interested in keeping track of the developments in the following notefiles: bussoftware, accounting, fortuneword, multiplan and basic.

To create the list that the **SEQUENCER** needs, you enter the **SEQUENCER** subsystem from the top level of the bulletin board by typing SEQ. You will then see the following output:

```
Notefile Sequencer Subsystem
------------------------------------
You may type:
    'E'      to edit your sequencer list
    'A'      to engage the sequencer in
               autoscan mode.
  RETURN     to engage the sequencer in
               normal scan mode.
  'HELP'     if you're not sure.
  'QUIT'     to return to the Notefile
               Selection Index.
Sequencer -8:48->
```

In order to create the sequencer list you should choose the '**E**' option. When you do this, the **SEQUENCER** will first print out the notefiles that are currently in your sequencer list. If this is the first time you've used the **SEQUENCER**, then this list will be empty. You will then see a prompt much like the following:

```
Sequencer Editor -8:52->
```

You can do two things at this prompt - you can **add** notefiles to your **SEQUENCER** list or you can **delete** them. Ok, so to add **bussoftware** to the list you simply type:

```
Sequencer Editor -8:52-> add bussoftware
```

You would do the same for any other notefiles you wish to add to the sequencer list. When you are done, you should type a **q**.

Now that you have your list specified, you can use the sequencer to scan through the notefiles automatically. To do this you need to be at the **Notefile Sequencer Subsystem** (which we displayed above). There are two ways to scan through the files. The first is called the **auto-scan mode**. If you choose the **A** option from the menu, the sequencer will cycle through the notefiles on the sequencer list, looking for notes that you haven't read yet. Your unread notes are displayed for you. When you start using the bulletin board regularly, we think you'll find this the method of choice for quickly scanning the notefiles for new notes.

The second method is called the **normal scan mode**. You enter this by typing a <CR> at the sequencer menu. There are two main differences between these modes. The first is that if you choose the **normal scan mode**, then you can modify the time and date that the sequencer uses when determining what note to display. The default time and date, which is used in the **auto-scan mode**, is the time and date that you last read notes in that particular note file. The second difference is that when you engage the sequencer in **normal scan mode**, the sequencer will pause between each note file it scans. Thus, you would see a series of displays (one for each of the notefiles specified in your sequencer list). Each display would look something like:

```
Next file is bussoftware.

Looking for notes written after
16:17 on 1/30/86

You may type:
  RETURN    to read those notes.
   QUIT     to stop the sequencer.
hh:mm mm/dd/yy to reset the date
            for note retrevial.
            hh:mm in 24-hr format.
Sequencer -8:48->
bussoftware: no new notes...
```

To actually reset the time you would type the time using the format show in the above listing (i.e.,

See Sequencer, page 14

/u/HELP!

Question: Every time I turn my Fortune on, it starts printing out the numbers 1 through 9. Why does it do that and what do those numbers mean?

Answer: These numbers are markers that indicate either that certain processes have begun or that they have been properly completed. As you've noticed, there are 9 numbers - but the markers begin even before the first number is seen.

Actually this is a good question because your computer may bomb out during this power up sequence. Then, it may help you or your dealer diagnose and solve the problem if you know what the numbers represent.

Ok, back to this mysterious numerology. When the Fortune is powered up, the first thing you see is the banner - this is the line that says "Fortune Systems 32:16 Please Wait." The banner indicates that the Read Only Memory (ROM) was able to initialize the console. The following table shows the five general phases of power up along with what each of the numbers mean.

First Phase: Power-up

<Banner>	Indicates that the Read Only Memory has been able to initialize the console.
1	The initial hardware diagnostics ran without error.
2	The system has found the boot file on the boot device in the EAROM, has loaded it into memory, and is about to execute.

Second Phase: Boot

3	The boot file has been initiated successfully.
4	The boot program has found the UNIX file and is reading it into Main Memory.

Third Phase: Kernel Initialization

5	The Kernel (Main Memory resident section of the operating system) was loaded into main memory.
6	The Kernel is about to execute the program **/etc/init**.

Fourth Phase: /etc/init

7	The program **/etc/init** was successfully started.

Fifth Phase: /etc/rc

8	The kernel has started **/etc/rc**.
9	Miscellaneous **/etc/rc** tasks completed.

Naturally, it is beyond the scope of this article to fully explain what to do in the case of a failure at any of these points. Sometimes the remedy is quick and painless and sometimes the remedy can be complicated. One safe way to go is to contact your dealer and have him solve your problem. We will periodically print articles on what to do in the case of the simpler problems.

[Fortune Systems Corporation has given us a set of Technical Tips which outline how to fix identified operating problems with the Fortune computer. We are in the process of putting these Tips on our Bulletin Board as a service to our Bulletin Board subscribers. The information here is based on the Technical Tip number 20.1, dated 1/1/83.]

Question: I received your Fortune Utilities Free Software Diskette recently. I installed the function key menu program and used it to shutdown my system (that's shift-function key 16). Afterwards I found that I could only log in as root or manager. Why is that?

Answer: Shutting down the system using this method, i.e., shift-function key 16 from Fortune Utilities Diskette Menu system, creates a special file called **nologin** in the **/etc** directory. If this file exists when anyone other than a superuser (such as root or manager) tries to log in, the **login** program checks the **/etc** directory for this file. If the file exists, it shuts non-superusers out. The solution to this problem is to log in as root or manager and remove the **/etc/nologin** file. This feature is useful when you are doing things like system backups where it is important that no one else use the computer while the backup is in progress.

Mark Palmerino

Sequencer, Cont'd from page 13

hh:mm mm/dd/yy). For example, if you wanted to see all the responses since January 2, 1986 you would type to the sequencer prompt `00:00 01/02/86`.

So, there you have it. By using the **SEQUENCER**, you customize the way the bulletin board works for you. You will no longer have to look through old notes that you've seen before. This means that your sessions will be be more efficient and more productive.

Mark Palmerino

Fortune Users:

G.T. HAWES & CO. Bought this ad to tell you that
Our prices on all Fortune Systems hardware and software are:

CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP
CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP
CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP
CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP
CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP
CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP
CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP CHEAP

And our delivery is:

FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST
FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST
FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST
FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST
FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST
FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST
FAST FAST FAST FAST FAST FAST FAST FAST FAST FAST

Fortune VARS, OEM's, System Integrators, Affiliate dealers and all resellers:

Our prices are cheaper than cheap -- our delivery is faster than fast!!

"If you don't see my name on your computer shipment
you probably paid more than you should have."

G.T. Hawes & Co., Inc.

475 W. Merrick Road
Valley Stream, NY 11580
516 568-0044

Stock Market Monitor

STOCK MARKET MONITOR is a software package designed to help busy people keep track of the stock market.

At a specific time each weekday, it will make a call to your favorite data service (e.g. The Source) and retrieve price and volume information for each stock in your portfolio. MONITOR handles the dial-up, logon and all the conversation, and works at high speed to minimize connect charges.

The information is converted to sensible form ("11 3/4" changed to "11.75") and is stored in your Fortune 32:16 in a daily log. MONITOR also prepares a report showing the daily value of your portfolio.

Price and volume information can also be loaded into a Multiplan model.

STOCK MARKET MONITOR works with a variety of modems and data services, and requires no other special 32:16 software. If you don't currently subscribe to a data service, we can help you sign up to one.

STOCK MARKET MONITOR..... $195
Multiplan Data Transfer Feature..... 50
Subscription to The Source..... 50

acacia technology corporation
1583 Brandywine / San Mateo, CA 94402
(415) 341-7673

VANTINE COMPUTER SERVICES

Professional Services for Business Microcomputer Users

- ☞Sales
- ☞Service
- ☞Software Customization
- ☞Consulting
- ☞Support

Over Two and a Half years supporting Fortune Users

VANTINE COMPUTER SERVICES
9 Priest Road, Sudbury, MA 01776
(617) 443-8412

Authorized Fortune Dealer

We Apologize...

In our last issue, Volume 3 Number 1, we incorrectly listed **Computer Trade Development's** telephone number. We listed the telephone numbers of the six Fortune Master Dealers in an article about the Master Dealer Program. **Computer Trade Development's** correct telephone number is **(313) 540-1282**.

/u/fortune *news*
20A Prescott Street
Suite 28
Cambridge, MA 02138

/u/fortune* *news*. . .

The Newsletter for Users of the Fortune 32:16 Computer **May 1986/Volume 3 Number 5**

The Fortune:Word Glossary

THE BASICS OF RECORDS PROCESSING - Part I

Dick Dow is owner of Dow Information Services which is a technical publications firm located in Rhode Island.

The Records Processing application of extended **Fortune:Word** is a mini-database manager that uses much of the familiar **Fortune:Word** terminology. This month we'll look at how Records Processing is structured and how its documents are set up. Next month we'll go deeper into the procedure for creating and manipulating information using this powerful program.

While there are some limitations to Records Processing compared with a standard database management system, there are also a great many advantages when maintaining certain types of information. One way we use it is for name/address lists: a) where there are not a great number of people involved (1,000 or less), b) where frequent changes (like address changes) are not needed, and c) where we need to print the information using sophisticated formats (merge print letters, for example, or 3x5 reference cards).

One of the really great advantages to Records Processing is that **Fortune:Word** glossaries can be used to create, add to, and change lists. In fact, the procedure for manipulating data is a glossary application built into Records Processing itself. If you are not familiar with glossaries, take time to learn so that you can really appreciate what Records Processing can do. (Editor's Note: See earlier issues of **/u/fortune** *news* for more information on glossaries).

Some disadvantages are that Records Processing uses a lot of memory and it is hard to search for a particular record. This makes it impractical for large lists.

Records Processing uses four types of documents to accomplish its purpose, and a minimum of two documents are used with every Records Processing application. Several others can be used, and documents can be added, changed, archived, printed, or deleted. The process is the same as for any **Fortune:Word** document. Here are the types of Records Processing documents and what they do.

The List Document (the data file)

The list document is the heart of your Records Pro-

See **The Glossary Entry**, page 8

Featured in this Issue. . .

The Fortune:Word Glossary -- Dick Dow begins a series on Records Processing in Fortune:Word . . . **Page 1**

The UNIX Directory -- Computerized Appointment book system - modifications to some programs on our **D.C. Grab Bag** diskette . . . **Page 2**

uucp -- An introduction to automatically transferring files and hooking into computer networks . . . **Page 6**

The BASIC Advisor -- Ray Wannall discusses passwords on BAS menus . . . **Page 1**

News from Fortune -- Financial report, 1.8 operating system update and a new UNIX Coprocessor . . . **Page 10**

The BASIC Advisor

Ray Wannall is president of BaSiC Software Corporation in Baltimore and is contributing to this publication independently.

Question: How do I go about assigning and changing passwords on my BAS menus? I have used the operator code restrictions, but I need more security.

Answer: To be perfectly honest, I am not overly impressed with the operator code restrictions. They work just fine until you try to create a new menu and someone with limited access tries to use it. But since that wasn't your question, I'll explain what I consider to be the best ways to protect the security of your files and selectors.

Most **BAS** systems allow you to manipulate the Selector Header and Detail records without previously installing **IDOL**. This is accomplished by using the 799 option at any **BAS** Selector. Let us assume that you wish to add the password "RUG" to your Business Utilities selection on the main **BAS** Selector. Here's the procedure to follow:

Log into Business Systems (b1 Global) and enter your operator code. When you arrive at the Fortune Systems Business Applications menu, make a note on a piece of paper that the Selector Number in the upper left-hand corner is 10. Also note that Business Utilities is listed as selection number 8. Now enter 799 and press <RETURN>. This will put you into the file maintenance screen for the Selector Dictionary Header Records and give you the option to Add, Change, Delete, Inquire or End. Enter 2 <RETURN> to Change, and you will be asked to ENTER INDEX: ___. Here you enter the

See **The Basic Advisor**, page 14

*/u/fortune *news* is published by The Cambridge Consortium, Inc. It is not affiliated with Fortune Systems Corporation.

DO YOU KNOW...

there is an organization that has a thorough knowledge of

FORTUNE BAS
MULTIPLAN IDOL

We have been providing widespread support to our Fortune clients for more than 2 years.

If you need:

- training
- support
- maximum computer performance
- peace of mind relating to processing
- customization
- application results
- minimum errors and confusion
- better ideas on how to increase the power of your Fortune

Call Dan Borrowman at (314) 837-7111

SUPERIOR COMPUTER SYSTEMS, INC.
917 N Hwy 67, Suite 105
P.O. Drawer T
Florissant, MO 63031

Making computers work for you

/u/fortune *news*. . .

Editors Josh Lobel, Mark Palmerino
Associate Editor Benzion Chanowitz
Technical Editor Scott Bradner
Advertising Manager Mark Palmerino

/u/fortune *news*. . . is published 12 times per year by The Cambridge Consortium, Inc. Neither **/u/fortune** *news* nor The Cambridge Consortium are affiliated with Fortune Systems Corp. Contact us regarding subscription rates. Quantity discounts are also available.

Back issues are offered at the following prices:

Vol 1, Numbers 1-3	$1/each
Vol 1, Number 4	$.50/each
Vol 1, Number 5-8	$3/each
Vol 2, Number 1-7	$3/each
Vol 3, Number 1-	$4/each

Although every attempt is made to insure that our contents are error-free and without risk, The Cambridge Consortium is not responsible for losses that occur due to correct or incorrect interpretation of **/u/fortune** *news*.

For more information about us, please call or write:

/u/fortune *news*, 20a Prescott Street, Suite 28, Cambridge, MA 02138. Telephone 617 876-4763.

The text for this newsletter was prepared using the **vi** editor and **Fortune:Word** on a FORTUNE 32:16 computer. It was then transferred using Kermit to a PDP 11/44 computer running UNIX and prepared for typesetting on a laserprinter with the **dltroff** program.

UNIX® is a trademark of Bell Laboratories, Inc.
Fortune 32:16®, Fortune:Word® and FOR:PRO® are trademarks of Fortune Systems, Corp. Multiplan is a trademark of Microsoft. IBM-PC is a trademark of IBM. MS-DOS is a trademark of Microsoft.

©1986 by The Cambridge Consortium, Inc. 5/86

The UNIX Directory

A Computerized Appointment Book System

If you're like most Fortune owners, you are always looking for ways that your computer can make life easier. It is, after all, a very versatile piece of hardware. Well, wouldn't it be nice if your Fortune could help you organize your life by periodically reminding you of important business phone calls, meetings and other engagements? Wouldn't it be nice if it could help you remember the even more important things like anniversaries, Mother's Day, Father's Day, etc.?

In fact, the Fortune can do that! This month, as promised last month in the article entitled **Free Software** *Highlights*, we are going to accomplish a few things. First, we will explore how to use two programs that are on our **D.C. Grab Bag** diskette. These programs are called **appt** and **note** and they are used to keep a simple computerized appointment calendar. Second, we will show you how to modify them so that they will work with your Fortune even if you don't have Fortune's **Development Utilities**. Third, we will touch upon some UNIX shell programming techniques.

Wouldn't it be nice if your Fortune could help you organize your life by periodically reminding you of important business phone calls, meetings and other engagements?

First, we will explain the principle behind the operation of these programs because later when we modify them we'll have some idea of what modifications will be necessary. The idea behind these programs is very simple. Let's keep one file (which we'll call **calendar**) that has many lines in it. The number of lines is not important (you'll have as many lines in it as you have things to be reminded of). What's important is the structure of the line. The structure of the line will be, symbolically:

```
mmm dd <short message>
```

The above display is meant to represent that each line will have three elements. The first element, mmm, means **month**, like **Jan** or **Feb** or **Jul** or **Oct**. So, the first element represents the month and it is always the first three letters of the month. This convention happens to match the way that the UNIX **date** command prints out the month, as in:

```
Sun Apr 20 16:26:22 EDT 1986
```

Continued on next page

The UNIX Directory, Cont'd from page 2

The above output comes from the **date** command and notice that the second word (in UNIX jargon, the second *argument*) is Apr which is the 3 letter abbreviation for April.

Going back to the structure of the line in the **calendar** file, the dd refers to the day of the month, like **1** or **5** or **19** or **23** and so on. This also matches the way that the **date** command prints out the day of the month - notice the third word (i.e., argument) in the above **date** output.

Now, the final part of the **calendar** line is <short message> which is meant to represent any relatively short message that will remind you of something. If we were to look at a typical **calendar** file, the lines might look something like the following:

```
May 5 Meet with Jim at 2:30 pm for Business Plan
May 8 Arrange for Flowers and Card for Mom
May 9 make sure Flowers and Card are on their way
May 11 Mother's Day
May 26 Memorial Day
Jun 12 Prepare for Business Plan Session Tomorrow
Jun 13 Don't forget to get something for Dad
Jun 13 Finalize Business Plan
Jun 15 Father's Day
Jun 23 Leave for Business Conference in Tahoe
Sep 1 Labor Day
```

As you can see, each line has the same elements, (1) a three letter abbreviation for the month (mmm), (2) a number representing the day of the month (dd) and (3) a phrase that will serve to remind you of something important (<short message>).

Ok, we have a file called **calendar** which has these lines in it. Now we'll have a program (called **note**) which will allow you to add lines to this file. This program is on the **D.C. Grab Bag** diskette and does not need any modification to work on your Fortune. Again, its purpose is simply to allow you to add lines to the **calendar** file. For example, suppose you are sitting at your Fortune and your receive a phone call from a friend who wants to invite you to a party in two weeks. You can add this information to your **calendar** file by running the **note** program from the UNIX shell. Note that there is nothing magic about **note** - you could use your favorite editor to add lines to the **calendar** file.

Finally, the **appt** program is the one that does the hard work. This program needs to do basically one thing - look through your **calendar** file and display any lines that match the requested date. For example, if we were to type appt Jun 13 then we would want to see:

```
Jun 13 Don't forget to get something for Dad
Jun 13 Finalize Business Plan today
```

Now, the **appt** program can do two other special things. First, if it is invoked without a month or a day, by just typing **appt** to the UNIX shell, then it will look through the **calendar** file for any lines that match the *current* day and the *next* day. So, if the date is June 12 and you type **appt** to the UNIX prompt, you would see:

```
Jun 12 Prepare for Business Plan Session Tomorrow
Jun 13 Don't forget to get something for Dad
Jun 13 Finalize Business Plan
```

Also, if you simply give it a month (like May) it will type all the lines that specify things to be done in May.

Ok, so we've explained, in principle, what **appt** should do. Now let's look at the UNIX shellscript that you would find on the **D.C. Grab Bag** diskette: (Note: we show this shellscript as it appears on the **D.C. Grab Bag** diskette. See the **Readers Respond** column for some remarks regarding shellscripts.)

```
# APPT:  A PERSONAL CALENDAR SEARCHING ROUTINE.
#  grep IS USED INSTEAD OF fgrep IN THE LAST TWO
#  CASES BECAUSE fgrep DOES NOT EVALUATE CHARACTER
#  EXPRESSIONS.  USING grep THEN REQUIRES
#  TRANSLATING THE MONTH TO ALL LOWER CASE
#  BECAUSE THE -y SWITCH WILL NOT CONVERT UPPER
#  TO LOWER CASE #  WITH grep.  SPACES INSIDE THE
#  QUOTES ARE REQUIRED.
#  LAST MODIFIED 24MAR84.

case $# in
  0)    set `date`
        fgrep -y "$2 $3 " ${HOME}/calendar
        NEXTDAY=`expr $3 + 1`
        fgrep -y "$2 $NEXTDAY " ${HOME}/calendar;;
  1)    MONTH=`echo $1 | tr A-Z a-z`
        grep -y "$MONTH" ${HOME}/calendar;;
  *)    MONTH=`echo $1 | tr A-Z a-z`
        grep -y "$MONTH $2 " ${HOME}/calendar;;
esac
```

Now don't go away just yet. We know that many of you will think the above shellscript looks like spaghetti (or worse) and will feel that there is nothing you can get from this article. We urge you to stick with this because we think you'll be rewarded with a better understanding of shellscripts and UNIX by the end of the article. In addition, at the end of the article we will present a version of the **appt** program that will work on **your** Fortune computer.

Back to the shellscript. Find the line that says case $# in. This is the beginning of a shellscript loop structure called a *case* statement. The strange looking $# is actually a special shell variable which records the number of arguments with which a shellscript is called. For example, the UNIX shellscript we are using is called **appt**. When we type appt Jun 12, we've called **appt** with **two** arguments and $# equals 2. If we type appt Jun, we've called **appt** with **one** argument and $# equals 1. Finally, if we type **appt** all by itself,

Continued on next page

The UNIX Directory, Cont'd from page 3

we've called **appt** with **zero** arguments and $# equals 0.

If $# equals 0, then the lines between 0) and 1) are performed. In this shellscript, these lines (1) call the UNIX **date** command (set `date`) so that it can find out what month and day it is. Then, (2) it uses a program called **fgrep** to look through the **calendar** file to match lines for the current day. (Last month, in the UNIX Directory, we explained how **grep** works) After that, (3) it adds 1 to the day (NEXTDAY=`expr $3 + 1`) so that it can look through the **calendar** file for any lines that match the next day (again using **fgrep**). So this section of code handles the **appt** command when it is typed with zero arguments and what it does is to get the current date by using the UNIX **date** command. Once it has the current date, it looks through the **calendar** file for any messages for this day. Then it adds 1 to the current day. If the date is May 5, then it will calculate the next day to be May 6. It then looks through the **calendar** file for any messages for May 6. Note that this method of finding messages for the next day will work in most, but definitely not all, cases. What will the shellscript do if the current date is April 30?

Unless you have Fortune's **Development Utilities**, you don't have **fgrep**, so we'll have to modify that later.

If $# equals 1, then the lines between 1) and *) are performed. You may ask, "what in the world does the following line do:

```
MONTH=`echo $1 | tr A-Z a-z`
```

Ok, let's take it apart piece by piece. Look at the section "tr A-Z a-z". **tr** is a UNIX command which stands for **tr***anslate*. We can't explain all the peculiarities of **tr** but suffice it to say the the above **tr** command has the effect of changing all upper case letters into lower case letters. Now just hold on to that thought for a moment. Look at the section "echo $1". Remember we said that $# is a special shell variable. Well, $1 is also a shell variable. It records the first argument of the UNIX shellscript. For example, if we had originally typed appt Jun, then $1 equals **Jun** because **Jun** is the first argument. So, echo $1 has the effect of sending the first argument to the **tr** command (because of the pipe symbol |). Remember that the **tr** command here has the effect of changing upper case letters to lower case letters. Thus, the entire effect of this line (i.e. MONTH=`echo...) is to make the shell variable MONTH have the first argument in all lower case letters. So, Jun would be jun and Mar would be mar and so on.

The **grep** command then looks through the **calendar** file and matches any lines that begin with the month that was requested.

This part will have to be modified as well because the **tr** and the **grep** command are part of the **Development Utilities**.

Modification 1

The first modification we will suggest is very straightforward if you happen to have our **Compressor's Delight** diskette because we have put a version of the **grep** command on it. This modification is really easy because this version of **grep** ignores the case of the letter. That means we can dispense with the **tr** command entirely. So, take a look at Modification 1:

```
case $# in
  0)    set `date`
        grep -f "$2 $3 " ${HOME}/calendar
        NEXTDAY=`expr $3 + 1`
        grep -f "$2 $NEXTDAY " ${HOME}/calendar;;
  1)    grep -f "$1" ${HOME}/calendar;;
  *)    grep -f "$1 $2 " ${HOME}/calendar;;
esac
```

This is a simple modification and will take you about 15 minutes to make the changes to your version of **appt**, using your favorite editor. The **-f** flag on the **Compressor's Delight** version of the **grep** command serves to suppress the output of the filename.

Modification 2

We present this next version of the **appt** command for those of you who have neither Fortune's **Development Utilities** nor our **Compressor's Delight** diskette. The basic strategy here is to use the omni-present but much maligned UNIX **ed** editor to perform the pattern matching function of the **grep** command. In addition, we've created a *very* simple shellscript, called **mconvert**, that takes care of making sure the month argument is of the appropriate format. (Note, this shellscript is not absolutely necessary. You can choose to leave it out. We suggest leaving it in since it will increase the chances of finding a line in the **calendar** file even if you type the month in strange ways.):

```
case $# in
  0)    set `date`
         echo "g/$2 $3/p" | ed calendar
        NEXTDAY=`expr $3 + 1`
         echo "g/$2 $NEXTDAY/p" | ed calendar;;
  1)    MONTH=`mconvert $1`
        echo "g/$MONTH/p" | ed calendar;;
  *)    MONTH=`mconvert $1`
        echo "g/$MONTH $2/p" | ed calendar;;
esac
```

The shellscript mconvert:

```
:
# Handle most likely ways someone will type
# the month.
case $1 in
        JAN|jan|Jan) MONTH=Jan;;
        FEB|feb|Feb) MONTH=Feb;;
```

Continued on next page

The UNIX Directory, Cont'd from page 4

```
        MAR|mar|Mar) MONTH=Mar;;
        APR|apr|Apr) MONTH=Apr;;
        MAY|may|May) MONTH=May;;
        JUN|jun|Jun) MONTH=Jun;;
        JUL|jul|Jul) MONTH=Jul;;
        AUG|aug|Aug) MONTH=Aug;;
        SEP|sep|Sep) MONTH=Sep;;
        OCT|oct|Oct) MONTH=Oct;;
        NOV|nov|Nov) MONTH=Nov;;
        DEC|dec|Dec) MONTH=Dec;;
        *) MONTH=Invalid;;
esac
echo $MONTH
```

Alright, if you look at this last version of the **appt** shellscript, then you'll see lines like:

```
echo "g/$MONTH $2/p" | ed calendar
```

which mimics the function of the **grep** command. We won't explain how this works in detail, but we do refer you to any good book on UNIX which will generally have a chapter on how to use the **ed** command. The basic idea here is that we tell **ed** to look through the **calendar** file for lines that match the month and day and then print these lines out.

The second thing that you must do is to use your favorite editor to create a file called **mconvert** which has the lines that we've displayed above.

Installation

Ok, you've modified your **appt** shellscript to conform to one of our suggestions. Now you'll need to put these programs somewhere on your disk. We suggest that you put them in **/usr/cmd** or **/usr/public** so that it is clear that they are not regular UNIX commands. However, you may opt to put them somewhere else like **/usr/bin**. Once you've done this, you and other users on your system will be able to access both **appt** and **note** in order to keep a computerized appointment system.

If you start using **appt** and **note** to keep a computerized appointment book, you will want to make it as automatic as you can. The most basic way to automate the running of the **appt** program is to add a line to your **.profile** file. For more information on this we refer you to the October 1985 (Volume 2 Number 5) **UNIX Directory** which covered the **.profile** file in-depth.

Problems and Limitations

The programs presented here are not meant to be a complete and feature-packed implementation of a computerized appointment book system. There are some limitations. First, as we've mentioned, the system does not deal with month boundaries very well. When looking for the next day on April 30, it will attempt to find a date of April 31st, which is not an oft-occuring date in the year! Similarly, the system has no way to deal with years. If you happen to keep this system for more than a year, you may begin to get messages that were meant for last year. Of course, you could periodically prune your **calendar** file to prevent this from happening. Second, these programs provide no error checking to speak of. For example, if you type `appt Jam 32` when you wanted to type "Jan 22," you will receive no hint that you did something wrong. In the case of these shellscripts, we can say, "Let the user beware and be careful."

Conclusion

We've explained how to use and modify the two programs **appt** and **note** which you will find on our **D.C. Grab Bag** diskette. These programs are useful for keeping a simple appointment book system on your Fortune computer. The modifications we've given will allow you to use these programs even if you don't have Fortune's **Development Utilities** package or our **Compressor's Delight** diskette.

Mark Palmerino

Fortune Users:

JUST POSSIBLY
THE CHEAPEST PHONE CALL
YOU'VE EVER MADE!

- 70 MB disk drives
- 10 MB REMOVABLE disk drives
- Expansion cabinets

- Laser Printers
  8 thru 26 Page Per Minute
  (no extra software required!)

- Genicom Impact printers
- Multi-tech 2400 Baud modems

Shop around... then call us.

Robbins & Associates
(206) 771-2286

* * * * * * * * * * *

Call us **COLLECT** today for a quote

Linking Computers Together with UUCP

In past issues, we have discussed a variety of communications programs -- **ITE, (Interactive Terminal Emulator)**, **kermit**, and briefly, **uucp (unix to unix copy**, also known as **Fortune to Fortune copy**). Each of these programs has its place for making your Fortune talk to the outside world. **ITE** and **kermit** are useful for connecting to other computers, such as bulletin boards, and **kermit** is especially useful for transferring files between systems. Generally these are used in an "interactive" mode, which means that you log onto the other computer and operate just as if you were a terminal on their system. (A new version of **kermit**, version 4.3, has been released which has some file-server capabilities -- more about that next time.) In contrast, **uucp**, is used to *automatically* transfer files and mail between computers which use the UNIX operating system.

The word **network** is one of the most popular computer buzzwords and can be heard at fashionable parties everywhere. Networks are used to link computers together so that they can share resources such as hard disks, printers, or modems. Generally these network systems involve software and hardware, including the wire necessary to glue them together. **uucp** is a more informal "network" that simply uses software and very basic hardware to transfer files between any computers that are on the network. The computers can be across the room, or across the world from each other. However, both computers require that **uucp** be installed in order for a message to be successfully sent and received.

We have recently started using **uucp** and are quite excited about its capabilities. We have one computer for our bulletin board, plus two others, one in the same office, another several miles away. All of the computers have names -- the one across town is called **jrp**. The best way to illustrate what **uucp** can do is to give an example.

I want to send John a message, at his account "john" on the **jrp** computer, so I give the command:

```
mail jrp!john <CR>.
```

Then I type my message. When I'm all done, I type a `CTRL-d`. That's when the fun begins. My computer signals the modem to call the jrp machine. When the connection is made, it automatically issues the correct information to log on to that machine. Once logged in, the computer transfers my message across the wire, and mails it to john. When john logs in, the computer tells him he has mail, and he reads my message. Note that while all this is happening I'm off doing something else.

Three things should be added to the above example: 1) It is possible to use a menu interface to **uucp** so that you don't have to use UNIX commands, 2) A number of special system files must be set up prior to the routine use of **uucp** (which we will talk about in a future issuc), and 3) In order to send mail you need to have the Fortune **Development Utilities** installed on your system because it includes the UNIX **mail** program.

However, even if you don't have **mail** installed, **uucp** can be used quite effectively to transfer any files from one machine to another. (File transfer that does not use **mail** can be done with other commands. We will discuss this in a future issue). Because **uucp** can be used to transfer **binary** as well as **ASCII** files, it can handle programs, Multiplan spread sheets, Fortune:Word documents, etc.

UUCP can be used to transfer *binary* as well as *ASCII* files, it can handle programs, Multiplan spread sheets, Fortune:Word documents, etc.

The fact that it is automated has another useful purpose. If you have some large files that you want to transfer to your client in California, the cheapest time to do it is at night when the phone rates are low. Using the **at** and/or **cron** programs described in Volume 1 Number 8, your computers can talk to each other at 3 AM while you are sleeping.

Networking Systems -- USENET

As mentioned above, **uucp** can be used to establish informal networks. There are many Fortune users in the Dallas area that are all tied together via **uucp**. This allows them to freely exchange information between all of their systems. Because UNIX has some of its deepest roots in academic environments, many institutions are informally linked via a network called **USENET**. All this network really is is a collection of computers that know about some other computers. If I want to get a message to a friend on the other side of the country without spending money for a phone call or stamp, I could probably tell a friend here in Boston the message, and tell them to relay it to someone else they think might know my friend. Studies have shown that a chain of about five or six people is all that is needed to have the message get to my friend. Each link along the chain is much shorter than the distance from Boston to San Francisco, so I may save on phone expenses.

In practice, this is basically how the **USENET** system works. If I want to send a message from my

See **UUCP**, page 18

SPRING SPECIALS

System	Price
Fortune 32:16 SX 70	CALL
Fortune 32:16 SX 45	CALL
Fortune 32:16 XP 30	CALL
Fortune 32:16 Model 20	CALL
Fortune FIS 1000 Workstation	CALL
Fortune 256KB Memory Addition	CALL
Fortune For:Pro 1.8 Update	CALL

Free Spring Cleaning

For a limited time, JFM will perform complete preventative maintenance on your existing Fortune system at no charge with any purchase.

Call For Details

Talk to us, then talk to our clients

JFM BUSINESS SYSTEMS
The Merchandise Mart
Lobby Suite 144
Chicago, Illinois 60654
Call (312) 644-0870

Authorized Fortune Systems Master Dealer

The Glossary Entry, Cont'd from page 1

cessing application. This is where you store all the pieces of information you need to use: names and addresses, telephone numbers, part numbers, or whatever. The document is made up of *records*, which are groups of similar types of information. Each record contains a number of *fields*, where the actual information is contained. If you are creating a membership list, for example, each member would constitute a record. His or her last name might be one field, first name another field. His street, city, state, zip code, telephone number, and membership number could be other fields in the record. By the same token, if you were creating an inventory list, the part number, part description, unit price, and quantity on hand might each be a field in the record for a particular part. All the member records (that is, all the members) would make up the member file or list document for the membership list. All the part records would make up the part file or parts list document.

Each field has two parts: a *field label* and the field's contents. The label is enclosed in merge symbols (< >). The labels must be identical, and in the same order, within each record. An easy way to ensure this, as well as to help you enter the field contents quickly, is to use a glossary which prints a field label, waits for you to enter its contents, then prints the next label, waits, and so on until you come to the end of the record. The glossary then prints the open and close merge symbols which mark the end of the record and begins the next record for you. An example of this kind of glossary will be included next month.

Since each record must have the same field labels within it, be sure to consider all the information you want to include (although you can make changes later if you decide to add more information; a glossary can help with this as well). A particular record can have some of the fields blank -- just be sure that the labels are there. Another consideration is whether or not to combine information in a field. For instance, if you don't think you'll ever have to find all the members in a particular city or state, you might combine city and state into one field. You can use a separate field for the zip code in order to sort the list for bulk mailings or for geographic distribution by zip. (More on sorting later). The field label can be as long as you want, but we have found that a good way to save space is to keep the labels short, particularly if you will be setting up a large file. Instead of <firstname> and <lastname>, we use <fnm> and <lnm>.

The Format Document

The second type of document associated with Records Processing is the format document. It lets you display the information in your list document in different ways. You may have several of these for each list document. If you have an inventory list, you might have one format document which produces a report, listing your inventory by part number, description, and price. Another format document might print part labels, and still another could be a purchase order, which automatically includes parts whose quantity is below a certain number.

In the format document, you can put tabs, margins and page breaks where you want them. You can also include text that will appear in the finished document along with the information from your list. If you are creating a parts report, the text might be column headings. If you are preparing a merge print letter, the format document text would be the body of the letter. We have a list of schools from which we print labels or envelopes. Some mailings are marked for the attention of the guidance counselor while others go to the principal. The "Attention:" line in the address is included in the format document rather than in the list document to provide flexibility and to save space.

When you create your format document, set it up as you want the final product to appear. In every place you need data from your list document, type in the field label with its enclosing merge symbols. You can use the fields in any order and you can use a field more than once in a format document. If you are printing several records on a single page, type an open merge symbol (<) and use a command called *repeat* followed by a number in parentheses which indicates the number of records you want listed on a page. If you wanted to list the first fifty parts on page one, for example, the next fifty on page two, and so on, you would put an open merge symbol, then "repeat(50)" below it, then the field labels where each field should appear, and finally a close merge symbol (>):

```
<repeat(50)
<partnumber>    <description>    <price>
>
```

A page break should be inserted after the close merge symbol.

In the case of a merge print letter, type the letter in its usual form, substituting the appropriate field labels for the name, address, salutation, etc. You can use a special field called <date1> to put the system date on your letter automatically. (Tip: don't try to use <date1> on a header or footer page -- it won't work.) Here's what the beginning of a merge print letter would look like:

```
<date1>

<firstname> <lastname>
<street>
<city>, <state> <zipcode>

Dear <firstname>,
```

Notice that we used the field label <firstname> in two different places. Notice, too, that we put the

Continued on next page

The Glossary Entry, Cont'd from page 8

punctuation (the comma after the city and at the end of the salutation) where it would normally come. The space between the state and zip is included as well. In other words, treat each field label as though it was the information in your list document. Don't forget to put a page break at the bottom of your letter.

The Control Document

The control document is a special type of glossary document that is used to manipulate your list as it is being copied from the list document to whatever format you have chosen for it. (Note that neither the format document nor the control document changes the list document itself -- just the way it appears for a particular use.) A control document can sort your list by whatever category (or group of categories) you wish. It can also select records that meet certain criteria and use them while ignoring others.

The control document is created like any glossary document, and uses the same type of entries. The commands within the entry may be slightly different and these will be discussed in more detail next month.

Sorting the list, one of the most common uses for the control document, is very simple to set up. We use "entry z" to indicate a zipcode sort for an address list, and the entry would look like this:

```
entry z
{sort<zipcode>}
```

If you wanted your list to be sorted both by zip and alphabetically by last name within the zipcode, use the following:

```
entry z
{sort<zipcode><lastname>}
```

In both these cases <zipcode> and <lastname> are your field labels. If you want to sort in descending order (from the highest number to the lowest or from the Zs to the As), simply alter your entry to read:

```
entry z
{sort descending <zipcode>}
```

You can sort by one field if the length of each of your records is no more than 2,048 characters. You can sort by up to four fields at the same time if the length of each record is no more than 1,024 characters.

The Output Document

The final document in the Records Processing structure is the output document -- a new document created automatically from your list of information as formatted and manipulated by the format and control documents. The output document can appear only on your screen for editing purposes (and can be printed later when you are happy with it), it can be sent directly to the printer without creating

Continued on next page

UNISOFT INC.

Experts in Computer Consulting - Sales - Service

Presents

FORTUNE SYSTEMS HELP

We Provide:

- Expert trouble-shooting
- Years of programming and accounting experience
- *Free Initial Consultation*
- Training Available for:
  * BAS/IDOL/BAS and Multiplan
  * UNIX and Ext. Fortune:Word
- Hardware Repairs & Upgrades
- FORTUNE trained experts at a cost that will surprise you!
- *Trade-Ins Welcome*
- Specializing in:
  * Hard Disk Trade-In
  * Hard Disk Upgrades

Authorized Fortune Systems Dealer
Authorized Fortune Systems Repair Depot

CALL: (303) 671-8600 EMERGENCY HELP: (303) 691-HELP

UNISOFT INC. 14190 E. Jewell Avenue Suite #7 Aurora, Colorado 80012

The Glossary Entry, Cont'd from page 9

a document you can edit, or you can do both simultaneously.

It's a good idea to look at the output document before printing if you have set up new list, format, or control documents. This will let you correct any problems without wasting paper and printer time. When you have an application which you have used many times and which you know needs no editing, you might wish to send it directly to the printer.

An example of the kind of editing that is sometimes necessary is this: we customarily include two address lines in every name and address record. If a particular record uses only one line, we place "xx" in the unused field. When it comes time to create letters, envelopes, or labels, we send the list to an output document where we use global search and replace to delete all the "xx" along with the return that follows them. This eliminates the blank line that would appear if the field were simply left empty. The output document can then be sent to the printer.

Next month we'll publish a glossary for creating list documents, look at selective record retrieval, and how to put everything together.

EQUIPMENT *SERVICE* **SUPPLIES**
ACCESSORIES **TRAINING** *UPDATES*

PIONEER APPLICATIONS
49 WALNUT STREET
WELLESLEY, MA 02181

Pioneer Applications can supply *all* of your needs. As the FORTUNE DEALER with the largest installed base in New England we can handle any situation requiring Fortune expertise.

SERVICING FORTUNE USERS
FOR OVER 3 YEARS

If it's new equipment, service, support, software, consultation or just a friendly voice, Pioneer will respond immediately. **Spring special**: 70MB hard drives, *call now.*

CONTACT:
ALLISON BLACK at (617) 237-5478 or
VICTORIA COLE at (203) 236-2373

News *From* Fortune

Fortune Systems' Finances

Fortune Systems Corporation recently issued its financial report for the first quarter of 1986 ending March 31 and the news is good. The company reported its first operating profit since it went public three years ago. There was a net income of $345,000 on sales of $10.7 million compared with a loss of $3.8 million on sales of $9.8 million for the same quarter last year. The profit amounted to earnings of $.02 per share. In part, the profit can be traced to Fortune's success in lowering its break-even rate of $40 million last year to $10 million this year. This was accomplished by reducing its workforce from 499 to 276, eliminating some senior vice president positions, moving to headquarters that were 34% smaller, and moving most manufacturing operations overseas.

The company also released its 1985 Annual Report reviewing its activity for the past year. We have covered much of this activity in **/u/fortune** *news...* Twenty five new hardware and software products were introduced through a publicity campaign, Stir It Up. (See Volume 2 Numbers 4 & 6 for reviews). Pete Taylor was made President of Fortune Systems International, headquartered in Monte Carlo. Thirty nine percent of Fortune's business came from the international marketplace in 1985 compared with 28% in 1984; and it hopes to further penetrate this market in the future. Alcatel Thompson of France, Fortune's largest OEM customer, accounted for 14% of net revenues in 1985 and has now installed more than 3,000 Micromega systems. The Micromega is its version of the Fortune 32:16. (Thompson owns 18.6% of the outstanding shares of Fortune's common stock). Fortune also signed a series of agreements with the Kirloskar group of companies in India, including the $2.4 million sale of software source code and development equipment. Fortune sees India as a lucrative market for its products. For the year 1985, Fortune lost $23.6 million (or $1.12 per share) on total revenues of $47.5 million.

FOR:PRO 1.8.1.1 Released

Fortune Systems recently announced the release of the FOR:PRO 1.8.1.1 operating system update for the Fortune 32:16 family of computers. This release corrects the diskette drive formatting problem experienced with half-height floppy drives that occurred with FOR:PRO 1.8 and 1.8.1. The update is fully compatible with earlier versions of the 1.8 operating system.

This update allows users with the "TEC" half-height diskette drive to format unformatted flexible disks. It also includes a new version of the UNIX

See News, page 12

USE YOUR FORTUNE TO ACCESS SYSTEMS SUPPORTING VT100 TERMINALS, WITH

HANDSHAKE™

V

VT100 TERMINAL EMULATOR

Send or Receive Data • Run Programs on the Remote
Use Printers and other Resources on the Remote • Copy Files

HANDSHAKE V offers:

- PF1-PF4 Keys
  Scroll/No Scroll
  Print Hard Copy of Screen
  Receive to Printer
- Application Keypad Mode
- Baud Rates up to 9600 BPS
- Choice of Two Terminal Modes:
  ANSI 3.64 or VT52
- Efficient Line Drawing Graphics
- Menu Driven Programs
- Autodial & Autolog Allowing
  Access to More than One System
  With a Single Keystroke
- User Friendly HELP Screens
  Extensive On-Screen Prompts

FOR ADDITIONAL INFORMATION CONTACT:

SST DATA, INC.

4701 W. Schroeder Dr., #100
Milwaukee, WI 53223
Telephone (414) 355-6990
Telex: 590597 (SST INC)
EasyLink 62792652

Ask about other HANDSHAKE Data Communication Products for the Fortune 32:16 including: HANDSHAKE I.T.E. link to the non-UNIX world, Interactive Terminal Emulator; HANDSHAKE II link to the IBM mainframe, BSC or SNA/SDLC, 327x emulator; HANDSHAKE III link to IBM Systems 34/36/38, 5251 emulator; HANDSHAKE DG link to Data General, 410/460 emulator; HANDSHAKE FP, Prime interface; HANDSHAKE COLT, Computer Linked Telex.

News, Cont'd from page 10

kernel and a new version of the boot program. In addition it includes a new disktab file which has descriptions of some new hard disk drives that are compatible with the Fortune system.

The FOR:PRO 1.8.1.1 update is not protected and is available for all users of FOR:PRO 1.8. In order to receive a copy, contact your dealer. It can be installed using the **S5** option on the **Global Menu**.

We have also put a copy of the update on our bulletin board, and it can be downloaded from the system if you have the kermit communications program running on your computer. There is an uncompressed version which will take about an hour at 1200 baud. A compressed version of the programs is also on the system which will reduce the transmission time to about 45 minutes. (We are also about to put a 2400 baud modem on our system -- you can try logging in at that speed if you have a 2400 baud modem.)

The following is excerpted from a product announcement that we received from Fortune Systems Corporation.

DESCRIPTION

Fortune Systems announces the availability of the Fortune UNIX Coprocessor. The UNIX Coprocessor opens the doors to the UNIX world for users of IBM PC XT™, IBM PC AT™, and compatibles. It offers performance and responsiveness not regularly seen in the MS-DOS™ world. The UNIX Coprocessor is also bundled with Fortune:Word, Fortune's professional word processing package; with FOR:PRO/PC, Fortune's enhanced UNIX operating system; and with a Fortune Workstation Keyboard. Fortune:Word and FOR:PRO/PC are offered for the UNIX Coprocessor in English, French, and German.

The UNIX Coprocessor features an 8-MHz MC68010-family processor, the MC68451 Memory Management Unit, and 512 kilobytes (KB) of main memory. This memory is expandable to 2 megabytes (MB). The UNIX Coprocessor offers the same proven Fortune software protection mechanism as the Fortune 32:16 family of products.

The Fortune Workstation Keyboard is the only keyboard commercially available in 14 different language configurations, including French, German, Spanish/Portuguese, Greek, Turkish, Norwegian/Danish, French Canadian, British/U.K., French/Swiss, German/Swiss, Swedish/Finnish, Italian, and Dutch. It has a low profile, and is ergonomically designed to fit into any office automation environment. The Fortune Workstation Keyboard combines a Wang-like word processing keyboard layout with additional MS-DOS function keys, making it both practical and flexible. Included with the UNIX Coprocessor bundle is special keyboard handler software that, when loaded into the standard MS-DOS operating system, allows the Fortune Workstation Keyboard to work with an IBM PC or compatible machine.

The Fortune UNIX Coprocessor features the 8088 Host Operating System (HOS). The HOS Operating System supports the interface between Intel's 8088 processor and Motorola's 68010 processor. It also logically defines MS-DOS devices and allows the user to switch between FOR:PRO/PC and MS-DOS. MS-DOS Version 2.1 or later is the host operating system for the Fortune UNIX Coprocessor. All system I/O to devices such as printers, terminals, disk drives, and modems is channeled through the host environment. All I/O and interrupts are handled by MS-DOS block and character device drivers.

A Utility program on the Fortune UNIX Coprocessor supports file conversions between MS-DOS and FOR:PRO/PC. The user can access files in MS-DOS and then convert them to FOR:PRO/PC. Files can be copied back and forth between both operating systems.

The Fortune UNIX Coprocessor also provides true multiuser, multitasking capabilities. It allows users to simply toggle between FOR:PRO/PC and MS-DOS applications while UNIX applications are executed in background mode.

PERFORMANCE AND LIMITATIONS

As MS-DOS is basically a single-user, single-tasking environment, the use of the I/O channels should be minimized during multiuser sessions. All system I/O to devices is channeled through MS-DOS.

The Fortune UNIX Coprocessor has been tested with the Fortune MS-DOS/UNIX Workstation, and IBM PC XT/AT. These are the only products currently recommended for use the the Fortune UNIX Coprocessor. If a customer desires to use the UNIX Coprocessor in conjunction with an IBM PC XT/AT compatible product, arrangements can be made to test and qualify the UNIX Coprocessor with that product, provided that Fortune receives a purchase commitment for 25 or more UNIX Coprocessors.

The UNIX Coprocessor requires PC-DOS or MS-DOS 2.1 or later. It also requires Fortune-supported software on 48 TPI diskettes. Fortune recommends that the system have a hard-disk capacity of 10 MB or more.

Note: As with many add-on cards in the IBM PC marketplace, care should be taken to insure the ability to coexist with previously installed hardware and software.

See News, page 18

JONES BUSINESS AUTOMATION CENTER

a FORTUNE SYSTEMS MASTER DEALER

and the JBAC AFFILIATE DEALER NETWORK are ORGANIZED TO SUPPORT FORTUNE END-USERS with OUTSTANDING PROFESSIONAL SYSTEMS SUPPORT at REASONABLE PRICES

* BAS/IDOL SUPPORT
* FORTUNE:WORD
* MULTIPLAN
* CUSTOM DEVELOPMENT
* COMMUNICATIONS
* HARDWARE UPGRADES
* MAINTENANCE
* LASER PRINTERS
* OCR PAGE READERS
* COMPUTER NETWORKS

Inquire about: **The JBAC "PERFORMANCE SOLUTION"**
Upgrading your present system to
a new SUPER PERFORMANCE SX-70
the Price Performance Leader

The JBAC Affiliate Dealer Network		
ADMS - Advertising Data Mgmt 4173 First Avenue Tucker, GA 30084 (404) 936-8100	**ATOM - Automated Time/Office** 126 Fieldstone Drive Ringwood, NJ 07456 (201) 962-6497	**MicroAge Computer Store** 1939 Poplar Avenue Memphis, TN 38104 (901) 722-8280
BRINC Business Systems 1662 Savannah Highway Charleston, SC 29407 (803) 763-0599	**CompuSystems of Georgia** 2200 Northlake Parkway Tucker, GA 30084 (404) 491-8004	**DataLogix** 3000 Dundee Road Northbrook, IL (312) 291-9225
DataLogix 200 Central Park Avenue Hartsdale, NY 10530 (914) 997-1627	**Future Systems** 2525 Washington Road Augusta, GA 30904 (404) 737-8313	**Software Sciences** 3100 Evans Avenue Fort Myers, FL 33901 (813) 334-1646
DRS Enterprises 1912 Alyssum Place Winston-Salem, NC 27107 (919) 724-9122	**Synergistic Business Systems** 4300 Paper Mill Road Knoxville, TN 37909 (615) 584-0211	**MultiUser Systems, Inc** 10818 N. W. 6th Court Miami, FL 33168 (305) 758-2007

JONES BUSINESS AUTOMATION CENTER

834 Tyvola Road, Suite 112
Charlotte, NC 28210
(704) 525-0582

The Basic Advisor, Cont'd from page 1

10, which represents the Selector Number to change, and press <RETURN>. Defaults for Selector 10 are displayed and you are asked to ENTER FIELD NUMBER TO CHANGE OR 'F4' TO END: . Press <F4> to end and <F2> to answer no to the HARD COPY question. You are again asked to ENTER INDEX.

At this point I need to go off on a tangent for a moment. When **IDOL/BAS** was first developed in the 1970's, it was on the BASIC IV minicomputer (micros were not yet available). The old BASIC IV's had a total of four function keys, rather than the sixteen available on the Fortune. When the **BAS** was translated, function keys six through nine were defined for Help, Next Record, Skip and Preset Value respectively. Unfortunately someone forgot to expand function key usage for **IDOL**. It still uses the original four: <F1> (Yes or 'CR'), <F2> (No, Preset Value or Next Record), <F3> (Back up or Same Record) and <F4> (End, Help or Previous Record). With all of this in mind, let's get on with adding a password.

At ENTER INDEX, press the <F2> key to bring up the first Detail Entry for Selector 10 (which should be selection 1, General Ledger). Since we do not wish to change this selection, press <F4> to end and <F2> for no hard copy. If you wish, you may continue scanning with <F2> until you get to the Business Utilities selection, or you may skip to selection 8 at ENTER INDEX by entering *8 (asterisk and 8). (The *8 could have been entered when we were still at the Header Screen, but I wanted to demonstrate how the old function key system worked.) Now you should be looking at the Selector Detail Record for the Business Utilities. Notice that field number 18 is labeled SEL PASSWORD. At the ENTER FIELD NUMBER TO CHANGE question, enter 18 and <RETURN>. At field 18 you are allowed three characters for the password. You may use letters, numbers, spaces (one or two) and other normal typewritten characters when defining passwords. We will enter RUG <RETURN>, <F4> to end and <F2> for no hard copy. At the ENTER INDEX question press <RETURN> or <F1> to exit the Change mode. Enter <F4> or END <RETURN> to go back to the menu.

Before the menu comes up you will get a message that says PLEASE WAIT! DEFINING SELECTOR SCREEN. In a second or two the screen will be displayed and you can try out your new password. Enter 8 (for Business Utilities) and <RETURN>. Now you will be asked to ENTER PASSWORD at the bottom of the screen, and whatever you enter will not be printed on the screen. If you enter the wrong password you will get the message, INVALID PASSWORD, RETURN TO CONTINUE. The computer will give you three tries at the password. If you fail in three attempts, you will be automatically logged off to ENTER OPERATOR CODE. In this case, a note is written to the END OF DAY files, and your mischievous activities will be reported! In order to access the Business Utilities, enter RUG and <RETURN>.

If you wish to remove the password from the Selector, follow the procedures for adding until you arrive at the point where you entered RUG at field 18 of the Selector Dictionary Detail Records. Instead, enter three spaces to cancel the password.

Adding passwords to your **IDOL**-defined data files requires that **IDOL** be on the system. If you have **IDOL**, here's the procedure to follow to put the password RUG onto your General Ledger Master Chart of Accounts file (CGLMS).

First find the file number of CGLMS with the DFN option at any **BAS** or **IDOL** selector. (If you do not remember how to use DFN, see **The BASIC Advisor, /u/fortune** *news* vol. 2 no. 6.) The file number will be 163; write it down. Go back to the selector and enter 798 <RETURN> to bring up the FILE/ELEMENT DICTIONARY data entry screen.

Again you are offered the options of ADD, CHANGE, DELETE and INQUIRY. Enter 2 (to change) and <RETURN>. You are then asked to enter a FILE NUMBER. Type in 163 and press <RETURN>. This brings the header information for CGLMS onto the screen, and you are asked to ENTER FIELD TO CHANGE OR END:. Notice in the upper right-hand corner area of the screen is field 21. PASSWORD :. If you have no password on CGLMS, the field will be blank. Enter 21 <RETURN> and type RUG into the field. (Again 3 characters are allowed.) Once the entry has been made, you will be asked to ENTER FIELD TO CHANGE OR END:. Press <F4> to end, <RETURN> or <F1> to exit the FILE NUMBER field, and <F4> at the ADD, CHANGE, etc. line to exit the dictionary maintenance.

Now that you are back at the selector, enter 797 <RETURN> and 163 <RETURN> for the file number. The screen will clear and you will be asked to INPUT PASSWORD. Type in RUG <RETURN>. **IDOL** will re-define CGLMS, print a sample screen and ask for another file number. (This re-define procedure always occurs the first time you try to access a file after it has been altered.) You may exit the 797 utility by pressing <F4>. Anyone wishing to perform file maintenance on CGLMS from this time forward will need to know the password. Operators are given three tries and reported if unsuccessful.

Removing passwords from files is conducted the same way, except three spaces are entered in field 21 in the FILE/ELEMENT DICTIONARY data entry screen (798).

OUR Memory Doesn't Cost You A FORTUNE . . .

Is Your Fortune Falling Short Of The Mark?

USA's F1MB One Megabyte Memory Board Has The Ability To Support More Users And Substantially Increase Data Throughput For The Already Capable Fortune Computer System

Compatibility

The F1MB is absolutely 100% compatible with Fortune's own 1M board.

Featuring 36 top quality 256K (150ns) socketed DRAMs, the F1MB plugs directly into the Fortune ECC slot. It may be used in conjunction with up to four of the Fortune ¼M boards, totalling a powerful Two Megabytes of memory.

Fortune models SX, XP45, and late release XP30's accept the F1MB out of the box. And now, even older model XP30's and model 20's can be increased to a full 2M with simple modification of the system board.

Upgrade Service Associates is currently soliciting inquiries on patch procedures for older style system boards. Please contact us directly with your requirements.

Affordability

USA is proud to have equalled and exceeded the quality and performance of our competition, while offering substantial savings for both the dealer and end user.

Repairability

RAM socketing throughout the F1MB facilitates repair in the field - reducing maintenance costs and system down time.

Dependability

All boards are burned in for 72 hours and tested prior to shipment. Our One Year Repair Or Replace Warranty underlines our confidence in the product.

Accountability

Upgrade Service Associates is a growing company, and we've done our homework well. But we still remember what it's like to be a customer, and will be responsive to your needs and ideas.

For Further Information And Pricing Contact:

Fortune, SX, XP45, XP30, and Model 20 are Registered Trademarks of Fortune Systems Corporation.

Upgrade Service Associates

3 Marion Road, Burlington, Massachusetts 01803 •(617) 273-0388

Readers Respond...

David Kloes of UniKomp in Houston contacted us with a helpful hint for people who use the multi-volume floppy technique to backup their hard disk. We paraphrase and embellish his suggestion.

Whenever you perform a backup using the -B option of the cp command, e.g. cp -Broust /dev/fd02 770 /u, a special index file is created called **__part__**. This file contains a complete list of each file that has been backed up to the floppies and on which disk the file resides. It can be used to quickly locate a file that you want to restore by specifying which disk the file is on.

For example, this is a brief section of a **__part__** file from one of our backups:

```
1 r /u/jnl/jrp/bills/woman.fr 000001
1 r /u/usrgrp/bin/display.status 000002
1 r /u/jnl/peek/mprc 000001
1 r /u/usrgrp/vol6/nextmeeting.fr 000001
1 r /u/gbh/85budg/revnotes1 000002
1 r /u/gbh/85budg/revnotes1.dc 000001
1 r /u/usrgrp/11et.out.fr 000001
2
2 r /usr/cmd/oldclock 000037
2 r /usr/cmd/multirefmt 000013
2 1 /usr/cmd/mpformat 000000
2 1 /usr/bin/multirefmt 000000
2 1 /u/jnl/multirefmt 000000
2 r /usr/cmd/ml 000001
```

Note that the first column contains either the number 1 or 2. This number indicates the volume number that contains the file. The third column contains the full pathname of the file. As you can see, it is fairly easy to ascertain where a file is once you find it in the list.

Since you are using a computer, it is not necessary to actually read through the entire list. The **grep** command described in last month's issue can be used very effectively to find files. Let's suppose you want to find a file called **grepnote**. To find it you would simply follow these steps:

1. Insert Volume One of the backup into the floppy drive.
2. Mount the floppy by typing mount /dev/fd02 /f <CR>.
3. Obtain a directory listing of the floppy by typing ll /f <CR>. This should create a list like this:

```
# ll /f
total 78
-rw-rw-rw- 1 root  70287 Apr 23  1986 __part__
-rw-rw-rw- 1 root     25 Apr 23  1986 __vol01__
drwxrwxr-x 8 acctg   128 Apr 23  1986 u
drwxrwxr-x 7 root    112 Apr 23  1986 usr
```

5. Note the first file. The lines on either side of the word **part** are two **underline characters**. To find your file, you just type: grep grepnote /f/__part__ <CR>.

   This will find the line (or lines), that contain the filename, and you will be able to quickly see which floppy contains it. You can then use the **cp** command to restore that single file with a command like: cp -t /f/u/dave/grepnote /u/dave <CR>. This will copy the file back into Dave's library. Note that if the file is very large, it may appear on several floppies, in which case you will need to use the standard **restore** method of using the **cp** command.

6. Unmount the floppy by typing: umount /dev/fd02 <CR>.

grep is included on Fortune Systems **Development Utilities** package. A public domain version is available on our **Compressor's Delight** disk in our software library.

If you don't have **grep**, the **more** command can be used in a similar fashion. To use **more**, mount the floppy as above. Then type more /f/__part__ <CR>. The screen will fill with the first section of the file, and you will see the word **More** displayed at the bottom of the screen. If you press the space bar, the file will be displayed one screen at a time. If you hit the right slash followed by the filename you are looking for, followed by a RETURN, (/grepnote <CR>) the computer will skip to the point in the file which contains the filename and display that screen. If the file is not found, you will get the message **Pattern not found**. To quit **more**, either hit the letter **q**, or use the space bar to move all the way through the file. When you are done, unmount the floppy as in step 6 above.

John Burns of Fortune Systems Corporation left a message on our Bulletin Board regarding general pointers about UNIX shellscripts:

I just wanted to make a few comments about the command files that were in The UNIX Directory:

Command files written for the Bourne shell should not begin with a sharp (#) because csh(1), if given the chance, may misinterpret it.

From the csh(1) man page: "Since many systems use either the standard version 6 or version 7 shells whose shell scripts are not compatible with this shell, the shell executes such a 'standard' shell if the first character of a script is not a '#', i.e. if the script does not start with a comment."

If you insist on starting with the comment use a colon (:) instead of the sharp (#). The ':' is a special shell command that does nothing.

See Respond, page 17

Respond, Cont'd from page 16

Avoid the use of true(1), use the special command ':'; it produces the same result and because it is built-in it does not cause the spawning of a subshell.

String comparisons using test(1) should be rewritten using the subshell.

Editor's Note: John's suggestions are very good ones. It is important to make sure that the proper UNIX shell is invoked to process a shellscript. This is because there is more than one "shell." The usual shell is known as the **Bourne** shell, named after the man who programmed it. This is the default shell on the Fortune. Another common shell, called the **C Shell**, is available with Fortune's Development Utilities. We happen to use the C shell most of the time because of certain very helpful features (e.g. remembering the commands you've typed). The suggestion regarding rewriting comparisons using **test** to use the **case** command will result in faster shellscripts. Thanks, John, for the useful comments.

NIFE™ POWER

New Technology Uninterruptible Power Supply Protects Microcomputers and Provides 100% Power Conditioning

SAB NIFE has **eliminated** the battery charger from the typical **UPS** package. A **new technology** reduces weight, component count and yields highest efficiency which **saves** the user **money** in operating cost.

Advanced technology has eliminated the transfer switch. For protection of your valuable data. Choose **NIFEPOWER,™** the state of the art **UPS** from **SAB NIFE Inc.** For information call 1-800-556-6764. (In RI, 333-1170)

SAB NIFE

NIFEPOWER™ is a registered Trademark of SAB NIFE, Inc.

☞SOFTWARE FOR THE FORTUNE 32:16☜

MULTIPLAN LINK - can pass data from BAS/IDOL/BASIC files to Multiplan and back. For example, it could be used to send data from any of the BAS files (Chart of Accounts, Customer Master, Inventory Master, etc) **$295.00**

RECORDS PROCESSING LINK - can pass data from BAS/IDOL/BASIC files to Records Processing and back. For example, it could be used to take customer address data from the Customer Master and will automatically make the Records Processing List document for mailouts. **$295.00**

TERMINAL/PRINTER LINK - provides capability to print Multiplan, and BAS/IDOL reports on a printer connected to a workstation; view any report on the terminal before printing; convert any report to Fortune:Word format; accumulate multiple BAS/IDOL reports before printing; backup reports onto diskette; copy or rename report files; print multiple copies of the report on either the primary and/or terminal printer. Also has option to print BAS/IDOL reports on HP LASER printer ($50 extra). **$195.00**

TELEPHONE LINK - on line telephone message, telephone directory and inter-office mailbox/memo system which can also be used as a calendar reminder system. Provides control, printing and viewing for each user. **$295.00**

FINANCIAL LINK - stores BAS Income Statement and Balance Sheet data in files. Once stored it can be sent to Multiplan or instantly printed. **$195.00**

MENU LINK - provides software and documentation for adding items to the Global Menu as well as a sub-menu for your use. Includes backup scripts for user directories and BAS/IDOL/BASIC programs and data. **$95.00**

CALCULATION LINK - provides programs for amortization, depreciation, loan repayment, averaging (with graph), linear correlation and breakeven analysis. **$95.00**

KOMPACT PERSONNEL ACCOUNTING - provides for data capture of personal, wage/salary, job, education, salary/wage, appraisal, training, dental, medical and life insurance data. Includes over 80 pre-defined reports. **$495.00**

CRT LINK - provides information for using other terminals on Fortune Systems in BASIC/IDOL/BAS. **$95.00**

Other products are also available. National and International sales.

UNI-KOMP, 9803 Rocktree, Houston, Texas 77040 (713) 937-7782

BACKUP AND RESTORE UTILITY

PROTECTION MADE EASY!

- o Back up all or selected user directories
- o Restore all or selected user directories or files
- o Format diskettes
- o Review backup volumes
- o Menu-driven, easily installed
- o Complete documentation

Also copies the files necessary to preserve your system configuration--a tremendous help if your system ever needs to be cold booted!

$150.00 Backup/Restore Utility
$200.00 Utility + 2 boxes 5 1/4" 96 TPI high-quality diskettes

For more information, call CTD at (313) 540-1282.

To order, make check payable to *Computer Trade Development Corporation.* Send to:

BACKUP/RESTORE UTILITY
Computer Trade Development Corporation
Suite 1386
Birmingham, MI 48010

CTD

Company:_____
Contact:_____
Address:_____
City:_____
State:_____ **Zip:**_____

UUCP, Cont'd from page 6

office to San Francisco, I would first send it to Harvard. Harvard might send it to a system in Chicago. Chicago sends it to Denver. Denver sends it to UC Berkeley. Then Berkeley sends it to my friend. In order to make this work, you have to instruct the computer about which "path", or chain to use to get the message to where you want it to go. Many large computers that use **uucp** and USENET have a special program called **pathto** which tells you the best path.

One disadvantage of using **uucp** (rather than **kermit** for instance) is that it takes some effort to get it installed on your machine. **uucp** is available from your Fortune dealer, and when you buy it you will get a very complete manual for installing it. It's been said that Fortune's documentation is among the best for **uucp**. In our next issue, we will give you some specific advice about installing it on your Fortune, and show you how you can use it to easily exchange information.

Josh Lobel

News, Cont'd from page 12

Marketing Strategy

The marketing objective is to provide an entry-level UNIX product to the installed base of IBM PC XT/AT users. By purchasing the Fortune UNIX Coprocessor, MS-DOS users not only enhance and upgrade their present system to the UNIX environment, but also protect their initial investment in MS-DOS software and hardware. Their single-user MS-DOS system has now been transformed into a multiuser, multitasking UNIX/MS-DOS system.

With the use of a UNIX Coprocessor and additional terminals, an IBM PC XT/AT or compatible becomes a general business of office automation system unit for one to three users. Not only does the UNIX Coprocessor enhance the utility of a system, but it also saves money and delays hardware obsolescence. In addition, with a single Fortune UNIX Coprocessor, three IBM PC XT/AT's or compatibles, may be linked together using Fortune:Works.

With the Fortune UNIX Coprocessor, users of IBM PC XT/AT microcomputers and compatibles now reap many of the benefits Fortune 32:16 customers have long had. The UNIX Coprocessor also can be a value-added feature for the 32:16 product line when sold or added to the Fortune Intelligent Workstation.

Free Software

"It may be free, but what is it?!" That's a good question. We presently have 4 disks of free software. There is something very useful or entertaining for everyone. What follows is a very brief description of the 4 available disks.

Disk 1: Fortune Utilities

This valuable disk is from Fortune Systems Corp., although *it is not supported by Fortune Systems.* It includes many useful programs and fun games. No Fortune owner should be without it!

Disk 2: Fortune: Word Tutorial

This disk is also from Fortune Systems, and contains a demonstration script for Fortune:Word and the Ext. Fortune:Word modules. It's a great way to learn the power of Fortune:Word (and it might even help you find a feature that you've missed).

Disk 3: D.C. Grab Bag

Called the **D.C. Grab Bag** in honor of the Washington D.C. users' group that provided many of the original programs. We have added some as well as several that our readers have submitted. Note that this disk has a useful editor and the famous **kermit** file transfer program along with many more games.

Disk 4: Compressor's Delight

Our newest diskette has several very useful utility programs along with many games. Two of the highlights are the programs **compress** and **grep**.

If you are interested in any, or all, of these diskettes you should send **$10, per diskette**, which includes the cost of the disk and mailing. Please make your check payable to **The Cambridge Consortium, Inc.**

Fortune Users:

G.T. HAWES & CO.
An Authorized Fortune Master Dealer
Announces a Fortune Liquidation Sale where

NO REASONABLE OFFER WILL BE REFUSED!

It's true. We're ever on the look out for *good buys* and we struck a proverbial FORTUNE! Two major users wanted out and we were there. We said, "HELLO." They said, "GOODBYE." And what a *GOOD BUY* it was! You guessed it, we're overstocked and must move the product. Take advantage of our prices because they were never lower.

CALL GEORGE AND SAY HELLO, SO I CAN SAY
GOODBYE!!!

NO REASONABLE OFFER REFUSED!
516-568-0044
Equipment Schedule Available on Request

"If you don't see my name on your computer shipment you probably paid more than you should have."

G.T. Hawes & Co., Inc.

475 W. Merrick Road
Valley Stream, NY 11580
516 568-0044

Looking for Business Accounting Software that will conform to your Business?

Look to DSM

Applications

- Can be tailored to your organization's specific needs
- Economy of Standardization
- Follows Generally Accepted Accounting Principles (GAAP)
- Luxury of Adaptibility
- Data Base Incorporation

Software for:

- Not-for-Profits
- Construction
- General Business
- Property Management
- Steel Service Center
- Law Office Management

Hardware

Fortune Systems, NCR, AT & T, IBM

Call or Write
Data Systems Management, Inc.
277 Coon Rapids Blvd.
Minneapolis, MN 55433
(612) 780-5200

DSM

All software was developed using PROGRESS, a powerfull relational data base which allows you to tailor the software to your specific needs.

PROGRESS

NEW PRODUCTS

For BAS Business Applications® and Fortune: Windows®

BAS FILES DISPLAY - Brings names, addresses and phone numbers from your busi-files into a ¼ screen window **39.95**

BaSiC MINI CALC - A desk calculator on a ¼ screen window **39.95**

COMBINATION DISKETTE - Includes both files and calculator **59.95**

Memorial Day Special

Order your **COMBINATION DISKETTE** before May 31 and we'll include Free Games Package.

BaSiC SOFTWARE CORPORATION

5201 Powhatan St., Baltimore, MD 21207
301-448-9460

/u/fortune *news*
20A Prescott Street
Suite 28
Cambridge, MA 02138

/u/fortune* *news* ...

The Newsletter for Users of the Fortune 32:16 Computer June 1986/Volume 3 Number 6

The Fortune:Word Glossary

The Basics of Records Processing Part II

Dick Dow is owner of Dow Information Services which is a technical publications firm located in Rhode Island and is contributing to this publication independently.

In the last issue, we looked at the structure of Fortune:Word's records processing application. We noted that the **list document** is where the data is stored, and contains a collection of records with identical types of information broken down into fields. A **format document** structures the information contained in the **list document** in a variety of ways and a **control document** manipulates the order in which each record is selected (by sorting specified fields) or selects only certain records according to specified criteria. The **output document** is the finished product, and may be seen on the screen for editing prior to printing or archiving, may be sent directly to the printer, or may appear both on the screen and on the printer.

Now let's look at a way in which a **Fortune:Word** glossary can be used to create a **list document**. This is a membership list, and in addition to the basic name, address, telephone number, and membership number, we are going to include an expiration date in one of two categories: a one year membership and a lifetime membership. In addition, since the majority of our members will be drawn from only a few communities, and since all will be within one state, we will use the decision making abilities of the glossary to fill in some of the information. A glossary for creating the list can be a separate entity, but we have found that combining that function with the control document function (to sort and/or select records) makes it easier to find and use.

Thus we will create a **list document** (the membership data file) which we will call "*mem.ls*" and a **glossary document** (control document) which we will call "*mem.gl*". The **list document** is created from the **Fortune:Word** menu and is left empty for the time being. The glossary is also created in the normal way and the entries are as follows. Explanations are written as comment lines so you could copy this glossary exactly and it would still run. If you

See **Records's Processing**, Page 14

Featured in This Issue. . .

The Basics of Records Processing, Part II -- all you need to know to get started with **Records Processing** featured in this month's Glossary Entry. . .**Page 1**

Support, Updates, and **DIM** elucidated in **The Basic Advisor. . .Page 1**

Automate your communications with other computers using **Fortune-to-Fortune Copy (uucp)**. We review the basics you'll need to get up and running. . .**Page 6**

New products, updates, technical tips are all included in this month's **News from Fortune. . .Page 2**

The Basic Advisor

Ray Wannall is president of BaSiC Software Corporation in Baltimore and is contributing to this publication independently.

Question: We just received a notice in the mail that Fortune Systems is not going to support BAS anymore. What gives?

Answer: About a year ago, Fortune Systems decided that providing technical tips to repair "bugs" in the original **BAS** applications was more than they wanted to handle. At that time, Fortune signed an agreement with Choice Systems in Atlanta, GA. Choice agreed to fix, for a fee, all of the current **BAS** problems and supply future updates as required. However, fixing flaws is one thing, but providing modifications is another animal. Now, it seems, neither Fortune nor Choice wishes to be involved in the business of modifying **BAS**. (I can't really blame either one of them. Modifications should be left to the vast network of software firms who have been working with **BAS** over many years.)

Since you are a reader of **/u/fortune** *news*, you have nothing to worry about. If you find you need help, watch the ads and refer back to the list of **BAS** support companies we published in Volume 3, Number 2.

See **Basic Advisor**, Page 21

*/u/fortune *news* is published by The Cambridge Consortium, Inc. It is not affiliated with Fortune Systems Corporation

NIFE™ POWER

New Technology Uninterruptible Power Supply Protects Microcomputers and Provides 100% Power Conditioning

SAB NIFE has **eliminated** the battery charger from the typical **UPS** package. A **new technology** reduces weight, component count and yields highest efficiency which **saves** the user **money** in operating cost.

Advanced technology has eliminated the transfer switch. For protection of your valuable data. Choose **NIFEPOWER,™** the state of the art **UPS** from **SAB NIFE Inc.** For information call 1-800-556-6764. (In RI, 333-1170)

SAB NIFE

NIFEPOWER™ is a registered Trademark of SAB NIFE, Inc.

/u/fortune *news. . .*

Editors: Josh Lobel, Mark Palmerino
Associate Editor: Benzion Chanowitz
Technical Editor: Scott Bradner
Advertising Manager: Mark Palmerino

/u/fortune *news. . .*is published 12 times per year by The Cambridge Consortium, Inc. Neither /u/fortune *news* nor The Cambridge Consortium are affiliated with Fortune Systems Corp. Contact us regarding subscription rates. Quantity discounts are also available.

Back issues are offered at the following prices:

Vol 1, Numbers 1-3	$1/each
Vol 1, Number 4	$.50/each
Vol 1, Number 5-8	$3/each
Vol 2, Number 1-7	$3/each
Vol 3, Number 1-	$4/each

Although every attempt is made to insure that our contents are error-free and without risk, The Cambridge Consortium is not responsible for losses that occur due to correct or incorrect interpretation of /u/fortune *news.*

For more information about us, please call or write:

/u/fortune *news,* 20A Prescott Street, Suite 28, Cambridge, MA 02138. Telephone 617 876-4763.

The text for this newsletter was prepared using a Fortune 32:16 for text input. The information was then transferred to an Apple Macintosh Plus with kermit and formatted for typesetting using the Aldus Pagemaker program.

UNIX™ is a trademark of Bell Laboratories,Inc.
Fortune 32:16™, Fortune:Word™ and FOR:PRO™ are trademarks of Fortune Systems Corporation. Multiplan™ is a trademark of Microsoft. IBM-PC™ is a trademark of IBM. MS-DOS™ is a trademark of Microsoft. Macintosh Plus™ is a trademark of Apple Computer. Pagemaker™ is a trademark of Aldus Corporation. HP Laserjet™ is a trademark of Hewlett-Packard Corp.
©1986 by The Cambridge Consortium, Inc. 6/86

News from Fortune

Fortune Systems has recently sent out a number of "press releases" about their activities. What follows are excerpts from these announcements.

FortuneMS-DOSWorkstationdeletedfromFortune Systems Price List

5/19/86 -- Fortune Systems announces the elimination of the **Fortune MS-DOS™ Workstation** product line. This announcement shows Fortune's firm support of the industry standard multiuser operating system, UNIX. The decision was made in light of a high level of competition and a dramatic drop in IBM PC prices.

Fortune Systems feels this positive decision reinforces the market areas in which Fortune has decided to compete and further focuses the company's direction in the Workgroup marketplace.

Fortune Systems will continue to support and enhance hardware and software features that allow connectivity between PCs and the Fortune product line.

Fortune retains its link to the PC world with its latest announcement of the **Fortune Multiplier**. The **Fortune Multiplier** offers coexistence between MS-DOS and UNIX.

The need for a professional IBM PC/XT keyboard is solved with the **Fortune Workstation Keyboard**. The only additional equipment to be purchased is **Keyboard Handler** software and the **Fortune Workstation Keyboard Cable** to the IBM PC.

Below is a list of hardware and software Fortune Systems currently sells to support PC connectivity.

Part Number	
1003336-01	Fortune Multiplier w/Keyboard
1003336-02	Fortune Multiplier (Without Keyboard)
1006351-01	Fortune:Works (DOS workstation)
1006462-01	Fortune:Works (Graphics Update)
1003457-01	Fortune Workstation Keyboard Cable to IBM PC
1003366-01	Fortune Workstation DB-9 Video Cable
1003723-01	Fortune Workstation Keyboard
1006302-02	Fortune Keyboard Handlers

Continued on next page

Fortune News, Continued from previous page

Burrem(BurroughsTerminalEmulator)/BurremFT (Burrem File Transfer) RELEASE 1.0

DESCRIPTION:

5/20/86 Custom Products Division of Fortune Systems Corporation is pleased to announce the availability of **Burrem**, a Burroughs terminal emulator. **Burrem** emulates a Burroughs TD830 terminal and communicates with the host via the **Fortune Intelligent Communications Controller (Comm B)** using an RS-232 protocol.

We are also announcing **BurremFT,** a companion product which allows asynchronous file transfer between the Fortune and Burroughs environments. **Burrem** and **BurremFT** provide Fortune users the ability to communicate with a Burroughs mainframe while still allowing access to **FOR:PRO** and Fortune applications.

PERFORMANCE AND LIMITATIONS:

Both **Burrem** and **BurremFT** support up to four concurrent users per Fortune System, using asynchronous protocol for communication at a speed of 9600 baud using the RS-232 interface. **Burrem** and **BurremFT** can be used with small, medium and large Burroughs units. If the host communication protocol is Burroughs TDI, a TDI/RS-232 protocol convertor will be required.

BurremFT can be custom modified to accommodate any asynchronous file transfer at 9600 baud.

PREREQUISITES:

Hardware Requirements:

The **Burrem** and the **BurremFT** require the Fortune 32:16, with a minimum 512K RAM, an **Intelligent Communications Co-processor Board (Comm B)**, part # 2001652-02, an RS-232 Convertor cable (connects Comm B to TDI convertor), part # 1000633-10 or equivalent, and a Burroughs TDI/RS-232 Interface Convertor.

Software Requirements:

The Burrem and the **BurremFT** require the **FOR:PRO Operating System** Version 1.7.4 or later, the "DC-SPOOLER" on the Burroughs host when using **BurremFT**. The file transfer product is designed to interface with "DC-SPOOLER" on the host. For information contact CORE Technology at:

CORE Technology, 7201 West Saginaw, Suite 115, Lansing, MI 48917, **(517) 321-4659**

AVAILABILITY: July 30, 1986

PRODUCT: **Extended Laser Interface,** Release 1.1 June 16, 1986

DESCRIPTION:

The Custom Products Division of Fortune Systems is pleased to announce version 1.1 of the **Extended Laser Interface**, previously known as the **LaserJet™ Extended Interface.** The **Extended Laser Interface** allows the Fortune system to use laser printers, the latest technology available in desktop publishing.

The **Extended Laser Interface** allows the Fortune system to interface with the Hewlett-Packard™ LaserJet Printer as well as equivalent model laser printers made by Xerox™ and Canon™ When using the HP LaserJet printer, the Interface allows the Fortune system to utilize all of the current HP LaserJet printer capabilities including proportional spacing and the abililty to change fonts within a **Fortune:Word**TM document.

The **Extended Laser Interface** fully supports changing pitch from the **Fortune:Word** print menu when using a fixed pitch font such as the 10 pitch Courier, supplied by HP as a default. This gives the HP LaserJet user 10,12, and 15 pitch printing capability from the default or other fixed pitch fonts.

The **Extended Laser Interface** supports the HP 92286B Proportional Spacing font and the HP 92286D 12 pitch Prestige Elite font cartridges. The fonts available on the LaserJet printer cartridge 92286B are:

Bold proportional type.

Large proportional type.

Small proportional type.

Italic proportional type.

These fonts and characters are selectable from within a **Fortune:Word** document. For support of other font cartridges or soft fonts you would like designed, contact the Custom Products Division of Fortune Systems.

The **Extended Laser Interface** also supports a true 8-bit data path, providing the capability necessary to use laser printers such as the Hewlett-Packard, Xerox and Canon.

Paragraphs that contain multiple fonts cannot be justified Overstrike or double underscore cannot be used along with another font; for example, you cannot double underscore text that is in small

See Fortune News, page 4

Fortune News, Continued from page 3

proportional type when another font(i.e., 10 pitch Courier font) is the primary font.

Although users connected via Fortune:Link can access the laser printer, the system physically connected to the laser printer cannot access other printers via Fortune:Link.

Updates to the original LaserJet Interface are available.

PRODUCT: X3.64 Terminal Filter, Release 1.1, June 16, 1986

DESCRIPTION: Custom Products Division of Fortune Systems Corporation announces the availability of the **X3.64 Terminal Filter**. When installed on a Fortune system, this software package allows the user to connect a GVT or Televideo 970 terminal to the system, and then access **Fortune:Word** and **Multiplan** from the terminal. Although the package is written for the GVT and Televideo 970, other terminals conforming to the ANSI X3.64 standard can be used. Terminal filters can be custom written by the Custom Products Division for terminals that are not ANSI X3.64 standard.

Tech Tips from Fortune

Fortune Systems periodically publishes technical tips and software bulletins to notify users of know problems and possible solutions. We are including several recent ones here.

Product: FOR:PRO 1.8.1.1

Problem Description:

System will auto re-boot if not powered off within 5 seconds after the message "Please Turn the Fortune Systems 32:16 off" is displayed.

Corrective Action:

If system is at the date and time screen after a shutdown, the system has re-booted. A complete shutdown should be run again before the machine is powered down.

Turn off system immediately after the message "Hardware shutdown complete" is displayed.

Product: Extended HP Printer Interface

Problem Description:

After installing HP Laserjet software, laser printer will not print legal sized documents (8-1/2 x 14").

Corrective Action:

Entries in the /etc/printcap and /usr/lib/wheels/hp.whl files need to be edited to enable printing on legal size (8-1/2 x 14 paper).

1. Line 3 of the \U\printcap\u\ file needs to be edited. The line in the printcap file currently reads as follows:

```
:is=\E&dl2e7.6c66F\E&sOC:\
```

Insert \EE after the = sign so that the line reads as follows:

```
:is=\EE\E&l2e7.6c66F\E&sOC:\
```

2. Line 12 of the /usr/lib/wheels/hp.whl file also needs to be edited. The line currently reads as follows:

```
+init
"\E&l0o136pl34F\E(8U\E(s0p10h10v0s0b5t"
```

Replace the line with the following:

```
"\E&l0o136pl34F\E(8U\E(s0p10h10v0s0b5t"
```

Replace the line with the following:

```
+init
"\E&l84p0O\E(8U\E(s0p10h12v0s0b3T"
```

Product: Extended HP Printer Interface

Problem Description:

After installing HP Laserjet software, the laser printer will not print. The printer control menu shows the job printing but there is no printout.

Corrective Action:

First, eliminate a hardware malfunction as a possible cause of the problem. Be sure that the cabling and system configuration for the printer are correct and run the printer self test. If hardware functions, follow the procedure below to check the printcap file and make sure that there is only ONE entry for the HP Laserjet.

See Fortune News, page 12

Our Memory Doesn't Cost You A Fortune. . .

Is Your Fortune Living Up to its Potential?

Would you like to:

Speed up your computer?
Add more users?
Use the latest software, like Fortune:Windows™?

Compatibility:
The F1MB is absolutely 100% compatible with Fortune's own 1 Meg memory board.

Dependability:
All boards are burned in for 72 horus and tested for 48 prior to shipment. *Our One Year Repair of Replace Warranty underlines our confidence in the product.*

Affordability:
USA is proud to have equalled and exceeded the quality and performance of our competition, while offering substantial savings for both the dealer and end user.

Repairability:
The F1MB fascilitates repair in the field as it is 100% RAM socketed. Now you can repair a board by simply replacing a defective chip rather than face an expensive board exchange -- reducing maintenance costs and system down time.

If you have a Fortune SX, XP45 or recent model XP30, you can use our board by simply plugging it in. Older systems can be upgraded to use our board with a simple and economical modification.

For Details Call:

Upgrade Service Associates
3 Marion Road, Burlington, MA 01803
Phone: 617 239-8114

Our Guarantee: If you're not completely satisfied within 30 days, *USA* will gladly refund your money.

Fortune:Windows™ is a trademark of Fortune Systems Corporation

UUCP Fundamentals -- Part II

How to Get Started

In our last issue, we extolled some of the virtues of the **Fortune-to-Fortune Copy** program, which is generically known as **uucp**, or Unix-to-Unix Copy. **Fortune-to-Fortune Copy** is used to transfer both ASCII and binary files between UNIX systems automatically. As we noted, this program is very powerful, but somewhat more complicated to install than most Fortune programs. There are probably many users out there who have a copy of **Fortune-to-Fortune Copy** sitting on their shelf but have never bothered to install it. What we will try to do in this article is get you to the point where **uucp** will work on your Fortune. Please note that Fortune Systems has published a new guide for **Fortune-to-Fortune Copy** which explains in great detail the instructions for its use.

Before we delve into the details, the terms **guest** and **host** should be explained breifly. A **host** computer lets other computers login to it just as if they were users on terminals. A **guest** computer pretends that it is simply a terminal on some other computer. For the duration of this article, we will assume that you will be using your computer to login on a remote computer; hence the port will be attached to a **HOST** computer. Since another computer will be the **host**, you will be logging in to that system.

Installing Fortune-to-Fortune Copy

The first step in installation is the same as that for installing most Fortune Software, choosing the **S5 Product Maintenance** selection from the **Global Menu**. Install the **Fortune-to-Fortune Copy** program as usual.

Next, you must define one of your ports to connect up to a host. If you are already using a terminal emulator program such as ITE Handshake, or kermit you already have a port that has been appropriately defined. You can use that same port to call out using **Fortune-to-Fortune Copy**. If not, you will need to follow this next set of instructions.

Defining a Port

You may use either the SIO connector -- the horizontal port on the center back of the computer -- or any of the ports on a Comm-A board. Choose a free port that you will use to call out of. Your computer can be linked to the other computer either by a direct wire, or through a modem.

Continued on next page

UNISOFT INC.

Experts in Computer Consulting - Sales - Service

Presents

FORTUNE SYSTEMS HELP

We Provide:

- Expert trouble-shooting
- Years of programming and accounting experience
- *Free Initial Consultation*
- Training Available for:
  * BAS/IDOL/BAS and Multiplan
  * UNIX and Ext. Fortune:Word
- Hardware Repairs & Upgrades
- FORTUNE trained experts at a cost that will surprise you!
- *Trade-Ins Welcome*
- Specializing in:
  * Hard Disk Trade-In
  * Hard Disk Upgrades

Authorized Fortune Systems Dealer
Authorized Fortune Systems Repair Depot

CALL: (303) 671-8600 EMERGENCY HELP: (303) 691-HELP

UNISOFT INC. 14190 E. Jewell Avenue Suite #7 Aurora, Colorado 80012

Fortune-to-Fortune, Continued from previous page

Use the **S2 39 Define Device Connections** option to define the port You might want to first ask for the Table of your existing devices Note how the port you are defining is currently set -- giving particular attention to whether it is currently defined as a terminal. Also, note if any of the ports are currently labeled Comm, which is the correct designation for a Host port. If there are Comm ports, count how many there are. Next, choose the port number you will use and the next screen will be a new menu. Choose option 1 -- Define Category of Device Connected to Port. You are then asked if the port will be used to connect a Printer, a Terminal, or a Host. Choose Host. You will then be asked to enter a logical computer number. If this is the first port you have defined as host, enter 1. If it's the second, enter 2, etc.

You will then be asked to enter the type of host computer that will be attached to this port. Enter the appropriate choice. Don't be afraid to enter Unknown if you don't know what kind of computer the host is, of if you will be connecting to several different kinds of computers.

Finally, you will need to enter the default baud rate for the port.

If your port had been configured as a terminal, you will want to shutdown the computer, or kill the login process. If it wasn't configured as a terminal, there is no need to shut the computer down at this point.

That's the end of the first step. If you have a terminal emulator program like kermit or ITE, it would be a good idea to try connecting up to your new port and make sure that it works.

Configuring the Fortune-to-Fortune Copy files

In order to work properly, **Fortune-to-Fortune Copy** must have several files in the /usr/lib/uucp installed and correctly configured, as well as a file called /etc/sitename. The following is a list of these files:

Directory	File	Function
/etc	**sitename**	Your system's name
/usr/lib/uucp	**L.sys**	Lists each remote system you will be calling along with the appropriate information for making the call.
	L-devices	Lists the ports on your system that will be used to make the calls.
	USERFILE	Establishes guidelines for access to files on your system.

In addition to these files, you may have a file called **L.cmdfile** which will list the commands that can be remotely executed on your system, and a file called **Logins** that will give more specific information about who may log into your system. There is detailed information about these files in the **Fortune-to-Fortune Copy** manual distributed by Fortune Systems.

In order to best understand how these files work, we'll provide a brief description of how **Fortune-to-Fortune Copy** works when you send a file from one system to another. For the sake of our discussion, let's suppose that your computer is called **howard**. You want to send a file to a computer with the sitename **bingo**. You give the **Fortune-to-Fortune Copy** command on your computer. That starts up the **uucp** process. After a moment your screen or prompt comes back. That's when the fun begins. Now your computer, **howard**, checks the **L.sys** file for the calling instructions for **bingo**, including a phone number. Then it checks the L-devices file to check and see which port and what baud rate to use to make the call. **howard** follows those instructions and calls **bingo**. **bingo** answers the call and asks for a login name. Typically **howard** will login as **uucp**. If needed, a password can be provided. The **bingo** machine will check the **USERFILE** to be sure that **howard** has the permission to copy the file, and then a process called uucico will be started and suddenly **howard** and **bingo** start communicating. If all has gone well, the file will be copied. The system is basically straightforward, although putting all of the right information into the configuring files can be tricky.

The article will confine itself to basic setup instructions. More elaborate information can be found in Fortune's **Fortune-to-Fortune Copy** guide.

Here is a file by file description of what you need to do. You will need to edit these files with an editor like **ed**, **vi**, or **screen**.

`/etc/sitename`

This file is easy. It contains the name of your system. In our sample case, it would contain the word **howard**. If you want to maintain compatibility across the UNIX world, keep your name 6 characters or less. If you choose a name longer than 7 characters the excess will be ignored by **Fortune-to-Fortune Copy**.

See **Fortune-to-Fortune**, page 9

Dear Fortune Owner:

As you have probably read in */u/fortune news,* Fortune Systems SX models are significantly more powerful and faster than the PS and XP models. If you have a PS or XP model system, you should take advantage of our special promotion on SX upgrades. For a limited time, and while our supplies last, we will upgrade your system, including the new SX Cabinet at no additional charge for the cabinet. With this special offer, you may choose any combination of the following upgrades and/or expansions:

- *SX Processor with Current Release of Multi-User For:Pro*
- *60 MB Internal Streaming Tape*
- *30 MB, 45 MB, 70MB High Speed Hard Disk*
- *6-Port Intelligent Asynchronous Controller*
- *Upgrade to Current Release of all Fortune Software*
- *Fortune:Windows (Included at NO Charge)*

Additionally, we can perform these upgrades within 24 hours of receiving your system. In fact, if you ship your system overnight on a Friday for Saturday delivery, we will have it back to you on Monday morning, all ready to plug in and run; not losing the use of your system for a single work day. And, we will issue you a full 90-day warranty on the entire system.

JFM provides services to Fortune Systems owners nationwide either direct or through our authorized Affiliated Dealers. If you haven't used our services, I invite you to discover JFM; you will be glad you did.

Sincerely,

Morris Frydman
President

Talk to us, then talk to our clients

The Merchandise Mart
Lobby Suite 144
Chicago, Illinois 60654
Call (312) 644-0870

Authorized Fortune Systems Master Dealer

Fortune-to-Fortune, Continued from page 7

/usr/lib/uucp/L.sys

This file contains all of the information that your system needs to call another system -- in our case, **bingo**. The following line is a sample **L.sys** entry for **bingo**:

```
bingo Wk0800-2359 tty04 1200 tty04 \
 "" AT\015\c OK \
 ATdt16176481263\015\c \
 CONNECT-A/\c-CONNECT-A/\c-CONNECT-A/\c-CONNECT \
 PAUSE 0 ETURN>:-\015\c-ETURN>:-\015\c-ETURN>: uucp
```

First of all, this entire entry could all be typed on a single line. It has been split in parts here for readability. If your entry continues on more than one line, add a backslash to the end of the line to indicate that it the entry is continued on the next line.

This is what each of the fields in the above entry mean:

System name: (`bingo`) sitename of the system you are calling.

Time: (`Wk0800-2359`) The allowable times for your system to call the host system. This allows you to take advantage of discount long distance times or restricting calls to normal work hours. If you don't want any restrictions, replace Wk0800-2359 with Any. If you never want to call, replace it with Never. The system is very flexible with these times.

Call device: (`tty04`) This is the port that will be used to call out of. This was discussed above. It can be either a tty device, e.g. `tty03`, or a cul device, e.g. `cul3`. It can't be a port that is configured as an "enabled" or login port.

Speed: (`1200`) The baud rate that the computer should use to call out. For a modem, this is probably 1200. If you are connected by a direct wire, it can be up to 9600. Note that this must be the same as the baud rate specified on the login port of the host computer.

Phone: (`tty04`) This is the same as the **Call device**.

Login: Now things get interesting. This is the sequence of strings that will get sent back and forth in order for you to login to the remote system. It is a "call and response" system where you wait for something from the remote system, then you send something. Each time your computer receives something back, it checks to be sure that it's what was expected before sending the next piece of information.

For instance, when you sit down at your Fortune, it says "`Please type in your name and press <RETURN>:`". When you see the word <RETURN>: you type in your login name. The system then may ask for a password, or may just log you in.

Let's look closely at the login sequence in our **L.sys** file to see how the call and response has been set up.

`"" AT\015\c OK` This waits for the system you are calling to send "", or the null string. This tells your system to be the first one to make a real move, since it expects nothing from the other system. When your system gets nothing (which should be all of the time), it sends out the sequence `AT\015\c`. If you have a Hayes compatible modem, you will know that AT stands for Attention. The `\015` is the octal (that's base 8) equivalent of the `RETURN` key. The \c instructs the computer to send out that string without an additional line feed (which would probably cause the modem to abort, as it does when you hit a RETURN while it is dialing). Finally, your computer looks for the string "OK". This is sent back by the modem in response to the AT command.

`ATdt16176481263\015\c` This string gives your modem the normal ATdt dial command. 16176481263 is the phone number for our bulletin board computer. You would substitute the correct number for the system you wish to call.

`CONNECT-A/\c-CONNECT-A/\c-CONNECT-A/\c-CONNECT`
This is a combined call and response that is issued after the modem types the phone number. Your system is waiting for the word CONNECT to come from your modem. (If your modem says CONNECT 1200, then you might want to change the word CONNECT in this line to 1200 -- e.g. `1200-A/\c`.) If it doesn't get the word CONNECT, it sends an A/, which tells the modem to redial the last number. (These are all Hayes commands Normally the calls and responses are separated by spaces. This is a special case where they are separated by hyphens. If the word CONNECT does come back right away, the program will skip ahead to the next link in the sequence, which in this case is the `PAUSE0`. If it doesn't come back, **uucp** will continue to try the same sequence as many times as you include it with the hyphens. In this example, it will retry up to four times. This allows for busy numbers, or dialing errors, or whatever.

`PAUSE0 ETURN>:-\015\c-ETURN>:-\015c-ETURN>: uucp`

Once your system gets CONNECT, it pauses for 0 seconds then waits for the word ETURN>: from the

See Fortune-to-Fortune, page 10

Fortune-to-Fortune, Continued from page 9

other computer. (Note that the **>:** are both included, because that is exactly how the login screen appears on your computer. You have to be very precise with these strings.) If it doesn't get it, a RETURN (`\015`)is sent out to stimulate the login prompt. Note that only the end of the word ETURN>: is sent. Generally only the last seven characters of the desired string are entered. It tries three times to find the ETURN>: string. (One pitfall with the Fortune login is that the same string ETURN>: is issued when it is asking for your name as well as your password. **Fortune-to-Fortune Copy** sometimes gets confused thinking that it has gotten the name login ETURN>: when in fact the other computer is waiting for a password. If you follow these instructions, it shouldn't be a problem, but be careful.) Once it gets back the ETURN>: string, it sends out **uucp**, which should be the account name the other computer is expecting as a login.

If you are using a Hayes compatible modem and logging into a Fortune computer, you should be able to use an **L.sys** entry very similar to the example given above. You may need to change the sitename and the call out port, but the login sequence, which is the most difficult part, should work fine.

/usr/lib/uucp/L-devices

This file contains the names and baud rates of the ports you are using to call out. A typical entry would be the following:

```
tty01 tty01 0 1200
cul4  cul4  0 1200
```

The entries in this file contain the **type, line, call-unit,** and **speed.** The **type** and **line** are always the same. They specify the port you are using. As you can see, they can either be **tty** devices or **cul** devices. The **call-unit** entry in the third column is always 0 on the Fortune. The final column contains the speed you are calling out at, and can be anything from 300 to 9600 baud. If you are calling out on a modem, you are limited to the fastest speed that the modem can handle, probably 1200 or 2400 baud. If you have a direct wire hookup, you can go up to 9600 baud. The speed(s) listed in the **L.sys file** should be listed in this file also.

/usr/lib/uucp/USERFILE

The **USERFILE** controls the access that users will have on your system. It specifies which directories or files can be copied into or out of your computer. Fortune Systems says that this is the most complicated file to use, and in a sense it is, because it is the file that maintains your security. When you install **Fortune-to-Fortune Copy**, a default **USERFILE** is placed in the directory. You should be able to make **Fortune-to-Fortune Copy** work just with the entries contained in that file. We will give a brief explanation of this default file. If you have special security needs, you will want to modify it.

Default USERFILE:

```
uucp, /tmp /usr/spool/uucppublic
UNKNOWN, /tmp /usr/spool/uucppublic
root, /
, /
```

A complete entry in this file might consist of the following information:

```
user, system, c, path(s)
uucp, bingo, c, /tmp /usr/spool/uucppublic /usr/games
```

For a moment we will discuss this actual entry rather than the default file. Note that in the line above, all of these fields are filled in. The user **uucp** (**uucp** is a login account on the host system) may login from the system called **bingo**. The letter **c** in the entry indicates that when the system **bingo** calls in, your system should hang up and call **bingo** back. This is a security device to ensure that the system calling in as actually **bingo**. In most cases, you will probably just leave the letter **c** out. When **uucp bingo** is logged in they can copy from or to the directories `/tmp`, `/usr/spool/uucppublic`, and `/usr/games`. This prevents remote users from copying things from/to your entire disk.

Now take a look at the default file. Note that the first line of the default file above lists **uucp** as the user, but doesn't have a sitename following the comma. An entry like this is the default entry for any **uucp** that logs in without a recognized sitename. Note that these users are limited to `/tmp` and `/usr/spool/uucppublic`. Again, this limitation is a security measure. The second line which lists UNKNOWN as a user without any sitename is also a default for any user with a name other than **uucp** who starts up the **uucp** programs. The final line is the default listing for local users of the system. It's the default because it doesn't list any user name. It doesn't contain a sitename either because that is irrelevant for local users. This entry gives local users access to copy to and from the entire file system. Note that the UNIX file permissions still govern which files can be read and written by individual users.

Testing your work

Those are the basics for setting up your system. We will publish more details about customizing **Fortune-**

Continued on next page

Fortune-to-Fortune, Continued from previoius page

to-Fortune Copy to meet your needs if there is interest. In the meantime, we suggest that if you already own **Fortune-to-Fortune Copy** you try installing it and using it. You can try it with our bulletin board if you like. Put this entry in your L.sys file:

```
ufortune any tty04 1200 tty04 "" AT\015\c OK \
 ATdt16176481263\015\c \
 CONNECT-A/\c-CONNECT-A/\c-CONNECT-A/\c-CONNECT \
 PAUSE 0 ETURN>:-\015\c-ETURN>:-\015\c-ETURN>: uucp
```

Substitute the correct port number for `tty04` if your modem is attached to a different port. Then use the **Communications Selection** from the **Global Menu** to copy the file `/tmp/uucptest` from the **ufortune** system to `/tmp/uucptest` on your system. (You can do this from the shell by typing:

```
uucp ufortune!/tmp/uucptest /tmp/uucptest <CR>
```

The format for the **uucp** command is very similar to the **cp** command -- `uucp source destination`. The only difrference is that with **uucp** you may specify both the sitename and the filename for either the source or the destination. An exclamation mark separates the sitename from the filename, as in **`ufortune!/tmp/uucptest`**.)
After the lights stop flashing, you should check the `/tmp` directory and see if the file has arrived. Type: `ll /tmp/uucptest <CR>`. If the file is there, type `more /tmp/uucptest <CR>`.) If the file didn't arrive, check a file called `/usr/spool/uucp/LOGFILE` for error messages. You can see the end of this file by simply **catting** it to your screen (See the article elsewhere in this issue for notes on **cat**.)

If all else fails, you can run **uucp** in the debug mode by typing:

```
/usr/lib/uucico -x4 -r1 -sufortune  <CR>
```

The **-s** flag refers to sitename -- substitute a different sitename if you are not calling **ufortune**. Once you hit return, **uucp** will spring into action. You should see the call and response routine along with any error messages. Once it begins, it takes several minutes for it to complete itself and come back to a prompt. Be patient. If it exits immediately, try removing the any files in the `/usr/spool/uucp` directory that begin `ST*`. You may also need to remove files in the same directory that begin `LC*`. (These files are lock files for the port you have designated. They may be left by the aborted **uucico** command. Do not remove

See Fortune-to-Fortune, page 12

Looking for Business Accounting Software that will conform to your Business?

Look to DSM

Applications
- Can be tailored to your organization's specific needs
- Economy of Standardization
- Follows Generally Accepted Accounting Principles (GAAP)
- Luxury of Adaptibility
- Data Base Incorporation

Software for:
- Not-for-Profits
- Construction
- General Business
- Property Management
- Steel Service Center
- Law Office Management

Hardware
Fortune Systems, NCR, AT & T, IBM

Call or Write
Data Systems Management, Inc.
277 Coon Rapids Blvd.
Minneapolis, MN 55433
(612) 780-5200

All software was developed using PROGRESS, a powerfull relational data base which allows you to tailor the software to your specific needs.

Dealer Inquiries Invited
PROGRESS is a trade mark of Data Language Corporation.

EQUIPMENT *SERVICE* SUPPLIES
ACCESSORIES TRAINING *UPDATES*

PIONEER APPLICATIONS
49 WALNUT STREET
WELLESLEY, MA 02181

Pioneer Applications can supply *all* of your needs. As the FORTUNE DEALER with the largest installed base in New England we can handle any situation requiring Fortune expertise.

SERVICING FORTUNE USERS FOR OVER 3 YEARS

If it's new equipment, service, support, software, consultation or just a friendly voice, Pioneer will respond immediately. **Spring special**: 70MB hard drives, *call now.*

CONTACT:
ALLISON BLACK at (617) 237-5478 or
VICTORIA COLE at (203) 236-2373

Fortune-to-Fortune, Continued from page 11

them if some other program is currently using the port for something else, e.g. **ITE**.)

You may find that you need to alter the **L.sys** file as a result of the debug session. Simply go in with screen and change the file. It will likely take several attempts to get everything working just right, but it will be worth it.

Josh Lobel

Fortune News, Contined from page 4

Procedure:

1. Log onto the system as root.

2. `ed /etc <RETURN>`

3. `ed printcap <RETURN>`
   (System will respond with a number representing how many characters are in your file.)

4. `g/Hewlett/gp <RETURN>`
   (Displays how many entries in printcap file.)

5. If more than one line displays in step 4 above, remove the printcap file and copy the printcap file off of the **Single User For:Pro Volume 1** diskette. Then re-install the **Extended HP Laserjet** software.

6. If only one line is displayed in step 4 above, contact your Fortune representative for assistance.

Product: FOR:PRO Operating System

Problem Description:

If you are using the multivolume copy (`cp -B`) option and you cancel out of the process, or if the process aborts, refer to pages 10-11 of the FOR:PRO 1.8 release letter or follow the procedure below to restart the copy.

Corrective Action:

Insert Volume 1 of the backup and read the `_part_` (i.e., two underscores are before and after "part" in the filename) file. This file serves as a table of contents for the entire backup The leftmost number displayed is the backup volume number. The next field displayed contains one of the following characters:

d = directory
r = ordinary file
b = block device
c = character device
* = beginning of a split file
- = continuation of a split file

To identify where to restart the copy, select a volume that does not begin with a dash (-). A dash indicates the continuation of a split file between volumes. For example, if you aborted the backup on volume 5, but the second character displayed for volume 5 is a dash, you would begin the backup with volume 4 (Refer to pages 10 and 11 of the FOR:PRO 1.8 release letter for an example).

Procedure:

1. Insert Volume 1 of the backup into the floppy drive.

2. `mount /dev/fd02 /f <RETURN>`

3. `more /f/_part_ <RETURN>`

4. To restart the copy, enter:

   `cp -BrostXC /dev/fd02 790 vol# / <RETURN>`

   where vol# is the number of the volume for the restart.
   (The system responds with "Please insert volume 1. Press <RETURN> when ready or <CANCEL> to quit").

5. Insert Volume 1. After you insert Volume 1, your backup will resume with a prompt to insert the volume you requested to restart the backup.

Product: SMC Business Basic

Problem Description:

Cannot print from Business Basic onto a local area network (LAN printer).

Corrective Action:

The current version of BASIC does not use lpr to print documents. In order to print from BASIC, a printer must be on the same system. A future release of BASIC will use lpr to print.

Product: FOR:PRO Operating System

Problem Description:

When fsck is executing, if it cannot obtain enough memory to keep its tables, the system will prompt the user for a scratch filename.

See Fortune News, page 19

YOUR FORTUNE IS MULTI-TASKING——MAKE IT WORK FOR YOU
SAVE TIME AND MONEY SENDING TELEX MESSAGES WITH

HANDSHAKE™

COLT - COmputer Linked Telex

Use Your Fortune to Access all Western Union EasyLink Services:

Telex • Telegram • Cablegram • Mailgram

- **Automatic Send/Receive Functions**
- **User Selects Text Editor**
  - *** Fortune:Word or**
  - *** COLT's editor TED**
- **Time/Date Stamp on all Messages**
- **On-Line Database**
  - *** Store Prototype Messages**
  - *** Store Telex Directory Information**

Eliminate your out-dated Telex equipment

Eliminate your expensive dedicated lines

Eliminate low speed, high cost transmission

GET HANDSHAKE COLT, COmputer Linked Telex, from:

SST DATA, INC.

4701 W. Schroeder Dr., #100
Milwaukee, WI 53223
Telephone (414) 355-6990
Telex: 590597 (SST DATA INC)
EasyLink 62792652

Ask about other HANDSHAKE Data Communication Products for the Fortune 32:16 including: HANDSHAKE I.T.E. link to the non-UNIX world, Interactive Terminal Emulator; HANDSHAKE II link to the IBM mainframe, BSC or SNA/SDLC, 327x emulator; HANDSHAKE III link to IBM Systems 34/36/38, 5251 emulator; HANDSHAKE DG link to Data General, 410/460 emulator; HANDSHAKE FP, Prime interface; HANDSHAKE V, VT100 Terminal Emulator.

Records Processing, Cont'd from page 1

understand what's happening, simply delete the comments to save space. I'm indebted to Al Polson of Barrington, RI, a fellow member of the Boston Area Fortune Users' Group, who worked with me on developing this glossary.

/*MEMBER LIST GLOSSARY
by Dick Dow & Al Polson*/

```
entry a {     /*This is the entry which begins
```
building a record. It prints the first field label surrounded by merge symbols and then waits until the actual information is entered and <RETURN> is pressed before putting in a return symbol and printing the next field label. Note that the field labels are abbreviated to save space, but that they could be used in more readable form, e.g. "*number*" instead of "*num*". "**Call keysin**" is the command that returns control to the keyboard so that you can enter the information.*/

```
merge  "num"  MERGE
call  keysin  return
```

/*The following two field labels are for the last name and the first name (abbreviated "*lnm*" and "*fnm*"). Using separate fields allows for alphabetical sorting by last name while still using the name in the normal way on documents. If you wanted to do this and have only one "*name*" field, you would have to always write your name with the last name first. This is the kind of decision you have to make when planning a database.*/

```
merge "lnm" MERGE
call keysin return
merge "fnm" MERGE
call keysin return
```

/*The next two fields provide for address lines. This is useful if your member has mail sent to a company address or if you need to include apartment or rural route information that won't fit on one line. If a particular entry needs only one line, fill in the second address field with a "**". Then you can send your output document to the screen and, using search and replace, delete all instances of "**" followed by a return. This means there will be no blank lines in your address.*/

```
merge "ad1" MERGE
call keysin return
merge "ad2" MERGE
call keysin return
```

/*If your membership is primarily drawn from certain communities, you might want to use a subroutine like the following which will considerably reduce the amount of typing you have to do. At this point, our glossary calls another entry (`entry *`) which deals with the city, state, and zip code. The glossary then returns here to continue with the membership information. We will explain the rest of the glossary and then look at entry * separately.*/

```
  glossary "*"
```

/*This next field label allows entry of the member's telephone number.*/

```
merge "tel" MERGE
call keysin return
```

/*Two types of membership are available in this mythical club: annual and lifetime. The glossary uses a simple "**if test**" to find out which type of membership you are assigning. It does this by taking a variable called **exp** (for expiration) and setting it to zero, then stating that it will be equal to whatever key you hit. It looks at that keystroke and if it is an "1" the glossary automatically fills in "**LIFE**" after the expiration field label. If you type any other key, the glossary allows you to fill in the expiration month and day (06/27 for example) and then puts in a "/87" for you. It should be possible to write a routine which takes the current system date and adds a year to it if expiration is always a year from when you are actually entering the information. We don't always make entries so promptly, however, and this seems to work just as well.*/

```
merge "exp" MERGE
exp = 0
exp = key
if(exp == "1")
     {"LIFE" return}
else
     {call keysin "/87" return}
```

/*This completes the information for this record and the glossary puts in open/close merge symbol (**< >**) to signify the end of the record. It then asks you (with a message in the "*prompt*" area, the upper right corner of the screen, if you want to stop here. If you type "y" or "Y", the glossary stops running. To start again you must press the **<LF/GL>** key and "a" for entry a. On the other hand, if you press any other key than "y" or "Y" the glossary will begin entry a over again and you can enter another record. Note also another "if test" concerning which line you are on. If you have elected to keep the glossary running in order to enter another record, the glossary will check to see which line you are on. If you are below line 56, it will automatically enter a page break before calling entry a again.*/

Continued next page

Records Processing, continued from page 14

```
merge MERGE return
call prompt("stop? type y")
answer = key
if(answer == "y" | answer == "Y")
     {call prompt("") exit}
call prompt("")
if(line > 56)
     {insert page execute}
call a }

/*ENTRY *: AUTOMATIC ENTRY OF CITY,
STATE, AND ZIP by Dick Dow*/
```

/*This entry is called by *entry a* at the point that the city, state, and zip need to be entered in the record. It prints the field label for each, but rather than calling for typed input, it asks for a "city code" -- a one letter code which will cause the glossary to automatically enter the city ("cty") field label and name and then go on to fill in the rest of the labels and information. You can have as many codes as there are characters on the keyboard (and you can remember), but here are the ways we handle different situations.*/

```
entry * {
city = 0
merge "cty" MERGE
call prompt ("type city code")
city = key
```

/*In this first instance, our code ("h") is for a city with a single zip code. Typing "h" causes the glossary to fill in the state and the zip and to return to *entry a* for the rest of the record's information.*/

```
if(city == "h")
{
     "Hope" return
     merge "sta" MERGE "RI" return
        merge "zip" MERGE "02831"
return
 }
```

/*In the next case, the code ("0") refers to one zip code area of a city which has several zip codes. It works just as the example above, filling in the state and the full zip code and returning to entry a.*/

```
if(city == "0")
{
     "Cranston" return
     merge "sta" MERGE "RI" return
        merge "zip" MERGE "02910"
return
}
```

See Records Processing, Page 17

VANTINE COMPUTER SERVICES

We Are Not The

Biggest Biggest Biggest

Nor The

Cheapest Cheapest Cheapest

We Simply

Care Care Care

VCS is an Authorized Fortune Systems Dealer. We understand and support the products we sell.

We provide sales, upgrades, service and support to our Fortune Clients from Maine to California

We don't ask you to believe us. We want you to believe our clients. Call for references today.

Vantine Computer Services can provide the solution for you!

VANTINE COMPUTER SERVICES
9 Priest Road, Sudbury, MA 01776
(617) 443-8412

Authorized Fortune Dealer

DO YOU KNOW...

there is an organization that has a thorough knowledge of

FORTUNE BAS
MULTIPLAN IDOL

We have been providing widespread support to our Fortune clients for more than 2 years.

If you need:

- training
- support
- maximum computer performance
- peace of mind relating to processing
- customization
- application results
- minimum errors and confusion
- better ideas on how to increase the power of your Fortune

Call Steve Rosenfeld at (314) 837-7111

SUPERIOR COMPUTER SYSTEMS, INC.
917 N Hwy 67, Suite 105
P.O. Drawer T
Florissant, MO 63031

Making computers work for you

JONES BUSINESS AUTOMATION CENTER

a FORTUNE SYSTEMS MASTER DEALER

and the JBAC AFFILIATE DEALER NETWORK are ORGANIZED TO SUPPORT FORTUNE END-USERS with OUTSTANDING PROFESSIONAL SYSTEMS SUPPORT at REASONABLE PRICES

* BAS/IDOL SUPPORT
* FORTUNE:WORD
* MULTIPLAN
* CUSTOM DEVELOPMENT
* COMMUNICATIONS
* HARDWARE UPGRADES
* MAINTENANCE
* LASER PRINTERS
* OCR PAGE READERS
* COMPUTER NETWORKS

Inquire about: **The JBAC "PERFORMANCE SOLUTION"**
Upgrading your present system to
a new SUPER PERFORMANCE SX-70
the Price Performance Leader

The JBAC Affiliate Dealer Network		
ADMS - Advertising Data Mgmt 4173 First Avenue Tucker, GA 30084 (404) 938-8100	**ATOM - Automated Time/Office** 126 Fieldstone Drive Ringwood, NJ 07456 (201) 962-6497	**MicroAge Computer Store** 1939 Poplar Avenue Memphis, TN 38104 (901) 722-8280
BRINC Business Systems 1662 Savannah Highway Charleston, SC 29407 (803) 763-0599	**CompuSystems of Georgia** 2200 Northlake Parkway Tucker, GA 30084 (404) 491-8004	**DataLogix** 3000 Dundee Road Northbrook, IL (312) 291-9225
DataLogix 200 Central Park Avenue Hartsdale, NY 10530 (914) 997-1627	**Future Systems** 2525 Washington Road Augusta, GA 30904 (404) 737-8313	**Software Sciences** 3100 Evans Avenue Fort Myers, FL 33901 (813) 334-1646
DRS Enterprises 127 E. Murphy Street Madison, NC 27025 (919) 427-0864	**Synergistic Business Systems** 4300 Paper Mill Road Knoxville, TN 37909 (615) 584-0211	**MultiUser Systems, Inc** 10818 N. W. 6th Court Miami, FL 33168 (305) 758-2007

JONES BUSINESS AUTOMATION CENTER

834 Tyvola Road, Suite 112
Charlotte, NC 28210
(704) 525-0582

Records Processing, Continued from page 15

/*If you have members from a city with multiple zip codes, but don't want to try to remember each zip code area, do it this way. The glossary will fill in the first four digits of the zip and then stop so you can enter the correct fifth digit. Note that using "*keyin*" rather than "*keysin*" permits only a single keystroke. You do not have to hit <EXECUTE> to continue, but by the same token if you make a mistake you will have to end the glossary in order to get back and correct it.*/

```
if(city == "W")
{
     "Warwick" return
     merge "sta" MERGE "RI" return
     merge "zip" MERGE "0288"
     call keyin return
}
```

/*Finally, for those members living in other cities which you don't want to bother coding, here is a general code ("r") that fills in some of the information (the state and the first three digits of the zip code) and lets you fill in the rest.*/

```
if(city == "r")
{
     call keysin return
     merge "sta" MERGE "RI" return
     merge "zip" MERGE "028"
     call keysin return
 }
call prompt ("")      /*clears the
prompt area*/ }
```

/*CONTROL DOCUMENT ENTRIES*/

/*This same glossary, "*mem.gl*", can be used as your **control document** for sorting and selecting records to the output document. **Entry s** provides an alphabetical listing by last name and first name. **Entry n** provides a listing by membership number. **Entry z** is a zip code sort with last names alphabetized within each zip code.*/

```
entry s
{sort <lnm><fnm>}

entry n
{sort <num>}

entry z
{sort <zip><lnm>}
```

/*The next entry, **entry x**, uses the select and save record function of a records processing control document to aid in printing a list of records where the membership has expired. The *select_record* command is followed by an "if test" to find those expiration dates earlier than the date specified and to save the records where such dates are found. The entry then sorts those records selected by their expiration date and the member's last name.*/

```
entry x
{
   select_record
   {
     if(<exp> < 06/19/86")
     save_record
   }
   sort <exp><lnm>
}
```

Running the Glossary

Once you have created your glossary, cancel out of the document and, hopefully, it will verify properly (again, be sure you have learned how to use **Fortune:Word** glossaries to get the most out of **Records Processing**). Attach the glossary.

Now go back and edit the document "*mem.ls*" which you previously created. The document is blank, but when you press <LF/GL> followed by "a", you will see the field label "<num>" appear. Type in the first membership number and press <EXECUTE>. The glossary will print a return graphic and the next field label <lnm>. Type in the last name and the glossary will go on through first name and first address line and then second address line. If you have no second address line, type "**" here and then press <EXECUTE>. The glossary will print a return.

The glossary now calls *entry **** which contains the information relating to the city, state, and zip. The glossary will print "<cty>" with a prompt in the upper right corner of the screen for you to enter the city code. If you type "W", for example, the glossary will print "Warwick" and a return, then "<sta>RI" and a return, then "<zip>0288" and wait for you to fill in the final zip code digit. As soon as you have typed it, the glossary will return to entry a, pick up where it left off, and print "<tel>", waiting for you to type in the telephone number and press <EXECUTE>.

When the glossary asks for the expiration date, type "l" if it is a lifetime membership. The "l" will be overwritten by "LIFE". If it is an annual membership, type any other key (we usually just hit the space bar) then type the month and day of expiration and press <EXECUTE>. The glossary will fill in the year.

See Records Processing, Page 18

Records Processing, Continued from Page 17

As the expiration date was the final field in the record, the glossary will print "<>" (an open and close merge) to indicate the end of the record. A prompt will appear asking if you want to stop at this point. If you do, type "y" or "Y" and you will exit the glossary. You can either go back and correct a problem or save the document. If you want to resume, just type <LF/GL> followed by "a". If you do not wish to stop when prompted, hit any key (again, we usually just press the space bar), and the glossary will return to `entry a`, first checking your position on the page. If you have gone past line 56, a page break will be inserted before the next record is begun.

Printing an Expiration List

Suppose you want to print a list of all the members whose memberships have expired. You will need four documents in all: the **list document** (*mem.ls*), a **format document** (which you will now create), the **control document** (*mem.gl*), and an **output document** which will be created by Records Processing.

Go to the main menu and use the document creating selection. Call this document "*memexp.fm*" for "**Expired Membership Format**." Change the format line with tabs to create three columns. Then type in the column headings and the field labels as below, using the "**repeat**" command to list 50 entries on each page (see last issue for use of the repeat command):

```
EXPIRATION DATE    MEMBER NAME     MEMBERSHIP NUMBER

<repeat(50)
<exp>              <lnm>, <fnm> <num>
>
```

Then put in a page break. If you expect to have a number of pages, you might want to go to the header or footer page and put "`page #`" for automatic pagination. When you are through, cancel out of and save this document.

From the **Fortune:Word** main menu, press <COMMAND> followed by "`rpr`". The Records Processing menu will ask for the **list document**. Type "`mem.ls`" and <RETURN>. For **format document**, type "`memexp.fm`" and <RETURN>. For **control document,** type "`mem.gl`" and specify entry "x", followed by <RETURN>. You can then choose to make an output document only, make the output document and print it, or print the output document only. Choose the default, which is to make the output document but not print it. When asked for a name, call it "*`memexp.out`*". **Records Processing** will now select the records with expired memberships, sort them by expiration date and then alphabetically by member name, and arrange the information according to your format document. When completed, you are returned to the main menu and you can then edit your output document like any other **Fortune:Word** document, print it, archive it, or delete it.

Note that if you have special instructions for printing a document of this type, you can include those on the print menu for the format document. Just select the print function and fill in the format document's name. Then you can change the left margin, the length of the paper, where you want the first header or footer to print, the type size, and if you are using a tractor feed or sheet feed. Every time you use this format document, the print information will be transferred to your output document.

As you can quickly see, the uses of **Records Processing** are endless. You might, for example, want to print letters to all your expired members reminding them that their next year's dues should be paid. You may have a special program for lifetime members that you want to notify them about.

And you will also, as you continue to work with glossaries and with **Records Processing,** find more sophisticated ways to make your information management tasks easier. Don't hesitate to share your discoveries with the rest of us!

Current Release Levels for Fortune Software

The following are the release levels for all Fortune software as of May 30, 1986

3270 Interactive Terminal Emulator	1.1
3270 SNA Emulator	1.0
3780 Remote Job Entry	1.0
Business BASIC Interpreter	1.4
C-Language	1.7
COBOL-Micro Focus(Compiler&Runtime)	2.5
COBOL-Micro Focus(Developer Package)	2.5
Development Utilities	1.1.1
Document Conversion	1.0
FOR:PRO Multi-User	1.8.1.1
FORTRAN-77 Language	1.3
Field Engineering Diagnostics	4.0
Fixit	1.1
Fortune-to-Fortune Copy (FTF)	1.1.1
Fortune:Link	1.0
Fortune:Windows	1.0.1
Fortune:Word (All Modules)	2.1.1
Fortune:Works(32:16)	2.1
Fortune:Works(DOS Workstation)	2.1
Fortune:Works(Graphics Update)	2.1

See **Releases**, page 19

Fortune News, Continued from page 12

Corrective Action:

If fsck does not obtain enough memory to keep its tables, it requires the use of a scratch file. The fsck command has a -t option that provides for the name of a scratch file if needed If the fsck command was issued without the -t option, then the system will prompt the user to enter the name of a scratch file.

Respond with a filename that is not on the file system being checked. The file will be removed when fsck completes.

When fsck requires a scratch file the following message will display:

needs scratch file
filename =

Enter a scratch filename after the equals (=) sign.

Product: Fortune:Link Remote printer won't work

Problem Description:

Remote printer(s) will not work after installation or reinstallation of Fortune:Link.

Corrective Action:

Any installation or reinstallation of Fortune:Link software requires that the file /m/rc/arcnet.rc be redefined if remote printer(s) are required. See Chapter 4 of the Fortune:Link Administrators Guide for complete details on this procedure.

Releases, Continued from previous page

GSS "C" Bindings	1.0.1
GSS Drivers (UNIX)	1.0.1
GSS Metafile Interpreter	1.20
GSS Plottalk	1.0.2
GSS Plotting System	1.0.2
Graphwriter Basic Set	2.2
Graphwriter Combination Set	2.2
Graphwriter Extension Set	2.2
HP Laser Printer Interface	1.1*
Interactive Terminal Emulator (ITE)	1.1.1
Language Development Tools	1.8.2
Multiplan	1.4.2
Pascal Language	1.3
Rubix	2.0
TEKTRONIX 401X Emulator	1.1
Tape Streamer	1.2.1
X.25	1.0

*as of 6/15/86

Fortune Users:

JUST POSSIBLY
THE CHEAPEST PHONE CALL
YOU'VE EVER MADE!

- 70 MB disk drives
- 10 MB REMOVABLE disk drives
- Expansion cabinets

- Laser Printers
  8 thru 26 Page Per Minute
  (no extra software required!)

- Genicom Impact printers
- Multi-tech 2400 Baud modems

Shop around... then call us.

Robbins & Associates
(206) 771-2286

* * * * * * * * * * *

Call us **COLLECT** today for a quote

BACKUP AND RESTORE UTILITY

PROTECTION MADE EASY!

- o Back up all or selected user directories
- o Restore all or selected user directories or files
- o Format diskettes
- o Review backup volumes
- o Menu-driven, easily installed
- o Complete documentation

Also copies the files necessary to preserve your system configuration--a tremendous help if your system ever needs to be cold booted!

$150.00 Backup/Restore Utility
$200.00 Utility + 2 boxes 5 1/4" 96 TPI high-quality diskettes

For more information, call CTD at (313) 540-1282.

To order, make check payable to *Computer Trade Development Corporation.* Send to:

BACKUP/RESTORE UTILITY
Computer Trade Development Corporation
Suite 1386
Birmingham, MI 48010

CTD

Company:_____
Contact:_____
Address:_____
City:_____
State:_____ **Zip:**_____

SOFTWARE FOR THE FORTUNE 32:16

MULTIPLAN LINK - can pass data from BAS/IDOL/BASIC files to Multiplan and back. For example, it could be used to send data from any of the BAS files (Chart of Accounts, Customer Master, Inventory Master, etc). Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

RECORDS PROCESSING LINK - can pass data from BAS/IDOL/BASIC files to Records Processing and back. For example, it could be used to take customer address data from the Customer Master and will automatically make the Records Processing List document for mailouts. Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

TERMINAL/PRINTER LINK - provides capability to print Multiplan, and BAS/IDOL reports on a printer connected to a workstation, view any report on the terminal before printing, convert any report to Fortune:Word format; accumulate multiple BAS/IDOL reports before printing, backup reports onto diskette; copy or rename report files; print multiple copies of the report on either the primary and/or terminal printer. Also has option to print BAS/IDOL reports on HP LASER printer ($50 extra) **$195.00**

TELEPHONE LINK - on line telephone message, telephone directory and inter-office mailbox/memo system which can also be used as a simple calendar reminder system. Provides control, printing and viewing of on-line data for each user. **$295.00**

FINANCIAL LINK - stores BAS Income Statement and Balance Sheet data in files. Once stored, it can be sent to Multiplan or instantly printed. **$195.00**

MENU LINK - provides software and documentation for adding items to the Global Menu as well as a sub-menu for your use Includes backup script for user directories and BAS/IDOL/BASIC programs and data. **$95.00**

CALCULATION LINK - provides programs for amortization, depreciation, loan repayment, averaging (with graph), linear correlation, and breakeven analysis. **$95.00**

KOMPACT PERSONNEL ACCOUNTING - provides for data capture of personal, wage/salary, job, education, salary/wage, appraisal training, dental, medical and life insurance data. Includes over 80 pre-defined reports/worksheets. **$495.00**

CRT LINK - provides information for using other terminals on a Fortune System (WYSE, TV, IBM 3101). **$95.00**

AUTOMATIC REPORT LINK - provides capability to print numerous IDOL defined reports automatically based on frequency or application without having to select each from a different menu. You identify frequency/application codes. **$95.00**

TAPE LINK - provides one menu for all tape backup utilities. Provides daily, weekly, monthly and individual backup and restoration routines. Includes rewind, retension, and tape unit startup/stop utilities. User friendly. **$95.00**

PRINTER LINK - provides printcaps for Okidata 84, 2410, TI 855, and Anadex 6500. Includes shell scripts for changing pitch and LPI. Allows these printers to be used in word processing. **$95.00**

MODEM LINK - allows Hayes 1200 Smartmodem or compatible to be used on one port for both incoming and outgoing calls. **$95.00**

INSTALL LINK - provides documentation and software example for installing your product/menu through "S5". **$95.00**

REPORT LINK - modifies IDOL Report Generator Program to allow stacking of up to 8 fields in an IDOL defined report. Also corrects problem with displaying subtotals for all stacked fields. **$95.00**

WESTERN LINK - provides instructions on how to use your Fortune for sending/receiving telex messages using Western Union's Easylink. Can replace your telex! Fortune ITE software is required. **$95.00**

BACKUP LINK - provides menu selection for backing up /b (BAS/IDOL/BASIC), /u (Fortune:Word, Multiplan) and entire system onto diskette. Also has program to "unhang" a terminal that is hungup just by entering the terminal number; a program to restore one file from a multi-volume backup; and a selection to tell you which volume a backed up file is on. **$95.00**

UNI-KOMP 9803 Rocktree Houston, Texas 77040 (713) 937-7782

Basic Advisor, Continued from page 1

*Question: I can't imagine why the **@T** command you discussed in your article for April (/u/fortune news, Vol 3, No 4) didn't work for you. I tried it and it worked just fine. I use the **SHELL** command frequently when I program in Business **BASIC**, and being able to enter **BASIC** commands in lower-case letters has saved me a lot of aggravation. Maybe there's something wrong with your 32:16.*

*Question: Try the **@T** again. It works okay on my machine!*

Answer: It works on your machine and the machines of all the other programmers who called me. Maybe I should have asked for calls from those who had the same negative results I experienced. I thought I was going to develop some kind of a complex until I heard from Jim Dyson of Fortune Systems.

Jim, who is, according to many, the **BASIC/BAS/IDOL** guru for our favorite computer manufacturer, informed me that not all releases of Business BASIC accept the **@T** command. He said I probably had one of the earlier versions of the **BASIC** interpreter and therefore could not make **@T** work (and with my luck, that figures). So, from all of us struggling programmers, I send thanks to Mr. Paul Janoski.

Question: Guess what! There's a new IDOL/BAS update on the market from Fortune Systems. This one is dated 4-11-86 and is labeled "Release Level 2.0". Fortune wants $75.00 for it. My question is, if I install this diskette will it disrupt any of the custom programming on my 32:16? If so, I don't want it.

Answer: I have not personally had the opportunity to examine this product (what can you expect from someone with such an early version of Business BASIC?), but I found someone who has one. My source is highly reliable, and his name has appeared in this publication several times in the past. (I am withholding his name this time, because he seems to squawk every time he sees it in print, especially when he is very busy.)

"Source" examined the diskette at my request and advised me that it contained only minor program adjustments. He is of the opinion (and I concur) that installing this product will cause no damage to most customized **IDOL/BAS** installations. If you have any doubts, ask the person who modified your system whether he/she changed any of the original programs. If so, have that person examine the Update Diskette to see if any of your modifications will be disturbed by its installation. If you do not wish to rekindle a relationship with the original programmer, do an End-of-Day backup of your program files before you install the upgrade. That way if something goes wrong, you can restore with the Start-of-Day procedure and lose nothing.

*Question: I have been reading your articles and have especially enjoyed the BASIC programming questions. I would like to see you discuss commands (such as **DIM** with string variables) which, I think, are unclear in the Business BASIC manual.*

Answer: I did not realize there was a problem with the documentation for the **DIM** command until you brought it to my attention. Actually, the book is fine if you ignore the program example on page 1-33. Let's take a look at that program and see if we can find what's wrong with it.

```
10 REM STRING VARIABLES
20 DIM A$(35),B$(5,"*")
40 PRINT
50 PRINT "ENTER YOUR NAME, LETTERS ONLY"
60 INPUT A$
65 PRINT
70 PRINT B$,A$,B$
80 SAVE "STRING",350,0,0
90 END
```

See Basic Advisor, page 22

For The Best

B.A.S. and IDOL

SUPPORT and TRAINING

Call

Jackie Biggs

at

Executive Data Services, Inc.
415-359-7644

We Support Modified Software !

Basic Advisor, Continued from page 21

When you run this program, here's what happens:

```
System: ENTER YOUR NAME, LETTERS ONLY
User:   VIC PERSON

System: *****VIC PERSON*****

System: READY
```

The primary problem in this program example is that it uses a lot of unnecessary commands. All we really need to know is how the **DIM** (or **DIM**ension) command works on the variable B$ in statement 20. (After statement 20 is executed, B$ contains five asterisks). Aside from the fact that this is, in general, a pretty stupid program, three major defects exist.

Defect 1: Why on earth would you dimension A$ to 35 spaces in statement 20 when you will be asking the user to redefine the variable at statement 60 with the INPUT command? Whatever A$ was before statement 60 is executed will be lost forever in the memory of the programmer.

Defect 2: This may be nit-picking, but I do not like the **PRINT** commands at statements 40 and 65. Business BASIC provides a handy little mnemonic called 'LF' (**L**ine **F**eed) which, when used appropriately in statements 50 and 70, eliminates the need for the **PRINT** commands. Then again, why must you skip a line in the first place? It certainly doesn't help explain **DIM**.

Defect 3: Statement 80 really burns me up. Once you have entered the program and run it, a permanent program called "**STRING**" will have been written to your hard disk. Now isn't that just what every system needs? The book has tricked you into storing this gem forever. I suggest that if you follow the book, type ERASE "STRING" when you are done.

To demonstrate how the **DIM** command works with a string variable, let's forget this program and try to create the line of characters that appears on the screen at the end of a page on your word processing documents. This line is 80 characters long and contains alternating "=" and space characters. In the Business BASIC command mode, type in the following program:

```
10 REM REAL LIFE STRING VARIABLES
20 DIM A$(80,"=")
30 FOR I=2 TO 80 STEP 2
40 LET A$(I,1)=" "
50 NEXT I
60 PRINT A$
70 END
```

Type `RUN <RETURN>` to see the page end line (A$).

In this example, A$ consists of 80 "equals" signs after statement 20 is executed. (If you wish to examine it at this point, type `25 PRINT A$.`) The loop at statements 30 thru 50 replaces every other character in A$ with a space.

Question: You frequently write about things that are way over my head. Why don't you gear your articles more to the end users?

Answer: I write about subjects that are brought to my attention. If you have something that you wish to see discussed, please call me at (301) 448-9460, and I will try to accommodate you. I prefer to take calls on Tuesdays and Thursdays between 8:30 and 9:00 AM (Eastern Time) and between 4:00 and 6:00 PM (Eastern Time). To date, programmers have been providing me with more input than end users, but you are certainly welcome to try and change that.

Stand Alone Version
$849.00
Ask About the Interactive Version, too!

JOB COSTING

The Right Tool for All Trades

BaSiC SOFTWARE CORPORATION

For More Information Call or Write...
BaSiC Software Corporation
5201 Powhatan St., Baltimore, MD 21207
301-448-9460

Fortune Users:

G.T. HAWES & CO.
An Authorized Fortune Master Dealer
Announces a Fortune Liquidation Sale where
NO REASONABLE OFFER WILL BE REFUSED!

It's true. We're ever on the look out for *good buys* and we struck a proverbial FORTUNE! Two major users wanted out and we were there. We said, "HELLO." They said, "GOODBYE." And what a *GOOD BUY* it was! You guessed it, we're overstocked and must move the product. Take advantage of our prices because they were never lower.

CALL GEORGE AND SAY HELLO, SO I CAN SAY
GOODBYE!!!
NO REASONABLE OFFER REFUSED!
516-568-0044
Equipment Schedule Available on Request

"If you don't see my name on your computer shipment you probably paid more than you should have."

G.T. Hawes & Co., Inc.

475 W. Merrick Road
Valley Stream, NY 11580
516 568-0044

PROGRAMMERS

The leading manufacturer of status display panels for the fire alarm and security industry is seeking programmers and junior programmers to develop new applications and refine existing programs.

Successful candidates will be experienced in IDOL/BAS and SMC Business Basic for Fortune and Basic 4 systems. Familiarity with UNIX and system set-up and maintenance is desirable.

Candidates interested in joining a growing team-oriented organization located in Falls Church, Virginia should write or call with resume and salary history, to:

WSA
308 Hillwood Avenue
Falls Church, Virginia 22046
703 241-0687

Expand your Fortune® without spending a fortune.

New equipment costs big bucks. The Good Fortune Trader sells fully tested burned-in equipment, refurbished to Fortune® standards at prices that won't make waves with your cash flow. Call or write today for a listing of our specials.

THE GOOD FORTUNE TRADER

60 DAY WARRANTY

a division of Computer Services of Atlanta

(404) 392-9938

1534 Dunwoody Village Parkway/Atlanta, Georgia 30338

A Note about Printing

You may have noticed two things about this newsletter -- 1) it looks slightly different than other newsletters, and 2) it's later than usual. There is one explanation for both of these phenomenon, and it's called a **Macintosh Plus**. Yes, you heard it right. We at The Cambridge Consortium have introduced a new computer into what had been a strictly **Fortune** shop. We may sound defensive, but we think there is no reason to be -- *the right tool for the right job* has always been a strongly held belief around here, and at this time, the Macintosh seems to be the right tool for typesetting.

We still use and love our Fortune computers. In fact, much of the text for this newsletter was keyed in using Fortune:Word and even checked for spelling. We then moved it over to the Macintosh using the **kermit** software package and finally prepared it for typesetting using the **Pagemaker** program from **Aldus**. This program gives us much more flexibility in terms of design, and will ultimately make our preparation process much quicker.

So the moral of the story is you will be getting your future newsletters sooner, and you may expect that the design of /u/fortune *news* may change in coming months.

/u/fortune *news*

20A Prescott Street
Suite 28
Cambridge, MA 02138

/u/fortune* *news...*

The Newsletter for Users of the Fortune 32:16 Computer **July 1986/Volume 3 Number 7**

Users' Spotlight

Editor's Note: *With this issue, we begin a new series,* ***Users' Spotlight,*** *that will appear occasionally in* ***/u/fortune news.*** *In this column, we will illustrate how* ***Fortune 32:16*** *owners typically and innovatively use their machines to support work-related activity. Each article will focus on a particular profession or business, based on responses to our Users' Survey and telephone interviews.*

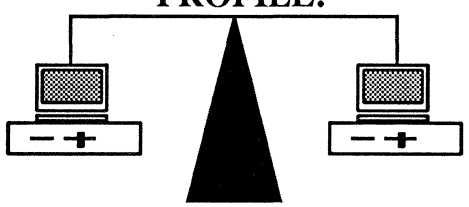
This month's article focuses on how attorneys and legal services offices around the country use the Fortune and its software. In future issues of ***/u/fortune news,*** *we will present articles on how users in medicine, business, health, education, etc. use their systems and software.*

Last year, we mailed a Users' Survey which a number of our readers completed and returned. In the survey, we asked questions about hardware configurations, types of printers used, software, support, and peripherals. This information gave us a sense of how Fortune users take advantage of their computers, of what they identify as problem areas, and how in many cases they propose innovative solutions to those problems.

Recently, we conducted a series of telephone interviews with several subscribing attorneys in order to get additional information about the applications problems that may be particular to each field. Our hope was to identify similar concerns and to discover the users who, in solving problems for themselves, could serve as a resource for others.

PROFILE:

How Lawyers Use Fortunes



Hardware:

One of the interesting results of our survey was the similarity between offices in terms of hardware configurations. A typically configured small law office looks like this: 756K to 1 megabyte of memory with a 20-30 megabyte hard disk running a single CPU with 1-5 workstations. Most offices interviewed said they back up their data (daily and/or weekly) on floppies, and most expressed an interest in switching

See **Lawyers**, Page 15

Featured in This Issue...

Users' Spotlight -- A new column which explores how a particular group of people use their Fortunes. This month we feature **Lawyers. . .Page 1**

The Basic Advisor -- Adding more than eight terminals, BASIC support, etc. . **.Page 1**

News from Fortune -- Read all about Personnel Changes, New Dealers and New Divisions ... **Page 2**

Our tickler **Glossary Entry** should help you remember those important dates. Don't forget to see **Page 3.**

The Unix Directory features a discussion of ***cat*** and ***more*** -- two UNIX commands for viewing files. . **.Page 5**

BASIC printing problems, adding memory to a Fortune/Wyse 1000 and more -- all in **/u/help. . .Page 11**

The Basic Advisor

Ray Wannall is president of Basic Software Corporation in Baltimore and is contributing to this publication independently.

I find it necessary, once again, to depart from the usual question and answer format to discuss some tidbits of information that have been floating around for a few weeks. If you recall, last year the SMC software was acquired by John-L Johnson and Concept Omega. I reported in this column that since John-L was the original father of **IDOL** and **BAS**, this was potentially good news for Fortune end users. To date, however Concept Omega has not been of any help to us Fortune people. But if the information I have been hearing lately is true (and I have no reason to believe it is not), this may change very soon.

According to sources at both Fortune and Concept Omega, the latter will be "selling and supporting" Fortune Business Applications. I don't know yet if this means Concept Omega will be handling the **BAS** we all love or the Thoroughbred they all love, but this has to be a step in the right direction. I am also not yet informed of a time frame for this transfer of responsibility. I must emphasize that Concept Omega

See **Basic**, Page 10

*/u/fortune *news* is published by The Cambridge Consortium, Inc. It is not affiliated with Fortune Systems Corporation.

NIFE™ POWER

New Technology Uninterruptible Power Supply Protects Microcomputers and Provides 100% Power Conditioning

SAB NIFE has **eliminated** the battery charger from the typical **UPS** package. A **new technology** reduces weight, component count and yields highest efficiency which **saves** the user **money** in operating cost.

Advanced technology has eliminated the transfer switch. For protection of your valuable data. Choose **NIFEPOWER,**™ the state of the art **UPS** from **SAB NIFE Inc.** For information call 1-800-556-6764. (In RI, 333-1170)

SAB NIFE

NIFEPOWER™ is a registered Trademark of SAB NIFE, Inc.

/u/fortune *news. . .*

Editors: Josh Lobel, Mark Palmerino
Associate Editor: Benzion Chanowitz
Technical Editor: Scott Bradner
Advertising Manager: Mark Palmerino

/u/fortune *news. . .* is published 12 times per year by The Cambridge Consortium, Inc. Neither **/u/fortune** *news* nor The Cambridge Consortium are affiliated with Fortune Systems Corp. Contact us regarding subscription rates. Quantity discounts are also available.

Back issues are offered at the following prices:

Vol 1, Numbers 1-3 $1/each	Vol 2, Number 1-7 $3/each
Vol 1, Number 4 $.50/each	Vol 3, Number 1- $4/each
Vol 1, Number 5-8 $3/each	

Although every attempt is made to insure that our contents are error-free and without risk, The Cambridge Consortium is not responsible for losses that occur due to correct or incorrect interpretation of **/u/fortune** *news*.

For more information about us, please call or write: **u/fortune** *news*, 20A Prescott Street, Suite 28, Cambridge, MA 02138. Tel. 617 876-4763.
The text for this newsletter was prepared using a Fortune 32:16 for text input. The information was then transferred to an Apple Macintosh Plus with kermit and formatted for typesetting using the Aldus Pagemaker program.

UNIX® is a registered trademark of Bell Laboratories,Inc. Fortune 32:16™, Fortune:Word™ and FOR:PRO™ are trademarks of Fortune Systems Corporation. Multiplan™ is a trademark of Microsoft. IBM-PC™ is a trademark of IBM. MS-DOS™ is a trademark of Microsoft. Macintosh Plus™ is a trademark of Apple Computer. Pagemaker™ is a trademark of Aldus Corporation. HP Laserjet™ is a trademark of Hewlett-Packard Corp.

©1986 by The Cambridge Consortium, Inc. 7/86

News From Fortune

Davis Accepts Position at Fortune as Executive Vice President of Marketing.

James Campbell, Fortune's President and CEO, announced that **Robert A. Davis** has joined Fortune Systems Corporation in the newly created position of executive vice president of marketing. Davis has extensive experience in establishing and building reseller networks as well as in establishing solid OEM relationships. Campbell says, "Bob will be putting together a strong worldwide marketing and sales team and a program to complement our excellent products and sound financial position." Davis assessed Fortune as follows, "Today, Fortune is more than a hardware or software company. It is an integrated organization. The existing hardware line provides outstanding price/performance value for the reseller and small to medium-sized business or departmental end-user. The hardware that will be introduced later this year is going to be very competitive for both resellers and OEMs. Fortune also has outstanding office automation software, educational/training capabilities, and support, proving that the company is fully dedicated to satisfying the reseller's wants and needs."

Before accepting the new position at Fortune, Davis was executive vice president and chief operating officer at CADTRAK Corporation, a CAD/CAM company. During his 25 years in marketing and sales, Davis was vice president of sales for Qantel and then left to set up Northern California Business Computers, a Qantel distributorship. After building the distributorship into one of the largest and most successful of Qantel's dealers, he sold the company and rejoined Qantel where, as vice president for branch operations and vertical markets, he rebuilt the firm's reseller network. Davis' experience also includes marketing and sales management for Basic/Four and ten years with Olivetti, where he held a variety of national and international positions.

* * * * *

Olin Joins Fortune Systems as Vice President of Sales

In making the announcement of Paul Olin's appointment as Vice President of Sales, Davis noted that one of the ingredients which has been missing in Fortune's team has been an aggressive sales leader who has the experience needed to work with field sales organizations, resellers, and OEMs. "We have some sound agreements with major accounts and distributors," Davis commented. "Now we need a person like Paul who is a strong team builder and team player to work closely with and assist our resellers and OEMs. He has worked in both arenas and knows how to stimulate sales, growth, and success."

See News, Page 22

The Glossary Entry

(If you have read the article on lawyers elsewhere in this issue, you may have noticed ***Mark Fellman's*** *use of a tickler file. This file is used to create a list of "reminders" which help him keep track of appointments, due dates, etc.* ***Mr. Fellman*** *does this job by using a combination of glossary functions and* ***Records Processing,*** *all from within* ***Fortune:Word.*** *We felt that his efforts presented an elegant and simple solution to what is a common problem -- keeping track of time . So,we've decided to present his work this month. The text in this article is a combination of our language and his.)*

An Overview of How it Works

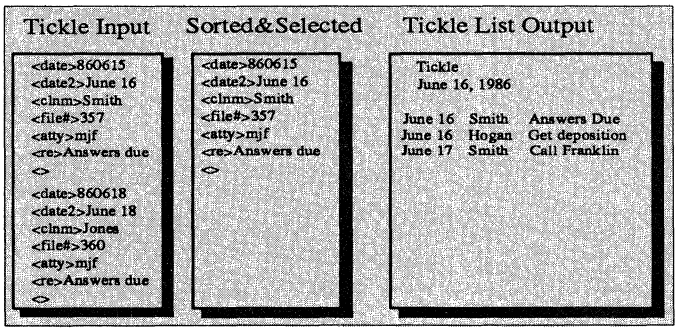
A **Fortune:Word** document called "ticklelist" is created. It will contain two versions of the date you want to be tickled, who's tickling, who should be tickled, and why. Here is a sample entry:

```
<date>860715
<date2>July 15
<clnm>Smith
<file#>357
<atty>mjf
<re>Answers to questions due
<>
```

Those of you who have read our articles on **Records Processing** will recognize the format of the field names which have been included between the < and > symbols, e.g. date. We'll discuss the specifics of each field shortly.

Once this file has been filled with dates, it can be sorted by the **date** and **atty** fields using the **Records Processing** module of **Fortune:Word**. This will produce a list of reminders for each attorney sorted by due dates. Once the list is printed out and saved, expired reminders can be edited out of the file.

As with all systems, the more automatic the process, the more likely that it will actually be used. Mark Fellman has refined the system sufficiently to make it useful. He uses a glossary entry to enter the reminders, and the glossary entry will easily allow you to make entries while you're working on another **Fortune:Word** document.



Setting up the Files

The first thing you need to do is to create a document called **ticklelist**. Mark Fellman has created this document in a library called **/u/office/tickler**. You may put it anywhere, but if there are several people who will need access to it, it will be necessary to make it available to everyone. The easiest way to do this is to use the **Fortune:Word** Menu to create the library and document, just as you would any other document.

The Glossary entry will easily allow you to make entries while you're working on another Fortune:Word document.

(There's no magic here.) For the moment you will just create an empty document. Do not type anything in yet. You then need to change the permissions for the document. You can do this from the **Global Menu** using the **S1, Option 19** command. You can also do it easily from the UNIX shell by logging in as `root`, changing to the directory where the file resides (e.g. `cd /u/office/tickler <CR>`), and changing the permissions by typing:

```
chmod 777 ticklelist* <CR>.
```

The **chmod 777** command sets the permissions on a file so that anyone can read or write to the file. The **asterisk** ensures that any file that begins with the pattern "ticklelist" will be affected. This is necessary because each **Fortune:Word** file has associated files that end in **.dc** and **.fr** -- in our example they are **ticklelist.dc** and **ticklelist.fr**.

The Glossary to Enter Ticklers

In order to take full advantage of the system, you will also want to create several glossary entries. If you are already using a glossary, you may want to add these entries to your existing glossary. If not, you probably should create a glossary in the **/u/office/tickler** library and call it something like **ticklegl**. Use the same **chmod** command explained above to change the permissions for this file.

Mr. Fellman enters ticklers by using a glossary entry from other **Fortune:Word** documents. This is done by opening a window in the document you are working in, bringing in the tickler file (the one that you created above), adding an entry, and closing the window, leaving you just where you were when you started. The entry is as follows:

See Glossary, Page 12

Our Memory Doesn't Cost You A Fortune. . .

Is Your Fortune Living Up to its Potential?

Would you like to:

Speed up your computer?
Add more users?
Use the latest software, like Fortune:Windows™?

Compatibility:
The F1MB is absolutely 100% compatible with Fortune's own 1 Meg memory board.

Dependability:
All boards are burned in for 72 hours and tested for 48 prior to shipment. *Our One Year Repair or Replace Warranty underlines our confidence in the product.*

Affordability:
USA is proud to have equaled and exceeded the quality and performance of our competition, while offering substantial savings for both the dealer and end user.

Repairability:
The F1MB facilitates repair in the field as it is 100% RAM socketed. Now you can repair a board by simply replacing a defective chip rather than face an expensive board exchange -- reducing maintenance costs and system down time.

If you have a Fortune SX, XP45 or recent model XP30, you can use our board by simply plugging it in. Older systems can be upgraded to use our board with a simple and economical modification.

For Details Call:

Upgrade Service Associates
3 Marion Road, Burlington, MA 01803
Phone: (617) 239-8114

Our Guarantee: If you're not completely satisfied within 30 days, *USA* will gladly refund your money.

The UNIX Directory

Viewing Files with *cat* and *more*

This month we will discuss two very simple UNIX commands that are used primarily for viewing the contents of files (though **cat** has other uses, which we will touch upon later). These two commands are very easy to use and are very useful to everyone who uses the Fortune computer. Chances are you have already used these commands numerous times. However, I hope that this discussion of **cat** and **more** will reward you with some new understanding of how these commands work.

Let me say at the outset that I was surprised to find just how versatile these commands can be. Each one had hidden capabilities that I didn't know about until just recently - and some of these capabilities are useful indeed! For example, I cannot tell you how many times I have used **more** to view a file and then after being half way through the file, wished that I could somehow back up and see some previous portion on the file without getting out of **more** and then restarting it. Well, I just found out that if you hit a single quote character ('), **more** often jumps back to the beginning of the file.

cat: Concatenating Files in UNIX

I think, though, that we are getting a bit ahead of ourselves. Let's start at the beginning. **cat** is a command which can be used for viewing a file on your terminal. It once was extremely useful for this purpose, especially when terminals were hard copy (running at slow speeds like 110 and 300 baud). However, the command is not that useful for this purpose now because when a file is "catted" to a Fortune terminal, which may be set to 19200 baud (or even faster if it is the console), it flashes by too quickly to be read. Actually the name "cat" comes from the word con*cat*enate, which means to join. The main thing that **cat** does is concatenate multiple files together and then prints them on your terminal (called the standard output). Suppose, as I was writing this article on **cat** and **more,** I decided to put my explanation of **cat** in a file called *cat.text* and the explanation of **more** in file called *more.text.* If I wanted to see both files on my terminal, I could type:

```
$ cat cat.text more.text
```

This command would type the contents of *cat.text* immediately followed by the contents of *more.text.* One handy thing that can be done is to *redirect* what would naturally come to the terminal into a new file. For example, if I wanted to create a new file called *all.text* which contained my explanations of **cat** and **more,** I could type:

```
$ cat cat.text more.text > all.text
```

The greater-than sign (>) in UNIX instructs the computer to take the output that would normally come to the terminal (in this case the contents of the two files *cat.text* and *more.text*) and put (or redirect) it into the file called *all.text.* Be careful because the **cat** command will overwrite anything that might have previously been in *all.text.* Also, be very careful about saying something like:

```
$ cat  cat.text more.text > more.text
```

because UNIX will first erase the contents of *more.text* and then try to concatenate *cat.text* with *more.text.* Unfortunately, by the time the **cat** command gets to the *more.text* file, it is empty!

There is one other major use for the *cat* command. It can be used as quick way to create files.

There is one other major use for the **cat** command. It can be used as a quick way to create files. Normally we would use either a UNIX editor (like **ed, vi** or **sc**) or **Fortune:Word** to create a file. However, there may be times when we just want to quickly put some information in a file. We can use **cat** to do this. If I just wanted to create a file called *quickfile* that had some piece of information in it, I would do the following:

```
$ cat > quickfile
This is some interesting piece
of information that is in this file
^D
```

When you use the **cat** command in this way, whatever is typed on your keyboard gets redirected to the named file. The way you stop this redirection is to type a CTRL-D (^D). The other day I had a situation where this came in quite handy. I was at a friend's house who had an IBM-PC with a modem and communication program (not **kermit**!). He wanted to learn how to send a file from his IBM-PC to a Fortune. The process is a conceptually simple one. First, one has to log onto the Fortune and tell it that something is about to be typed on the keyboard. This could be done by getting into the input mode of an editor or, as I chose at the time, by issuing the command `cat > filename`. Then we had to switch back to the IBM-PC and instruct the communications program to send the file. Once the

See UNIX Directory, Page 6

UNIX Directory, continued from page 5

file was sent, we switched back to the Fortune, issued a CTRL-D (which told **cat** that we were done sending text), and basked in the glory of another successful file tr sfer. *(editor's note: although this is an effective way to move a file from one machine to another, it is not foolproof, and probably won't work well with large files.)*

Special uses of *cat*

There are some special uses for **cat** that may come in handy some day. First, if you use the `-n` flag, **cat** will output the contents of the named files with each line prepended by a number. This is a quick way to number lines in a file. This function of **cat** is essentially the same as the UNIX **num** command. If you use the `-b` flag in conjunction with the `-n` flag, then blank lines are not numbered. Another flag that is sometimes useful is the `-v` flag. This instructs the **cat** command to make control characters and non-printing characters visible. In fact, we had a situation recently where we were transfering some Wordstar files from an IBM-PC to our Fortune and after the transfer, we wanted to take a look at the file.

The -v flag instructs the *cat* command to make control characters and non-printing characters visible

Since Wordstar files use control characters to code attributes of the document (like bold print, underlining, etc.), we used the `-v` flag on **cat** to verify that files had been transferred properly. A final potentially useful option is the `-s` flag which instructs **cat** to squeeze multiple blank lines into one blank line. Thus, if a file has four blank lines in a row, the output would have only one blank line (Note: This does not alter the original file, only the output to your terminal).

more: Viewing Files without Being a Speed Reader

more is the UNIX command built for viewing files. It acts like **cat** except that it displays one screenful of the file on the terminal and then pauses. This allows the user to read whatever is on the terminal. To get the next screenful of the file, you simply hit the space bar. With this much of a description, you could use **more** to great advantage. However, there is much more to **more**. In fact, there is so much more that we will only be able to touch on a fraction of it. The description of **more** in the **FOR:PRO** book will give you many more details.

The simplest invocation is `more filename <CR>`. When you do this, you will see the contents of the file displayed on the terminal one screenful at a time. Usually, at the bottom of the screen, you will see the string `--More--`, which indicates that there is more

more also allows you to search through a file for a particular pattern

of the file to view, and what percentage of the file you have seen. If you want to continue you should hit the space bar. **more** also allows you to search through a file for a particular pattern. Say, for example, that you have a file with names and phone numbers in it. If the name of the file is **numbers**, then the command:

```
$ more +/Smith numbers
```

will display the screenful of the file where the first occurrence of Smith is found. You can also look for patterns once you've started **more** running. Whenever you see the `--More--` at the bottom of the screen, you can type a slash (/) followed by the pattern that you wish to find. The program will begin at the current point in the file and search through to the end of the file for that pattern. If the pattern cannot be found, **more** prints out `Pattern not found.`

Actually, there are a number of commands that can be entered when **more** pauses between screens. Try typing `h`. This will give you a screen with all the commands that **more** understands. Some other, potentially useful, commands are:

```
d displays 11 lines rather than a full screen
= displays the current line number
v starts the UNIX vi editor at the current line
q exits the more program, same as CANCEL/DEL
```

Some other commands can be preceded by a number. For example, if you type `54s`, **more** will skip the next 54 lines and then display a screenful of the file. Typing `7f` will cause **more** to skip 7 screenfuls of the file and then print a screenful of the file. We've already said that `/pattern` searches for the first occurrence of *pattern*. However, `3/pattern` will skip to the third occurrence of the *pattern*. Finally, the single quote (`'`) will cause **more** to back up to the position from which the last search started. If there was no previous search command, then it goes back to the top of the file.

There is much more to both of the UNIX commands **cat** and **more**. We urge our readers to try them. For more information about these commands, we direct you to the **FOR:PRO** manual as well as any book on the UNIX system.

Mark Palmerino

OPTIONS for *cat*
(adapted from FOR:PRO Programmer's Manual)

-n Causes the output lines to be numbered sequentially from 1. Giving **-b** with **-n** causes numbers to be omitted from blank lines.

-s Causes the output to be single spaced.

-u Causes the output to be completely unbuffered.

-v Causes non-printing characters to be printed in a visible way. Control characters print like ^X for control-x. A **-e** option can be given with **-v** and causes the ends of lines to be followed by the character '$'; the **-t** option with **-v** causes tabs to be printed as ^I.

OPTIONS for *more*
(adapted from FOR:PRO Programmer's Manual)

-c Draws the page starting from top of terminal. This usually makes it easier to read as the page is being drawn. This option will work only if the terminal has clear to end of line capability.

-d This causes *more* to print "Hit space to continue, Cancel to abort" at the bottom of each screen. This is especially useful for unsophisticated users.

-f Count logical lines rather than screen lines. If there is a long line that is folded, *more* may count it as more than one line. This option is especially useful for output that contains nonprinting characters.

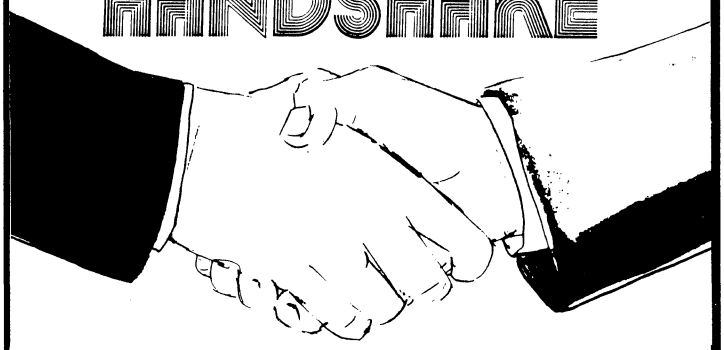
-l Do not treat ^L (a form feed character) in a special manner. Normally, *more* will pause at this point even if the screen has not filled up.

-n n is an integer which adjusts what *more* thinks is the size of the screen.

-s This squeezes out multiple blank lines. Thus, if you have 5 blank lines in a row, this option will cause more to display only one blank line. Very useful for maximizing the amount of readable information on the screen.

-u Suppress *more's* normal way of handling underline or standout escape sequences.

SST DATA, INC. found a Data Communication Solution for ILLINOIS BELL with a HANDSHAKE™



Illinois Bell's Distribution Services Department needed easy access to systems in the company's headquarters. After reviewing data communication software in the marketplace, Illinois Bell, an Ameritech company, chose HANDSHAKE ITE, interactive terminal emulators, and HANDSHAKE V, VT 100 terminal emulators, for general communications between 36 department locations with Fortune microcomputers in offices throughout the stat of Illinois and corporate headquarters in Chicago. As needed, each of the offices uses HANDSHAKE software from SST Data, Inc. to dial in and log onto the in-house Office Automation and Time Sharing systems and for other general communication needs.

Meeting the criteria for general communications in a user-friendly package, HANDSHAKE was the solution for Illinois Bell.

For information on HANDSHAKE data communication solutions, contact:

SST DATA, INC.

4701 W. Schroeder Dr., #100
Milwaukee, WI 53223
Telephone (414) 355-6990
Telex: 590597 (SST DATA INC)
EasyLink 62792652

Dear Fortune Owner:

As you have probably read in */u/fortune news,* Fortune Systems SX models are significantly more powerful and faster than the PS and XP models. If you have a PS or XP model system, you should take advantage of our special promotion on SX upgrades. For a limited time, and while our supplies last, we will upgrade your system, including the new SX Cabinet at no additional charge for the cabinet. With this special offer, you may choose any combination of the following upgrades and/or expansions:

- *SX Processor with Current Release of Multi-User For:Pro*
- *60 MB Internal Streaming Tape*
- *30 MB, 45 MB, 70MB High Speed Hard Disk*
- *6-Port Intelligent Asynchronous Controller*
- *Upgrade to Current Release of all Fortune Software*
- *Fortune:Windows (Included at NO Charge)*

Additionally, we can perform these upgrades within 24 hours of receiving your system. In fact, if you ship your system overnight on a Friday for Saturday delivery, we will have it back to you on Monday morning, all ready to plug in and run; not losing the use of your system for a single work day. And, we will issue you a full 90-day warranty on the entire system.

JFM provides services to Fortune Systems owners nationwide either direct or through our authorized Affiliated Dealers. If you haven't used our services, I invite you to discover JFM; you will be glad you did.

Sincerely,

Morris Frydman
President

Talk to us, then talk to our clients

JFM BUSINESS SYSTEMS
The Merchandise Mart
Lobby Suite 144
Chicago, Illinois 60654
Call (312) 644-0870

Authorized Fortune Systems Master Dealer

Two *NEW* Free Software Diskettes

This month we are introducing two new diskettes. We have not been idle in our constant quest for useful and fun software for the Fortune 32:16. Actually, our problem is not that it is a difficult task to find such software but rather that we have too much. This month we introduce two new software diskettes and will introduce at least one more next month.

DISK 5 DISK 6

NEW VERSION OF KERMIT CALLED C-KERMIT

One of the new introductions is a diskette that contains the latest and *best* version of **KERMIT**. As we have explained in the past, **Kermit** is a terminal emulator and file transfer program. This new version of **Kermit** is interactive and has many new features. One of the most useful features is the combination of **command completion** and **command-line help**. For example, **command completion** means that if I want to execute the command "Connect", I can simply type "co" and then a space character and **C-Kermit** will know I mean Connect. **Command-line help** means that I can type a question mark (?) wherever I want on the **C-Kermit** command line and **C-Kermit** will respond by telling me what options I have at this point. Other features include:

- Extensive On-line Help
- Auto Dials the Phone
- Can act as a File Server
- Automatic Log-on Procedures
- Records Your Session into a File
- Information about Effective Transfer Rates
- Do UNIX Commands From Within Kermit

PERLMAN'S UNIX|STAT

This diskette contains a collection of programs written by Gary Perlman that perform a wide range of statistical analyses. The capabilities range from basic descriptive statistics (like calculating means, medians, standard deviations and frequency histograms) through measures of association between two variables (like pearson correlation, contigency tables and Oneway Analysis of Variance) to multivariate statistics (like regression and N-way Analysis of Variance including repeated measures). These programs are very easy to use and integrate nicely with UNIX. Included is a program which makes data extraction and transformations very easy.

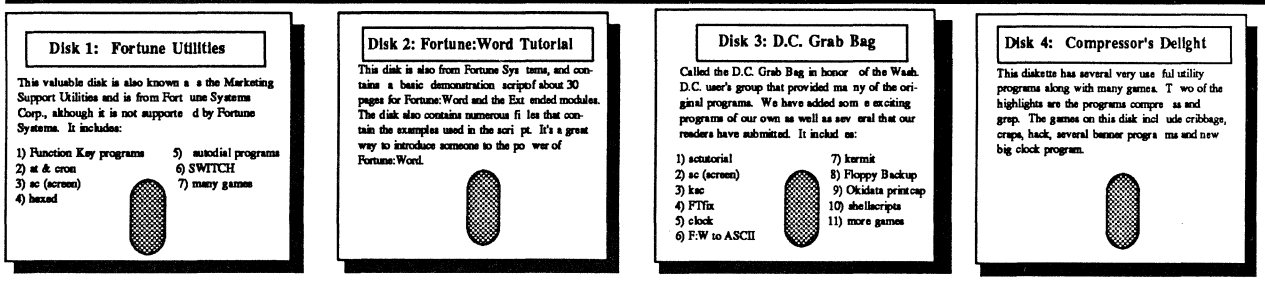
Data Manipulation Programs

abut	join data file
colex	column extraction
dm	column oriented data manipulator
maketrix	create matrix type file from free format file
perm	randomly permute lines
probdist	probability distribution function
reverse	reverse lines and characters
series	generate a series of numbers
transpose	transposes matrix type file
validata	verify data file consistency

Analysis Programs

anova	multi-factor analysis of variance
calc	interactive calculator
contab	contigency tables and chi-square
desc	descriptions, histograms, frequency tables
dprime	signal detection d' and beta calculations
oneway	oneway anova and error bar plots
pair	paired data statistics, regressions, plots
regress	multivariate linear regression
stats	summary statistics
ts	time series analysis and plots

And don't forget our other diskettes



Special Offer: Order all six and get one free!

To order your diskette(s), simply send $10, *per diskette*, to The Cambridge Consortium, Inc., 20A Prescott Street, Suite 28, Cambridge, MA 02138. Please make your check payable to The Cambridge Consortium, Inc.

Basic, continued from page 1

normally supports its dealers only, but support should trickle down nicely to all of the end users.

More important than support, I cannot imagine that Concept Omega would fail to port some of their other products, such as **CONCEPT-R** and **IDOL II**, to the Fortune. After all, they are primarily a software manufacturer. And any big business transaction that results in more software becoming available for the Fortune is, in my humble opinion, just ducky.

I bring all of this up now because Concept Omega is holding their annual Dealer's Conference in San Diego on September 10-12, 1986. Yours truly and company will attend. (If any of you will be there, please make an effort to look me up. I am the balding guy with the beard.) I should have all of the facts in order once I return, and I will address this subject again in the first available issue of **/u/fortune** *news*. Now, do I see a hand raised in the back of the room?

Question: I am thinking of buying one of the new 6-port COMM A boards, but I am concerned about the ability of my Business Applications to handle more active terminals. Isn't it true that you are allowed only seven terminals in BAS at one time?

Answer: Eight in the original design (T0 through T7). But like almost everything else, this is alterable. You must do several things, the most important of which is setting up additional ipl files through UNIX. Here's how we create an ipl file for terminal eight (T8). You must follow this proceedure for each additional terminal you wish to define:

First, log onto the system with the **root** account. From the prompts, type in the following commands (<CR> means hit the RETURN key):

```
# cd /b/ipls          <CR>
# cp iplT7 iplT8      <CR>
# ed iplT8            <CR>
```

At this time, you will see a number printed on the screen and lose your # prompt. Don't panic yet. The following commands which you will enter apply to pure, unaltered ipl files. If you at any time do not see the same response on your computer which is shown here, skip to the ".<CR>" command and do not try to edit the files unless you are totally familiar with the UNIX text editor. With that in mind, let's continue:

```
8p                    <CR>
system responds: DEV T7,7,3216,,,,,tty

s/T7/T8/p             <CR>
system responds: DEV T8,7,3216,,,,,tty

15p                   <CR>
system responds: IPL 1,2,T7,CSYST0

s/T7/T8/p             <CR>
system responds: IPL 1,2,T8,CSYST0

. (That's a period)   <CR>
w                     <CR>
system responds with a number

q                     <CR>
```

You should now have your # prompt back. This procedure expands the number of terminals available for Business Applications only. If you wish to expand **IDOL**, Business Training Developmental **BASIC** (Languages off the Global) or Business Surveys, you must make similar alterations to the iipl, tipl dipl and sipl files respectively. In these cases, the line numbers to be altered with the editor will be different. When you are done, use **CTRL-D** to exit UNIX (hold down the gray CTRL key and type the letter D).

Next you will have to make an adjustment to the **BASIC** interpreter (a file in /usr/bin called, appropriately, **BASIC**). Log in as **root** and do the following:

```
# cd /usr/bin          <CR>
# ed BASIC             <CR>
```

The system responds with a number and you lose the prompt.

```
9p                     <CR>
System responds:       [0-7])

s/7/9/p                <CR>
System responds:       [0-9])

.  (Period, again)     <CR>
w                      <CR>
q                      <CR>
```

Back at the # prompt use, **CTRL-D** to exit. You will have to make similar adjustments to **IBASIC**, **DBASIC** and **SBASIC** if you wish to use the new terminals with other **BASIC** global options. Again, the line numbers to edit may vary.

Don't leave yet, we're not quite finished. There exist in your Business Applications certain files which are terminal slaves and we must set them up for the new terminals. Go ahead and log onto the computer with your normal account and go into B1 from the global menu. Use an unrestricted operator code (such as DOL) to enter, and at the menu enter UTL <CR>. At the **BUSINESS BASIC UTILITIES MENU**

See Basic, Page 11

Basic, continued from page 10

enter number 13 **(DEFINE FILE AREA)** and <CR>. Here are your responses to the questions asked:

```
ENTER FILE TYPE (I=IND, D=DIR, S=SRT, P=PGM): Enter I and <CR>
RECORD SIZE       : Enter 1024 and <CR>
NUMBER RECORDS    : Enter 1 and <CR>
DISK NUMBER       : Enter 0 and <CR>
FILE NAME         : Enter *REPT8 and <CR>

FILE '*REPT8' DEFINED ON DISK '0', 'RETURN' TO CONTINUE: <CR>
```

Repeat the above proceedure for ***REPT9,** etc., if necessary. Then, at the same screen setup respond as follows:

```
ENTER FILE TYPE (I=IND, D=DIR, S=SRT, P=PGM): Enter P and <CR>
NUMBER OF PAGES : Enter 1 and <CR>
DISK NUMBER     : Enter 0 and <CR>
FILE NAME       : Enter CUTFT8 and <CR>

FILE 'CUTFT8' DEFINED ON DISK '0', 'RETURN' TO CONTINUE: <CR>
```

Again, repeat as needed for other terminals. You may exit the program as well as the utilities menu by using the '**F4**' key. There is one more series of slave files called ***SRT***n (n is the terminal number), but it is not necessary to set these up since they are defined as needed by the programs.

I must caution you at this point that adding terminal numbers above 9 may be hazardous to your system. Some **BAS** programs, such as those handling operator statistics and AR Cash Receipts Entry, use the terminal ID number as a key field for certain files, and you will probably need to have some serious frontal lobotomy work performed on the software to handle them. But, if you are the adventurous type, go ahead and try.

If you find that your version of the Business Utilities will not define files for ***REPT10** and/or **CUTFT10** because of the length of the names, you will have to define them using the **INDEXED** and **SAVE** commands available in Business BASIC. From the > prompt (selection 20 from the Utilities Menu) type the following:

```
>INDEXED "*REPT10",1,1024,0,0      <CR>
>SAVE "CUTFT10",1,0,0              <CR>
>RUN "DOL"                         <CR>
```

Once you have completed all of the above, you are ready to run your extra terminals in Business BASIC.

UNISOFT INC.

Experts in Computer Consulting - Sales - Service

Presents

FORTUNE SYSTEMS HELP

We Provide:

- Expert Trouble-shooting
- Years of programming and accounting experience
- *Free Initial Consultation*
- Training Available for:
  - * BASIC/IDOL/BAS and Multiplan
  - * UNIX and Ext. Fortune:Word
- Hardware Repairs & Upgrades
- FORTUNE trained experts at a cost that will surprise you!
- *Trade-Ins Welcome*
- Specializing in:
  - * Hard Disk Trade-In
  - * Hard Disk Upgrades

Authorized Fortune Master Dealer

Authorized Fortune Systems Repair Depot

CALL: (303) 671-8600 EMERGENCY HELP: (303) 691-HELP

UNISOFT INC. 14100 East Jewell Avenue Suite 27 Aurora, Colorado 80012

Glossary, continued from page 3

```
entry t /*entry to open a window and call
         up the tickler list document to
         insert a tickler entry*/
{
command "w" execute
"/u/office/tickler/ticklelist" execute
call display(false)
goto"e"
insert
      merge "date" MERGE return
      merge "date2" MERGE return
      merge "clnm" MERGE return
      merge "file#" MERGE return
      merge "atty" MERGE return
      merge "re" MERGE return
      merge MERGE return
execute
up(7)
right(6)
call display(true)
}
```

Entry t first uses the **command "w"** command to open the window; and the document to be opened is then specified. This command opens a full screen window, and displays the tickler file. Next, the line **call display(false)** is used to turn off screen updates while the glossary is executing. (The final screen will be displayed at the completion of the entry). **goto"e"** sends us down to the end of the document, where the proper **Records Processing** labels for each field are inserted. Note the use of lowercase and uppercase for the word **merge**. The uppercase is the equivalent of a **SHIFT merge**. The **execute** command completes the **insert** command after the "merges". **up(7)** sends the cursor up seven rows to the first merge row and the **right(6)** moves the cursor six columns to the right where data entry will begin. Finally, the **call display(true)** turns the display back on.

The operator then enters the tickler information on each line. The easiest way to move down one line at a time is to use the **down-arrow-key.** This will drop you right to the end of the line below. Note the two date fields. The first entry is in the following format: **yrmoda**, e.g. **860715** would be July 15, 1986. This is necessary because, when the dates are sorted, they are simply sorted as strings of characters. If you put the date in the normal format of 07/15/86, it would come after 01/12/87, because 01 sorts before 07. This is a particular problem in the latter part of the year where 12's and 01's are easily confused. The second date line is in a more intelligible format as July 15. This will be the text that is actually displayed on the tickler printout. The other fields are particular to Mr. Fellman's office, although they could easily be adapted for any use. He has fields for client name, file #, attorney, and the reminder text.

If the operator wants to add more than one reminder while they are in the tickler file, they use entry T. It is very similar to the above entry, except that there is no longer a need to open a new window. Here it is:

```
entry T /*used to add another record for
         multiple tickler entries while
         using entry t */
{
call display(false)
insert
      merge "date" MERGE return
      merge "date2" MERGE return
      merge "clnm" MERGE return
      merge "file#" MERGE return
      merge "atty" MERGE return
      merge "re" MERGE return
      merge MERGE return
execute
call display(true)
}
```

Finally, once all entries are completed, you save the document using the normal CANCEL/DEL EXECUTE sequence of keys, and presto, you're right back in your original document.

Printing the Tickler List

In order to print the Tickler List, the **Records Processing** functions of **Fortune:Word** are used. These were described in the last two issues, so the mechanics of **Records Processing** won't be explained in detail here. Once you have selected **Records Processing** (either through the **Advance Features** of the **Fortune:Word** Menu or by using **Command rmr**) enter /u/office/tickler/ticklelist as the **List Document**. For the format document, you enter a file called /u/office/tickler/tickleform which contains the following:

```
                    TICKLE
                   <date1>

DATE          FILE NAME                    REMARKS


<repeat(35)
<date2>       <clnm> <re>
>
```

This file places the heading information at the top of the page. Note that the < and > signs are actually

See Glossary, Page 13

Glossary, continued from page 12

generated by the **merge (f6)** and **SHIFT merge** keys. **<date1>** converts the system date into a normal English version, e.g. July 15, 1986. **<date2>** does the same thing without the year included. The **repeat(35)** places 35 ticklers down the page. The final **merge** symbol is the closing pair of the open **merge** on the **repeat**.

Presumably, if you have several people using the tickler file, you would want to sort by person (in our example, **atty**) and by **date**. It's conceivable that you might want to sort by client and by date also so that the senior partner could review the schedules of his subordinates. To extract all of the tickles for **mjf**, you would use a control entry like the one below:

```
entry m
{
select_record
{if((<atty>=="mjf")&(<date>>=860701&<date><=8607
31))
save record}
sort <date>
}
```

This entry selects all tickler records where the **atty** name matches "**mjf**" and the **date** is greater than or equal to (>=) July 1st and the **date** is less than or equal to (<=) July 31st. Once the records are selected, they are sorted by date. Enter this entry into /u/office/ticker/ticklegl or another glossary, and supply the name of the glossary when asked for the control document and `entry m` when asked for the entry. If you have several people you are tickling, you will want to have several entries like `entry m` with different initials in each one.

Mark Fellman has some advice about output documents. He suggests that it is best to select the **Output document only** option from **Records Processing**. He names the output document something like `mf.jul15` so that it is easy to determine who the ticklelist is for and when it was run. Once you have created the output document, be sure to edit it and take out any extra page breaks that may have accumulated at the end of the file.

Housekeeping

There are several little matters that need to be addressed to keep the tickler running smoothly. The first is the use of page breaks. The **Fortune:Word** program is page oriented. This means that you must set page breaks in all documents or you will have problems with the system. Consequently, it is important that you put page breaks into the ticklelist document. The key here is to place page breaks only between records and never in the middle of one. Placing a page break in the middle of a record will result in a page break showing up in the output document, but not necessarily where you would want one. It is not a fatal error, because it can be edited out, but it does add one more step to the process.

The second matter deals with managing the size of the ticklelist document. Mr. Fellman recommends that once a month a copy of the ticklelist be made. This can be done by using the `F11` function key while in an **index** of the library where you are keeping the tickler components. He generally names the copy of the ticklelist, `ticklelist.cpy`. Once you have done this, go to **Records Processing** and enter `ticklelist` in response to the query **list document name**. Then leave the **format document** query blank, enter `ticklegl` for the control document and specify `entry x`. (The contents of `entry x` follow this paragraph.) Then, name the output documents `ticklelist.s`. This will result in a ticklelist sorted by date in descending order. You can then edit out those tickler entries that are no longer relevant. After you have done this, you can delete the original `ticklelist` document and rename the `ticklelist.s` document `ticklelist` by using the `F12` function key while in the index. Keep the `ticklelist.cpy` document as a backup just in case something goes wrong.

This is what you should enter as `entry x`:

```
 entry x
{
sort descending<date>
}
```

Some Possible Enhancements

As we worked on this article, a couple of possible enhancements came to mind. First of all, by using a test like `if(line > 50)` you could automatically check the position on the page in the `ticklelist` and insert a page break if necessary. Dick Dow did this in his entry in last month's newsletter. (See Vol. 3, No. 6, page 15.) If you wanted to get fancy, you could have a routine which converts the date from one format to another. You could either enter the date in the **yrmoda** format and have it converted to normal (July 15, 1986) format, or you could have it go the other way. See the Glossary Entry in Vol. 2, No. 4, page 4 for some hints on how to do this kind of conversion.

Josh Lobel

/u/help

*Question: I have tried to use the **kmenu** program on your software diskette. When I do, however, I keep getting the message "cannot open /dev/cul##". What am I doing wrong?*

Answer: **Kmenu** is a front end for **Kermit** which is a terminal emulator and file transfer program. These programs can be found on our **DC Grab Bag diskette**. The above question was left by one of our subscribers on the **/u/fortune** Electronic Bulletin Board. We publish our response to his question.

The above problem can happen for a couple of reasons.

First of all, **kmenu** requires a device with the name *cul#* as in */dev/cul2*, to operate. This can be created in one of two ways. If you are using a port only for calling out (a disabled port set up for communications from the Global Menu), you can simply make a linked copy of the tty device by typing:

```
cp -l /dev/tty02 /dev/cul2
```

You can substitute the number of whatever port you are using for your modem. For example, if you are using port 4, instead of port 2 as above, then specify:

```
cp -l /dev/tty04 /dev/cul4
```

If you are using a port to call out and to have people log in, then you will have to follow the instructions given in **Volume 2 Number 7** of the newsletter. Basically this involves using the **mknod** command to create the new device and setting up an **rc** file to establish the correct protocols when the computer is booted.

Be sure that the permissions on this device are correct -- they should be **rw-rw-rw**. This is accomplished by typing:

```
chmod 666 /dev/cul2
```

The specific **cul** device that you have is analogous to port 2, so your modem should be plugged in there. If the name is not **cul2**, change it so it is. When you use **kmenu**, you will tell it to use line 2. (This is selected from the options menu.)

Finally, you must have the **rc** file in the **/m/rc** directory to setup port 2. A sample of this file is in the **lib/communications** directory on the bulletin board (as well as in **Volume 2 Number 7**). Make sure you have it going to the correct port also.

I suggest you try **kmenu** first using a regular disabled

See **/u/help**, Page 21

Fortune Users:

JUST POSSIBLY
THE CHEAPEST PHONE CALL
YOU'VE EVER MADE

- 70 MB disk drives
- 10 MB REMOVABLE disk drives
- Expansion cabinets
- Laser Printers
  8 thru 26 Pages Per Minute
  (no extra software required!)

- Genicom Impact printers
- Multi-tech 2400 Baud modems

Shop around... then call us.

Robbins & Associates

(206) 771-2286

* * * * * * * * * *

Call us COLLECT today for a quote

For The Best

B.A.S and IDOL SUPPORT AND TRAINING

Call

Jackie Biggs

at

Executive Data Services, Inc.
415-359-7644

We Support Modified Software !

Lawyers, continued from page 1

to tape backup. But they cited the high cost of this method as the primary reason for putting off such a move. (We will publish an article on backup methods in the near future.)

Most people use **QUME** or **NEC** printers. A typical letter quality printer like the **NEC 3510** is widely used with varying degrees of satisfaction. Some users find it excellent; some have run into a fair number of mechanical problems with it. **M. Arthur Auslander**, a New York patent and copyright attorney, noted that his NEC kept burning out its boards until he had the machine grounded. Laser jet printers were obviously a hit with those users who had purchased one; the speed and silence are attractive features. The **HP Laserjet™** seems to be the most popular model. **Kathy Hendrickson,** who manages a Minnesota law firm, has five terminals hooked to a single NEC Spinwriter™ with a dual cartridge bin feeder. Any of these terminals can send to either of the bins, eliminating the need to manually "feed" the printer when the user wants to insert a different kind of paper or letterhead.

Software:

Law offices have many of the same needs as other Fortune users -- powerful word processing and business management. In a law office, business management primarily consists of such functions as **Time and Billing, Client Tracking**, etc.

It should be no surprise that most attorneys interviewed noted that they primarily use their Fortune for word processing. **Fortune:Word** is one of the main reasons that Fortune hardware was initially chosen. **Fortune:Word** gets a tremendous rating for processing documents and correspondence. Users who were not initially familiar with word processing felt that it was a simple package to learn and to teach others, like newly hired office staff. Some of **Fortune:Word**'s uses are fairly conventional, in the sense that it doubles as a quick way to type in and print out single documents with varying formats and page lengths.

Using **Fortune:Word** to process the long documents typical of law offices is particularly beneficial. **Robert Resnick**, a Boston attorney, cites secretarial time saved with the system over manual systems (like typing) as a way of cutting office costs. **Jeffrey Weiner** of New Jersey adds that the productivity of the office has increased dramatically while staff size remains stable.

To further cut down on time spent, **Fortune:Word** glossaries offer an efficient way of repeatedly merging existing documents (long or short) with other information. Information such as client and attorney addresses is stored in glossary entries in advance and inserted into existing documents at the necessary time.

Other lawyers use glossaries to store longer documents such as form letters or repeated paragraphs that can be rearranged at will. Though some offices have used the glossary system, others have not had the time to learn its use. (*editor's note: Glossaries are really quite easy to use -- take a look at our Glossary Entry columns to learn the basics.*) One lawyer would have made more use of **Fortune:Word** glossaries if he could globally attach it to all of his libraries. (*Editors' Note: This is possible, and we will describe how to do it in a coming Glossary Entry column*).

Some legal offices that have a great deal of repetitious boilerplate text make extensive use of glossaries. Please see the accompanying article about the Prosecuting Attorneys' offices in Michigan.

Mark Fellman, a Minneapolis attorney specializing in labor law, has made good use of **Fortune:Word**'s glossary features by creating a **tickler** program. This is used to remind all of the attorneys of delivery dates for all their commitments. The tickler file contains information about the client, attorney, reminder, and date due. Periodically the operator asks the program to sort forward a certain number of days (say, either every day between today and 30 days from now). The tickler is accessed by attorney, date on which a reminder is to be given, and type of case. A reminder list is generated and distributed to the parties involved. If an operator is working in **Fortune:Word** and wants to add a new entry to the tickler document (say, a new client number to search for or an upcoming date by which time an important document must be mailed), he or she executes a glossary entry which automatically opens up a **Fortune:Word** window with the tickler file displayed. The operator is automatically prompted to enter the appropriate information. When he or she is through, the tickler file is closed and the operator returns to the original work. This month's **Glossary Entry** features **Mark Fellman's** tickler program for those who are interested in using it.

Though most people are satisfied with the word processing component of their system, they have not found it as easy to automate other office functions, such as time and billing accounting. To assist them in processing a client's bill, some users have opted to buy one of the commercial packages. In such cases, the packages cited, like the **Time and Financial Management System for Lawyers (TFM)** by **Gill & Piette, Inc.** have proven quite useful. (Typically, however, this is an altogether too-expensive option for most small firms. Most third party vendors advertising in the **Fortune Software Catalog** retail their time and billing software at

See Lawyers, Page 16

Lawyers, continued from page 15

between $1,500 and $10,000, depending on the features and add-on modules available).

To circumvent this problem, some offices have developed their own time and billing packages. **Mark Fellman** has programmed what he describes as a rough T&B package into the system, using **Fortune:Word's Record Processing** module. The logic he uses to construct the program is similar to that employed in the tickler program. This is how he describes the program. The date of the service is entered twice (once as month/day/year and once as year/month/day). The second format is used for sorting purposes only -- otherwise January entries from all years would be sorted before February. By putting the year first, entries are correctly kept within the proper order. Then, in one entry the attorney or operator keys in the attorney's name, the client file number that is assigned to a particular client, the description of what has been done for the client, and the amount of time spent on the case. **Records Processing** is used to sort by client file number and attorney; and it produces the amount of time spent on the case. At the same time, the program creates a **Fortune:Word** file to generate the bill. **Mark Fellman** notes that, because the program isn't quite perfect, the operator must at this point manually edit the bill, but at least a significant portion of the work has already been done.

Because he sends out few monthly bills (15-20), **Mark Fellman** says this system is adequate, but if he were to send 50-100 bills it would become too cumbersome. (In a similar way, **William Stack** of Atlanta, has "jury rigged" a time and billing system using Multiplan. The idea is basically the same. Enter the information about a lawyer's time as it occurs and then to sort it by client to prepare a bill.)

M. Arthur Auslander, a patent lawyer in New York, has worked with a programmer to create his own management system using the **Conetic Systems** tools, including both Time and Billing and client retrieval. He finds that it works quite well. In this system, he stores client information, such as name, address, client i.d. #, and work done for the client per day or month. Given any bit of this information, he can access the client's file. The same information is also used for time and billing functions and in addition can be used to remind himself of key dates, documents, etc.

Typically, he uses three **Fortune:Windows** simultaneously with one window open for **Fortune:Word**, one for his time entries, and one for information retrieval. This allows him to quickly shift from one application to another. If a client calls with a question about the status of his work, **Mr. Auslander** can quickly make an time entry for billing purposes and, an instant later, run a query about that client's folder.

Other users have also successfully programmed their own software to take advantage of large data sets in database management systems. **Phil Bertenthal** of **Contra Costa Legal Services** of California obtained a software package programmed by a friend to help manage cases in **dBaseII**. (Note: **dBase II** normally runs on **CPM** or **MSDOS** machines. However, **Touchstone Corporation** markets a program called **MIMIC** which allows the Fortune to run **CPM** programs. Although this may not be the most efficient way to use the power of Fortune computers, it is sometimes convenient given the large library of applications that have been written using **dBase**.) In a recent case he needed to track 7,000 documents in the database, and this program (called the "**Case Management Information System**") successfully tracked and identified the method for accessing each of those documents.

Another **Time & Billing** package we are aware of is produced by **Clockwork Systems**. This system uses the **Progress** language to create what sounds like a powerful system. This program was devised by some of the same people who created the

See Lawyers, Page 18

VANTINE COMPUTER SERVICES

We Are Not The
Biggest **Biggest** **Biggest**

Nor The
Cheapest **Cheapest** **Cheapest**

We Simply
Care **Care** **Care**

VCS is an Authorized Fortune Systems Dealer. We understand and support the products we sell.

We provide sales, upgrades, service and support to our Fortune Clients from Maine to California.

We don't ask you to believe us. We want you to believe our clients. Call for references today.

Vantine Computer Services can provide the solution for you!

VANTINE COMPUTER SERVICES
9 Priest Road, Sudbury, MA 01776
(617) 443-8412

Authorized Fortune Dealer

JONES BUSINESS AUTOMATION CENTER

a FORTUNE SYSTEMS MASTER DEALER

and the JBAC AFFILIATE DEALER NETWORK
are
ORGANIZED TO SUPPORT FORTUNE END-USERS
with
OUTSTANDING PROFESSIONAL SYSTEMS SUPPORT
at REASONABLE PRICES

* BAS/IDOL SUPPORT
* FORTUNE:WORD
* MULTIPLAN
* CUSTOM DEVELOPMENT
* COMMUNICATIONS
* HARDWARE UPGRADES
* MAINTENANCE
* LASER PRINTERS
* OCR PAGE READERS
* COMPUTER NETWORKS

Inquire about: **The JBAC "PERFORMANCE SOLUTION"**
Upgrading your present system to
a new SUPER PERFORMANCE SX-70
the Price Performance Leader

The JBAC Affiliate Dealer Network		
ADMS - Advertising Data Mgmt 4173 First Avenue Tucker, GA 30084 (404) 938-8100	**ATOM - Automated Time/Office** 126 Fieldstone Drive Ringwood, NJ 07456 (201) 962-6497	**MicroAge Computer Store** 1939 Poplar Avenue Memphis, TN 38104 (901) 722-8280
BRINC Business Systems 1662 Savannah Highway Charleston, SC 29407 (803) 763-0599	**CompuSystems of Georgia** 2200 Northlake Parkway Tucker, GA 30084 (404) 491-8004	**DataLogix** 3000 Dundee Road Northbrook, IL (312) 291-9225
DataLogix 200 Central Park Avenue Hartsdale, NY 10530 (914) 997-1627	**Future Systems** 2525 Washington Road Augusta, GA 30904 (404) 737-8313	**Software Sciences** 3100 Evans Avenue Fort Myers, FL 33901 (813) 334-1646
DRS Enterprises 127 E. Murphy Street Madison, NC 27025 (919) 427-0864	**Synergistic Business Systems** 4300 Paper Mill Road Knoxville, TN 37909 (615) 584-0211	**MultiUser Systems, Inc** 10818 N. W. 6th Court Miami, FL 33168 (305) 758-2007

JONES BUSINESS AUTOMATION CENTER
834 Tyvola Road, Suite 112
Charlotte, NC 28210
(704) 525-0582

Lawyers, continued from page 16

Informatics General system that is often used in large law offices. A demonstration version of the program is available from **Clockwork Systems** for $295.00.

If you are currently using a package that we have not discussed here, we welcome your comments and will publish them in a future article. If you are considering buying a package, we encourage you to either get a demonstration program or get several references from the vendor before making a commitment.

Here are names and addresses of the vendors we know about who supply Legal software for the Fortune:

APPGEN Professional Time Reporting
Software Express, Inc., 2529 Briarpark Boulevard, Suite 700, Houston, TX 77042, **(800) 231-0062.**

Attorney Business Management System
TOM Software, PO Box 66596, 127 SW 156th St, Seattle, WA 98166 **(206) 246-7022**

Case Managment Systems and Legal Accounting
Guardian Automated Systems, 420 Main Street, Suite 600, Buffalo, NY 14202 **(716) 842-6410**

CLMS: Legal Time Management Package
Clockwork Systems, 3696 Eaglerrock Drive, Doraville, GA 30340 **(404) 934-7662**

Legal Services System
PAC Corporation, 1617 St. Mark's Plaza, Suite A, Stockton, CA 95207 **(209) 951-8697**

The Law Office Manager
Integrated Circuit System, 333 East Ontario, Chicago, IL 60611 **(312) 642-0707**

Time & Financial Management System for Lawyers
Gill & Piette, 908 Pennsylvania Avenue, SE, Washington, D.C. 20003 **(202) 546-6170**

Time Billing
Computer Consulting, Suite 206, 7002 Moody Street, La Palma, CA 90623 **(213) 865-9533**

Karen Parrish

Equipment *Service* **Supplies**
Accessories **Training** *Updates*

PIONEER APPLICATIONS
49 WALNUT STREET
WELLESLEY, MA 02181

Pioneer Applications can supply *all* of your needs. As the FORTUNE DEALER with the largest installed base in New England, we can handle any situation requiring Fortune expertise.

SERVICING FORTUNE USERS FOR OVER 3 YEARS

If it's new equipment, service, support, software, consultation or just a friendly voice, Pioneer will respond immediately. **Spring special:** 70 MB hard drives, *call now.*

CONTACT
ALLISON BLACK at (617) 237-5478 or
VICTORIA COLE at (203) 236-2373

BACKUP AND RESTORE UTILITY

PROTECTION MADE EASY!

- o Back up all or selected user directories
- o Restore all or selected user directories or files
- o Format diskettes
- o Review backup volumes
- o Menu-driven, easily installed
- o Complete documentation

Also copies the files necessary to preserve your system configuration--a tremendous help if your system ever needs to be cold booted!

$150.00	Backup/Restore Utility
$200.00	Utility + 2 boxes 5 1/4" 96 TPI high-quality diskettes

For more information, call CTD at (313) 540-1282.

To order, make check payable to *Computer Trade Development Corporation.* Send to:

BACKUP/RESTORE UTILITY
Computer Trade Development Corporation
Suite 1386
Birmingham, MI 48010

Company:_____
Contact:_____
Address:_____
City:_____
State:_____ **Zip:**_____

PROSECUTING ATTORNEYS USE THE FORTUNE

The ability to access information quickly and efficiently recently paid off for the **Macomb County Prosecutor's Office**, when it was awarded **Best Office in the National Association of Counties for "Word Processing and Computerization."** Prior to using the Fortune, this office, like many others, relied on manual methods, like typing, for producing enormous numbers of documents, many of them slightly modified versions of one another.

Use of the Fortune and its software has taken hold in Prosecuting offices we contacted, with exciting applications for attorneys, their staff and judges.

The combined use of the *Informix* database and *Fortune:Word* glossaries makes it easy to process arrests in Michigan. Warrants are created accurately and quickly with the extensive use of Fortune computers.

John Goergen of the **Michigan Prosecuting Attorney's Council** has worked extensively with an **Informix** database and the Fortune to create several valuable programs for the Council and around 20 county prosecutor's offices in Michigan. One of the most valuable programs allows a prosecutor to create an arrest warrant for a particular person, in which a good deal of the document text is the same as would appear on other warrants, but in which certain information (such as date and location of violation, type of weapon or controlled substance used, etc.) is specific to that particular warrant. To use the program which creates a warrant for someone's arrest, an operator keys in the person's name and the date, and the program creates the beginnings of a **Fortune:Word** document with the appropriate opening warrant verbiage, the name and address of the defendant, etc.

Fortune:Word glossaries make the next steps in creating a particular warrant document relatively easy. In Michigan there are over 6,000 chargeable offenses; the Prosecutor's Council has keyed into the database as glossary entries about 500 of the most most common. Each entry, or crime committed, comes with its own "charge code" from the Warrants Manual and this is the code it is given for use in the glossary.

Any glossary entry can be accessed depending on the information wanted for the warrant, and the program begins filling in that information and the "boilerplate" information standard to each warrant. During this process, the operator may be required to manually edit some portions of the document, but this is apparently minimal. After this process is completed, a full warrant gets printed out. The same process applies to generating subpoenas.

The Council has also developed an "Issues" or Brief Bank for the appellate court system, which may also be of interest to individual law offices. A prosecuting attorney interested in examining certain kinds of cases may want to know the last time that particular issue or a related issue was argued before the court. The Issues Bank is a program which keeps track of the issues argued, before which judge, when and where. As with the charge codes for warrants, any issue can be queried according to an index code it has been assigned. Once the attorney queries a main category for the issue, he or she can examine major and minor subcategories for additional kinds of cases under the general issue heading. **Bernadette Battani**, a Systems Analyst for the Macomb County (MI) Prosecutor's Office gives an example of the system. An attorney wants to know about cases where a defendant has questioned the counsel he or she has been appointed. The attorney queries "counsel" and is given a main category for all counsel-related issues under which major and minor categories are listed. The attorney is interested in reading the cases where "effective assistance" has been questioned, and so queries just those cases in the major category. Under the minor category, the attorney still has several choices about which particular set of cases to examine, and chooses "at sentencing" to learn which of those defendants questioned their counsel at the time of sentencing.

Another interesting use of **Informix** and **Fortune:Word** is a program they call "Mailer". The Council has several membership lists, such as all members, members of the Board of Directors, etc. "Mailer" allows an operator to search for certain categories of members (i.e., all members who live in a certain section of the state). This search can occur along any of these parameters even after the address has been entered once. Informix creates list document which **Fortune:Word's Records Processing** then calls up.

Karen Parrish

SOFTWARE FOR THE FORTUNE 32:16

MULTIPLAN LINK - can pass data from BAS/IDOL/BASIC files to Multiplan and back. For example, it could be used to send data from any of the BAS files (Chart of Accounts, Customer Master, Inventory Master, etc). Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

RECORDS PROCESSING LINK - can pass data from BAS/IDOL/BASIC files to Records Processing and back. For example, it could be used to take customer address data from the Customer Master and will automatically make the Records Processing List document for mailouts. Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

TERMINAL/PRINTER LINK - provides capability to print Multiplan, and BAS/IDOL reports on a printer connected to a workstation, view any report on the terminal before printing, convert any report to Fortune:Word format; accumulate multiple BAS/IDOL reports before printing, backup reports onto diskette; copy or rename report files; print multiple copies of the report on either the primary and/or terminal printer. Also has option to print BAS/IDOL reports on HP LASER printer ($50 extra) **$195.00**

TELEPHONE LINK - on line telephone message, telephone directory and inter-office mailbox/memo system which can also be used as a simple calendar reminder system. Provides control, printing and viewing of on-line data for each user. **$295.00**

FINANCIAL LINK - stores BAS Income Statement and Balance Sheet data in files. Once stored, it can be sent to Multiplan or instantly printed. **$195.00**

MENU LINK - provides software and documentation for adding items to the Global Menu as well as a sub-menu for your use Includes backup script for user directories and BAS/IDOL/BASIC programs and data. **$95.00**

CALCULATION LINK - provides programs for amortization, depreciation, loan repayment, averaging (with graph), linear correlation, and breakeven analysis. **$95.00**

KOMPACT PERSONNEL ACCOUNTING - provides for data capture of personal, wage/salary, job, education, salary/wage, appraisal training, dental, medical and life insurance data. Includes over 80 pre-defined reports/worksheets. **$495.00**

CRT LINK - provides information for using other terminals on a Fortune System (WYSE, TV, IBM 3101). **$95.00**

AUTOMATIC REPORT LINK - provides capability to print numerous IDOL defined reports automatically based on frequency or application without having to select each from a different menu. You identify frequency/application codes. **$95.00**

TAPE LINK - provides one menu for all tape backup utilities. Provides daily, weekly, monthly and individual backup and restoration routines. Includes rewind, retension, and tape unit startup/stop utilities. User friendly. **$95.00**

PRINTER LINK - provides printcaps for Okidata 84, 2410, TI 855, and Anadex 6500. Includes shell scripts for changing pitch and LPI. Allows these printers to be used in word processing. **$95.00**

MODEM LINK - allows Hayes 1200 Smartmodem or compatible to be used on one port for both incoming and outgoing calls. **$95.00**

INSTALL LINK - provides documentation and software example for installing your product/menu through "S5". **$95.00**

REPORT LINK - modifies IDOL Report Generator Program to allow stacking of up to 8 fields in an IDOL defined report. Also corrects problem with displaying subtotals for all stacked fields. **$95.00**

WESTERN LINK - provides instructions on how to use your Fortune for sending/receiving telex messages using Western Union's Easylink. Can replace your telex! Fortune ITE software is required. **$95.00**

BACKUP LINK - provides menu selection for backing up /b (BAS/IDOL/BASIC), /u (Fortune:Word, Multiplan) and entire system onto diskette. Also has program to "unhang" a terminal that is hungup just by entering the terminal number; a program to restore one file from a multi-volume backup; and a selection to tell you which volume a backed up file is on. **$95.00**

UNI-KOMP 9803 Rocktree Houston, Texas 77040 (713) 937-7782

/u/help, continued from page 14

port, like your printer port. Just make the linked copy as described above (`cp -l /dev/tty01 /dev/cul1`). Then do the `chmod 666 /dev/cul1`. Then try **kmenu** on line 1. If that works, you can try the dual use method that I have referred to with the **mknod** command.

*Mike Eisen of Beacon Systems left a terrific suggestion for using **Fortune:Windows** in conjunction with our Bulletin Board. Of course, his suggestion is just as useful for sending files from your Fortune to ANY other electronic bulletin board system as well:*

Fortune:Windows makes uploading of shell scripts and other ASCII files to this BBS, an easy and economical matter.

First, prepare your notes on your own system, saved as straight text.

While in the same directory, run **dolphin** and log on to the BBS.

Invoke the note editor. If you have **screen**, go into insert mode. For **ed**, enter append mode.

Tell **Windows** that you want to '**Insert Data**', from a file (providing the saved file's name, as on your system), and to use '**Fortune:Word Line At a Time**" insertion.

You will see a perfect copy of your file type out on the screen in front of you.

When finished, tell your editor to exit and update. That's all there is to it. Obviously, if you could type continuously at 120 words per minute, this tip would be unnecessary.

Editor's Note: We'd like to reemphasize that this use of **Fortune:Windows** is an excellent way to both save money and cut down on aggravation. The main problem when using electronic bulletin boards long distance is that each stroke of the key board costs you money. Obviously, if you could do most of your work on your own machine (like composing questions or responses to questions) and then somehow transfer it to the bulletin board, then you would save time and money. Mike alludes to the fact that transfering information with **Fortune:Windows** would be equivalent to typing very fast. Actually, with a 1200 baud modem it would be like typing over 1400 words per minute. Even a lowly 300 baud modem delivers the equivalent of more than 300 words per minute (Calculation: 1200 baud = 120 characters per second which is 7200 characters per minute. Assume the average word is roughly 5 characters and you have 1440 words per minute. Consider further: If you type at a respectable 30 words per minute, then you would be able to type 1800 words in an hour. Using **Fortune:Windows** it would take less than 2 minutes to transfer this amount of words. Thus, 1/30th the time and 1/30th the cost - we'll let you figure out the savings...)

Panic in Fortune Town

I personally don't know very much about the hardware of the Fortune computer. So, when I see diagnostic messages on my screen I get scared. I bet you'll know what I felt like when I saw this one morning on my screen:

```
Parity regs: Low=0xE088 High=0x48C0
Int regs: PEND=0x7F PRES=0x2C
AT PC0x1A4E8 [0xEA000] panic: double
parity
```

Talk about panic? I took a deep breath and then came upon a, what seemed to me, reasonable solution: Go have a cup of coffee and hope the problem would go away. I came back and tried to reboot the system, but to no avail. Same error, same panicky feeling, same strange urge to have coffee. I didn't give in to the coffee urge this time but rather called my colleague, Josh Lobel, who you'll know as the editor of **/u/fortune** *news*. (Actually, I tried a few other things like booting the system from my cold boot floppies but they didn't work - nor did my little model of a Fortune with the pins sticking out of it). Josh suggested that the problem may be with one of the memory boards. So he thought I should take them out of the machine and put just two back. (At the time I had 4 256K boards in the machine). Then I tried to reboot the system. Voila! The system came up. I removed one of the boards and put another in and rebooted the system. It didn't come up, so I knew I found a bad memory board. I put the three memory boards back in the machine and it rebooted just fine - no error, no panic and now a legitimate desire for coffee.

Moral: Don't get too scared at ominous diagnostic messages on your Fortune. But as for me, I know just where the coffee is.

Mark Palmerino

(Editor's Note: Putting in or taking out memory boards is a fairly routine task. However, be aware that you do take a certain amount of risk whenever you take the top off of your Fortune. Never take the top off while your unit is plugged into an electrical outlet. And, as always, if you do damage while trying to fix a problem, it is your fault.)

News, continued from page 2

Olin comments that "the **UNIX** market is just beginning to grow and Fortune is not only financially sound, but it has a well-proven, solid multi-user microcomputer system, outstanding office automations software, and a team that is committed to giving VARs and OEMs the support and service they need. The firm's next generation of systems will further strengthen the company's position."

Prior to joining Fortune Systems, Olin held a number of key sales and marketing positions in the computer industry. These have included vice president of sales for Basic Four and vice president of marketing for Micro Five. His sales management experience includes positions with Qantal, UCC and Olivetti.

* * * * *

Fortune Forms an OEM Software Division

In a move that signals a new corporate direction, Jim Campbell announced the formation of an OEM software division and appointed Mary Susan Espy as vice president and general manager of the new division. The newly established group will be responsible for marketing Fortune's extensive software line, which until now has been available only on Fortune Hardware to integrators, distributors, and dealers. Campbell noted that there is a tremendous opportunity for Fortune, especially in the **UNIX**-based office automation software market, to leverage its comprehensive line of software products.

Espy, a five-year Fortune veteran, most recently served as the company's senior director of corporate development. She has also held the positions of director of sales operations and planning, assistant vice president of sales and marketing, senior manager of software applications development, and manager of business applications.

She pointed out that while the scientific/engineering market, the federal government, and the aerospace industry have been the driving forces behind **UNIX**, the operating system is gaining increasing popularity throughout the general MIS community. Fortune, with a software line that includes a **UNIX**-based operating system, office automation, languages, applications software, and communications products, plans to develop strategic relationships with the micro, mini, and mainframe computer manufacturers as well as system integrators.

"We are establishing an engineering staff to work out the technical issues and to support all software enhancements," said Espy. "We will also provide complete documentation for each software product to our manufacturer clients and either handle end-user support through an 800 number or train our customers to field end-user questions."

For more information about the OEM software division, call or write Mary Susan Espy, Fortune Systems Corporation, 300 Harbor Blvd., Belmont, CA 94002, (415) 593-9000.

* * * * *

Tyme/Fortune Office Automation Agreement

An agreement has been made which will permit Tyme Systems Ltd., a Canadian-based company, to sell Fortune Systems Corporation's **UNIX**-based multiuser, multitasking computer and office automation software to business and commercial users throughout Canada. This agreement should result in at least $5 million in sales over the next 12 months according to Lawrence Young, vice president and general manager of Tyme.

Young cited two reasons for Tyme's selection of Fortune's hardware and software for the Canadian business and commercial market. "First of all, Fortune already has a strong customer base." he commented. "The real deciding factor, though, is that Fortune is the only computer system manufacturer who is actually able to demonstrate and deliver office automation applications software that is written in both French and English. The bilingual software complies with the rules and regulations of our national and provincial governments."

Founded in 1973, Tyme Systems has more than 400 customers in Canada. The firm is located at 5490 Royal Mount Avenue, Montreal, Quebec H4P 187 (514) 341-6300.

Accutrend System Services

Serving the Great Northwest
Washington, Oregon, Idaho, Alaska

Sales, Service, Support,
Software and Maintenance

- **Office Automation**
- **Accounting Systems**
- **Communications**
- **Database**

Free monthly newsletter

Call or Write:
Accutrend System Services
P.O. Box 38
Puyallup, WA 98371
(206) 841-1992

Authorized Fortune Dealer

Fortune Users:

G.T. Hawes & Co.
An Authorized Fortune Master Dealer
Announces a Fortune Liquidation Sale

It's true. We're ever on the look out for *good buys* and we struck a proverbial FORTUNE! Two major users wanted out and we were there. We said, "HELLO." They said, "GOODBYE." And what a *GOOD BUY* it was! You guessed it, we're overstocked and must move the product. Take advantage of our prices because they were never lower.

CALL GEORGE AND SAY HELLO, SO I CAN SAY

GOODBYE!!!
NO REASONABLE OFFER REFUSED!
516-568-0044
Equipment Schedule Available on Request

P.S. As this ad was going to press another large user called... The price was so good we bought more... Call us and let us show you how we can expand your Fortune System the most economical way possible. We also sell parts for those users who perform their own system maintenance.

"If you don't see my name on your computer shipment you probably paid more than you should have."

G.T. HAWES & CO., Inc

475 W. Merrick Road
Valley Stream, NY 11580
516 568-0044 516 627-5556

PROGRAMMERS

The leading manufacturer of status display panels for the fire alarm and security industry is seeking programmers and junior programmers to develop new applications and refine existing programs.

Successful candidates will be experienced in IDOL/BAS and SMC Business Basic for Fortune and Basic 4 systems. Familiarity with UNIX and system set-up and maintenance is desirable.

Candidates interested in joining a growing team-oriented organization located in Falls Church, Virginia should write or call with resume and salary history, to:

WSA
308 Hillwood Avenue
Falls Church, Virginia 22046
703 241-0687

Expand your Fortune® without spending a fortune.

New equipment costs big bucks. The Good Fortune Trader sells fully tested burned-in equipment, refurbished to Fortune® standards at prices that won't make waves with your cash flow. Call or write today for a listing of our specials.

THE GOOD FORTUNE TRADER

60 DAY WARRANTY

a division of Computer Services of Atlanta

(404) 392-9938

1534 Dunwoody Village Parkway/Atlanta, Georgia 30338

Still working on last years financials?

Do you use accounting software on your Fortune? Everyone knows what the best word processor is for a Fortune computer, but when you want to keep track of your finances, the water becomes more cloudy. Is it best to use **BASIC, COBOL, Informix, Progress**, or another language to run a system. Do you use **BAS Applications, Conetics Books, Software Express**, etc.? Which is fastest, easiest, most powerful. Only you know for sure.

We'd like to write an article about accounting packages, but we need information from our readers. Please share your feelings about accounting with us. Write to us at the address below, or leave messages on our electronic bulletin board at (617) 648-1263.

/u/fortune *news*
20A Prescott Street
Suite 28
Cambridge, MA 02138

/u/fortune* *news...*

The Newsletter for Users of the Fortune 32:16 Computer **August 1986/Volume 3 Number 8**

The UNIX Directory

INTRODUCTION TO SHELL SCRIPTS

David E. Kloes is President of UNi-KOMP which is located in Houston, Texas. He provides UNIX seminars, writes software for the Fortune Computer and is vice-president of the Houston UNIXUser's Group. He is contributing independently to ***/u/fortune*** *news.*

In our never ending quest to use our Fortune systems more effectively, we cannot avoid the subject of shell scripts. A shell script is nothing more than a set of instructions to tell the computer what to do (a program). Thus it becomes another tool that will help us to manage and control our computer resources. Many of the programs in use on your Fortune system are shell scripts so that, in addition to the information provided in this series of articles, you will have many examples to learn from. You will find shell programming most useful in the daily administration of your system.

The purpose of this article is not to teach you everything there is to know about shell programming since entire books have been written on the subject. We will, however, teach you the basic concepts and commands so that you can get started. We assume that you have no programming experience and some basic knowledge of For:**Pro**. You will also need to have experience with at least one of the **For:Pro** editors (**ed, vi, screen,** also known as **sc**). Shell programs may not be entered using **Fortune:Word** unless they are converted to strip out special **Fortune:Word** characters. **Screen** is probably the easiest to use and is available from The Cambridge Consortium, Inc. (**Screen**, or **sc**, is available on the **Fortune Utilities** diskette, as well as the **DC Grab Bag** diskette). We will start out with some simple examples and explain new concepts as we build on the examples. We'll also warn you at the outset that for any given shell problem or program, 10 people will write it and accomplish the same end in 10 different ways. In fact, it is this diversity and flexibility that makes **UNIX** so popular. We mention this to hopefully fend off an influx of responses about how it could have been written better or more efficiently. Some of the examples are written to explain particular concepts and may not necessarily be the "best", most efficient or only way of writing the script. Generally to those of us that might be novice shell script writers, the main objective initially is that the program works.

THE SHELL

The *shell* (**Bourne Shell**) is an interpreter. It interprets

See The **UNIX Directory,** Page 14

Featured in This Issue...

Using **Records Processing** as a Typesetter - Dick Dow explains how to use **Fortune:Word** to make typesetting tasks easier...**Page 2**

Two Profitable Quarters - Details on Fortune's Second Quarter fortunes on **page 12.**

New Dealer Network is being built. **Robert Davis** and **Paul Olin** talk to **/u/fortune** *news*...**Page 5**

Problems with Records Processing and open files during backup - see **/u/help** on **Page 6.**

Writing shell scripts - This month's **UNIX Directory** begins a series on how to write shell scrips...**Page 1**

Hints on the **DIM** statement and how to restart **Accounts Receivables** in the **BASIC Advisor**...**Page 1**

The BASIC Advisor

Ray Wannall is president of BaSiC Software Corporation in Baltimore and is contributing to this publication independently.

Question: *I have a couple of things for you. First, something old in reference to your article about the* ***DIM*** *command (***/u/fortune*** ***news****, vol. 3, no. 6). I know several programmers who were originally trained on* ***DEC*** *and/or* ***Data General*** *equipment who are unaware that it is not always necessary to* ***DIM****(ension) a string variable before you reference a subscript. Strings may be created via padding or the* ***READ*** *and* ***INPUT*** *commands without a previous* ***DIM*** *statement.*

Also, the program example in the Business BASIC documentation book which uses the ***PRINT ""*** *commands is a left-over from the old* ***BB II BASIC*** *which had no mnemonics.*

Finally, something new. I recently installed the latest version of UNIX (1.8.1.1) and found a problem with ***BASIC.*** *It occurs when a program has been waiting for operator input for a significant amount of time and the* ***ESC*** *key is hit. This releases* ***BASIC*** *and sends you back to where you came from (e.g., the global menu). Did you know this?*

Answer: No, but I am not surprised. The original

See The **BASIC Advisor,** Page 7

*/u/fortune *news* is published by The Cambridge Consortium, Inc. It is not affiliated with Fortune Systems Corporation.

Do You Xerox Your Copy of /u/fortune news?

/u/fortune *news* is dedicated to Fortune owners and users and we publish it for the expressed purpose of helping everyone learn how to use their Fortune computer to its utmost potential. In some ways we are flattered that there are those who think enough of **/u/fortune** *news* to "pass it around." We certainly want Fortune users to benefit from our newsletter.

However, copying the newsletter and passing it around to friends and other Fortune owners, while being laudatory in some respects, is actually self-defeating. Why? Well, it costs money for us to produce **/u/fortune** *news*. We greatly depend on the subscription price of our newsletter to pay for the costs associated with **/u/fortune** *news*. Frankly, if we can't generate enough money from subscriptions, then we will not be able to pay our bills. And, if we can't pay our bills... We think you know what comes next.

So, if you currently copy **/u/fortune** *news* to give to other Fortune owners, consider that you are hurting us and ultimately yourself. Please stop. If you receive a photocopy, please consider contacting us for your own copy. Yes, you'll have to pay the subscription price, but you will be doing your part to ensure timely and useful information to all our subscribers for a long time to come. Naturally, we offer discounts on quantity subscriptions. Contact us and we will work something out.

/u/fortune *news*. . .

Editors: Josh Lobel, Mark Palmerino
Associate Editor: Benzion Chanowitz
Technical Editor: Scott Bradner
Advertising Manager: Mark Palmerino

/u/fortune *news*. . .is published 12 times per year by The Cambridge Consortium, Inc. Neither **/u/fortune** *news* nor The Cambridge Consortium are affiliated with Fortune Systems Corp. Contact us regarding subscription rates. Quantity discounts are also available.

Back issues are offered at the following prices:

Vol 1, Numbers 1-3 $1/each	Vol 2, Number 1-7 $3/each
Vol 1, Number 4 $.50/each	Vol 3, Number 1- $4/each
Vol 1, Number 5-8 $3/each	

Although every attempt is made to insure that our contents are error-free and without risk, The Cambridge Consortium is not responsible for losses that occur due to correct or incorrect interpretation of **/u/fortune** *news*.

For more information about us, please call or write: **u/fortune** *news, 73 Jason Street, Arlington,* MA 02174. Tel. 617 646-5603.
The text for this newsletter was prepared using a Fortune 32:16 for text input. The information was then transferred to an Apple Macintosh Plus with kermit and formatted for typesetting using the Aldus Pagemaker program.

UNIX® is a registered trademark of Bell Laboratories,Inc. Fortune 32:16™, Fortune:Word™ and FOR:PRO™ are trademarks of Fortune Systems Corporation. Multiplan™ is a trademark of Microsoft. IBM-PC™ is a trademark of IBM. MS-DOS™ is a trademark of Microsoft. Macintosh Plus™ is a trademark of Apple Computer. Pagemaker™ is a trademark of Aldus Corporation. HP Laserjet™ is a trademark of Hewlett-Packard Corp.

©1986 by The Cambridge Consortium, Inc. 8/86

The Fortune:Word Glossary

RECORDS PROCESSING AS A TYPESETTER

Dick Dow is owner of Dow Information Services which is a technical publications firm located in Rhode Island and is contributing to this publication independently.

There's a lot of talk these days about "desktop publishing" -- using a personal computer to compose and design newsletters and brochures, generally with a laser printer for the final typeset output.

However, the conventional form of publication production, setting type on a dedicated phototypesetter and then assembling it in "mechanicals" to go to a printer is still the best way to produce high quality pieces, publications with large formats, or publications which require the assembly of many type, art, and photographic elements on a single page.

We've been using a Fortune for several months now as a typesetting terminal, and sending the completed copy over the phone lines to a typesetting house where it is processed. Recently we discovered that **Fortune:Word Records Processing**, along with **Fortune:Word Glossaries**, can be a great help for many types of jobs. Even if you are not involved in typesetting, the following **Records Processing** application may suggest new ways you can use the program to solve your particular needs.

Many typesetters are now able to receive text by telecommunication. In addition, if you have some typesetting background, it is usually possible to embed typesetting commands into the text so that it can be processed without further intervention, saving both time and money. Here's how it works and how **Records Processing** can help.

Type is usually set on a dedicated terminal which has a number of special function keys not found on a computer keyboard. Some basic typesetting functions are line length, the size of the type, the style or font, and leading (the amount of white space between the lines of type). In addition, special commands indicate the formatting of the type (centered, flush left, flush right, justified, ragged left, ragged right, etc.). The typesetter can be programmed to recognized these commands as *mnemonics* printed out on a regular computer.

Continued on next page

We recently had to typeset a series of courses for an adult education catalog and discovered that **Records Processing** is uniquely suited for this kind of job. First, all the course descriptions had the same format. In published form, there was a bullet, followed by the course number in a medium typeface, followed by the title of the course in bold. Then came a one paragraph description of the course in a medium face slightly smaller in size than the heading, an instructor bio, a paragraph listing meeting dates and times, and a course fee.

If you read the two previous articles on **Records Processing**, you will immediately recognize each course as a potential *record* with a number of *fields*. That's what we thought, so we set up a list document in **Records Processing**, and, using a simpler form of the glossary we printed in Part II of the **Records Processing** article, began to key in the courses. The course number became the first field, the title was the second, the description the third, and so on, with separate fields for the instructor bio, the date/time, and the course fee. What makes **Records Processing** so useful here is that you can fit quite a bit of information into each field (up to 2,048 characters or if you have to sort the record); you have the advantages of word wrap. As with all Fortune:Word documents, you can edit easily, run the spelling checker ... in short, it's just like creating a word processing document.

Records Processing is uniquely suited for creating and updating typeset documents.

So far not too complicated. But it's in creating the format documents that you really begin to appreciate this program. Remember, the *format document* is the one that manipulates the data contained in your list document (the course data in this case) and you can have as many of these as you want. We set up two format documents: one printed out the course information as a double spaced manuscript for proofreading purposes and the other contained the typesetting codes.

In the proofreading format, we simply indicated where each field should be in relation to the others, put in extra spacing between fields, and indicated in the format line that this should be double spaced (**<COMMAND> s 2**).

The second **format document** was more complicated. We inserted all the typesetting mnemonics with the field labels indicating where each field's text should be. When the list document is merged with this format, all the courses are copied to an **output document** with the typesetting commands properly embedded and with no chance of missing one. If a change needs to be made in the type commands, it need only be made once on the format document and it will take affect with any number of courses merged.

Here's what our course **format document** looks like for manuscript printout.

```
31-<num>
<ttl>
<des>
<ldr>
<mtl>
MEETS: <dte>
COURSE FEE: .... <fee>
(Rev: <date1>)
```

Note that some text is contained in the format document. This prevents redundancy and saves space in the **list document**. The **31-** before the first field is the sequence number of the terms. By using it here, the format document can be updated each term rather than updating the entire course list. **<date1>** at the end automatically prints the date the course was revised.

Here's the typesetting **format document**.

```
[sz 13][st c1][pl 13][sl 10][rr]{[pi] k}[zl]
[il 0102]31-<num> [st c3]<ttl>[ix][us][ql]
[sz 8][st b1][pl 9][sl 11][rx]<des>[us][ql]
<ldr>[us][ql]
<mtl>[us][ql]
MEETS: <dte>[us][ql]
COURSE FEE: [ld] <fee>[sl 20][us][ql]
```

The typesetting mnemonics (commands read by the typesetting equipment) are enclosed in square brackets in the system we use. "**sz**" indicates the size, "**st**" the style, or font, "**pl**" the primary leading, and so on. The curly brackets in the first line call out the bullet from another font and then return to the font in use. The important thing to notice here, however, is how the field labels are contained within the mass of commands and text. Without **Records Processing**, we would have had to go through several hundred courses, inserting these commands in the appropriate place for each. The possibility for error was great, not to mention the time required.

Again, fixed headings like "**MEETS:**" and "**COURSE FEE**" are stored in the format document rather than in the list document.

Once the output document has been prepared, we run a special glossary containing several search and replace commands to make any necessary final adjustments.

If this use of **Records Processing** stimulates ideas for other applications to you, why not share them with us in a future issue. □

Our Memory Doesn't Cost You A Fortune. . .

Is Your Fortune Living Up to its Potential?

Would you like to:

Speed up your computer?
Add more users?
Use the latest software, like Fortune:Windows™?

Compatibility:
The F1MB is absolutely 100% compatible with Fortune's own 1 Meg memory board.

Dependability:
All boards are burned in for 72 hours and tested for 48 prior to shipment. *Our One Year Repair or Replace Warranty underlines our confidence in the product.*

Affordability:
USA is proud to have equaled and exceeded the quality and performance of our competition, while offering substantial savings for both the dealer and end user.

Repairability:
The F1MB facilitates repair in the field as it is 100% RAM socketed. Now you can repair a board by simply replacing a defective chip rather than face an expensive board exchange -- reducing maintenance costs and system down time.

If you have a Fortune SX, XP45 or recent model XP30, you can use our board by simply plugging it in. Older systems can be upgraded to use our board with a simple and economical modification.

For Details Call:

Upgrade Service Associates
3 Marion Road, Burlington, MA 01803
Phone: (617) 239-8114

Our Guarantee: If you're not completely satisfied within 30 days, *USA* will gladly refund your money.

Fortune Sells Solutions

A Conversation with Robert Davis and Paul Olin

In our last issue we noted that Robert A. Davis had recently been appointed to the position of Executive Vice President of Marketing. Simultaneously, Paul Olin was appointed as Vice President of Sales. Both men have extensive experience in marketing and sales of computer equipment, and have built several sales networks. We talked with them recently about their plans at Fortune.

Solutions and Support. These two words summarize Fortune's new commitment to the users of their equipment. "What the end user is interested in is a solution for their computer needs", says Olin. "We're going to get out of the business of selling boxes (computers) and get into the business of selling the total package." The "package" includes vertical market applications (programs that are designed for specific fields, such as legal, manufacturing, etc.) and the traditional office automation products for which Fortune is known. He feels that companies that can provide full technical and customer support will be the ones that will survive in the marketplace.

In order to do this, they are going to completely revitalize their dealer network. Bob Davis noted that Fortune may have made some mistakes in the past. "The kind of dealers we're looking for will have the ability to support their customers -- and we will give the dealers the tools they need to do that." They plan to have a two-pronged distribution channel, composed of large "Master Dealers" and smaller VAR's (Value Added Resellers) who will use the Master Dealers for access to Fortune Systems. In addition, some larger VAR's will continue to deal directly with Fortune Systems.

In their search for dealers, Olin outlined the qualities Fortune is looking for in their Master Dealers:

1) Hardware and Software Support for End Users
2) Sufficient quantity of equipment and spares on hand to provide timely service
3) Ability to provide complete maintenance
4) A sales force with the ability to sell in vertical markets
5) Support of a geographic region
6) Demonstration of a commitment to Fortune products by advertising
7) Ability to support a group of VAR's who will feed off of the Master Dealer and use the Master Dealers as their source for products and support

The VAR's will have similar qualities, but in general they will probably have a smaller customer base. "Our Master Dealers will be full service dealers in the true minicomputer sense of the word," cites Olin.

In the refined Master Dealer program, companies will not compete against each other because their concentration will be in their geographic area, where dealers can reasonably provide support.

Fortune is committed to a partnership with their dealers. They intend to help them hire the right people to provide the sales and service expertise that is needed. In addition, they will be certain that they are equipped with all of the latest demo equipment for customers to evaluate.

Olin noted that Fortune had walked away from their dealer base in the past. Many vendors had become alienated and discontinued carrying Fortune equipment. Others have developed a wait-and-see attitude. Fortune Systems intends to convince everyone that they mean business and that successful partnerships can be created.

"What the end user is interested in is a solution to their computer needs," says Olin. That is what Fortune intends to provide.

They have already taken several steps in this direction. Fortune will no longer have direct sales programs for large corporate clients. All purchasers will deal with a Master Dealer or VAR. This builds the dealer network and removes direct competition with Fortune Systems. For example, the Dearborn, Michigan office for Fortune, which was created to support Ford Motors, has been closed. Support will now be provided by Computer Trade Development, which is the Master Dealer in Michigan.

San Carlos Computer Supply, which was a Fortune outlet to sell refurbished equipment, has ceased sales to end users.

As part of the new commitment to support, Elaine Migliore, who had been laid off as Director of Education and Training at Fortune Systems, is now consulting with Fortune to rebuild this department. In order to bolster the vertical market applications and third party software vendors, Liz Muth has been hired as Manager of Software Services. Her responsibilities include overseeing the Independent Software Vendors (ISV) area. She has had over 20 years of experience working in this field.

In an effort to provide a solution to users who need more computing power, Fortune will be releasing a new product line based on the Motorola 68020 microprocessor. Although there aren't any details available at this time, a formal announcement and release is expected very shortly. □

Josh Lobel

/u/help

Question: We have been using Records Processing successfully for some time now. Recently a new employee tried to process a list and got an error message about a label problem in our format document. Everything appears to have the proper labels and merge symbols, etc. What could cause this problem?

Answer: It's hard to know exactly what your problem is, but here is one likely answer.

You know that around each label you must have the merge (**<** and **>**) symbols. These appear as bold greater-than or less-than signs. It is possible that someone accidentally put in a **>** symbol (SHIFT comma and period) and put it in bold rather than using the merge key (**F6**). Although both of these would appear the same on your screen, they are not identical. One way to ascertain whether this is the problem is to exit to UNIX and use the **more** command to examine the file.

From the Global Menu type an **!** mark to get to the **$** prompt. If you are not in the library with the **Fortune:Word** document, change to that library by using the command **cd library** <CR>. (If your library is **/u/joe**, you would type **cd /u/joe** <CR>).

Next use the **more** command by typing **more filename** <CR>, where filename is the name of your file. You may want to do this on your **list document** and your **format document**. (If you are viewing a long list document, you can search for a particular piece of information by typing **/search pattern**, where **search pattern** might be replaced with a name, address or such. **more** will automatically search for that pattern and display it on the screen.) You should see something like the following text:

```
\<\name\>\\B\
\<\addr\>\\B\
\<\city\>\, \<\st\>\  \<\zip\>\\B\

\X\<\x\name\X\>\x\\B\
\<\addr\>\\B\
\<\city\>\, \<\st\>\  \<\zip\>\\B\
```

Fortune:Word uses a three character sequence for all of its formatting commands. At the end of each line where there is a RETURN, you will notice the **\B** sequence which produces the **RETURN**. The merge symbols are specified by **\<** and **\>**. This is the correct way for them to appear. In the incorrect version, you will note that they are surrounded by the sequence to turn bold on and off -- **\X** and **\x**. If you look at the name sequence in the second example, it appears as **\X\<\x\name\X\>\\x\\B**. These are greater than and less than symbols masquerading as merge symbols and must be corrected.

To correct the problem, exit UNIX by pressing **CTRL-d**. Then go back to **Fortune:Word**, delete the greater than and less than symbols, and insert the correct merge symbols. This should fix the problem.

Question: I use the cp -Brust command to backup my entire /u directory. This takes some time because I use floppies. Can people use other terminals while this is happening, or will that cause a problem?

Answer: The answer is yes and no, or in other words, it depends. Your backup will not be ruined and neither will the files they are working on. However, any files that are being used *while* the backup is taking place will **not** be copied. If your users are willing to risk not having a backup of that particular file, then there is no problem. In any case, the file will continue to be stored on your hard disk. □

Josh Lobel

Accutrend System Services

Serving the Great Northwest
Washington, Oregon, Idaho, Alaska

Sales, Service, Support,
Software and Maintenance

- **Office Automation**
- **Accounting Systems**
- **Communications**
- **Database**

Free monthly newsletter

Call or Write:
Accutrend System Services
P.O. Box 38
Puyallup, WA 98371
(206) 841-1992

Authorized Fortune Dealer

BASIC Advisor, continued from page 1

BASIC on the Fortune had lots of trouble with UNIX, especially with numeric precision. It seems that when you used the **PRECISION** command to round off numbers, UNIX kept unwanted fractional parts in memory, thus throwing off subsequent calculations. I used to get around this by sending numbers through string variables and then re-assigning the numeric value to the original variable:

```
100 LET A$=STR(A:"#####"),A=NUM(A$)
```

Fortune Systems eventually corrected the precision problem and will, I am sure, fix your input problem. But in the meantime, there must be a way to get around it. Try this:

```
1000 INPUT (0,TIM=1,ERR=8200) A$
8200 GOTO 01000
```

If this doesn't work, try a **WAIT** statement at 8200 with the **GOTO 01000** statement at 8210. One of the reasons I love Business BASIC is that it always seems to be versatile enough to cope with cantankerous operating systems.

As far as the **PRINT ""** commands in the documentation are concerned, yes, I know this is a BB II hangover. But let's get with it! After all, this is 1986. If I spend all this money on a so-called modern language, I feel I am entitled to something more in documentation than a book which was last edited in 1975. (For those of you who are wondering what the heck a BB II is, it was the old Basic IV machine with disk drives that were powered by mice on a treadmill.)

And you are right about not having to **DIM** strings before referencing subscripts. (I, too, began as a Data General BASIC programmer many eons ago, but I have forgotten the language.) When you enter a program from any **BAS** selector you are provided with several restricted variables. One of these is **Z8$**, which is pre-defined as eighty spaces. This can be used to "pad" string variables before you reference a subscript. The following command will give you an **ERR 47** (Invalid Substring Reference) if the length of **A$** is less than six:

```
130 PRINT A$(4,3)
```

To avoid the error without using the **DIM** command all you need to do is pad **A$** using **Z8$** as follows:

```
110 IF LEN(A$)>=6 THEN GOTO 00130
120 LET A$=A$+Z8$(1,6-LEN(A$))
130 PRINT A$(4,3)
```

The **READ** and **INPUT** commands can create a string capable of providing you with a subscript as in the following:

```
100 READ (1) A$
130 PRINT A$(4,3)
```

I would still use statements 110 and 120 to insure that the length of **A$** is valid. You will encounter an error, however, if you try to **READ** or **INPUT** into a substring if the variable has not been previously **DIM**med or

Continued on next page

Participate in the PROGRAMMERS" GRAB BAG from SST DATA, INC.

It's Absolutely FREE!!!

While developing our HANDSHAKE software, the SST Data staff created a number of utility programs for Fortune computers. We are offering you an opportunity to get printouts of two utilities chosen at random from our list of ten small programs created by our development staff. Just fill out and return the form at the right.

SST DATA INC.

4701 W. SCHROEDER DR., SUITE 100
MILWAUKEE, WI 53223
• 414/355-6990

To: SST Data, Inc., 4701 W. Schroeder Dr., #100, Milwaukee, WI 53223

From:
Name:_____ Title:_____

Phone:_____
Company
Name:_____
Street
Address:_____
City &
State:_____ Zip:_____

YES Please send me a free utility for my Unix operating system from your PROGRAMMERS' GRAB BAG. I plan to install it on:

_____ (computer make, model)

Other computer systems currently in use at this firm include:

_____ (make, model)

_____ (make, model)

BASIC Advisor, continued from page 7

otherwise padded:

```
100 READ (1) A$(1,6)
```

Hopefully all of you DEC and Data General people are now experts in the use of Business BASIC substrings.

Question: *I need to change the precision on a data element which is used in several IDOL-defined files. I want to write my own utility rather than going through each file with 798. I have already edited all of the data entry screens and programs which use the element (print masks, input prompts, etc.) but I need to know which IDOL control files are maintained in the 798 utility. I figure I have to address the* ****FM*** *(File Maintenance) files, but what else needs to be altered?*

Answer: I usually try to find a way to live with the BAS files the way they are designed, because changing an element definition is usually a huge project. But I admire your spirit.

You are correct about the ***FM** files. These contain the screen definitions and element restrictions for **IDOL** file maintenance. Altering these files will be a major project in itself. I wish you luck.

You will also have to adjust records in the Global Element Dictionary (**CGLBD**), which holds data element parameters. The key to this Direct File is the element name (length of 15) plus the file number (length of three, masked: "000"). The precision field and, if you are expanding it, the length field will have to be updated.

If your edits to the element will cause the record length to increase, you must tell **CCNVZ**. This file contains a group of records called "File Information Records" which are accessed with the key **"F"** plus the 5- or 6-character file name. Adjust the "Record Size" field as needed for the new file.

That should take care of **IDOL**. Please call me back next year when you have finished the project. I'd like to know how it went.

Question: *I am re-starting Accounts Receivable after a two year break, and I would like to start fresh. The problem is I have a lot of old data in my Customer Master and Open Accounts Receivable files that I want to kill. Do I have go through file maintenance and delete each record individually, or is there some easier way to go about emptying these files?*

Answer: There are two ways to kill data, but I must caution you and everyone else that you had better be very sure you really want to destroy the data files before you try it. In fact, do a selective file backup (or even a complete End-Of-Day Data File backup) before you proceed.

If you have **IDOL** on your system, you may kill files with the **INITIALIZE SELECTED FILES** utility. Go into **IDOL** from the global menu, log in, and enter **S6** and <CR> (i.e., press <RETURN>). Selection number 16 allows you to kill and re-define up to five data files at one time. Names of files to be initialized this way must be exactly five characters long. When you are done, the initialized files will still reside on the system as they did before, but they will be empty.

If this sounds like too much effort, or if you have file names which are not exactly five characters long, you can write your own utility as follows (remember that <CR> means to hit the <RETURN> key):

From any BAS Selector, type **BASIC <CR>**

At the **>** prompt, type the following commands to kill **CCSMS**:

```
OPEN (1) "CCSMS"          <CR>
LET F$=FID(1)             <CR>
END                       <CR>
ERASE "CCSMS"             <CR>
FILE F$                   <CR>
RUN "DOL"                 <CR>
```

You will return to the BAS Selector with an empty Customer Master File. The Open Accounts Receivable File is **COPAR**. You may substitute **COPAR** in the **OPEN** and **ERASE** commands above to kill the data in that file also. □

BACKUP AND RESTORE UTILITY

PROTECTION MADE EASY!

- o Back up all or selected user directories
- o Restore all or selected user directories or files
- o Format diskettes
- o Review backup volumes
- o Menu-driven, easily installed
- o Complete documentation

Also copies the files necessary to preserve your system configuration--a tremendous help if your system ever needs to be cold booted!

$150.00 Backup/Restore Utility
$200.00 Utility + 2 boxes 5 1/4" 96 TPI high-quality diskettes

For more information, call CTD at (313) 540-1282.

To order, make check payable to *Computer Trade Development Corporation.* Send to:

BACKUP/RESTORE UTILITY
Computer Trade Development Corporation
Suite 1386
Birmingham, MI 48010

CTD

Company:_____
Contact:_____
Address:_____
City:_____
State:_____ **Zip:**_____

Dear Fortune Owner:

As you have probably read in */u/fortune news,* Fortune Systems SX models are significantly more powerful and faster than the PS and XP models. If you have a PS or XP model system, you should take advantage of our special promotion on SX upgrades. For a limited time, and while our supplies last, we will upgrade your system, including the new SX Cabinet at no additional charge for the cabinet. With this special offer, you may choose any combination of the following upgrades and/or expansions:

- *SX Processor with Current Release of Multi-User For:Pro*
- *60 MB Internal Streaming Tape*
- *30 MB, 45 MB, 70MB High Speed Hard Disk*
- *6-Port Intelligent Asynchronous Controller*
- *Upgrade to Current Release of all Fortune Software*
- *Fortune:Windows (Included at NO Charge)*

Additionally, we can perform these upgrades within 24 hours of receiving your system. In fact, if you ship your system overnight on a Friday for Saturday delivery, we will have it back to you on Monday morning, all ready to plug in and run; not losing the use of your system for a single work day. And, we will issue you a full 90-day warranty on the entire system.

JFM provides services to Fortune Systems owners nationwide either direct or through our authorized Affiliated Dealers. If you haven't used our services, I invite you to discover JFM; you will be glad you did.

Sincerely,

Morris Frydman
President

Talk to us, then talk to our clients

JFM
BUSINESS
SYSTEMS
The Merchandise Mart
Lobby Suite 144
Chicago, Illinois 60654
Call (312) 644-0870

Authorized Fortune Systems Master Dealer

Fortune Tech Tips

Issue Date: 3/18/86
Description: Horizontal video is out of adjustment. Characters appear to wrap around the left side of the screen reversed and double sized. Cursor appears long and faded.

Corrective Action: The 3.6 versions of the Basic Workstation have a feature which allows horizontal image movement through the soft set menu. If this is adjusted improperly, the symptoms above will appear.

To re-adjust the horizontal video, enter the soft-set menu by holding **<CONTROL/SHIFT>** and pressing the **<HELP>** key. When the menu comes up, hold the **<SHIFT>** key down, and press the right or left arrow keys to adjust the screen centering. Return to normal operation by pressing the **<HELP>** key.

Issue Date: 3/20/86
Description: Wyse built MS-DOS and Graphics workstations require jumper changes on the logic board when additional memory is added.

Corrective Action: Jumpers **J30** and **J31** are used in memory configuration. Proper jumper settings are shown below: (These jumper configurations apply to Wyse part number 99-0104-00 only.)

Memory Size	J30	J31
256K	IN	IN
512K	OUT	IN
640K	OUT	OUT

Issue Date: 3/16/86
Description: After converting to alternate console, printing from **BASIC** produces text which is unreadable. Or printing from **BASIC** prints on the console instead of the printer.

Corrective Action: The baud rate specified for **tty01** in the configuration menu is default baud rate for all ports. Since **tty01** is now the console, all printers will print at the same baud rate as the console. This is only true when printing from **BASIC**. This procedure allows the user to set the printer ports to the baud rate that is necessary for proper printing.

Text will appear on the console if a printer was once connected to **tty01**. **BASIC** prints to tty01 by default, but now **tty01** is the console. This procedure will change the default printer port to the new port that the printer is connected to.

Procedure:

1) Log onto the system as root.
2) cd /m/rc <CR>
3) ls <CR>

NOTE: If there is a file called menu.rc then continue to step 4, otherwise skip to step 8.

4) ed menu.rc <CR>
(The system will respond with the number 46)
5) s/01/xx/g <CR> (substitute the printer port number for xx)
6) w <CR>
7) q <CR>
(The system will again respond with the number 46)
8) ed tty.rc <CR>
(The system will respond with '?tty.rc'.)
9) a <CR>
10) stty (baud rate) savemodes>/dev/ttyxx <CR>
(e.g. substitute '1200' for '(baud rate)')
11) . <CR>
12) w <CR>
13) q <CR>
14) Now shutdown and bring the system back up and the printer should work as it did before.

Issue Date: 4/28/86
Description: When using **ITE** or other communications software via a **LAN** connection, attempts to disconnect from communications product only causes **LAN** to disconnect.

Corrective Action: When using a communications product such as **ITE** on a remote host connected via **Fortune:Link**, issuing the ~. disconnect command to **ITE** causes the remote host to disconnect. The problem is that **Fortune:Link** is interpreting the disconnect command rather than the communications product. If you want to end the **ITE** session only and remain connected to the host, the solution is to type ~~. to disconnect from the communications product only. If the **LAN** connection is routed through multiple nodes, an extra ~ must be added for each node. For example using **ITE** routed through two nodes would require ~~~. to disconnect only the **ITE** session. As always type ~. to disconnect from all nodes. □

JONES BUSINESS AUTOMATION CENTER

a *FORTUNE SYSTEMS MASTER DEALER*
and the
JBAC AFFILIATE DEALER NETWORK
are organized to
Support FORTUNE END-USERS with
Outstanding Professional Systems Support
at **Reasonable Prices**

Inquire about the JBAC "PERFORMANCE SOLUTION." Upgrade your present system to a new SUPER PERFORMANCE SX-70 which is the PRICE PERFORMANCE Leader.

* **BAS/IDOL SUPPORT**
* **FORTUNE:WORD**
* **MULTIPLAN**
* **CUSTOM DEVELOPMENT**
* **COMMUNICATIONS**
* **HARDWARE UPGRADES**
* **MAINTENANCE**
* **LASER PRINTERS**
* **OCR PAGE READERS**
* **COMPUTER NETWORKS**

The JBAC Affiliate Dealer Network

Synergistic Business Systems 4300 Paper Mill Road Knoxville, TN 37909 (615) 584-0211	**Advertising Data Mgmt.** 4173 First Avenue Tucker, GA 30084 (404) 938-8100
MicroAge Computer Store 1939 Poplar Avenue Memphis, TN 38104 (901) 722-8280	**Brinc Business Systems** 1662 Savannah Highway Charleston, SC 29407 (803) 763-0599
Software Sciences 3100 Evans Avenue Fort Meyers, FL 33901 (813) 334-1646	**DataLogix** 200 Central Park Avenue Hartsdale, NY 10530 (914) 997-1627
DataLogix 3000 Dundee Road Northbrook, IL (312) 291-9225	**DRS Enterprises** 127 E. Murphy Street Madison, NC 27025 (919) 427-0864
MultiUser Systems 10818 N.W. 6th Court Miami, FL 33168 (305) 758-2007	**Software Engineering Assoc** 301 East Greenville Street Anderson, SC 29621 (803) 225-0101

JONES BUSINESS AUTOMATION CENTER

834 Tyvola Road, Suite 112
Charlotte, NC 28210
(704) 525-0582

NEWS FROM FORTUNE

FORTUNE SYSTEMS REPORTS SECOND QUARTER PROFITS

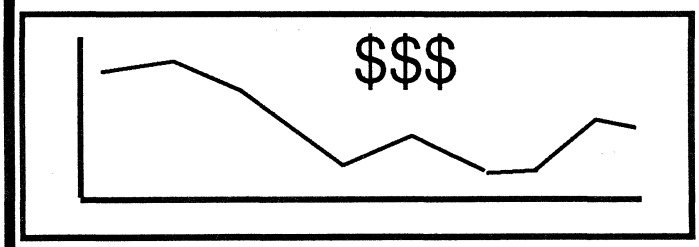
Fortune Systems Corporation reported a profit of $253,000, or $.01 per share, on revenues of $9,717,00 for the second quarter ended June 30, 1986. For the second quarter of 1985, Fortune reported a profit of $222,000, or $.01 per share, on revenues of $14,628,000. Cash balances increased to $22,704,000 at June 30, 1986.

For the six months ended June 30, 1986, the company reported a profit of $598,000, or $.03 per share, a significant improvement from the loss of $3,570,000, or $.17 per share, for the first six months of 1985. Revenues for the 1986 and 1985 six-month periods were $20,375,000 and $24,437,000, respectively.

"For the first time since the company went public in March 1983, we were able to generate consecutive quarterly profits," stated Fortune President and Chief Executive Officer **James S. Campbell**. "Revenues decreased over the comparable 1985 quarter as a result of a purchasing moratorium which Fort Motor Company imposed on all computer vendors, a $2,200,000 sales of software source code in 1985, and the continuing softness of the computer market. Despite lower revenues, we were able to show a profit as a result of reduced operating expenses coupled with significant improvement in our gross margins. The 1986 second quarter results were also favorably impacted by approximately $400,000 from the successful recovery of prior reserves established for costs associated with excess facilities and severance.

"The company increased its cash balances for the second consecutive quarter, also a company first," continued **Mr. Campbell**. "We were able to increase our cash balances by approximately $530,000 in spite of the use of approximately $760,000 to repurchase 429,500 shares of our common stock."

He added that the company's new senior marketing and sales staff is presently developing a series of aggressive OEM, systems integrator, and value-added reseller (VAR) programs which should strengthen the company's position in the marketplace during the latter part of the year. "Initial discussions with our master dealers and VARs have been very positive," added **Mr. Campbell**.



To the Editor

Dear Sir:

I read with interest your article about the use of Fortune systems in law firms. (*Vol. 3, No. 7*) You correctly noted that many firms with Fortune computers use our Time & Financial Management System (TFM) software to perform their business data processing.

I fear, however, that you may have left the impression that our TFM software is very expensive and beyond the reach of small law firms. In fact, the basic TFM system including extensive reporting, integrated general ledger and financial statements is currently priced at $12,500 on the Fortune computer. The price is considerably higher when it is offered on larger unix-based machines.

It is important to keep the software price in perspective from both the vendor's and the law firm's point of view. TFM is the result of over 25,000 man hours of professional effort in development, implementation, testing, documentation and refinement over an eight year period. In the world of serious business software, this is the magnitude of effort needed to produce a high quality product. The product alone is not enough, however. An equally important element is a highly skilled professional staff to install the product, train the firm's personnel, assist the firm in converting from their current system, and to handle both accounting and technical questions for years after the installation is complete. The costs of our commitment to continued product refinement and a high degree of professional support for every client are reflected in our pricing. Substantially lower prices would require quality compromises which we will not make.

From the law firm's perspective, the cost of TFM to a three attorney firm over five years amounts to one hour per attorney per month. The TFM system can usually save more time than that just in the area of bill preparation, to say nothing of the other benefits of the system.

Finally, a comment about law firms who use attorney time to develop systems. The TFM price of $12,500 represents 3 to 4 weeks of effort for an attorney billing at the rate of $75 to $100 per hour. To divert this effort from legal work to system development is questionable economics. . . Even if such a firm somehow figured out how to be 10 times more productive than we are, it would still cost 2,500 hours of professional time, i.e. as much as $25,000 in billings, to obtain what we already have. In the final analysis, the notion that any serious business software system can be developed in a few weeks or even months is fanciful.

Thank you for your opportunity to clarify this point.

Sincerely,

Richard J. Gill

SOFTWARE FOR THE FORTUNE 32:16

MULTIPLAN LINK - can pass data from BAS/IDOL/BASIC files to Multiplan and back. For example, it could be used to send data from any of the BAS files (Chart of Accounts, Customer Master, Inventory Master, etc). Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

RECORDS PROCESSING LINK - can pass data from BAS/IDOL/BASIC files to Records Processing and back. For example, it could be used to take customer address data from the Customer Master and will automatically make the Records Processing List document for mailouts. Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

TERMINAL/PRINTER LINK - provides capability to print Multiplan, and BAS/IDOL reports on a printer connected to a workstation, view any report on the terminal before printing, convert any report to Fortune:Word format; accumulate multiple BAS/IDOL reports before printing, backup reports onto diskette; copy or rename report files; print multiple copies of the report on either the primary and/or terminal printer. Also has option to print BAS/IDOL reports on HP LASER printer ($50 extra) **$195.00**

TELEPHONE LINK - on line telephone message, telephone directory and inter-office mailbox/memo system which can also be used as a simple calendar reminder system. Provides control, printing and viewing of on-line data for each user. **$295.00**

FINANCIAL LINK - stores BAS Income Statement and Balance Sheet data in files. Once stored, it can be sent to Multiplan or instantly printed. **$195.00**

MENU LINK - provides software and documentation for adding items to the Global Menu as well as a sub-menu for your use Includes backup script for user directories and BAS/IDOL/BASIC programs and data. **$95.00**

CALCULATION LINK - provides programs for amortization, depreciation, loan repayment, averaging (with graph), linear correlation, and breakeven analysis. **$95.00**

KOMPACT PERSONNEL ACCOUNTING - provides for data capture of personal, wage/salary, job, education, salary/wage, appraisal training, dental, medical and life insurance data. Includes over 80 pre-defined reports/worksheets. **$495.00**

CRT LINK - provides information for using other terminals on a Fortune System (WYSE, TV, IBM 3101). **$95.00**

AUTOMATIC REPORT LINK - provides capability to print numerous IDOL defined reports automatically based on frequency or application without having to select each from a different menu. You identify frequency/application codes. **$195.00**

TAPE LINK - provides one menu for all tape backup utilities. Provides daily, weekly, monthly and individual backup and restoration routines. Includes rewind, retension, and tape unit startup/stop utilities. User friendly. **$95.00**

PRINTER LINK - provides printcaps for Okidata 84, 2410, TI 855, and Anadex 6500. Includes shell scripts for changing pitch and LPI. Allows these printers to be used in word processing. **$95.00**

MODEM LINK - allows Hayes 1200 Smartmodem or compatible to be used on one port for both incoming and outgoing calls. **$95.00**

INSTALL LINK - provides documentation and software example for installing your product/menu through "S5". **$95.00**

REPORT LINK - modifies IDOL Report Generator Program to allow stacking of up to 8 fields in an IDOL defined report. Also corrects problem with displaying subtotals for all stacked fields. **$95.00**

WESTERN LINK - provides instructions on how to use your Fortune for sending/receiving telex messages using Western Union's Easylink. Can replace your telex! Fortune ITE software is required. **$95.00**

BACKUP LINK - provides menu selection for backing up /b (BAS/IDOL/BASIC), /u (Fortune:Word, Multiplan) and entire system onto diskette. Also has program to "unhang" a terminal that's hungup just by entering the terminal #; a program to restore one file from a multi-volume backup; and a selection to tell you which volume a backed up file is on. **$95.00**

ASCII LINK - can pass data from BAS/IDOL/BASIC files to ASCII and back. For example, it could be used to send data from any of the BAS files (Chart of Accounts, Customer Master, Inventory Master, etc). Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

UNI-KOMP 9803 Rocktree Houston, Texas 77040 (713) 937-7782

UNIX Directory, continued from page 1

what you type into the computer into a language that the computer can understand. Luckily, the shell is in a language that the average user can understand. This article will only deal with the Bourne Shell (not the C-shell, since it is only available on machines that have Fortune Systems **Development Utilities** installed).

There are a number of ways that you can access the shell. One way is to enter a **!** at the bottom of the Global Menu (and most of the sub-menus). Another way is to login as **root** which will take you directly into the shell. Also, any login account can be changed to go directly into the shell by selecting **S2** (System Management), **11** (Modify existing account), **S** (Assign a shell script) and entering `/bin/sh`. The normal default for the assigned shell script is `/bin/menu`, which is the Global Menu. If you are **root** or **manager**, you will get the **#** as the shell promp; otherwise you will get a **$** prompt. This prompt indicates that the shell is waiting for you to enter a command. To exit the shell and return to the **Global Menu** or to the login screen, enter the command **exit** or depress **CTRL-D**.

Let's suppose that whenever we enter the shell, we want to know three pieces of information: the system date & time, who is on the system and what files are in our home directory (the directory we are automatically placed into when we login). This would normally require entering each of the three commands (**date, who, ls**) on a separate line (or **date;who;ls** on one line). Because we have better things to do with our time, we decide to write our first shell script which will run all of these commands at once. This is really the simplest kind of shell program we can write. Its purpose is to combine any number of commands together which can be executed by entering a single command.

So fire up your favorite editor and create a file (shell script) in your home directory called **info**. The shell program file will look like:

```
date
who
ls
```

Once the file is created, don't forget to change the permissions to make it executable (**chmod +x info**). Now to execute the program, just type in **info** to the UNIX prompt and it will execute all three commands for you. Congratulations - you've just written your first shell program!!

If we wanted this particular sequence of commands to be executed each time we logged in, we can add it to a **.profile** file in our home directory (if a file called **.profile** is present in our home directory, it is always executed each time we login). The **.profile** is also a shell script. In this case, it would only contain one command - our new command **info** (or we could have entered our three commands directly into the **.profile** file). After you create the **.profile** file, logout and log back in to see what it does. We will mention this file several times throughout this article because it can be used for many things. (For more information on the **.profile** file, see the **UNIX Directory** in *Volume 2 Number 5*).

One thing you should have noticed when you logged in was that your information probably didn't stay on the screen long enough for you to read it. There are at least two things we could do to give us enough time to read the information. One is to change the **ls** command to **ls | more -d**. This would pipe the output from the ls command to the **more** command which would allow the data to be read a screen at a time. We also need a pause after the last page of output since **more -d** does not pause after the last page of output. The **sleep** command is used frequently in shell programming for pauses, particularly if we want to display an error message or other output and want the user to have time to read the data. For example, the command **sleep 5** would cause the program to pause for 5 seconds before proceeding. Add the `sleep 5` command and the **more -d** to your **info** script, logout and back in again and see what happens. Your script should now look like this:

```
date
who
ls | more -d
sleep 5
```

SHELL VARIABLES

We like our shell script so much that we decide we want it to run for all users when they login and we want all users to be able to run it from the shell prompt regardless of what directory they are in. Before you can do this, however, we need to learn about *shell variables*.

In simple terms, there are two types of shell variables - those that are **pre-defined** and **user-defined**. Here some of the **pre-defined** variables:

HOME Indicates the pathname for your home (login) directory

PATH Indicates the directories that are searched to find any **For:Pro** command you run before you get the message **command not found**.

PS1 Indicates shell primary prompt (**$** or **#**)

PS2 Indicates shell secondary prompt

TERM Indicates terminal type from termcap file

Continued on next page

These are not all of the pre-defined variables. If you want to see all of your pre-defined variables, enter the **set** command at any shell prompt and it will not only show you the pre-defined variables but will also show you the value of each. We can access any of the pre-defined variables from our shell scripts. To find the value of any variable, we will use the **echo** command. We'll be looking at this command in more detail as we proceed. The **echo** command is used to display something on our terminal (unless we redirect it). For example, to find the value of the **HOME** variable, we would enter:

```
$ echo $HOME
/u/davek
```

Notice the variable name must be preceded by the **$** in order to display the value of the variable. A common error that beginning shell script writers make is to forget to precede the variable name with the **$**. The result of this error would be:

```
$ echo HOME
HOME
```

Instead of displaying the *value* of the HOME variable, we have literally displayed the word **HOME**. In addition to the pre-defined variables, we can create our own variables. For example:

```
$ x=1
$ display="Your magic number is: "
$ combine=$display$x
$ echo $x
1
$ echo $display
Your magic number is:
$ echo $combine
Your magic number is: 1
$ echo $display$x
Your magic number is: 1
$ echo "$display"2
Your magic number is: 2
```

The above may look rather imposing at first, but bear with us as we explain. First we set the variable **x** equal to 1. Then we set the variable **display** to the string **"Your magic number is: "**. We use the quotes in this case because we want the space after the ":" to be part of the value. The shell normally treats a space as a *delimiter* (divider) unless it is included in quotes. (To play it safe, the easy rule is to always include the right hand side in quotes so it will become habit and not a memory item.) We then set the variable **combine** to the value of the variable **display** and the value of the variable **x**. Next we **echo** the value of each of the variables. Note that the **combine** variable is simply a combination of the first two variables. The last two **echo** commands in the example simply show that we can **echo** two variables or a variable and literal text without reassigning them to another variable. This becomes important when writing our shell scripts since our goal is to minimize the amount of typing and to maximize the speed with which it is executed. In the above example, the **combine** variable is not really needed. Also notice that whenever we combine variable and literal text (**echo "$display"2**) the variable should be in quotes so the shell can distinguish the variable from the literal text. If we had entered:

```
$ echo $display2
```

Nothing is echoed since the shell is looking for a variable called **display2** and there is no such variable. In fact, you will find there are times when a double quote (") and times when a single quote (') are appropriate. For example, the following would not achieve the desired results:

INCORRECT:

```
$ echo "The "date" command displays the date"
The date command displays the date
$ echo 'The 'date' command displays the date'
The date command displays the date
```

We will find that the single and double quote are pretty much interchangeable with the **echo** command. Either of the following two entries would be appropriate:

CORRECT:

```
$ echo "The 'date' command displays the date"
The 'date' command displays the date
$ echo 'The "date" command displays the date'
The "date" command displays the date
```

In the **INCORRECT** examples, the word **date** is not enclosed in either single or double quotes when echoed. The general rule would be to use the double quote unless you need a double quote in your literal string.

We need to discuss two more important concepts concerning shell variables. First of all the shell is very particular about when spaces (*white space*) are appropriate and when they are not. We will attempt to point out these occasions as appropriate. Whenever we set variables, **DO NOT PUT SPACES ON EITHER SIDE OF THE = SIGN.** You will get error messages like the following:

```
$ x = 1
x:  not found
$ x =1
x:  not found
$ x= 1
1:  not found
$ x=1          <=   CORRECT
```

Continued on next page

UNIX Directory, continued from page 15

In the first two cases, the shell tries to execute **x** as a **UNIX** command and since there is no such command, we get the error message. In the third case, the space delimits the **x=** portion from the **1** portion and we get an error message as the shell tries to execute **1** as a separate command.

The last concept about variables is important and will be frequently used in your shell script writing. A variable may be set to the result of some command by putting the right side argument in backquotes (`). This key is the upper left side gray key on your keyboard: **Example:**

```
$ port=`tty`
$ echo $port
/dev/tty02
```

This example sets the variable **port** equal to the result of the **tty** command which tells you which terminal you are on. (You might just try typing **tty** to the UNIX prompt to see what this output looks like.) We can also combine this with literal text just like other variables:

```
$ echo "You are on port: " `tty`
You are on port: /dev/tty02
```

This can be used with any **UNIX** command and thus can be used in ways which are only limited by your imagination. For example:

```
$ dirlist=`ls $HOME`
$ echo $dirlist
file1 file2 file3
```

We are now putting some of the concepts we just talked about together. In this example, we are letting our own variable and a pre-defined variable do a job for us. First, remember that we said that **HOME** is a variable whose value is the pathname of our home directory. So we set the variable **dirlist** equal to the result of a **ls** command in our home directory. In this case there are three files in our home directory so the value of **dirlist** is the name of the three files. Later we will show you how to apply this concept in more useful ways.

By putting together some basic concepts, we can create more complicated and useful shell scripts.

Okay, with that ground work laid, let's continue on with our **info** shell script to accomplish the mission at hand. Remember, one of the things we wanted to do was to change this script so that it could be used by all users.

We left that shell script looking like this:

```
date
who
ls | more -d
sleep 5
```

Right now the script is in our home directory and when the **ls** command is run, we get a list of files in our home directory. The only thing needed to make the script globally useful to all users is that we need the **ls** command to be executed in the user's home directory (not ours). There are two ways we could do this:

Example 1:

```
cd $HOME
date
who
ls | more -d
sleep 5
```

This script would do the trick. The **cd $HOME** would insure we are in the home directory of the user. Remember the value of the variable **HOME** will be different for each login account; in fact, the value will be the pathname of the home directory for that particular user. The home directory is determined at login time from the **/etc/passwd** file and assigned to the variable **HOME**.

For those of you that have your thinking caps on, here's a question for you. What directory will you be in after this shell script has run? Don't cheat and read ahead for the answer!

Example 2:

```
date
who
ls $HOME | more -d
sleep 5
```

This example will accomplish the same purpose. The **ls** command will be executed in the user's home directory.

The answer to the above question is that you will be in the same directory you were in when you executed your shell script. *This is an important point.* A shell script is a program and after it has finished running, the shell forgets just about everything in the program (with a few exceptions which we will point out when appropriate). Therefore, even though our shell script executed the **cd** command, the shell doesn't remember that once the program has been completed.

Frequently in our shell scripts, we will run other shell scripts from our initial script. This would be the case, for example, if our shell script was a menu and when the

Continued on next Page

user selects an item from the menu, we execute another shell script and then return to the original menu. Following up on the point made above, we said that the shell forgets some things as it goes from one shell script to another. This is true of shell variables. Normally, if we set a variable in shell script 1, the variable and its value are forgotten in shell script 2, etc. The shell has a special command called **export** to take care of this problem. Whenever we **export** a variable, it makes the variable and its value accessible to **subsequent** shell scripts. Example:

```
x=1
name="david"
export x name
```

The variables and values of **x** and **name** would now be available is subsequent shell scripts. *Variables cannot be exported backwards.* For example, we could not assign these variables and values in shell script 2 and have them accessible in shell script 1.

Variables may be changed simply by giving them a new value. This is true of the pre-defined and our own variable. Although it is not wise to change pre-defined variables, there have been some shell scripts we've written where this was necessary. For example, to change the value of the **HOME** variable:

```
$ HOME=/u/davek/bin
```

The **cd** command, in fact, uses this pre-defined variable to determine our home directory when we enter **cd** without any arguments. Once we logout and back in again, however, the value of HOME is set back to its original value. Here's another example:

```
$ PS1="Prompt: ";export PS1
Prompt:
```

Remember we said earlier that the **PS1** variable sets our primary shell prompt, which is a **#** for **root** and **manager** and **$** for all other users. So, if we don't like this prompt and want to assign a new one, we could substitute whatever prompt for the word **Prompt** in the above example. The above example changes our prompt from **$** to the word **"Prompt: "**. If we put this command in our **.profile** file, this would set our new prompt each time we logged in.

And yet another example:

```
$ PATH="$PATH":/u/davek/bin; export PATH
```

We mentioned above that the **PATH** variable determined which directories were searched whenever we executed a shell command. Some of us create our own **bin** directory which has our own shell scripts and programs. The above command would set the **/u/davek/bin** directory in the search path. Again, we could add this

Continued on next page

UNISOFT INC.

Experts in Computer Consulting - Sales - Service

Presents

FORTUNE SYSTEMS HELP

We Provide:

- Expert Trouble-shooting
- Years of programming and accounting experience
- *Free Initial Consultation*
- Training Available for:
  * BASIC/IDOL/BAS and Multiplan
  * UNIX and Ext. Fortune:Word
- Hardware Repairs & Upgrades
- FORTUNE trained experts at a cost that will surprise you!
- *Trade-Ins Welcome*
- Specializing in:
  * Hard Disk Trade-In
  * Hard Disk Upgrades

Authorized Fortune Master Dealer

Authorized Fortune Systems Repair Depot

CALL: (303) 671-8600 EMERGENCY HELP: (303) 691-HELP

UNISOFT INC. 14100 East Jewell Avenue Suite 27 Aurora, Colorado 80012

UNIX Directory, continued from page 17

command to our **.profile** script.

Last example:

```
$ TERM=wyse50; export TERM
```

The **TERM** variable tells the shell the type of terminal we are on as specified in the **/etc/termcap** file. We could change this variable in our **.profile** file, for example, if we were using a terminal other than a Fortune (**FT**) terminal on our system. This variable is used to tell the **vi** editor (as one example) which entry in the termcap file to use to specify terminal characteristics. Assuming we had an entry for a **wyse50** terminal in our **/etc/termcap** file, the above example would tell the shell to use that entry for our terminal. (Note: this variable is normally set by the **/etc/ttytype** and **/etc/devtype** files when we **Define Device Connections** from the **Global Menu**, but can also be changed in the **.profile** file).

Now that we have made our **info** program global to all users, there are a couple of ways that we could have it executed for all users when they login. First of all, in either method, we can make the **info** command universally executable for all users by changing its permissions to be **executable** for all users and put it in **/usr/bin** which is in all users search path (**PATH** variable). Remember we created the **info** file in our home directory which is not searched for other users.

One way to have the **info** script executed for all users has already been discussed. It would be to create a **.profile** for each user in their home directory and execute the **info** command. This is the most cumbersome way to do it. The best way would be to add the **info** command at the end of the **/etc/profile** file. Let's make sure we understand how this file differs from the **.profile** file located in each user's home directory. The **.profile** file is executed for one user only if it is present in his/her home directory. The **/etc/profile** (there is no period (.) before the word *profile*) is executed for **all** users who login into the **Global Menu**. We now have the ability to execute commands for a given user or for all users (that is, all users who get the **Global Menu** at login). Since we want our `info` script to run for all users, we put the command in the **/etc/profile** file. You will notice that if you look at this file on your system, it already has some entries that **Fortune Systems Corp.** has put into it.

By putting the **info** script command in the **/usr/bin** directory and making it executable (**chmod a+x /usr/bin/info**), we can also execute it from the **#** or **$** prompt. If you remember, this was the last item on our list for the **info** command.

This concludes our initial step into the world of shell scripts. Although the material may seem somewhat disoriented at this time, hopefully, it will become more meaningful as our article continues in coming issues of **/u/fortune** *news*. What we have described thus far is how to use shell scripts to combine several UNIX commands together to get a job done for us without having to type in each individual command. Our discussion will continue with other uses for shell scripts; other shell commands and concepts; and will conclude with a script that we will print and go through line by line, hopefully to tie all the material together. We leave you with the following real life example of two simple shell scripts. The first is called **m** and mounts the floppy disk drive, tells you how much space is available on the diskette and lists the contents of the **/f** directory. The second is called **u** and unmounts the floppy disk drive.

Mount (m):

```
dev="/dev/fd02"
mount $dev /f
df $dev
ls /f
```

Unmount (u):

```
umount /dev/fd02
```


VANTINE COMPUTER SERVICES

We Are Not The
Biggest **Biggest** **Biggest**

Nor The
Cheapest **Cheapest** **Cheapest**

We Simply
Care **Care** **Care**

VCS is an Authorized Fortune Systems Dealer. We understand and support the products we sell.

We provide sales, upgrades, service and support to our Fortune Clients from Maine to California.

We don't ask you to believe us. We want you to believe our clients. Call for references today.

Vantine Computer Services can provide the solution for you!

VANTINE COMPUTER SERVICES
9 Priest Road, Sudbury, MA 01776
(617) 443-8412

Authorized Fortune Dealer

Fortune Users:

G.T. Hawes & Co.
An Authorized Fortune Master Dealer
Announces a Fortune Liquidation Sale

It's true. We're ever on the look out for *good buys* and we struck a proverbial FORTUNE! Two major users wanted out and we were there. We said, "HELLO." They said, "GOODBYE." And what a *GOOD BUY* it was! You guessed it, we're overstocked and must move the product. Take advantage of our prices because they were never lower.

CALL GEORGE AND SAY HELLO, SO I CAN SAY

GOODBYE!!!
NO REASONABLE OFFER REFUSED!
516-568-0044
Equipment Schedule Available on Request

P.S. As this ad was going to press another large user called... The price was so good we bought more... Call us and let us show you how we can expand your Fortune System the most economical way possible. We also sell parts for those users who perform their own system maintenance.

"If you don't see my name on your computer shipment you probably paid more than you should have."

G.T. HAWES & CO., Inc

475 W. Merrick Road
Valley Stream, NY 11580
516 568-0044 516 627-5556

DO YOU KNOW...

there is an organization that has a thorough knowledge of

FORTUNE BAS
MULTIPLAN IDOL

We have been providing widespread support to our Fortune clients for more than 2 years.

If you need:

- training
- support
- maximum computer performance
- peace of mind relating to processing
- customization
- application results
- minimum errors and confusion
- better ideas on how to increase the power of your Fortune

Call Dan Borrowman at (314) 837-7111

SUPERIOR COMPUTER SYSTEMS, INC.
917 N Hwy 67, Suite 105
P.O. Drawer T
Florissant, MO 63031
Making computers work for you

Expand your Fortune® without spending a fortune.

New equipment costs big bucks. The Good Fortune Trader sells fully tested burned-in equipment, refurbished to Fortune® standards at prices that won't make waves with your cash flow. Call or write today for a listing of our specials.

THE GOOD FORTUNE TRADER
60 DAY WARRANTY
a division of Computer Services of Atlanta
(404) 392-9938
1534 Dunwoody Village Parkway/Atlanta, Georgia 30338

The Cambridge Consortium, Inc. is expanding. We will be moving to new quarters very shortly. Until our new offices are ready, please write to us at the return address below. Please note that this interim address is a new address. Our temporary phone number is (617) 646-5603.

/u/fortune *news*
73 Jason Street
Arlington, MA 02174

/u/fortune* *news*. . .

The Newsletter for Users of the Fortune 32:16 Computer **September 1986/Volume 3 Number 9**

The BASIC Advisor

Ray Wannall is President of BaSiC Software Corporation in Baltimore and is contributing to this publication independently.

Question: *I looked over your article on expanding the number of terminals in BASIC (/u/fortune news, vol. 3, no. 7) to see if you covered everything. It seems you got all the hard ones, but you missed basic.Tpool. - Richard Gill, Gill & Piette, Inc.*

Answer: I certainly hope you give me at least a B+ for effort. Here is how to expand the basic.Tpool files:

Log onto the system as root, and, at the # sign, type:

```
cd /b/basic.Tpool          'CR'
cp 7 8                     'CR'
```

Hold down the grey **CTRL** key and press **D** to exit UNIX.

This sets up a **Tpool** file for terminal 8. You set up terminal 9 similarly, except use the command cp 7 9. Be sure you follow all of the other steps in the article.

Question: *I am enclosing a response to an inquiry from a dealer that takes a checklist approach on how to take a Fortune up to 16 BASIC/IDOL/BAS users. Beyond 16 users, we would have to replace the Fortune BASIC with a Thoroughbred version supporting up to 32 users, anyway. But personally, I'll wait until the Formula has been running effectively for half a year before I'll try anything that big.*

Embedded within the report enclosed are several additions and variations of your ***/u/fortune*** *article. At this point I feel it's comprehensive, but it is on our* ***Fortune:Word*** *active library if we discover anything new (you don't have to take that as a challenge).*

By the way, the dealer has not yet paid for this research production, but I'm sure he'll derive some nice benefits from it. Sound familiar?

Have a good time in San Diego. - Steve Rosenfeld, Superior Computer Systems, Inc.

Answer: Now who out there hasn't sent a check to Mr. Rosenfeld? Shame on you!

I shall attempt to paraphrase the two-page, seven-point enclosure for taking a Fortune up to 16 ports into

See **The Basic Advisor**, Page 2

Featured in This Issue. . .

User Spotlight -- What do **Ray Esposito** and **Dick Randall** have in common besides their Fortune? Find out. . .**Page 18**

News from Fortune -- Fortune Systems is taking to the road with over 25 new products, including an exciting new generation development. Find out where you can see them. . .**Page 17**

The UNIX Directory -- **Dave Kloes** details the *for loop* and *case statement* in his continuting series on how to write shell scripts. . .**Page 12**

Know everything there is to know about Multiplan? By the time you finish this series of articles you just might. Part One. . .**Page 1**

An update on adding more terminals to BASIC -- and more in this month's **Basic Advisor**. . .**Page 1**

Automatic reboot after power failure -- Here's a feature that is a must for systems accessed via modem. . .**Page 10**

Lawyers Unite! -- **Page 8**

Introduction to Multiplan

Beginning with this issue we will present a series of articles that will introduce Fortune's spreadsheet program, Multiplan, to the uninitiated and the initiated alike. In this first installment we cover some of the basics of starting and using Multiplan. In addition we outline a suggested sequence of steps for defining the problem and implementing a solution in Multiplan. In subsequent installments we will actually present working models and build them up from scratch so that you will gain hands on experience. Next month, for example, we will take you through a model which performs Loan Amortization Analysis.

Getting into and out of Multiplan

There are two main ways of beginning a **Multiplan** session. The first is to use the **Global Menu**. Starting **Multiplan** this way is very straightforward. You simply choose the **P1** option and hit the **EXECUTE** key and momentarily you will see the **Multiplan** Menu which will be similar to figure 1. At this point, to actually get into a worksheet you would choose the **Run Multiplan** option and then hit the **EXECUTE** key. In a second you would see an empty **Multiplan** worksheet which we show in figure 2. Once inside of **Multiplan**, you can begin creating a new worksheet, with data and formulas, by working with the screen as you see it. On the other

See **Intro to Multiplan**, Page 3

*/u/fortune *news* is published by The Cambridge Consortium, Inc. It is not affiliated with Fortune Systems Corporation.

Accutrend System Services

Serving the Great Northwest
Washington, Oregon, Idaho, Alaska

Sales, Service, Support,
Software and Maintenance

- **Office Automation**
- **Accounting Systems**
- **Communications**
- **Database**

Free monthly newsletter

Call or Write:
Accutrend System Services
P.O. Box 38
Puyallup, WA 98371
(206) 841-1992

Authorized Fortune Dealer

/u/fortune *news*. . .

Editors: Josh Lobel, Mark Palmerino
Associate Editor: Benzion Chanowitz
Technical Editor: Scott Bradner
Advertising Manager: Mark Palmerino

/u/fortune *news*. . .is published 12 times per year by The Cambridge Consortium, Inc. Neither **/u/fortune** *news* nor The Cambridge Consortium are affiliated with Fortune Systems Corp. Contact us regarding subscription rates. Quantity discounts are also available.

Back issues are offered at the following prices:

Vol 1, Numbers 1-3	$1/each	Vol 2, Number 1-7	$3/each
Vol 1, Number 4	$.50/each	Vol 3, Number 1-	$4/each
Vol 1, Number 5-8	$3/each		

Although every attempt is made to insure that our contents are error-free and without risk, The Cambridge Consortium is not responsible for losses that occur due to correct or incorrect interpretation of **/u/fortune** *news*.

For more information about us, please call or write:
/u/fortune *news*, 73 Jason Street, Arlington, MA 02174. Telephone (617)646-5603.
The text for this newsletter was prepared using a Fortune 32:16 for text input. The information was then transferred to an Apple Macintosh Plus with kermit and formatted for typesetting using the Aldus Pagemaker program.

UNIX™ is a trademark of Bell Laboratories,Inc. Fortune 32:16™, Fortune:Word™ and FOR:PRO™ are trademarks of Fortune Systems Corporation. Multiplan™ is a trademark of Microsoft. IBM-PC™ is a trademark of IBM. MS-DOS™ is a trademark of Microsoft. Macintosh Plus™ is a trademark of Apple Computer. Pagemaker™ is a trademark of Aldus Corporation.

©1986 by The Cambridge Consortium, Inc. 9/86

The Basic Advisor, Cont'd from Page 1

BASIC/IDOL/BAS. Here goes.

1. Hardware. Only SX processor models with the 2Mb motherboards allow 16 ports. Other models may be upgradable. The full 2Mb memory is probably needed. Also, cabling may be a problem; Superior doesn't believe in trusting a RS-232C signal beyond 350 feet. (That would make for a cozy office, wouldn't it?)

2. Manufacturer issues. Fortune's **BASIC** interpreter is not supported by either Fortune or Concept Omega, so success cannot be guaranteed. Also, there may be a problem with memory allocation with a larger number of terminals running.

3. **FOR:PRO**. **IPL** files and **Tpool** need to be changed (we know that now).

4. **BASIC**. Our version of the interpreter numbers the terminals T0 through T9, then TA through TF (16 terminals.) We believe that the number could continue from TG to TZ (it is not hexadecimal coding), but we can't 100% confirm this area.

5. Operations. Loading other BASIC products or applications, including training discs, subsequent to the IPL file changes may undo the shellscripts. Also, the backup/restore programs would have to be modified or they will not backup the shellscript changes. (A cold boot or disk crash would destroy the changes.)

6. IDOL. Make sure IDOL work files **CUTFT#**, ***REPT#** and ***SRT#** exist for each new terminal (we know that too). Also, handling of selector-return parameters for terminals beyond T7 needs to be addressed (see 7 below).

7. **BAS Applications**. Cash Receipts Entry/Journal and Non-A/R Cash Entry would require program change. Operator Statistics is not a problem because terminal numbers are handled as ASCII strings. Also, IDOL handles saving selector parameters for T0-T7, but does not provide for more terminals. (When exiting a program, TA may return to T7's menu.) Perhaps we could link BASIC terminal assignments to UNIX ports to help solve this issue, thus providing an enhanced security/management control feature.

Thank you, Mr. Rosenfeld, for the input. I intend to have a great time in San Diego and absorb tons of information!

To our readers: I have received several requests to repeat and update the survey of BAS/IDOL support companies which we did last year. This time, I want to give everyone fair warning. I will be gearing up in November to take calls from support companies. If you missed the last listing, keep an eye on this column so you don't miss the 1987 listing. □

Intro to Multiplan, Cont'd from page 1

hand, if you have already worked with a worksheet that you would like to read in, you would use the **Transfer** option. We'll get into that in a bit more detail next month.

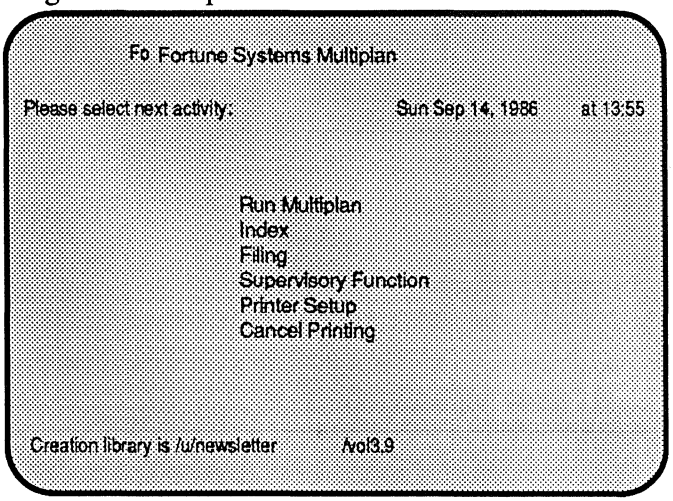
The other way of getting into **Multiplan** is through the UNIX shell. If you like, you simply type the word **master** and you will be presented with the **Multiplan** menu just as you see it when you choose the **P1** option from the **Global Menu**. A second, and sometimes more convenient, method is to use the **mp68** command. If you type this to the UNIX prompt, then you will skip the **Multiplan** Menu and will be taken directly into an empty **Multiplan** worksheet. The most direct way of accessing an already created **Multiplan** worksheet is also to use the **mp68** command. (Note, however, that this method of accessing **Multiplan** will not make available any printing characteristics that you may have provided in the printer setup). Suppose you have already created a **Multiplan** worksheet and you named it *cashflow*. The following command would take you directly into this specific **Multiplan** worksheet:

```
$ mp68 cashflow
```

Getting Acquainted

Take a look once again at figure 2 which is what your terminal screen would look like once you get an empty **Multiplan** worksheet. Notice the **#1** in the upper left hand corner. This number tells you which window of **Multiplan** that you are in. **Multiplan** allows you to open multiple windows and this feature can be quite

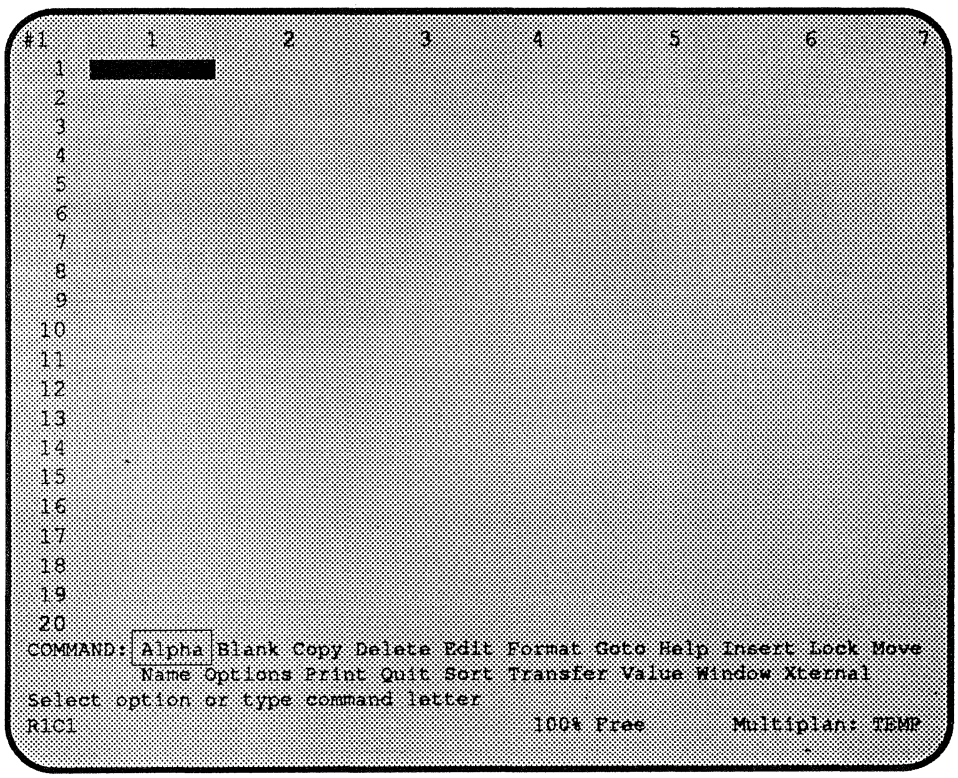
Figure 1. Multiplan Menu



useful in certain situations. When you open a brand new worksheet, as shown in figure 2, **Multiplan** automatically calls it window **#1**. Now look at the numbers that extend across the top. These numbers range from 1 to 7 and represent columns. These are the columns which you can see on the screen at this moment. **Multiplan** allows you to use **63** columns. The numbers down the side, which range from 1 to 20, represent rows. **Multiplan** allows **255** rows.

At the junction of row 1 and column 1 you will see a black box. This represents the *cell cursor*. The cell the cursor is on is called the *active cell*, the cell that can accept information. In this case, the cell cursor is on **R1C1** (get used to this notation because it is used extensively in **Multiplan**). You make a cell active by moving the cell cursor to that particular cell.

Figure 2. Multiplan Worksheet Screen



The two lines right below row 20 are the *command lines*. You'll notice that there are twenty words there (**Alpha** to **Xternal**) in alphabetical order. These are the commands that **Multiplan** accepts. There is a box around the word Alpha and this box represents another cursor, the *command cursor*, which is used to select the command you want to perform. There are two ways to select one of these commands. The first is easy, simply type the first letter of the command and that command will be started. The second ways is to use the **RETURN** key. When you depress this key, the command cursor will sequentially move through the commands. When you get to the command that you want, hit the **EXECUTE** key. (Note: The **SPACE BAR** will have the same effect as the **RETURN** key in this

See **Intro to Multiplan**, Page 5

Our Memory Doesn't Cost You A Fortune. . .

Is Your Fortune Living Up to its Potential?

Would you like to:

Speed up your computer?
Add more users?
Use the latest software, like Fortune:Windows™?

Compatibility:
The F1MB is absolutely 100% compatible with Fortune's own 1 Meg memory board.

Dependability:
All boards are burned in for 72 hours and tested for 48 prior to shipment. *Our One Year Repair or Replace Warranty underlines our confidence in the product.*

Affordability:
USA is proud to have equaled and exceeded the quality and performance of our competition, while offering substantial savings for both the dealer and end user.

Repairability:
The F1MB facilitates repair in the field as it is 100% RAM socketed. Now you can repair a board by simply replacing a defective chip rather than face an expensive board exchange -- reducing maintenance costs and system down time.

If you have a Fortune SX, XP45 or recent model XP30, you can use our board by simply plugging it in. Older systems can be upgraded to use our board with a simple and economical modification.

For Details Call:

Upgrade Service Associates
3 Marion Road, Burlington, MA 01803
Phone: (617) 239-8114

Our Guarantee: If you're not completely satisfied within 30 days, *USA* will gladly refund your money.

Fortune:Windows™ is a trademark of Fortune Systems Corporation

Intro to Multiplan, Cont'd from page 4

case and the **BACKSPACE** key will move backwards along the command lines). You will quickly get used to the first method because it is much faster.

The line directly below the command lines is known as the Message line. The message line is where **Multiplan** proposes the next action for you to take or tells you if you have made an error.

At the left hand bottom corner you will see **R1C1**. This records the current location of the cell cursor and the contents of this cell. Since this is a brand new worksheet, there is nothing in **R1C1**, so nothing is displayed. If you were to move the cell cursor, you would notice that the **R1C1** would continually change so as to indicate which cell the cell cursor is in.

On that same line, approximately in the middle, you will see the *Free indicator*. This records how much of the worksheet is filled. Again, since this is a new spreadsheet, the *Free indicator* tells us that 100% of the worksheet is available.

Finally, in the bottom right hand corner you will see **MULTIPLAN: TEMP**. This area displays the name of the worksheet you are working on. This was a brand new worksheet so **Multiplan** called it **TEMP**.

Opening Up a Worksheet and Moving Around

If you are new to **Multiplan** you might wish to open up a worksheet and try some of the commands as we discuss them. We discussed the two basic ways to start **Multiplan** above. Try one of them and get to the screen that looks like figure 2.

Now, the cell cursor is at position **R1C1**. It's easy to move this cursor around. One way to do this is to use the arrow keys that are located on the standard Fortune keyboard just to the left of the numeric keypad. Try pressing the *right arrow*. You will see the cell cursor move to the right and stop at **R1C2**. Also notice at the bottom left hand part of the screen that the cell indicator has changed to **R1C2**. Try the different arrow keys so that you get a feel for moving the cell cursor.

There are other ways of moving around the worksheet besides using the arrow keys. Some of the Function keys on the Fortune Keyboard have been defined to make certain types of moving easy (see the sidebar on the Function keys and what they do in **Multiplan**). Function keys 13 through 16 allow you to move an entire screen at a time. For example, if you press **F15** you will notice that you have moved down exactly 20 lines (you may have to look carefully at the numbers on the left hand side of the screen to notice this if you are working

See **Intro to Multiplan**, Page 6

Our disks just keep piling up! And that's good for you.

We recently introduced two new disks. At the same time we offered you all six diskettes for $50. We are extending that offer. But in addition, we are offering you your choice of five for $42.00 and your choice of 4 diskettes for $33.00. Don't wait to take advantage of this inexpensive way of increasing your stock of Fortune software!

Send your requests to us at:
The Cambridge Consortium, Inc.
73 Jason Street
Arlington, MA 02174

Intro to Multiplan, Cont'd from page 5

with an empty worksheet).

F12 is the **Home** key which has the effect of moving the cell cursor to **R1C1** immediately - no matter where you are in the worksheet. **R1C1** is known as the *Home position*. **F11** is the **End** key and it has the effect of moving you to the last cell in the worksheet that has something in it. With a brand new worksheet, the End key will have no effect.

The last main way of moving around is to use the **GOTO** command. The **GOTO** command is used when you want to move to a particular cell in your worksheet but it would be inconvenient to move there with any of the other methods explained. For example, suppose you wanted to move to row 173 and column 45 (R173C45). You would use the **GOTO** command.

Hints for the efficient use of Multiplan

The following information is taken from a book entitled *Managing Your Business With* ***Multiplan*** which was written by Ruth Witkin (Microsoft Press, copyright 1984). One of her chapters discusses some suggestions for approaching the solution of a problem with **Multiplan** in a systematic manner.

First, she begins by noting, as does the Fortune **Multiplan** manual, that **Multiplan** is used to help you with a *model*. A model is a set of loose ideas that hopefully will help you solve a problem. A simple example of a model is a check book register. It solves the problem of keeping track of how your money comes into and goes out of your checking account.

Step one, then, is to define your model. This is best done on paper. Obviously, you don't solve the entire problem on paper because then you wouldn't need **Multiplan**. But you should begin the planning process on paper. It is at this point that you should begin planning things such as titles, how long they should be, and what formulas you will use. Once you have worked these things out in rough detail you can start **Multiplan**.

Ruth Witkin suggest a series of steps that should help you in developing the **Multiplan** worksheet. One of the first things you should do is turn off the *automatic recalculation*. Whenever you enter anything into a cell, **Multiplan** will automatically recalculate everything in the worksheet. When your worksheet is small, this may not seem to be a problem. However, if your worksheet is of even a moderate size, you sometimes wait a long time for the sheet to recalculate. Turning off the recalculation is

Continued on next page

Participate in the PROGRAMMERS" GRAB BAG from SST DATA, INC.

It's Absolutely FREE!!!

While developing our HANDSHAKE software, the SST Data staff created a number of utility programs for Fortune computers. We are offering you an opportunity to get printouts of two utilities chosen at random from our list of ten small programs created by our development staff. Just fill out and return the form at the right.

4701 W. SCHROEDER DR., SUITE 100
MILWAUKEE, WI 53223
• 414/355-6990

To: SST Data, Inc., 4701 W. Schroeder Dr., #100, Milwaukee, WI 53223

From:
Name: _____ Title: _____

Phone: _____
Company
Name: _____
Street
Address: _____
City &
State: _____ Zip: _____

YES Please send me a free utility for my Unix operating system from your PROGRAMMERS' GRAB BAG. I plan to install it on:

_____ (computer make, model)

Other computer systems currently in use at this firm include:

_____ (make, model)

_____ (make, model)

very easy and is part of the Options command. Also, even with the automatic recalculation turned off, you can make the sheet recalculate by pressing the **F7** Function key.

The next step is to adjust your column widths. If you've done some preliminary planning, then you'll have a good idea of how wide your columns should be. **Multiplan** automatically sets the width of the columns to 10 characters, but you can make a column be as narrow as 3 characters or as wide as 32 characters. Changing the size of a column is used mainly to make the appearance of your worksheet pleasing to the eye and a nicely spaced worksheet is much easier to read.

Step 3 is to enter any titles that will not change. You should know what your titles are because you have worked this out on paper. Your titles can be anywhere in the worksheet and are entirely dependent on your design. To enter titles you will use the **Alpha** command, because it allows you to put text into a cell. One reason to put all the titles in first is because once you use the **Alpha** command you remain in this mode and can move around your sheet filling in titles without having to select the **Alpha** command again.

Step 4 is to enter dashed lines. Dashed lines are used to underline titles and to set off parts of the worksheet from one another. The easiest way to put dashed lines in is with the use of the **REPT (REPeaT)** function. We will illustrate this in the example worksheet we present next month.

Step 5 is to format the titles. Text is automatically left-justified when you enter it but **Multiplan** allows you some flexibility here. For example, you can center text in a column. If your titles run across a column, you may want to use the format continuous option - more about that next month.

Step 6 is to format the number cells. Numbers are automatically right-justified unless you instruct **Multiplan** to do something differently. There are many number formats and we will show you a couple next month. For more information on other formats, you should consult the **Fortune Multiplan** guide page 3-9 to 3-12.

Step 7 is to name the cells the you want to be named. Naming cells allows you to build consistent and easily understandable formulas. It is also best to name all cells at one time, so you establish a regular routine for developing your spreadsheet.

Step 8 is to name and save the worksheet. Up to this point, you have been working with a worksheet that resides in the memory of the computer. You need to save this to the disk so that you can retrieve it at some later point in time. To save a worksheet you use the **Transfer** command and then the **Save** sub-option and then give it a name. After you press the **EXECUTE** key, **Multiplan** will save the current worksheet into a file for you on disk.

Step 9 is to enter some test or real numbers. The real guts of your worksheet will be the formulas you have developed. However, formulas must work on numbers and you should put some test numbers in the proper places before you enter your formulas. Also, you might consider entering some simple numbers that will allow you to check your formulas for accuracy.

Step 10 is to enter the formulas. Formulas are mathematical and/or logical operations based on the contents of one or more cells in your worksheet. Thus, one important component of building formulas is how to refer to cells. There are a number of ways of doing this including *absolute reference* and *relative reference*. The details can not be presented here, however, you will get a feel for this topic by working with us through our examples in subsequent articles. We also refer you to the **Fortune Multiplan** guide, pages 4-14 to 4-32.

Step 11 is to copy the formulas to their appropriate locations.

Step 12 is to lock the formulas. Locking formulas protects them from accidental change. If you happen to inadvertently type a number into a cell that contained a formula, then **Multiplan** will erase that formula and replace it with the number. This is often *not* what you want.

Step 13 is to save the worksheet again. Now that it contains your formulas, you should save it so that you do not lose your work.

Step 14 is to enter the real numbers. After your formulas have been entered, it is time to enter the real data.

The final steps include specifying your print parameters, printing the worksheet and the formulas and finally, of course, exiting **Multiplan**.

These steps should give you a good set of guidelines for developing **Multiplan** worksheets efficiently and effectively. Beginning next month we will work through some worksheets in order to further define and explain Multiplan's capabilities.

We know that many of our subscribers use Multiplan extensively and consequently we are sure that there must be some useful and interesting worksheets that you have developed. We ask you to send them along to us, along with instructions on their use. In return, we will package them up on a diskette and make them available to anyone who is interested. □

Mark Palmerino

Letters

Fortune:Word Legal Users Unite!

Through a somewhat circuitous route, we recently received this letter which should be of interest to our subscribers who are lawyers:

Mr. Morris Frydman
JFM Business Systems

Dear Morris:

Enclosed is a copy of a Word Processing Bulletin recently published by the ABA Section of Economics of Law Practice.

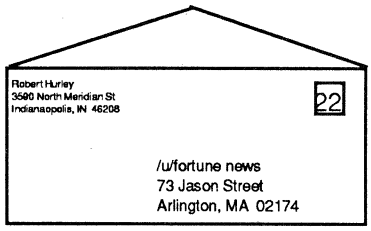
You will note that they have formed a word processor user's groups for CPT, IBM, MICOM, SYNTREX, XEROX, DICTAPHONE, LANIER, NBI, and WANG.

We were thinking about responding to their request that we join one of these groups by suggesting that they form a **Fortune:Word** processing group and that we would volunteer the group initially. However, I doubt that the ABA should be interested in supporting such a group unless there was an installed base of several hundred law firms using Fortune. I would think that such a group would be beneficial to Fortune and all of its dealers. Is there any way that we can pry loose from Fortune some data on their installed base in law firms, and if it is large enough, perhaps a mailing list for announcements and invitations to join the group?

Very truly yours,

Robert L. Hartley, Jr.
Martin, Wade, Hartley & Hollingsworth
3590 North Meridian Street
Indianapolis, IN 46208
(317) 924-6700

If you are interested in participating in such a group, we encourage you to call Mr. Hartley and express your support. There could be great benefit from sharing glossaries, training techniques, etc.



PSG

We are the *largest* Authorized Fortune Dealer in the Pennsylvania, New Jersey and Delaware area. As part of the **P**rofessional **S**olutions **G**roup of companies with over 35 years in business automation, we have the *stability* that our customers demand

Our services include a factory trained field engineering team, a complete parts inventory including system upgrade capability and a software support department.

For all of your Fortune needs contact:

Mark Shields of PSG at
2550 Eisenhower Avenue
PO Box 762
Valley Forge, PA 19482

or you can call Mark at (215) 666-1400

Dear Fortune Owner:

As you have probably read in */u/fortune news*, Fortune Systems SX models are significantly more powerful and faster than the PS and XP models. If you have a PS or XP model system, you should take advantage of our special promotion on SX upgrades. For a limited time, and while our supplies last, we will upgrade your system, including the new SX Cabinet at no additional charge for the cabinet. With this special offer, you may choose any combination of the following upgrades and/or expansions:

- *SX Processor with Current Release of Multi-User For:Pro*
- *60 MB Internal Streaming Tape*
- *30 MB, 45 MB, 70MB High Speed Hard Disk*
- *6-Port Intelligent Asynchronous Controller*
- *Upgrade to Current Release of all Fortune Software*
- *Fortune:Windows (Included at NO Charge)*

Additionally, we can perform these upgrades within 24 hours of receiving your system. In fact, if you ship your system overnight on a Friday for Saturday delivery, we will have it back to you on Monday morning, all ready to plug in and run; not losing the use of your system for a single work day. And, we will issue you a full 90-day warranty on the entire system.

JFM provides services to Fortune Systems owners nationwide either direct or through our authorized Affiliated Dealers. If you haven't used our services, I invite you to discover JFM; you will be glad you did.

Sincerely,

Morris Frydman
President

Talk to us, then talk to our clients

The Merchandise Mart
Lobby Suite 144
Chicago, Illinois 60654
Call (312) 644-0870

Authorized Fortune Systems Master Dealer

/u/help

MEMO
RE: Fortune 32:16, start-up; Eliminating need to enter date and time after crash
DATE: August 20, 1986
FROM: Mark S. Anderson
Integrated Professional Services, Inc.
11 East Cliff Street
Somerville, N.J. 08876

The purpose of this Memorandum is to address the problem that the standard Fortune 32:16 system will not re-boot itself into operation after a power failure. The result of even a brief power outage is that someone must go to the console and enter the date and time before operation can be restored. This is primarily of concern to those who use their systems at some distance from the main console, for instance from a dial-up terminal.

The Fortune computers seem to be quite tolerant of power failures. After any "crash," whether because of a power outage or otherwise, the machine will automatically do a file system check and will automatically repair any minor file system inconsistencies. However, the machine will not come up until the current date and time are manually inserted at the console. The only apparent purpose for this is that it keeps the computer's date and time correct. The disadvantage is that it requires intervention at the console.

The solution is conceptually simple. One of the "shell-scripts" which controls the machine is modified so that after a crash the machine proceeds through the start-up sequence without requiring that the date and time be entered. At the same time, a message is appended to the "message of the day" file to inform subsequent users that the date and time should be corrected. The modification may be accomplished using any of the usual text editors, including **ed**, **sc**, **screen**, or **vi**. The modification works like this:

1. If the machine is being brought up after a regular "shutdown" routine, there is no change.

2. If the machine has gotten into the start-up routine without having gone through a shutdown, the machine goes straight through start-up to the login message without pausing to ask for the date and time.

The shell script is in the file **/etc/rc**. This shell script controls the progress of the start-up from about the time that the **9** appears on the screen until about the time the login message appears. The area which needs to be changed is lines **56** through **63** of **/etc/rc**, reproduced below. Note that the line numbers are **not** part of the shell script, but are only for reference.

```
56 /etc/setdt < /dev/console > /dev/console 2>&1 # e
57
58  sync
59  if test -r /etc/.fsclean
60  then
61      rm /etc/.fsclean
62      operation=up
63  else
64      operation=restart
```

Line 56 calls the program **/etc/setdt**. This is a **For:Pro** command which allows the date and time to be set by the system manager. While it is probably most familiar as a part of the start-up routine, it can also be run as a **For:Pro** command or it can be run by choosing "s2 system management" from the Global Menu and selection "36 set date and time" from the system management menu.

Line 57 is blank. Line 58 issues the **For:Pro sync** command to ensure all disk writes have been completed.

Lines 59 and 60 are the start of an "if" statement, testing for the existence of the file **/etc/.fsclean**. This file is created as a part of the shutdown routine, and is removed whenever the machine is brought back up. Thus, this file will exist if the machine is being started up after a shutdown, but will not exist if the machine is being started up after a crash. This is one of the keys to the modification.

Lines 60 through 62 are one branch of the "if" statement.

See **Shutdown**, Page 22

VANTINE COMPUTER SERVICES

We Are Not The
Biggest Biggest Biggest

Nor The
Cheapest Cheapest Cheapest

We Simply
Care Care Care

VCS is an Authorized Fortune Systems Dealer. We understand and support the products we sell.

We provide sales, upgrades, service and support to our Fortune Clients from Maine to California.

We don't ask you to believe us. We want you to believe our clients. Call for references today.

Vantine Computer Services can provide the solution for you!

VANTINE COMPUTER SERVICES
9 Priest Road, Sudbury, MA 01776
(617) 443-8412

Authorized Fortune Dealer

JONES BUSINESS AUTOMATION CENTER

a *FORTUNE SYSTEMS MASTER DEALER*
and the
JBAC AFFILIATE DEALER NETWORK
are organized to
Support FORTUNE END-USERS with
Outstanding Professional Systems Support
at **Reasonable Prices**

Inquire about the JBAC "PERFORMANCE SOLUTION." Upgrade your present system to a new SUPER PERFORMANCE SX-70 which is the PRICE PERFORMANCE Leader.

* **BAS/IDOL SUPPORT**
* **FORTUNE:WORD**
* **MULTIPLAN**
* **CUSTOM DEVELOPMENT**
* **COMMUNICATIONS**
* **HARDWARE UPGRADES**
* **MAINTENANCE**
* **LASER PRINTERS**
* **OCR PAGE READERS**
* **COMPUTER NETWORKS**

The JBAC Affiliate Dealer Network

Synergistic Business Systems 4300 Paper Mill Road Knoxville, TN 37909 (615) 584-0211	**Advertising Data Mgmt.** 4173 First Avenue Tucker, GA 30084 (404) 938-8100
MicroAge Computer Store 1939 Poplar Avenue Memphis, TN 38104 (901) 722-8280	**Brinc Business Systems** 1662 Savannah Highway Charleston, SC 29407 (803) 763-0599
Software Sciences 3100 Evans Avenue Fort Meyers, FL 33901 (813) 334-1646	**DataLogix** 200 Central Park Avenue Hartsdale, NY 10530 (914) 997-1627
DataLogix 3000 Dundee Road Northbrook, IL (312) 291-9225	**DRS Enterprises** 127 E. Murphy Street Madison, NC 27025 (919) 427-0864
MultiUser Systems 10818 N.W. 6th Court Miami, FL 33168 (305) 758-2007	**Software Engineering Assoc** 301 East Greenville Street Anderson, SC 29621 (803) 225-0101

JONES BUSINESS AUTOMATION CENTER

834 Tyvola Road, Suite 112
Charlotte, NC 28210
(704) 525-0582

The UNIX Directory

INTRODUCTION TO SHELL SCRIPTS PART 2

David E. Kloes *is President of UNI-KOMP which is located in Houston, Texas. He provides UNIX seminars, writes software for the Fortune Computer and is vice-president of the Houston UNIX User's Group. He is contributing independently to* **/u/fortune** *news.*

In Part 1, we introduced you to some of the basic concepts of shell scripts. Part 2 will continue with these basic concepts, concentrating primarily on shell commands. In this part, we will occasionally wander away from the examples at hand to explain new shell concepts.

A shell command can be either a simple command (one command line) or a combination of commands which work together (multiple command lines). In this part we will talk about the combination commands.

THE *for do done* LOOP

```
for <name> in <word>
do <list>
done
```

First of all, we must point out that whenever this loop is used, it **must** contain the words *for*, *do* and *done*. This loop is used to perform actions on a group of items (words). Basically, it says the for each of these "words", we want to do this "list" of things until we are "done". As each "word" is acted upon, its value is assigned to the variable "name".

As a simple example to explain the concept, let's say that we want to write a shell script that will create a screen that looks just like the Fortune system countup sequence (we're only going to use the first 5 numbers) when we power the machine on. Let's try this for starters:

```
for count in 1 2 3 4 5
do echo "$count"
done
```

Create a shell script called "count"; enter this sequence of commands using your favorite UNIX editor; change the permissions to make it executable (chmod +x count); and enter the command count at the UNIX prompt. Your output should look like this:

```
$ count
1
2
3
4
5
```

Let's talk about what the shell script is doing. The first time through the loop, the variable *count* has the value of "1". Our *do list* in this example just has one thing to do and that is to echo the value of the variable back to the terminal screen. Since there is another *word* to be worked on, the loop continues by giving the variable *count* the value of "2". This is also echoed to the screen. The process continues until the last *word* (5) is done. At this point, the program would continue with whatever shell command followed the **done** line. Since there are no other commands, the program terminates. Note that the list of *words* could be anything including pre-defined variables, our variables, words, numbers, etc. Example:

```
x="junk"
for name in 1 apples `tty` $x $HOME
do echo "$name"
done

1
apples
/dev/console
junk
/u/davek
```

The **for, do, done loop** is more commonly used in shell scripts, for example, to take action on a group of files. Consider the following example that would search the entire system for all of the **core** files produced by "core dump" messages:

```
for name in `find / -name core -print`
do rm $name
   echo "$name removed"
done
```

The **find** command would produce a list of files named **core** (refer to Part 1 to refresh your memory on the meaning of the backquotes (`)). The **do list** has two actions - remove the file and echo back the pathname of the file it removed.

Let's continue on with our countup sequence example. We have now produced the countup sequence from 1 to 5, however, the numbers are listing down instead of horizontally. This was by design, so we can introduce you to a new concept.

The **echo** command normally produces a line feed/carriage return each time it is used. In fact, an echo all by itself is nothing more than a carriage return. The **-n** option when used with the **echo** command, tells the system not to produce the carriage return. This will be especially useful in our shell scripts, when we use the echo command to prompt a user for input and we want to leave the cursor at the end of the line so they can enter their information. With this bit of knowledge let's try this:

```
for count in 1 2 3 4 5
do echo -n "$count
   echo -n
done
```

See **The Unix Directory**, Page 14

SOFTWARE FOR THE FORTUNE 32:16

MULTIPLAN LINK - can pass data from BAS/IDOL/BASIC files to Multiplan and back. For example, it could be used to send data from any of the BAS files (Chart of Accounts, Customer Master, Inventory Master, etc). Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

RECORDS PROCESSING LINK - can pass data from BAS/IDOL/BASIC files to Records Processing and back. For example, it could be used to take customer address data from the Customer Master and will automatically make the Records Processing List document for mailouts. Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

TERMINAL/PRINTER LINK - provides capability to print Multiplan, and BAS/IDOL reports on a printer connected to a workstation, view any report on the terminal before printing, convert any report to Fortune:Word format; accumulate multiple BAS/IDOL reports before printing, backup reports onto diskette; copy or rename report files; print multiple copies of the report on either the primary and/or terminal printer. Also has option to print BAS/IDOL reports on HP LASER printer ($50 extra) **$195.00**

TELEPHONE LINK - on line telephone message, telephone directory and inter-office mailbox/memo system which can also be used as a simple calendar reminder system. Provides control, printing and viewing of on-line data for each user. **$295.00**

FINANCIAL LINK - stores BAS Income Statement and Balance Sheet data in files. Once stored, it can be sent to Multiplan or instantly printed. **$195.00**

MENU LINK - provides software and documentation for adding items to the Global Menu as well as a sub-menu for your use Includes backup script for user directories and BAS/IDOL/BASIC programs and data. **$95.00**

CALCULATION LINK - provides programs for amortization, depreciation, loan repayment, averaging (with graph), linear correlation, and breakeven analysis. **$95.00**

KOMPACT PERSONNEL ACCOUNTING - provides for data capture of personal, wage/salary, job, education, salary/wage, appraisal training, dental, medical and life insurance data. Includes over 80 pre-defined reports/worksheets. **$495.00**

CRT LINK - provides information for using other terminals on a Fortune System (WYSE, TV, IBM 3101). **$95.00**

AUTOMATIC REPORT LINK - provides capability to print numerous IDOL defined reports automatically based on frequency or application without having to select each from a different menu. You identify frequency/application codes. **$195.00**

TAPE LINK - provides one menu for all tape backup utilities. Provides daily, weekly, monthly and individual backup and restoration routines. Includes rewind, retension, and tape unit startup/stop utilities. User friendly. **$95.00**

PRINTER LINK - provides printcaps for Okidata 84, 2410, TI 855, and Anadex 6500. Includes shell scripts for changing pitch and LPI. Allows these printers to be used in word processing. **$95.00**

MODEM LINK - allows Hayes 1200 Smartmodem or compatible to be used on one port for both incoming and outgoing calls. **$95.00**

INSTALL LINK - provides documentation and software example for installing your product/menu through "S5". **$95.00**

REPORT LINK - modifies IDOL Report Generator Program to allow stacking of up to 8 fields in an IDOL defined report. Also corrects problem with displaying subtotals for all stacked fields. **$95.00**

WESTERN LINK - provides instructions on how to use your Fortune for sending/receiving telex messages using Western Union's Easylink. Can replace your telex! Fortune ITE software is required. **$95.00**

BACKUP LINK - provides menu selection for backing up /b (BAS/IDOL/BASIC), /u (Fortune:Word, Multiplan) and entire system onto diskette. Also has program to "unhang" a terminal that's hungup just by entering the terminal #; a program to restore one file from a multi-volume backup; and a selection to tell you which volume a backed up file is on. **$95.00**

ASCII LINK - can pass data from BAS/IDOL/BASIC files to ASCII and back. For example, it could be used to send data from any of the BAS files (Chart of Accounts, Customer Master, Inventory Master, etc). Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

UNI-KOMP 9803 Rocktree Houston, Texas 77040 (713) 937-7782

The Unix Directory, Cont'd from page 12

This script would produce the following results:

```
$ count
1 2 3 4 5 $
```

Here the first **echo** command will echo the number to the terminal without the carriage return. The second **echo** will put a space between each of the numbers - again without the carriage return. One remaining problem however, is that the UNIX **$** prompt appears after the number 5 instead of on the next line. We will use this problem to introduce you to the next shell command. Before we continue on, however, we need to explain three more concepts. First, note that the two **echo** commands are under each other and that the for, do, done are aligned. It is good shell programming practice to use alignment and indentation to make the script more readable. When our shell scripts get more elaborate and have more than one multiple command, it is easier for us (and someone else) to read them if we keep logical parts aligned. We will hold to this standard in our examples so that you can see how each of the lines should be aligned.

Second, the use of the two **echo** commands is an example of poor shell programming. Why use two echo commands when we can use one and produce the same results? The script is more appropriately written:

```
for count in 1 2 3 4 5
do echo -n "$count "
done
```

We have seen shell scripts that use one echo command for each line of a menu. It works, but the time it takes to paint the screen is very slow since the shell must execute a whole series of echo commands instead of one.

POOR PROGRAMMING:

```
echo "                My Menu"
echo
echo "          1.  List files"
echo "          2.  Change directories"
echo "          3.  Exit menu"
```

GOOD PROGRAMMING:

```
echo "                My Menu

                1.  List files
                2.  Change directories
                3.  Exit menu"
```

One last point before moving on to the **case** command. Even though we are putting commands into a shell script, it is sometimes easier to test them directly from the UNIX prompt. This is especially useful as a debugging tool and is quicker than firing up an editor, making the change and exiting to test the command. If we were to type our **count** script directly into the shell, it would look like this:

```
$ for count in 1 2 3 4 5
> do echo -n "$count "
> done
1 2 3 4 5 $
```

Notice that the prompt changed to **>** for the *do* and the *done* lines. This is the shell secondary prompt (remember **PS2**?). The shell displays the secondary prompt whenever we enter a multiple command where the shell expects further input. When we entered the *for* line, the shell knows there has to be a *do* and a *done* to complete the sequence and therefore gives us the secondary prompt of **>** instead of **$**. Have you ever been typing commands into the shell and gotten the **>** prompt? This occurred because you accidently hit a sequence of keys where the shell expected some other input. If this ever happens accidently, simply depress the **CANCEL/DEL** key to return to the primary prompt. As a special note, the **CANCEL/DEL** key is used to terminate any shell program. Because of this, the **CANCEL/DEL** key creates some problems for shell programs which we will address later.

THE *case, esac* COMMAND

```
case <word> in
      pattern) list;;
      pattern) list;;
esac
```

This command sequence says that if the *word* matches this *pattern* then do this *list* of things. The sequence begins with the word **case** and ends with the word **esac** which is case spelled backwards (we will see this technique used again in other command sequences). Basically, this sequence is used to take action whenever a given condition is met and many times is interchangeable with the **if** command which we will discuss later. It is most commonly used for yes/no responses, and to determine actions based on selections from menus.

In our *count* example, we wanted to get the $ prompt to appear on the screen after the countup sequence. We will use the **case** command (and later the **if** command) to do this as an example of how the command works.

```
for count in 1 2 3 4 5
do echo -n "$count "
   case $count in
     5) echo;;
   esac
done

$ count
1 2 3 4 5
$
```

Notice first of all that we have aligned the *for, do, done* and have also aligned the *case, esac.* Indentations were made to make the script more readable. For each number in the loop, we are evaluating it. If the number is 5, we will do a bare echo which is simply a carriage return/line feed. Notice that the 5 in the **case** is followed by a) and ended by a double semicolon (;;). Common errors are forgetting the ;;, or the *esac.* Also multiple cases may be evaluated at the same time. Let's look at this example:

```
case $answer in
      yes|y|Yes|YES) answer="yes";;
         no|n|No|NO) answer="no";;
                  *) echo "You must answer
'yes' or 'no'"
                     sleep 2;;
esac
```

This would be typical use in a shell script where the user must answer a "yes/no" question. This case sequence would test for each of the possible answers a user may enter. Notice that each case is ended with its own ;;. We are also introducing you to the use of shell metacharacters in your shell programs. In the case command the use of the "|" means "or". The "*" in the last case means "any other pattern". Each case is evaluated until it meets a condition. The list of things for only that condition is then executed. Control then passes to the statement after the esac. To emphasize this point, what would happen if the *) condition was the first pattern in the case? The answer is that all cases would meet this condition and none of the other patterns would ever be tested. The list of things to do would always be associated with the "*)" case.

For our yes/no example, if the user does not enter one of the yes or no patterns, control falls to the*) case and they will get the error message. The **sleep 2** tells the program to pause for 2 seconds so the user can read the error message.

One last comment before wrapping up this session. If you have not already done it, **DO NOT NAME ANY OF YOUR SHELL SCRIPTS** "test". This is a UNIX command and when you enter **test** to run your shell script, UNIX will execute the **test** command. The only result on your screen is that your prompt will appear on the next line. New shell programmers have spent many hours trying to figure out why their shell script would not run.

In Part 3, we'll see how to get the prompt after the 5 in our **count** script to print on the next line using the **if then else fi** command sequence. We'll also spend some time talking about the **test** and **expr** commands. □

FORTUNE TRAINING AT ITS BEST!

COMPUTER TRADE DEVELOPMENT CORPORATION (CTD), Fortune System's largest Master VAR, offers the most comprehensive repertoire of Fortune applications training available.

23 full-time certified instructors and an impressive facility, featuring eight classrooms fully-equipped with one terminal per student and Hewlett-Packard LaserJet printers ensure quality education. And because class size is limited to ten, you are assured of maximum instructor-to-student attention. Courses are targeted to all experience levels:

Meet Your Fortune System	*Fortune:Windows*
System Administration	*Using Your HP LaserJet*
Fortune:Word, 1-Day Overview	*Introduction to Multiplan*
Fortune:Word, 4-Day Intensive	*Multiplan: Spreadsheet Design*
Advanced Fortune:Word	*Advanced Multiplan*
Glossary Programming	*FOR:PRO for End Users, Intensive*
Fortune:Word Mini Database	Practical FOR:PRO, 1-Day Overview...

...and many more, including C, the vi screen editor, and FOR:PRO shell programming. The company that customized Fortune training for Ford Motor Company now makes the best training available to you! (Onsite classes can be arranged, if you prefer.)

RESERVE YOUR SEAT TODAY!

1-313-271-3020

COMPUTER TRADE DEVELOPMENT CORPORATION
One Parklane Boulevard, Suite 1404 East
Dearborn, MI 48126

Moving Around Fortune:Word

I've been sitting around for the last several days thinking about what new and exciting things I could explain about **Fortune:Word** this month and I kept coming up with blanks. As I meditated on my word processing use, including any tricks I know of, I suddenly remembered one of my favorite pastimes -- reading instruction manuals.

If you're like me, you probably turned on your computer and starting using **Fortune:Word** immediately. You consulted the manual, but most of it didn't make any sense, and in any case, you quickly learned that you could get along just fine without mastering hundreds of pages of instructions. In my case, that was about three years ago. Tonight when I opened the manual I could quickly flip through about 3/4's of it because I already knew what it contained. But this time I found lots of shortcuts and special features that are built into the program that now made sense to me.

Some of the more helpful things I found deal with the great variety of ways there are to move around within a document. It's possible to move by a character, word, sentence, paragraph or a page at a time. It's also possible to jump precisely to specially marked places within your text. If you don't want to take the time to peruse Section 5 of your manual, we'll endeavor to present you with the fundamentals right here.

Using the MODE Command to Adjust Speed Moving

Anyone who has used **Fortune:Word** for more than 7 seconds knows that the cursor (arrow) keys will move you a character at a time in any direction you please. If you hold down the **SHIFT** key and hit either the **UP** or **DOWN** arrow key, the cursor will jump either forward one word (**SHIFT DOWN**) or backword one word (**SHIFT UP**). This is indicated by the word "word" which probably appears on the far left of the second line of your screen. If you hit the **MODE** key followed by a **PERIOD**, that indicator will change to the word sent, which stands for sentence. Now when you hit **SHIFT DOWN**, the cursor will jump to just after the next period, which should be then end of a sentence. Pressing the **MODE** key followed by the **RETURN** key changes the cursor mode to the paragraph mode, and pressing **MODE** followed by the **F2/PAGE** key changes it to page mode. In any of these cases, pressing **SHIFT UP/DOWN** arrow will jump in whatever increment you have selected with the **MODE** command.

Using the GOTO key

As you might imagine, the **GOTO PAGE/F16** key moves you to any page in your document. You also may know that if you hit **GOTO B** you will be at the beginning of your document and **GOTO E** brings you to the end. You can also use the letters **H**, **F**, and **W** to go to the **Header Page, Footer Page,** and **Work Page** respectively. If you want to quickly go to the top of the next or previous page, you can use the **GOTO PREV SCRN** or **GOTO NEXT SCRN**. **GOTO UP/DOWN** arrow takes you to the top or bottom of the page you are working on. Finally, **GOTO LEFT/RIGHT** arrow takes you to the left or right margin of the line you are working on.

Using GOTO with Bookmarks

As Fortune states in their manual, sometimes you might like to just slip a bookmark into a particular place in your document. **Fortune:Word** allows you to "slip" 10 bookmarks into your text. Once inserted, you can very quickly jump to exactly one place or another within your document. This is especially useful for large documents where you need to get around quickly. One idea would be to put a bookmark at the start of each chapter. Then, without knowing the page number of each one, you could immediately move from one to another.

Inserting bookmarks is an easy task. Simply place the cursor where you want the mark to appear, and then press **COMMAND (F13) # NOTE (F7)**. Instead of actually pressing the number sign (#), you can press a number from 0-9. These correspond to the 10 bookmarks you can set.

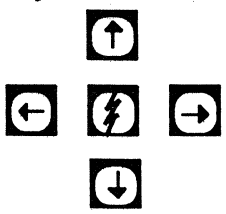
To locate a bookmark, just press **GOTO # NOTE**. Again substitute a number from 0-9 for the number sign. Of course before using this command, you must first assign the bookmarks.

What about Vertical Scrolling, you ask?

Each time you reach the bottom or top of a page, the screen shifts to a new section of text. Generally the screen will scroll 14 lines at a time, preserving some of the text that was already on the screen. If you like, you can change this to any other number between 1 and 20. You accomplish this by pressing **COMMAND # V**, where the number sign is a number from 1-20 and **V** is the letter **V**. The easiest way to test the effect of this is to try it out. Once you have reset the **Vertical Scrolling**, move the cursor past the beginning or end of the screen, and you will see you have changed the scrolling amount. This can be useful when it is important to see a specific block of text grouped together.

Hopefully this information will be useful. Next time you find yourself with nothing to do, and you've exhausted **/u/fortune** *news*, try perusing the **Fortune:Word Operator's Guide** -- you won't be wasting your time. □

Josh Lobel



Fortune Systems Takes to the Road

This fall Fortune Systems is making a nationwide tour to introduce a major new generation hardware product along with over 25 other new products. These new products will meet the needs of users who desire greater processing power than the Fortune 32:16 can provide. Although there wasn't any concrete information available at press time about this product, it is rumored that it is a completely new computer that will support many more users than the 32:16 family.

The tour will stop in the following cities on these dates:

San Francisco	October 3rd
Atlanta	October 13 & 14
Chicago	October 16 & 17
New York	October 21 & 22
Boston	October 23 & 24

In addition to their products, we expect that many of the people at Fortune who shape the company will also be attending.

Attendance to the demonstrations is by invitation only. If you are interested in seeing Fortune's latest offerings and haven't received an invitation, call your dealer or Mary Olivera at Fortune Systems for more information.
(415) 598-4510

News from Fortune

New Tape Expansion Cabinet Series

Fortune Systems announces a new series of Tape and Disk Expansion Cabinets, including a 60 MB Streaming Tape Expansion Cabinet. Fortune users now have the capability to decrease file back-up time by using a 60 MB streaming tape cartridge as opposed to a 20 MB tape. The 60 MB tape expansion cabinet is intended to replace the present 20 MB tape expansion cabinet.

The 60 MB Tape Expansion Cabinet features a 1/4 inch streaming tape in a 5-1/4 inch package. The drive is designed to allow selective access to data. It also performs software distribution, program loading, data collection and archiving.

The expansion cabinet is available in a 70 MB hard disk configuration with upgrade kits to include either an additional 70 MB hard disk or a 60 MB streaming tape drive.

The 60 MB Streaming Tape Upgrade Kit and the 70 MB Hard Disk Upgrade Kit for the "new" expansion cabinets, listed above, are not interchangeable with any of the earlier expansion cabinets -- previously sold expansion cabinets were produced from a different mold than the "new" expansion cabinets. □

Looking for Business Accounting Software that will conform to your Business?

Look to DSM

Applications
- Can be tailored to your organization's specific needs
- Economy of Standardization
- Follows Generally Accepted Accounting Principles (GAAP)
- Luxury of Adaptibility
- Data Base Incorporation

Software for:
- Not-for-Profits
- Construction
- General Business
- Property Management
- Steel Service Center
- Law Office Management

Hardware
Fortune Systems, NCR, AT & T, IBM

Call or Write
Data Systems Management, Inc.
277 Coon Rapids Blvd.
Minneapolis, MN 55433
(612) 780-5200

All software was developed using PROGRESS, a powerfull relational data base which allows you to tailor the software to your specific needs.

DSM PROGRESS

Dealer Inquiries Invited
PROGRESS is a trade mark of Data Language Corporation.

BACKUP AND RESTORE UTILITY

PROTECTION MADE EASY!

- o Back up all or selected user directories
- o Restore all or selected user directories or files
- o Format diskettes
- o Review backup volumes
- o Menu-driven, easily installed
- o Complete documentation

Also copies the files necessary to preserve your system configuration--a tremendous help if your system ever needs to be cold booted!

$150.00	Backup/Restore Utility
$200.00	Utility + 2 boxes 5 1/4" 96 TPI high-quality diskettes

For more information, call CTD at (313) 540-1282.

To order, make check payable to *Computer Trade Development Corporation.* Send to:

BACKUP/RESTORE UTILITY
Computer Trade Development Corporation
Suite 1386
Birmingham, MI 48010

CTD

Company:_____
Contact:_____
Address:_____
City:_____
State:_____ Zip:_____

User Spotlight

'When you wish upon a star, makes no difference who you are, When you wish upon a star your dreams come true'

Walt Disney

Those of us who think this song is just a jingle have yet to meet Ray Esposito and Dick Randall. For these men, the founders of **The Brass Ring Society** and the **Remme Foundation**, respectively, there just couldn't be truer words around. This is because both are in the business of making dreams come true: for Ray Esposito this means fulfilling the dream -- any dream -- of a terminally ill child anywhere in the U.S., Canada or Mexico; Dick Randall provides transportation and recreation for mentally and physically handicapped people and the elderly in the Boston area, things that others of us take for granted. We recently talked with these men to see how, with the aid of their Fortune computer systems, they run their organizations. We also examined the problems unique to the ongoing demands of driving non-profit groups such as theirs and how they solve these problems.

******

Talking with Ray Esposito gives the impression that desire and energy can make a dream come true, no matter how difficult it initially may seem. He should know. Three years ago, he retired from the insurance business. "I burned out and on my way to D.C. to open an art gallery stopped off in Tulsa, Oklahoma to create **The Brass Ring Society.**" Since then Ray has fulfilled the dreams of over 50 dying children. Initially he envisioned the project as a small one: a small foundation which could be easily "created", then turned over to a local Tulsa resident, while he continued with his retirement plans. But that relatively modest plan got modified as soon as Ray chaperoned the first child the Society helped, a five year old terminally ill girl, and Ray, by his own account, "got hooked." Today the Society assists 5 to 10 children a month, and has plans to serve a minimum of 100 children per month in the future. Currently there are 13,000 terminally ill children in the United States, most of whom suffer from Cystic Fibrosis or Duchennes Syndrome, a fatal form of Muscular Dystrophy. Ray would like to help them all.

For an organization with a seemingly difficult and emotionally charged mission, the Society was founded on a simple premise: any child, 19 years old or under, living anywhere in the United States, Canada or Mexico who is certified as being terminally ill can have his or her dream fulfilled. How is this possible? Ray Esposito is quick to point out that most children, particularly those aged 12 and under, want to visit Disneyland or Disneyworld; and those arrangements are by now quite simple to make. For example, the Society is currently processing 19 applications, and fifteen children will visit Disneyland in July of this year. In order of preference, the next most desired dream category is to meet a favorite celebrity, "like Mr. T, Mr. Rogers, Michael J. Fox, Loretta Lynn." In the spring of 1984, the then-young Society fulfilled a young boy's dream of meeting Michael Jackson. Not only was Ray able to arrange this, he also saw that the child spent a day with the singer just visiting with him and making videos. After the child died Jackson dedicated his next album to him. Ray downplays the difficulty of these accomplishments. Such an arrangement, he says, "simply might take a lot of phone calls."

In fact, it is the initial telephone contact which is the hardest part of fulfilling a dream for a child. Each dream costs an average of $1,200, so if the money is there, the child and his or her family can be on their way. But in the case of the less typical request, such as the child who wanted to go trout fishing in Oregon, sometimes all the Society has to go on is the telephone. In this case, the Governor's Office of the state of Oregon was contacted, and they assisted with the dream. Even if the request is confounding -- what if a child wants to see a turtle race? Ray asks -- the Society has at this point amassed a large amount of literature on local and regional attractions and on names and dates of events occuring around the country to get them started. If it can't be found in the information they already have, which high school and college volunteers have keyed into their Fortune sytem (one CPU with and 9 workstations), the process of getting on the phone and looking for it begins. If it's out there, Ray is convinced that they will find it.

Using the Fortune has proved critical to the initial success of **The Brass Ring Society**. Ray seems convinced that without the use of **Fortune:Word** ("an exceptional word processor") and the Informix database that runs on the system, there would be little chance the Society could function. For instance, one database is entirely devoted to tracking financial donors, on which they are entirely dependent for support. (He has three other databases on Informix: one for personal information on each child; one devoted to the "One Hour Dream" -- explained below; and one for statistical studies). Survival without such a system seems unlikely; and the contributions to it that Fortune, software vendors and other organizations have generously made have helped keep them afloat. Fortune Systems donated the entire system they are now using. Various groups donated seven modems and a software company donated a package, worth $1,500, designed for generating maps (crucial to dividing and researching the country by region). A company in New Hampshire donated an accounting package called Real World. The Society is in the process of setting up the package and at some point Ray hopes to do all of the Society's accounting on the Fortune.

Continued on next page

Other plans for the future include implementing a program written by Relational Data Systems for use on Informix, which Ray calls "The One Hour Dream." Ideally, he says, some years down the road, they would receive a request about a particular child and, within an hour, would be able to generate an itinerary for that child and his or her family. This would be accomplished through an extensive database on Informix which would hold a vast amount of local and regional information around the country. So for instance a call would come in for a child who wants to see a rodeo. After checking with the child's doctor to see when he or she could travel, a volunteer could query the system for all rodeos occuring nationally during those dates. One would be picked and the system would access all other information specific to the city where it would occur. All available hotels, car rental companies and any other services the child and family might need would get accessed, checked for availability and for the possibility of reduced rates. Reservations would be made and the itinerary printed.

As it is now, if the request is not one that they have fulfilled before, they have to use the more time consuming research-it as-you-go method. Ray would also like to see more software packages available for use with UNIX and at affordable prices. The Society is just getting into desktop publishing and UNIX is not made for that yet. The software which is available for UNIX is expensive. Other organizations, he notes, might easily afford certain packages, but "if you're a poor charity...you can't spend $3 - $4,000 for a program." And given the operating priorities of the Society, if money does get raised and they have the choice of buying a program or fulfilling a child's dream, it's the child that comes first.

He adds that, though it is hard to believe, such an attitude is not necessarily prevalent in organizations similar to **The Brass Ring Society.** Some groups (of which there are about 26 in the U.S.) don't have qualms about calling in the media every time a child's dream is about to be fulfilled. **The Brass Ring Society** will never allow this, because he believes that "it's obscene to put these kids on television" and that "it violates the dignity of the child." But this means that every month or so he gets a call from a local network affiliate asking him if he has changed his mind. From the sound of it, this is a question that doesn't bear asking for Ray.

Ray Esposito is convinced that the dreams of terminally ill children are worth responding to. He says he's very comfortable with what he's doing and that "it's the best thing I've done with my life." If organizations as yet unfamiliar with the Society were just to hear about it, they would be convinced of the worthwhile goals he and the volunteers who assist him try to meet; and then "the problem of money will eventually solve itself."

For the time being, there are plenty of terminally ill kids who aren't being helped. Ray Esposito's dream is to be able to have a network that can reach them all and that can expand to help the chronically ill. Way down the road, he envisions "one of the largest charities in the world with a budget of $20 million." If this is a dream, then for Ray Esposito, it's as good as accomplished.

******

There is sadness in Dick Randall's voice as he describes the ways in which mentally and physically handicapped people historically have been treated. The founder and director of the **Remme Foundation** in Waltham, MA describes one case in which a disturbed boy was locked for years in an attic, only to be blinded by the sun when he finally escaped. When he first tried to arrange a bowling trip for a group of mentally handicapped people, he met tremendous resistance from the bowling alley's owner who erroneously feared the people would damange his property. "Only after pleading with him and talking with him" was Dick able to get the use of the facility; and now, he says, he and the owner are the best of friends. It's not that people are innately bad, Dick argues, just that "they don't understand, and what they don't understand they fear."

After thirty years of numerous fundraising efforts, assistance with the Massachusetts Special Olympics, a summer camp for kids, and an international exchange program, one gets the impression that it's people like Dick Randall who have helped people overcome their fear of others. He claims that someone might be physically disabled, or not as mentally able as a "normal" person, but that basically "we're all retarded, we're all handicapped...with some of us it's more obvious." Dick Randall has made it his job to insure that every one of us has access to basic human services, like transportation to facilities, events, and the recreational activities that many of us might just take for granted. For instance, he says, discussing a typical problem, there are 27 steps to the top of the Waltham Public Library; a wheelchair bound person "simply can't get to the wealth of knowledge in that library." And transportation to anywhere remains the number one problem for disabled people, not medical problems, as the able-bodied often assume.

Like Ray Esposito, Dick got involved with assistance to the handicapped thinking that he would stay on about two years and "put in his socially concerned time, so to speak." At that time, the **Remme Foundation** was a fledging organization, mainly involved with transporting the mentally handicapped. Soon, it expanded to assist all physically handicapped as well. It now assists any elderly person who needs help. Unlike other organizations, the **Remme Foundation** will assist with the transportation, but the recreation as well, and will provided staff who draw up itineraries, accompany people on trips and assist in the care of passengers. Or

See **User Spotlight**, Page 20

User Spotlight, Cont'd from page 19

it can just provide the transportation while another organization provides medical assistance and recreation. The Foundation recently assisted the Massachusetts Special Olympics Committee in just this way, providing three transportation vans for participants.

Ongoing programs also help insure that the disabled people who live in the area can meet for recreation, exercise and socializing. The initial bowling group has now expanded into a weekly league; a social is held once a month with a live band; Dick's own Bingo license supports an active fundraising Bingo group (which handles approximately $7 - $8,000 every weekend). In addition, in 1983 the Foundation purchased 52 acres in Wareham, MA and has built a summer camp, the **Remme Recreation Center**, for disabled youngsters. Each summer almost 1,500 kids attend the camp, with a ratio of 3 campers to one staff person (if the child is ambulatory) and 1 camper per staff person (if the child is wheelchair-bound). Because the Foundation has been able to expand its operation over the last few years, it recently purchased a $50,000 special wheelchair vehicle, so that now children with especially difficult disabilities (like soft or brittle spines) can participate in activities.

Part of the reason for the **Remme Foundation's** success, Dick says, is the ease with which he can administer particular programs and events with the Fortune computer (70 megabyte hard disk; 1 megabyte memory -- probably soon upgraded to 2 megabytes; 3 workstations; a NEC printer, a Centronics 351 and a Centronics Line Printer capable of printing 400 characters/second). At the **Remme Recreation Center**, for instance, a special medical program keeps track of all medications to be administered per day. Punching in the date will bring up for printing every child's medication and dosage and other information like the attending physician in case of emergency. Without such a program, attending to the needs of the 125 or so children at the camp at any one time would be a particularly difficult task.

For accounting purposes the Fortune is equally important. **Business Basic** makes handling the revenues from the 37 Bingo games played and from the concessions served easy to tabulate. State taxes are accounted for quickly and easily; Dick can run **Records Processing** through **Fortune:Word** to get weekly, monthly, quarterly and annual printouts of accounts. And, he says, he can "balance everything to the penny" with the program.

Soon he will also be able to do more intricate tasks on the system. One plan is to keep track of all the vehicles used for transportation to check efficiency. He will track

Continued on next page

idc™

MULTI-USER COMPUTER SYSTEMS

Specially Designed For Your Business

INTERDISCIPLINE CONSULTANTS, INC.

(201) 848-8500

235 EVERETT AVENUE
WYCKOFF, NJ 07481

FORTUNE END USERS
Do you have SUPPORT needs?

We offer services for those with:
- **frequent needs on a "monthly-retainer" basis**
- **occasional needs on a "pay-as-you-go" basis**

We're there when you need us to:
- **provide ongoing emergency programming & systems support**
- **customize programs for your present & future needs**
- **give on the job training**

mileage, cost per mile per vehicle, and each vehicle's maintenance record. He will also be able to keep track of passengers, see where each has been taken, etc. The system also currently tracks volunteers (380 strong) and maintains basic information like names, addresses, times worked, and skills.

The Fortune computer also assists him with the almost constant fundraising necessary for the ongoing maintenance of the organization. Funding drives at a local college and a dinner dance are two of the annual events; Dick also puts the transporting vehicles to good use when they are not being used by the handicapped. He charters them for trips to Atlantic City. The **Remme Foundation** also holds a monthly lottery in which it keeps 60% of the proceeds and 40% is returned to members in the form of cash prizes. Dick points out that the original purpose of this lottery was to attract 200 members into a club which would make small investments toward the support of the Foundation. This "200 Club" now supports 684 members.

Dick has also put his Fortune to business use outside the Foundation. His company, Precision Inventories, takes parts inventories for automobile dealers. In the past, (even the not so distant past), they would punch their physical inventory counts onto punch cards. These were processed at a remote location on an IBM computer and after a few trips back and forth with cards and printouts they would have a final report for the dealership.

Now they use their Fortune to enter all of the parts. Their inventories are either processed directly using a system developed with Progress, or in certain cases, sent to an IBM computer using the RJE 3780 (Remote Job Entry) communications program with a bisync modem running at 4800 baud. The mainframe compares the numbers with a General Motors master parts list which includes hundreds of thousands of parts. Instead of going back and forth with cards and printouts, they simply send some files back and forth, and are then ready to produce a printout on their high-speed Centronics Line Printer. This system has saved them a great deal of time and effort.

All in all, Dick Randall is a good example of someone who has been able to get his Fortune computer to perform a great variety of custom tasks. And he's used the greater efficiency that he's gained to help others and run a successful business. □

Karen Parrish

UNISOFT INC.

Experts in Computer Consulting - Sales - Service

Presents

FORTUNE SYSTEMS HELP

We Provide:

- Expert Trouble-shooting
- Years of programming and accounting experience
- *Free Initial Consultation*
- Training Available for:
  - * BASIC/IDOL/BAS and Multiplan
  - * UNIX and Ext. Fortune:Word
- Hardware Repairs & Upgrades
- FORTUNE trained experts at a cost that will surprise you!
- *Trade-Ins Welcome*
- Specializing in:
  - * Hard Disk Trade-In
  - * Hard Disk Upgrades

Authorized Fortune Master Dealer

Authorized Fortune Systems Repair Depot

CALL: (303) 671-8600 EMERGENCY HELP: (303) 691-HELP

UNISOFT INC. 14100 East Jewell Avenue Suite 27 Aurora, Colorado 80012

Shutdown, Cont'd from page 10

If the test on line 59 is true (if the file **/etc/.fsclean** exists) the variable "operation" is set to "up." Lines 63 and 64 are the other branch of the "if" test. If **/etc/.fsclean** does not exist, operation is set to "restart." Experienced For:Pro users will note that the "if" statement is incomplete; the balance has not been reproduced.

The modification is made by replacing lines 56 through 64 of **/etc/rc**. The new lines are reproduced below, and each line is labeled with the appropriate line number. Remember that these line numbers are for reference only and should **not** be typed in:

```
56  sync
57  if test -r /etc/.fsclean
58  then
59    /etc/setdt < /dev/console > /dev/console 2>&1 # e
60    rm /etc/.fsclean
61    operation=up
62  else
63    echo "
64 ^\G\"COMPUTER HAS BEEN DOWN. DON'T USE UNTIL TIME IS
    RESET BY SYSTEM MANAGER !^\G@
65 (System manager: run 'setdt' from root prompt, remove
    this msg from /etc/motd.)" >>/etc/motd
66      operation=restart
```

Lines 56 through 58 are the same as the old lines 58 through 60. They have simply been relocated slightly. Likewise, line 59 is the same as old line 56, lines 60 through 62 are the same as old lines 61 through 63 and lines 66 is the same as old line 64.

The change involves moving the call to **/etc/setdt** after the test for the existence of **/etc/.fsclean**. This means that there will be no request to set the date and time unless the file **/etc/.fsclean** exists. As indicated above, this will happen only if operation of the machine was last stopped through the use of the shutdown program. This is the only change that is actually necessary to accomplish the purpose. However, there will be no indication to subsequent users that the machine has been down, and that the date and time may be incorrect. So we have also added lines 63 through 65, which append a message to the file **/etc/motd**, the "message of the day" file. Lines 64 and 65 are more than 80 characters each, so they will be "wrapped" onto two lines on the screen with most editors. Type each with only one return, at the very end.

The file **/etc/motd**, if it exists, is sent to the terminal screen every time a user logs in. The message on lines 63 through 65 is formatted to appear partly in bold on the screen. The sequence ^\ is produced by holding down the gray "CTRL" key while typing the character \. If this particular message does not suit your purposes, any other message can be substituted between the quotation marks on line 63 and the last quotation marks on line 65. If your message is shorter, you may have fewer lines to your shell script, but this is perfectly okay.

After the modification, nothing is changed unless the operation of the computer is terminated without going through the shutdown routine. If the machine crashes or is reset, or if a "reboot" is intentionally requested, the machine will skip the request for date and time when it comes up. As in the standard configuration, the machine will automatically perform a file system check whenever it detects that operation has been terminated without a shutdown. As long as the file system check is successful, the machine will come back into full operation without any operator intervention. Each subsequent user will be informed that the date and time may be incorrect until this message is removed by the system manager. Whenever this message is detected, the system manager should reset the system date and time either through the For:Pro command "setdt" or through the global menu as indicated above. You must be logged in as "manager" or "root" in order to correct the system date and time. Then the message should be deleted from the file **/etc/motd**. If your machine is not using the "message of the day," this can be accomplished by issuing the For:Pro command **> /etc/motd** or by typing from the global menu **!> /etc/motd.** If you are using "message of the day," the file **/etc/motd** will have to be edited. You can use the same editor which you used to create your message of the day. □

DO YOU KNOW...

there is an organization that has a thorough knowledge of

FORTUNE BAS
MULTIPLAN IDOL

We have been providing widespread support to our Fortune clients for more than 2 years.

If you need:

- training
- support
- maximum computer performance
- peace of mind relating to processing
- customization
- application results
- minimum errors and confusion
- better ideas on how to increase the power of your Fortune

Call Steve Rosenfeld at (314) 837-7111

SUPERIOR COMPUTER SYSTEMS, INC.

917 N Hwy 67, Suite 105
P.O. Drawer T
Florissant, MO 63031

Making computers work for you

FORTUNE USERS
G. T. HAWES & CO., INC.
AN AUTHORIZED FORTUNE MASTER DEALER
ANOUNCES
"LET GEORGE DO IT"

Problem: Your company needs more computing power, an SX upgrade more RAM, terminals, etc. But you're on a strict budget and need "more for less."

Solution: Call 516-568-0044 and "Let George Do It." Compare our low prices and fast service with anyone!

Problem: Your dealer is dead and your stomach has that sinking feeling when you call his/her number and the phone company tells you "Too bad, so sad but computer dealer X went to bit heaven."

Solution: Call 516-568-0044 and "Let George Do It." We have extensive financial resources, watch our expenses like a hawk and will be around as long as you want us!!!

Problem: You have money to spend but "I don't get any respect." My computer dealer never calls me back on time; and when he does he suffers from terminal boredom.

Solution: Call 516-568-0044 and "Let George Do It." When you wave the green, we know how to say "Thank You!"

Problem: With so many options available, what is really the best for me?

Solution: Call 516-568-0044 and "Let George Do It." We will consult with you on the phone and give you a free second opinion.

When all you problems are solved and your friends ask how --
Smile and say "I Let George Do It."

Equipment Schedule available for the asking.

"If you don't see my name on your computer shipment your probably paid more than you should have."

G. T. HAWES & Co., Inc.
475 W. Merrick Road
Valley Stream, NY 11580
516-568-0044

Expand your Fortune® without spending a fortune.

New equipment costs big bucks. The Good Fortune Trader sells fully tested burned-in equipment, refurbished to Fortune® standards at prices that won't make waves with your cash flow. Call or write today for a listing of our specials.

THE GOOD FORTUNE TRADER

60 DAY WARRANTY

a division of Computer Services of Atlanta

(404) 392-9938

1534 Dunwoody Village Parkway/Atlanta, Georgia 30338

PROGRAMMERS

The leading manufacturer of status display panels for the fire alarm and security industry is seeking programmers and junior programmers to develop new applications and refine existing programs.

Successful candidates will be experienced in IDOL/BAS and SMC Business Basic for Fortune and Basic 4 systems. Familiarity with UNIX and system set-up and maintenance is desirable.

Candidates interested in joining a growing team-oriented organization located in Falls Church, Virginia should write or call with resume and salary history, to:

WSA

308 Hillwood Avenue
Falls Church, Virginia 22046
703 241-0687

Fortune Systems, Inc. is going on the road with over 25 new products, including a major new generation piece of hardware. See page 17 to see the schedule for the tour.

/u/fortune *news*
73 Jason Street
Arlington, MA 02174

/u/fortune* *news*. . .

The Newsletter for Users of the Fortune 32:16 Computer October1986/Volume 3 Number 10

The Basic Advisor

Ray Wannall is President of BaSiC Software Corporation in Baltimore and is contributing to this publication independently.

Question: *Regarding expanding the number of terminals available for the* ***Business Applications,*** *doesn't the* ***Fortune Software License Agreement*** *specifically restrict the number of terminals to eight?*

Answer: Thanks a lot! Because of your question I had to read the entire four pages of small-print legalese called the **Fortune Systems Corporation Customer Software License Agreement**. I hope I will be able to understand common English after this.

I, hereinafter identified as the party of the third part, learned, to wit, that the aforementioned agreement forbade my transmitting, directly or indirectly, my Fortune Software to places such as Afghanistan and Namibia, not to mention any Group P, Q, W, Y, or Z country (thank goodness X is still open).

As far as number of users is concerned, the only references I found were contained in part 16, OPERATING SYSTEMS, and part 17, IDENTIFICATION. Part 16 says, in effect, that **Fortune Software Products** are licensed for Single-User or Multi-User Fortune Operating Systems, and that any upgrade from Single-User to Multi-User requires that a new agreement be signed. Part 17 asks the customer to *check one* for maximum number of users: Single-User or Multi-User. There was nothing in the document that specifically limited the number of terminals on a Multi-User system.

Now I will concede that there are probably several versions of the software agreement in circulation. I have referred to the one I believe is the most recent. If you have signed an agreement which specifically limits the number of users, you should certainly abide by that agreement.

It is time, once again, to put aside questions. Back in July (**/u/fortune** *news*, Volume 3, Number 7) I mentioned in this column that I would address the question of the relationship between **Concept Omega** and Fortune users after the Dealer's Conference in San Diego. Now that the conference is history, I am loaded up with with things to report, much of it good news.

Before I begin, I want to thank **Mr. Dave Kloes** and his wife **Sharon** of **Uni-Komp**, **Mr. Dave Moskoff** and **Ms. Daphne Fotiades** of **Maritech**, and **Mr. Mike Eisen** of **Beacon Systems**, all subscribers to **/u/fortune** *news*

See **The Basic Advisor,** Page 22

Featured in this Issue. . .

News from Fortune -- Fortune's unveils the powerful new **Formula**, **Fortune:Word 3.0**, **For:Pro 2.0** and more. . .**Page 2**

The UNIX Directory -- In a continuing series on the fundamentals of shell programming, Dave Kloes discusses **test** and **if** statements, or **else**. . .**Page 6**

Tech Tips for BAS Users -- Recent tech tips from Choice on **BAS 2.0** programs. . .**Page 14**

Bugs, Bulletins, and General Help -- We may have the solution to your problem, or at least an answer to a question. . .**Page 15**

Getting started with Multiplan -- The second article in a series presents some real examples, including a loan amortization. . .**Page 18**

Text editors -- Some alternatives to **Fortune:Word** for die-hards who started using computers before they were friendly. . .**Page 16**

Multiplan

Last month we covered some of the basics of starting and using Multiplan. In addition we outlined a suggested sequence of steps for defining the problem and implementing a solution in Multiplan. This month we will take you through a Multiplan worksheet step by step so that you will have some real hands-on experience with developing a solution to a problem. The particular problem we have chosen to solve is **Loan Amortization** calculation. As Ruth Witkin explains in her book *Managing Your Business with Multiplan* (from which we have borrowed this solution of the Loan Amortization problem): "Before you commit yourself or your company to a long-term debt, whether for day-to-day operations, business expansion, or investment opportunities such as real estate, here are the questions that need to be answered: How much money do you need to borrow? How long will you need it? How can you repay it?

Loan Amortization Analysis

This particular worksheet can be executed nearly automatically. You will enter three numbers: the amount you plan to borrow, the rate of interest you expect to pay

See **Multiplan**, Page 18

*/u/fortune *news* is published by The Cambridge Consortium, Inc. It is not affiliated with Fortune Systems Corporation.

Accutrend System Services

Serving the Great Northwest
Washington, Oregon, Idaho, Alaska

Sales, Service, Support,
Software and Maintenance

- **Office Automation**
- **Accounting Systems**
- **Communications**
- **Database**

Free monthly newsletter

Call or Write:
Accutrend System Services
P.O. Box 38
Puyallup, WA 98371
(206) 841-1992

Authorized Fortune Dealer

/u/fortune *news*. . .

Editors: Josh Lobel, Mark Palmerino
Associate Editor: Benzion Chanowitz
Technical Editor: Scott Bradner
Advertising Manager: Mark Palmerino

/u/fortune *news*. . .is published 12 times per year by The Cambridge Consortium, Inc. Neither /u/fortune *news* nor The Cambridge Consortium are affiliated with Fortune Systems Corp. Contact us regarding subscription rates. Quantity discounts are also available.

Back issues are offered at the following prices:

Vol 1, Numbers 1-3	$1/each	Vol 2, Number 1-7	$3/each
Vol 1, Number 4	$.50/each	Vol 3, Number 1-	$4/each
Vol 1, Number 5-8	$3/each		

Although every attempt is made to insure that our contents are error-free and without risk, The Cambridge Consortium is not responsible for losses that occur due to correct or incorrect interpretation of /u/fortune *news*.

For more information about us, please call or write:

/u/fortune *news*, 73 Jason Street, Arlington, MA 02174. Telephone (617)646-5603.

The text for this newsletter was prepared using a Fortune 32:16 for text input. The information was then transferred to an Apple Macintosh Plus with kermit and formatted for typesetting using the Aldus Pagemaker program.

UNIX™ is a trademark of Bell Laboratories,Inc. Fortune 32:16™, Fortune:Word™ and FOR:PRO™ are trademarks of Fortune Systems Corporation. Multiplan™ is a trademark of Microsoft. IBM-PC™ is a trademark of IBM. MS-DOS™ is a trademark of Microsoft. Macintosh Plus™ is a trademark of Apple Computer. Pagemaker™ is a trademark of Aldus Corporation.

©1986 by The Cambridge Consortium, Inc. 10/86

News From Fortune

For:Pro 2.0
New Fortune:Word 3.0
and. . .The Formula

This month we received several product announcements from Fortune Systems, including a press release on their newest addition to their hardware line -- the **Fortune Formula.** We are reproducing these announcements for your information with only very minor editing. In the near future we will give you more information about these new products and our impressions of them.

Fortune Systems Announces Formula Line

New 68020 Hardware Unveiled

*(Ed. Note: The following is the text of a press release announcing the new **Fortune Formula.** The **Formula** uses a **Motorola 68020** microprocessor to achieve expanded performance. This machine is faster than the **32:16** series, and can accomodate a greater number of users. Its memory currently ranges from 1 to 16 megabytes, and will be expandable to 64 megabytes in 1987. By using the new **SCSI** controller, it can currently access disks up to 140 megabytes. A built-in 60 megabyte tape streamer is included with the **Formula.** As more information becomes available, we will keep you informed.)*

(Belmont, CA, Oct. 20) **Fortune Systems Corporation** will unveil its new **Fortune Formula** line of multiuser, multi-tasking supermicrocomputers, specifically designed to accommodate the Company's recent commitment to dealers and value-added-resellers (VARs), at noon today in New York City's Marriott Marquis hotel. Fortune Systems will announce the first ten dealers in the Company's new reseller network, and commitments from those ten dealers to purchase up to $40 million of **Fortune's Formula** and **32:16** systems over the course of the next three years.

According to **Fortune Systems'** executive vice president of marketing, **Robert A. Davis**, "The new **Fortune Formula** offers both an upgrade path for **Fortune Systems'** current installed base of 55,000 users, and an opportunity for dealers and VARs in the minicomputer and 32-bit microcomputer marketplaces to enter the UNIX world. Not only do the open architecture and modular design of the **Fortune Formula** offer the price/performance and expandability required in this marketplace, they also facilitate the conversion of proprietary minicomputer software developed by these dealers and VARs."

Continued on next page

The **Fortune Formula** is based on the **Motorola MC68020** microprocessor, operates at 16.5 MHz, and provides RAM capacity of 64 megabytes and disk storage of more than one gigabtye. The entry-level configuration, includeing 60 megabyte streamer tape, 70 megabyte hard disk and one megabyte RAM, will retail for $21,900.

The **Fortune Formula** is initially available with the Company's **For:Pro** operating system, which is based on Bell Laboratories' Version 7 with popular BSD enhancements. The Company's recent implementation of **QIC Basic, Business Basic, UX-Basic** and **Micro Focus Level II COBOL ET** have mad the porting of numerous vertical market applications transparent to Fortune hardware. According to **Paul C. Olin**, vice president, sales, "**Qantel** and **Basic 4** vertical applications with **Fortune's** own office automation software will provide a complete solution to end-users not available from any other vendor in this marketplace today."

The **Fortune Formula** can provide up to 80 RS-232 serial ports for interfacing workstations, communication lines, printers and other devices. **SNA, X.25** and asynchronous communications protocols are supported, and an optional **Fortune:Link** controller can link the **Fortune Formula** to a network maximum of 255 addressable systems. These systems may be the current **Fortune 32:16** systems, other **Fortune Formulas** and/or IBM PCs or PC-compatible computers.

Fortune is a pioneer in the UNIX multi-user marketplace, and was the first company to introduce the systems into the business environment.

Fortune:Word Release 3.0

Product Description

Fortune Systems announces the availability of **Fortune:Word 3.0**, an enhanced version of Fortune's powerful and versatile word processing software. **Fortune:Word** increases productivity and continues to meet the needs of a broad range of users -- novices to experts. Dedicated word processors, administrative personnel, technical professionals, managers, and executives all benefit from using **Fortune:Word.**

The new version replaces all prior releases of **Fortune:Word** and **Extended Fortune:Word**. All previous basic features and advanced modules are contained within **Fortune:Word Release 3.0**. New features have been added to meet the needs of the marketplace. Some exciting additions to the product are:

Ability to Access Help in a Fortune:Word Window
Autosave
Block Copy/Move/Delete
Case Conversion
Case Sensitive or Insensitive Search

See **Fortune News,** page 4

Our Memory Doesn't Cost You A Fortune

Would you like to:

Speed up your computer?
Add more users?
Use the latest software, like Fortune:Windows™?

Compatibility:	The F1MB is absolutely 100% compatible with Fortune's own 1 Meg memory board.
Affordability:	USA is proud to have equaled and exceeded the quality and performance of our competition, while offering substantial savings for both the dealer and end user.
Dependability:	All boards are burned in for 72 hours and tested for 48 prior to shipment. *Our One Year Repari or Replace Warranty underlines our confidency in the product.*
Repairability:	The F1MB facilitates repair in the field as it is 100% RAM socketed. Now you can repair a board by simply replacing a defective chip rather than face an expensive baord exchange -- reducing maintenance costs and system down time.

If you have a Fortune SX, XP45 or recent model XP30, you can use our board by simply plugging it in. Older systems can be upgraded to use our board with a simple and economical modification.

For Details Call: *Upgrade Service Associates*
3 Marion Road, Burlington, MA 0180 Phone: (617) 239-8147

Our Guarantee: If you're not completely satisfied within 30 days, USA will gladly refund your money.

Fortune News, continued from page 3

Cursor Movement (GO TO PAGE key) to next Tab, Indent, Decimal Tab, Center, Right-flush Tab, Required Space or Discretionary Hyphen
Document Assembly
Document Password Protection
Expanded On-line Help, including new Help Screens, Help in a Window, Function Key Help, and Help for Commands
Floating Footnotes
Forms Processing
Improved Spelling Correction
Insert/Delete within Insert
Interface Enhancements
Mathematical Calculations
Multiple Filing from Index
New Manuals
Optional Foreign Language Modules
Page Break without Insert
Performance Improvements
Print from End of Edit Options
Print Documents from an Index
Print Queue Access while Editing
Productivity Glossaries
Right Flush Tab
Save Changes While Editing
Search with Attributes
Shell Escapes from Menus
Sort Within a Document
Suppression of Status Line Display for Faster System Throughput
View Subsidiary Index
Wildcard Searches

As noted above, a new set of comprehensive user guides has been designed to make learning easier. Examples and hints are included to encourage use of the many capabilities of this product. As a special bonus for all **Fortune:Word** users, a diskette containing helpful Glossaries is included in the package. The Glossaries contain stored text and keystrokes that can be easily recalled to perform repetitive functions.

Several time-saving features have been included to save editing changes efficiently, allow faster submission of print jobs, and optimize filing operations.

The Autosave feature allows users to automatically write editing changes to the system disk from within a document each time a preset number of keystrokes is reached.

Users can also save changes at any time while creating or editing a document by using a command keystroke sequence.

A print selection from the End of Edit Options menu allows users to print immediately after saving a document.

Multiple print operations can be performed directly from an index.

Multiple document filing operations can be performed directly from an index.

A detailed description of all of these features will be included in a future issue.

As noted above, a new set of comprehensive user guides has been designed to make learning easier. Examples and hints are included to encourage use of the many capabilities of this product.

Fortune:Word 3.0 is available as an update to **Fortune:Word 2.0**, or as a complete product. **For:Pro 1.8** or later is necessary in order to use **Fortune:Word 3.0**.

Enhanced For:Pro 2.0 Announced

Product Description

Fortune Systems Corporation announces the release of **For:Pro 2.0** for the **Fortune 32:16** series of multiuser systems.

For:Pro 2.0 is the latest version of Fortune's enhanced UNIX operating system for the **Fortune 32:16** series. Featured in **For:Pro 2.0** are: multiuser support for 1 to 16 users; numerous performance and functional features; and an enhanced version of **For:Frog** hard disk optimizer program. An updated release of **For:Pro 2.0** is available to users with **For:Pro 1.8** or later version installed.

For:Pro 2.0 increases a system's expansion paths with support for 3.5 Megabytes of memory, four Intelligent Asynchronous Controllers (CommA-6), and full Fortune: Link networking support.

For:Pro 2.0 consists of a three-diskette cold boot set, a multiuser diskette, and a **For:Frog** hard-disk optimizer program diskette. In addition, multiuser support for 17 to 32 users is also available as a single diskette upgrade product.

For:Pro 2.0 increases a system's expansion paths with support for 3.5 Megabytes of memory, four Intelligent Asynchronous Controllers (CommA-6), and full Fortune:Link networking support. In addition to providing greater expandability, **For:Pro 2.0** offers users performance advantages in areas such as the disk input/output, disk optimization and kernel enhancements. Specific features supported in **For:Pro 2.0** are outlined below.

Continued on next page

For:Pro 2.0 Features

- Incorporating Fortune:Link 1.1
- Incorporating Distributed File System
- Support 3.5MB Addressable Memory
- Support four CommA-6 Controllers
- SCSI Command Set Support for QIC-36 (Tape), ST-506 (Disk) and Embedded SCSI Disks
- **For:Frog** - Optimized for Disk Performance
- Expanded Fixed Disk Bad Block Sparing
- Enhanced Device Connection Menu
- Enhanced Line-Printer Spooler

With this release of **For:Pro 2.0** Fortune has enhanced **For:Frog** for performance and added user functionality. **For:Frog** is a hard-disk optimizer program which reorganizes the file system structure on the hard disk to provide maximum disk performance with maximum data protection. A significant performance improvement on hard-disk I/O operations can be noticed after running **For:Frog** on the **Fortune 32:16** system. It is the only hard-disk optimizer program available in the UNIX community.

Performance and Limitations

Fortune Systems continues to provide greater functionality, improved performance and added expansion with the release of **For:Pro 2.0. For:Pro 2.0** improves performance for random disk reads by approximately 10% over **For:Pro 1.8** and delivers additional performance improvements with increased memory and **SCSI** disk and tape support.

The use of the **For:Frog** utility included with **For:Pro 2.0** also provides a significant performance improvement on disk I/O operations. This release of **For:Frog** reduces the actual run time of the optimizer with improvements in the resulting file system organization. It also provides the capability to optimize file systems up to 350 Megabytes. A resident utility called "**frag**" provides the user with a measure of the level of file system fragmentation.

All software applications, domestic and international, that operate on **For:Pro 1.8** will function in the same manner on **For:Pro 2.0**, with the exception of some 1.7 applications that are incompatible with 1.8 and 1.8.1 because of improper signal handling.

Competitive Analysis

For:Pro offers users an "user friendly" interface that is recognized as an excellent front-end to the UNIX operating system and available with no other multiuser system. This release includes an enhanced auto-configuration menu interface which dynamically recognizes option cards and provides the system administrator a simple means for system configuration.

See **Fortune News**, page 12

Looking for Business Accounting Software that will conform to your Business?

Look to DSM

Applications
- Can be tailored to your organization's specific needs
- Economy of Standardization
- Follows Generally Accepted Accounting Principles (GAAP)
- Luxury of Adaptibility
- Data Base Incorporation

Software for:
- Not-for-Profits
- Construction
- General Business
- Property Management
- Steel Service Center
- Law Office Management

Hardware
Fortune Systems, NCR, AT & T, IBM

Call or Write
Data Systems Management, Inc.
277 Coon Rapids Blvd.
Minneapolis, MN 55433
(612) 780-5200

All software was developed using PROGRESS, a powerfull relational data base which allows you to tailor the software to your specific needs.

DSM PROGRESS

Dealer Inquiries Invited
PROGRESS is a trade mark of Data Language Corporation.

BACKUP AND RESTORE UTILITY

PROTECTION MADE EASY!

- o Back up all or selected user directories
- o Restore all or selected user directories or files
- o Format diskettes
- o Review backup volumes
- o Menu-driven, easily installed
- o Complete documentation

Also copies the files necessary to preserve your system configuration--a tremendous help if your system ever needs to be cold booted!

$150.00 Backup/Restore Utility
$200.00 Utility + 2 boxes 5 1/4" 96 TPI high-quality diskettes

For more information, call CTD at (313) 540-1282.

To order, make check payable to *Computer Trade Development Corporation.* Send to:

BACKUP/RESTORE UTILITY
Computer Trade Development Corporation
Suite 1386
Birmingham, MI 48010

CTD

Company:_____
Contact:_____
Address:_____
City:_____
State:_____ Zip:_____

The UNIX Directory

INTRODUCTION TO SHELL SCRIPTS

David E. Kloes is President of UNI-KOMP which is located in Houston, Texas. He provides UNIX seminars, writes software for the Fortune Computer and is vice-president of the Houston UNIX User's Group. He is contributing independently to ***/u/fortune*** *news.*

For those of you that missed the first two parts, let us digress (I love that word) and briefly summarize what we have covered so far. In Part 1, we defined the **shell** and introduced you to some of the basic concepts of shell scripts. In Part 2, we discussed some new concepts and introduced you to the **for do done** loop and the **case esac** command sequence. In Part 3 we will discuss the **if then elif else fi** command sequence and the **test** command. As usual, we will also divert your attention occasionally to discuss new concepts.

In case we forgot to mention it (and we did), much of the material we are covering can be found in the **Introduction to FOR:PRO Manual** or the **FOR:PRO Programmers Manual** under the **sh** command.

THE if then elif else fi COMMAND

The **if** statement is one of the conditional statements the shell provides for accomplishing a task when a certain condition does/does not exist. It works like the **if** command found in almost any programming language. The syntax of the command is:

```
if <condition1>
      then <list1>
elif <condition2>
      then <list2>
else <list3>
fi
```

This basically says that if condition1 exists, then do the things in list1, otherwise if condition2 exists, then do the things in list2, otherwise for all other conditions do the things in list3. The most difficult part of this command is learning and understanding the **condition** syntax. It is well worth the effort, however, because the **if** command is also one of the most powerful and most used commands in shell scripts. In addition, the same **condition** rules will also apply to other shell commands we will be discussing.

(1) Testing for a Successful or Unsuccessful Result

One way to use the **if** statement is to simply test whether an operation is successful or not. For example:

```
if cp -routB /dev/fd02 770 /b
       then echo "^GBackup was successful!"
         exit
fi
```

Notice first of all that whenever an **if** command is used, it must end with the "fi" (if spelled backwards). If you forget it, you will get a **syntax error** (we will discuss syntax errors in more detail later). As in this example, the **if** statement does not have to have the **else** or **elif**. This script will backup the **/b** directory; beep the terminal; and display the successful message if the backup completes successfully. Let's quickly discuss a new concept we have introduced in this example.

The *if* statement is the shell's way of testing things.

Notice we have imbedded the ^G control character in our **echo** line. The ^G beeps the keyboard whenever it is used in a shell script. For this particular control character, we simply depressed the "CTRL" and "G" keys to produce the character in the echo line. We will discuss control characters in more detail later on.

What will happen if the backup is not successful? The answer is nothing since we do not have an **else** statement. If the backup is not successful in this case, the only error messages we would see would be any error messages produced by the **cp** command. For the sake of "user friendliness", it is good programming technique to tell users when something is successful and when it is not. Let's modify our script for this:

```
if cp -routB /dev/fd02 770 /b
   then echo "Backup was successful!"
else echo "^GBackup was not successful!"
     exit
fi
```

If the backup fails (is not successful) for any reason, the terminal will be beeped; the "not successful" message will be displayed; and the program will "exit". The **exit** command is used in a shell script to exit the program. It is not required at the end of the program but is required if we want to exit the program at some point before the end of the program. We must also point out that any number of commands can be included in the **if** statement - we may want a number of commands to be run successfully before we take another action or continue on with our shell script.

(2) Using the **test** Command

The **test** command (refer to the **test** command in your **FOR:PRO** or **UNIX** manual) may be used as our condition to test for any number of things. Let's list out

Continued on next page

some of the more common tests and give an example of each:

-f tests for a file
-d tests for a directory
-w tests for write permission
-r tests for read permission
s1 = s2 tests for string1 equal to string2
s1 != s2 tests for string1 not equal to string2

-o is the **or** operator
-a is the **and** operator

These are the more common test evaluations. We will save the discussion of numeric evaluation for next time and concern ourselves with only the string comparisons for now. Here are some examples and explanations:

```
echo -n "Please enter name of file: "
read filename
if test -f $filename
   then more $filename
else echo -n "
File '$filename' does not exist.
Depress <RETURN> to continue: "
     read bunk
     exit
fi
```

Let's explain how the **test** command works here and also introduce a new concept. First of all, we prompt the user for the name of a file. The user may enter a full pathname if the file is not in the current directory or he may enter a filename if the file is in the current directory. Part of what we have to be aware of when we write our shell scripts is how **UNIX (FOR:PRO)** will react when data is entered by the user. We could leave out the **test** altogether and let **UNIX** give its normal error message for any given condition. however, **UNIX** messages are not as "user friendly" as we would like them to be and we prefer to give a more meaningful message.

The **read** command waits for user input and will take whatever the user enters in response to the question and (in this case) assign it to the variable name *filename.* We could have used any name for this variable. Remember from our previous articles, that we can now access the value of this variable by putting a **$** in front of it. That is to say that if the user enters "/u/davek/letter", the value of the variable "filename" will be "/u/davek/letter" and may be accessed by using **$filename**.

The **test -f** tests to see if the name the user entered is a valid file. If it is, then the **more $filename** will display the file on the terminal a screen at a time. The **-f** test will fail if the name the user entered cannot be found. If would also fail if it were a directory name instead of a file name. If the test fails, the message is displayed and the program waits for user input again. In this case, whatever the user enters will be assigned to the variable *bunk* (which is sort of a garbage buffer) and the program will "exit" regardless of what he enters.

To explain the rest of the **test** arguments, we will show different ways that the **if test** condition can be entered. Consider the following example:

```
if test -d $filename
```

This test will be successful (true) if the name that is entered is a directory, otherwise it will not be successful (false).

```
if test -w $filename
```

This test would be successful if the file has **write** permission for the current user. Normally we would use this type of test if the shell script we were writing involved some UNIX operation where the user had to have write permission.

```
if test -r $filename
```

This test would be successful if the file had **read** permission and would normally be used for operations that required this permission. Again, our goal in using these tests is to keep control of our shell scripts and anticipate **UNIX** error situations where we can provide the ability to give the user a more verbose description of what the problem is. We may also decide that the normal **Permission denied UNIX** response is adequate and not do any specific testing. The nice thing is that the tools are provided, if you want to use them.

We may also do multiple tests using the **-a** (and) and **-o** (or) operators. For example:

```
if test -f $filename -a -r $filename
```

This would test if the name entered is a filename **AND** if it has read permission. **Both** conditions must be met in order for the test to be successful.

```
if test -d $filename -o -f $filename
```

This would test if the name entered is a file **OR** it is a directory. If **either** condition is true, then the test is successful.

One important sidenote here. Problems can occur when the "test" is based on input from an operator such as in the above examples. What would happen if the operator just depresses the **<RETURN>** key (does not enter anything)?. The answer is that we will get the syntax error **test: argument expected** and the user under some circumstances ***MAY DROP INTO THE SHELL***. One of the reasons we write shell scripts is so that the user is **not** in the shell. A shell script can abort if a shell syntax error is received. For this reason, you should test your shell scripts very thoroughly and try to anticipate things a

Continued on next page

UNIX Directory, continued from page 7

normal end user would do. Hitting the **<RETURN>** key is a very probable response, therefore we should program for this happening (see below for one possible way of programming for this). Let's finish up Part 3 by looking at another example that will incorporate some new concepts:

```
filename=""
echo -n "^LPlease enter name of file or
'exit': "
read filename
if test "$filename" = "exit"
   then exit
fi
if test "$filename" != ""
   then if test -f $filename
           then more $filename
        else echo -n "
File '$filename' does not exist.
Depress <RETURN> to continue: "
             read bunk
             exit
        fi
else echo -n "
You must enter a filename or 'exit'.
Depress <RETURN to continue: "
read bunk
fi
```

This looks like a busy little shell script but when you break it down by command sequence, it becomes more meaningful. Let's take it line by line to reinforce some concepts and commands we have talked about in the series of articles so far:

```
filename=""
```

This line simply initializes the variable *filename*.

```
echo -n "^LPlease enter name of file or
'exit': "
read filename
```

The first line clears the screen and displays what is between the quotes on the terminal. The ^L (CTRL-L) is the control character on the Fortune terminal for clearing the screen. Remember the **-n** option means "no newline" and in effect leaves the cursor at the end of the line so the user can respond. Note the space between the : and the " - this separates the question from the user input. The **read** line takes whatever the user types in and puts it in the variable *filename*. We could have used any word instead of "filename".

```
if test "$filename" = "exit"
   then exit
fi
```

Continued on next page

We are the *largest* Authorized Fortune Dealer in the Pennsylvania, New Jersey and Delaware area. As part of the **P**rofessional **S**olutions **G**roup of companies with over 35 years in business automation, we have the *stability* that our customers demand

Our services include a factory trained field engineering team, a complete parts inventory including system upgrade capability and a software support department.

For all of your Fortune needs contact:

Mark Shields of PSG at
2550 Eisenhower Avenue
PO Box 762
Valley Forge, PA 19482

or you can call Mark at (215) 666-1400

UNIX Directory, continued from page 8

This tests to see if the user entered the word "exit" and will exit the shell script if the test is successful (true). We have introduced the concept of **string comparisons** here. Let us mention some rules about string comparisons. These are the rules that we have found to work over the various versions of UNIX that can be found (Xenix, UNIX System 3, Version 7, System V, etc.). If you follow these rules (even though other methods may work), you will normally stay out of trouble. Our goal is usually to try to develop rules that when followed, keep us from having to do an abnormal amount of "debugging".

First, there should be a space on either side of the "=" sign. Your shell script may not test correctly if either space is omitted.

Second, enclose the strings on both sides of the "=" sign in quotes.

Finally, some other common errors are to forget the word **test** and leaving one of an unmatched pair either of " or of '. Be sure to check for these.

```
if test "$filename" != ""
```

Notice that in the program above the **if**, **else**, and **fi** for this test are all aligned down the left column. This is to make the shell script readable and so that we associate which portions of the **if, then, elif, else, fi** belong to each other. This is especially important in cases such as this where we have multiple **if** statements. The extracted one line above is our check to give an error message if the user depresses the **<RETURN>** key all by itself without entering anything else. We set the variable *filename* equal to "" at the start of our shell script. The only way it could still be equal to "" is if the user depresses the **<RETURN>** key by itself. If someone does this, the "You must enter a filename or 'exit'." message is displayed and the program logically ends. We will show you later on how to display the error message and continue to ask the question until the user gives a valid response.

The **!=** stands for **not equal to**. If the user gives a valid response which means that **$filename** is not equal to "", then the following portion of the shell script is executed:

```
        if test -f $filename
           then more $filename
        else echo -n "
File '$filename' does not exist.
Depress <RETURN> to continue: "
             read bunk
             exit
        fi
```

If this test is successful (the name the user entered is a file) then the **more $filename** portion is executed, otherwise the error message is produced. Notice we have aligned the **if, else, fi** lines for this portion of the shell script also.

Next time we'll continue with the **if** and **test** statements and also learn about numeric comparisons. Be sure to try out these examples. As we continue with this series of articles, we will be building on concepts and commands from previous parts. If you missed any of the previous articles, it would be worth your while to contact **The Cambridge Consortium, Inc.** for back issues. □

An autologout shell script

The autologout script is designed to check for any login processes that are associated with a port to see how long they have been idle. If a process is found to be idle for longer than a preset time, then that process is killed. The bare bones of a shell script to accomplish this is presented below. *Please take note that this is a particularly unfriendly autologout procedure because it does not attempt to alert the idle user to the impending doom* - most autologout procedures will send messages to the idle user's screen informing them that their process is about to be terminated. This one does not. If you decide to use this script without modification, we suggest that you use a relatively high idle time, such as 30 or 60 minutes. Actually, we do not suggest that you use this script unless you modify it to be a bit more friendly. The main problem is that in many installations it is not unusual for someone to be in something like **FORTUNE:WORD** and walk away from their desk for lunch or a meeting. If our shell script were running, the **FORTUNE:WORD** session would be terminated and all changes the user had made to that particular file would be irrevocably lost. Sometimes, however, this is the price to pay for leaving a terminal unattended. We would be very interested in modifications to this shell script. If you work with it and make it more user friendly, then please send us a copy so that we can publish it and put it on a future disk. Also, if you have another version of an autologout script (or program) that we could have, we would be delighted to publish it as well.

The script uses the **finger** command to find out how long a login process has been idle. There are various sources for the **finger** command available in the public domain. You can obtain an executable version of the **finger** we use by downloading it from our bulletin board. We have also packaged this version of **finger** on our new Software Disk #7 called **UNIX TOOLS**.

How the script works

We first set the idle time to 60 minutes. You can obviously modify this variable. However, the algorithm

Continued on next page

Auto logout, continued from page 9

we use prevents you from making this larger than 1 hour. This is a limitation but we chose to incur it so that our script would be more compact.

The next command, **IFS** etc., sets the **internal field separator** to just a carriage return so that we can access each line of output from the **finger** command as one word. This is necessary so that the next **for** loop will consider a line (from the **finger** command) as a word.

Once inside the for loop we set the **IFS** back to a space and a carriage return so that we can now access each word in the current line that we are processing. The **set** command makes each word in the line accessible as an argument, as in $1, $2 and so on.

The following displays the output from the **finger -i** command:

```
Login       TTY            When              Idle
root        console   Tue Sep 16 08:31
root        tty02     Tue Sep 16 10:05    1 day  2 hours
mbp         tty03     Tue Sep 17 11:21    3 minutes 27 seconds
```

This display is shown to illustrate that it is the *eighth* word on the line that we are interested in. The first line "Login..." is the title line and only contains 4 words. Since there isn't an eighth word, we won't do anything. The second line also doesn't have an eighth word which indicates that **root** on the **console** has not been idle at all. The eighth word on the third line is **day**. This will key us off to kill the process. The eighth word on the fourth line is minutes and will key the script to check the seventh word to make sure it is larger than 60.

All of this is coded in the **case** statement. If the eighth word is any one of the words day, days, hour or hours, then the script calls another shellscript called **knockout** which will kill the process associated with the device from the finger command (in $2, which is the second word on the line from the finger output). You can see that if the eighth word is minutes, then we check the 7th word using the **if, test** and **expr** commands. Tune into future installments of the UNIX Directory for a more in-depth treatment of the **expr** command.

The first box below is the script for the **autologout** shell script. The second box is the script for the **knockout** shell script.

```
:
toozng=60;;
IFS='
'
for i in `finger -i`
do
  echo -n $i
  IFS=' '
  set `echo $i`
  case $8 in
     day|days|hour|hours) knockout $2;;
     minutes)
          if test `expr $7` -gt `expr $toolong`
          then
             knockout $2
          fi;;
  esac
  echo
  IFS='
'
done
```

```
:
: kill the processes associated with the device in
: the first argument
:
device=$1
case $device in
  console) device=con;;
esac
IFS='
'
for i in `ps`
do
  IFS=' '
  set `echo $i`
  case tty$2 in
     $device) echo -n " kill -9" $1;;
  esac
  case $2 in
     $device) echo -n " kill -9" $1;;
  esac
  IFS='
'
done
```


PROGRAMMERS

The leading manufacturer of status display panels for the fire alarm and security industry is seeking programmers and junior programmers to develop new applications and refine existing programs.

Successful candidates will be experienced in IDOL/BAS and SMC Business Basic for Fortune and Basic 4 systems. Familiarity with UNIX and system set-up and maintenance is desirable.

Candidates interested in joining a growing team-oriented organization located in Falls Church, Virginia should write or call with resume and salary history, to:

WSA

308 Hillwood Avenue
Falls Church, Virginia 22046
703 241-0687

MAKE YOUR FORTUNE GROW

One of the greatest assets of your Fortune Systems 32:16 is its built-in ability to expand to match your need.

Whether you want to increase your number of workstations, upgrade system capability, or tie-in your Fortune Systems cluster to a mainframe, Micro Resources can give you the prompt, complete service you need.

As *the* Fortune Systems Master Dealer in Ohio, we have the hardware and software you need available for immediate shipment. From Fortune:Link to uninterruptible power supplies, from storage expansion systems to Development Utilities, we have the exact blend of hardware, software and systems expertise needed to make your Fortune Systems 32:16 do exactly what you want it to in the most reliable and cost-efficient manner possible.

And our genuinely friendly staff programmers and systems analysts are available to provide you with whatever additional support you may need, from software consultation and customization to integrated systems design and specification.

So whether you need a quick question answered or a new service agreement, Business Applications Update or just some networking cable, call Micro Resources.

We'll help your Fortune Grow.

Micro Resources

535 Metro Place South Dublin, Ohio 43017 (614) 766-2335

Fortune News, continued from page 5

Advantages in addition to the user interface are found in the overall performance of the operating system in large (8-12 users) user configurations, kernel specific enhancements, BSD 4.2 functions and stability. It provides users and resellers an extensive set of installation and configuration features that no other UNIX operating system provides.

Mail included in For:Pro 2.0

With the release of **For:Pro 2.0**, the UNIX **mail** facility will be included with the standard operating system. In the past this program was a part of the **Development Utilities**. The **mail** program allows users to send and receive mail among users of their own system, and also from other systems with the use of the **Fortune-to-Fortune** copy program.

32:16 SX Product Line -- Memory Expansion 3.5 Megabytes

Product Description

Fortune Systems announces **3.5 megabytes memory** capability for the **SX** product line. This memory expansion allows users to run more memory resident programs without degradation in performance. Increased memory reduces program and data swapping, which reduce the load on the system and allows for faster response time. With this, more users can run larger programs without impacting one another.

Current **SX** customers who wish to take advantage of this upgrade can purchase a **Memory Upgrade Kit**. This kit includes a ribbon cable and a **1MB** memory card; the memory cards are attached to the ribbon cable. Additional memory cards can be purchased to fulfill end-users memory requirements.

Present customers, who have **XP** systems, and wish to upgrade to **3.5 MB** of memory can purchase an **SX Upgrade Kit**, Kit I through IV. These upgrade kits include the modified SX motherboard which accepts the upgrade to 3.5 megabytes of memory.

Performance and Limitations

Under most circumstances, end-users can expect improved performance from the SX due to reduced swapping.

In order to address 3.5 megabytes, it is necessary to use **For:Pro 2.0**, which has recently been announced. The hardware is available immediately.

C Programming Language for the Fortune 32:16 and Fortune Formula

Product Description

Fortune Systems Corporation announces the availability of the latest version of **Fortune C Programming Language** for the **For:Pro** operating system. The **Fortune C Language** includes the C compiler developed for the **Fortune 32:16** series and Fortune's newest system, the **Fortune Formula**.

The Fortune Systems enhanced version of the **C Programming Language** is a general purpose programming language based on the C language developed by Bell Laboratories. It is a fast and versatile language suitable for applications programming as well as for developing operating systems and compilers.

This release provides cross-compilation capabilities between the **Fortune 32:16** and support for the **Fortune Formula** which is based on the Motorola 68020 microprocessor and with an option of the Motorola 68881 floating point coprocessor. This release is compatible with previous releases, except that it requires **Language Development Tools 2.0.**

Fortune C Language offers **For:Pro** developers and users a significant performance increase and full Motorola 68020 support for the **Fortune Formula**. In addition to these features the latest release contains the following principal features which are specific to the **Fortune Formula**:

- Position Independent Code
- Full Motorola 68020 and 68851 Support
- IEEE 754 Floating Point Standard and the Motorola 68881
- Compiler Target Switch

Performance and Limitations

This release of **Fortune C Language** produces executable modules quickly and efficiently. Specific optimization features available are:

- Basic Optimization
- Stack Checking Code Option
- Motorola 68881 Floating Point Coprocessor Special Instructions
- Single Precision Floating Point Functions
- Structured Data References Alignment

This release of the compiler provides the capability to cross-compile from the **Fortune 32:16** series to the **Fortune Formula** and vice versa. In addition, the ability to produce position-independent code has been added. Other changes include enhanced code quality, especially relating to bit field, "short", and "char" data types.

Continued on next page

Significant performance advantages are a result of the compiler's position independent code and Motorola 68020 32-bit support and IEEE 754 floating point support.

Position independent code allows object code to be produced which is relocatable, potentially discardable, and able to be down-loaded to ROM's while ensuring that the object codes is position independent. Support for the Motorola 68881 Floating Point Coprocessor will dramatically improve the performance of arithmetic functions.

The **C Language Release 2.0** is compatible with the following products on the **Fortune 32:16** series: **Language Development Utilities 1.8.1, 1.8.2 and 2.0; C Compiler 1.7; Pascal Compiler 1.3; Fortran Compiler 1.3.**

There is no binary compatibility between **Fortune 32:16** series and **Fortune Formula**.

Prerequisites

- **Fortune 32:16** system with 512K bytes of memory or **Fortune Formula**
- **For:Pro 2.0** for the **Fortune 32:16** series or **For:Pro 3.0** for the **Fortune Formula**
- **Language Development Tools 2.0**
- If an earlier release of Language Development Tools is installed, you must install Release 2.0 before installing the C compiler. □

DO YOU KNOW...

there is an organization that has a thorough knowledge of

FORTUNE BAS
MULTIPLAN IDOL

We have been providing widespread support to our Fortune clients for more than 2 years.

If you need:

- training
- support
- maximum computer performance
- peace of mind relating to processing
- customization
- application results
- minimum errors and confusion
- better ideas on how to increase the power of your Fortune

Call Steve Rosenfeld at (314) 837-7111

SUPERIOR COMPUTER SYSTEMS, INC.
917 N Hwy 67, Suite 105
P.O. Drawer T
Florissant, MO 63031

Making computers work for you

Unix Tools

We've gathered up some useful UNIX programs and are now making them available on DISK #7

Rolodex: This is a computerized rolodex file that no Fortune computer should be without. We've found that it completely takes the place of our traditional rolodex file.

Finger: This program is used to display various statistics about user accounts. We use it in the shell script that automatically logs a user off of the computer if they have been idle. It is extremely useful for other things as well.

Monitor: This program will keep running a specified command every *n* seconds and will display the output on your monitor. It is a great way to monitor the progress of some process in your computer.

Less: This program is just like the UNIX command **more** but it does more than **more** (why they called it less is beyond me). For example, you can back up one page at time - which is something you can't really do with **more**. It also has many more features.

Last: This program accesses the /usr/adm/wtmp file and lists everyone who has signed in and how long they have been on the computer for that session. If you give it an argument of the account name, then it just lists the login times for that person.

Bm and Para: These are **grep** like programs which do string searches in files. **Bm** - which stands for Boyer-Moore - is just like **grep** except it is much faster. **Para** is like grep except it prints out the *para*graph that a string is found in rather than the line.

Cut and Paste: Two **UNIX** commands that allow you to cut and paste from one file to another. **Cut** allows you to extract any series of columns from one file and write it to another file. **Paste** takes a file and inserts it between any column you specify. These can be very handy tools!

And Games too...

You've got to try **eliza** - this program is artificially intelligent and takes the part of a "non-directive" therapist. Do you have any problems? **Eliza** will cleverly get to the bottom of them! We also include an updated verstion of **hack** which is sure to provide hours of excitement for those of you who like to crawl around in dangerous caves. Finally, we have **puzzle** - you'll just have to get the disk to find out about this one.

This disk, like the others, costs **$10**. As introduced last month, if you buy four or more disks, then you will receive some breaks on the price. Six disks cost $50, five disks cost for $42 and four disks for $33. If interested, send you requests to us at 73 Jason Street, Arlington, MA 02174.

Tech Tips

This month's tech tips have come from Fortune Systems. We will try to keep our readers abreast of problems that are known to Fortune.

Elizabeth M. Muth, the **Manager of Software Services** at Fortune Systems recently forwarded us the following tech tips for BAS applications revision 2.0. They were sent to her from **Choice** in Marietta, Georgia.

This tech tip will correct the aging of current invoices into the future category on the **A/R Trial Balance** and **A/R Customer Inquiry.**

```
LOAD "CARICO"
7340
EDIT 7342 C[ ] D[REM]
SAVE

LOAD "CARRE2"
EDIT 7184 C[71] R[9]
7184
SAVE
```

This tech tip is to correct the special logic used in the entry of the **G/L Income Statement Account Record.**

```
LOAD "CGLSEO"
EDIT 103 C[ELSE GOTO 00] R[600]
600 REM CHECK SUB ACCT
602 IF POS(STR(J9:"00")="020304050608010")=0 THEN GOTO 110
605 LET X7$=Z8$(1,6-LEN(X7$))+X7$
610 CLOSE (6) ; OPEN (6) "CGLIS"
620 READ (7,KEY="C",DOM=621)
630 LET M9$=KEY(7,END=699); READ (7)
640 IF M9$(1,1)<>"C" OR M9$(2,2)="~~" THEN GOTO 699
650 READ (6,KEY=M9$(2,2)+X7$,DOM=670)
660 GOTO 630
670 LET B9$="SUB ACCOUNT NUMBER "+M9$(2,2)+X7$+" NOT FOUND
IN  FORMAT FILE",C9=1; GOTO 9000
699 GOTO 9000
SAVE
```

Elizabeth M. Muth, the **Manager of Software Services** at Fortune Systems recently forwarded us the following tech tips for BAS applications rev. 2.0.

This tech tip is to correct the problem of the payroll journal update clearing the accounting period record.

```
LOAD "CPRRJ5"
EDIT 200 C[E7,] R[T]
EDIT 8180 C[<] R[T]
SAVE
```

This tech tip will correct the quantity in the inventory master and inventory locations (3-5) not being correctly updated when invoicing for warehouses 3-5.

```
LOAD "CBIDA5"
EDIT 5020 C[I1$(] R[1] [5]
EDIT 5305 C[A1$+] R[2] D[)]
EDIT 5650 C[A1$+] R[2] D[)]
SAVE
```

□

FORTUNE TRAINING AT ITS BEST!

COMPUTER TRADE DEVELOPMENT CORPORATION (CTD), Fortune System's largest Master VAR, offers the most comprehensive repertoire of Fortune applications training available.

23 full-time certified instructors and an impressive facility, featuring eight classrooms fully-equipped with one terminal per student and Hewlett-Packard LaserJet printers ensure quality education. And because class size is limited to ten, you are assured of maximum instructor-to-student attention. Courses are targeted to all experience levels:

Meet Your Fortune System	*Fortune:Windows*
System Administration	*Using Your HP LaserJet*
Fortune:Word, 1-Day Overview	*Introduction to Multiplan*
Fortune:Word, 4-Day Intensive	*Multiplan: Spreadsheet Design*
Advanced Fortune:Word	*Advanced Multiplan*
Glossary Programming	*FOR:PRO for End Users, Intensive*
Fortune:Word Mini Database	Practical FOR:PRO, 1-Day Overview...

...and many more, including C, the vi screen editor, and FOR:PRO shell programming. The company that customized Fortune training for Ford Motor Company now makes the best training available to you! (Onsite classes can be arranged, if you prefer.)

RESERVE YOUR SEAT TODAY!

1-313-271-3020

COMPUTER TRADE DEVELOPMENT CORPORATION
One Parklane Boulevard, Suite 1404 East
Dearborn, MI 48126

Monthly Bug Status

9/23/86

(Ed. Note: Each month Fortune Systems distributes a report on know bugs. What follows is a recent list)

Problem: Multi-volume back-up failure (cp with -B option). **Date Problem Reported**: 3/18 **O/S Level** 1.8.1 and below. **Work around available (Y/N)**:Y Software Bulletin #2. **Status**: Problem should be resolved with level 2.0 of O/S; still needs to be tested.

Problem: Canceling out of multi-volume tape back-up results in "preamble incorrect" message when attempting to restore or display tape. **Date Problem Reported**: 8/26/86 **O/S Level**: 1.8/1.2.1 Streamer Tape software **Work around available (Y/N)**:N Software Bulletin: #16. **Target Fix Date**: ASAP. **Status**: Problem has been escalated to engineering.

Problem: In Basic, if one escapes to the shell, re-enters Basic and then uses an input statement - Basic will time out if the input statement is left indefinitely on the screen. **Date Problem Reported**: 9/3/86 **O/S Level**: 1.8.1.1. **Status**: Problem is in development -- trying to reproduce.

Software Bulletin:

cp command with -B option when used with For:Pro 1.8.1.1

If mounting of diskette fails when using the cp command with the -B option during a multi-volume backup, succeeding mounts will fail with the following error messages:

```
mount: mount failed: no such device or address
cp: could not mount volume on /dev/fd02
```

Corrective Action:
The current work around is to go to another terminal on the system and manually unmount the diskette (`umount /dev/fd02`), then issue a `rdconf /dev/fd02` command on the command line at the shell prompt. This will allow you to continue with your multi-volume backup. □

/u/help

Michael F. Smith of Athens, TN, sent us a list of questions which we then forwarded to **Elizabeth M. Muth, Manager of Software Services at Fortune Systems Corporation**. While follows are both the questions and the answers:

*The **ls** prgram has options **A** and **B** which are not defined.*

Answer: The **A** option is documented in the **FOR:PRO Programmer's Manual** (page 1-167). It is used to list all entries except those beginning with '.' and '..'.

*The **who** program has a **w** option which is not defined. It is used this way in the shutdown script.*

Answer: The **-w** is only pertinent for the console. It merely indicates which terminals are active and who should sign-off prior to shutdown.

*The **rdconf** program has options **U S** and **P** which are not described. These are used in the coldboot script.*

Answer: The options **U S** and **P**, in rdconf are as follows:

U - Defined users from cold boot.
S - Blocks in swap area.
P - Partition length.

*The **cp** program appears to have options **C E K** and **M** which are not defined in any of the manuals from Fortune or commercial sources. What are they?*

Answer: Sorry, I don't have the answer to this question.

The termcap file has several references to files which are not supplied with the Fortune operating system. For example:

```
/usr/lib/tabset/vt100
/usr/lib/tabset/stdcrt
/usr/lib/tabset/teleray
```

While I have not had specific use for these, it is one of my dearest pet peeves to see references to things which do not exist on the system! It would be a nice feature for Fortune Users to share any termcap or printcap which they have developed. (Ed. note: we will gladly act as a clearing house for termcap and printcap entries -- we will distribute them on disk or via our bulletin board).

Answer: The termcaps listed are for terminals not sold or supported by Fortune. In all likelihood they were developed elsewhere and reference made to them only to indicate they have been done before.

*In the **disktab** of **For:Pro 1.8.1.1** some of the drives have more than 8 heads. I had been told that the Fortune system could only handle drives with a maximum of 8 heads and 1024 cylinders. Are these drives usable on the older systems or does Fortune have a new controller for the SX70? If a new controller is used, can the older ones be retrofitted with proms to handle the large drives?*

Answer: The **disktab** for **For:Pro 1.8.1.1** includes drives with more than 8 heads. However, any drive with 9 or more heads does require a different hard disk controller card than the standard **WD** board. The other controller card is the **SCSI** board, and no, the older controller cannot be retrofitted.

*Mark Anderson wanted to know if he could use more than 3 **Comm-A** cards with his Fortune.*

Answer: **For:Pro 1.8.1.1** will not support more than 3 **Comm-A** cards, however level **2.0** will. (See announcement elsewhere in this issue regarding the release of **For:Pro 2.0.**) □

On Text Editors:

Michael Smith had one last question for us:

There are different strokes for different folks, which leads me to the word processing program. I do not have the same admiration for ***Fortune:Word*** *that others have. I am a dedicated production freak and find* ***Fortune:Word*** *to be slow and cumbersome. Yes, I am a 'dot command' type aficionado. The productivity I enjoy with* ***Spellbinder*** *on my 1978 vintage* ***Cromemco*** *with 8" drives is greater than with* ***Fortune:Word*** *with hard disk drives. This leads to the question: Is there a word processing program such as* ***Spellbinder*** *available for Fortune. Some such programs exist for unix systems but when the supplier is contacted, they do not have plans to port it to the Fortune box.*

Answer: You're right -- everyone seems to have their own favorite editor, and it often is the first one they learned. We happen to feel that some editors are better for "word processing" while others are better for program development. The good news is that there are several editors to choose from on the Fortune besides just **Fortune:Word.**

Many users have obtained the **screen** or **sc** editor. (This is available on our **D.C. Grab Bag Disk** as well as the **Fortune Utilities Disk**.) Screen is a powerful and easy to use editor that has been substantially enhanced and is now for sale from **Bell Technologies** in Fremont, CA. For more information you can contact them at 415 659-9097.

The **vi** or **visual** editor is available as part of **Fortune's Development Utilities** package. This is a very powerful editor that has several utilities that make it very good for programming -- auto indent, find matching pairs of braces, brackets and parentheses, etc. When combined with the **nroff** program, it allows very complicated formatting as well. **nroff** is a "dot command" formatter that will work with any ascii text. It is also distributed as part of the **Development Utilities**.

The **Rand** editor will be available on our next software disk. This editor is also very powerful and includes several features like multiple file editing, block moves, keystroke save files, etc. We have spoken with many people who use it exclusively.

As far as we know, the **EMACS** editor which is also very popular has not been ported to the Fortune yet.

If any of our readers have other favorites, please write.

Unipress *is now marketing a program to convert Bourne scripts to C code. But alas, they are not porting it to the Fortune. They will sell the source code for $4995 whereas the compiled program is $695. I do not have the C expertise to make such a port. However, I was thinking that if 10 or 15 readers where interested that perhaps a group purchase could be made that would entice* ***Unipress*** *to make the port or authorize a member with the requisite expertise to make the port.*

Answer: That sounds like a good idea. Interested readers should contact Mr. Smith at (615) 745-9549 in TN. □

idc™

MULTI-USER COMPUTER SYSTEMS

Specially Designed For Your Business

INTERDISCIPLINE CONSULTANTS, INC.

(201) 848-8500

235 EVERETT AVENUE
WYCKOFF, NJ 07481

FORTUNE END USERS
Do you have SUPPORT needs?

We offer services for those with:
- **frequent needs on a "monthly-retainer" basis**
- **occasional needs on a "pay-as-you-go" basis**

We're there when you need us to:
- **provide ongoing emergency programming & systems support**
- **customize programs for your present & future needs**
- **give on the job training**

FOR:PRO

INTRODUCTION TO FOR:PRO / UNIX / XENIX - WORKSHOP

DECEMBER 9, 10, 11 1986 HOUSTON, TEXAS

David Kloes, President of **UNI-KOMP**, Houston, has for over 5 years, been on the leading edge of **FOR:PRO**, **UNIX** and **XENIX** Operating Systems. Dave has taught 15 **UNIX** Workshops in the last couple of years and is the current president of the **Houston UNIX Users Group**.

In the Workshop you will learn the following:

UNIX/XENIX File Structure, Basic **UNIX/XENIX** Commands, "ed" Editor, System Administration Duties, System Backup, Standard System Files, Advanced **UNIX/XENIX** Commands, "vi" Editor, Shell Scripts, System Administration Commands, Communication Commands

The Workshop is specially designed for anyone who is involved with **UNIX/XENIX** based hardware and/or software. It's purpose is to help you realize more profit through better understanding of the operating system. You will learn how to use the basic functions of the operating system, and also actually write your own Shell Scripts. Hands-on opportunity to practice the most commonly used commands will be provided each participant.

Workshop fee is **$495** if payment is received by December 1st. After that time, the workshop fee is $550. This fee includes three days of classes and lunch each day. Group rates are available for more than one participant from the same organization.

To register, call **UNI-KOMP** today and you will be sent a registration packet. Space is limited, so call **(713) 937-7782** today to get your packet.

UNI-KOMP 9803 ROCKTREE HOUSTON, TEXAS 77040

Multiplan, continued from page 1

for the loan, and the length of the loan period. The worksheet then calculates the monthly payments, annual loan payment and total loan payment, as well as interest paid, principal paid, and principal remaining for each month of the term you specify. The principal and interest will be self-liquidating which means the amounts paid will be sufficient to retire the loan at the time of maturity. This particular worksheet will be set up to handle amortization for three years.

OK. Start Multiplan with one of the methods described last month so that you have an empty Multiplan screen. Now, turn off automatic recalculation by typing **O**, then **N** and then **EXECUTE**.

The worksheet contains only four columns, so we'll make column one 20 characters wide and columns two through four 18 columns wide to give us ample space and yet keep everything on the screen. To make column one 20 characters wide, first make sure the cell cursor is positioned somewhere in column 1 (it should already beat **R1C1** if you haven't moved it since opening up the worksheet). Now type **F** for Format, then **W** for Width, then **20** followed by the **EXECUTE** key. To set column 2 to 18 characters, move the cell cursor to column 2 (using the right arrow key). Now type **F**, then **W**, then **18** and then **EXECUTE**. Repeat this for columns 3 and 4.

Now we enter the Titles. Place your cell cursor in cell **R1C2** and type an **A** for the Alpha command. Then type **LOAN AMORTIZATION ANALYSIS**. Now use the direction keys to move the cursor to **R3C1**, the next cell that will contain a title. When you move the cursor, **LOAN AMORTIZATION ANALYSIS** moves from the command line at the bottom of the screen to its cell. You won't see this title completely until later, when you format the titles. You now see **ALPHA/VALUE** on the command line, and this means you can type another title. Use the table below to put the titles in the cells indicated:

R3C1	Amount Financed
R4C1	Annual Interest (%)
R5C1	Term in Years
R6C1	Monthly Loan Payment
R7C1	Annual Loan Payment
R8C1	Total Loan Payment
R10C1	End of
R11C1	Month
R10C2	Interest
R11C2	Paid
R10C3	Principal
R11C3	Paid
R10C4	Principal
R11C4	Remaining

After you type the last title (Remaining), hit the **EXECUTE** key.

Now we'll enter dashed lines. First we'll use the **REPT** function and an equal sign to enter a double-dashed line across row 2. Place your cursor on **R2C1** and start the Value command by typing **V**. Then type **REPT("=",20)** followed by the **EXECUTE** key. You will see these equal signs fill up **R2C1**. To extend that across the sheet we'll use the Copy command. Leave your cell cursor on **R2C1** and type **C**, then **R**, then **3** and then an **EXECUTE**. *Voila*, you have a double dashed line across your sheet.

We also want a copy of this line in row nine. Leave your cursor on **R2C1** and type **C**, then **F**, then **:4** (which specifies the last column to copy from). Now hit the **TAB** key which will highlight the to cells field. Now type **R9C1** and then hit the **EXECUTE** key. You should see our double dashed line appear across the four columns of Row 9.

Finally, we want a single dashed line in row 12. To do this, we will perform the same process that we did to put a double dashed line in row 2. Put your cursor in **R12C1**. Type **V**, then type **REPT("-",20)** and **EXECUTE**. Now move the cell cursor to **R12C2** and type **C**, **R**, **3** and **EXECUTE**. Done.

Let's format the Titles and Number cells. First, we will spread the **LOAN AMORTIZATION ANALYSIS** title into the cell to the immediate right. Put your cursor on **R1C2** and type **F**, then **C**, then **:3** (which specifies the last cell to be continuous). Next, type two **TABS**, select the **Cont** option by typing **C** and finish the command with **EXECUTE**.

Now center the end of month title (rows 10 and 11, column 1) and the cells that will contain the month numbers (row 13 through 48, column 1). Put your cursor on **R10C1** and type **F**, **C**, **:R48C1**, then **TAB**, **C** and **EXECUTE**. Now right-justify the titles in rows 10 and 11 (columns 2 through 4). Place the cell cursor on **R10C2** and type **F**, **C**, **:R11C4**, **TAB**, **R** and **EXECUTE**.

Now for the Number cells, let's make the amount financed, monthly loan payment, annual loan payment, and total loan payment formatted as dollar values. Place your cursor on **R3C2** and type **F**, **C**, **,R6:8C2**, then 2 **TABs**, then **$** and **EXECUTE**. You won't see anything happen to your sheet at this point, but you have just instructed Multiplan to display any numbers in those cells in the dollar format.

The annual interest cell, **R4C2**, and the cells in the interest paid, principal paid, and principal remaining columns will be set to a fixed format with two decimal places. Place your cursor on **R4C2** and type **F**, **C**, **,R13:48C2:4**, then 2 **TABs**, then **F**, then **TAB**, then **2** and **EXECUTE**.

Continued on next page

As a final touch, we'll have Multiplan put commas in large numbers. Leave your cursor on **R4C2** and type **F**, **O**, **Y** and **EXECUTE**.

Entering the Formulas

In total, there are 9 formulas that we will put into this worksheet. Some are relatively simple and some are a bit more complex. We will present each of these formulas in turn, as presented by Ruth Witkin, and try to make them understandable.

Formula 1 will amortize the amount financed (**R3C2**) in a series of equal monthly installments including principal and interest (**R6C2**). To convert the annual interest rate into a monthly interest rate, the formula will divide the annual rate by 12. This will appear as **RATE/100/12**. To convert the term from years to months, the formula will multiply the length of the term by 12 and will appear as **TERM*12**. Formula 1 begins with an open parenthesis, one of the symbols that prepares a cell for a formula. Therefore, you don't have to type V (for Value) to prepare the cell to receive the formula. To enter the formula, place your cursor on **R6C2** and type:

(LOAN*RATE/100/12)/(1-(1+RATE/100/12)^(-TERM*12))

Finally, type the **EXECUTE** key. (Note: the **^** is the symbol that causes the preceding term to be raised to a certain power. In this formula, the term **(1-(1+RATE/100/12)** is raised to the power **(-TERM*12)**.)

Formula 2 will give you the annual loan payment by multiplying the monthly loan payment by 12. Place your cursor on **R7C2** and type **V**, then **PAY*12**, then **EXECUTE**.

Formula 3 gives you the total loan payment during the life of the loan. The formula **PAY*TERM*12** does this and to enter the formula, place your cursor on **R8C2** and type **V**, then **PAY*TERM*12**, then **EXECUTE**.

Formula 4 will place month numbers in column 1. There are various ways of doing this. One simple way is to use Multiplan's **ROW** function. If we just enter **ROW()** into a cell in a worksheet, Multiplan will display the current row. If we were to put this into **R13C1**, Multiplan would display **13**. However, we want to display **1** at this point because it is the first month. In **R14C1**, we want to display 2 because it is the second month and so on. So, we will enter **ROW()-12** which will put the proper number in the cell. First, we will put this formula in **R13C1** and then use the **COPY** command to copy this formula into the other rows that need this formula. Place your cursor in **R13C1** and type **V**, then **ROW()-12**, then **EXECUTE**. Next, type **C**, then **D**, then **35** and **EXECUTE**. Now, if you press the **F7** key to recalculate, you will see the numbers appear.

Continued on next page

UNISOFT INC.

Experts in Computer Consulting - Sales - Service

Presents

FORTUNE SYSTEMS HELP

We Provide:

- Expert Trouble-shooting
- Years of programming and accounting experience
- *Free Initial Consultation*
- Training Available for:
  - * BASIC/IDOL/BAS and Multiplan
  - * UNIX and Ext. Fortune:Word
- Hardware Repairs & Upgrades
- FORTUNE trained experts at a cost that will surprise you!
- *Trade-Ins Welcome*
- Specializing in:
  - * Hard Disk Trade-In
  - * Hard Disk Upgrades

Authorized Fortune Master Dealer

Authorized Fortune Systems Repair Depot

CALL: (303) 671-8600 EMERGENCY HELP: (303) 691-HELP

UNISOFT INC. 14100 East Jewell Avenue Suite 27 Aurora, Colorado 80012

Multiplan, continued from page 19

Formula 5 will be used to calculate the interest paid on the loan in the first month (**R13C2**). Place your cursor on **R13C2** and type **V**, then **LOAN*RATE/100/12**, then **EXECUTE**.

Formula 6 will appear a bit complicated but it demonstrates a nice use of the **IF** statement which is to blank out cells which would otherwise contain values that we don't want displayed. In this case, if our loan is paid off in less than 36 months (the current extent of our worksheet), then formula 6 would display zeros in the remaining spaces. This is not pleasing to the eye, so we use the **IF** statement to calculate the value only if we are in the life of the loan, otherwise the formula will display nothing (which is what "" does).

The purpose of this formula is to calculate the interest paid in the second month (and remaining months after we copy it down the column). The basic formula is **R[-1]C[+2]*RATE/100/12.** (Note: the notation **R[-1]C[+2]** is called a relative cell reference and what it says is to take the value in the cell one row up (-1) and two columns to the right (+2) and then multiply it by RATE/100/12).

This is what the formula will look like:

IF(RC[-1]<=TERM*12,R[-1]C[+2]*RATE/100/12,"")

The **TEST** portion in the **IF** statement (i.e. **RC[-1]<=TERM*12**) will compare the term in months with the month number in column 1 (which is what **RC[-1]** refers to). **IF** the number in column 1 is less than or equal to (**<=**) the term in months, the formula will use the formula after the first comma (called the **THEN** statement) to calculate the interest paid. If the number in column 1 is greater than the term in months, the formula will use the formula after the second comma (called the **ELSE** statement) to make the cell appear blank. Again, the **ELSE** statement ("") will prevent a long string of zeros from appearing down the column after the monthin which the loan is currently being liquidated.

To enter this formula, place your cursor on **R14C2** and type **V**, then **IF(RC[-1]<=TERM*12,R[-1]C[+2]*RATE/100/12,"")**, and then **EXECUTE**. To put this formula in the remaining cells in column 2, type **C**, then **D**, then **34** and then **EXECUTE**.

Formula 7 will again use the **IF** statement. We will calculate the principal paid at the end of the first month (**R13C3**) and deduct the interest paid (**R13C2**) from the monthly loan payment. As before, the **ELSE** statement will prevent zeros from appearing when the month number is greater than the term that you select. Put your cursor on **R13C3** and type **V**, then **IF(RC[-2]<=TERM*12,PAY-RC[-1],"")**, then **EXECUTE**. Now copy this down the column by typing **C**, then **D**, then **35** and **EXECUTE**.

Formula 8 will deduct the principal paid (**R13C3**) from the amount of the loan to give you the principal remaining at the end of the first month. Place your cursor on **R13C4** and type **V**, then **LOAN-RC[-1]**, then **EXECUTE**.

Formula 9 will do the same thing but in the context of an **IF** statement so that we do not produce any unsightly zeros. You might ask, at this point, why do I have to put in two formulas? Why don't I put the same formula in **R13C4** that I am going to put into **R14C4**? After all, they both deduct the principal paid from the amount of the loan to give me the principal remaining at the end of the current month, don't they?

Well, yes they do. However, let's take a look at the **THEN** statement of Formula 9: **R[-1]C-RC[-1]**. Now remember Formula 8: **LOAN-RC[-1]**. The difference between these two formulas is that Formula 8 uses the initial starting value of the loan, which is in our cell called **LOAN**, so Formula 8 must start with the proper value. After we start, then our other formula, Formula 9, will simply look in the cell one row up in the sheet for the proper number to start with. This is a process that you will use often in creating worksheets - many times the first occurence of the formula will be a slightly special variant and then each formula below will be the same. This is the same thing we did with Formulas 5 and 6.

To put Formula 9 in place, position your cursor on **R14C4** and type **V**, then **IF(RC[-3]<=TERM*12,R[-1]C-RC[-1],"")**, then **EXECUTE**. Copy this formula down the remaining 34 cells with the same technique that we have used before.

Now that we are done with the formulas for this model, we want to **LOCK** them into place. You

Figure 1: Finished Multiplan Worksheet

```
#1          1                2                3                4
 1                  LOAN AMORTIZATION ANALYSIS
 2 ==================================================================
 3 Amount Financed            $90,000.00
 4 Annual Interest (%)             15.00
 5 Term in Years                      30
 6 Monthly Loan Payment        $1,138.00
 7 Annual Loan Payment        $13,656.00
 8 Total Loan Payment        $409,679.86
 9 ==================================================================
10        End of                Interest        Principal        Principal
11        Month                    Paid             Paid        Remaining
12 ------------------------------------------------------------------
13          1                  1,125.00            13.00        89,987.00
14          2                  1,124.84            13.16        89,973.84
15          3                  1,124.67            13.33        89,960.51
16          4                  1,124.51            13.49        89,947.02
17          5                  1,124.34            13.66        89,933.36
18          6                  1,124.17            13.83        89,919.52
19          7                  1,123.99            14.01        89,905.52
20          8                  1,123.82            14.18        89,891.34
COMMAND: Alpha Blank Copy Delete Edit Format Goto Help Insert Lock Move
         Name Options Print Quit Sort Transfer Value Window Xternal
Select option or type command letter
R1C1                                  70% Free      Multiplan: AMOR1
```

Continued on next page

can do this by leaving your cursor on **R14C4** and type **L**, **F**, and finally **Y**.

We are being careful, so now is another judicious point at which to save the worksheet. Do that by typing, **T**, **S**, (Multiplan should display the name **AMOR**, so there is no need to retype it), and **EXECUTE**. Multiplan will recalculate the worksheet and then ask, **Overwrite existing file (Y/N)?** Answer with a **Y**.

Now try it! Place your cursor at **R3C2** and type the number **90000** and press **EXECUTE**. Now press **F7** to recalculate and presto-chango, your spreadsheet should look like figure 1.

This particular worksheet was constructed so as to handle loans of up to 3 years which is 36 months. This can be easily modified so that loans of up to 20 years can be computed. Recall that Multiplan allows us to use 255 Rows. In this worksheet, however, we have used only 48 Rows. Therefore, if we want to extend this worksheet, we need only go to Row 48 and copy down the formulas in R48C1, R48C2, R48C3 and R48C4 down 207 Rows. If you did this, you could calculate loans for slightly over 20 years.

Conclusion

In this article we presented some basic Multiplan techniques and worked through a Loan Amortization worksheet to demonstrate some of these techniques. Next month we will continue with another worksheet and introduce some more concepts.

We know that many of our subscribers use Multiplan extensively and consequently we are sure that there must be some useful and interesting worksheets that you have developed. We ask you to send them along to us, along with instructions on their use. In return, we will package them up on a diskette and make them available to anyone who is interested. □

Expand your Fortune® without spending a fortune.

New equipment costs big bucks. The Good Fortune Trader sells fully tested burned-in equipment, refurbished to Fortune® standards at prices that won't make waves with your cash flow. Call or write today for a listing of our specials.

THE GOOD FORTUNE TRADER

60 DAY WARRANTY

a division of Computer Services of Atlanta

(404) 392-9938

1534 Dunwoody Village Parkway/Atlanta, Georgia 30338

Western Blue Corporation

is pleased to announce the addition of

JACKIE BIGGS

As a Member of the Technical Support Department
(Formerly of Fortune Systems Corporation)

Western Blue Corp. Offers Full Support of:

Unix	Multiplan
Bas	Informix
Fortune:Word	Communications

Western Blue Corporation
1720 S. Amphlett Blvd.
Suite 130
San Mateo, CA 94402
415-345-2583

The Basic Advisor, continued from page 1

who took the time to look me up in San Diego. Having friends around made the conference a more enjoyable experience. **Mr. Steve Rosenfeld** of **Superior Computer Systems** was unable to attend due to a schedule conflict, but he sent his best wishes. We hope to be with Mr. Rosenfeld next year in Florida.

This year's conference was the first **Thoroughbred Dealer** gathering since 1981. (1982-1985 represented the lean, SMC years.) In 1981 there were a total of 30 companies in attendance. This year 150 people representing 95 companies participated in the activities. In addition to dealers and manufacturers from The United States, we met people from such places as Canada, Switzerland, Sweden, The United Kingdom, Australia, Puerto Rico, Jamaca and South Africa. It was obvious to all that John-L Johnson has been abundantly successful in his first year at the helm.

And, to our surprise, we met **Mr. Paul Olin**, Vice President of Sales, **Mr. Doug Krause**, District Manager in California, and **Ms. Elizabeth Muth**, Manager of Software Services, from Fortune Systems. These three, along with others, represent a new wave of management at Fortune. I must say that I was impressed. Having now had the opportunity to put my hands on many of the latest small business computers, I still believe the Fortune machines are among the best available. Maybe this new team can finally turn the company into a major manufacturer.

John-L, himself a native Texan, threw a wild west party on one of the evenings, complete with barbecue, "bull shooters" (an alcoholic concoction which completely neutralizes gray matter) and square dancing to a live Country and Western band. In this relaxed atmosphere Dave Kloes and I were able to privately discuss with John-L the future of Thoroughbred Software and Fortune Systems. (I'll hasten to assure you that neither Dave nor I chose to sample the "bull shooters".)

In this relaxed atmosphere Dave Kloes and I were able to privately discuss with John-L the future of Thoroughbred Software and Fortune Systems.

John-L explained that Concept Omega would not be able to support the Fortune Systems BAS Applications that are currently being used, because Fortune (or someone) has dramatically modified the original programs. He feels, as I do, that those installations should continue using the network of Fortune support companies. However, if a Fortune user wishes to purchase the **Thoroughbred Business Applications** in lieu of the **BAS** version, Concept Omega will offer dealer support for those packages.

Concept Omega also offers a line of word processing products which may be installed on a Fortune. But John-L admits that there is no better word processor on the market than **Fortune:Word**. I imagine too that **Multiplan** is a better spreadsheet choice for Fortune than Concept Omega's **MODEL CONCEPT**.

Other Concept Omega products which may be of interest to Fortune Systems users are the BASIC LIBRARIAN, a full-screen text editor for Business BASIC, CONCEPTG, interactive business graphics, and CONCEPTR, a new, easier-to-use report generator. I imagine your dealer can get these products for you. If not, give us a call.

That covers just about everything except **IDOL II**. Yes, sometime next year IDOL II will be available for the world, and that includes Fortune Systems users. IDOL II is a part of Concept Omega's 4th Generation Software. Just what is 4th Generation Software? For that matter, what is IDOL II? Next month, both of these questions will be answered in /u/fortune news. I will cover 4th Generation Software, and Dave has graciously offered to write about IDOL II. We'll see you then.

Also coming up next month will be our annual call for software support companies for Fortune Systems. We hope to publish the list in the January, 1987, issue. Please take note! I don't want anyone to feel left out this year! □

ATTENTION FORTUNE SYSTEM OWNERS

Paying for a new power supply ?
Fixing or upgrading your present one could save you hundreds of $$$$.

Repairing	(Zenith)	$ 75.00
	(Western Electric)	$ 95.00
Upgrading of Zenith to 250 watts		$ 300.00

Terminal supplies repaired also.

Repaired supplies warranted for 90 days.
Upgraded supplies warranted for 1 year.

Quantity discounts available. (Call for quote.)

POWER SUPPLIES "R" US
5916 Southwind Dr.
San Jose, CA., 95138
408-365-7678

Dear Fortune Owner:

As you have probably read in */u/fortune news*, Fortune Systems SX models are significantly more powerful and faster than the PS and XP models. If you have a PS or XP model system, you should take advantage of our special promotion on SX upgrades. For a limited time, and while our supplies last, we will upgrade your system, including the new SX Cabinet at no additional charge for the cabinet. With this special offer, you may choose any combination of the following upgrades and/or expansions:

- *SX Processor with Current Release of Multi-User For:Pro*
- *60 MB Internal Streaming Tape*
- *30 MB, 45 MB, 70MB High Speed Hard Disk*
- *6-Port Intelligent Asynchronous Controller*
- *Upgrade to Current Release of all Fortune Software*
- *Fortune:Windows (Included at NO Charge)*

Additionally, we can perform these upgrades within 24 hours of receiving your system. In fact, if you ship your system overnight on a Friday for Saturday delivery, we will have it back to you on Monday morning, all ready to plug in and run; not losing the use of your system for a single work day. And, we will issue you a full 90-day warranty on the entire system.

JFM provides services to Fortune Systems owners nationwide either direct or through our authorized Affiliated Dealers. If you haven't used our services, I invite you to discover JFM; you will be glad you did.

Sincerely,

Morris Frydman
President

Talk to us, then talk to our clients

JFM BUSINESS SYSTEMS
The Merchandise Mart
Lobby Suite 144
Chicago, Illinois 60654
Call (312) 644-0870

Authorized Fortune Systems Master Dealer

I WANT YOU

IF YOU ARE USING THE BAS PAYROLL PACKAGE ON YOUR FORTUNE SYSTEM, I WANT YOU TO KNOW THAT YOUR CURRENT SYSTEM WILL NOT CORRECTLY PRINT YOUR W-2 FORMS. THE FORMAT OF THE W-2 HAS BEEN CHANGED AND YOU NEED AN UPDATE TO CONFORM TO THESE NEW REQUIREMENTS FROM THE INTERNAL REVENUE SERVICE.

*** NOW AVAILABLE — UPDATE DISKETTE ***

MAIL YOUR CHECK FOR $29.95 TO:

LMI INFORMATION SYSTEMS
RE: W-2 PROGRAM UTILITY UPDATE
94A WESTPARK ROAD
DAYTON, OHIO 45459
PHONE (513) 439-4321

NOTE: OHIO RESIDENTS ADD 6% STATE SALES TAX

FACTS: Your current software is

tired
inflexible
out of date
hard to modify

FACTS: You long for software that is

FLEXIBLE! (many user selected options)
MORE COMPREHENSIVE DATA!
UP TO THE MINUTE!
COMPLETE WITH LITTLE OR *NO* MODIFICATION!

Let us send you more information on the software that has the SOPHISTICATION you need and want at a cost that is SURPRISING! Yes, it's worth the investment to change software rather than do without what you need (what you thought was in your current software, BUT IS NOT), or to pay a great deal of money for modifications.

FACTS:
Order Processing
Inventory
Accounts Receivable
Accounts Payable
Sales Analysis
General Ledger
Payroll
Menu and Security System
Job Streaming

FACTS: Software that's not like all the rest. It's better.

COMPUSYSTEMS
OF GEORGIA, INC.

2200 Northlake Pkwy
Tucker, Georgia 30084
(404) 491-8004

Coming soon. . . Using a Laser Printer with your Fortune; How to use the **Rolodex** program on our latest software disk;. Thoughts on **Fortune:Word 3.0**; What's in Fortune's **Developoment Utilities;** How Fortune Systems uses Fortune computers; New Product Announcements From Fortune; and more.

/u/fortune *news*
73 Jason Street
Arlington, MA 02174

/u/fortune* *news...*

The Newsletter for Users of the Fortune 32:16 Computer November1986/Volume 3 Number11

An ElectronicRolodex

*(Editor's Note: Last month we announced a new software diskette in our library -- **UNIX TOOLS**. One of the programs included on this disk is a program called rolo. This article describes its use. For a copy of the **UNIX TOOLS** diskette, use the order card enclosed, or mail $10 for shipping and handling to the **Cambridge Consortium** at the return address on the back page.)*

Computers are becoming part of everyday life in many businesses -- they are used for word processing and accounting, and ocasionally for database applications. It's our experience that most users find the time to master a program or two, but they are too busy going about their actual work to spend time to really explore their computers. It has often seemed strange that given the inherent power of Fortunes that they haven't infiltrated further into our daily activities. The **Electronic Rolodex** we will discuss in this article provides a solution to this dilemma -- it is extremely easy to use and can replace and enhance one piece of standard desktop equipment, the rolodex.

What is an Electronic Rolodex

With the **Electronic Rolodex** you can store people's names, addresses, company names, phone numbers and any comments that are relevant. Then the computer can easily search for an "entry" using the person's name or any of the other fields mentioned above. This makes it an easy matter to locate someone if you can only remember their first name, city, or street, etc. It's also easy to find someone using their company name. This can be tedious in a manual rolodex where cards are rarely cross-indexed. In a sense, the **Electronic Rolodex** is like a database program, but since it is specialized for this purpose, it is smaller and less complicated.

The **Electronic Rolodex** is not a mailing list program -- it isn't set up to print labels. It's main function is an online database of phone numbers and addresses, which it does superbly well.

Installing Rolodex

If you get our **UNIX TOOLS** diskette, all you will need to do is follow the instructions from the **S5 Install** selection off of the **Global Menu** to install the rolodex. This will copy the **rolo** program to your hard disk. If you want to be able to use the rolodex program from the **Global Menu**, you will have to find an unused number and install it there. The program installation is very simple.

In addition to the actual program, the *data files* for your phone numbers will automatically be created. In this program, each user on your system has their own

See **Rolodex,** page 4

Featured in this Issue...

Basic Advisor -- Ray Wannall discusses 4th generation software...Page 3. In a related article, David Kloes presents **IDOL II**...**Page 17**.

/u/help -- this month Josh Lobel helps with problems relating to **file permissions** and **kermit**. Also, he shows you how to display your directory as a prompt...**Page 15**.

Getting Started with Multiplan -- In this third part, we present a spreadsheet which automatically keeps and reconciles your checkbook...**Page 1**.

The UNIX Directory -- David Kloes continues his explication of the some shell programming commands and introduces the **while** loop...**Page 12**.

Rolodex -- This electronic Rolodex is both fast and useful. Find out more about it...**Page 1**.

Fortune's third quarter results...**Page 2.**

Multiplan

Check Ledger and Reconciliation

In the last two issues of **/u/fortune** *news* we introduced you to **Multiplan** and walked through an implementation of a worksheet which amortized loans. This month we will tackle another, slightly more involved, worksheet.

As a business owner, home owner or just plain citizen, you probably face, each month, the tedious task of **reconciling your check book**. Spreadsheets are uniquely suited to automating much of this process and Multiplan is no exception. This month we will build a check ledger and reconciliation worksheet that will be useful for most people.

Again, the information for this article can be found in **Ruth Witkin's** *Managing Your Business with Multiplan*. As she says, "This check ledger spreadsheet will maintain your checking account automatically, give you an accurate running balance, and provide a one-page record of your monthly transactions. The transactions summary can help you analyze your monthly cash flow and identify income and expense relationships, and it can become the basis for a cash budget and other

See **Multiplan**, page 7

*/u/fortune *news* is published by The Cambridge Consortium, Inc. It is not affiliated with Fortune Systems Corporation.

Expand your Fortune® without spending a fortune.

New equipment costs big bucks. The Good Fortune Trader sells fully tested burned-in equipment, refurbished to Fortune® standards at prices that won't make waves with your cash flow. Call or write today for a listing of our specials.

THE GOOD FORTUNE TRADER

60 DAY WARRANTY

a division of Computer Services of Atlanta

(404) 392-9938

1534 Dunwoody Village Parkway/Atlanta, Georgia 30338

/u/fortune *news*. . .

Editors: Josh Lobel, Mark Palmerino
Associate Editor: Benzion Chanowitz
Technical Editor: Scott Bradner
Advertising Manager: Mark Palmerino

/u/fortune *news*. . .is published 12 times per year by The Cambridge Consortium, Inc. Neither **/u/fortune** *news* nor The Cambridge Consortium are affiliated with Fortune Systems Corp. Contact us regarding subscription rates. Quantity discounts are also available.

Back issues are offered at the following prices:

Vol 1, Numbers 1-3	$1/each	Vol 2, Number 1-7	$3/each
Vol 1, Number 4	$.50/each	Vol 3, Number 1-	$4/each
Vol 1, Number 5-8	$3/each		

Although every attempt is made to insure that our contents are error-free and without risk, The Cambridge Consortium is not responsible for losses that occur due to correct or incorrect interpretation of **/u/fortune** *news*.

For more information about us, please call or write:

/u/fortune *news*, 73 Jason Street, Arlington, MA 02174. Telephone (617)646-5603.

The text for this newsletter was prepared using a Fortune 32:16 for text input. The information was then transferred to an Apple Macintosh Plus with kermit and formatted for typesetting using the Aldus Pagemaker program.

UNIX™ is a trademark of Bell Laboratories,Inc. Fortune 32:16™, Fortune:Word™ and FOR:PRO™ are trademarks of Fortune Systems Corporation. Multiplan™ is a trademark of Microsoft. IBM-PC™ is a trademark of IBM. MS-DOS™ is a trademark of Microsoft. Macintosh Plus™ is a trademark of Apple Computer. Pagemaker™ is a trademark of Aldus Corporation.

©1986 by The Cambridge Consortium, Inc. 10/86

Fortune Systems Corporation Reports Third Quarter Results

(Editor's Note: The following information is based on a press release from ***Fortune Systems Corporation****)*

(Belmont, CA, Oct. 28, 1986) **Fortune Systems Corporation** today reported revenues of $6,063,000 for the third quarter ended September 30, 1986, compared to $8,188,000 for the third quarter ended September 30, 1985. Despite lower revenues for the quarter, the Company reported a reduced loss of $2,933,000 or $.14 per share, a 33% improvement over the comparable period loss of $4,380,000 , or $.21 per share.

For the nine months ended September 30, 1986, the Company reduced its losses by 71% to $2,335,000 or $.11 per share, on revenues of $26,438,000. Revenues for the comparable 1985 period were $32,625,000 with a corresponding loss of $7,950,000, or $.38 per share.

"As previously announced, the impact of restructuring and implementing a new reseller marketing strategy, coupled with reduced Ford revenues and costs associated with the introduction of our new Fortune Formula supermicro-computer product line, depressed revenues and caused the Company to report a loss for the third quarter," stated Fortune president and chief executive officer James S. Campbell. "During the quarter, Fortune also sold its German subsidiary," continued Campbell. "That sale is consistent with our strategy to have an independent distributor in place in each of the international countries in which we do business.

"As a result of the quarterly loss, cash balances decreased to $19.3 million," stated Campbell. "Such balances remain adequate to support our computer hardware business, new software business and acquisition strategy."

Condensed Consolidated Statements of Operations

Quarter Ended September 30:	1986	1985
Net revenues	6,063,000	8,188,000
Cost of revenues	4,821,000	6,611,000
Gross profit	1,242,000	1,577,000
Operating expenses	4,924,000	6,315,000
Operating Loss	<3,682,000>	<4,738,000>
Gain from sale of German subsidiary	471,000	--
Interest Income, Net	278,000	358,000
Income tax credit and reversal through extra-ordinary charge	--	--
Net Loss	<2,933,000>	<4,380,000>

□

The Basic Advisor

Ray Wannall is President of BaSiC Software Corporation in Baltimore and is contributing to this publication independently.

What is 4th Generation Software? As a software dealer, I am forever being approached by software manufacturers to represent their products, and for about two years, many have described their wares as being "4th Generation". I never really paid much attention to this claim, primarily because I had no idea what they were talking about. I am still foggy on "Level 2.1.3.1.1", which must be very exciting, because ads for this stuff always end with several exclamation points.

As it turns out, these "4th Generation" claims were a lot of baloney (and I thought I could trust salespeople). It wasn't until the beginning of this past summer that a definition of the phrase was created. This was done by the U. S. Department of Commerce, National Bureau of Standards in a special publication called NBS 500-138 (which would have been much more impressive had it been named NBS 500-138.1.3.1.1). It seems to me that the trend is to eventually put the software technicians out of business, although 4th generation will certainly not do it alone. Nevertheless, new software will be much more friendly for the nontechnical user. Let's look at the defined features and see how they stack up against the **IDOL/BAS** we have been using for years.

1. A 4th generation package will include an integrated DBMS (Data Base Management System). The **IDOL** on the Fortune Systems meets the defined standards here, but **IDOL** is not what anyone could call a friendly system for the nontechnician. The new **IDOL II** will exceed 4th generation requirements, but it, too, will operate best under the fingers of a pro. I consider **IDOL** in combination with **Business BASIC** one of the best development tools on the market today, and I don't believe anyone will come up with a better DBMS than **IDOL II** with 4th generation software.

2. A 4th generation package will contain a nonprocedural subset geared for the nontechnical end user. A nonprocedural language is one that is more problem-oriented than a procedural language, which focuses more on computer resources such as hardware and software. Nonprocedural languages are easier to learn and use and require less time to program, but are not nearly as flexible as procedural languages. Your **IDOL Program Generator** is an example of a nonprocedural subset which is geared for professionals. At this time there is not a nonprocedural subset available for the layman in the **IDOL/BAS** software.

3. 4th Generation software must provide nonprocedural and procedural subsets geared for professional data processing personnel. **IDOL** and **Business BASIC** have been doing this very efficiently for many years.

]4. 4th Generation software will include a screen generator. This is provided by **IDOL** on the Fortune under selector 2, "DEFINE A CRT SCREEN".

5. 4th Generation software will feature Operating System independence (or transparency). I believe that most modern application languages, certainly including **Business BASIC**, fulfill this requirement.

6. There will be a ten-to-one improvement in programming productivity as measured in lines of code when compared with the 3rd generation language. This, to me, sounds like typical governmental hogwash. I'm no lawyer, but I believe I can meet the requirements of this section by simply stringing ten numbered lines together in one program line. This would not significantly reduce the processing speed, nor would it make the program code any easier to understand. Come to think of it, it would be a complete waste of my time!

7. 4th generation software will provide a report generator. We all know **IDOL** has a report generator.

8. 4th generation software will include a query language. There is not a query language available at this time with **IDOL/BAS**. As a matter of fact, this will be the focus of **Concept Omega's** R & D work for 1987. It will be very interesting to see what they consider a good query language.

Reviewing the parameters in this publication, we can see that 4th generation languages will not be program generators, but they will be a step in the right direction. I figure we will still be at least three generations away from the day when a novice will be able to buy a computer off the shelf and create his own custom applications. Until that time, I do not plan to retire.

***Question**: I wrote to you back in August (/u/fortune news, Volume 3, Number 8) about changing the precision on a data element which is used in several IDOL-defined files. Per your request, I am letting you know that I have completed the project. I found that in addition to the files you listed as needing to be altered, one must also make adjustments to the file **UBSQ**. Thanks for your help.*

Answer: I am amazed you were able to do that in such a short amount of time. I honestly figured to be answering your next call from my room in a nursing home. Thank you for the **UBSQ** tip.

***Question**: Update on the **BASIC** bomb on **1.8.1.1** (/u/fortune news, Volume 3, Number 8). I mentioned that when the **ESC** key is hit after a program has been waiting for operator input **BASIC** is released. Tracing events before the bombing I found that it occurs each time a **SHELL** command is used before the **INPUT** statement. However, if the **ESC** is touched immediately after the prompt is displayed, it works as expected.*

*Also, I have a programming tip that some of your readers may be able to use. In working with code written by several other programmers, I've noticed that they are not aware that **KEYs** may be used for look ahead and control breaks. They read the record before examining the contents to determine if a control break is required when the information is in the key. The following works well for me, and it makes programming code quicker and easier:*

```
K$=KEY(X,END=NNNN)
READ (X) VARIABLES
```

See **Basic Advisor**, page 17

Electronic Rolodex, continuyed from page 1

individual rolodex data file. This is convenient because it makes smaller databases and also allows people to keep some information somewhat under their own control (other people will not be able to modify entries). If you use this feature, it is also possible to view other people's rolodexes if you choose to. We'll describe how to do this shortly.

If you desire, you can link everyone's data files together so that they are all merged into one big datafile. You would do this by following these steps:

Login as manager

Install the rolodex program.

Get to a UNIX shell by typing an exclamation point from the **Global Menu**.

Start the rolodex program by typing `rolo <CR>`. Once it comes up, you can exit it by typing a backslash ("\") followed by a RETURN (<CR>). This will bring you back to the # sign. You have now created the empty data files.

Change the permissions on the files by typing:
`chmod 666 .rolo* <CR>`

Then copy the files to each users directory that you would like to have access to the rolodex directory. For instance, if you have a person with the account **/u/john**, you would type:

```
cp -l .rolo* /u/john  <CR>
```

The hyphen ell argument (`-l`) in the copy command tells the system to make a *linked* copy of the files -- that way when changes are made by one person, they will be seen by everyone. Repeat the above command for each user you want to share datafiles.

Using the Rolodex Program

You can start the program up in several ways. If you are at a shell prompt such as `$` or `#`, you can just type the word `rolo <CR>` and the program will come up. If you have installed it on the **Global Menu**, you can make the appropriate selection from the menu. If you are inside a **Fortune:Word** document, you can hit the `F13 Command` key and when it asks "`Which command?`" simply type in `!rolo <CR>`.

Once you have brought up the program, you will see the following screen:

```
        TOP LEVEL MENU

  +  :  add a new entry
  %  :  scan the rolodex entry by entry
  $  :  search by some item other than 'Name'
  *  :  save any changes made so far
  !  :  pretty print the rolodex database to a disk file
  \:    exit (any changes you've made will be saved)
```

To search for an entry by name, just type in the name and press `RETURN`.

The first thing you will want to do is to add some names to your rolodex. As you can see, if you hit a "+" <CR>, the program will prompt you for a new entry. The first thing it asks for is the person's name. We suggest you enter it as "Lastname, Firstname". This will allow the computer to create a sorted printout if you wish. Next you will be prompted for work and home phone numbers. *Note that you cannot use the numbers on the keypad to enter these numbers.* Next you will be asked for company name and address. When you are typing the address line, you can include a semicolon on the line to indicate line breaks. For instance, our address might be entered as "73 Jason Street; Arlington, MA". When this displays on the screen, it will be broken into two lines. If you realize you have made a mistake on a line that you have already passed by, you can hit the up-caret (^ -- `SHIFT 6`) to back up to the problem line.

After you have been prompted for all of the standard fields, you can enter your own "custom" fields by responding with a "`y`" <CR> when it says "`Additional fields?`". You can then add labels and fields of your own choosing. Later you will be able to "scan" the rolodex according to the name of your defined field, so be consistent when assigning names. For instance, if your company sells soaps you might want to include a field for "**product lines**", where you would list the different detergents your customers are interested in. If you always call this additional line "**product lines**" as opposed to "**product line**" or "**products**", you will be able to scan on it. For instance, you could find all of your clients who use Mr. Bubble.

Finishing an entry

When you have filled in all of the information you desire for a particular person, you will get the message "`Add new entry to rolodex?`". This question refers to the person you just entered, so unless you have made some mistake, the answer to this question is `y` <CR>. When you hit return, the person is added to the database.

Locating a person

The easiest way to locate an entry from the **TOP LEVEL MENU** is to just type part of the name you are looking for. It is not necessary to make a menu selection at all. It doesn't matter whether it is upper or lower case, and you only need type enough to tell the program who you are looking for. If you are trying to find Wolfgang Schmidt, you could probably just type Wolf and the computer would zero in on it. On the other hand, if you're looking for John L. Smith, you might have to type the whole name in because of the multitude of Smiths. If more than one entry matches your search string, the computer will display them all and ask which one you want.

If you choose, you can also locate an entry based on any of the other fields. Note that on the **TOP LEVEL MENU**,

Continued on next page

a $ indicates a search by some item other than 'Name'. When you enter a $, you will be asked the name of the field you will be using. In our soap expample, we used "**Product lines**". Then you will enter the search string, and all matching entries will be selected.

To update or delete an entry, first find it by searching. Once it is on the screen a menu will prompt you to update or delete the entry.

Other options from the Menu

You have several other choices from the **TOP LEVEL MENU** which allow you to scan all of the entries or search by a field other than the name. When scanning, if you press `RETURN` you will proceed to the next entry, if you press "<" `<CR>` you will go back to the previous entry in the database. The "`pretty print the rolodex file`" option does not work with this version, although it is possible to print a list of names and phone numbers using command line arguments, which are discussed below. No matter where you are in the program, the backslash ("\") will exit that level (like the `CANCEL/DEL` key in **Fortune:Word**) and typing a question mark will give you help on whatever operation you are attempting.

Command line arguments

If you call the **rolo** program from the shell or from **Fortune:Word**, it is possible to provide special options to the program. The most basic is to include someone's name, for example, `rolo lobel <CR>`. This will quickly produce any listings that match **lobel**. If there are multiple listings, you are asked to select the one you want, or you can scan through the all of the entries that match your search criteria. You can also place more than one name on the command line, e.g. `rolo lobel smith . . .` This command will first bring up the **lobels**, then the **smiths**, etc.

If you want to use another user's rolodex file, you can use the "**-u**" flag, e.g. `rolo -u josh <CR>` will make the program work with Josh's rolodex. This option is useful if you want to peer into someone else's address book, or view the master phone list, etc.

Finally, you can produce a printout of names and phone numbers sorted by the name field by typing `rolo -s <CR>`. This will produce a list on the screen. If you want it to pause as it scrolls by, you can pipe it through more by using the vertical bar on the left hand side of the keyboard:

`rolo -s | more <CR>`.

Likewise, it can be sent to the printer with the following command: `rolo -s | lpr <CR>`.

The **Electronic Rolodex** will quickly become indispensible and requires very minimal setup time. We encourage all Fortune users to get it from us. □

Josh Lobel

Our Memory Doesn't Cost You A

Would you like to:

Speed up your computer?
Add more users?
Use the latest software, like Fortune:Windows™?

Compatibility:	The F1MB is absolutely 100% compatible with Fortune's own 1 Meg memory board.
Affordability:	USA is proud to have equaled and exceeded the quality and performance of our competition, while offering substantial savings for both the dealer and end user.
Dependability:	All boards are burned in for 72 hours and tested for 48 prior to shipment. *Our One Year Repari or Replace Warranty underlines our confidency in the product.*
Repairability:	The F1MB facilitates repair in the field as it is 100% RAM socketed. Now you can repair a board by simply replacing a defective chip rather than face an expensive baord exchange -- reducing maintenance costs and system down time.

If you have a Fortune SX, XP45 or recent model XP30, you can use our board by simply plugging it in. Older systems can be upgraded to use our board with a simple and economical modification.

For Details Call: ***Upgrade Service Associates***

3 Marion Road, Burlington, MA 01803 Phone: (617) 239-8147

Our Guarantee: If you're not completely satisfied within 30 days, *USA* will gladly refund your money.

MAKE YOUR FORTUNE GROW

One of the greatest assets of your Fortune Systems 32:16 is its built-in ability to expand to match your need. Whether you want to increase your number of workstations, upgrade system capability, or tie-in your Fortune Systems cluster to a mainframe, Micro Resources can give you the prompt, complete service you need.

As *the* Fortune Systems Master Dealer in Ohio, we have the hardware and software you need available for immediate shipment. From Fortune:Link to uninterruptible power supplies, from storage expansion systems to Development Utilities, we have the exact blend of hardware, software and systems expertise needed to make your Fortune Systems 32:16 do exactly what you want it to in the most reliable and cost-efficient manner possible.

And our genuinely friendly staff programmers and systems analysts are available to provide you with whatever additional support you may need, from software consultation and customization to integrated systems design and specification.

So whether you need a quick question answered or a new service agreement, Business Applications Update or just some networking cable, call Micro Resources.

We'll help your Fortune Grow.

Micro Resources

535 Metro Place South Dublin, Ohio 43017 (614) 766-2335

Multiplan , continued from page 1

important cash control documents...When your bank statement is received, simply "tick" off the checks and deposits listed on the statement by typing the number 1 in the appropriate column. These number 1 "check marks" will automatically trigger the calculations for the bank reconciliation and the transactions summary."

We will now walk through the building of this worksheet. All the commands that you should type while inside of Multiplan will be in **bold** type. Commas between commands should not be typed - unless they are in bold type! Also, the word **Tab** means to use the TAB key. Just follow along and by the end of this article you will have your own check ledger and reconciliation system.

Adjusting Column Widths

We'll start by instructing Multiplan to widen and shorten the columns we'll be using so that the appearance of this worksheet will be pleasing. In order to increase the width of Column 1, put your cursor in column 1 and type **F**, **W**, **11**, **EXECUTE**. Now use the same procedure to alter the widths of the following columns:

Column:	2	3	4	5	6	7	8	9	10	11
Width:	7	21	7	11	4	11	4	14	12	12

Entering Titles

There are quite a few titles in this worksheet but they are required to make the check ledger understandable. Let's begin in **R1C5** by typing **A**, **CHECK LEDGER AND RECONCILIATION**. Use the following table to put the rest of the titles into their correct position:

R3C1 CURRENT PERIOD TRANSACTIONS
R5C2 Check
R5C4 Account
R5C5 Check
R5C7 Deposit
R5C9 Running
R5C10 ***** Outstanding *****
R6C1 Date
R6C2 Number Paid To
R6C4 Code
R6C5 Amount
R6C6 C
R6C7 Amount
R6C8 D
R6C9 Balance
R6C10 Check
R6C11 Deposit
R8C3 Balance Forward
R24C3 Ending Balance
R26C1 PREVIOUS PERIOD OUTSTANDING
R35C1 RECONCILIATION
R35C9 CURRENT PERIOD SUMMARY
R37C1 Checkbook Balance
R37C9 Opening Balance
R38C1 Less Bank/Other Charges
R38C9 Total Deposits
R39C1 Plus Bank/Other Credits
R39C9 Total Checks
R40C9 Ending Balance
R41C1 Adjusted Checkbook Balance
R41C9 Number of Deposits
R42C9 Number of Checks
R43C1 Bank Statement Balance
R43C9 Outstanding Deposits
R44C1 Plus Deposits in Transit
R44C9 Outstanding Checks
R45C1 Less Outstanding Checks
R46C9 PREVIOUS PERIOD SUMMARY
R47C1 Adjusted Bank Statement Balance
R48C9 Outstanding Deposits
R49C2 Difference
R49C9 Outstanding Checks

Entering Dashed Lines

Single- and double-dashed lines will be used to separate portions of the worksheet. To put a long double-dashed line in row two we will use Multiplan's **REPT** function to repeat an equal sign 21 times (which is enough to fill up any cell in this particular worksheet) and then we will copy it across the worksheet. Go to cell **R2C1** and type: **V**, **REPT("=",21)**, **EXECUTE**. Now to copy, type: **C**, **R**, **10**, **EXECUTE**.

We use this type of double dashed line in two other places so we'll copy it to these places by typing: **C**, **F**, **:11**, **Tab**, **R25C1,R34C1**, **EXECUTE**. Now for some

Continued on next page

BACKUP AND RESTORE UTILITY

PROTECTION MADE EASY!

- o Back up all or selected user directories
- o Restore all or selected user directories or files
- o Format diskettes
- o Review backup volumes
- o Menu-driven, easily installed
- o Complete documentation

Also copies the files necessary to preserve your system configuration--a tremendous help if your system ever needs to be cold booted!

$150.00 Backup/Restore Utility
$200.00 Utility + 2 boxes 5 1/4" 96 TPI high-quality diskettes

For more information, call CTD at (313) 540-1282.

To order, make check payable to *Computer Trade Development Corporation*. Send to:

BACKUP/RESTORE UTILITY
Computer Trade Development Corporation
Suite 1386
Birmingham, MI 48010

CTD

Company:_____
Contact:_____
Address:_____
City:_____
State:_____ **Zip:**_____

single-dashed lines in rows 4, 7 and 27. Go to **R4C1** and type **V, REPT("-",21), EXECUTE**. Now copy it by typing: **C, R, 10, EXECUTE**. Now to copy this long line to rows 7 and 27 by typing: **C, F, :11, Tab, R7C1,R27C1, EXECUTE**. We have some shorter single- and double-dashed lines in various other parts of the worksheet. Instead of just typing them in, we will copy portions of our long lines. We need a five column piece of a single-dashed line to start in **R36C1**. To do this first put your cursor in cell **R4C1** and then type: **C, F, :5, Tab, R36C1, EXECUTE**. Now put your cursor on **R2C1** and repeat this procedure to copy a 5-column piece of the double-dashed line from row 2 to row 50. Type **C, F, :5, Tab, R50C1, EXECUTE**.

For some other pieces, put your cursor on **R4C1** and type: **C, F, :3, Tab, R36C9,R47C9, EXECUTE**. Now again, put your cursor on **R2C1** and type: **C, F, :3, Tab, R45C9,R50C9, EXECUTE**. Now for some short single dashed lines in **R40C5**, put your cursor on **R40C5** and type **V, REPT("-",11), EXECUTE**. To put a short double-dashed line in **R42C5** put your cursor in that cell and type **V, REPT("=",11), EXECUTE**. Now copy the short dashed lines to **R46C5** and **R48C87** by putting your cursor on **R40C5** and typing: **C, F, ,R42C5, Tab, R46C5,R48C5, EXECUTE**.

For the last line, move your cursor to **R8C4** and type **V, REPT("-",11), EXECUTE**. Then type **C, R, 3**. Finally, complete the Balance Forward arrow by moving the cursor to **R8C8** and typing **A, --->, EXECUTE**.

Formatting the Titles

Many of the titles that we typed in are too long to be completely displayed in one cell. Therefore, we will use Multiplan's Format command to make the contents of this cell continue over to other cells. To make this go quicker we will handle some of the titles in groups. So let's go to cell **R1C5** and type **F, C, :9,R3C1:3,R5C10:11, Tab, Tab, C, EXECUTE**. Again, go to **R26C1** and type **F, C, :3,R35:47C1:3, Tab, Tab, C, EXECUTE**. And again, go to **R35C9** and type **F, C, :R49C10, Tab, Tab, C, EXECUTE**.

Now center the titles in rows 5 and 6, columns 4 through 11. These titles are not continuous so they can be centered. Titles in continuous format cannot be centered automatically. To center the noncontinuous titles go to **R5C4** and type **F, C, :9,R6C4:11, Tab, C, EXECUTE**. Our last title formatting procedure will be right-justify the titles in **R8C3, R24C3**, and **R1C11**. Go to **R8C3** and type **F, C, ,R24C3,R1C11, Tab, R, EXECUTE**.

Formatting Number Cells

The first thing we will do is center column 4 that will contain the account code. Go to **R9C4** and type **F, C, :R23C4, Tab, C, EXECUTE**. Now we want to left-justify the cells in which we will put the check numbers. Go to **R9C2** and type **F, C, :R24C2,R28:33C2, Tab, L, EXECUTE**. Many of the cells in this worksheet will hold dollars and cents values. Let's format these cells by going to **R9C5** and typing **F, C, :R23C5,R9:23C7, Tab, Tab, $, EXECUTE**. These are not the only cells we want to format in this manner so let's complete the job by using the **COPY** command. Leave your cursor on **R9C5** and type **C, F, Tab, R8:24C9,R28:33C5,R28:33C7,R37:49C5 ,R37:40C11,R43:44C11,R48:49C11, EXECUTE**. (That is a lot of cells, but if you are careful in your typing you will format the appropriate cells.)

Finally, let's make the cells that hold the outstanding checks and deposits have two decimal places. Move your cursor to **R9C10** and type **F, C, :R33C11, Tab, Tab, F, Tab, 2, EXECUTE**. We want Multiplan to insert commas automatically. To do this move your cursor to **R2C1** and type **F, O, Y, EXECUTE**.

Naming Cells

Here is an interesting trick that Ruth Witkin has. She suggests that we name all the cells whose contents will be erased and replaced with new entries each month. In this way, we can erase the appropriate portions of an old worksheet with one command. To do this put the cursor on **R9C10** and type **N, CURRENT, Tab, R8:23C1,R9C2,R9:23c3:8,R8C9,R38:39C5,R43C5, EXECUTE**.

Saving and Naming the Spreadsheet

Let's save the current version of the spreadsheet under the name check. To do this type **T, S, check, EXECUTE**.

Entering the Formulas

The real guts of a spreadsheet are its formulas. There are 19 formulas that drive our spreadsheet checkbook. Since there are so many, we will basically instruct you on how to enter them into the spreadsheet. A slightly more verbose description is given in Ruth Witkin's book. Before we start entering the formulas, be sure that you have turned of the automatic recalculation (**O, N, EXECUTE**).

Formula 1 - Check Numbers: The purpose of this formula is to give you sequential check numbers in column 2 based on the check number you enter in **R9C2**. Put your cursor on **R10C2** and type **1+R[-1]C, EXECUTE**. Now copy this formula through row 23 by typing **C, D, 13, EXECUTE**.

Formula 2 - Running Balance: Put your cursor on **R9C9** and type **V, R[-1]C+RC[-2]-RC[-4], EXECUTE**. Now copy this formula to the appropriate cells by typing **C, D, 15, EXECUTE**.

Formula 3 - Outstanding Check "Check Mark": This formula checks for the number 1 (one) in the column which indicates that a check is debited to your bank account. Note the use of the **IF** command which we spoke about in last month's issue. To enter this formula put your cursor on **R9C10** and type **V, IF(RC[-4]=1,0,RC[-5]), EXECUTE**.

Formula 4 - Outstanding Deposit "Check Mark": The same idea as the preceding formula. Put your cursor on **R9C11** and type **V, IF(RC[-3]=1,0,RC[-4]),**

Continued on next page

EXECUTE. Now copy formulas 3 and 4 to the appropriate cells by putting your cursor back on **R9C10** and typing **C, F, :11, Tab, R10:23C10,R28:33C10, EXECUTE.**

Formula 5 - Checkbook Balance: Put your cursor on **R37C5** and type **V, R[+3]C[+6], EXECUTE.**

Formula 6 - Adjusted Checkbook Balance: This balance formula takes into account the bank charges and bank credits to give you an adjusted balance. Put your cursor on **R41C5** and type **V, R[-4]C-R[-3]C+R[-2]C, EXECUTE.**

Formula 7 - Deposits in Transit: Move your cursor to **R44C5** and type **V, R[-1]C[+6]+R[+4]C[+6], EXECUTE.** The cell below this cell (**R45C5**) also needs a formula. Because of the way the spreadsheet has been designed it will use the exact same formula. So we will need to copy the formula in **R44C5** down into **R45C5**. Do this by leaving your cursor on **R44C5** and typing **C, D, 1, EXECUTE.**

Formula 8 - Adjusted Bank Statement Balance: Put your cursor on **R47C5** and type **V, R[-4]C+R[-3]C-R[-2]C, EXECUTE.**

Formula 9 - Difference: This formula will subtract the adjusted bank statement balance from the adjusted checkbook balance to give you the discrepancy, if any. Put your cursor on **R49C5** and type **V, R[-8]C-R[-2]C, EXECUTE.**

Formula 10 - Opening Balance: With your cursor on **R37C11**, type **V, R[-29]C[-2], EXECUTE.**

Formula 11 - Total Deposits: Move your cursor to **R38C11** and type **V, SUM(R[-29]C[-4]:R[-15]C[-4]), EXECUTE.**

Formula 12 - Total Checks: Put your cursor on **R39C11** and type **V, SUM(R[-30]C[-6]:R[-16]C[-6]), EXECUTE.**

Formula 13 - Ending Balance: Place your cursor on **R40C11** and type **V, R[-16]C[-2], EXECUTE.**

Formula 14 - Number of Deposits: Move your cursor to **R41C11** and type **V, COUNT(R[-32]C[-4]:R[-18]C[-4]), EXECUTE.**

Formula 15 - Number of Checks: Put your cursor on **R42C11** and type **V, COUNT(R[-33]C[-6]:R[-19]C[-6]), EXECUTE.**

Formula 16 - Outstanding Deposits: With your cursor on **R43C11**, type **V, SUM(R[-34]C:R[-20]C), EXECUTE.**

Formula 17 - Outstanding Checks: Place your cursor on **R44C11** and type **V, SUM(R[-35]C[-1]:R[-21]C[-1]), EXECUTE.**

Formula 18 - Outstanding Deposits: Move your cursor to **R48C11** and type **V, SUM(R[-20]C:R[-15]C), EXECUTE.**

Continued on next page

PSG

We are the *largest* Authorized Fortune Dealer in the Pennsylvania, New Jersey and Delaware area. As part of the **P**rofessional **S**olutions **G**roup of companies with over 35 years in business automation, we have the *stability* that our customers demand

Our services include a factory trained field engineering team, a complete parts inventory including system upgrade capability and a software support department.

For all of your Fortune needs contact:

Mark Shields of PSG at
2550 Eisenhower Avenue
PO Box 762
Valley Forge, PA 19482

or you can call Mark at (215) 666-1400

Formula 19 - Outstanding Checks: Put your cursor on **R49C11** and type **V**, **SUM(R[-21]C[-1]:R[-16]C[-1])**, **EXECUTE.**

Ok, that's it for the formulas. Now we can move onto using this spreadsheet. However, before we do that, we should lock the formulas and save the formulas. To lock the formulas leave your cursor on **R49C11** and type **L**, **F**, **Y**. To save the formulas, leave your cursor where it is and type **T**, **S**, **EXECUTE**. In a second you will see the question "*Overwrite existing file (Y/N)?*" Type a **Y**.

Entering Text and Numbers

Now it is time to put something in this spreadsheet so that we can see it work. If you haven't already done it, make sure the automatic recalculation is turned off by typing **O**, **N**, **EXECUTE**. We'll use Ruth Witkin's example and to do this you will need to type the text shown in **Figure 1** into your version of the spreadsheet. To start, go to R8C1 and type **A** for the **Alpha** command and type **Aug 1,84**. Then move your cursor to the next cell containing text or a number. Type the next entry and move your cursor to the next cell. Continue in this way until you have made your screen version of the spreadsheet correspond to **Figure 1**. Now for the moment of truth. Press the **F7** recalculate key and watch. In a few moments your screen should look like **Figure 2**.

To use this spreadsheet on a month by month basis you still need to know a couple of things. Let's suppose that **Figure 2** represents a real month's checkbook for you. You will, of course, want to print this out. If you don't know how to do this, we refer you to **Fortune System Corporation's Multiplan Guide** for the specifics of printing. To set up for the next month's use of the spreadsheet it will be necessary to erase certain portions of this spreadsheet. First blank out the entries with the "check mark" (rows 28 and 30). To do this put your cursor on **R28C1** and type **B**, **:R28C8,R30C1:6**, **EXECUTE**. Now refer to **Figure 2** (or your printout of the completed sheet) for the currently outstanding checks and deposits (those without the "check mark" in rows 12, 16, 20, 21, and 22). Type these entries into the previous period outstanding section.

Now blank out all the old information in one easy command. We can do that because, as you will recall, we named the cells we want to erase and we called them **CURRENT**. To do this leave your cursor where it is and type **B**, **CURRENT**, **EXECUTE**. Now recalculate (**F7**) and you are ready for another month of adding checks and reconciliation.

One last thing. We have left a limited number of rows for recording checks. You can add more rows if you like. To do that, follow these steps as specified by Ruth Witkin: **Step 1**: Create a blank master by using the Blank command to erase the cells you named **CURRENT**. **Step 2:** Place your cursor on row 23, type **IR**, and type the number of additional rows. **Step 3:** Copy the formulas in columns 2, 9 10, and 11 down into your new rows, and press **F7**. **Step 4:** Format the new rows for alignment and dollar and decimal coding. **Step 5:** Use the **Transfer Save** to store the changes on disk.

In the next installment we will build a cash disbursement spreadsheet which will be linked to this one.

Mark Palmerino

Figure 1

```
     1       2          3                4        5    6        7    8        9          10          11
1                                  CHECK LEDGER AND RECONCILIATION                                 CHECK
2 ================================================================================================================
3 CURRENT PERIOD TRANSACTIONS
4 ----------------------------------------------------------------------------------------------------------------
5          Check                        Account   Check        Deposit          Running    ***** Outstanding *****
6 Date     Number                        Code     Amount   C   Amount    D      Balance       Check       Deposit
7 ----------------------------------------------------------------------------------------------------------------
8 Aug 1,84            Balance Forward----------------------------------------->  $20,678.00
9 Aug 2       894     Regency Press          7      $230.00   1  $2,789.23   1
10 Aug 2              Hold Stationary        2      $161.19   1
11 Aug 7              Brown Realty           3      $557.00   1
12 Aug 11             Pelham U-Drive         8      $480.89       $743.00   1
13 Aug 14             Computer Tutor         6      $375.00   1   $567.45   1
14 Aug 14             Payroll                1    $5,321.67   1
15 Aug 20             VOID
16 Aug 21             Market Newsletter     12       $45.00
17 Aug 22             HJ-Exp Reimbursement   8      $220.00   1  $2,350.00   1
18 Aug 23             Eastern Power          5      $432.90   1
19 Aug 23             Petty Cash Fund       10       $50.00   1  $5,500.00   1
20 Aug 29             Payroll                1    $4,950.00       $1,122.33   1
21 Aug 30             Arco Plumbing          9      $532.56
22 Aug 30             Telephone Co.          5      $242.32   1  $1,089.32
23 Aug 30             Payroll Taxes          4    $3,081.17   1
24                          Ending Balance
25 ================================================================================================================
26 PREVIOUS PERIOD OUTSTANDING
27 ----------------------------------------------------------------------------------------------------------------
28 Jul 7,84   865     Parcel Express                $15.00   1    $432.75   1
29 Jul 10,84  867     License                       $75.00
30 Jul 28,84  888     Pro Advertising              $456.00   1
31
32
33
34 ================================================================================================================
35 RECONCILIATION                                                    CURRENT PERIOD SUMMARY
36 ------------------------------------------------                  ----------------------------------------
37 Checkbook Balance                                                 Opening Balance
38 Less Bank/Other Charges                        $8.00              Total Deposits
39 Plus Bank/Other Credits                        $0.00              Total Checks
40                                               ------              Ending Balance
41 Adjusted Checkbook Balance                                        Number of Deposits
42                                               ======              Number of Checks
43 Bank Statement Balance                     $23,145.76             Outstanding Deposits
44 Plus Deposits in Transit                                          Outstanding Checks
45 Less Outstanding Checks                                           ========================================
46                                               ------              PREVIOUS PERIOD SUMMARY
47 Adjusted Bank Statement Balance                                   ----------------------------------------
48                                               ======              Outstanding Deposits
49 Difference                                                        Outstanding Checks
50 ------------------------------------------------                  ========================================
```

Figure 2

	1	2	3	4	5	6	7	8	9	10	11
1					CHECK LEDGER AND RECONCILIATION						
2											
3	CURRENT PERIOD TRANSACTIONS										
4											
5		Check		Account	Check		Deposit		Running	***** Outstanding *****	
6	Date	Number		Code	Amount	C	Amount	D	Balance	Check	Deposit
7											
8	Aug 1,84		Balance Forward---------->						$20,678.00		
9	Aug 2	894	Regency Press	7	$230.00	1	$2,789.23	1	$23,237.23	0.00	0.00
10	Aug 2	895	Hold Stationary	2	$161.19	1			$23,076.04	0.00	0.00
11	Aug 7	896	Brown Realty	3	$557.00	1			$22,519.04	0.00	0.00
12	Aug 11	897	Pelham U-Drive	8	$480.89		$743.00	1	$22,781.15	480.89	0.00
13	Aug 14	898	Computer Tutor	6	$375.00	1	$567.45	1	$22,973.60	0.00	0.00
14	Aug 14	899	Payroll	1	$5,321.67	1			$17,651.93	0.00	0.00
15	Aug 20	900	VOID						$17,651.93	0.00	0.00
16	Aug 21	901	Market Newsletter	12	$45.00				$17,606.93	45.00	0.00
17	Aug 22	902	HJ-Exp Reimbursement	8	$220.00	1	$2,350.00	1	$19,736.93	0.00	0.00
18	Aug 23	903	Eastern Power	5	$432.90	1			$19,304.03	0.00	0.00
19	Aug 23	904	Petty Cash Fund	10	$50.00	1	$5,500.00	1	$24,754.03	0.00	0.00
20	Aug 29	905	Payroll	1	$4,950.00		$1,122.33	1	$20,926.36	4,950.00	0.00
21	Aug 30	906	Arco Plumbing	9	$532.56				$20,393.80	532.56	0.00
22	Aug 30	907	Telephone Co.	5	$242.32	1	$1,089.32		$21,240.80	0.00	1,089.32
23	Aug 30	908	Payroll Taxes	4	$3,081.17	1			$18,159.63	0.00	0.00
24			Ending Balance						$18,159.63		
25											
26	PREVIOUS PERIOD OUTSTANDING										
27											
28	Jul 7,84	865	Parcel Express		$15.00	1	$432.75	1		0.00	0.00
29	Jul 10,84	867	License		$75.00					75.00	0.00
30	Jul 28,84	888	Pro Advertising		$456.00	1				0.00	0.00
31										0.00	0.00
32										0.00	0.00
33										0.00	0.00
34											
35	RECONCILIATION								CURRENT PERIOD SUMMARY		
36											
37	Checkbook Balance				$18,159.63				Opening Balance		$20,678.00
38	Less Bank/Other Charges				$8.00				Total Deposits		$14,161.33
39	Plus Bank/Other Credits				$0.00				Total Checks		$16,679.70
40									Ending Balance		$18,159.63
41	Adjusted Checkbook Balance				$18,151.63				Number of Deposits		7
42									Number of Checks		14
43	Bank Statement Balance				$23,145.76				Outstanding Deposits		$1,089.32
44	Plus Deposits in Transit				$1,089.32				Outstanding Checks		$6,008.45
45	Less Outstanding Checks				$6,083.45						
46									PREVIOUS PERIOD SUMMARY		
47	Adjusted Bank Statement Balance				$18,151.63						
48									Outstanding Deposits		$0.00
49	Difference				$0.00				Outstanding Checks		$75.00
50											

Note: Last month we left an important part out of the Multiplan article. Before entering the formulas you should name certain cells. These cells are as follows: LOAN = R3C2; RATE = R4C2; TERM = R5C2; PAY = R6C2.

UNISOFT INC.

Experts in Computer Consulting - Sales - Service

Presents

FORTUNE SYSTEMS HELP

We Provide:

- Expert Trouble-shooting
- Years of programming and accounting experience
- *Free Initial Consultation*
- Training Available for:
  - * BASIC/IDOL/BAS and Multiplan
  - * UNIX and Ext. Fortune:Word
- Hardware Repairs & Upgrades
- FORTUNE trained experts at a cost that will surprise you!
- *Trade-Ins Welcome*
- Specializing in:
  - * Hard Disk Trade-In
  - * Hard Disk Upgrades

Authorized Fortune Master Dealer

Authorized Fortune Systems Repair Depot

CALL: (303) 671-8600 EMERGENCY HELP: (303) 691-HELP

UNISOFT INC. 14100 East Jewell Avenue Suite 27 Aurora, Colorado 80012

The UNIX Directory

INTRODUCTION TO SHELL SCRIPTS

David E. Kloes is President of UNI-KOMP which is located in Houston, Texas. He provides UNIX seminars, software for the Fortune Computer and is vice-president of the Houston UNIX User's Group. He contributes independently to ***/u/fortune*** *news.*

In Part 3, we discussed the "if then else elif fi" command sequence; the **test**, **read** and **exit** commands; and some new concepts such as control characters. Part 4 will continue the discussion of these commands and concepts.

Here is an instant replay of our **countup** shell script from Part 2:

```
for count in 1 2 3 4 5
do echo -n "$count "
   case $count in
     5) echo;;
   esac
done
```

When this script was run it produced the following results:

```
$ count
1 2 3 4 5
$
```

Remember we said that **echo** in the case portion of the script gave us a newline if the variable "count" was equal to 5. Just to show you an example of the versatility of the shell (and to give you an example of how the same job can be done in several different ways), we will produce the same result using the **if** statement:

```
for count in 1 2 3 4 5
do echo -n "$count "
   if test "$count" = "5"
      then echo
   fi
done
```

So far we have only talked about string comparisons. We will now discuss numerical comparisons and then show how the above script can produce the same result by doing a numerical comparison in the **if** statement.

USING THE *if* AND *test* COMMANDS FOR NUMERICAL COMPARISONS

When we discussed string comparisons, we used = and != to evaluate two strings. When comparing numbers, we use the following:

-eq	equal
-ne	not equal
-lt	less than
-gt	greater than
-ge	greater than or equal to
-le	less than or equal to

A common beginner's error in shell programming is to use these expressions for string comparisons or to use the =, != for numerical comparisons. Here is an example of the above script using the numerical comparison:

```
if test $count -eq 5
```

Normally we do not enclose the variable/values on either side of the **-eq** in quotes for numerical comparisons. Again, this is not a requirement; however, it will keep you out of trouble if you are programming on other versions of UNIX. The numerical expressions can also be used in conjunction with the **-a** (and) and **-o** (or) operators. For example:

```
if test $count -eq 5 -a $count -eq 4
  (equals 5 and equals 4)

if test $count -eq 5 -o $count -eq 4
  (equals 5 or equals 4)
```

The numerical and string comparisons may also be used with each other:

```
if test $count -eq 5 -o "$name" = "David"
```

When we are working with numbers in our shell scripts, we will often use the **expr** command for such things as incrementing a counter. Let's talk about this command and then rewrite our example above using the concept.

USING THE *expr* COMMAND

The **expr** command is commonly used in our shell scripts for computations involving numbers and variables. Let's say for example that you are at the shell prompt ($) and you want to know the result of the following computations:

a. 1+2 b. 3/2 c. 5x5 d. 6-4

To get the results of these computations using the **expr** command:

a. expr 1 + 1 **b.** expr 3 / 2 **c.** expr 5 * 5 **d.** expr 6 - 4

When these commands are typed in at the Unix prompt, the result of the calculation is displayed.

```
$ expr 1 + 1
2
```

This also holds true for using variables and numbers:

```
$ x=1
$ expr $x + 1
2
```

NOTE THAT THE SPACES ON EITHER SIDE OF THE SIGN MUST BE PRESENT. If you do not have the spaces, the following result will occur:

```
$ expr 1+1
1+1
```

This is certainly not the same as 2. Remember that when setting up values for variables that **SPACES MUST NOT BE USED.**

```
$ x=1                          correct
$ x = 1
x: command not found           syntax error - incorrect
```

Continued on next page

With these rules in mind let's take a concept we taught you earlier and apply it to expressions. Remember that we said a variable could be set to the result of any UNIX command by putting it in backquotes (`). What would **x** be equal to in the following examples?

```
Example 1                    Example 2
x=`expr 1 + 5`               x=1
                             x=`expr $x +1`
```

In example 1, **x** would be equal to the result of the expression 1 + 5 which is 6. In example 2, **x** is initially set to the value of 1. It then would be equal to the result of the expression $x + 1 which is the same as 1 + 1 which is 2. This is in fact the way we set up a counter in a shell script. There is certainly a lot more to the **expr** command; however we will limit the discussion here to the four simple operations of addition (+), subtraction (-), division (/) and multiplication (*).

Let's put all this together now and rewrite our count 1 thru 5 shell script using **expr** and the **if** statement:

```
x=0
while test $x -lt 6
do x=`expr $x + 1`
   echo -n "$x "
   if test $x -eq 5
      then echo
           exit
   fi
done
```

Nothing is ever simple. If it looks like we are about to teach you a new concept - you are right. We have introduced the **while do done** loop in this example.

USING THE *while do done* LOOP

```
while <condition> do <list> done
```

This command sequence basically says that while this "condition" exists, continue to "do" this list of things until "done" (until the condition no longer exists). The **while do done** loop can be very useful in shell programming and we will see some other uses as we go along. For now, let's look at it in the context of what we have done in the above count script.

`x=0`	this initializes x to 0
while test $x -lt 6	this says that while the value of x isless than six, "do" the things between the do and the done statement
`x=`expr $x + 1``	increments x by 1. Value is 1 the first time through the loop, 2 the second time time through the loop, etc.
`echo -n "$x "`	echoes the value of x followed by a space with no newline
`if test $x -eq 5` then echo exit `fi`	if the value is 5, does line feed and exits.
`done`	associated with the "while do". The loop is closed or finished when the test condition in the "while" line is met (when x equals 6).

The **while do done** sequence is called a "loop" because the statements between the "do" and "done" will continue to be executed over and over again until the "while" condition is satisfied (or we force the exit). Notice also that the **test** command is used in the **while** loop that same as it is used in the **if** and all the same rules apply.

Part of our task is to teach the shell syntax and commands. The other part is to teach concepts and to show you the versatility of shell programming. In the above example, the **if** logic is not really needed and we could rewrite the script as follows:

```
x=1
while test $x -lt 6
do echo -n "$x "
   x=`expr $x + 1`
done
echo
```

In the first example (using the **if** statement) we exited from the normal flow of the **while** loop by putting **exit** in the **if** statement. The above example is the more efficient way of writing the script and demonstrates the real flow of the **while** loop. Remember we said that the statements between the **do** and the **done** would continue to be executed until the test condition was met. **WHEN THE CONDITION IS MET IN A "while" STATEMENT, CONTROL OF THE PROGRAM NORMALLY FLOWS TO THE STATEMENT AFTER THE "done".** In this example, the only way that the last **echo** statement would be executed would be if the condition was met (i.e. x is equal to 5). Since this is exactly when we want this statement to be executed, we let the program flow normally. We are not trying to confuse you or make the problem more complex than it is. The point is there are many ways of writing the script to accomplish the same end. If you don't understand the mechanics of doing it a particular way, do it in a way that is easy for you to understand. The more important issue here is that you understand that a **while** loop will continue to execute until the condition is met and then program control goes to the statement after the **done**. One final thought - for those of you who think the solution would be to use our original script of "for count in 1 2 3 4 5" - what would you do if you had *1000* numbers?

We will talk about the **while** loop again later on. Let's end this session by completing discussion of the **if** statement. The only part of this command we have not demonstrated yet is the **elif** and **else**. Let's give an example:

```
if test $num -eq 1
   then echo "The number is 1"
elif test $num -eq 2
     then echo "The number is 2"
elif test $num -eq 3
     then echo "The number is 3"
else echo "The number is greater than 3"
fi
```

This basically says that if the number (where the number is the value of the variable "num") is 1, print "The number is 1", otherwise if the number is 2, print "The number is 2", otherwise if the number is 3, print "The number is 3", otherwise for all other numbers, print "The number is greater than 3". This could be used for example, if we had a menu and wanted to take different actions for each item on the menu.

This completes our discussion of the **if**, **test**, **expr** and **while** statements. Next time we'll talk about some new commands and concepts.□

SOFTWARE FOR THE FORTUNE 32:16

MULTIPLAN LINK - can pass data from BAS/IDOL/BASIC files to Multiplan and back. For example, it could be used to send data from any of the BAS files (Chart of Accounts, Customer Master, Inventory Master, etc). Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

RECORDS PROCESSING LINK - can pass data from BAS/IDOL/BASIC files to Records Processing and back. For example, it could be used to take customer address data from the Customer Master and will automatically make the Records Processing List document for mailouts. Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

TERMINAL/PRINTER LINK - provides capability to print Multiplan, and BAS/IDOL reports on a printer connected to a workstation, view any report on the terminal before printing, convert any report to Fortune:Word format; accumulate multiple BAS/IDOL reports before printing, backup reports onto diskette; copy or rename report files; print multiple copies of the report on either the primary and/or terminal printer. Also has option to print BAS/IDOL reports on HP LASER printer ($50 extra) **$195.00**

TELEPHONE LINK - on line telephone message, telephone directory and inter-office mailbox/memo system which can also be used as a simple calendar reminder system. Provides control, printing and viewing of on-line data for each user. **$295.00**

FINANCIAL LINK - stores BAS Income Statement and Balance Sheet data in files. Once stored, it can be sent to Multiplan or instantly printed. **$195.00**

MENU LINK - provides software and documentation for adding items to the Global Menu as well as a sub-menu for your use Includes backup script for user directories and BAS/IDOL/BASIC programs and data. **$95.00**

CALCULATION LINK - provides programs for amortization, depreciation, loan repayment, averaging (with graph), linear correlation, and breakeven analysis. **$95.00**

KOMPACT PERSONNEL ACCOUNTING - provides for data capture of personal, wage/salary, job, education, salary/wage, appraisal training, dental, medical and life insurance data. Includes over 80 pre-defined reports/worksheets. **$495.00**

CRT LINK - provides information for using other terminals on a Fortune System (WYSE, TV, IBM 3101). **$95.00**

AUTOMATIC REPORT LINK - provides capability to print numerous IDOL defined reports automatically based on frequency or application without having to select each from a different menu. You identify frequency/application codes. **$195.00**

TAPE LINK - provides one menu for all tape backup utilities. Provides daily, weekly, monthly and individual backup and restoration routines. Includes rewind, retension, and tape unit startup/stop utilities. User friendly. **$95.00**

PRINTER LINK - provides printcaps for Okidata 84, 2410, TI 855, and Anadex 6500. Includes shell scripts for changing pitch and LPI. Allows these printers to be used in word processing. **$95.00**

MODEM LINK - allows Hayes 1200 Smartmodem or compatible to be used on one port for both incoming and outgoing calls. **$95.00**

INSTALL LINK - provides documentation and software example for installing your product/menu through "S5". **$95.00**

REPORT LINK - modifies IDOL Report Generator Program to allow stacking of up to 8 fields in an IDOL defined report. Also corrects problem with displaying subtotals for all stacked fields. **$95.00**

WESTERN LINK - provides instructions on how to use your Fortune for sending/receiving telex messages using Western Union's Easylink. Can replace your telex! Fortune ITE software is required. **$95.00**

BACKUP LINK - provides menu selection for backing up /b (BAS/IDOL/BASIC), /u (Fortune:Word, Multiplan) and entire system onto diskette. Also has program to "unhang" a terminal that's hungup just by entering the terminal #; a program to restore one file from a multi-volume backup; and a selection to tell you which volume a backed up file is on. **$95.00**
to be sent and provides logical retrieval and key range selection. **$295.00**

UNI-KOMP 9803 Rocktree Houston, Texas 77040 (713) 937-7782

/u/help

***Dave Vantine** of **Vantine Computer Services** recently provided us with this puzzler. A customer upgraded to a 70 megabyte disk. Although the entire old disk was backed up on floppies, it was determined that the most expedient way to reconstruct the disk files was to attach both disk drives to the system and do a direct copy from one to the other. This all worked fine. However, when anyone other than the super user tried to log in to the system, the error message "no shell" was flashed briefly and then the login screen reappeared.*

Solution: Anytime the super user can do things that others can't, it implies a problem with permissions. The first thing to check was ownership of the home directories of all accounts -- all **/u** directories should be owned by their owners so that they can write in them. Next, the permission of **/bin/sh**, the shell program was examined to be sure it was executable by everyone. It was. Finally, the permissions on the root directory (/) were checked. This is done using the `ls -sail` command. The top two listings are generally . and .. as shown below:

```
# ls -sail
total 273
    2    3 drwxrwxr-x41 root        2240 Nov 18 07:04 .
    2    3 drwxrwxr-x41 root        2240 Nov 18 07:04 ..
 3994    1 -rw-rw-r-- 1 root         424 Sep 30 09:00 .cshrc
 1967    1 -rw-rw-r-- 1 root          90 Dec  4  1985 .mprc
 2502    1 -rw-rw-rw- 1 root          48 Sep 22  1985 .notes
```

The **dot** (.) directory is the current working directory, and the **dot dot** (..) is the parent directory. Since the **root** directory has no parent (it's at the top of the tree -- or bottom if you prefer) the . and .. directories are the same in this case. Note that these directories have three x's in the permissions column which indicates they are executable by everyone. This is necessary for anyone other than the super user to use this system. When these permissions were viewed on the offending system they were as follows:

```
# ls -sail
total 273
    2    3 drw-rw-r--41 root        2240 Nov 18 07:04 .
    2    3 drw-rw-r--41 root        2240 Nov 18 07:04 ..
```

By changing these with the `chmod 775 /.` command, the correct permissions were set, and the problem was cured. Apparently this problem can also occur when you do a complete restore from a streaming tape.

***Question**: I obtained **kermit** from your software library. When I try to use it, my terminal hangs and the only way I have to fix the problem is to shut down the system and restart it. What's the problem?*

Answer: **Kermit** is a communications program that allows your computer to talk to the outside world, usually via a modem. When you run **kermit**, you essentially redivert the output of your keyboard from the monitor to the port you have your modem attached to. If the modem isn't working properly, or if you can't connect to another system, then your terminal will appear to be frozen because nothing is responding to your gentle touch on the keyboard. In order to "unfreeze" your terminal, all you have to do is to disconnect **kermit** from the port. This is done by using the escape character -- a special character that signals **kermit** that something special is about to happen. In kermit, the escape character is a caret ("^"), which can be obtained by hitting a `SHIFT 6`. With **ckermit**, the escape character is `CTRL-\`, or a control backslash. Immediately after you hit the escape character, type a lower case "`c`". This will bring you back to the **kermit** prompt, at which point you can use `CANCEL/DEL` with **kermit**, or type quit with **ckermit** to exit **kermit**.

For more information on how to use **kermit**, read the files called **README**, **kmenu.doc**, or **kermit.doc** on your floppy disk. (Type `more /f/kmenu.doc` or `lpr /f/kmenu.doc` to view the files.) There is also a summary of some **kermit** basics in **Volume 1 Number 7** of **/u/fortune** *news*.

Another reason you may be having a problem is because the dip switches on your modem are incorrectly set. If you have a **Hayes** modem, switches 3 and 8 should be down, and all the rest are up. Be sure to note the current position of your switches before you change them in case it is necessary to go back to the way you were.

***Question**: Is it possible to have the UNIX prompt display the current working directory? I have friends who have IBM computers and they say this is no problem.*

Answer: You may have hit upon the one area of superiority of DOS over UNIX. As far as we know, it is *not* possible to have the **Bourne shell** (which is the standard shell) display your working directory. If anyone has discovered a way to do this, please let us know.

However, the **C shell** can be made to display the directory. The **C shell**, or **csh**, is not included with the **For:Pro** operating system -- it is part of **Fortune's Development Utilities**. There are long standing arguments over which is the better shell, although it is generally conceded that each have their strengths and weaknesses. We almost always use the **C shell** because of several of its strengths, which include: Command history, which allows the repetition of previously used commands with a few keystrokes; Command substitution, which allows you to edit and rerun previously run commands; and the ability to display the working directory.

We won't go into detail here about all of the ways to use the **C shell**. If you own the **Development Utilities** and want to try it out, just type `exec csh` from your regular prompt and the **C shell** will be invoked. In order to really take advantage of it, you will want to create the `.cshrc` file discussed below.

See **/u/help**, continued on page 16

/u/help , continued from page 15

In order to alter your prompt with the **C shell** enter the following lines into your `.cshrc` file: (If you don't have a `.cshrc` file in your home directory, just use an editor such as **screen** or **vi** to create one and include the lines below.)

```
set history=20
alias h history
set prompt = "\! <`pwd`> "
alias cd 'cd \!*; set WDIR = `pwd`;set prompt = "\!
<`basename $WDIR`> "'
```

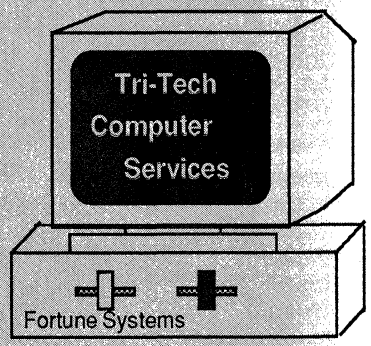
(The last two lines should actually be typed on a single line on your terminal -- our columns are too narrow for it to fit.) The first two lines are a bonus -- they activate the **history** command which lets you repeat commands. (Type `h` for a list of your last 20 commands. Repeat any of them by typing `!#` or `!letter(s)`, where **#** is the number of the command shown in the list, or where **letter** is the first letter or two of the command you want to repeat.)

The line which begins set prompt initially sets your prompt. The \! combination prints the consecutive command number of your command. This is followed by a space and a < symbol. The `` `pwd` `` section tells the shell to run the `pwd` command (print working directory) and put the results into the text for the prompt. Lastly, the directory is contained by a > symbol and a space. All of these extra symbols and spaces and command numbers are arbitrary -- they are included for aesthetic reasons. The line which begins `alias` in essence creates a little shell routine every time you type `cd`. It replaces the **cd** command with a command which changes the directory, sets the value of `WDIR` to the output of the **pwd** command, and finally sets the prompt equal to the directory, in a way very similar to the set prompt command on the previous line. The only extra function is the **basename** command, which truncates the complete path of the directory to just its right-most component, e.g. `/u/josh/newsletter` becomes `newsletter`. This makes the screen less cluttered.

In a future issue of **/u/fortune** *news* we will show you the basics of the C shell. Like the Bourne shell, it is basically a command interpreter. As we said earlier, however, it has some nice features that can make certain tasks easier.

Josh Lobel

A unique and personal company has been developed to better service the Fortune end-users of New England. **Tri-Tech**'s professional technical support team of **Jim Monterosso**, **Norman Westland**, **Victoria Cole** and **Allison Black**, offers you a combined total of 12 years experience exclusively with the **Fortune 32:16** Computer.



We at **Tri-Tech** are aware of the end-users concern to have a responsible and professional computer service company readily available for emergencies, questions or just friendly reassurance. It is this area that **Tri-Tech** has dedicated it's energies to provide the best possible technical support team in New England.

We provide local on-sight service as well as nationwide depot service by *Fortune Trained, Fortune Authorized* personnel.

In addition to technical support, **Tri-Tech** offers a variety of customer services.

New System Sales	Laser Printers	Consulting
Upgrades	Customized Cabling	Training
Supplies	Modifications	Software

CALL **Tri-Tech Computer Services, Inc.** TODAY!!

In Massachusetts Call (617) 965-4453 / In Connecticut Call (203) 828-9784

Basic Advisor, continued from page 3

```
PROCESS RECORD
DIM K0$(40,$FF$)
K0$=KEY(X,ERR=NEXT LINE)
CHECK FOR CONTROL BREAK BRANCHING
```

It would be very unusual to have a record key with Hex FF, and they sort high in case a relation other than ***NOT EQUAL*** *is needed. The* ***DIM*** *is used to put a known value not contained in the file into* `K0$`*; otherwise,* `K0$` *would contain the last correctly read value.*

Answer: Keep those cards and letters coming! We can use all the help we can get.

I have one comment on your **BASIC** bomb. I try to avoid **SHELL** in a saved program no matter which version of **UNIX** is present. Although I realize this is not always possible, I have found that bugs and bombs are frequently traced to a **SHELL** command. And if you wish to put your programs onto another **UNIX** system, you will probably have to rewrite the **SHELL** lines anyway.

ATTENTION SUPPORT COMPANIES:

I would like to start the new year with a fresh list of IDOL/BAS support companies for our readers. Since we are reciving support from neither **Fortune** nor **Concept Omega**, there is a need for your input. Send your company name, phone number, city of origin, contact person and method of billing. Please call me at (301) 448-9460. □

IDOL II

by ***David E. Kloes***

In September, **Concept Omega** held its annual Dealer Conference in San Diego. Concept Omega markets the **Thoroughbred** products better known to us as **SMC Basic**, **IDOL** and **BAS**. One of the announcements made by Concept Omega at the conference was the availability of **IDOL II** in December, 1986. This article will highlight the major features and changes in IDOL II.

FORMATS - Formats in IDOL II will specify the organization of records in a file in the same way that IDOL uses a Record Layout. The major differences between Formats and Record Layouts are that File Definition and File Maintenance specifications are not included in the Format but are handled in separate IDOL II components, allowing greater versatility in file maintenance and handling. In addition, the data element specifications available in IDOL II are extended beyond those allowed in IDOL, including the definition of special Text Fields that allow unlimited text entry to be associated with a record. The following changes will be made regarding data element definition:

-- Element length and precision indicator have been combined.
-- Key Indicator is similar but no secondary key specification.
-- Padding Indicator has been expanded.
-- Value Test, Valid Values have been combined and expanded.
-- HELP function has been expanded.
-- Preset Indicator and Value have been combined and expanded.
-- Delete Indicator and Value have been combined and expanded.
-- Special Edit functions have been expanded to include Pre and Post Input Processing
-- Audit has been expanded.

SCREENS

Screens are not tied to Files but to Formats, which allows you to use the same screen in the maintenance of different files with identical layouts. In addition, you may create different screens to maintain a single file. Default screens that include all items in the record can be generated automatically by the system. Customized screens that include all or only selected items in the file can be created.

Creation of customized screens utilizes a word-processing type, full-screen editor. This allows you to freely key in text and use editing functions to freely position text and fields. HELP text displayed in Windows can be positioned in the screen. In addition to display of fields for file maintenance, there is also the option to construct display-only fields from data.

Data entry utilizes full editing features. Screens can

Continued on next page

DO YOU KNOW...

there is an organization that has a thorough knowledge of

FORTUNE BAS
MULTIPLAN IDOL

We have been providing widespread support to our Fortune clients for more than 2 years.

If you need:

- training
- support
- maximum computer performance
- peace of mind relating to processing
- customization
- application results
- minimum errors and confusion
- better ideas on how to increase the power of your Fortune

Call Steve Rosenfeld at (314) 837-7111

SUPERIOR COMPUTER SYSTEMS, INC.
Making computers work for you

917 N Hwy 67, Suite 105
P.O. Drawer T
Florissant, MO 63031

IDOL II, continued from page 17

be constructed to display reverse video, background, foreground, underscore, blink and other video characteristics. Text fields can be entered through Windows.

VIEWS - Database maintenance can be accomplished in a unique manner with a View. This is a way of looking at the file as if through a window (like spreadsheets), where you can see a number of records and fields all at once. This method of maintenance or inquiry allows you to get a better "feel" for your database. Views also have a number of features that allow you to expand upon their utility:

Views, like Screens can be used to maintain different files or alternate Views can be used to maintain a single file. Both system generated and customized Views are available. In customizing views you can omit and/or rearrange the order of display for fields. You can also change the field width for display, effectively creating small scrolling "windows".

A number of powerful commands are available in a View to allow you to Sort, Count, List, Change, Delete, Copy and Move records. Modification of data utilizes full editing features. There is complete movement within a view utilizing the cursor and tab keys, and horizontal scrolling when you got past the edge of the screen. Multiple keys can be selected for a particular View.

LINKS - In IDOL II, access to a file is controlled by the Link. The Link contains the name of the File that is to be accessed, the name of the Format that describes the record layout and other important file information, such as what Terminals or Operators are allowed access to the File and whether the File should be audited.

Major new features to IDOL II include unique views into your database, customized screens, and links.

The advantage of using a Link in this way is that access is better controlled and record definition is more coordinated. A Link also allows you to establish a relation between a number of maintenance Screens and Views and a single file or between a single Screen or View and a number of different but identically designed files. In addition, this also allows you to define a file with multiple or different record layouts within the file.

MENUS - Menus in IDOL II are more versatile than those in IDOL. This is accomplished by using two elements to create a menu. One is a list of all of the menu selections and appropriate processing for the selection. This is similar to what is available in IDOL except that it is considerably enhanced and much easier to enter and maintain. The other element in menu definition is an IDOL II Screen for creation of the graphic display of the menu. By using the Screen for the graphic display, you get the advantage of using a full screen editor to create your menu display and also get to take advantage of video display features; reverse video, foreground, background, etc.

MESSAGES - Four message types will be available to avoid the need to place hardcoded messages in programs: Prompts and Constants; Yes/No Messages; Input Messages; and Non-Input Messages. Defining Messages is as easy as the other definition functions in IDOL II and utilizes all of the same techniques; function keys to initiate special procedures, input and text editing using special edit keys, HELP documentation available for assistance, etc.

ADDITIONAL NOTES - Some of the other changes include: field length increased to 20 characters; password protection down to field level; and automatic expansion of files when out of room. IDOL I and IDOL II will coexist on the same system. Concept Omega indicated no special price breaks for IDOL II even though you have IDOL I. All in all, it looks like an exciting product. □

Looking for Business Accounting Software that will conform to your Business?

Look to DSM

Applications
- Can be tailored to your organization's specific needs
- Economy of Standardization
- Follows Generally Accepted Accounting Principles (GAAP)
- Luxury of Adaptibility
- Data Base Incorporation

Software for:
- Not-for-Profits
- Construction
- General Business
- Property Management
- Steel Service Center
- Law Office Management

Hardware
Fortune Systems, NCR, AT & T, IBM

Call or Write
Data Systems Management, Inc.
277 Coon Rapids Blvd.
Minneapolis, MN 55433
(612) 780-5200

All software was developed using PROGRESS, a powerfull relational data base which allows you to tailor the software to your specific needs.

DSM PROGRESS

Dealer Inquiries Invited
PROGRESS is a trade mark of Data Language Corporation.

Dear Fortune Owner:

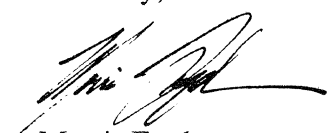
As you have probably read in */u/fortune news,* Fortune Systems SX models are significantly more powerful and faster than the PS and XP models. If you have a PS or XP model system, you should take advantage of our special promotion on SX upgrades. For a limited time, and while our supplies last, we will upgrade your system, including the new SX Cabinet at no additional charge for the cabinet. With this special offer, you may choose any combination of the following upgrades and/or expansions:

- *SX Processor with Current Release of Multi-User For:Pro*
- *60 MB Internal Streaming Tape*
- *30 MB, 45 MB, 70MB High Speed Hard Disk*
- *6-Port Intelligent Asynchronous Controller*
- *Upgrade to Current Release of all Fortune Software*
- *Fortune:Windows (Included at NO Charge)*

Additionally, we can perform these upgrades within 24 hours of receiving your system. In fact, if you ship your system overnight on a Friday for Saturday delivery, we will have it back to you on Monday morning, all ready to plug in and run; not losing the use of your system for a single work day. And, we will issue you a full 90-day warranty on the entire system.

JFM provides services to Fortune Systems owners nationwide either direct or through our authorized Affiliated Dealers. If you haven't used our services, I invite you to discover JFM; you will be glad you did.

Sincerely,



Morris Frydman
President

Talk to us, then talk to our clients

JFM
BUSINESS
SYSTEMS
The Merchandise Mart
Lobby Suite 144
Chicago, Illinois 60654
Call (312) 644-0870

Authorized Fortune Systems Master Dealer

ATTENTION FORTUNE SYSTEM OWNERS

Paying for a new power supply ?
Fixing or upgrading your present one could save you hundreds of $$$$.

Repairing (Zenith) $ 75.00
(Western Electric) $ 95.00
Upgrading of Zenith to 250 watts $ 300.00

Terminal supplies repaired also.

Repaired supplies warranted for 90 days.
Upgraded supplies warranted for 1 year.

Quantity discounts available. (Call for quote.)

POWER SUPPLIES "R" US
5916 Southwind Dr.
San Jose, CA., 95138
408-365-7678

FACTS: Your current software is

tired
inflexible
out of date
hard to modify

FACTS: You long for software that is

FLEXIBLE! (many user selected options)
MORE COMPREHENSIVE DATA!
UP TO THE MINUTE!
COMPLETE WITH LITTLE OR *NO* MODIFICATION!

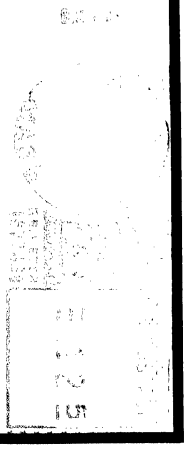
Let us send you more information on the software that has the SOPHISTICATION you need and want at a cost that is SURPRISING! Yes, it's worth the investment to change software rather than do without what you need (what you thought was in your current software, BUT IS NOT), or to pay a great deal of money for modifications.

FACTS:
Order Processing
Inventory
Accounts Receivable
Accounts Payable
Sales Analysis
General Ledger
Payroll
Menu and Security System
Job Streaming

FACTS: Software that's not like all the rest. It's better.

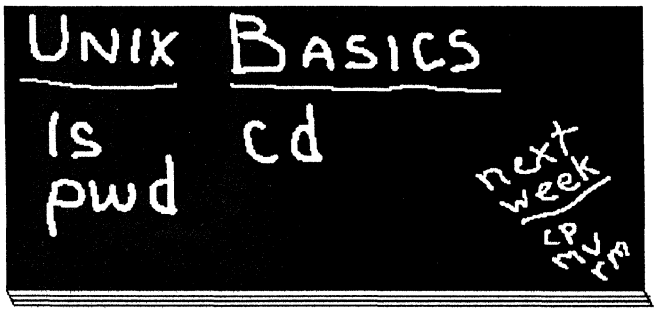
COMPUSYSTEMS
OF GEORGIA, INC.
2200 Northlake Pkwy
Tucker, Georgia 30084
(404) 491-8004

/u/fortune
73 Jason Street
Arlington, MA 02174



/u/fortune* news

The Newsletter for Users of the Fortune 32:16 Computer — December 1986/Volume 3 Number 12



Most of the readers of our newsletter fall into two basic groups with regard to knowledge of **UNIX**. First, there are those who know **UNIX** fairly well. They feel more comfortable working in one of the **UNIX** "shells" than working with and through the **Global Menu**. In fact, most of these folks (I guess I fall in this group) have a great, if not irrational, admiration of this particular operating system.

On the other hand, there are those who try as much as possible to keep the **UNIX** side of the Fortune computer at arm's length. Only with great difficulty and much trepidation will they dive down into the murky depths of this thing (beast?) called **UNIX**. And the **Global Menu**? Well, it is the "natural" way to work with a computer and its beauty borders on the crystal spheres of Aristotle.

The avowed purpose of this article is to introduce the second group of people to just **6** fundamental **UNIX** commands which will provide the foundation for "getting around" in the **UNIX** operating system. Of course, 6 commands is not the be all and end all of **UNIX**. Rather, it is something like learning **Fortune:Word**. If you know 3 or 4 (or maybe 6) basic **Fortune:Word** commands, you can begin to take advantage of one of the most powerful word processors that exists in the market. The same claim is being made for the commands that we introduce in this article - which if you will learn them and practice them, you will be able to use **UNIX** more effectively.

So, what are these 6 magic commands? First, they are definitely not magic. In fact, they are not even sexy - they are just useful. Listed, they are as follows:

ls	list files in your directory
cd	change directory
pwd	print working directory
cp	copy a file
rm	remove, or delete, a file
mv	move, or rename, a file

We will take each of these in turn and show you some of the bare bone basics of how to use them. Before we get into that, however, I want suggest that there are two other commands of immense usefulness. These two commands are **cat** and **more**. They are basically used to view the contents of some file on your computer screen. They should be added to the above list, but I have chosen not to discuss them simply because they were treated in some detail in Volume 3 Number 7 of /u/fortune news. If you want more information about them, please consult that article.

The UNIX shell is a command interpreter. For the record, the Global Menu is also a command interpreter.

The very first thing I want to do is review how to go from the **Global Menu** to the shell. At the **Global**

See **Unix Basics**, page 14

The BASIC Advisor

Ray Wannall is President of BaSiC Software Corporation in Baltimore and is contributing to this publication independently.

Question: *I enjoy your articles in the news and have used a number of the tips you have given.*

*I am not a programmer but have found many needs for special files and reports so I've gotten into **IDOL** as a means of "survival". I've been using it rather extensively for about a year and have found the key to many of the practical puzzles that are not covered by the manuals. However there are several things I*

See **BASIC Advisor**, page 8

In This Issue...

Index of /u/fortune news–Quick access to 1986...**Page 17**.

The UNIX Directory–Part 5 in our series on shell scripts...**Page 11**.

Multiplan–Dave Hinze shares some valuable insights...**Page 4**.

The BASIC Advisor–Having trouble with formfeeds?...**Page 1**.

Unix Basics–Six basic **UNIX** commands for everyone...**Page 1**.

/u/help–more on cp's flags, also laser printer help...**Page 7**.

News from Fortune...**Page 2**.

*/u/fortune *news* is published by The Cambridge Consortium, Inc. It is not affiliated with Fortune Systems Corporation.

/u/fortune *news...*

Editors
Josh Lobel
Mark Palmerino

Associate Editor
Benzion Chanowitz, Ph. D.

Technical Editor
Scott Bradner

Advertising Manager
Mark Palmerino

/u/fortune *news...* is published 12 times per year by The Cambridge Consortium, Inc. Neither /u/fortune news... nor The Cambridge Consortium are affiliated with Fortune Systems Corporation.

Subscriptions cost $48 for one year within the U.S.A. For foreign subscriptions, add $27 each year for added postage and handling costs.

Back issues are offered at:

Vol 1, Numbers 1 - 3	$1/each
Vol 1, Number 4	$.50/each
Vol 1, Numbers 5 - 8	$3/each
Vol 2, Numbers 1 - 7	$3/each
Vol 3, Numbers 1 - 12	$4/each

For Back Issues delivered to a foreign address, please add $15 to the order total for added postage and handling. Software diskettes cost $10 each. Please add $5 per order for foreign destinations.

Although every attempt is made to insure that our contents are error-free and without risk, The Cambridge Consortium is not responsible for losses that occur due to correct or incorrect interpretation of **/u/fortune** *news.*

For more information, call or write:

/u/fortune *news*
73 Jason Street
Arlington, MA 02174
(617) 663-7700, (617) 646-5603

The text for this newsletter was prepared using a Fortune 32:16 for text input. The information was then transferred to an Apple Macintosh Plus with kermit and formatted for typesetting using the Aldus Pagemaker program.

UNIX™ is a trademark of Bell Laboratories, Inc. Fortune 32:16™, Fortune:Word™ and FOR:PRO™ are trademarks of Fortune Systems Corporation. Multiplan™ and MS-DOS™ are trademarks of Microsoft. IBM-PC™ is a trademark of IBM. Macintosh Plus™ is a trademark of Apple Computer. Pagemaker™ is a trademark of Aldus Corporation.

©1986 by the Cambridge Consortium, Inc. 12/86

News From Fortune

Fortune:Word 3.0 and For:Pro 2.0 Released Next Year

In a recent issue, we printed the announcement of **Fortune:Word 3.0** and **FOR:PRO** 2.0. We should clarify that this does not mean that these products are actually ready to be shipped. Fortune Systems does not want to specify when these products will be available, but we expect that they will begin shipping early next year. We apologize for any confusion that resulted from our earlier article.

Various Products Released on Multiplier Hardware

(Editor's Note: *The Multiplier is the name Fortune Systems has given to its **IBM For:Pro** card. This is a piece of hardware which plugs into an **IBM-PC** or compatible and allows it to run full 1.8 **For:Pro** with up to 3 users. These notices refer to additional software that is now available for The Multiplier.)*

Fortune Systems announces the release of **Development Utilities** on Multiplier Hardware. **Development Utilities** features many utilities for the use of the developer and the everyday user. Some highlights from the **Development Utilities** include **sccs**, a program which is useful in tracking release and modifications to software. A combination of **cron**, **at** and **atrm** is useful for performing unattended processing; **dcheck**, **bcheck** and **icheck** are useful for disk management and diagnostics, and **awk** for report generation. The program **man** and man pages are not included in this release.

In a continuing effort to increase the functionality of the Multiplier and to fulfill Fortune's customers' needs for a multiuser spreadsheet, Fortune Systems announces the release of **Multiplan** on the Fortune Multiplier. **Multiplan**, like other Multiplier applications can be manipulated by three users simultaneously. Each user can share information or protect a file in the same manner as a file is protected on the 32:16.

With **Multiplan**, users can convert single user **Visicalc** files into multiuser **Multiplan** files. Spreadsheets can be converted into text and can be edited, revised or merged with other documents through **Fortune:Word.**

Fortune announces availability of the latest release of **Micro Focus LEVEL II COBOL** for the Fortune Multiplier. This release allows developers to create COBOL applications on the Fortune Multiplier at a level of implementation that was previously available on larger systems.

Principal features offered in the latest release of Micro Focus Level II COBOL-ET version 1.1.2 are:

GSA certified to the Federal High Level standard for ANSI 74 COBOL

Very fast compile speed

Very high speed program execution

Numeric and function key support for the Fortune keyboard

Cross reference listing

Enhanced interactive screen handling

Efficient B-tree ISAM

Execution of shell commands from a COBOL program

Supports large files to 1 Gigabyte

256MB data/800MB code space allowed

Optimize memory utilization

Enables easy upgrade from other selected COBOLs to Micro Focus Level II COBOL, including Ryan McFarland COBOL. □

ATTENTION LAW FIRMS

You made a smart decistion when you bought your Fortune computer for word processing and office automation.

Now take another giant step forward and install Gill & Piette's

TIME & FINANCIAL MANAGEMENT SYSTEM FOR LAWYERS (TFM)

on your existing Fortune system.

The product of over 25,000 hours of research, development and testing, TFM is a total law office accounting and management information system featuring:

- *autoflex billing in multiple, user-definable formats*
- *integrated client accounting and general ledger*
- *flexible management reporting for work in progress, write offs, write ups, time history, billing realization, etc.*
- *integrated modules for Accounts Payable, Docketing, Conflict checking, Trust accounting, Finance Charges*
- *multiple realization measure, including attorney billing realization and fee receipts by working attorney*

Support? We have a national software support and maintenance program that covers everywhere from Bermuda to Washington State. Nobody can support you installations as well as we will, and we've got the track record to prove it.

HOW MANY PROFESSIONAL HOURS CAN YOU AFFORD TO SPEND DESIGNING YOUR OWN SYSTEM OR TINKERING WITH SOMEBODY ELSE'S UNSUPPORTED SOFTWARE?

MAKE THE RIGHT BUSINESS DECISION WITHOUT FURTHER DELAY AND INSTALL TFM, THE FINEST LAW OFFICE MANAGEMENT SYSTEM AVAILABLE TODAY!

FOR MORE INFORMATION
CALL

GILL & PIETTE, INC

(202) 546-6170

Multiplan - A Reader Responds

In this month's Multiplan article we present some very helpful information contributed by David Hinze of ICT, Inc. in Shelby, MI. In the inset box below, Dave presents his version of a check register. In the article to the left he explains several useful and clever hints for using Multiplan to its fullest potential. Thanks, Dave. We hope more readers will contribute their ideas.

UNDERLINING COLUMNS

One simple way to underscore column headings is to use the **REPeaT** command. Combining this with the **LENgth** command produces a nice result.

For example,

```
REPT("-",LEN(R[-1]C))
```

will create a dashed line under whatever text is in R[-1]C. If the

Continued on next page

```
#1        1                    2                    3          4          5
  1 A BETTER CHECK REGISTER........ SEE INST. ON LINE 61
  2 FILE = BTR.CKBK  BY D. HINZE, ICT, INC. SHELBY, MI
  3 ================
  4                      TOTAL CHECKS------>      189.75
  5                      TOTAL DEPOSITS---->                  510.10
  6                                            CHECK      DEPOSIT
  7    DATE              PAYEE/DESC             AMOUNT      AMOUNT      BALANCE
  8 --------- ------------------------------ ----------- ----------- -----------
  9           BEGINNING BALANCE                          ENTER---->     2,000.00
 10 TODAY     A. T. & T.                         189.75                 1,810.25
 11           FOR OUT WATS LINE
 12 TOMORROW  DAILY RECEIPTS                                 510.10     2,320.35
 13           INCLUDES SERVICE CHGS OF 10.22
 14
 15           FORMULA IN COL 5 ENDS AT THIS ROW COPY DOWN AS NEEDED
 16
 17
 18
 19
 20
COMMAND: Alpha Blank Copy Delete Edit Format Goto Help Insert Lock Move
         Name Options Print Quit Sort Transfer Value Window Xternal
```

This worksheet for recording cash items is better than simple worksheets because it allows the user to skip a line or lines without changing the formulas after every blank line.

Blank lines are needed often for extra descriptions, and to continue to another worksheet page.

A simple formula for the balance column would be equal to: the balance from the line above **(R[-1]C)** + current line deposit **(RC[-1])** - current line check **(RC[-2])**. If there was no check, there would be no balance and therefore no beginning balance for the next line.

This worksheet names the very first balance **'BEGBAL'**, the first check and **'CK1'** and the first deposit as **'DEP1'**. The formula for each balance cell then looks at the current line to see if there was any activity not equal to zero. If there was, it takes the *FIRST* beginning balance (**BEGBAL**) plus *ALL* of the deposits from the very first through the current row, minus *ALL* of the checks from the very first through the current row.

Added bonus - no **#VALUE!** error if text is entered into either the check or deposit column.

To use this worksheet you need to name three cells as follows:

R9C5 BEGBAL
R9C3 CK1
R9C4 DEP1

Also, you need three formulas. Put the following formula in both R4C3 and R5C4:

```
SUM(R8:253 C)
```

A longer formula is found in R10C5 and this formula should be copied down to as many cells as you have checks:

```
IF(SUM(RC[-2]:RC[-1])<>0,-
BEGBAL+SUM(DEP1:RC[-1])-
SUM(CK1:RC[-2]),"  ")
```

word *Total* is in R[-1]C then the underline will be five dashes long (the number of characters in the word *Total*).

UNDERLINING SUB-TOTALS

After underscoring your column headings, name the area with the underlines **UL**. When you reach a spot in your worksheet to make a sub total, simply do a **Copy From UL**.

DOUBLE UNDERSCORE TOTALS

This is as simple as example 1. Now that you have named **UL** in example 2, to double underscore simply use the command:

```
REPT("=",LEN(UL))
```

AN EASIER WAY TO COPY RIGHT OR DOWN

The **COPY** command is nice, but on a long worksheet, after you set up formulas in 15 columns on line 1, you wish to copy them down, it is boring to do a **COPY DOWN** on each of the 15 columns. Try, however, doing a **COPY DOWN** and do a **RETURN** instead of **EXECUTE**. This will position your cursor at the **starting at:** space. Now press the colon : key and move the cursor to the last cell you to which you wish to copy down. An **EXECUTE** now will copy down all your cells at once. The same logic will work with **COPY RIGHT** as well.

COPY DOWN OR RIGHT BUT HOW MANY TIMES?

How many times do you have to use math to figure the number of cells to copy down or across. You are on row 17 and want to copy down to row 43. You can figure the number yourself or let **Multiplan** do it for you. Simply do a **COPY DOWN** and where it says number of cells enter **43-17**. **Multiplan** will do the subtraction for you and store the number for your next operation.

EASIER WAY TO VIEW A PORTION OF YOUR MULTIPLAN INDEX

Does your Multiplan index fill most of your screen and never have files printed in any kind of order? Do many of your worksheets begin with the same letter or letters? As an example, lets say you have a library that has many cluttered items and you have a file that begins with a **P** but you don't know the full name. Do a **Transfer Load** and enter a **P*** instead of the full name and then press an arrow key. Surprise! You have an index of only the files which start with a **P**. Know it starts with a **Pxt**? Use **Pxt***. Know it ends with a **W**? Try ***W** !

SWITCHING LIBRARIES WITH A TWIST

If you have several sublibraries or switch between libraries often, you can do switch easily without having to exit **Multiplan** to change libraries. Simply go to the **Transfer Options** selection and press return. This will place you in the field headed **setup**:. You may now enter the full pathname of any other library you wish. **EXECUTE** and you are in the new library. For ease in sublibraries, you can replace your home directory with a **.** (that's a period). For example, if you are logged in as dave and wish to change to sublibrary payroll. You can type **./payroll** instead of **/u/dave/payroll** This works because in **UNIX** your *current* directory is designated **..** In the above example if you had changed to the subdirectory payroll and then wished to change to another subdirectory called financial, you would have to type **../financial**. Using **..** moved you back up one directory to your home directory and then the **/financial** moved you to subdirectory **financiall**.

PRINT MULTIPLE COPIES OF YOUR WORKSHEET !!!

With the addition of a simple **UNIX** file you can print multiple copies of a **Multiplan** file with ease. I have named my file **printit**. (Editor's Note: **printit** is an example of a shellscript. For more information on how to make use shellscripts, we suggest you read the series in the **UNIX Directory** by Dave Kloes beginning with Volume 3 Number 8)

To use **printit** , first you need to set your print options and margins in **Multiplan** as normal. Then instead of printing it on the printer you print it to a file. Almost any name can be selected for the file as **Multiplan** will tell you if there is another file with that name.

Then when you are done with Multiplan, exit and go into the shell. Type:

```
printit fullfilename N
```

and **UNIX** will do the rest. In this shell script, **fullfilename** means the full directory path and name such as **/u/dave/payroll/W2**. **N** stands for the number of copies you wish to print. When it is done, it deletes the temporary file you just created.

The shell script is as follows:

```
cat $1 | lpr -h -n $2 -p 2
+P15
rm $1
```

(You must have a return after the second program line.)

The program uses the variable **$1** to capture the filename and **$2** to capture the number of copies. The character between the **$1** and **lpr** is the uppercase grey key located to the left of the tab key. The **-h** option deletes the printing of the banner. **-n** looks to **$2** for the number of copies. **-p 2** in our case uses printer 2. This can be deleted if you wish to use the regular line printer or substitute your logical printer number you wish to use. **+P15** changes the pitch (Characters Per Inch) from the printer default to 15. You may change yours to 12 or leave the **+P15** out if you want to print at 10. Another option may be to add a third variable **$3** to capture any desired pitch.

The second program line deletes the file after it has been printed. This could have been done in line 1 but is shown this way for ease of understanding.

I have placed this file in **/usr/bin** so all users can have access to it. **Root** will have to add the program to **/usr/bin**. Be sure to do a **chmod +x printit** to make it executable. Check for an existing file first.

I hope these are hopeful and I look forward to tips from other readers.□

Our Memory Doesn't Cost You A Fortune

Would you like to:

Speed up your computer?
Add more users?
Use the latest software, like Fortune:Windows™?

Compatibility:	The F1MB is absolutely 100% compatible with Fortune's own 1 Meg memory board.
Affordability:	USA is proud to have equaled and exceeded the quality and performance of our competition, while offering substantial savings for both the dealer and end user.
Dependability:	All boards are burned in for 72 hours and tested for 48 prior to shipment. Our One Year Repair or Replace Warranty underlines our confidency in the product.
Repairability:	The F1MB facilitates repair in the field as it is 100% RAM socketed. Now you can repair a board by simply replacing a defective chip rather than face an expensive board exchange -- reducing maintenance costs and system down time.

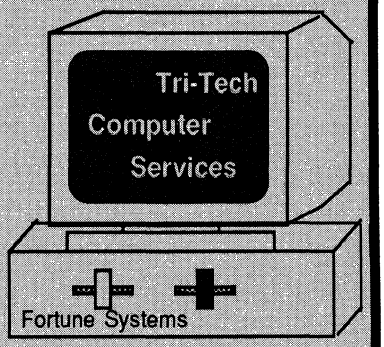
If you have a Fortune SX, XP45 or recent model XP30, you can use our board by simply plugging it in. Older systems can be upgraded to use our board with a simple and economical modification.

For Details Call: ***Upgrade Service Associates***

3 Marion Road, Burlington, MA 01803 Phone: (617) 239-8147

Our Guarantee: If you're not completely satisfied within 30 days, *USA* will gladly refund your money.

A unique and personal company has been developed to better service the Fortune end-users of New England. **Tri-Tech**'s professional technical support team of **Jim Monterosso**, **Norman Westland**, **Victoria Cole** and **Allison Black**, offers you a combined total of 12 years experience exclusively with the **Fortune 32:16** Computer.



We at **Tri-Tech** are aware of the end-users concern to have a responsible and professional computer service company readily available for emergencies, questions or just friendly reassurance. It is this area that **Tri-Tech** has dedicated it's energies to provide the best possible technical support team in New England.

We provide local on-sight service as well as nationwide depot service by *Fortune Trained, Fortune Authorized* personnel.

In addition to technical support, **Tri-Tech** offers a variety of customer services.

New System Sales	Laser Printers	Consulting
Upgrades	Customized Cabling	Training
Supplies	Modifications	Software

CALL **Tri-Tech Computer Services, Inc.** TODAY!!
In Massachusetts Call (617) 965-4453 / In Connecticut Call (203) 828-9784

/u/help

*We received this letter from **Jim Dyson** who is the **Technical Support Manager** for Fortune System International in Monaco. He responds to a question in Volume 3 Number 10 concerning some unexplained flags for **cp**, as well as some questions pertaining to **rdconf** and **mkconf**.*

I just received the October issue of **/u/fortune**, and I wanted to add my input on the questions from Michael Smith reported in **/u/help**, which Liz Muth has already responded to.

In addition to the **rdconf** program option **-U**, the **mkconf** program also has this same **-U** option for setting the number of 'swap units' in the configuration block. Caution is in order, though, because *changing the swap size on a working system will cause irreparable damage.*

The undefined cp options **C**, **E**, **K**, and **M** are outlined below:

C — This is a continuation of a previously aborted multi-volume backup (**-B** option). If a multi-volume backup is aborted, it can be restarted if at least the first volume was successful. This information was published in Software Bulletin Number 5, issued in April 1986.

E — This option is used with multi-volume restore (**-R** option) to exclude certain files from being restored. I have never seen it used, and I know that it has limitations. The syntax is to put the name of the file to be excluded directly after /dev/fd02 in the restore command line. I am not sure of its limitations, and would not recommend its use.

M — This option is exactly equivalent to setting the flags **rspT**. It was a developers shortcut from the early days of the company.

K — Never heard of this one, and don't think it's valid. It doesn't show up in the usages messages from cp, so I don't know how its existence (or non) came up.

Thanks, Jim, for that helpful information. We appreciate the time you spent helping us answer that question.

Michael Smith wrote us with the following advice:

Using a Laser Printer with Informix

In working with Informix with a laser printer (QMS Kiss), I found the line count in reports to be bothersome. Informix uses line feeds to go to the top of the next page, while the printer will not print a page until either the print area is filled or a form feed is received. This inconsistency meant that the laser printer would not print in a convenient and predictable manner. The following technique has proved to be a satisfactory solution to the problem.

In the **DEFINE** section:

```
Variable ff type character length
1
```

will establish a variable to be used in the page header section. The first pass will have a null value.

In the **OUTPUT** section:

```
Top margin 0
Bottom margin 0
Lines xx
```

where the **xx** indicates the actual number of lines you wish to print. (A normal page with an inch margin at the top and bottom would normally have 54 lines of text.) By setting the top and bottom margins to zero, the normal white space at the top and bottom of the page is eliminated in the **OUTPUT** definition.

In the **PAGE HEADER** section

```
Print ff
Print " "
Print " "
```

will print the value of **ff**. Then it will print two blank lines to space down into the image area of the printer.

Also, in the **PAGE HEADER** section after the above lines, include:

```
Let ff = ascii 12
```

This will make the variable **ff** equal to the form feed character that the printer needs to eject a page. Note that this happens after the first page is printed. If you put it above the Print **ff** command, a blank sheet of paper would be ejected before your document begins to print.

If the output is being sent to a system file that is to be printed with **lpr** then no additional code is required since **lpr** uses a form feed on the last page under the normal definition. If the defaults have been changed for lpr then code changes should be included to compensate for the alterations.

If the output is being redirected in the command line, as in:

```
acego -q pgm > /dev/tty01
```

then **ON LAST RECORD** should be included to

```
print ff
```

so that the final page is ejected. If you have a form feed at the end of the file and use **lpr**, then a blank page will be ejected. If the output is being printed with **pr**, then the command should be sent as follows:

```
pr -t -f filename > /dev/tty01
```

Continued on next page

The `-f` flag will cause **pr** to use form feeds in lieu of line feeds. The lines per page option may be used with some arbitrary large number to prevent pr from ejecting inappropriately. This would be necessary in situations where the laser printer is set to 8 lines per inch, or landscape orientation, for instance.

If any reader has a better way to address this problem, I am anxious to hear about it. (Editor's Note: please send your solutions to /u/fortune news, and we will share them.)

When using the **OUTPUT** section to pipe the report to **lpr**, I have not been successful in using options, such as `lpr -h`. Does anyone know a method to get the options to stick?

BASIC Advisor

Continued from Page 1

would like to do but have not been able to accomplish. I am hoping you can help.

*1. We use our Fortune system for order entry, multiplan, word processing, telex (**Colt** from **SST**) and telecommunications (**ITE** from **Unixworks**). Every program, except **IDOL** reports, gives a form feed at the end of its run on the printer. I have found that I can use "end of report" logic to give a form feed but this is not only a pain, you don't actually get to the end of the report if it uses totals. Is there a way to automatically force a form feed at the very end of each report, totaled or not?*

*2. Due to a "procedural problem" I would like to have a summary report which compares open order quantity totals against the committed quantity in the inventory master. I was able to run a detail report which produced many pages which had to be manually scanned to see if these quantities agreed, but I could find no way to bring the committed value read from the second file (inventory master) down to the subtotal line. **IDOL** is very sensitive about letting you modify the subtotal logic even though you are given the option of modifying it.*

Ideally, this would be an exception report showing only those items where the committed quantity does not equal its open order quantity subtotal. Can you help me?

Answer: Your letter actually contained six questions about the **IDOL** report generator. I will address these first two here and, hopefully, pick up on the remaining four in future articles.

Before we start, I would like to say a few words about your general predicament. You are a prime example of the new breed of "survivors" which was created by Fortune Systems in it's early days of marketing. Back when Fortunes were being hawked in the shopping marts and toy stores of the nation, unknowing salespeople were

Continued on next page

SAVE ON PROGRAMMING COSTS!
ENHANCE YOUR BASIC/IDOL/BAS REPORTING CAPABILITIES!
FREE "SmartCable" WITH EACH ORDER.

THOUROUGHBRED REPORT is an easy-to-use, powerful report generating system for BASIC/IDOL/BAS that allows you to create a variety of customized reports and forms without having to program. Great for creating such things as Customer Lists, Financial Reports, Order Forms, Payroll Checks, and Ad Hoc reports. Capabilities by far surpass those of the standard IDOL report generator. Highlights include:

* Multiple concurrent file access - can access any BASIC/IDOL/BAS file for a report and can access multiple files at the same time. For example, it can access your accounting, payroll, name and address, customer and sales files simultaneously.

* Full report windowing - supports horizontal scrolling of the report, allowing complete on-line viewing of the report.

* Full control automatically - computes control and page breaks, totals, subtotals, sorts on any field. It retrieves specific records or a range of records.

* Full-screen, free-form screen painter - allows easy placement of headings, footings, boldface, underlining and word processing text into the report. Simple editing commands allow move, delete and change operations.

* No programming knowledge required - unlike the IDOL report generator, complex reports can be created without programming.

* Saves on programming costs - eliminates the need to pay for custom programming of reports. The product will pay for itself in no time!

FREE SmartCable WITH EACH ORDER. SmartCable automatically connects any two RS232 serial devices. Eliminats those cable making frustrations. Cost of Thoroughbred Report is $1895.00.

UNI-KOMP, 9803 Rocktree, Houston, Texas 77040 - (713) 937-7782
Thoroughbred Report is a trademark of Concept Omega Corporation.

peddling **IDOL** as a programming tool for the novice end user. Most buyers threw in the towel when they encountered the first systems flow chart in the user's manual. But a select few, such as yourself, bit the bullet and pressed on. You have my respect and sympathy. The original authors of **IDOL** never intended to teach the world to program, but since you've come this far, let's see what, if anything, we can do to help.

1. The form feed issue is one I have heard many times before. A complete software system works fine if all components follow the same protocol, but Fortune Systems were originally glued together with a spreadsheet from here, disk drive from there, accounting package from somewhere else and a power supply from a box in the back room. I am tempted to tell you that the **IDOL** method of feeding forms was here first, so go complain to someone else (but I'm not that mean).

I think it will help you if you understand how the **IDOL** reports are generated. When you define a permanent report, your responses to the questions are translated into **BASIC** programming code and written into an indexed file named **RnnnXX** (where **R** is constant, **nnn** is the file number from which the report is generated and **XX** is a unique, user-defined report name). When you request to run the report, the coding is read from that file, MERGEd with a "skeleton" program named **CUTFI2**, and SAVEd under the name CUTFTn (Tn being the number of the terminal running the report). You can print a listing of **CUTFI2** with the **BASIC** utilities if you wish to examine the flow of logic. If you wish to see how it looks after the **MERGE**, run an **IDOL** report on **T0** (terminal number zero) and then print a listing of **CUTFT0**.

A program named **CUTFIB** is executed at the end of all **IDOL** reports. You can force an end-of-report form feed at this point by adding a line of code to **CUTFIB**. To do this, from any **IDOL** or **BAS** selector, type in BASIC and <CR> (carriage return). At the **BASIC** prompt, type the following commands:

```
LOAD "CUTFIB"  <CR>
200 PRINT (6,ERR=201) <FF><CR>
SAVE <CR>
RUN "DOL" <CR>
```

This will bring **IDOL** reports into sync with your unix applications, but will disrupt continuity within **IDOL/BAS**. There are form feeds at the beginning of each printed report (whether **IDOL** defined or hard coded in **BASIC**) which are going to be very difficult to disable. Form feeds are built into the heading print line of each report, and it would be an enormous task to go through and tell each program to skip the form feed on page one only. Therefore, if you want to have the form feed at the end of your reports, you will have to live with having your printer feed an extra blank page between reports.

*The formfeed issue is one I have heard many times before. You can force an end-of-report formfeed by adding a line of code to **CUTFIB.***

2. Solving this problem pushes the **IDOL** report generator a little further than it was designed to go, but I think we can do it. We are going to have to "fool" the subtotaling routine. See if this works when defining a report for **COODT**:

Begin by defining a new field with a length of 8, precision 0. The field definition should be 0 (i.e., zero, no quotes). For purposes of identification here, we will call this "NEW FIELD".

During the "Before Report Logic" open a channel (e.g. 1) to the **Inventory Master File**. Also set a variable, such as **B8**, to -9999.99 or some other unique number which will not appear in the quantity committed field for any item number:

```
5600 CLOSE (1); OPEN (1)"CINV1"
5610 LET B8=-9999.99
```

When you get to "After Read Logic" examine **B8** to see if it is -9999.99. If so, read the **Inventory Master File** using the item number, and put the Quantity Committed value into **B8**:

```
5011 IF B8=-9999.99 THEN READ
(1,KEY=E$(10,12))*,*,*,*,*,*,B8
```

(Note, this may not be the proper field position in the record, but I'm sure you get the concept.)

At the "Before Subtotal Logic" it's time to play our trick on the report generator. Tell the subtotal routine to forget everything it has been calculating in the **NEW FIELD** with the following command:

5700 LET A(4)=B8, B8=-9999.99

Because we will be sorting by item number, **B8** will contain the committed quantity from the previous item number read. By resetting the value to -9999.99, we will cause **B8** to pick up the next committed quantity value when the "After Read Logic" is again encountered. I must emphasize that when we use **A(4)** we are assuming that you will be displaying only two subtotals in the report. If you wish to see subtotals for more than two fields, see pages 4-8 and 4-9 in the **IDOL** reference guide for an explanation of the **A** array.

Next, once defining special logic is completed, be sure to sort the report by item number (field 3). Also, you must ask for totals for the Order Quantity (field 6) and **NEW FIELD** (field 16). Request subtotals by Item Number (field 3).

As far as getting an "Exception Report", I must advise you not to press your luck. If you define the above report as a summary report only, you should be able to find the information you need without printing several hundred pages of detail. Be happy for small favors! □

SOFTWARE FOR THE FORTUNE 32:16

MULTIPLAN LINK - can pass data from BAS/IDOL/BASIC files to Multiplan and back. For example, it could be used to send data from any of the BAS files (Chart of Accounts, Customer Master, Inventory Master, etc). Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

RECORDS PROCESSING LINK - can pass data from BAS/IDOL/BASIC files to Records Processing and back. For example, it could be used to take customer address data from the Customer Master and will automatically make the Records Processing List document for mailouts. Allows selection of fields to be sent and provides logical retrieval and key range selection. **$295.00**

TERMINAL/PRINTER LINK - provides capability to print Multiplan, and BAS/IDOL reports on a printer connected to a workstation, view any report on the terminal before printing, convert any report to Fortune:Word format; accumulate multiple BAS/IDOL reports before printing, backup reports onto diskette; copy or rename report files; print multiple copies of the report on either the primary and/or terminal printer. Also has option to print BAS/IDOL reports on HP LASER printer ($50 extra) **$195.00**

TELEPHONE LINK - on line telephone message, telephone directory and inter-office mailbox/memo system which can also be used as a simple calendar reminder system. Provides control, printing and viewing of on-line data for each user. **$295.00**

FINANCIAL LINK - stores BAS Income Statement and Balance Sheet data in files. Once stored, it can be sent to Multiplan or instantly printed. **$195.00**

MENU LINK - provides software and documentation for adding items to the Global Menu as well as a sub-menu for your use Includes backup script for user directories and BAS/IDOL/BASIC programs and data. **$95.00**

CALCULATION LINK - provides programs for amortization, depreciation, loan repayment, averaging (with graph), linear correlation, and breakeven analysis. **$95.00**

KOMPACT PERSONNEL ACCOUNTING - provides for data capture of personal, wage/salary, job, education, salary/wage, appraisal training, dental, medical and life insurance data. Includes over 80 pre-defined reports/worksheets. **$495.00**

CRT LINK - provides information for using other terminals on a Fortune System (WYSE, TV, IBM 3101). **$95.00**

AUTOMATIC REPORT LINK - provides capability to print numerous IDOL defined reports automatically based on frequency or application without having to select each from a different menu. You identify frequency/application codes. **$195.00**

TAPE LINK - provides one menu for all tape backup utilities. Provides daily, weekly, monthly and individual backup and restoration routines. Includes rewind, retension, and tape unit startup/stop utilities. User friendly. **$95.00**

PRINTER LINK - provides printcaps for Okidata 84, 2410, TI 855, and Anadex 6500. Includes shell scripts for changing pitch and LPI. Allows these printers to be used in word processing. **$95.00**

MODEM LINK - allows Hayes 1200 Smartmodem or compatible to be used on one port for both incoming and outgoing calls. **$95.00**

INSTALL LINK - provides documentation and software example for installing your product/menu through "S5". **$95.00**

REPORT LINK - modifies IDOL Report Generator Program to allow stacking of up to 8 fields in an IDOL defined report. Also corrects problem with displaying subtotals for all stacked fields. **$95.00**

WESTERN LINK - provides instructions on how to use your Fortune for sending/receiving telex messages using Western Union's Easylink. Can replace your telex! Fortune ITE software is required. **$95.00**

BACKUP LINK - provides menu selection for backing up /b (BAS/IDOL/BASIC), /u (Fortune:Word, Multiplan) and entire system onto diskette. Also has program to "unhang" a terminal that's hungup just by entering the terminal #; a program to restore one file from a multi-volume backup; and a selection to tell you which volume a backed up file is on. **$95.00**

UNI-KOMP 9803 Rocktree Houston, Texas 77040 (713) 937-7782

The UNIX Directory

Introduction to Shell Scripts - Part 5

by David E. Kloes

David E. Kloes is President of UNI-KOMP which is located in Houston, Texas. He provides UNIX seminars, software for the Fortune Computer and is vice-president of the Houston UNIX User's Group. He contributes independently to ***/u/fortune*** *news.*

In Part 4, we completed our discussion of the **if then elif else fi**, **for do done**, and **while do done** command sequences. That leaves us with a clean slate and in this issue, we will be discussing some new commands and concepts.

THE *until do done* COMMAND SEQUENCE

Remember we said that the **while do done** command sequence is used to loop through a set of instructions while some condition exists. We have another command sequence that basically functions the same except that it loops *until* some condition exists. This command sequence is appropriately called the **until do done** command sequence. Let's take a look at our last **countup** shell script using the **while** statement:

```
x=1
while test $x -lt 6
do echo -n "$x "
   x=`expr $x + 1`
done
echo
```

Now let's rewrite the same shell script using the **until do done** loop. If you've forgotten what the above shell script does, review your last **/u/fortune** issue.

```
x=1
until test $x -eq 6
do echo -n "$x "
   x=`expr $x + 1`
done
echo
```

This shell script produces the same result as the first example using the **while** loop. The only changes we have made is we have replaced the word **"while"** with **"until"** and we have changed the **"-lt 6"** to **"-eq 6"**. The logic of the **until** loop says to process through the loop until the value of **x** is equal to 6. Like the **while** example, this means that the instructions between the **do** and the **done** are executed for the values 1-5. When the value of **x** is equal to 6, program control goes to the statement following the **done** statement.

The choice of the **while** or **until loop** is usually a programmer preference, depending on the job to be done since they pretty much function in the same way. To reinforce these two commands, let's take a look at a more realistic example of how they can be used. Many times in a shell script, we will ask a question which requires an operator response. If the question is answered with a valid response, we continue on with the shell script. If it is answered with an invalid response, however, we will want to display some error message and force the operator to answer the question again. The **until** or **while loop** can be used for this type of situation.

Many times in a shell script, we will ask a question which requires an operator response. The *until* or *while* loop can be used for this type of situation.

```
good="no"
while test "$good" = "no"
do echo -n "^L^GDo you want to
continue (y/n)? "
   read answer
   case $answer in
     y|Y|yes|YES) good="yes";;
       n|N|no|NO) exit;;
               *) echo -n "
^GInvalid response
Depress <RETURN> to continue: "
                  read bunk;;
   esac
done
```

In this example, we initially set the variable **good** to "no". The **while** statement says that as long as this variable is equal to "no", continue to execute the commands between the the word **do** and **done**. Since we set the variable equal to "no" initially, the loop will be executed the first time it is encountered. Whether the loop continues to be executed (the question is asked again) depends on what the operator answers. If they answer `y`, `Y`, `yes` or `YES`, we set the value of the variable **good** equal to "yes". This means the question will not be asked again and the shell script will continue with whatever statements follow the **done** because the **while** test says to re-ask the question only if the variable **good** is equal to "no".

If the operator answers the question `n`, `N`, `no` or `NO`, they have correctly answered the question and we **exit** since they have indicated correctly that they do not want to continue. In this case, we do not have to worry about the **good** variable since **exit** will exit us from the shell script.

For any other answer, the error

Continued on next page

message is displayed, and we leave the value of **good** set to "no" which means the question will be asked again. This loop will continue to be executed until they give a correct response. Note that we also clear the screen and ring the bell (^L^G) each time the question is asked. We also ring the bell and wait for any key on the keyboard to be depressed when the error message is received. This method is preferred over the **sleep** command since it will insure the user has seen and responded to an invalid answer. Now let's look at the same shell script using the **until** loop:

```
good="no"
until test "$good" = "yes"
do echo -n "^L^GDo you want to
continue (y/n)? "
   read answer
   case $answer in
     y|Y|yes|YES) good="yes";;
       n|N|no|NO) exit;;
               *) echo -n "
^GInvalid response
Depress <RETURN> to continue: "
                  read bunk;;
   esac
done
```

The only line we changed was the **until** line which now says to continue to ask the question until the variable **good** is "yes".

THE *break* COMMAND

There are occasions when you are executing a **while**, **until** or **for loop** and you want to get out of the loop but you do not want to exit the program. In this situation, instead of using the **exit** command, use the **break** command. When a break is encountered, the loop is exited and control passes to the statement following the **done**.

This concludes our discussion of *command sequences*. We do, however, have some other single shell commands and some new concepts to discuss.

EXECUTING MULTIPLE SHELL SCRIPTS

There are a variety of reasons why we would have a need to execute other shell scripts from an initial shell script. We might have a main shell script that draws a menu and then depending on the menu item selected, another shell script is executed. After that shell script is finished, we may or may not want control to come back to the menu shell script.

We may also have *called* program routines that are frequently used routines that can be used by any shell script. In the above examples, the "Invalid response" message can be put into a **called** shell script to keep us from having to retype the same lines over and over again in our shell scripts. Here's an example where we'll use two shell scripts to save some coding time (See Figure 1.)

Figure 1.

```
Program 1: main.sh

export msg
good="no"
until test "$good" = "yes"
do echo -n "^L^GContinue (y/n)? "
   read answer
   case $answer in
     y|Y|yes|YES) good="yes";;
       n|N|no|NO) exit;;
               *) msg="Invalid response"
                  message.sh;;
   esac
done
```

```
Program 2: message.sh

echo -n "
^G$msg.
Depress <RETURN>: "
read bunk
exit
```

If you need some coffee or a break, now is a good time because you need your thinking caps on for this discussion. If you haven't noticed, we are beginning to make our discussions and examples more complex. At some point we must assume you have absorbed and practiced the material as we have been going along. If you have not, the remaining material may not make much sense to you since we now assume you have read and understand commands and concepts we have discussed to this point.

We have created two separate shell scripts here. One is called **main.sh** and it is the first shell script that runs. The second one is called **message.sh** and is called from the first shell script, if an invalid response is entered. You will notice also that we have a new command in the first line of **main.sh** - the **export** command.

PASSING VARIABLE VALUES (*export*)

We now digress (there's our favorite word again) to a previous discussion. We ran a shell script that had the **cd** command in it and asked you what directory you would be in after you exited the shell script. The answer was that you would be in whatever directory you were in when the shell script was executed. At that time, we pointed out to you that once a shell script had been executed, the shell forgot the fact that it had changed directories.

In more technical terms, a shell script starts up a *process*. While the process is running, the shell remembers all of the environmental parameters that were set in the shell script. Once the script has been exited, however, the environment goes back to the state it was in prior to the script being run. The same concept is true for shell variables. While you are in a particular shell script, any variables that have been set are remembered. If the shell script is exited or if another shell script is executed, the value of any variables is forgotten by the shell. The export command is used to keep the value of any variable in the environment for use by **SUBSEQUENT** shell scripts. Previous shell scripts, however, will not remember value changes to variables even though they are exported.

In the above example, we **export** the **msg** variable in the **main.sh** shell script so that it can be referred to in

Continued on next page

the **message.sh** shell script. If we had several different error messages throughout the **main.sh** script, we could *call* the **message.sh** script assigning a different value (message) to the **msg** variable as needed. Obviously this wouldn't be worth the effort of a second shell script if we only had *one* shell script and *one* error message. But if we have several messages in one shell script or several shell scripts with different error messages, the call script would save us the time of having to put the error logic in each place. The above method, therefore, gives us the ability to use one message script that can be used as needed. If we have several different people writing shell scripts, it also produces a standard way of producing messages regardless of who the author is. This same concept can be used for other commonly used routines when writing shell scripts.

PASSING VARIABLE VALUES (*Positional Parameters*)

A second method of passing variables and their values from one shell script to another is through the use of **positional parameters**. Refer now to Figure 2.

Whenever we run one of our shell scripts, we can pass parameters (variables) to the shell script by entering them after the program name. When this is done, the name of the shell script that is being run can be referenced as **$0**. Each **positional parameter** after that can be referenced as **$1** through **$9**. In example 1 above, **$0** is equal to **name.sh**, **$1** is equal to **David** and **$2** is equal to *nothing* since we did not pass anything else. Anytime there is a space (the **UNIX** *delimiter*) between words, the next "word" is assigned to the next **positional parameter**. In example 2, **$0** is still equal to the program name **name.sh**, **$1** is equal to **David** and **$2** is equal to **Kloes**.

Example 3 presents an interesting situation. Why is *Kloes* omitted when the line is echoed? Remember we said that "white space" (spaces) is what **UNIX** uses to determine the **positional parameters**. In this case, **$0** is **name.sh**, **$1** is **David** and since there is a space between "David" and "E." and "Kloes", **UNIX** sees **E.** as the second positional parameter (**$2**). **Kloes** is the third **positional parameter** (**$3**) but our **name.sh** shell script says to only echo **$1** and **$2** so **$3** (Kloes) is omitted. Example 4 is how to get around this problem. By putting the quotes around **David E.**, **UNIX** will now treat this as one string or parameter.

Using this second concept of **positional parameters**, let's rewrite our two shell scripts (See Figure 3.).

Notice that we have removed the **export** line from **main.sh** and changed **message.sh** to **message.sh "$msg"**. Note the quotes around **$msg** so that the entire message (regardless of the number of words) is passed as **$1**. We then changed the **message.sh** script to echo **$1** instead of **$msg**. Either the **export** or **positional parameter** method may be used to achieve the same results. A typical session using either of these methods is shown in Figure 4.

We've probably taxed your brain enough for one session. We will continue with this topic next time and (as usual) introduce you to some new material. Be sure to practice using these examples and some of your own.

Figure 2.

```
Program:  name.sh

echo "My name is: $1 $2"

Example 1:  $ name.sh David
                My name is: David
Example 2: $ name.sh David Kloes
                My name is: David Kloes
Example 3:  $ name.sh David E. Kloes
                My name is: David E.
Example 4:  $ name.sh "David E." Kloes
                My name is: David E. Kloes
```

Figure 3.

```
Program 1: main.sh

good="no"
until test "$good" = "yes"
do echo -n "^L^GContinue (y/n)? "
   read answer
   case $answer in
     y|Y|yes|YES) good="yes";;
       n|N|no|NO) exit;;
                  *) msg="Invalid response"
                     message.sh "$msg"
   esac
done
```

```
Program 2: message.sh

echo -n "
^G$1.
Depress <RETURN>: "
read bunk
exit
```

Figure 4.

```
$ main.sh                (Screen clears, bell rings)
Continue (y/n)? xyz      (xyz is an invalid response so message.sh
                          runs)
Invalid response.
Depress <RETURN>:        (After <RETURN> is depressed, message.sh
                          is exited and we reenter main.sh where
                          we exited to run message.sh.  "good" is
                          still equal to "no" so loop runs again)
Continue (y/n)? y
$
```

Unix Basics

Continued from Page 1

Menu prompt, simply type `!sh`. This will "spawn" a **UNIX** shell, known as the **Bourne** shell (named after the man who created it). As we have explained in the past, a shell is a *command interpreter.* That is, it takes whatever you type in and tries to do it. For the record, the **Global Menu** is also a command interpreter. Instead of typing the **command, however, you use various tricks to highlight the name of a command, push the EXECUTE** key, and then the **Global Menu** starts that particular command running.

Now, if you type `!sh` you will soon be presented with a `$` (dollar sign) which is called the *prompt.* It is your signal that the computer is ready to accept a command. For example, you could type `ls <CR>` (the `<CR>` means to type the **RETURN** key on your keyboard). Go ahead, try it. You will see a listing of the files in the directory that you are in. By the way, this **UNIX** command is exactly what you are doing when you select the **S1** option from the **Global Menu** and then choose **Option 25** and then choose **list.** Yes, that's right, the **Global Menu** is running the **UNIX** ***ls*** command. Try it. First type `ls <CR>` to the **UNIX** `$` prompt. Look at this display and hold it in your mind's eye for a moment. Then type **CTRL-D** (that is, hold down the grey **CTRL** key and simultaneously depress the **D** key). This will take you back to the **Global Menu.** Then type **S1,** followed by **25,** followed by **l,** then another **<CR>.** The display you see will be identical to the display produced by the **UNIX** ***ls*** command.

Back out of the Menu and get to the **UNIX** shell again. Now we will show you a variant of the **ls** command. Type `ls -l <CR>` (that `-l` is a dash and then the letter ell) to the dollar sign prompt. You will soon be greeted with a listing of the file names in the directory you are in, but this time you will see much more information. The **-l** part of the `ls -l` command stands for *long listing.* It gives you some very helpful information about each file, such as its **file permissions,** its **size** in characters, the **date** it was last modified, and so forth. This is actually a wealth of information and is very useful in many circumstances. If you want to learn more about what each of these things mean, we direct you to your trusty **FOR:PRO** manual or any

*Most **UNIX** commands have switches that modify the way the command works. The **pwd** command, however, has no switches. It does one thing–prints out the name of the directory in which you are working.*

Continued on next page

SST DATA INC.

Introduces

HANDSHAKE™ BLAZER

for high speed data communications. The newest member of the HANDSHAKE family offers error free transmission of binary and ASCII files between UNIX systems using low cost, dial-up lines. HANDSHAKE BLAZER achieves throughput speeds up to 9600 bps with these features:

* **End to End Error Correction** utilizing CRC algorithm
* **Auto Restart** permits restart from point of interruption
* **Send, Receive and Mail** supports commands from all nodes
* **User Friendly** menus, function keys Autodial, Autolog
* **Automatic Send Capability** with easy to use menus
* **Speed Optimizer** insures maximum speed compatible with line conditions
* **Audit Trail of File Transfers** complete history of transactions
* **Interactive Terminal Option** interactive mode also supports text file transfer

SST Data, Inc. 4701 W. Schroeder Dr. Milwaukee, WI 53223 (414) 355-6990 Telex 590597

book on the **UNIX** system. By the way, this output is exactly what you would get if you chose **Option 25** from the **S1** screen in the **Global Menu** and then chose the **d** (for display) sub-option.

So far, we have learned how to use the **ls** command. It lists the names of the files in the directory in which you are working. But how do you know which directory this is? The **UNIX** command called **pwd** tells you this. **Pwd** stands for **p**rint **w**orking **d**irectory. This command corresponds to another one of the options on the **S1** screen of the **Global Menu**. This option is number **27, Name Current Directory**. Again, the main purpose of this command is to remind you where you are in the system. Granted, most of the time you will remember where you are, or perhaps you don't move around too often. However, there are times when it is necessary, or convenient, to change directories. As you move around more and more, you will find this command increasingly useful. Interestingly, most **UNIX** commands have options, or switches, that modify the way the command works. For example, remember how the **ls** command by itself just printed out the names of the files in the directory? Well, when we added the **-l** extension, or switch, we altered what the ls command did - we received more information. The **pwd** command, however, has no switches. It does one thing and one thing only - prints out the name of the directory in which you are working.

We have alluded to the idea of moving around in the **UNIX** file structure. This is accomplished with the **cd** command. This command has its **Global Menu** counterpart, also on the **S1** screen. Choose the 23rd option and see what happens. You will see the following instruction, **Enter Directory You Wish To Go To:** This is the Global Menu's way of asking you to input the name of a directory because the whole purpose of the **cd** command is to switch you from your current directory to another directory. This kind of switching can become necessary all the time. For example, suppose one of your colleagues, who shares an account on the same Fortune computer that you do, walks up to you and wants to look at a file in his or her directory. How do you do that? Well, one way, from the **UNIX** shell, would be to **cd** into your colleague's directory and then use the **cat** or **more** command to view the contents of the requested file. But all of this was predicated upon somehow getting to your colleagues directory. Suppose your account name is smith. This means that your home directory would probably be /u/smith. If jones walks up to you and wants to look at file in /u/jones, you would simply type `cd /u/jones <CR>`. This would change directories from **/u/smith** to **/u/jones**. Then you could do whatever jones wanted you to do.

In this first part of Unix Basics we introduced you to three of six fundamental **UNIX** commands. These commands were **ls**, **pwd,** and **cd** (change directories). In a future issue we will discuss the last three of these fundamental commands which are called **cp**, **mv**, and **rm**. These commands are used for copying, renaming, and deleting the contents of files. □

Mark Palmerino

UNISOFT INC.

Experts in Computer Consulting - Sales - Service

Presents

FORTUNE SYSTEMS HELP

We Provide:

- Expert Trouble-shooting
- Years of programming and accounting experience
- *Free Initial Consultation*
- Training Available for:
  - * BASIC/IDOL/BAS and Multiplan
  - * UNIX and Ext. Fortune:Word
- Hardware Repairs & Upgrades
- FORTUNE trained experts at a cost that will surprise you!
- *Trade-Ins Welcome*
- Specializing in:
  - * Hard Disk Trade-In
  - * Hard Disk Upgrades

Authorized Fortune Master Dealer

Authorized Fortune Systems Repair Depot

CALL: (303) 671-8600 EMERGENCY HELP: (303) 691-HELP

UNISOFT INC. 14100 East Jewell Avenue Suite 27 Aurora, Colorado 80012

MAKE YOUR FORTUNE GROW

One of the greatest assets of your Fortune Systems 32:16 is its built-in ability to expand to match your need.

Whether you want to increase your number of workstations, upgrade system capability, or tie-in your Fortune Systems cluster to a mainframe, Micro Resources can give you the prompt, complete service you need.

As *the* Fortune Systems Master Dealer in Ohio, we have the hardware and software you need available for immediate shipment. From Fortune:Link to uninterruptible power supplies, from storage expansion systems to Development Utilities, we have the exact blend of hardware, software and systems expertise needed to make your Fortune Systems 32:16 do exactly what you want it to in the most reliable and cost-efficient manner possible.

And our genuinely friendly staff programmers and systems analysts are available to provide you with whatever additional support you may need, from software consultation and customization to integrated systems design and specification.

So whether you need a quick question answered or a new service agreement, Business Applications Update or just some networking cable, call Micro Resources.

We'll help your Fortune Grow.

Micro Resources

535 Metro Place South Dublin, Ohio 43017 (614) 766-2335

Index to /u/fortune *news*

This past year has been a productive one for us at **/u/fortune** *news*. As a publication, we have grown in the quantity of information that we publish each month. One year ago at this time **/u/fortune** *news* was a 16 page newsletter that was published just slightly more than every two months. This year we have produced 12 issues and have grown from 16 pages to 24 pages. All in all that's more than a 100% increase.

But quantity isn't everything. We hope you will agree that the quality of the newsletter has continued to improve. We basically attempt to provide the kind of information that will help you use your Fortune hardware and software to its greatest potential. We think we have been successful at that. However, we know that there is always room for improvement. So, please write us whenever you think of something that we can do to better help you use your equipment.

Yes, we have published much useful information this year and to help you have easy and quick access to it, we are publishing an end of year index of the contents of **/u/fortune** *news* for 1986. On the remaining pages in this issue you will find this index. In order to use the index effectively, however, you should be aware of some abbreviations and notational conventions.

The particular issue and page number of where a keyword can be found is notated as follows: **V.N:P**. **V** stands for Volume, **N** stands for Number and **P** stands for page number. For example, 3.3:6 would indicate Volume 3, Number 3 and page 6.

The following table summarizes the article abbreviations used in the index:

BAS The BASIC Advisor
Help /u/help
News News from Fortune
RR Readers Respond
MP Multiplan
UD The UNIX Directory
FW The Fortune:Word Glossary

***FPSD**
BAS (3.3:6)
***TPSD**
BAS (3.3:6)
.cshrc file
Help (3.11:16)
/etc/rc
Help (3.9:10)
1.8.1.1 For:Pro
BAS (3.8:1)
4th Generation Software
BAS (3.11:3)
68020 product announced
News (3.10:2)
798 Utility
BAS (3.8:8)
@T command/BASIC
BAS (3.6:21)
A/R Customer Inquiry/Aging
Tech Tips (3.10:14)
A/R Trial Balance/aging invs
Tech Tips (3.10:14)
ABA user group
RR/Fortune:Word Legal Users Unite (3.9:8)
Adding Terminals to BASIC
BAS (3.9:1)
Adding Terminals/BAS liscens
BAS (3.10:1)
ADDR/BASIC
RR/Re: Error 33 in BASIC (3.4:2)
aging of current invoices
Tech Tips (3.10:14)
Alt Console/BAS prnt
Tech Tips (3.8:10)
amortization
MP/Loan Amortization Analysis (3.10:1)
appointment calendar
UD/A Computerized Appointment Book (3.5:2)
appt command
Free Software Highlights/D.C. Grab Bag (3.4:6)
ASCII file transfer
Linking Computers Together w/ UUCP (3.5:6)
ASCII format
FW/Moving Documents into Fortune:Word (3.3:1)
autologout shell script
Autologout Shell Script (3.10:9)
backup, multi-volume
Help (3.4:8)
bad blocks
UD/Bad Blocks/What to Do With Them (3.1:1)
BAS Support/Choice Systems
BAS (3.6:18)
BAS/IDOL Release Level 2.0
BAS (3.6:21)
basename
Help (3.11:16)
BASIC Support/Concept Omega
BAS (3.7:1)
BASIC to ASCII
FW/Moving Documents into Fortune:Word (3.3:8)
BASIC/Adding more terminals
BAS (3.9:1)
BASIC/Adding more users
BAS (3.7:10)
BASIC/File xfer with Windows
Help (3.7:21)
BASIC/LAN printer problem
Tech Tips (3.6:12)
binary file transfer
Linking Computers Together w/ UUCP (3.5:6)

Continued on next page

bookmarks
FW/Moving Around Fortune:Word (3.9:16)
boot, system startup
Help/Power Up Sequence (3.2:14)
break command
UD/Intro to Shell Scripts, Part 5 (3.12:12)
bulletin board
Electronic Bulletin Board/Using the Sequencer Subsystem (3.2:12)
Burroughs Terminal Emulator
News (3.6:3)
C Programming Lang 2.0 anncd
News (3.10:12)
calculator
FW/Making a Fortune:Word Calculator (3.2:1)
calling shell scripts
UD/Intro to Shell Scripts, Part 5 (3.12:12)
case esac command/Shell
UD/Introduction to Shell Scripts, Pt 2 (3.9:12)
cat command/how to use
UD/Viewing Files with cat and more (3.7:5)
CCNVZ
BAS (3.8:8)
BAS (3.2:3)
cd command
Unix Basics (3.12:1)
CGLBD
BAS (3.8:8)
check ledger
MP/Check Ledger and Reconciliation (3.11:1)
chl command/Multipln
Multiplan Shortcuts (3.1:9)
chmod
UD/Introduction to Shell Scripts, Pt 1 (3.8:14)
Choice Systems/BAS Support
BAS (3.6:18)
CMT command/BASIC
BAS (3.4:11)
Comm-A board support w/2.0
Help (3.10:15)
comments in shell scripts
RR/Multi-volume Backup, Shell Comments (3.5:16)
communications
UUCP Fundamentals -- Part II/Configuring Fortune-to-Fortune Copy (3.6:6)
communications/kermit
Help (3.7:14)
Concept Omega
BAS (3.7:1)
BAS (3.10:22)
configuration block
Help (3.1:14)
Control Codes/F:Word
FW/Moving Documents into Fortune:Word (3.3:8)
control document/Records Pro
FW/Basics of Records Processing, Pt I (3.5:9)
FW/Basics of Records Processing, Pt II (3.6:17)
COODT
BAS (3.12:9)
copy right or down shortcuts
RR/Multiplan Tips (3.12:5)
cp -B command problem
Help (3.8:6)
RR/Multi-volume Backup, Shell Comments (3.5:16)
Monthly Bug Status Report (3.10:15)
cp -B command problem w/1.8
Help (3.4:8)
cp -Broust/restart in middle
Tech Tips (3.6:12)
cp command arguments
Help (3.12:7)
crash recovery
Help (3.9:10)
csh prompt
Help (3.11:15)
CSHTO
BAS (3.3:6)
cul devices/communications
Help (3.7:14)
cursor movement/Fortune:Word
FW/Moving Around Fortune:Word (3.9:16)
CUTFI2
BAS (3.12:9)
CUTFIB
BAS (3.12:9)
date1 command/Records Proc
FW/Basics of Records Processing, Pt I (3.5:8)
Davis, Robert
Fortune Sells Solutions/Robert Davis and Paul Olin (3.8:5)
News/Davis/Olin Join Fortune (3.7:2)
dealers, master
News/Fortune's Master Dealer Program (3.1:8)
Desktop Publishing
FW/Records Processing as a Typesetter (3.8:2)
DIM command
BAS (3.8:1)
BAS (3.6:21)
disk error
UD/Bad Blocks/What to Do With Them (3.1:1)
disktab file in 1.8.1.1
Help (3.10:15)
done command
UD/Intro to Shell Scripts, Part 5 (3.12:11)
drives, limits by controller
Help (3.10:15)
electronic rolodex
Electronic Rolodex, An (3.11:1)
else command
UD/Intro to Shell Scripts, Part III (3.10:6)
emacs editor
Help (3.10:16)
erasing bulk data/BASIC
BAS (3.8:8)
Error 11/BASIC
BAS (3.2:3)
Error 13/BASIC
BAS (3.2:3)
Error 2/BASIC
BAS (3.3:6)
Error 33/BASIC
BAS (3.2:3)
RR/Re: Error 33 in BASIC (3.4:2)
BAS (3.4:10)
Error fixing in BASIC
BAS (3.4:10)
error message "No Shell"
Help (3.11:15)
Error, hard disk
UD/Bad Blocks/What to Do With Them (3.1:1)
ESC key releases BASIC
BAS (3.11:3)

Escapes and Exceptions
BAS (3.3:1)
BAS (3.4:10)
exit command
UD/Intro to Shell Scripts, Part III (3.10:6)
expiration list/F:W Glossary
FW/Basics of Records Processing, Pt II (3.6:18)
export
UD/Introduction to Shell Scripts, Pt 1 (3.8:17)
export command
UD/Intro to Shell Scripts, Part 5 (3.12:12)
expr command
UD/Intro to Shell Scripts, Part III (3.11:12)
External Copy
Multiplan Shortcuts (3.1:14)
fgrep
UD/A Computerized Appointment Book (3.5:4)
Financial Report
News/1st Quarter 1986 Results Reported (3.5:10)
News/3rd Quarter 1986 Results Reported (3.11:2)
News/2nd Quarter 1986 Results Reported (3.8:12)
News/4th Quarter 1985 Results Reported (3.2:2)
find
Help (3.1:4)
finger command
Autologout Shell Script (3.10:9)
floppies, clearing w/o fmtng
Help (3.4:9)
floppy formatting w/1.8
Help (3.4:8)
font control
Special Print Fonts with Multiplan/Using the Wheels Argument (3.3:2)
for do done loop/Shell
UD/Introduction to Shell Scripts, Pt 2 (3.9:12)
For:Pro 1.8.1.1
News/1st Quarter 1986 Results Reported (3.5:10)
For:Pro 1.8.1.1/Shutdown
Tech Tips (3.6:4)
For:Pro 2.0
News (3.12:2)
For:Pro 2.0 announced
News (3.10:4)
form feed at end of reports
BAS (3.12:9)
Format Document/Records Proc
FW/Basics of Records Processing, Pt I (3.5:8)
Formula announced
News (3.10:2)
Fortune-to-Fortune Copy
Linking Computers Together w/ UUCP (3.5:6)
UUCP Fundamentals -- Part II/Configuring Fortune-to-Fortune Copy (3.6:6)
Fortune:Link printer problem
Tech Tips (3.6:19)
Fortune:Windows/BASIC xfer
Help (3.7:21)
Fortune:Word 3.0
News (3.12:2)
Fortune:Word 3.0 announced
News (3.10:3)
fsck command/scratch file
Tech Tips (3.6:12)
grep command
UD/The grep Command (3.4:1)

Continued on next page

PSG

We are the *largest* Authorized Fortune Dealer in the Pennsylvania, New Jersey and Delaware area. As part of the **P**rofessional **S**olutions **G**roup of companies with over 35 years in business automation, we have the *stability* that our customers demand

Our services include a factory trained field engineering team, a complete parts inventory including system upgrade capability and a software support department.

For all of your Fortune needs contact:

Mark Shields of PSG at
2550 Eisenhower Avenue
PO Box 762
Valley Forge, PA 19482

or you can call Mark at (215) 666-1400

UD/A Computerized Appointment Book (3.5:4)
RR/Multi-volume Backup, Shell Comments (3.5:16)
header page/Fortune:Word
FW/Using Prototype Documents (3.4:12)
HOME
UD/Introduction to Shell Scripts, Pt 1 (3.8:14)
IBM Compatible Keyboard
News/1st Quarter 1986 Results Reported (3.5:12)
IBM-PC Multiplier Hardware
News/1st Quarter 1986 Results Reported (3.5:12)
News (3.6:2)
icheck
UD/Bad Blocks/What to Do With Them (3.1:6)
idle time logout script
Autologout Shell Script (3.10:9)
IDOL II discussion
IDOL II (3.11:3)
IDOL, basic use
BAS/Creating Basic Reports with IDOL (3.1:1)
if test for numbers
UD/Intro to Shell Scripts, Part III (3.11:12)
if then else fi command
UD/Intro to Shell Scripts, Part III (3.10:6)
Importing to Fortune:Word
FW/Moving Documents into Fortune:Word (3.3:8)
Informix/Laser Printer
Help (3.12:9)
ipl files/BASIC--Addng Users
BAS (3.7:10)
ITE/Lan disconnect
Tech Tips (3.8:10)
kermit/frozen terminal
Help (3.11:15)
kermit/kmenu can't open dev
Help (3.7:14)
KEYs
BAS (3.11:3)
kmenu -- problems with error
Help (3.7:14)
L-devices/Fort-to-Fort Copy
UUCP Fundamentals -- Part II/Configuring Fortune-to-Fortune Copy (3.6:7)
L.sys/Fortune-to-Fortune Cop
UUCP Fundamentals -- Part II/Configuring Fortune-to-Fortune Copy (3.6:7)
labels, 4-up/BASIC
BAS (3.2:3)
Laser Interface, Rel 1.1
News (3.6:3)
laser printer w/Informix
Help (3.12:9)
Laser Printer/Legal sz docs
Tech Tips (3.6:4)
Laser Printer/Printcap
Tech Tips (3.6:4)
lawyers, How they use Fortns
Users' Spotlight/How Lawyers Use Fortune Computers (3.7:1)
legal software
Users' Spotlight/How Lawyers Use Fortune Computers (3.7:1)
Legal Users user group
RR/Fortune:Word Legal Users Unite (3.9:8)
libraries/Fortune:Word
FW/Using Prototype Documents (3.4:13)
list document/Records Proc
FW/Basics of Records Processing, Pt I (3.5:1)
FW/Basics of Records Processing, Pt II (3.6:1)
loan amortization
MP/Loan Amortization Analysis (3.10:1)
logout shell script
Autologout Shell Script (3.10:9)
loops/Shell programming
UD/Introduction to Shell Scripts, Pt 2 (3.9:12)
lower case mode in BASIC
BAS (3.6:21)
lowercase mode in BASIC
BAS (3.4:11)
ls command
Unix Basics (3.12:1)
ls command flags
Help (3.10:15)
m
Tech Tips (3.6:8)
mailing list/glossary entry
FW/Basics of Records Processing, Pt II (3.6:1)
Merge symbols in RP
Help (3.8:6)
MicroTrak
Project Management Sfw/MicroTrak (3.4:1)
mkconf command arguments
Help (3.12:7)
mode command/Fortune:Word
FW/Moving Around Fortune:Word (3.9:16)
more command/how to use
UD/Viewing Files with cat and more (3.7:6)
mp68
Special Print Fonts with Multiplan/Using the Wheels Argument (3.3:4)
MP/The Basics -- Starting from Scratch (3.9:3)
mpinit
Special Print Fonts with Multiplan/Using the Wheels Argument (3.3:4)
MS-DOS wrkstatn discontinued
News (3.6:2)
multi-volume backup probelm
Help (3.4:8)
multi-volume backup problem
RR/Multi-volume Backup, Shell Comments (3.5:16)
Monthly Bug Status Report (3.10:15)
Help (3.8:6)
multi-volume backup/restart
Tech Tips (3.6:12)
Multiplan
MP/The Basics -- Starting from Scratch (3.9:6)
MP/Loan Amortization Analysis (3.10:1)
MP/Check Ledger and Reconciliation (3.11:1)
RR/Multiplan Tips (3.12:4)
Multiplan and printers
Special Print Fonts with Multiplan/Using the Wheels Argument (3.3:2)
multiple shell scripts
UD/Intro to Shell Scripts, Part 5 (3.12:12)
Multiplier for IBM-PC
News (3.6:2)
Multiplier Hardware Released
News/1st Quarter 1986 Results Reported (3.5:12)
Multiplier Sftwre: COBOL
News (3.12:2)
Multiplier Sftwre: Dev Utils
News (3.12:2)
no shell error message
Help (3.11:15)
nologin
Help/Power Up Sequence (3.2:14)

Continued on next page

non-profit organizations
Users' Spotlight/Non-Profits Use Fortunes (3.9:17)
note command
Free Software Highlights/D.C. Grab Bag (3.4:6)
OEM Software Division
News/Davis/Olin Join Fortune (3.7:22)
Okidata 193 switch settings
Help (3.4:15)
Olin, Paul
Fortune Sells Solutions/Robert Davis and Paul Olin (3.8:5)
News/Davis/Olin Join Fortune (3.7:2)
othello
Free Software Highlights/D.C. Grab Bag (3.4:6)
output document/Records Proc
FW/Basics of Records Processing, Pt I (3.5:10)
padding/BASIC
BAS (3.8:7)
parity errors/bad mem boards
Help (3.7:21)
passing variables
UD/Intro to Shell Scripts, Part 5 (3.12:12)
passwords/BASIC
BAS (3.5:1)
PATH
UD/Introduction to Shell Scripts, Pt 1 (3.8:14)
pattern matching
UD/The grep Command (3.4:1)
permissions/root directory
Help (3.11:15)
positional parameters
UD/Intro to Shell Scripts, Part 5 (3.12:13)
PRECISION
BAS (3.8:7)
BAS (3.11:3)
Print Spooler/BASIC
FW/Moving Documents into Fortune:Word (3.3:8)
Print to File/Fortune:Word
FW/Using Prototype Documents (3.4:13)
printcap file
Special Print Fonts with Multiplan/Using the Wheels Argument (3.3:2)
Help/Altering the Printcap file (3.3:14)
Printer Features, Using them
Help/Altering the Printcap file (3.3:14)
profile
UD/Introduction to Shell Scripts, Pt 1 (3.8:14)
Program Exceptions/BASIC
BAS (3.3:1)
Program Size/BASIC
BAS (3.2:3)
project management
Project Management Sfw/MicroTrak (3.4:1)
prompt, UNIX
Help (3.11:15)
prototype documents
FW/Using Prototype Documents (3.4:1)
PS1
UD/Introduction to Shell Scripts, Pt 1 (3.8:14)
PS2
UD/Introduction to Shell Scripts, Pt 1 (3.8:14)
pwd command
Unix Basics (3.12:1)

Continued on next page

BAS–The BASIC Advisor | News–News from Fortune | MP–Multiplan | FW–The Fortune:Word Glossary
Help–/u/help | RR–Readers Respond | UD–The UNIX Directory | (3.5:11)–Vol. 3 Num. 5, page 11

Expand your Fortune® without spending a fortune.

New equipment costs big bucks. The Good Fortune Trader sells fully tested burned-in equipment, refurbished to Fortune® standards at prices that won't make waves with your cash flow. Call or write today for a listing of our specials.

THE GOOD FORTUNE TRADER

60 DAY WARRANTY

a division of Computer Services of Atlanta

(404) 392-9938

1534 Dunwoody Village Parkway/Atlanta, Georgia 30338

ATTENTION FORTUNE SYSTEM OWNERS

Paying for a new power supply ?
Fixing or upgrading your present one could save you hundreds of $$$$.

Repairing	(Zenith)	$ 75.00
	(Western Electric)	$ 95.00
Upgrading of Zenith to 250 watts		$ 300.00

Terminal supplies repaired also.

Repaired supplies warranted for 90 days.
Upgraded supplies warranted for 1 year.

Quantity discounts available. (Call for quote.)

David Oakley
POWER SUPPLIES "R" US
5916 Southwind Dr.
San Jose, CA., 95138
408-365-7678

Quarterly Report
News/2nd Quarter 1986 Results Reported (3.8:12)
News/4th Quarter 1985 Results Reported (3.2:2)
Quarterly Reports
News/1st Quarter 1986 Results Reported (3.5:10)
News/3rd Quarter 1986 Results Reported (3.11:2)
Rand editor
Help (3.10:16)
rc file/fix to bypass date
Help (3.9:10)
rdconf
UD/Bad Blocks/What to Do With Them (3.1:1)
rdconf command arguments
Help (3.12:7)
read command
UD/Intro to Shell Scripts, Part III (3.10:7)
reconciling your check book
MP/Check Ledger and Reconciliation (3.11:1)
Records Processing
FW/Basics of Records Processing, Pt I (3.5:1)
Release Levels/Fortune Sftwr
Current Release Levels for Software (3.6:18)
reminder system
FW/Tickler File Using Fortune Glossary (3.7:3)
repeat command/Records Proc
FW/Basics of Records Processing, Pt I (3.5:9)
REPT command
RR/Multiplan Tips (3.12:4)
rm
UD/A Safe rm Shellscript (3.1:2)
UD/Safe rm, Part 2 (3.3:10)
rm, safe -- caution
RR/Re: Error 33 in BASIC (3.4:16)
rolodex
Electronic Rolodex, An (3.11:1)
safe rm, part 2
UD/Safe rm, Part 2 (3.3:10)
searching through files/more
UD/Viewing Files with cat and more (3.7:6)
Sequencer, Bulletin
Electronic Bulletin Board/Using the Sequencer Subsystem (3.2:12)
shell programming
UD/Introduction to Shell Scripts, Pt 1 (3.8:1)
shell variables
UD/Introduction to Shell Scripts, Pt 1 (3.8:14)
sitename/Fort-to-Fort Copy
UUCP Fundamentals -- Part II/Configuring Fortune-to-Fortune Copy (3.6:7)
sleep
UD/Introduction to Shell Scripts, Pt 2 (3.9:15)
sort command
UD/The sort command (3.2:5)
sort command/Records Process
FW/Basics of Records Processing, Pt I (3.5:9)
sparing bad blocks
UD/Bad Blocks/What to Do With Them (3.1:6)
subtotals
BAS (3.12:9)
Support, BASIC
BAS (3.2:1)
SX Memory Expansion - 3.5 mg
News (3.10:12)
symbolic format
Multiplan Shortcuts (3.1:9)
Tape Expansion Cabinet/new
News (3.9:17)
TERM
UD/Introduction to Shell Scripts, Pt 1 (3.8:14)
Terminal problems
Tech Tips (3.8:10)
test command
UD/Intro to Shell Scripts, Part III (3.10:6)
test command for numbers
UD/Intro to Shell Scripts, Part III (3.11:12)
test, shell command
RR/Multi-volume Backup, Shell Comments (3.5:17)
Text editors
Help (3.10:16)
tickler file/F:W Glossary
FW/Tickler File Using Fortune Glossary (3.7:3)
Time and Billing/Legal
Users' Spotlight/How Lawyers Use Fortune Computers (3.7:1)
Tpool
BAS (3.9:1)
tr
UD/A Computerized Appointment Book (3.5:4)
true, shell command
RR/Multi-volume Backup, Shell Comments (3.5:17)
Typesetting/F:W Glossary
FW/Records Processing as a Typesetter (3.8:2)
UBSQ file
BAS (3.11:3)
UNIX
Book Review/A User Guide to the UNIX System (3.2:10)
UNIX Co-processor
News/1st Quarter 1986 Results Reported (3.5:12)
UNIX Shell Basics
Unix Basics (3.12:1)
until do done command
UD/Intro to Shell Scripts, Part 5 (3.12:11)
USENET
Linking Computers Together w/ UUCP (3.5:6)
USERFILE/Fort-to-Fort Copy
UUCP Fundamentals -- Part II/Configuring Fortune-to-Fortune Copy (3.6:7)
users' spotlight
Users' Spotlight/How Lawyers Use Fortune Computers (3.7:1)
Users' Spotlight/Non-Profits Use Fortunes (3.9:17)
uucp
Linking Computers Together w/ UUCP (3.5:6)
UUCP Fundamentals -- Part II/Configuring Fortune-to-Fortune Copy (3.6:6)
variables, passing
UD/Intro to Shell Scripts, Part 5 (3.12:12)
vertical scrolling
FW/Moving Around Fortune:Word (3.9:16)
viewing files on your screen
UD/Viewing Files with cat and more (3.7:5)
while do done loop
UD/Intro to Shell Scripts, Part III (3.11:13)
Windows, Fortune
Help (3.7:21)
windows/Multiplan
Multiplan Shortcuts (3.1:14)
workpage/Fortune:word
FW/Using Prototype Documents (3.4:12)
Wyse Terms/Memory
Tech Tips (3.8:10)
X3.64 Terminal Filter
News (3.6:4)
zap command
Free Software Highlights/D.C. Grab Bag (3.4:6)

Dear Fortune Owner:

As you have probably read in */u/fortune news*, Fortune Systems SX models are significantly more powerful and faster than the PS and XP models. If you have a PS or XP model system, you should take advantage of our special promotion on SX upgrades. For a limited time, and while our supplies last, we will upgrade your system, including the new SX Cabinet at no additional charge for the cabinet. With this special offer, you may choose any combination of the following upgrades and/or expansions:

- *SX Processor with Current Release of Multi-User For:Pro*
- *60 MB Internal Streaming Tape*
- *30 MB, 45 MB, 70MB High Speed Hard Disk*
- *6-Port Intelligent Asynchronous Controller*
- *Upgrade to Current Release of all Fortune Software*
- *Fortune:Windows (Included at NO Charge)*

Additionally, we can perform these upgrades within 24 hours of receiving your system. In fact, if you ship your system overnight on a Friday for Saturday delivery, we will have it back to you on Monday morning, all ready to plug in and run; not losing the use of your system for a single work day. And, we will issue you a full 90-day warranty on the entire system.

JFM provides services to Fortune Systems owners nationwide either direct or through our authorized Affiliated Dealers. If you haven't used our services, I invite you to discover JFM; you will be glad you did.

Sincerely,

Morris Frydman
President

Talk to us, then talk to our clients

JFM
BUSINESS
SYSTEMS
The Merchandise Mart
Lobby Suite 144
Chicago, Illinois 60654
Call (312) 644-0870

Authorized Fortune Systems Master Dealer

BACKUP AND RESTORE UTILITY

PROTECTION MADE EASY!

- o Back up all or selected user directories
- o Restore all or selected user directories or files
- o Format diskettes
- o Review backup volumes
- o Menu-driven, easily installed
- o Complete documentation

Also copies the files necessary to preserve your system configuration--a tremendous help if your system ever needs to be cold booted!

$150.00 Backup/Restore Utility
$200.00 Utility + 2 boxes 5 1/4" 96 TPI high-quality diskettes

For more information, call CTD at (313) 540-1282.

To order, make check payable to *Computer Trade Development Corporation.* Send to:

BACKUP/RESTORE UTILITY
Computer Trade Development Corporation
Suite 1386
Birmingham, MI 48010

CTD

Company:_____
Contact:_____
Address:_____
City:_____
State:_____ Zip:_____

Looking for Business Accounting Software that will conform to your Business?

Look to DSM

Applications

- Can be tailored to your organization's specific needs
- Economy of Standardization
- Follows Generally Accepted Accounting Principles (GAAP)
- Luxury of Adaptibility
- Data Base Incorporation

Software for:

- Not-for-Profits
- Construction
- General Business
- Property Management
- Steel Service Center
- Law Office Management

Hardware

Fortune Systems, NCR, AT & T, IBM

Call or Write
Data Systems Management, Inc.
277 Coon Rapids Blvd.
Minneapolis, MN 55433
(612) 780-5200
Dan Boorkman

DSM

All software was developed using PROGRESS, a powerfull relational data base which allows you to tailor the software to your specific needs.

PROGRESS

Dealer Inquiries Invited
PROGRESS is a trade mark of Data Language Corporation.

/u/fortune *news*
73 Jason Street
Arlington, MA 02174