



NewsLetter

August 1988

Copyright 1988, USUS, Inc.

All the News that Fits, We Print

Sam'l Bassett, Editor-in-Chief

William D. Smith, Associate Editor

Board of Director Nominations

Hurry! Hurry! Hurry! You still have time for submitting a nomination for the Board of Directors to the nominating committee. If you don't know anyone to nominate, consider nominating yourself! The next issue of the NewsLetter (Sept./Oct. issue) will be the election issue.

Orphan Software Project

We are compiling a list of UCSD operating system based programs. The list will serve several purposes. One: it will allow USUS to provide a current list of available software. Two: it will give USUS a base from which to investigate orphaned software (software that is no longer supported), with an eye to producing updates, support and/or utilities. Three: it will show which programs have the greatest need and/or distribution. These programs would be the best immediate candidates for USUS to consider in any effort to provide updates and additional features. Four: it will give USUS leverage with various software copyright holders if we can show that our organization has real people with an interest in their program.

If you own or use any programs based on the UCSD operating system, please give us as much of the following information as possible: program name, version number, place and date of purchase, hardware requirements, type of documentation provided, current support status (does anyone support this product? If so how well?) If shown in the documentation, on the disk, or on the screen when booting, include the program author, vendor and copyright holder with any addresses and telephone numbers given. Please also include your own name, address and telephone number along with any comments on your experiences with the program(s).

You can send written copy to Beverly Henderson, P.O. Box 1389, El Granada, CA 94018; or call her at (415) 355-0383 (there is an answering machine on the number). If you

prefer to use electronic mail, send your message to Alex Kleider 71515,447 on CompuServe.

Administrator's Speaks

by Hays Busch

If time permits, I will try to write this column for every Newsletter and give you some interesting information about USUS. Here's my first effort. Some statistics about current USUS Membership.

As of June 30, USUS has 509 active members.

Now, what class of members are they?

- * 3 are Institutional
- * 37 are Professional
- * 462 are General
- * 7 are Student
- * 435 are USA Members
- * 21 are Canadian Members
- * 53 are International Members

AND where are the International Members located?

- * 8 are in Switzerland
- * 8 are in West Germany
- * 6 are in Australia
- * 4 are in Belgium
- * 4 are in South Africa
- * 3 are in England
- * 3 are in Sweden
- * 2 are in Hong Kong
- * 2 are in Italy
- * 2 are in Japan
- * 1 each are in Bahama, China, Denmark, France, Holland, Ireland, Malasia, Portugal, Spain, and Taiwan.

AND what type computers do our members use with UCSD Pascal?

- * 135 (27%) use SAGE/STRIDE or Pinnacle
- * 119 (23%) use some form of APPLE 2 or 3
- * 80 (16%) use IBM PC, XT or AT (including one PS2 machine)

- * 42 (8%) use TI 99/4a
- * 93 (18%) use "other" machines from Atari to Zenith.
- * 39 (8%) did not indicate a machine.

It is my hope we can get an "incentive program" started soon which will reward you current members for each new member you can recruit.

More about that in an up-coming issue of the Newsletter.

Board of Directors Minutes (June 14, 1988)
by A. Hays Busch and Samuel B. Bassett

MINUTES OF THE MEETING OF THE BOARD OF DIRECTORS OF USUS, INC., HELD AT THE LAKE TAHOE GENERAL MEETING ON JUNE 14, 1988

This meeting was held at the CalNeva Lodge, Lake Tahoe, Nevada. It was called to order at 4:55 pm by Samuel Bassett, Chairman of the Board. Board Members Samuel Bassett, Jon Nevins, Robert Spitzer, and Eli Willner were present, which constituted a quorum for the conduct of USUS business. Board Member Harry Baya was absent. Since the meeting was an Open Meeting, a number of USUS Officers and Members were also present.

Robert Clark, USUS Treasurer advised the Board that his was the only authorized signature on the USUS account at Sovran Bank in Maryland. He suggested that new signature cards be obtained and that his signature and the signature of one other Officer of Board Member be authorized thereon. This would protect USUS in case Mr. Clark should be unable to sign USUS checks for whatever reason. After brief discussion, Mr. Willner offered the following resolution, seconded by Jon Nevins.

RESOLVED

- 1) That Mr. Clark obtain new signature cards from Sovran Bank. That his signature and that of Weber Baker (USUS President) be affixed thereon. That only one signature be required for any USUS check.
- 2) That the draft resolution required by the Sovran Bank for establishment of a corporate account be incorporated verbatim, as if it had been proposed and voted upon, as the official act of the Board.

By vote of "aye" the resolution was passed unanimously.

Eli Willner, Board Member advised the Board that he must resign as Editor of the USUS Journal. He indicated that the press of business was such that he did not have time to devote to the activity. After discussion, Mr. Willner offered the following resolution, seconded by Robert Spitzer.

RESOLVED

Mr. Willner's resignation as Editor of the USUS Journal be accepted with regret and thanks for past services. That Mr. Bassett, editor of the USUS Newsletter receive all editorial materials now in Mr. Willner's possession and that Mr. Bassett begin a search for a replacement editor for the USUS Journal from among the active USUS Membership.

By a vote of "aye" the resolution was passed unanimously.

Mr. Bassett asked for a motion to approve all minutes of Board of Directors Meeting held by electronic conference on CompuServe Information Services MUSUS Forum. There being no discussion, Mr. Bassett offered the following resolution, seconded by Mr. Spitzer.

RESOLVED

Minutes of Directors Electronic Conferences as listed in Data Library 8 of MUSUS are hereby approved without change.

By a vote of "aye" the resolution was passed unanimously.

Mr. Arley Dealey, Principal System Operator (SysOp) for MUSUS asked the Board of Directors to clarify the authority for this position. After considerable discussion, the following resolution was offered by Mr. Bassett, seconded by Mr. Spitzer.

RESOLVED

The Primary System Operator (SysOp) for MUSUS is Arley Dealey. He is the Chief Agent for USUS, Inc. for all matters relating to MUSUS and is the primary agent of the Corporation authorized to deal directly with CompuServe Information Services on all matters relating to MUSUS (ie., that under normal circumstances he is the only person so authorized). All Assistant System Operators for MUSUS are responsible to Mr. Dealey for activities relating to the conduct of MUSUS. Mr. Dealey is directly responsible to the President of USUS for all instructions and the execution of

duties of all the System Operators. All activities and instructions relating to MUSUS are subject to review by the Board of Directors of the Corporation, however, and disputes between SysOps and between SysOps and the President may be appealed to, and will be adjudicated by, the Board of Directors.

By a vote of "aye" the resolution was passed unanimously.

There being no further business to come before the Board of Directors, the meeting was adjourned at approximately 5:30 pm.

Respectfully submitted by:
A. Hays Busch, USUS Administrator
Samuel B. Bassett, Chairman, Board of Directors

Board of Directors Minutes (July 12, 1988)
by Samuel B. Bassett

MINUTES OF THE SPECIAL MEETING OF THE BOARD OF DIRECTORS OF USUS, INC., HELD VIA THE TELECONFERENCING FACILITY OF COMPUSERVE INFORMATION SERVICE ON JULY 12TH, 1988.

Present at the beginning of the meeting were:
10:08:07 PM EDT Tuesday, July 12, 1988

<u>User ID</u>	<u>Name</u>
71735,1776	Samuel Bassett, Chaircritter
73007,173	William D. Smith, Ass't SysOp
72747,3126	Robert Clark, Treasurer
70260,306	A. Hays Busch, Administrator

Joining the meeting at the indicated times were:

75226,3643	A. Robert Spitzer, Board Member	10:18:24	PM	EDT
76456,416	Jon Nevins, Board Member	10:44:24	PM	EDT

Matters dealt with were:

Miscellany:

William Smith reported that he had downloaded some questions about a "pretty-printer" program from MUSUS, and inquired as to their suitability for inclusion in the Newsletter, the Editor assured him of it.

Hays Busch stated that Howard Sweet was still interested in helping with the Newsletter and Journal, and the Editor agreed to work with Mr. Sweet directly.

Hays Busch inquired whether Eli Willner had sent any of the material he had collected for the

Journal to the Editor, and the Editor replied that he had not.

Bob Clark suggested that Directors who miss 3 successive Board meetings be summarily replaced, and the Chairman demurred, saying that THAT would require an amendment to the By-Laws that he (the Chaircritter) wasn't sure could be managed. He did admit that the idea had merit, however.

Hays inquired if anyone had had communication with the President, Weber Baker, recently, and no one had.

At this point in the discussion, Bob Spitzer logged in, and began taking part in the discussions.

William Smith suggested that, after the By-Laws were changed to set Directors' terms at 2 years, the two directors receiving the highest vote totals in the upcoming elections be seated for two-year terms, and the other three elected directors be seated for one-year terms. The Chaircritter admitted that he found the idea very interesting.

William then asked if the election materials should go out with the September/October Newsletter, and all agreed that they should.

At 10:44 PM EDT, Jon Nevins joined the conference, thus producing a Quorum of the Board, and the meeting was declared to be officially in session.

NOMINATING COMMITTEE

The Chaircritter asked that the Nominating Committee, consisting of Henry Baumgarten, William Smith, and Alex Kleider, be accepted as official. The motion was seconded by Jon Nevins and unanimously approved.

LIAISON NEGOTIATIONS

The Chaircritter moved that the Resolution posted to Section 8 of MUSUS, authorizing A. Robert Spitzer to negotiate with Liaison, Inc. for a contract for their services to USUS be adopted as written.

The Treasurer note that USUS' funds were becoming limited, and that the costs to USUS of such a contract should be examined closely.

Bob Spitzer discussed the costs, noting that the worst case should be that USUS would have to pay \$1995 the first year, and \$1200 every year thereafter, but pointing out that these "up-front" costs would be offset by the Corporate

Memberships of both Liaison Inc., and Pecan, and that the income from memberships (both General and Professional) generated by the linkage with Liaison (and the technical support involved) would handily cover that outlay.

The Treasurer reiterated his concern that the Board examine costs carefully before committing itself to the contract, Jon Nevins expressed a concern that we work hard at increasing professional memberships, and Bob Spitzer replied that 12 Professional memberships per year would pay for the yearly cost. He also stated that having the contract would mostly support Professional members.

There was some discussion of the possible parameters of the contract, but the Chaircritter expressed himself as being extremely unwilling to specify any monetary limits to Dr. Spitzer's authority to negotiate with Liaison, feeling that doing so would handicap negotiations needlessly.

Bob Spitzer stated that he would concentrate on positive issues in the negotiations, accentuating USUS' willingness to pay for a high level of technical support, primarily of professional members in a limited fashion, so as not to discourage individuals from establishing their own contracts with Liaison, when that was appropriate. He also stated that he intended to avoid discussion of other issues, especially negative ones, and that he would not be committing the Board and the Corporation in the negotiations, just establishing a contract for Board approval.

At this point, the Chaircritter, scenting a consensus, moved the issue to a vote, and it was approved unanimously by the quorum of Directors present.

PUBLICATIONS COMMITTEE

The Chaircritter moved that the Board establish a Publications Committee, consisting of at least Henry Baumgarten and Arley Dealey, to serve as a Journal Review Board, and to oversee the state of USUS publications in general.

After an early unanimous vote by the Directors present, discussion ensued as to the composition of the Committee. It was proposed that the Editors and prospective editors of USUS' publications be made members of the Committee.

The Chaircritter explained that what he had in mind was a Publications REVIEW Committee, similar to the IEEE Publications Review

Board(s), rather than a working committee of the people involved in the mechanics of actually publishing the material -- working toward the future possibility of a formally reviewed journal at some point.

Those present indicated that they saw the difference, but thought that getting publications published had first priority.

A second unanimous vote approved the formation of the Committee.

TECHNICAL SUPPORT COMMITTEE

The Chaircritter moved that the Board authorize the creation of a Technical Support Committee, composed of from 4 to 7 technically expert members who would serve as USUS' representatives in dealing with Liaison, Inc., and serve as a technical resource for other USUS members; he also moved that the Board seek appropriately qualified members to serve on the committee.

Bob Spitzer immediately volunteered to act as the Stride support member of the committee.

Jon Nevins stated that he would like to see the committee members be selected on a competitive basis. Bob Spitzer suggested that committee members serve at the discretion of the Board. William Smith suggested that only Professional members be considered for membership, which received wide approval.

Hays Busch opined that USUS should be very careful to appoint committee members who are experts on various machines -- especially the ones whose owners make up a large percentage of the membership, and not let the expertise be dominated by one or two machines, however technically exciting they may be in contrast to older machines.

The Chaircritter moved the issue to a vote, authorizing the committee, noting that Bob Spitzer was the first volunteer, and with an agreement to work out a detailed charter in the near future. As presented, the motion carried unanimously.

Hays Busch nominated Bob Spitzer as temporary chair, but the Chaircritter pointed out that the sole member of a committee is its de- facto chair.

MSDP

The Chaircritter moved that the Board authorize the creation of a Member Software Distribution Program (MSDP) Committee, with Alex Kleider

and William Smith as members, to formulate a policy of encouraging and facilitating software exchanges among members, and recommend ways in which this might be made lucrative for the members concerned and the organization, while being careful to assess any impacts on USUS in regard to both cash flow and non-profit status.

Jon Nevins immediately asked whether this committee would replace the Library committee, and expressed worry that the library might suffer from the emphasis and resources devoted to the MSDP proposal.

The Chaircritter opined that it shouldn't, in that the Library is essentially a free service to the members, and the explicit purpose of the MSDP was to provide a PAID service.

Bob Spitzer noted that programs can be sold at a profit, as long as the organization does not make a profit -- i.e. proceeds are used to forward the legitimate non-profit goals and activities of the corporation, such as better publications, more frequent meetings, etc.

William Smith opine that MSDP should be separate from the Library, as it meets a different need.

Hays Busch stated that there is no Library Committee as such, and that the Library is presently being worked on by the Apple and TI 99/4 SIG chairs, and is thus in fairly good shape, considering its neglect over the past few years.

Discussion continued, centering mostly on the necessity of preserving and encouraging the Library. The Chaircritter finally achieved an agreement to authorize the MSDP Committee, while directing the members to work out a charter for the Board's approval, addressing th concerns expressed in this meeting.

LIBRARY COMMITTEE

The Chaircritter proposed to investigate the formal authorization of a Library Committee, but it was thought that simply encouraging the work which was in progress, and prodding the Editors of the USUS publications to include more library-related material was sufficient, and the matter was dropped.

BY-LAW AMENDMENT

The Chaircritter noted that an Amendment to the By-Laws, authorizing two-year terms for Directors, was on the Agenda for this evening's

meeting, but considering the lateness of the hour (past midnight, EDT) and the fact that there were only three Directors present, that the matter should be deferred to a future meeting.

After agreeing that the next election for members of the Board should be part of the Sept/Oct Newsletter issue, all present agreed to meet again on Wednesday evening, July 26th, 1988 at 7/8/9/10 PM to take up the Amendment as the SOLE item of the meeting agenda.

The Special Meeting of the Board on CompuServe was adjourned at 12:20:43 PM EDT Tuesday, July 12, 1988, with agreement to resume on Wednesday, July 27, 1988 at 7 PM PDT / 8 PM MDT / 9 PM CDT / 10 PM EDT.

Minutes submitted by:
Samuel B. Bassett

Apple SIG Doings by Frank Lawyer

The kindly editor has graciously allowed us to continue a column of Apple oriented chit-chat, so on we go... Remember though, you have to help. Send in those ideas, articles, coding wonders..., whatever. Write me or call me if you have snippets. If you feel extra ambitious, send full articles or letters directly to the NewsLetter.

By now, all of you with interest in Apple machines should have received a separate bulletin which outlined where we were and what we were trying to do to advance the cause of the Apple SIG. This was an an expansion of the Apple SIG column in the last NewsLetter. Basically, anybody who had marked Apple as their main hardware interest in a membership form was made full honorary member of the Apple SIG, no papers to sign. Any others out there who would like periodic Apple bulletins, or a copy of the last one, send your name to me or to Hays Busch.

SUMMARY...

Just in case you didn't get a bulletin, here's what we covered. We have started weeding through the library disks, with a pretty good rate of conversion so far (see more below). A new Apple welcome disk has been completed, which has a list of UCSD p-System books, a list of Apple resources, a list of UCSD software available for the Apple2 series, a history of the UCSD p-System, etc. All of these items are still being revised and enlarged, and we expect the files to grow as time goes on. We started to communicate via this column, and the bulletin.

You can participate by digging in, sending questions, tirades, whatever. Just be constructive. It's a waste of time to complain if you have no suggestions. You can get a new welcome disk by sending me your old one or a blank disk. The first "official" new welcome disk was dated 07/01/88. Since then there were more revisions, and it is now dated 08/01/88.

PRAISE....

Robert Dell has graciously donated an enhanced Apple2e to USUS. Hays has it running, and I know he's using it. Robert also donated some software, although I don't know just what at this writing. The only real problem was that he could only LOAN the disk drives to USUS, so we need some disk drives and disk controller cards. Any donations? USUS may be able to pay for these, if the prices are reasonable. Contact Hays Busch. Robert is also combing through some of the later library disks to see what can be transferred to the new set. Sounds like he is making good progress too.

HELP !!...

Looking for help... Long ago and far away, SofTech published an applications catalog. This contained a list of software that ran under UCSD p-System. Does anyone out there have one or have access to one which could be xeroxed?

Apple Computer has been out with v1.3 of Pascal for a couple of years now, but no review has graced these pages. Anyone willing to tackle this?

Beverly Henderson is organizing the "Orphan Software" project. She is trying to get information on p-System software which runs on orphaned machines, so that USUS may contact the authors and try to come to some sort of agreement on support and distribution of this software. (See the Orphan Software Project article at the beginning of this NewsLetter.)

LIBRARY DISKS...

We are going to use this column to announce the availability of Apple Library disks. We are continuing the process of weeding thru the USUS library and re-compiling the programs to make sure they run on Apple. The minimum environment will be an Apple2e with an 80 column screen, 2 5-1/4 inch drives and a printer, running the 64K option of Apple Pascal v1.3. If we get feedback that this isn't acceptable to some, then we might lower the base machine. The

intent is to have nearly all the programs run in a minimum environment, and to carefully note programs which require additional facilities. Contrary to previous USUS policy, the disks will include CODE files. In that way, you should be able to run a program immediately. The source code (.TEXT files) will be there for those who wish to re-compile and/or change the programs. In some cases, the source files were large, and the compilation cumbersome. The code files will save you the trouble.

These disks are Apple specific. The old programs are preserved on archive disks, so they won't go away. No attempt has been made to write in anything which will only run on Apple. We have tried to preserve portability as much as possible. Lots of bugs have been fixed. My goal is to have one Apple Library disk available each month. That makes a long term project, but it is manageable in terms of time. If we can do better, fine.

Four disk volumes of the conversion are ready. These are programs converted from the old USUS volumes 1 to 4, recompiled and tested in "the minimum environment" (see above). So we know the programs will run. They can all use some improvement (what can't?), but that gives some of you the chance to fix them. I have sent the directory listings to the Newsletter with this column, so they may be elsewhere in this issue (*They are*). The complete disk directory, old and new, is available on the welcome disk, along with an order form. Because we now have these Apple specific disks, the order form is a little different than the one we usually see.

An Apple volume on a 5-1/4 inch disk is 280 p-System blocks. The new Apple disks will be floppies, one volume on each side, a total of 560 blocks. You may specify volume names in pairs for ordering. If you include an extra \$2, we will send you a new Apple welcome disk with your order.

The old disks can still be ordered, and will remain so for the foreseeable future. As disks are converted, the "DIR" file on the welcome disk will be updated to show it has been converted to new disks, so that potential buyers will be warned.

The new disk volumes will adopt a new naming convention:

1. The first three characters will be "APP" (for Apple).

2. The next character will be a "2" for the Apple2 series (except GS), "G" for Apple2GS, "3" for the Apple3 and "M" for Macintosh. I'm pretty sure that the Apple2GS is source format compatible with the Apple2, although the p-Systems are not. I'm hoping that the directory formats are compatible also.
3. The next character will represent the type of information on the disk. "P" for general programs, "U" for utilities, "G" for games, "I" for Information, perhaps "S" for USUS material. The idea is to leave room for other categories, and have a hierarchical structure.
4. The last two characters will represent a disk number in the series from 01-99. That will allow a lot of possibilities.

So disk "APP2U01" would be an Apple 2 format, containing utility programs, volume #1 of the utility series.

The idea of the information type disk is to preserve some of the hard copy which seems to have disappeared over time. These Apple SIG columns and any Apple SIG bulletins will go on an information disk after they have appeared in the Newsletter. That way any new members can get caught up with recent history just by getting the Apple welcome disk and an information disk. It will begin a repository. Also, there were some Apple oriented articles in previous issues of the NEWS and REPORT. If we could get machine readable ones, fine, otherwise maybe someone can retype them (perish the thought). Newsletter and Journal articles would only be added after they have been sent to the membership, this is not intended to steal any of their thunder. A lot of other Apple specific information can go into tip files etc. While some may object that the library should consist of programs, I believe the idea of a "library" is to make information available.

Disks APP2U01, APP2U02, APP2I01, and APP2G01 are announced ready for order as of 08/01/88. All requests for new Apple library disks will be filled by flippies, one 280 block Apple volume per side. We are able to price the disks just as we have them now, at \$6 for the

flippy. This will give the Apple people better value, since a lot of the second side was empty on the old format. An order will be able to specify any new Apple volume numbers in pairs. The order will be filled by copying the volumes one to each side of the flippy. I can see ahead that some things like the adventure game, may need 2 volumes to fit. Until we finish conversion, probably in about 12 to 18 months, we will honor orders for the old volumes in the old format. As old volumes are converted, their directories will be marked so that potential purchasers will know that they have been converted. You will probably be better off with the new disks.

Since there will still be some changes in the first volumes as programs are enhanced, and perhaps moved, we have worked out an incentive for disk purchasers. For a period of 6 months, the volumes are renewable. This means that for a period of 6 months from the announcement of availability either here in the Newsletter or in an Apple SIG bulletin, you may trade your old volume for an enhanced version (if any). We will announce when changes have been made. You will send us your old disk, and we will copy the new volume to it, replacing the old one. This offer applies only to the new series of Apple disks, not to the moldy oldies. You will also include \$1 to cover the return postage. We will assume we can reuse your mailer and your disk. The few extra cents is to give us a cushion, in case the resident elephant sits on your disk mailer. This policy is similar to many vendor enhancement policies. It will allow us a little more time to get the programs moved and working, and assure you that you can order now, and still get the final version when it becomes available.

FINALLY...

Well, I have more, and a few Q/A things, but I can see the kindly editor becoming impatient with his blue pencil, so I will leave it for next time.

Frank Lawyer, 126 Demott Lane, Somerset, NJ 08873; (201) 828-3616

New Library Disk Directories

What follows is a listing of the directories of the newly released Apple disks. These disks were developed by the Apple SIG. See the Apple SIG Doings column for more information on the naming conventions.

APP2I01:

UNITS.DOC.TEXT
 APSIG.01.TEXT
 APSIG.02M.TEXT
 DISK.INFO.TEXT
 REQUESTS.TEXT
 UNIVERSAL.TEXT
 USUS.NEWS.TEXT
 LIBR.J14.TEXT
 UCSD.HIST1.TEXT
 UCSD.HIST2.TEXT
 SU880705.TEXT
 COMAPP2I01.TEXT
 DIRAPP2I01.TEXT

Info on UNITS, SEGMENTS and EXTERNAL Routines
 Apple SIG column for Newsletter of 07/88
 Apple SIG bulletin sent to members 07/88
 Information on use, packing and shipping disks
 Requests for programs and routines needed for library
 Suggestions for some universal routines for library
 News of the formation and early doings of MUSUS
 Suggestion for a Library information program
 Part 1 of the UCSD p-System History
 Part 2 of the UCSD p-System History (in progress)
 List of files on first 7 old Apple disks, sorted by name
 Comments on the above files
 You're reading part of it

APP2U01:

COMBINE.TEXT
 COMBINE.CODE
 CRT.I.O.TEXT
 CRT.I.O.CODE
 FORMAT.DOC.TEXT
 FORMAT1.TEXT
 FORMAT2.TEXT
 FORMAT.CODE
 FORMAT.TEXT
 PRETTY.DOC.TEXT
 PRETTY.TEXT
 INITVAR.TEXT
 PRETTY.CODE
 COMAPP2U01.TEXT
 DIRAPP2U01.TEXT

A simple little thing to combine 2-4 text files
 Code file for above
 very powerful, crash-proof data entry UNIT for CRT menus.
 Code file for above. Must be linked and installed.
 Documentation (from Pascal News) for FORMAT
 part of FORMAT.TEXT (subfile)
 part of FORMAT.TEXT (subfile)
 CodeFile for FORMAT
 large, wonderful Pascal program pretty printer. (see doc)
 Documentation for both FORMAT and PRETTY
 The second Pascal pretty printer, from the Pascal News.
 part of PRETTY.TEXT (subfile)
 Code file for the above
 Comments on programs above
 You're reading part of it

APP2U02:

L.TEXT
 L.CODE
 SIMP.TEXT
 SIMP.CODE
 LINECOUNTR.TEXT
 LINECOUNTR.CODE
 PRIME1.TEXT
 PRIME1.CODE
 PRIME2.TEXT
 PRIME2.CODE
 DELETE.LF.TEXT
 DELETE.LF.CODE
 HEXOUT.TEXT
 HEXOUT.CODE
 SHELLMSORT.TEXT
 SHELLMSORT.CODE
 PERUSE.PG.TEXT
 PERUSE.PG.CODE
 COMAPP2U02.TEXT
 DIRAPP2U02.TEXT

A short but effective text printer with several options.
 Code file for above.
 A program to produce random text which sounds "official"
 Code file for above.
 Counts the lines of a textfile.
 Code file for above.
 Generates primes by the sieve method
 Code file for above.
 Generates primes via the division method
 Code file for above.
 Delete ASCII linefeeds from a textfile.
 Code file for above.
 Prints the equivalent hex code for each keyboard key
 Code file for above.
 Shell-Metzner sort for a textfile, 80 char lines
 Code file for above.
 Peruse a text file a page at a time on your CRT
 Code file for above.
 Comments on programs above
 You're reading part of it

APP2G01:

SIMP.TEXT	A program to produce random text which sounds "official"
SIMP.CODE	Code file for above.
BLACKJACK.TEXT	The famous casino game. Allows negative funds.
BLACKJACK.CODE	Code file for above.
CHASE.TEXT	Get away from the robots, but don't get zapped by the fence
CHASE.CODE	Code file for above.
WUMP.CAVE0.TEXT	A cave arrangement for WUMPUS game.
WUMP.CAVE1.TEXT	A cave arrangement for WUMPUS game.
WUMP.CAVE2.TEXT	A cave arrangement for WUMPUS game.
WUMP.CAVE3.TEXT	A cave arrangement for WUMPUS game.
WUMP.CAVE4.TEXT	A cave arrangement for WUMPUS game.
WUMP.CAVE5.TEXT	A cave arrangement for WUMPUS game.
WUMP.DOC.TEXT	Instructions for WUMPUS game.
WUMPUS.TEXT	Shoot the wumpus with a crooked arrow and watch the pits!
WUMPUS.CODE	Code file for above.
COMAPP2G01.TEXT	Comments on programs above
DIRAPP2G01.TEXT	You're reading part of it

NOTE: There are no program files or code files on an Information Disk. The files all contain text of some sort, and can be printed out on your printer, or scanned with the Editor. In general, all new Apple SIG related articles, bulletins, Newsletter articles etc. will go on these disks. Also, some older files which appear on the Library disks will move here if they are still interesting or germane.

LIBRARY NOTES

Any USUS member who wants one, can order a disk which has a full directory listing for each Volume of software currently available in the USUS Software Library. Disk formats that are available are as follows:

S A G E / S T R I D E / P I N A C L E	(1 2 8 0	block)
I B M	(6 4 0	block)
APPLE2	(280 block "flippy" disk,	560 total blocks)
TI 99/4a (SSSD "flippy" disk)		

These disks are copies of the "Welcome Disk" which is sent to all new USUS Members. In addition to the SW Library directories, the disks contain SW Order and Donation forms, information and order forms for PowerTools (where appropriate), name and address listings for USUS officers, directors and staff, and various other informative files.

These disks currently list EVERY program available. So not all the programs will run on every machine without modification. This is specially true for APPLE and TI 99/4a. (The APPLE and TI 99/4a SIGs are working to create "customized library disks" for these machines. If you have time to spare, they will welcome your help with this project.)

USUS could also use volunteers for SAGE/STRIDE/PINACLE and IBM formats who would review the programs and do an "editorial house keeping" job on the files for these machines. For machines not mentioned, hardcopy listing are available.

Cost for the Welcome disks are \$2.00 for USA and Canada, \$3.50 for International. This just barely covers disk, mailer and postage cost. The hardcopy listing costs \$5.00 and \$7.00 respectively because they cost more to mail and must be xeroxed.

Listings for all NEW USUS Software Library Volumes will continue to be published in the NewsLetter. But if you have lost your old listings, this is a way for you to come up to date. Just send your check, in US dollars drawn on a US bank, to the USUS address. Mark the envelope "Library Listings" to help expedite your order.

A History of the UCSD p-System*

by Frank Lawyer

(Copyrighted 1988 -- all rights reserved)

The language Pascal was designed by Niklaus Wirth of the Technical University in Zurich Switzerland in the late 1960s and first implemented in 1970. It was designed as a language to be used to teach good programming skills and program design. Languages which were in popular use at that time ran mostly on large mainframes, and were not readily available to students. Some languages which might otherwise have been used, lacked features which were important to educational use, such as the ability to structure programs. Wirth had in mind the need for the language to be sufficiently portable to be executed on a variety of hardware. Although Pascal was described in journals in 1971 and in use soon after, the "PASCAL - User manual and Report" was published by Jensen and Wirth in 1974. It remains the definition of Wirth's Pascal.

Pascal made possible the idea of "structured programming" which enables large programs to be constructed which are less prone to errors. It also facilitated Wirth's idea of "stepwise refinement" which allows the program design to start at a very high level, and be refined to increasing levels of detail, until the level of program code is reached. At that point, the program is essentially complete. This is sometimes called "top down design".

Wirth implemented several Pascal compilers. The first was written in 1970 using FORTRAN for a CDC batch machine. Its major claim to fame was that it became the base version for bootstrapping the following compiler, written in Pascal. The next one worked well enough to be the subject of 1971 journal articles. An improved syntax and formal definition was made in 1972 in a joint article with C.A.R Hoare. At this time, all the attention was in Europe, and since academics composed the major journal audience in the United States, no commercial interest was aroused.

Wirth was using a "pseudo-compiler" which instead of compiling to the host machine's native code, compiled to a low level "pseudo-code" or "p-code". This required an interpreter, written in the native machine language. But it enabled a portability concept. Wirth started to distribute the Pascal compiler to other interested universities, and shortly dialects began to spring to life.

Starting in 1973, the University of California at San Diego (UCSD), became interested in Pascal as a language for their introductory class on computer programming. This was a class common to many different degree programs, and taught concepts of structured programming. It was desirable to give all these students the best possible foundation in programming. The course used mostly programming examples based on mathematics, which was tough going for those without the mathematical background. ALGOL was popular for many systems programs at UCSD, and it seemed that Pascal, which was based on Algol, was a natural progression, since it included some I/O features that Algol lacked.

The first Pascal compiler at UCSD was written for a batch oriented Burroughs system. However, this made life somewhat difficult for the students. Most college batch systems of the day allowed only a few tries at compiling and testing per day, since the computers were expensive and were used mainly for the business of running the school. So Pascal was ported to a PDP-11, which allowed interactive compilation and testing. Soon the Pascal system was running on 10 PDP-11s, and was beginning to be ported to a number of smaller microcomputers. The 1976 introductory course used Pascal.

The movement to smaller independent microcomputers with differing architectures and somewhat limiting memory facilities, caused the development of the UCSD Pascal System*, and the UCSD p-System. In the p-System, an operating environment is provided for the user which enables file creation and handling, screen support and other support utilities. This same environment exists on all machines where the p-System runs, making it easy for the user to switch from machine to machine and still feel at home. They followed Wirth's lead by having the Pascal compiler produce p-code.

While Wirth's Pascal was adequate for the teaching environment, it was also recognized that Pascal would have to have certain extensions to cope with complex programs in various applications. Certain additions or extensions were made to Pascal to form UCSD-Pascal. Strings were added to allow the students to deal with non-numerical programs and examples. The building block approach to programming was facilitated by the development of "units", so that procedures could be developed and compiled independently, then called from a library. Disk I/O device drivers for random files were added.

Probably the most important concept was that of portability. The UCSD Pascal compiler and operating system were written in Pascal which compiled to a pseudo-code (p-code) for implementation either on physical hardware designed specifically for the purpose, or for a software implemented interpreter. This meant that the system could be moved to new hardware just by rewriting the p-code interpreter for the new hardware. The pseudo machine was a stack oriented machine with a number of registers. This design is relatively easily implemented on today's microcomputers. Since the rest of the system, utilities etc. had already been compiled to p-code, once the interpreter was written for a new machine, the whole system was ready to go. The effort to do this was on the order of a person-year, compared to many person-years to rewrite and compile the whole system to the native code of the new processor. It also saved considerable time in maintenance, since changes only had to be made to one high level system, and not to a number of different systems. Once written, the p-code interpreter for specific hardware could remain relatively static.

Professor Kenneth L. Bowles was head of the Institute for Information Systems group at UCSD. He was the sponsor and manager, and main driving force of the Pascal Project at UCSD. He wrote a number of books on UCSD Pascal, including a textbook used in the introductory course. Bowles guided UCSD Pascal through several versions.

In August 1977, UCSD began offering UCSD Pascal to users. There was a \$200 subscription fee, and documentation was available separately for a nominal charge. The first UCSD backed versions were for LSI-11 and PDP-11 processors, followed by 8080 and Z-80 processors.

By late 1977, the Terak Corporation in cooperation with UCSD, produced a UCSD p-System for their LSI-11 based Terak 8510/a machine. This was a graphics oriented machine, with 56K RAM and a floppy disk, video display of 320h X 240v pixels, and an integrated power supply. It retailed for \$7850, and software, including UCSD Pascal, was additional. This machine was up and running at UCSD before Christmas of 1977, and was being heavily used. There it was seen and used by representatives of BYTE magazine.

In March of 1978, Carl Helmers, the Editor and founder of BYTE magazine, met with Bowles and discussed the UCSD Pascal System in depth. Helmers became an exponent of Pascal, and especially UCSD Pascal. He recognized the need for a standardization in microcomputer operating systems, as code was already difficult to exchange because of the differences between micro operating systems and existing versions of BASIC. Pascal was already beginning to get some attention in the computer magazines at that time, but Helmers would give it a platform.

While the idea of standardization was beneficial to many segments of the relatively new microcomputer industry, Helmers also realized that some sort of standardized distribution of code could give BYTE an advantage, since they were already selling books and programs to hobbyists. An article by Bowles (written in February 1978) appeared in the May 1978 Byte magazine. This article made an appeal to readers of the magazine for help in standardizing the UCSD system. It described UCSD Pascal and the UCSD p-System. Bowles also asked readers to write to hardware manufacturers and request that UCSD Pascal be implemented on their hardware. Although UCSD had been quietly licensing to individuals for 2 years, BYTE ran a sidebar which told readers how to get a license. This article and some others which followed, made many people aware of the system. Interest began to spread rapidly. The BYTE articles ultimately produced over 1000 inquiries to UCSD about the system. At the time the first article was written in February 1978, the system was already running on 60 different pieces of equipment, but this represented only a few processors (CPUs) and brands of equipment.

The August 1978 issue of BYTE was devoted largely to Pascal. The cover featured a fanciful picture of "Pascals Triangle", a comic reversal of the Bermuda Triangle, where all the waters were calm and orderly, while ships were being wrecked on the "island of BAL" and the "COBOL straights". This issue was later referred to as the "Pascal issue", and was the second of Byte's August language issues. Bowles wrote another article for this issue, comparing Pascal to COBOL. UCSD was praised by several other writers.

In negotiating with potential commercial vendors for the system, UCSD took the position that the best way to port to a new machine was to rewrite the p-interpreter in the native code of the new

hardware. In that way, the minimum effort was involved, and the porting was easiest. Some firms were interested only in porting Pascal to their machines. After all, they already had an operating system for their hardware. However, it actually would involve more effort to rewrite the Pascal compiler. UCSD stuck firmly to the idea of porting the entire system, both as the easiest to implement for the vendor, and the best for the user.

UCSD p-System began to be licensed by the Regents of UCSD to hardware vendors. At the time of the first Byte article, Bowles said that implementations for new processors would probably not appear before October 1978. An 8085 implementation had already been achieved, and UCSD was in the process of bootstrapping to 6502 and 6800 machines. Machines already running were the Terak 8510A (above), An 8085 system from Northwest Microcomputer Systems, and a Z-80 system from Altos. In September of 1978, Bowles relayed to Helmers that Western Digital was developing a chip set which would directly execute p-code, and would be marketing a product the following January. The UCSD p-System would run unmodified. This innovation was called the "Pascal Micro Engine".

The next month, Bowles sent Helmers a memo of a plan to make the p-System available on an Apple2 computer. This was very important to UCSD, because Apple had already achieved a high profile, and was actively being sold by a dealer network. In fact, to many it had already reached the status of an "appliance". An Apple running p-System was envisioned to be available for about \$3000. Helmers expected this to set a standard for microcomputer systems, and he was right.

Early in 1979, NorthStar Computers offered a UCSD Pascal implementation for their existing Horizon Computer. This system was 8080 based. Their Pascal system sold for \$49, including documentation, and a 2nd disk containing an 8080/Z80 assembler and some Pascal utilities, sold for \$29. In January that same year, Western Digital started selling the Pascal MicroEngine. This was the p-System implemented in hardware, and ran 4-5 times as fast as equivalent software p-System interpreters of that time. An early 16 bit microcomputer, this machine included a complete UCSD Pascal Operating System based on UCSD release 3.0. Systems were available assembled, but in kit

form they were sold at a relatively low price to encourage designers to use the chip set.

The next NewsLetter will continue with the second in this series of articles.

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

O & / | A? ...

This column is provided so that you the members can ask questions (and answer the questions other members have asked) for which you don't know the answer. You can send a question (or answer) to the USUS address, attention editor, and I will include it in the next NewsLetter (also if it is an answer I will forward it on to the person asking the question, unless you indicate that you have already done so). Since most of the questions have a way of contacting the person who asks the question, please keep the editors informed of what is going on. This will enable us to keep the other members of USUS also informed.

All of the questions (and replies) in this NewsLetter were taken from MUSUS.

Tim Corica writes:

I am looking for a public domain "pretty printer" for Pascal with modifiable source code. I have some special needs related to a textbook project. Any help would be greatly appreciated.

Hays (our Administrator answers; CIS: 70260,306):

There are a number of "pretty printer" programs in the USUS Software Library. Most are rather old, but are reliable. By and large they are designed to format "text" written as program source material. (ie They create correct indentation, capitalization, etc.) Thus they represent the author's concept of correct programming style. All are directed toward formatting UCSD Pascal. They should also work with other Pascals, but I can not vouch for that. They are not "word processors" or "editors". There are a total of 36 volumes in the Software Library. Unfortunately these programs are on several disks, as opposed to one disk.

Raymond C. Weglein, Jr (1217 Ridervale Road, Towson, MD 21204; Phone: (301) 825-8278; CIS: 74140,14) writes:

Does anyone have a de-archiving utility in Pascal (UCSD)? There are many files and articles that I am interested in, but have no method of decoding them. Please point me in the right direction. Thanks

Eli Willner (CIS: 76703,500) replies:

Ray, most of the files on-line here (*on MUSUS Forum*) aren't archived; the primary reason is that arc files are common on PC-compatibles but not so common on other machines. What computer do you have? If PC, you can download ARCE (the shareware de-archiver) from IBMSW and many other Forums, use it to de-arc the source archive files you're interested in, and move source text from DOS to Power System using the DosFiler. There are also, I believe, programs with equivalent function on the Atari Forum. The CLM Forum had the complete source for SEA's archive utilities (in a UNIX C variant) on-line though they may recently have taken it off -- there is some legal maneuvering going on right now on that software. Given the source (which we have here) it shouldn't be that hard to adapt at least a de-archiver into UCSD Pascal (and shortly it'll be even easier to recompile the utility directly using Pecan C!)

David Sparkman (76347,2675) also replies:

I have seen a few articles on Huffman Encoding (one of several types commonly used in archiving), but have never tried to implement them due to the speed problems. Most archiving works on bit patterns which Pascal is not particularly suited for. (Yes it can be done but so slowly.) The better programs even drop out of C and into assembly to get decent speed. I personally use Dos to arc and de-arc (PkXarc is my favorite), and then use DosFiler to move things between Dos and the p-System. If you don't have a Dos Machine, you will need a friend that does! Once they are de-arc'd you can download them from the friend's PC.

Fred Carter (Intelco Corporation, 8 Craig Road, Acton, MA 01720; Phone: (508) 264-4485, 9am to 4pm Eastern Time; CIS: 70325,1100) writes:

Has anyone used the Power System to develop a PROM based application? Is it feasible to develop an application program and place it, along with the PME, into a PROM and have it start running when the power switch is turned on?

Specifically, I write programs for "intelligent" test equipment. Most of my work has been in

assembly language. I would like to use UCSD Pascal instead. I would be using the 6800 version (yes, you read that correctly, 6800 [NOT 68K]!) of the Power System.

My target system memory map is:
0000-007F I/O (Including an RS-232 port)
0 0 8 0 - 0 0 F F R A M
0100-7FFF RAM (Battery backed up.
Non-Volatile!)
8000-FFFF EPROM

In another message he further explains:

Each year millions of 8 bit computers are put into everything from automobiles to satellites. These are known as embedded systems. That is, often the user is unaware that a computer exists in the system and the program running is burned into a PROM (Programmable Read Only Memory) and is never changed.

I work for a company which makes test and measurement equipment which, as it happens, use Motorola 6801 microcomputers. With the addition of just 2 Integrated Circuit chips, we have a system with 32K bytes of RAM and 32K bytes of ROM.

Traditionally, instruments of this type have been programmed in Assembly Language. Sometimes this was done for performance, but usually it was done so the final program would be as small as possible. A good programmer can produce perhaps 1 new system per year this way.

So, wouldn't it be nice to program in a high level language like my data processing counterparts can? I think the answer is: Yes! About 9 years ago I ran across the UCSD p-System which, once adapted for your computer, would let you program in Pascal. I would now like to adapt the Power System to a new computer environment.

My environment has no disk drive, a "control panel" rather than a terminal but is otherwise similar to any other computing environment. I have two immediate problems for which I can use help.

The first is to create a P-Machine Emulator for the 6801 Processor. This processor is upward compatible with the 6800 Processor, but has some added 16 bit instructions from the 6809 Processor.

The second problem is the Assembler. How do I make a "6801.OPCODES" file? I haven't seen any information on adapting what is called the Adaptable Assembler.

Some other challenges come to mind. A suitable SYSTEM.PASCAL and SYSTEM.BIOS need to be made. While the BIOS is highly machine dependant, I would think that SYSTEM.PASCAL would need only small, if any, changes. Any thoughts?

Why consider the Power System environment? I have asked myself this question in regards to this project and have some answers. The first is the ease of editing, compiling and, when necessary, debugging in the Power System. A second is the concurrency primitives. In an instrument, the overall software design is greatly simplified if one thinks in terms of the various processes that the machine is to perform. In my case, there is a data acquisition process, a keypad interpreter process which defines the current processing wanted by the user, a calculation process, a display task and in the case of our battery operated equipment a low priority process which monitors the battery voltage. A multi-tasking (NOT multi-user) system significantly eases program design and development.

If I have intrigued you with my proposal for expanding the horizons of the Power System, I may be reached as given above, and thank you for your interest, moral support or help.

A reply from Eli (CIS: 76703,500):

I know that that's been done on a number of occasions: the one that comes to mind immediately is an intelligent terminal that is being controlled by a Power System application. In a typical application you'd probably want other parts of the runtime (like SYSTEM.PASCAL) to be PROM-based as well.

A reply from A. Robert Spitzer (CIS: 75226,3643):

Do you own the "Internal Architecture Guide"? I have one of the older (about 1982/IV.13 vintage) copies. It includes a definition of a P-machine emulator, a definition of the necessary BIOS, and a definition of the low level intrinsics required. I do not recall if the newer IAG also has this information. There is something called an "adaptable p-system", I believe. The idea, as I recall, was to give you all the tools you need to build your own implementation. Since I've never done this myself, I may be way off base. The machine level intrinsics are a p-code emulator; the p-codes were clearly documented in the IAG at one time. And you need some things like BlockRead and BlockWrite; I don't know what

the system needs in the way of clock interrupts, but it requires BIOS support for "attachable events" -- at least I think this is why Bill Bonham put that stuff in his BIOS in the first place. The Stride BIOS itself is a good model to study.

A reply from Verlene Bonham (MicroSage Comp. Sys. Inc; CIS: 75676,75):

MicroSage has been making stand-alone PROMS for 6 years now under the Power System. At first we (Bill Bonham) used the standard assembler. Later, with bigger PROMS, he wrote an assembler for the 680x0 family with a larger symbol table. Before that, at a company called Lynch Communications, our design team, Bill, myself, and 6 others, used the p-System to generate PROMS for terminals, switching systems, and many other telco applications. We have always felt the Power System is a good environment for generating stand-alone code.

Lee David Brauer (88 Grist Mill Road, Wethersfield, Ct. 06109; CIS: 76074,774) asks:

I have a problem with the Power System on an Apple IIGS. I am using a Grappler GS, which is a serial to parallel converter attached to the IIGS's serial port. There are controls for the printer which are accessible via DeskTop. If I interrupt the program to change the setting on the printer, and then return, the printer prints garbage although the program continues to run. I only have a 3 1/2 drive, so the IIGS System Disk with the DeskTop is not in the drive when the power system is running. Any ideas?

A reply form William Smith (CIS: 73007,173):

I have a couple of ideas. When you interrupt the printer, it might be in the middle of sending a command sequence. Changing the controls will send stuff to the printer which will be used by the printer to finish out the first command sequence, putting the printer in some unexpected state. The other idea depends on if the IIGS is like the Mac. On my Mac, the page sent to the ImageWriter is translated into a bit map and the bit map is sent to the printer. If the printer does not receive a command to turn off the bit map, characters received by the printer are treated as part of the bit map. Another explanation is that the printer expects to receive a certain number of characters in the bit map mode. Insertion of command sequences in the middle (of the bit map receive) by the control panel may throw of the alignment of the bit map thus displaying garbage.

Lee replies to William's reply:

I think I know what you mean. The interruption to go to the DeskTop would occur in the middle of a line. The control sequences are sent only once prior to the initiating of printing. The program in the DeskTop does send another control sequence. When printing is resumed, it bombs. I tried it again today. Even if printing is not interrupted, if the printer is reset from the DeskTop, when the program does try to print, it bombs. I think something in the BIOS of the power system must be affected.

We Get Letters...

The following letter was sent (May 30, 1988) by Frank Lawyer soon after he joined USUS. It contains his first impressions of USUS, some questions, and some suggestions. Due to a slight mix up, it has not been printed until now. Many of the things he has suggested, are now being done (some by Frank and some by others). If you have suggestions or want to volunteer to implement some of Frank's suggestions, please write.

Dear Editor:

I recently joined USUS and received my New Member package. I was pleasantly surprised that this took less than the four weeks I had expected. I think Hays Bush went out of his way to provide me a little something extra with my original inquiry, by sending me information on the USUS PowerTools.

I also liked the "welcome" disk that listed the contents of the library volumes. It provided an order form which can be printed out instead of having to rip the form out of the Newsletter, or xerox it. However, there are pricing discrepancies. The Newsletter is quoting \$8 per volume for Apple disks, and the order form shows \$7 per Apple volume. Since the date of the order form on the disk is more recent, I will use it, hoping I will save a few dollars. *(At the current time, the pricing for library volumes are \$6 each. \$2 is added for each additional volume on the disk (ie. 2 volumes on one disk is \$8). Also for 1.2MB and 3 1/2 inch disks, an additional dollar is charged to cover the extra cost of this media (cheaper in the long run since more library volumes can fit on a disk).)*

Hays (*our Administrator*) said that my package included the latest copies of the USUS News and Report and the Newsletter. I could not believe that the latest News and Report is over 2 years old, and the latest Newsletter is easily 6 months

old. The obvious reaction of a new member is a faint misgiving that they have wasted their money. If the publications of the organization are that far apart, it is going to be difficult to stay informed! I don't know what the planned Newsletter schedule was, but it would be nice to see one at least every two months. *(We are now trying for ten per year, one almost every month.)*

After reading the Newsletter several times, I can see that the main problem is a lack of material for timely publication. If I am going to do my share of complaining about the infrequency of the publications, I am also willing to pay the price of providing some letters etc. to help remedy this lack. *(Thanks Frank. As the readers can see from this issue, you are doing a lot.)*

Members need to sit down at their word-processors and provide the Editor with some more articles, programs etc. *(Yes, please do!)* I have been Editor of some publications in the past, and I know how frustrating it can be, waiting for some material to fill up that last page!

I think all new members are blessed with enthusiasm, otherwise they would not have become USUS members. Use that enthusiasm to provide some material. Don't be afraid that your efforts will be "too simple" or that you are a beginner etc. We are all beginners sometime, and others will benefit from your efforts. You can provide something that is unique, a new viewpoint. Sometimes those with more experience fail to see the fresh viewpoint of a beginner and fail to see some of the problems that a beginner can encounter.

Questions:

1. I see there is a fancy editor called Advanced System Editor (ASE). *(Pecan is now shipping this editor as the standard system editor with version IV.22.)* Where can I get a copy? *(Contact Pecan at (800) 63-PECAN)* Are there any reviews of this software available? Is it going to be worth my money? *(I use an earlier version of ASE and would never go back to the standard editor.)*

2. Are there minutes, handouts, or write-ups available from the National meetings? Some organizations make copies of Proceedings available. Does USUS? If not, is it a worthwhile idea? *(The only stuff which is available is what the members who attend the meetings write up and contribute to the Newsletter. Alex Kleider contributed the write*

up for the Tahoe meeting which appeared in the last NewsLetter. Hays BUSCH wrote up the Board of Director minutes which appear in this NewsLetter. At this time, USUS does not have a secretary to handle these things. It is being done by those who are committed and make the time.)

Some Suggestions:

1. As a new member, you feel as if you are coming in "in the middle of the movie". I would like to see a "Resource List". The contact list of the USUS Organization Board members and Officers is a start of this. I would like to see a brief history of USUS, some goals, what sort of help USUS needs, etc. This can include policies of the organization. Include some details on how MUSUS works and where to find topics there. In addition, I would like to see a list of available UCSD p-System based software, a list of vendors for same, prices etc. I would also like to see a list of UCSD Pascal reference books and material, with some possible comments as to their use for beginners, advanced, professional members. This could all be kept on disk as text files and distributed to new members just as the "welcome" disk is. If no such is currently available, then I will volunteer to start it.

2. I feel that the policy of obtaining USUS library software is somewhat restrictive in terms of signing agreements not to commercialize or to transfer to non-USUS members. Perhaps this is a wise policy, but there must be at least some of the programs which could be put in the public domain, and distributed to members without strings. There is a reasonable amount of UCSD public domain software available, yet none of it seems available through USUS. I think PD software could be kept on another set of library disks, so that there would be a minimum of conflict. PD UCSD p-System software could be donated with a simple form stating that the donator contributes this as public domain material. It could even be offered (Heresy!) to non-USUS members. It is a possible way to get others interested.

3. USUS needs more Public Relations. While going through some Public Domain UCSD Pascal disks, I found some of the early stuff done in 1980 by Jim Gagne, starting an organization for those interested in UCSD software. After reading several files, I began to think that an organization for UCSD Pascal might exist. Then I had to do homework until I located USUS. I went through my books until I found information

on USUS in the Washington Apple Pi book "Perfect Pascal Programs". Lucky find. If all the members have to go through this much trouble, the organization will grow slowly, if at all. I suggest that members write letters to the general purpose computer magazines and tell about USUS. If we have some general purpose words describing USUS benefits, use those. (If they exist, we can add them to the Resource List). If enough money is in the treasury, some ads might be appropriate in selected publications.

4. I'm sure there are good reasons why there has been more than a 2 year gap in the publication of the USUS News and Report. Available funds probably is a key factor. However, since this seems to be the official way of presenting the minutes of the Board, I find myself as a new member unable to figure out what the BoD has done for the last two years! The minutes I read were complete (and even interesting), but a gap of two years does nothing to help my understanding of USUS. I think most new members who join an organization like to just hang around for a short time and keep their eyes and ears open while they figure out what is going on. The lack of minutes doesn't help anyone to do that. I suggest that the missing minutes be printed in the Newsletter, which appears to be available more often. (*Starting this issue, all BoD minutes will be published in the next issue of the NewsLetter after the meeting.*) This will inform the members on a more timely basis, as well as provide more material for the Newsletter, which will then appear more regularly etc. This does not preclude also printing Board minutes in the USUS News and Report, it just makes them available to the membership on a more timely basis.

5. I feel a certain sense of "having to go on CompuServe". While I realize that MUSUS provides a good way to do certain things, I felt it was an option, not an obligation. It should be the business of USUS to provide information to the members in a timely manner through the publications and meetings, not conduct so much business on MUSUS that a USUS member is forced to go there to look for what is currently going on. While I feel that having Board meetings via teleconferencing on MUSUS is a fine idea, the obligation to provide timely notice of the minutes and other news remains. I will try to participate in USUS without MUSUS. This may prove impossible (*it did, Frank's on*

MUSUS almost every day now), but that remains to be seen.

6. I think the Newsletter could publish some personals ads of members looking to buy or sell hardware or software. Perhaps the ability already exists and there just weren't any in the last Newsletter? Why not publish rates for personals and "commercial" ads right in the Newsletter instead of making it a mystery? *(This will be done in a later issue. Currently, the rate for an unclassified ad is \$0.30 a line, 4 lines minimum. A line is the width of a column in this NewsLetter (about 45 characters). The ad will be boxed. You may use *bold* and underline, which will appear as bold and underline.)* We all realize that rates may change due to circumstances.

7. I can see from the letters in the last Newsletter (Winter 1987-88) that new members always seem willing to help. I hope that they will have the opportunity. From what I have seen so far, few new members who volunteer are given any direction. Only a few are not only going to volunteer but also aggressively pursue something to do. Maybe someone should volunteer to lead the volunteers?

8. How about some incentives? Everyone likes incentives. Perhaps the Newsletter might award a library disk of your choice (you'll still have to sign the agreement) for the best article or program each issue. I don't see any incentives for contributing to the library. How about a library disk of your choice for a program which was accepted? A free disk won't cost USUS much, but may encourage more participation.

9. While it is nice that 36 volumes of software are currently available, there seems no discount for anyone wanting to purchase a number of these disks. I don't disagree with the pricing for a few disks, but it gets pretty pricey if I want to get 15 or 20 disks. How about some volume discounts? The packing and shipping are pretty much the same for 5 disks as 20, and if you have to wait 4-6 weeks to get your disks, then a little bit of money could be saved by sending them at book rate. A few extra days won't hurt.

Some of my early concerns may prove to be unfounded or merely poor perception on my part. I hope so, because I would like to see USUS thrive and grow.

Sincerely, Frank Lawyer

MUSUS Tips

by William D. Smith, Ass't SysOp

In the last NewsLetter, figure 1 was given as an example of the way to set up a message file to put messages on MUSUS. Since then I have found that there is a faster (read cheaper) way of do this.

Fig. 1 New message file template

```
COMPOSE
<your message>
/EXIT
POST or MAIL for EasyPlex
<name> or ALL if for all members
<subject> 25 chars max
<section number> not needed with MAIL
Y
```

The following two figures illustrate this point (figure 2 is for **POST**ing a message to MUSUS and figure 3 is for **MAIL**ing a message via EasyPlex). The speed up comes from putting the section number (only for posting messages), the name of the recipient, and the subject on the same line as the **POST** (or **MAIL**) command. This keeps CIS from prompting you for this information and thereby saves some connect time (since CIS does not send the questions asking for the information or the menu asking for the section number).

Fig. 2 New message file template

```
COMPOSE
<your message>
/EXIT
POST SEC:<sec#> TO:<name> SUB:<subject>
```

Fig. 3 Mail from forum file template

```
COMPOSE
<your message>
/EXIT
MAIL TO:<name> SUB:<subject>
```

If you do not put all the information on the **POST** (or **MAIL**) command line, CIS will ask for the information which is missing.

Important, if all the information is on the **POST** (or **MAIL**) command line, CIS will not ask for a "Yes" prompt at the end (last line of figure 1). CIS will ask for the "Yes" prompt if it has to prompt for any of the information. This means, that if all the information is on the **POST** (or **MAIL**) command line, you do not get a chance to review and change it (but if you are uploading it as a file, you wouldn't be changing it anyway).

Note that when you **MAIL** a message, you can not and must put in a **SECTION** command.

There must be a space before the **SEC:**, **TO:**, and **SUB:** commands. There must not be any spaces between the command and the ":" and no spaces after the ":" and before the first character of the command value. You should be nice and include both the name and PPN of a recipient. The subject can not be more then 25 characters long.

Some other notes on using MUSUS. When you reply to a message, keep in mine the section and subject of the original message. If they, seem wrong in the context of your reply, you can change the section and subject (but not the name of the recipient) by using the **SEC:** and **SUB:** commands on the post command line as in figure 4.

```

Fig. 4 Reply, changing section and subject
REP <message #>
<your message>
/EXIT
POST SEC:<new sec #> SUB:<new subject>

```

Also use the **POST UNF** command when you are sending source code or tables. The **UNFormat** command keeps CIS from reformatting the output to fit on the readers terminal. Keep in mine though that all users do not have 80 column terminals, so only use it when you have to.

Tips

by William D. Smith

This column is rather short this month because of lack of time (I'm leaving on vacation before the end of the month and the NewsLetter has to go in the pipeline by then).

The tips this month consist of four ASE macros. It has come to my attention that not all members of USUS know what ASE is, so a little side trip. Originally when the p-System was first released, it had the first screen oriented editor that I ever worked with (this was back in 1980, and this sold me on the p-System). Later a company (now out of business) called Volition Systems released an editor they called the Advanced System Editor, ASE for short. This editor provided all the power of the regular system editor (soon to be known as the standard system editor) and then some. Pecan, who bought the p-System from SofTech, also eventually bought ASE from Volition. ASE is now the standard

editor shipped with most of Pecan's operating system products.

One of the things that made ASE so great was its macro facility which is what I'm talking about when I say ASE macros. This facility allows the user to record keystrokes from the keyboard or takeup an ASE macro into a macro (function) key. (ie. you can type one key to do a lot of work, especially repetitive tasks.)

I use this first macro to scroll the screen up by half a screen.

```

ls~      |{ set space character }
|'ScrollUp'|  |{ set function key name }
lh       |{ home cursor }
34|d     |{ go down 34 lines }
11|u     |{ go up 11 lines }
b        |{ go to beginning of the line }
|.

```

The vertical bar "|" prefixes a command to the macro takeup procedure. "ls<char>" will replace all <char>s found later in the macro with a space, and ignore all spaces in the macro. "|<macro name>" sets the function key name that you see when you use the "?" to cycle thru the editor prompts. "lh" is the command for moving the cursor to the home position of the screen (same as if you had enter the <home> key for your system. "34|d" moves the cursor down the screen 34 lines. On a 24 line screen, this means the screen will scroll up 11 lines. "11|u" moves the cursor up the screen 11 lines, form the last line on the screen leaving it in the middle of the screen. "b" move the cursor to the beginning of the current line. All macros must end with "|.". Repeatedly typing this function key (after it has been taken up) scrolls the screen up half a screen at a time. If your terminal has other then 24 lines, the repeat counts must be changed to have this macro work correctly. The editor allows 23 lines of text on a 24 line screen. 11 is half of 23 rounded down. 34 is 11 plus 23.

This second macro works the same, except that the screen is scrolled downward.

```

ls~      |{ set space character }
|'ScrollDown'| |{ set function key name }
lh       |{ home cursor }
11|u     |{ go up 11 lines }
11|d     |{ go down 11 lines }
b        |{ go to beginning of the line }
|.

```

Also note that this macro only works if your system has <InsLine> defined. <InsLine> is the character sequence which is sent from the computer to the terminal causing the terminal to insert an empty line at the cursor and push all lines from the cursor to the end of the screen downward.

This next macro inserts an empty, left justified line before the line the cursor is on.

```

|s~      |{| set space character }
|'InsLine'|{| set function key name }
|b       |{| go to beginning of the line }
|lb      |{| go to end of previous line }
|i       |{| Insert mode }
|ln      |{| a return }
|l1      |{| move to column one }
|le      |{| accept insert }
|ln      |{| go to beginning of next line }
|}
|.

```

When I write programs, I tend to begin each procedure on a new page. This is done by inserting the compiler directive "{ \$P }" on the line before each procedure. Also my screen will show the same number of lines as I normally print out. This next macro is used to bring the next procedure on the screen after the cursor to the top of the screen.

```

|s~      |{| set space character }
|'UpTop'|{| set function key name }
|lg$     |{| G(etCh "$" ) }
|u       |{| U(pTop )|.

```

G(etCh is a command which is not present in the standard editor. It moves the cursor from its current position on the screen to the position of the next character specified in the G(etCh command. U(pTop redisplay the screen so that the line containing the cursor appears at the top of the screen. This command is also not present in the standard editor.

To use any of these macros, type them into a file. Place the cursor at the beginning of the macro. Type the key you have defined as the <takeup> key and the type the function key you wish the macro to go into. After that, as long as you do not quit the editor or redefine the function key, you can type the function key to execute the macro.

By the way, if you have any questions about ASE, please send them and I will try to answer them. Also, if you are new at learning ASE, and

run into a problem and spend some time finding a solution, share your problem and solution with the other USUS members thru the NewsLetter.

From the Associate Editor

by William D. Smith

Well we have now come to the end of another issue. Tis getting easier doing the layout with each issue (experience I think). I would like to thank Frank Lawyer for the time he has spent writing articles and letters for this NewsLetter. We need more members who will spend some time helping make this organization successful again. The next issue of the NewsLetter will be arriving in your mail near the beginning of October. The deadline for submissions is September 16th. Don't forget that elections are coming soon. **Nominate someone, even yourself!**



USUS

NewsLetter

August 1988

Copyright 1988, USUS, Inc.

All Rights Reserved

Sam'l Bassett, Editor-in-Chief

William D. Smith, Associate Editor

<u>Page</u>	<u>Article</u>
1	Orphan Software Project
1	Administrator's report by Hays Busch
2	BoD Minutes (June 14, 1988)
3	BoD Minutes (July 12, 1988)
5	Apple SIG Doings by Frank Lawyer
7	New Library Disk Directories
10	A History of the UCSD p-System
12	Q & / A? ...
15	We Get Letters...
17	MUSUS Tips by William D. Smith
18	Tips by William D. Smith
19	From the Associate Editor William D. Smith
End	You're reading it

NOTE: The USUS Administrator, who handles almost all of our business, and especially new memberships and renewals, is Hays Busch. He can be reached at the USUS address in La Jolla, or at 2193 Montane Drive East, Golden, CO 80401-9125; (303) 526-0057.

Next NewsLetter coming September/October
Articles due by September 16, 1988

**USUS
P.O BOX 1148
LA JOLLA, CA 92038**

ADDRESS CORRECTION REQUESTED