CONTENTS

Winter 1985 USENIX Technical Conference .............................................................. 3
Call for Papers ......................................................................................................... 4
USENIX-Sponsored Workshops: Status Report ....................................................... 6
  UNIX and Distributed Systems Workshop; September 12-14, 1984 ....................... 6
  Communications and Networking Workshop ......................................................... 6
  UNIX and Computer Graphics Workshop ............................................................ 6
The UNIX Magician’s Handbook ............................................................................ 7
  The UNIX Wizard ................................................................................................. 7
  Inside Look at UNIX ......................................................................................... 7
  Tools .................................................................................................................. 8
  Other Examples ................................................................................................. 9
  UNIX Principles ............................................................................................... 10
  The File System .............................................................................................. 10
The 1984 Summer USENIX Convention Report ................................................... 11
Is This System Mannerist or Rococo in Style? ......................................................... 13
Trivia Quiz Answers ............................................................................................ 19
Local User Groups .............................................................................................. 23
USENIX Conference Proceedings Available ....................................................... 24
Thanks ................................................................................................................. 24
Tax Exempt Status Granted to USENIX ............................................................... 25
USENIX 4.2BSD Manuals: Status Report ............................................................ 25
4.2BSD Manual Order Form .................................................................................. 27
NOTICE

;login: is the official newsletter of the USENIX Association, and is sent free of charge to all members of the Association.

The USENIX Association is an organization of AT&T licensees, sub-licensees, and other persons formed for the purpose of exchanging information and ideas about UNIX and UNIX-like operating systems and the C programming language. It is a non-profit corporation incorporated under the laws of the State of Delaware. The officers of the Association are:

President Alan G. Nemeth
Vice-President Deborah K. Scherrer
Secretary Lewis A. Law
Treasurer Waldo M. Wedel
Directors Thomas Ferrin
Steve C. Johnson
Lou Katz
Michael D. Tilson

The editorial staff of ;login: is:

Technical Editor Brian Redman  [ucbvax|decvax!usenix!ber]
Copy Editor Michelle Peetz
Managing Editor Tom Strong  [ucbvax|decvax!usenix!tom]
Editorial Director Lou Katz  [ucbvax|decvax!usenix!lou]

Member services are provided through the Association office. Membership information can be obtained from the office:

USENIX Association
P.O. Box 7
El Cerrito, CA 94530
(415) 528-8649
[ucbvax|decvax!usenix!office]

Members of the UNIX community are heartily encouraged to contribute articles and suggestions for ;login:. Your contributions may be sent to the editors electronically at the addresses above or through the U.S. mail to the Association office at the address above. The USENIX Association reserves the right to edit submitted material.

;login: is produced on UNIX using troff and a variation of the —me macros. We appreciate receiving your contributions in n/troff input format, using any macro package. If you contribute hardcopy articles please leave left and right margins of 1" and a top margin of 1 1/2" and a bottom margin of 1 1/4". Hardcopy output from a line printer or most dot-matrix printers is not reproducible.

Acknowledgments

The Association uses a VAX 11/730 donated by the Digital Equipment Corporation for support of office and membership functions, preparation of ;login:, and other Association activities. It runs 4.2BSD, which was contributed and installed and is maintained by Mt Xinu. The VAX uses a sixteen line VMZ-32 terminal multiplexor donated by Able Computer of Irvine, California.

This newsletter may contain information covered by one or more licenses, copyrights, and/or non-disclosure agreements. No reproduction of this newsletter in its entirety or in part may be made without written permission of the Association.

\footnotesize
1UNIX is a trademark of Bell Laboratories. 2VAX is a trademark of Digital Equipment Corporation.
Winter 1985 USENIX Technical Conference

Pre-Announcement

The USENIX Association will hold its 1985 Winter Conference in Dallas, Texas, January 23—25, 1985. The USENIX conferences are dedicated to fostering the development and communication of research and technological information and ideas pertaining to UNIX® and UNIX-related systems. This announcement provides early information about the dates of events as well as persons to contact for further information. A pre-registration packet containing detailed Conference information and hotel reservation forms will be mailed in late November.

Meeting Headquarters: Fairmont Hotel
Ross at Akard
Dallas, TX 75201
(Reservation forms to follow in pre-registration packet)

USENIX Tutorials: Wednesday, January 23
All tutorials will be of a technical nature with a greater focus on advanced topics. Some of the tutorials may include licensed material. These will only be open to persons associated with a USENIX Institutional Membership. Verification of licenses and association may be required.

USENIX Technical Sessions: Thursday and Friday, January 24—25
Emphasis will be on highly technical sessions of an interactive nature. Panels and semi-formal workshops will be presented, as well as formal papers.

Abstract Deadline: November 16, 1984
(See the Call for Papers)

Concurrent with the USENIX Technical Conference, /usr/group will be holding their UNIFORUM tradeshow/conference at the Infomart, January 21—25. Continuous free shuttle bus service will be provided between USENIX hotels, /usr/group hotels, and the Infomart. For further UNIFORUM information, contact /usr/group at 408-986-8840.

If you did not receive this announcement directly and wish to be on the mailing list for receipt of the pre-registration packet, or if you need further information, please contact:

USENIX Conference Office
P.O. Box 385
16951 Pacific Coast Highway
Sunset Beach, CA 90742
213-592-1381
213-592-3243

UNIX is a trademark of AT&T Bell Laboratories
Call for Papers

Winter 1985 USENIX Technical Conference
January 23–25 (Wednesday — Friday)
Fairmont Hotel, Dallas, Texas

Abstracts due November 16, 1984

Program Chair:
Charisse Castagnoli

Program Committee:
Tom Ferrin
Steve Johnson
Lou Katz
Lewis Law
Alan Nemeth
Deborah Scherrer
Waldo Wedel

University of California, San Francisco
Bell Laboratories
Metron Computerware, Ltd.
Harvard University
Prime Computer, Inc.
mt Xinu, Inc.
NBI, Inc.

This conference will be slightly shorter than previous conferences, with an emphasis on sessions of a highly technical and more interactive nature. To encourage interchange between attendees and speakers, we plan to schedule panels and semi-formal workshops as well as formal papers. Suggestions for topics and speakers for these are being solicited. Because of the brevity and interactive nature of this conference, no proceedings will be published.

Suggested topic areas include, but are not limited to:

• Kernel enhancement, performance, real-time modifications
• Networks and distributed processing
• Hardware: evaluations, modifications for UNIX, design issues for supporting UNIX
• Operating systems architecture
• Programming languages and environments
• Mail, news, and uucp: transport systems, addressing, routing, user interfaces, legal issues, USENET problems and solutions
• Computer graphics
• Applications: CAI, system management and monitoring, real-time projects, novel database systems approaches
• System comparisons: 4.2BSD, System V.2, 2.9 BSD, etc.
• Standards and portability: kernel, commands, library, languages
• Future directions: security, standards, trends and predictions
Abstracts of papers should explain what is new and interesting about the work and should be between 300 and 1000 words. Proposals for panels and workshops should explain the proposed thrust of the discussion and contain suggestions for at least two members. All submissions must include:

- Title of presentation, panel, or workshop
- Full name of author
- Institution or company
- U.S. mail address
- Network address, if available
- Phone number
- Special requirements: audio-visual, room size, etc.

**Schedule:**

- Abstracts due: November 16, 1984
- Notification of acceptance: December 31, 1984

Abstracts should be submitted to one of the addresses below, either by U.S. or electronic mail. Any electronic mail submissions received will be immediately acknowledged, so try again if you do not receive an acknowledgment.

Charisse Castagnoli  
Teknekron Infoswitch Corp.  
1784 Firman Drive  
Richardson, TX  75081

allegra!convex!infosw!lusernix  
convex!infosw!charisse@rice.arpa
USENIX-Sponsored Workshops: Status Report

UNIX and Distributed Systems Workshop; September 12-14, 1984

This was the first small and informal workshop style meeting that USENIX has ever organized. There were approximately 100 attendees at the meeting and each attendee submitted a position paper that was reviewed by a program committee before the meeting. Most attendees regarded the workshop as clearly worthwhile and something that should be repeated again in the future, but it is also clear that there is plenty of room for improvement. There were several organizational problems with the particular hotel chosen for the event.

Communications and Networking Workshop

This workshop has been postponed until a future date. An announcement will be made in an upcoming issue of :login: when specific dates and a location are known.

UNIX and Computer Graphics Workshop

December 13—14, 1984
DoubleTree Hotel
Monterey, CA

USENIX is sponsoring a limited-enrollment workshop on current and future developments in interactive computer graphics under UNIX, including:

- Large scale graphics databases
- UNIX as a graphics development environment
- Future developments and directions.
- Real-time implementations
- High speed data transfer

The workshop will be structured to facilitate in-depth discussions of technical issues, and will have presentations in a number of formats, with ample time for questions and responses. There will be a computer graphics film and video presentation at a banquet.

In order to cover the extra expenses entailed in providing high quality visual presentations, the registration fee will be $200, which will include a reception. The hotel rate for this conference is a special $65/night for either single or double occupancy.

To ensure active participation by all attendees, applicants must submit a brief statement of qualifications and areas of graphics expertise. For further details and application information, contact the Program Chair:

Reidar J. Bornholdt
Room 7-444
Columbia University
College of Physicians & Surgeons
630 West 168 Street
New York, NY 10032

[harpo|cmcl2]!cucard!reidar or [ucbvax|decvax]!userix!reidar

The deadline for applications is November 1, 1984.

Program Committee:

Reidar Bornholdt, Columbia University, Chair
Lou Katz, Metron Computerware
Tom Duff, Bell Laboratories
Peter Langston, B.C.R
The UNIX Magician's Handbook

Elizabeth Bimmler*
Murray Hill, NJ 07974

ABSTRACT

The UNIX Wizard

UNIX† is traditionally taught by 'wizards.' Every installation, and there seem to be well over 3000 now, inevitably comes with its own set of gurus where UNIX freshmen can learn the art of UNIX programming.

Until recently, the prospective UNIX programmer had to find a guru and seek to become his apprentice. After a long training the student could then become a master and as a token of his excellence would be issued the superuser password. UNIX, to be sure, is not a trivial system, and as Kernighan and Pike note in the preface to their book: "as the UNIX system has spread, the fraction of its users who are skilled in its application has decreased."

The UNIX operating system has been acclaimed for its conciseness and structure, its portability, and perhaps more important still: for its availability. Commenting on the 6th Edition UNIX, John Lions made the following, often quoted, observation: "the whole documentation is not unreasonably transportable in a student's briefcase." As Kernighan and Pike have aptly countered in their book: "this has been fixed in recent versions." Considering that the kernel of the first edition UNIX was only 8K in size, on the average, the size of the system has more than doubled with every new edition issued. In comparison to other commercially available operating systems, the documentation that comes with the newer versions is still modest in size, but it certainly will not fit a student's briefcase anymore, unless, of course, it's left on the magnetic tape.

UNIX was necessarily taught by wizards, simply because those wizards never wrote down their art. Now that the UNIX system has become so popular the number of publications on UNIX is steadily increasing, and a small set of textbooks has finally appeared. In some cases these textbooks convey little more than the information already available from the system documentation, though perhaps presented in a more friendly way. Oddly enough, the people responsible for the development of the UNIX system have long hesitated in providing us with good quality textbooks on their system. "Well," said Rob Pike, "if somebody is going to do it anyway, why let outsiders do a bad job when insiders can do a bad job ...."

Inside Look at UNIX

Two textbooks on the UNIX system written by the 'insiders' have now appeared, and they haven't done such a bad job at all. A first book was written by Steve Bourne,‡ the author of the 7th edition UNIX 'shell' (the command interpreter). Bourne's book gives an excellent overview of the system, with one of the best introductions to the C language and UNIX system programming I have seen so far.

The second book, and the one we will consider more closely here, was written by Brian W. Kernighan and Rob Pike.

* Disclaimer: Ms. Bimmler is a colleague of the authors of the book reviewed. However, the ideas expressed herein may not be credited to Messrs. Kernighan or Pike.

Brian Kernighan is a long-time member of Bell Labs Computer Science Research Center where UNIX was born, nearly fifteen years ago. With others he is jointly responsible for many of UNIX' utilities. He was there when Ken Thompson and Dennis Ritchie experimented with the first versions, and it was Brian who, paraphrasing the name 'Multics,' suggested the term ‘UNIX’ as the final name for the system they developed.

Rob Pike is a true UNIX 'wizard' who joined the Bell Labs group more recently. In the three years he has been there, he has already left an impressive trail of accomplishments, most notably the development (together with Bart Locanthi) of the popular ‘Blit†’ multi-window terminal, which is perhaps best described as a novel type of programmable workstation for UNIX systems.

Kernighan and Pike's book is quite different in approach from the earlier book by Steve Bourne, though there definitely is some overlap in the material presented. Both books cover ‘shell programming,’ and ‘document preparation’ in some detail. Still, the book by Kernighan and Pike is clearly not aimed at the casual user of the UNIX system. On the contrary, they aim for the reader who has mastered the basics and who wants to expand his or her knowledge.

This approach is perhaps best illustrated by the terse reminder at the start of their chapter on ‘program development’ (chapter eight, p. 234): “We are assuming that you understand C.” For sure, not a very gentle introduction to the C language, but then again, the subject was covered at length in the earlier “C reference manual” by Kernighan and Ritchie. For those of us who do already understand C, though, the fifty pages that follow give an outstanding exposition of the development of a substantial and extremely useful program: an interpreter for a Basic-like language, written with the aid of the UNIX compiler writing tools lex and yacc°.

Tools

The book by Kernighan and Pike is crowded with many small examples of useful tools. Shortly after I had circulated a draft of the book at the University where I worked my system was literally swamped with all the little 24-hour watchdog programs that students and co-workers had discovered in the text. As an example, here is the shell script for a small program ‘watchfor’:

```bash
# watchfor: watch for someone to log in
PATH=/bin:/usr/bin
case $# in
  0)echo 'Usage: watchfor person' 1>&2
     exit 1
    esac
   until who | grep "$1"
      do
       sleep 60
    done
```

In this tiny shell script the use of while loops, comments, search paths, case switches, duplication of file descriptors, exit status and shell arguments is illustrated. Let us briefly go through it.

A sharp sign in column one indicates the start of a comment: the shell will simply ignore everything that follows up to the end of the line. The search path is stored in the predefined shell variable 'PATH', which stores a list of directory names, separated by colons. The listed directories are the only ones checked by the shell in an effort to locate a command given by the user. Traditionally UNIX commands are stored in directories /bin and /usr/bin, so it suffices here to set the search path for this script to these two directories.

† The terminal will be marketed by the Teletype Corp. as the model 5620 'dot-mapped display'.
° lex is a general tool for writing a lexical analyzer. yacc can be used to write parser generators.
Next we find a case switch on the number of arguments passed to the script. This number is squirrelled away by the shell in a variable named ‘$#’. If the number of arguments is zero the script will print an error message and exit, returning an error status ‘1’. With the magic ‘1> &2’ we combine the standard output channel (by convention file descriptor 1), where the user will expect the output from the script to appear, with the standard error output channel (file descriptor 2). The script as presented does not provide for an appropriate reaction if the user erroneously types more than a single argument, though the change is in fact trivial:

    case $# in
        1|until who | egrep "$1"
do
sleep 60
done ;;
*|echo 'Usage: watchfor person' 1>|&2
exit 1
esac

This time the loop is only executed if the user gives exactly one argument, hopefully the login name of someone who is likely to login to the system. The argument itself is available in the variable ‘$1’. The condition of the loop consists of two commands: who and egrep, which are connected via a ‘pipe’. The pipe will direct the output from the first command to the input of the second. who will list all active users, egrep will select lines that match the pattern it is given in ‘$1’. The condition of the until statement is set by the exit status of the pipeline. Again by convention this is the exit status of the last program in the pipe: in this case egrep. egrep will return ‘true’ (oddly enough this is the numerical value 0) if it found a match and ‘false’ (some value other than 0) if it didn’t, which is exactly the effect that we wanted here.

Other Examples

None of the examples in Kernighan and Pike’s book are artificial: they all turn out to be instructive, practical and, unfortunately, highly addictive. Here’s a short list of examples they discuss:

- simple aliasing functions, such as a little shell script named ‘cx’ as an abbreviation of ‘chmod +x’ to turn files into executable commands, or ‘le’ as an abbreviation of ‘wc –l’ to count the number of lines in a file;
- a small computerized telephone directory server ‘tel’, that uses grep to locate names and numbers in a list;
- a little C program ‘vis’ that makes non-printing characters in files visible;
- the following very original script that can be stored in a file named ‘2’, and linked to files named ‘3’, ‘4’, ‘5’,... etc.:

    pr –$0 –t –1 $*

The first argument to pr becomes the number of columns in which it will print its output. ‘–t’ turns off the usual page headers, and ‘–1’ sets the page length to 1. ‘$0’ is the name of the script itself. (If you have access to a UNIX system, try: ‘who | 2’ or ‘ls | 5’ and see just how useful this is.)

- straightforward implementations of simple database managing tools, ‘put’ and ‘get’, that can be used to keep track of the revisions that, for instance, popular programs incur;
- an especially elegant program named ‘bundle’ for packing and unpacking sets of text files (e.g. programs) for distribution via ordinary system mail.

All tools, and there are many more in the book itself, are examples of small, well structured scripts or programs that solve practical problems. In nearly all cases the book includes sample runs of the programs showing typical system responses.
UNIX Principles

The strength of the UNIX system, which characterizes the true "UNIX programming environment" can be summarized in a few points (see e.g. Kernighan and Pike, p. 131, or Bourne, p. 4-5):

- Files have no predefined format. The interpretation of a file's contents depends entirely on the program that reads it.
- Programs are tools. Each tool should perform one clearly defined function, and it should be optimized for that single function.
- The output from any program should be understandable as input to other programs. One should avoid fancy headers, trailers or spurious blank lines.
- Conversely, one should be able to replace the input of any program with the output of another. This rules out interrogative programs that prompt the user for arguments. All the information a program needs to run is given on the command line. The program either runs or fails, and returns the appropriate exit status.
- If no arguments are given a program should read the standard input and write the standard output (keyboard and screen by default), so that the program can always be used as a filter. Optional arguments precede filename arguments. Filename arguments may specify additional inputs, but never an output unless properly protected (e.g. with a `-' flag).

In an early version of UNIX one of the disk maintenance commands (a predecessor of `fsck`, named `check`) violated the last principle. When called without an argument this program would faithfully read and check the default filesystem (there was only one) and list the errors on its standard output. If an argument was given, the corresponding file would be used to store the error list. The next version of the command, made for systems with more than just a single disk, expected the name of the filesystem to be checked to be specified in its first argument (an input instead of an output). One day an unfortunate user went back to a precious copy of the old system, typed:

    check /dev/usr

and destroyed the user file system completely. Note that the all-important 'superblock' is one of the first blocks on disk and therefore also one of the first destroyed.

The File System

One of the good things about the book that Kernighan and Pike have written is that it gives access to information that one could obtain only from the gurus before. In very few other books you can find a succinct explanation of the implementation of the file system revealing the secrets of inodes and superblocks. There are chapters that explain how to use and how to program the shell, revealing the mysteries of meta-characters, quoting, shell variables and traps. Other chapters focus on the use of system calls in user programs, building up to the knowledge and insight required to make system programs that dig somewhat deeper into the bowls of, for instance, the file system.

Again, every topic discussed is illustrated with numerous small example programs, such as:

- a program 'waitfile' that waits until a file stops changing for a given period of time (say 60 sec.);
- a program that can correct typing mistakes in filenames by comparing the existing filenames to the user-specified filenames;
- an extremely useful program called 'readslow' that waits for the writer of a file to produce more output when the end of a file has been reached;
- a program called 'timeout' that sets an upper limit to the run time of an arbitrary other user program.
The book includes excellent tutorials on the use of esoteric programs such as *sed*, *awk*, *lex* and *yacc*. However, it is not meant to be a reference manual. "We feel it is more important to teach an approach and a style of use than just details," the authors write. The problem is of course that you can only illustrate an 'approach' to programming by either giving examples or by discussing the underlying principles. Some readers may feel that there is a little too much of the first and too little of the last in "The UNIX Programming Environment." I, for one, regret that not more material was included on the design and study of the algorithms that have been used to build the UNIX programming environment. Pattern matching tools such as *grep*, *egrep*, and *fgrep*, for instance, are not very well understood. There is a lot of folklore around these programs, which has spurred many an interesting lunch discussion (which program is fastest for which problem and why). It would certainly have been revealing to see some of this folklore substantiated or rejected by a thorough discussion of the algorithms used. But then, we have at least something to look forward to in a next book on the UNIX system. "The UNIX Programming Environment" is not a book for utter beginners. For those, however, who not only want to be able to *use* UNIX, but who want to be able to use it *well*, their book is an invaluable guide.

The 1984 Summer USENIX Convention Report

*Rob Pike*

AT&T Bell Laboratories

The 1984 Summer USENIX convention was held June 13-17 in Salt Lake City, Utah, home of the Mormon religion and anchovies. The convention base was the Hotel Utah, owned by the Church but not decorated like it. The center of the hotel is an ornate two-story lobby, with a great chandelier, gold trim, whorled carpets and illuminations, and a grand piano played in the afternoons while tea is served by two hotel maids. As many times before, USENIX was accommodated by a hotel of incongruous style.

The rooms were pleasant (despite the lack of Gideon Bibles), but mostly less magnificent than the lobby. From rooms on the top floor, one could hang out the window to admire the tin and concrete structures adorning the roof. The finest room, however, was that given to Lou Katz, who was attending his last USENIX convention as the organization’s president. His room, or rather rooms, included a bedroom, sitting room, living room and dining room, not to mention the servants’ quarters, all decorated in the Edwardian opulence of the lobby.

The technical sessions were held at the other end of Temple Square from the Hotel Utah, at Symphony Hall, a modern building with a glass-enclosed lobby that was sunny and well-populated even when refreshments were not available (and refreshments were available even at the end of coffee breaks, for the first time in my experience).

Although none of the talks in the technical sessions was outstanding, there were many good ones, few poor ones, and overall the sessions were significantly more interesting than at previous conferences. The main improvement was in the technicality of the talks. There was less marketeering and more attention to design and implementation of the various programs and systems described, as befits a technical conference (at last!). I needn’t describe the technical sessions further, as the conference proceedings, which were available on-site, do so admirably.

Near Symphony Hall, but considerably harder to find, was the vendor show, which was poorly attended by the conference goers. Among the factors affecting the low turnout were its distance from the technical sessions (you had to go outside), the quality of the technical sessions, the busy conference schedule, and a waning interest among USENIX attendees to see yet another dozen UNIX systems-in-a-box. Although the conference organizers thought it was their fault that the turnout was poor (in what was otherwise an outstandingly organized conference), the show was simply uninteresting and this was made known quite quickly through the conference rumor mill.
A novel event at the conference was the computer GO tournament. GO is an ancient Oriental board game with deceptively simple rules; most of its proponents claim it is much more difficult than chess. Because of the large board (19×19), the importance of strategy, and the extensive use of large, loose patterns in good tactics, computers have never done well at GO, and it was to encourage work in computer GO that the tournament was organized.

Only four programs entered, but they spanned a wide range of playing abilities. It was a foregone and correct conclusion that the winner would be NEMESIS, Bruce Wilcox's entry, because it was the only serious program in the competition. The other three were sham, and NEMESIS would have been unbeaten if it had not crashed while soundly trouncing Bruce Ellis's goanna, the second-place finisher. Third place went to Peter Langston's entry, ogo, which was a converted test program that played a famous sleazy strategy: playing moves symmetric with the (supposedly cleverer) opponent until the symmetry is broken by someone occupying the central point on the board. Last place went to jim, a program by Hank Dietz, that always committed suicide before its playing skill could be determined.

The tournament was played on a full-sized board, with a time limit of one hour of user VAX-11/780 CPU time per player per game. When the rules were announced a few months before the tournament, prospective entrants felt that was insufficient time, but when the tournament was played, no program came near the time limit, although jim might have had it finished a game. NEMESIS used a total of 36 minutes in 5 games, and the simplistic ogo took 6 minutes 31 seconds, just over a minute of CPU time per game. But the speed record must fall to goanna, which placed second in the world's first full-board computer GO tournament, using a total of 12 seconds of CPU time, an average of 2.4 seconds per game, or 2.7 milliseconds per move. Goanna was certainly the favorite of the surprisingly large and appreciative crowd at the tournament.

Another unusual feature of the convention was a trivia quiz. Contestants were given less than two days to answer 84 pointless questions about UNIX history (sample: What was the first edition of the manual to be typeset?). As the deadline for submissions neared, inquisitive groups swarmed around dignitaries and people known to have been there in the early days, ready to scribble down whatever clues might be revealed ("Mr. Chessen, what was your phone number at Bell Labs?").

The best entry — 60 correct — was under the pseudonym I. P. Stubbies, although none of the actual contributors to the entry were Australian. The best individual entry was submitted by Jim McKie, and had 57 correct answers; Ron Gomes had 56. Of course, neither of these really did the research individually, but the contest rules were deliberately vague about ethics. Because of the capricious judging, the best prize — an authentic UNIX Second Edition boot DECtape, went to McKie, Stubbies got a silly trophy and Ron Gomes got an official Sun Microsystems Bill Joy name badge. (The GO tournament prizes were even less interesting.)

Off-site activities: There was a dinner do at the Snowbird ski resort, where people saw snow for the first time since they went skiing the previous day; some people wandered over to the University of Utah, where they saw how to house a tertiary educational institute in secondary school buildings; and on the last night there was a party that I have been asked not to discuss.

Overall, a good conference, for the right reasons, but nothing to write to ;login: about.
Is This System Mannerist or Rococo in Style?

A Visit to the Promised Land
Salt Lake City and USENIX

P. Rayonnant

Sunday

Five o'clock, the new frontier beckons. Goodbye Dick, goodbye Doughball; don't eat any cables while I'm away. Why isn't the bus here? What, they changed the timetables last week? And this is a holiday anyway? Taxi. No, I don't know which travel agency ordered the tickets, don't you have them indexed by traveller-name? "For operational reasons we are flying this morning in a Bandicoot 20 seater across the North Sea. I'm your co-pilot, and once we're airborne, I'll be serving coffee."

London - yick. Gorky Park - the book was better.

I visited my first two U.S. airports, and got lost in both. There seems to be an assumption that the traveller knows which airline he is flying with. For the connecting flight from St. Louis to Salt Lake City my ticket just said WA691. Each airline has its own monitors, departure lounges, etc. and there is no information desk. Could be Wright Bros. Airlines, I suppose. If in doubt, ask a policeman.

Had a beer while waiting for the flight. Not promising.

Flying across the central U.S. with Western Airlines was dull, the country is flat, just like The Netherlands, except no one has bothered to make the waterways run in straight lines. How untidy.

In Salt Lake City I just followed the signs to "Baggage Reclaim." Of course, there are two....

This hotel is IMPRESSIVE. Must be old, the lifts are just like in The Netherlands, slobby. The single-room has a bed wider than it is long, three pillows wide, a Mormon-bachelor occasional-double? Speaking of Mormons, the Hotel Utah is right next to the centre of Salt Lake City, where the Temple is (I can look down into it from my room); the streets in Salt Lake City are not named, but numbered on a Cartesian grid ("100 South 300 West"), and yes, Temple Square is at 0,0.

The clock says eight in the evening, the body says "It's almost time to get up tomorrow", so it must be time for a wander around (short, everything is closed), and something to eat. "Hi Sir! Have you made your choice yet?" Mumble mumble. "Thank you, Sir! Will that be all?" Mumble mumble. "Thank you, Sir! My name's Larry, and I'll be your waiter for this evening." Mumble mumble ZZZZZZZZZZZZZZZZZZ.

Monday

No! Not yet! Not Yet! OK. I suppose so. Lunchtime. Wandered around. Still doesn't look very alive (I don't either I suppose).

Registration opens in the late afternoon, so I get registrated. Along with the registration packet comes the proceedings, a really handy thing to have before the conference; now I can ask even more interesting questions since I'll know it all beforehand. Why did I bother coming? It also means that I don't have to write everything down in a boring trip report, I only need to give the undocumented features not in the manual.

Finally bump into some people I know, time to eat and drink. Well, more eat than drink, Utah has some strange laws about the sale of alcohol. Fortunately beer is easy, but were we foolish enough to want wine with our meal, the restaurant has a Mini State-Liquor Store at the front (within and part of the restaurant). Customers who want wine have to go there themselves, buy the evil stuff and return to their seats. The restaurant is allowed to provide glasses and open the wine for you, but is not allowed to pour it, you have to do that yourself. Weird.
Back to someone's hotel room for a wine tasting, and very nice they tasted too.

Tuesday

Wandered around some more. Saw Star Trek III - The Search for Spock in the afternoon, for want of something better to do. I was looking less like Spock by time I came out.

There were some tutorial sessions on C-Style, 4.2BSD Internals, Networking, etc. Some were sold out, some had a handful of attendees. USENIX seems to find these tutorials popular, and wants to continue them at future meetings. The problem is finding people who know the the stuff and can communicate it well. Suggestions are welcomed.

The exhibition was also under way, less than 100 stands I would guess. Nothing really new. AT&T wouldn't let anyone touch their machines. Some of the 68K boxes are becoming quite nice; Integrated Solutions were claiming theirs was "what the Micro-Vax should have been".

< <= 'echo "Goodbye World!"

to the Bourne Shell was popular amongst the vendors I tried it on, some machines just crashed, others needed to be powered off.

The most interesting thing about the exhibition was the Ethernet. There was an Ethernet cable to which vendors were invited to attach their machines. I saw about six different brands of hardware all chatting away nicely (using TCP/IP protocols), including a laser printer station from Imagen. Apparently there was hardly any preparation for this Ethernet demonstration, all the more remarkable.

Tuesday Evening

Things are beginning to move now, people are beginning to arrive. And the hospitality suites are open!

Hospitality suites are places where, under the influence of free food and alcohol, you are scrutinised and pestered by representatives of the company supplying the food and drink with a view to them weaning you away from your current blissful employment.

But wait! There is something wrong! Here I am, eating and drinking in a hospitality suite, and the reps are ignoring me! What is wrong? Why is everyone looking at the T.V. set? I ask, and it seems there is a basketball game on, the last one of the season or something. It looks exceedingly dull, everyone runs to one end of the pitch, the ball may or may not go through a hoop attached to a board, then everyone runs back to the other end; the team which loses seems to be the one which fails to put the ball through the hoop most, as far as I could see each team gets an equal number of tries. If it was profiled, all the running around would probably be moved out of the loop.

Other hospitality suites were the same. Refreshing (at the time).

4 This wasn't the only strange game I saw on the television, there was also American football and baseball. American football is remarkable in that only one person on each team ever actually uses a foot to kick the ball, and he spends most of his time sitting on a bench, waiting for the ball to be placed in front of the goal posts. Maybe we should rename soccer handball, since only one person on each team is allowed to handle the ball?

The Dutch for baseball is "honkbal".
Wednesday Morning

Come Wednesday and it is time for the official conference to begin in earnest. The Symphony Hall is impressive, and I found the lack of central aisles (one had to enter from the sides) adequately compensated for by the comfy chairs and leg room.

There was a professional audio-visual set-up, three large screens at the back of the stage, one for overheads (hardly used), one for slides, and the central one with a projection of the speaker’s head and shoulders. Hello Mum. Slides were definitely the order of the day, and some poor soul slaved away night and day making slides from the overheads which everyone brought; he deserves some thanks.

Some computer-graphics films were shown on the multi-media equipment as the UNIX community filtered in for the start of the meeting (why do these things always start so early?). Someone told me they were from the last SIGGRAPH. Econo-Mars Earthtours looked preferable to British Caledonian.

First business of the day was some announcements. There were two competitions being run during the conference, the First USENIX Computer GO Tournament, run by Peter Langston, and the UNIX Trivia Quiz, run by Rob Pike, 84 questions along the lines of “What was the telephone extension of the inventor of multiplex files?”

The next meeting will be held in Dallas (there will be a simultaneous conference and trade-show held there by /usr/group, and future meetings will be in Portland (USENIX only), with Atlanta, Boston and San Francisco as possibilities after that. USENIX is also sponsoring limited attendance (approx. 120) workshops, on topics such as distributed systems and computer graphics; participants will have to send in a position paper first.

The USENIX Newsletter ;login: now has Brian Redman as technical editor, and USENIX is still looking at the possibilities of a journal and electronic publication.

After that, someone from the EUUG spoke for five minutes about what was happening in Europe and announced the forthcoming EUUG meeting in September in Cambridge. One of his slides appeared to be in Dutch, but it was just back to front and upside down.

There then followed the keynote address, *An Architecture History of the UNIX System* by Stuart Feldman (the man who brought you *make*). This was a real treat. It didn’t appear in the proceedings, as that would have spoiled the fun. As it was there was lots of fun, not all of it expected. The talk was basically a pictorial trip through the last thousand years of Western architecture, detailing the rise and decline, with side references to the history of the UNIX system. The added fun came as Stuart’s slides caused havoc with the slide-projector, having the audacity not all to be of the same thickness (many were borrowed from libraries and museums). At one point, as one refused to be ejected from the projector, there were cries to pass the remaining slides round the audience instead; coffee was taken early as the projector was dismantled.

Michael Tilson then gave a well-reasoned talk about the need for standards in UNIX, and the coming confrontation between UNIX and Godzilla. He gave a history of UNIX, including a slide depicting the real UNIX family tree, it looked like a plate of spaghetti.

Hot in pursuit came some talks about electronic mail and news. The most interesting was given by Robert Elz and described MCSNET, the Australian alternative to UUCP. There are some published papers on this, and it is quite interesting, all routing being taken care of by the network, explicit routes don’t need to be given. There was a lot of speculation as to how this would scale up to a network the size of the current UUCP network.

Other talks in this session were *Broadcasting of Netnews and Network Mail via Satellite*, *The Berkeley Internet Domain Server*, *MMDF II: A Technical Review* and *DRAGONMAIL: A Prototype Conversation-Based Mail System*.

I didn’t see all of these, having to go to a meeting over lunch, where I had a banana-split King Kong would have had trouble with.
Wednesday Afternoon

Apart from the effort needed to digest the world's largest banana split, the afternoon was spent listening to talks about distributed things.

The first session was about networks and distributed systems. In one talk about converting the BBN TCP/IP software for 4.1BSD to 4.2BSD, the claim was made that the throughput was comparable. Someone from the audience pointed out that the particular Ethernet controller being used was in fact the limiting factor in performance in both cases.

Other talks included how to synchronise the clocks of the different machines on a local area network, the LOCUS Distributed UNIX System, and Project Athena.

This last is interesting, as it gives a sense of scale of investment being made in the U.S. by computer manufacturers. Project Athena is a joint venture by MIT, DEC and IBM, which will install approximately 1600 VAX-based personal computers at MIT for an experiment in undergraduate teaching; about 2500 machines will be installed in all. 4.2BSD is the current base software. You can imagine the problems of networking, backup, maintenance and just plain scale. The results, whatever they are, will be interesting.

The second afternoon session (after the delights of fishing cans of strange liquids out of enormous buckets of ice; what is "root beer", it tasted of toothpaste?) was about distributed filesystems. Peter Weinberger talked about the Edition 8 Network Filesystem.

Peter was fun to watch, he bounced around so much, dashing over to try to read his own slides, that the audio-visual projectionist had to work up a sweat. The talk didn't differ a lot in content from that he gave in Amsterdam in 1982, but he'd had the time to replace the hand-written overheads by BTL-style slides which even he couldn't read.

The filesystem gives you network independence by using asymmetric virtual circuits, the client reads the server's files, the server handles permission translations and time differences. Remote objects are filesystems, not disks; pairs of machines negotiate a connection. Don't expect much from the network, as that's what you get.

The implementation uses user-level servers, and switches at the inode-level in the kernel to test if this is a remote file or not (doing it at the user library-level is described in the abstract as a "maintenance nightmare").

The other two talks in this session were also quite interesting, a system called IBIS from Purdue which is based on 4.2BSD and uses the host:filename syntax to access remote files, and the Livermore Interactive Network Communication System (LINCS). This runs in a very heterogeneous environment, and includes TCP/IP connections from VAXs running VMS/EUNICE to "real" UNIX machines.

The last part of the afternoon was taken up by a panel session on distributed file systems. Not a lot interesting was said. Bill Joy was in "Paris mode".

Wednesday Evening

Back to nature, a bus to a ski-lodge in the mountains, and the official conference reception. Sunshine, trees, snow, rushing streams, all to be experienced as you stand beside a bus with burned-out transmission.

But it was fun. Cable-car from the ski-lodge (Snowbird) to the top of a mountain (11,000ft), slither around in T-shirt and gutties. Back down for food and wine (altitude affects both your breathing and also the ease with which alcohol affects your system). Crossed-legs bus ride back to the hotel, then on to the DEC hospitality suite, it was open every night, all the time. This time people in uniforms came to break it up. What fun.
Thursday Morning

No one problem! Breakfast is though. "What you got?" "Oh! We have ... etc." "Got any muesli?" "What's that?" Kelloggs wins the day.

My enthusiasm for cornflakes is matched by my enthusiasm for talks about C compilers and other programming language issues. There was a talk about Modula-2 in the system-programming environment which didn't look too bad. However, many people's vocabulary doesn't extend beyond the third letter of the alphabet.

Coffee break. Well, they call it coffee; there's more strength in the C-compiler's type checking. There are also strange sticky pastries, guaranteed to necessitate the application of a Swiss Army knife to the moustache.

A number of talks about programming environments and windowing systems. Rob Pike described a development which we can't get of something he talked about before which we still haven't got. Pity, it's probably what I want.

A team got together over lunch to discuss questions such as "Which university stole UNIX by telephone?". I like pastrami. They disagreed over some of the finer points, and two different entries went in to the "Pastrami Trivia Quiz".

Thursday Afternoon

Into the kernel. Multiprocessor UNIX, what you need to do to make it run on a 3B20, the system can be configured to run in uni-processor mode (no performance penalty) or multi-processor mode (70% throughput increase).

Richard Miller, of porting UNIX to the 7/32 fame, talked about adding demand paging to System V. Only ps(1) need be changed, the performance is equivalent to the swapping system. Uses a working-set page replacement policy as opposed to the 4.3BSD global clock algorithm. It wasn't done on a Vax originally, though.

After more fishing around in buckets of ice for cans of refreshment, the talks resumed. Tom Killian gave his talk about processes as files (Andrew Hume gave a very brief presentation of this in Nijmegen). It really is neat the way it fits in with "the UNIX way of doing things". Debugging and ptrace just fall out naturally. Implementation goes through Peter Weinberger's Edition 8 filesystem, where there is a file system type for each of these weird things.

Thursday Evening

Time to hit the hospitality suites again. Silicon Graphics have their flight simulator in a hotel room, but the room is simulating the Black Hole of Calcutta. Still, they did have some Anchor Steam Beer, positively tasty compared to the competition. Have something to eat instead, visit the GO competition (won by a ported IBM-PC program), visit the Human Computing Resources Coffee and Liqueurs, drop in on the DEC suite on the way home, finally to bed. It must be the jet-lag.

Friday Morning

Slightly more of a problem. Cornflakes, weak coffee are beginning to take their toll.

Performance analysis and comparisons is what we have first. Someone at AIM collected a lot of benchmark data, then worked out how to display it; what it means is just the same as any other UNIX benchmark you've ever seen. Not a lot.

Two talks, one from a mannerist comparing SV to 4.2BSD, and one from a rococoist about how to speed up 4.2BSD. A big fight was expected, involving a somewhat partisan audience, but it never happened. Everyone was amazingly appreciative of both sides of the coin, or had also been to the DEC
hospitality suite. Both systems have their good and bad points, you pays your money, you get the same performance provided you matched the system to the use.

Before the coffee break, Rob Pike announced the results of the Trivia Quiz. He was surprised about how much people out there actually knew about the dark history of UNIX. However, the winning entry from a team only managed 60 correct answers out of 84. The second placed entry managed 56 correct, and received the best prize, a DECTape labelled "11/20 8K sys with RP11 units July 5/72".

The talks in the remainder of the morning were missed due to a combination of worrying about an export licence for the tape I’d just received, and the need to take an early lunch with some people who had to leave the conference early.

Friday Afternoon

One of the interesting things about the talks throughout the conference was the number which dealt with forecoming technology, more saying "what are we going to do with this when it arrives?" rather than "here’s my solution to last years problem". And two things which are coming are optical disks and the need to distribute software as automatically as possible where possible. I wouldn’t say I agreed with the possible solutions offered to these two particular examples, but at least some people are looking a bit further than yesterday.

That was more or less the conference, and a good one it was too. There were some other talks and panel sessions, but (and this is my only complaint about the conference organisation) they were held in another place, about 7 minutes walk away and in parallel with the other talks. This made it impossible to see whatever was best at the time. Some looked like they might be interesting, the future of BSD, the UUCP/USENET projects (doomed to failure in my humble opinion) and the ANSI C standard draft.

Throughout the conference Rob Pike was taking pictures of people for the face-server, which would be digitised into high-resolution form, then digital faces of any size are generated on demand; some commonly used ones, such as the 48x48x1 faces for mail announcement are precomputed.

There was strong competition for the "hairy knees of the conference" award, Salt Lake City being a warm place. However, after a day spent skiing in his shorts, the twin pillars of flame owned by the runner-up from the Nijmegen conference, Andrew Hume, made all other entrants pale in comparison.

Friday Evening

There was a "wizards" party up in a cabin in the mountains. Fun trekking through the snow and trees to somewhere where the door was a window, the fridge was a snowdrift at the backdoor and the toilet a gopher (or similar beast) hole.

It was great, especially trying to get down in the dark through a snow covered forest to a car after the snowdrift was empty.

Saturday Morning/Afternoon/Sunday

Somewhere in here I negotiated two U.S. airports without mishap (pity British Caledonian can’t learn to do that too).

What’s the collective name for a bunch of U.S. kids? A brace. No wonder they all need psychoanalysis in later life, having to go through puberty with scaffolding on their faces.

Somewhere in here Saturday became Sunday very quickly. No, "Reuben, Reuben" is not about sandwiches.

"I climbed on the back of a giant anchovy, and flew off through a gap in the clouds...."
Trivia Quiz Answers

Below are the answers to last the trivia quiz that appeared in the last issue of ;login:. The syntax for the answers is:

left to right precedence
a | b is a or b
a & b is a and b (any order)
a ! b is a and not b

1. The source code motel: your source code checks in, but it never checks out. What is it?
    sccs
2. Who wrote the first UNIX screen editor?
    irons
3. Using TSO is like kicking a [what?] down the beach?
    dead whale
4. What is the filename created by the original dswh?
    core
5. Which edition of UNIX first had pipes?
    third | 3
6. What is .=-0=.-?
    empire
7. Which Stephen R. Bourne wrote the shell?
    software | 1138 | regis
8. Adam Buchsbaum’s original login was sjb. Who is sjb?
    sol & buchbaum
9. What was the original processor in the Teletype DMD-5620?
    mac & 32
10. What was the telephone extension of the author of mpx?
    7775
11. Which machine resulted in the naming of the ‘‘NUXI problem’’?
    series 1 | series one
12. What customs threat is dangerous only when dropped from an airplane?
    belle | chess machine
13. Who wrote the Bourne shell?
    bourne
14. What operator in the Mashey shell was replaced by ‘‘here documents’’?
    pump
15. What names appear on the title page of the 3.0 manual?
    dolotta & petrucelli & olsson
16. Sort the following into chronological order: a) PWB 1.2, b) V7, c) Whirlwind, e) System V, f) 4.2BSD, g) MERT.
    cagbef | c a g b e f
17. The CRAY-2 will be so fast it [what?] in 6 seconds?
    infinite | np-complete | p = np
18. How many lights are on the front panel of the original 11/70?
    52
19. What does FUBAR mean?
   failed unibus address register
20. What does “joff” stand for?
   jerq obscure feature finder
21. What is “Blit” an acronym of?
   nothing
22. Who was rabbit! bimmler?
   rob
23. Into how many pieces did Ken Thompson’s deer disintegrate?
   three | 3
24. What name is most common at USENIX conferences?
   joy | pike
25. What is the US patent number for the setuid bit?
   4135240
26. What is the patent number that appears in UNIX documentation?
   2089603
27. Who satisfied the patent office of the viability of the setuid bit patent?
   faulkner
28. How many UNIX systems existed when the Second Edition manual was printed?
   10 | ten
29. Which Bell Labs location is HL?
   short hills
30. Who mailed out the Sixth Edition tapes?
   biren | irma
31. Which university stole UNIX by phone?
   waterloo
32. Who received the first rubber chicken award?
   mumaugh
33. Name a feature of C not in Kernighan and Ritchie.
   enum | structure assignment | void
34. What company did cbosg ! ccf work for?
   weco | western
35. What does Bnews do?
   suck | gulp buckets
36. Who said “SEX, DRUGS and UNIX”?
   tilson
37. What law firm distributed Empire?
   dpw | davis & polk & wardwell
38. What computer was requested by Ken Thompson, but refused by management?
   pdp-10 | pdp10
39. Who is the most obsessed private pilot in USENIX?
   goble | ghg
40. What operating system runs on the 3B-20D?
   dmert | unix/rtr
41. Who wrote find(1)?
   haight
42. In what year did Bell Labs organization charts become proprietary?  
83
43. What is the UNIX epoch in Cleveland?  
1969 & dec & 31 & 19:00
44. What language preceded C?  
nb
45. What language preceded B?  
bon|fortran
46. What letter is mispunched by bcd (6)?  
r
47. What terminal does the Blit emulate?  
jerq
48. What does "trb" stand for (it's Andy Tannenbaum's login)?  
tribble
49. allegra ! honey is no what?  
lady
50. What is the one-line description in vs.c?  
screw works interface
51. What is the TU10 tape boot for the PDP-11/70 starting at location 100000 (in octal)?  
012700 172526 010040 012740 060003 105710 012376 005007
52. What company owns the trademark on Writer's Workbench™ Software?  
at & t communications
53. Who designed Belle?  
condon|jhc
54. Who coined the name "UNIX"?  
kernighan|bwk
55. What manual page mentioned Urdu?  
typo
56. What politician is mentioned in the UNIX documentation?  
nixon
57. What program was compat(1) written to support?  
zork|adventure
58. Who is "mcteq"?  
michael & toy & esquire
59. What was "ubl"?  
rogue|under bell labs
60. Who bought the first commercial UNIX license?  
rand
61. Who bought the first UNIX license?  
columbia
62. Who signed the Sixth Edition licenses?  
shahpazian
63. What color is the front console on the PDP-11/45 (exactly)?  
puce
64. How many different meanings does UNIX assign to ","?  
lots|many|countless|myriad|thousands

Volume 9, Number 4  September 1984 21
65. Who said, "Smooth rotation butters no parsnips"?
   john & tukey
66. What was the original name for cd (1)?
   ch ! dir
67. Which was the first edition of the manual to be typeset?
   4 | four
68. Which was the first edition of UNIX to have standard error/diagnostic output?
   5 | five
69. Who ran the first UNIX Support Group?
   maranzano
70. Whose Ph.D. thesis concerned UNIX paging?
   ozalp & babaoglu
71. Who (other than the obvious) designed the original UNIX file system?
   canaday
72. Who wrote the PWB shell?
   mashey
73. Who invented uucp?
   lesk
74. Who thought of PWB?
   evan ivie
75. What does grep stand for?
   global regular expression print | g/re/p | g/regular expression/p
76. What hardware device does "dsw" refer to?
   console & 7
77. What was the old name of the "sys" directory?
   ken
78. What was the old name of the "dev" directory?
   dmr
79. Who has written many random number generators, but never one that worked?
   ken | thompson
80. Where was the first UNIX system outside 127?
   patent
81. What was the first UNIX network?
   spider
82. What was the original syntax for "ls -l | pr -h"?
   ls -l:"pr -h" | "$ls -l"<pr -h
83. Why is there a comment in the shell source "/* Must not be a register variable */"?
   registers & longjmp
84. What is it you're not expected to understand?
   6 | 5 & process
Local User Groups

The USENIX Association will support local user groups in the United States and Canada in the following ways:

- Assisting the formation of a local user group by doing an initial mailing for the group. This mailing may consist of a list supplied by the group, or may be derived from the USENIX membership list for the geographical area involved. At least one member of the organizing group must be a current member of the USENIX Association. Membership in the group must be open to the public.
- `:login:` will publish information on local user groups. Information on local groups giving the name, address (phone number and/or net address), time and location of meetings, special events, etc. is welcome.

Please contact the USENIX office if you need assistance in either of the above matters. Our current list of local groups follows.

The Front Range group meets about every two months at different UNIX sites for informal discussions.

Front Range Users Group
N.B.I., Inc.
P.O. Box 9001
Boulder, CO 80301
Attn. Steve Gaede
(303) 444-5710
hao!nbires!gaede

Dallas / Fort Worth UNIX User's Group
Advanced Computer Seminars
2915 L.B.J. Freeway, Suite 161
Dallas, TX 75234
Attn. Irv Wardlow
(214) 484-UNIX

Unigroup is a non-profit organization in the New York City area for users and vendors of products and services for UNIX systems.

Unigroup of New York
G.P.O. Box 1931
New York, NY 10116

The UNIX Users of Minnesota meets on the first Wednesday of each month. For information on times and locations contact:

UNIX Users of Minnesota
Carolyn Downey
(612) 934-1199

In the Atlanta area there is a group for people with interest in UNIX or UNIX-like systems:

Atlanta UNIX Users Group
P.O. Box 12241
Atlanta, GA 30355-2241
Marc Merlin (404) 255-2848
Mark Landry (404) 874 6037

There is an informal group that meets in the Washington, D.C., area every two months or so. The current contact for that group is:

Neil Groundwater
Analytic Disciplines, Inc.
Suite 300
8320 Old Courthouse Road
Vienna, VA 22180
(703) 893-6140
npg@lbl-csam
USENIX Conference Proceedings Available

Proceedings for the following USENIX conferences are available from the organizations listed. California residents please add applicable sales tax. Payments must be in US dollars payable on a US bank.

Salt Lake City — Summer 1984, and Toronto — Summer 1983

Copies of the proceedings of the Salt Lake City Conference are available for $25 per copy, and of the Toronto Conference for $30 per copy. Add $15 per copy for overseas postage. Send your check or purchase order to:

USENIX Association
P.O. Box 7
El Cerrito, CA  94530

Payment must be received before proceedings will be shipped.

Washington DC UniForum Conference — Winter 1984

Copies of the proceedings of the UniForum Conference are available for $30 per copy, plus $20 per copy for overseas postage. They may be ordered from:

/usr/group
4655 Old Ironsides Drive, #200
Santa Clara, CA  95054

San Diego UNICOM Conference — Winter 1983

Copies of the proceedings of the San Diego UNICOM Conference are available for $25 per copy, plus $15 per copy for overseas postage. Send your check or money order to:

Software Tools Users Group
1259 El Camino Real, #242
Menlo Park, CA  94025

Thanks

A tip of the ol' "Hatlo" goes to Ken O'Mohundro, President of Able Computer of Irvine, California, for his company's donation of a VMZ-32 sixteen line terminal multiplexer for use with the Association's VAX-11/730 computer. The VMZ-32 is software compatible with DEC's DZ-32 multiplexer and also offers full modem control on all 16 ports. The new multiplexer will be used to connect several terminals in the Association's El Cerrito office, as well as allowing for dialup access by Board members. uucp connections to ucbvax and decvax have already been established and Association members wishing to send electronic mail to the office can do so via

{decvax|ucbvax}@usenix!office
Tax Exempt Status Granted to USENIX

The USENIX Association has been granted Federal tax exemption status, effective April 25, 1984, under the IRS 501(c)(3) statute. State tax exemption status has also been granted, effective January 19, 1984. The achievement of this status exempts USENIX from paying corporate income tax and also makes available other benefits such as reduced mailing rates. In addition, any contributions to the organization from companies or individuals (not including membership fees) may now be considered tax-deductible. To retain this status, USENIX must maintain its educational nature and refrain from engaging in certain activities such as political lobbying and commercial activities not intended as support for the organization (for example, selling of products at the conference vendor exhibits).

USENIX 4.2BSD Manuals: Status Report

As announced in the April 1984 issue of :login: (Vol. 9, No. 2), USENIX has sponsored the printing of 4.2BSD manuals for Institutional and Supporting Members holding a 4.2BSD source license. A description of the layout and format of the manuals as well as an order form was included in the original :login: announcement, and order forms were sent to all recipients of the 4.2BSD distribution tapes as well. On July 3rd, after waiting approximately 30 days for orders to accumulate, the printer was told to reproduce the following quantities of manuals:

- User's Manuals: 5,000
- Programmer's Manual: 5,000
- System Manager's Manuals: 1,800

It was planned that this quantity of manuals would prove sufficient to fill all orders generated over the next 3-4 month period. Little did we realize how popular these manuals would prove to be! As of September 10th the following orders had been received and processed:

- User's Manuals: 5,900
- Programmer's Manuals: 4,800
- System Manager's Manuals: 2,000

Thus the original printing run is completely sold out of the User's and System Manager's Manuals, and only a few sets of the Programmer's Manuals are left. We have negotiated with the printer to make another printing run in order to fill the backlog of outstanding orders and to provide a supply of manuals for future orders that are received during the next few months. As stated in the original :login: announcement, it is uncertain how long manuals will be available for. If your site is interested in manuals and has not already ordered, you are encouraged to do so promptly. An order form is included in this issue of :login:.

Incidently, we have received many positive comments concerning the new manual layout and format. Thanks are due Sam Leffler of Lucasfilm and Mike Karels of the University of California, Berkeley for their hard work at organizing the manual sections and producing the typeset masters for the printer. John Lasseter, also of Lucasfilm, did the cover design.

Thomas Ferrin, Board Member
Your article for ;login: is welcome; send it electronically to
{ucbvax|decvax!usenix!login

or through the U.S. Mail to

Editor, ;login:
USENIX Association
P.O. Box 7
El Cerrito, CA 94530
4.2BSD Manual Reproduction Authorization and Order Form

Please return both copies

USENIX Member No.: ________________

Purchase Order No.: ________________

Date: ________________

As the representative of a USENIX Association Institutional or Supporting Member in good standing, and as a bona fide license holder of both a 4.2BSD software license from the Regents of the University of California and a UNIX™/32V or System III or System V license or sublicense from Western Electric Company, and pursuant to the copyright notice as found on the rear of the cover page of the UNIX/32V Programmer's Manual stating that

"Holders of a UNIX™/32V software license are permitted to copy this document, or any portion of it, as necessary for licensed use of the software, provided this copyright notice and statement of permission are included",

I hereby appoint the USENIX Association as my agent, to act on my behalf to duplicate and provide me with such copies of the Berkeley 4.2BSD Manuals as I may request.

Signed: __________________________________________

Institution: _________________________________________

Ship to: 

Name: ____________________________________________

________________________________________

________________________________________

________________________________________

Phone: ____________________________________________

Billing address, if different:

Name: ____________________________________________

________________________________________

________________________________________

________________________________________

Phone: ____________________________________________

The prices below do not include shipping and handling charges or state or local taxes. All payments must be in US dollars drawn on a US bank.

<table>
<thead>
<tr>
<th>Manual</th>
<th>Price per copy</th>
<th>Quantity</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2BSD User's Manual</td>
<td>$18.50</td>
<td>_______</td>
<td>$______</td>
</tr>
<tr>
<td>4.2BSD Programmer's Manual</td>
<td>$17.00</td>
<td>_______</td>
<td>$______</td>
</tr>
<tr>
<td>4.2BSD System Manager's Manual</td>
<td>$9.50</td>
<td>_______</td>
<td>$______</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>_______</td>
<td>$______</td>
</tr>
</tbody>
</table>

[ ] Purchase order enclosed; invoice required
(Purchase orders must be enclosed with this order form.)

[ ] Check enclosed for the manuals: $__________
(Howard Press will send an invoice for the shipping and handling charges and applicable taxes.)

Make your check or purchase order out to Howard Press and mail it with this order form to:

Howard Press

c/o USENIX Association

P.O. Box 7

El Cerrito, CA 94530

for office use  l.v.: __________ check no.: __________ amt. rec'd: __________
Trivia Quiz Answers

4.2BSD Manuals Available

Workshops

Winter '85 Meeting