

# ;login:

The USENIX Association Newsletter

Volume 8 Number 4

September 1983

## CONTENTS

UniForum Trade Show and Technical Conference .....	3
Call For Papers for the 1984 Winter UniForum Conference .....	3
USENIX Conference Scheduled for Salt Lake City .....	4
Go Tournament at Salt Lake City Conference .....	4
USENIX Software Distribution Tapes .....	6
1982 Tapes .....	6
Problem with 1982 32V Tape .....	6
1983 Tapes .....	6
Call for Tape Submissions .....	7
USENIX Board Election Nominating Committee Formed .....	7
San Diego Proceedings Still Available .....	8
Toronto Proceedings .....	8
New York UNIX Users Group Formed .....	8
2.9BSD — Berkeley Software Distribution for PDP machines — Now Available .....	8
USENIX Association Toronto Conference Notes .....	11
Opening Session .....	11
Programming Tools 1 .....	13
UNIX Implementation 1 .....	15
Programming Tools 2 .....	17
UNIX Implementation 2 .....	18
User Interface 1 .....	21
UNIX Implementation 3 .....	22
User Interface 2 .....	24
Compilers and Languages 1 .....	25
UNIX Implementation 4 .....	27
Compilers and Languages 2 .....	30
UNIX Directions .....	31
Networking .....	33
Applications .....	35
UNIX Mail .....	36
Standards, Validation, and Portability .....	38
Open Session .....	40
Information on Vendor Exhibits at the Toronto Conference .....	41
Cross Compiler List .....	50

The deadline for submissions for the October issue of *;login:* is October 12

## NOTICE

*;login:* is the official newsletter of the USENIX Association, and is sent free of charge to Individual, Public and Institutional 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 not-for-profit corporation incorporated under the laws of the State of Delaware. The officers of the Association are:

President	Lou Katz	Directors	Bruce S. Borden
Vice-President	John L. Donnelly		Alan G. Nemeth
Secretary	Lewis Law		Deborah K. Scherrer
Treasurer	Thomas Ferrin		Waldo M. Wedel

The editor of *;login:* is Tom Strong.

Membership information can be obtained from the Association office:

USENIX Association  
P.O. Box 7  
El Cerrito, CA 94530  
(415) 528-UNIX

Members of the UNIX community are heartily encouraged to contribute articles and suggestions for *;login:*. Your contributions may be sent to the editor electronically at

ucbvax!g:usenix

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 1½" and a bottom margin of 1¼". Hardcopy output from a line printer or most dot-matrix printers is not reproducible.

---

This newsletter may contain information covered by one or more licenses, copyrights, and/or non-disclosure agreements. Permission to copy without fee all or part of this newsletter is granted only to Institutional members of the USENIX Association, provided that copies are made only for internal use at the member campus or plant site. No other reproduction of this newsletter in its entirety or in part may be made without written permission of the Association.

---

\*UNIX is a trademark of Bell Laboratories.

;login:

## UniForum Trade Show and Technical Conference

The UniForum Trade Show and Technical Conference will be held January 17→20, 1984, at the Washington-Hilton Hotel in Washington, D. C. The technical conference will be administered by the USENIX Association, with Reidar Bornholdt of Columbia University as the program chair. General information on UniForum may be obtained from:

UniForum  
Suite 205  
2400 East Devon Avenue  
Des Plaines, IL 60018  
800-323-5155 (312-299-3131 in Illinois)

## Call For Papers for the 1984 Winter UniForum Conference

The 1984 Winter meeting will be a combined USENIX-/usr/group conference. There will be two concurrent programs; one programmed by /usr/group, the other by USENIX. This is a call for papers for the USENIX program.

Papers should be technical in nature and of interest to the USENIX community. Talks presented at previous meetings will not be accepted. Suggested topic areas include but are by no means limited to:

Systems — kernel enhancements, ports to new machines, networks, optimizations, performance issues, real time mods, etc.

Programming tools and environments — editors, utilities, new languages and reports on developments in established languages, etc.

Applications — database systems, graphics, mail systems, real time projects, etc.

UNIX issues — trends and/or directions in the UNIX world, standards, security, etc.

Authors must submit an abstract of at least 250 words by 1 November 1983. Papers will be chosen from these abstracts. Abstracts must include the following:

Title  
Name of Author  
Institution or Company  
Mailing address (U.S. Mail and network)  
Phone number  
Audio-visual requirements.

Abstracts should be submitted to the USENIX program chair, preferably by electronic mail:

Reidar Bornholdt  
Room 9-451 P&S  
630 West 168 Street  
New York, NY 10032  
[ucbvax!]harpo!cucard!reidar

;login:

## USENIX Conference Scheduled for Salt Lake City

A USENIX Association conference is scheduled for June 12→15, 1984, at the Hotel Utah in Salt Lake City, Utah. A vender exposition is scheduled at the Salt Palace to run concurrently with the technical sessions. Randy Frank of the University of Utah is the local arrangements chair. The program chair is Spencer Thomas of the University of Utah. Additional information will be provided in upcoming issues of ;login:.

## Go Tournament at Salt Lake City Conference

The First Ever  
USENIX COMPUTER

# GO TOURNAMENT

	A	B	C	D	E	F	G	H	J	K	L	M	N	O	P	Q	R	S	T	
19	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	19
18	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	18
17	+	+	0	@	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	17
16	+	+	+	0	+	+	+	0	+	@	+	+	+	+	+	@	+	+	+	16
15	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	15
14	+	+	0	0	+	+	+	0	+	@	+	+	+	+	+	+	+	+	+	14
13	+	+	@	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	13
12	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	12
11	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	11
10	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	10
9	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	9
8	+	+	+	+	+	+	+	+	+	+	+	+	0	0	0	0	@	+	+	8
7	+	+	0	@	+	+	+	+	+	+	+	+	0	@	@	@	@	@	+	7
6	+	+	@	0	0	+	+	+	+	+	+	+	+	0	0	0	@	+	+	6
5	+	+	0	+	+	+	+	+	+	+	+	+	+	+	0	@	@	+	+	5
4	+	+	+	0	+	+	+	+	+	+	+	+	+	+	0	@	+	+	+	4
3	+	+	@	@	+	@	+	+	+	+	+	+	+	@	@	0	@	+	+	3
2	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	2
1	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	1
A	B	C	D	E	F	G	H	J	K	L	M	N	O	P	Q	R	S	T		

To be held during the Summer 1984 USENIX conference in Salt Lake City, Utah.

### Probable Rules

1. The board will be 19 x 19. This size was chosen rather than one of the smaller boards because there is a great deal of accumulated Go "wisdom" that would be worthless on smaller boards.
2. The board positions will be numbered as in the diagram above. The columns will be labeled 'A' through 'T' (excluding 'I') left to right. The rows will be labeled '19' through '1', top to bottom.
3. Play will continue until both programs pass in sequence. This may be a trouble spot, but looks like the best approach available. Several alternatives were considered: (1) have the referee decide when the game is over by identifying "uncontested" versus "contested" area; (2) limit the game to a certain number of moves; all of them had one or another unreasonable effect.

;login:

4. There will be a time limit for each program. This will be in the form of a limit on accumulated "user" time (60 minutes?). If a program goes over the time limit it will be allowed some minimum amount of time for each move (15 seconds?). If no move is generated within the minimum time the game is forfeit.
5. The tournament will use a "referee" program to execute each competing pair of programs; thus the programs must understand a standard set of commands and generate output of a standard form.
  - a. Input to the program. All input commands to the program will be in the form of lines of text appearing on the standard input and terminated by a newline.
    - (1) The placement of a stone will be expressed as letter-number (e.g. "G7"). Note that the letter "I" is not included.
    - (2) A pass will be expressed as "pass".
    - (3) The command "time" means the time limit has been exceeded and all further moves must be generated within the shorter minimum time limit.
  - b. Output from the program. All output from the program will be in the form of lines of characters sent to the "standard output" (terminated by a newline) and had better be unbuffered.
    - (1) The placement of a stone will be expressed as letter-number, as in "G12". Note that the letter "I" is not included.
    - (2) A pass will be expressed as "pass".
    - (3) Any other output lines will be considered garbage and ignored.
    - (4) Any syntactically correct but semantically illegal move (e.g. spot already occupied, ko violation, etc.) will be considered a forfeit.

The referee program will maintain a display of the board, the move history, etc.
6. The general form of the tournament will depend on the number of participants, the availability of computing power, etc. If only a few programs are entered each program will play every other program twice. If many are entered some form of Swiss system will be used.
7. These rules are not set in concrete ... yet; this one in particular.

Comments, suggestions, contributions, etc. should be sent via *uucp* to harpo!psl or via U.S. Mail

to

Peter Langston  
Lucasfilm Ltd.  
P.O. Box 2009  
San Rafael, CA 94912

## USENIX Software Distribution Tapes

Each year the USENIX Association sends one or two Software Distribution Tapes to its Institutional members. The tapes are sent only after the institution has paid its dues for the year and has sent a copy of its UNIX license(s) and signed copies of the tape release form to the office.

### 1982 Tapes

As of early September, all 1982 Institutional members have been sent their tape, with the exception of eighteen members who have not returned their signed tape release forms.

### Problem with 1982 32V Tape

The Association has been notified that the EMACS contribution from Computer Corporation of America (CCA) that was on the 1982 32V tape contained some code that Bell Laboratories asserts is proprietary. All USENIX Institutional members who were sent a 32V tape are required to stop using and destroy all copies of the version of EMACS that came with the 1982 32V distribution tape. Letters to this effect have been mailed by the office. CCA has offered to 32V tape holders a current version of CCA EMACS to replace the version that has to be destroyed.

### 1983 Tapes

Almost all of the USENIX Association 83.1 Software Distribution Tapes have been mailed to those Institutional members who have fulfilled the requirements mentioned above. As of early September, 63 institutions had not sent a copy of their license to the office and 121 had not returned their tape release form. These institutions and institutions who join the Association during the rest of the year will be sent their tapes as soon as they complete the required steps.

The contributions for the 83.1 tapes were grouped onto four distributions, each containing all the software that could be sent to the holder of a given license. The four distributions are for binary and Mini-UNIX license holders (the tape with no disclosure information (NoD)), Version 6 license holders, Version 7 and 32V license holders, and System III and V license holders. The contributions and the license holders who will receive them are listed in the following table.

USENIX Association 83.1 Tape Contributions

NoD*	V6	V7	32V	S3/5	Contribution
x	x	x	x	x	LOGO implementation Version 3 from Brian Harvey, Lincoln-Sudbury High School and Atari
x	x	x	x	x	a benchmark suite and assorted utilities from Martin Tuori, Defense and Civil Institute of Environmental Medicine
x	x	x	x	x	MENUNIX and some statistical data analysis tools from Gary Perlman of U.C. San Diego and Bell Labs
x	x	x	x	x	a line printer spooler and some random utilities from Yoram Shoham of Geotronics Corp
x	x	x	x	x	some tools for extracting cost information from files and including them in proposals from Geoffrey Kodosky of National Instruments
	x	x	x	x	a restricted UNIX environment for stand-alone utilities and diagnostics from Geoffrey Kodosky

;login:

NoD*	V6	V7	32V	S3/5	Contribution
		x	x	x	Version 7 drivers for Dicomed COM device via DR-11B, modified TM tape driver and Xylogics disk controller and bootstrap code for Xylogics controller from Martin Tuori
		x	x	x	an argument line cracker from Yoram Shoham
		x	x	x	a screen editor based on <i>ed</i> from J. D. Wise of Rice University
				x	System III <i>uucp</i> with "all known bugs fixes" from Steven McGeady of Tektronics

\* No disclosure material

## Call for Tape Submissions

One of the primary goals of the USENIX Association is to provide mechanisms for the sharing and distribution of code within the UNIX community. Because of the somewhat complicated licensing structure imposed by Western Electric, many installations find the overhead necessary in distributing code prohibitively high. USENIX, however, has mechanisms for verifying licences and dealing with the problems inherent in distributing software.

USENIX is eagerly soliciting code or products suitable for (free) distribution to its members. These include device drivers, additions to the C or system programming libraries, bug fixes, enhancements to the kernel or utilities, research projects available for testing, and new tools and packages implemented on UNIX.

Tapes can be mailed directly to the USENIX Office or code can be sent via *uucp* to `ucbvax!g:usenix`.

The contributed software is distributed to Institutional members of USENIX under the conditions of a release form that give the Institution the right to use the software. The release form releases the contributor of all liabilities and does not transfer any property or redistribution rights to the recipient.

For further information, contact the USENIX Office or Deborah K. Scherrer, Tape Committee Chairman, at 415-486-5181.

## USENIX Board Election Nominating Committee Formed

The voting membership of USENIX has approved the composition of the nominating committee for the next election of the Board of Directors. The members are:

Bob Lummis, Albert Einstein College – Chair  
Randy Frank, University of Utah  
Heinz Lycklama, Interactive Systems Corp.  
Peter Langston, Lucasfilm, Ltd.  
Ken Thompson, Bell Laboratories  
Mike Tilson, Human Computing Resources Corp.  
Mike Zuhl, Tektronix, Inc.

This committee has been charged with the responsibility of insuring that there is at least one candidate for each office. The nominating committee will submit its nominations to the Board of Directors by December 15, 1983. These nominations will be announced at the January meeting. Additional nominations from the members at large will close on February 15, 1984.

;login:

## **San Diego Proceedings Still Available**

Copies of the proceedings for the San Diego UNICOM conference are still available from the Software Tools Users Group. They are over 350 pages long and include all papers presented by the speakers as well as reports on many of the presentations.

The price is \$25 per copy, plus \$10 per copy for overseas postage. Send your check or money order made out to "Software Tools Users Group" to:

STUG  
1259 El Camino Real, #242  
Menlo Park, CA 94025

## **Toronto Proceedings**

The proceedings for the Toronto Conference are being prepared. There may be some extra copies available. If you are interested in receiving a copy and did not place your order before the deadline (August 30) you can place your name on a waiting list by contacting the USENIX office. When all the copies that were requested before the deadline have been sent, the office will notify those on the waiting list and the remaining copies will be sold on a first come, first served basis. The price will be \$30 per copy, plus \$5 per copy for overseas postage.

## **New York UNIX Users Group Formed**

A local UNIX users group called Unigroup of New York, Inc. has formed in New York City as a non-profit organization for users and vendors of products and services for UNIX systems. It plans to publish a quarterly newsletter, a local *uucp* directory, and a directory of local companies offering UNIX system products and services. Unigroup may be reached at

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

## **2.9BSD — Berkeley Software Distribution for PDP machines — Now Available**

[This is a slightly edited copy of the announcement of the release of 2.9BSD...Ed.]

Release 2.9 of the Second Distribution of Berkeley PDP-11 Software for UNIX is now available from the Computer Science Division of the University of California at Berkeley. It is a complete V7 system, including the kernel, all standard utilities, and additional Berkeley products. It will run on any PDP-11 with memory management hardware and at least 192K bytes of memory, including the 11/23, 11/24, 11/34, 11/34A, 11/40, 11/45, 11/55, and 11/70. It supports most common disks (RK05/06/07, RL01/02, RM02/03/05, RP03/04/05/06, and emulations of these) and tapes (TM02/03, TM11, and TS11). The distribution is available in source form for holders of Version 7, System III, or System V licenses, or in binary form for appropriate UNIX license holders.



;login:

The major kernel changes since 2.8BSD are:

- Process control, a mechanism for stopping and restarting jobs in foreground or background, and the new reliable signal mechanism that supports it. This is nearly identical to the process control facility of 4.1BSD (the Berkeley VAX UNIX distribution).
- *vfork*, a more efficient version of *fork*.
- Automatic reboots, after crashes or on demand.
- Automatic detection of hardware configuration at boot time, with most of the configuration-dependent addresses and vectors in a single ASCII file.
- Much easier kernel configuration process, with most parameters in one machine description file.
- Numerous efficiency changes. System overhead has markedly decreased in a number of areas: floating point traps (90% decrease), overlay switches (45% decrease), and system calls (22% decrease).
- Many bug fixes; the system is now far more robust.

Other features of the kernel, which were also in the 2.8BSD release, include hashing buffers and inodes, moving buffers and clists out of kernel data space, and the 1Kb block file system. The system supports kernel overlays, allowing it to run on non-separate I/D machines. It also supports user overlays, so that *ex* version 3 can be run, even on non-separate machines.

The Berkeley tty driver is included; it correctly handles erase and kill characters on crt and printing terminals, including correctly backspacing over tabs and control characters.

The enhanced Berkeley implementation of the TCP/IP network facility is included.

Changes to the kernel are conditionally compiled with mnemonic names, making it easy to turn on and off features. This kernel contains contributions from Berkeley's Computer Systems Research Group, the U.S. Geological Survey system, DEC's UNIX Engineering Group, Tektronix (to mention a few).

In addition to the standard Version 7 UNIX system utilities and those related to the the kernel modifications mentioned above, the package includes the following.

The *ex/vi* editor

Two different versions of *ex* are included: versions 2 (currently 2.13) and 3 (currently 3.7). Version 2 will (barely) fit on a PDP-11 with separate I/D and will run on a standard Version 6 or 7 system. Version 3 (the current production editor) is larger and requires either the "user overlay" feature of the supplied kernel or a VAX.

This version improves the terminal driving capabilities of the editor to provide screen editing facilities using intelligent terminals. All terminal capabilities are described in a file so that the editor can drive new terminals after their descriptions are edited into the file. Padding and other requirements of intelligent terminals are dealt with in the editor. The editor is particularly designed so as to be usable as a screen editor on low speed dialup lines with intelligent and unintelligent terminals.

The screen editing facilities of the editor have been expanded so that it is possible to perform editing completely within "visual" or "open" modes.

Other new facilities include parameterless keystroke macros (version 3 only) and word abbreviations; the capability to filter parts of the editor buffer through commands; text registers into which lines may be placed and carried over when editing new files; a "tags" facility for editing large programs that are broken into many files.

While *ex* edits only one file at a time, it remembers its position in the previous file so that it is possible to switch back and forth between two files easily.

;login:

#### The Relational Database management system *ingres*

*ingres* is a database management system that presents data as a collection of tables ("relations"). Facilities include the usual *append*, *delete*, *replace*, and *retrieve*, as well as a variety of database utilities including bulk load and store, dynamic storage structure reconfiguration, full crash recovery, data integrity controls, views, and macros.

*ingres* currently requires separate I/D space and floating point hardware.

#### A shell *csh*

This shell incorporates good features of earlier shells, previous Berkeley shells, and the Version 7 shell. It also has a history mechanism (similar to that of *interlisp*) so that previous commands can be repeated and/or corrected. It provides a user interface to the process control facility.

#### An error message understanding program *error*

This program looks at the output from a program compilation generated by *make* or a compiler, inserts the error messages as comments into the program or programs affected at the proper points, and invokes the editor on those files, positioned at the first error. It understands the error messages generated by the Ritchie and Portable C compilers, *lint*, *make*, Berkeley Pascal, the VAX assembler, the loader, and *f77*. Several useful interactive options exist.

#### A macro package for *nroff/troff*. —me

This macro package is easy to use and especially easy to adapt to different formats. It works with both *nroff* and *troff* and provides a large number of hooks for special requirements. A beginners introduction and a reference manual are also available.

#### A *Mail* program

This mail program uses */bin/mail* to do the actual mailing and concentrates on providing a simple and friendly environment for processing large amounts of mail.

#### TCP/IP

An experimental PDP-11 implementation of the Berkeley VAX TCP/IP networking code is available. Sites that wish to use it will almost certainly have to do some debugging.

#### The Berkeley Network

The Berknet provides a facility for file transfer between machines, network mail, and remote execution of commands in a batch request fashion. It is used at Berkeley to connect a network of UNIX systems. It can be run using terminal ports to connect machines.

The package also includes the instructional Pascal system and other software.

The distribution is provided on two tapes, one of which is bootable and contains the standalone utilities required to bring up a root file system and the kernel. The remainder of the sources, documentation and binaries are in *tar* format.

The distribution costs \$200 US. For further information, including copies of the license agreement and other required forms, and more information on the contents of the distribution, contact

Berkeley PDP-11 Software Distribution — 2.9BSD  
Computer Science Division, Dept. of EECS  
573 Evans Hall  
University of California  
Berkeley, CA 94720  
(415) 642-6258

;login:

## USENIX Association Toronto Conference Notes

The following are reports on the presentations at the Summer '83 USENIX Toronto Conference at the Hilton Harbour Castle Hotel on July 13→15, 1983. The reports have been prepared by a variety of people; the reporter for each session is noted at the beginning of the session.

### Opening Session

Wednesday, July 13, 8:45→10:15

Chair: Michael Tilson

Reporter: *Tom Strong*  
Editor, ;login:  
USENIX Association Office  
P.O. Box 7  
El Cerrito, CA 94530

### Welcome

*Michael Tilson*  
Human Computing Resources Corporation  
10 St. Mary Street  
Toronto, Ontario Canada M4Y 1P9

Mike Tilson was the program chair for the Toronto Conference. He thanked everyone who had helped put the meeting together. He noted that in 1975 he was at a similar meeting with about 50 people, and that the current meeting has over 1200 registrants and takes much more work to organize.

### Welcome and News from USENIX

*Lou Katz*  
President, USENIX Association

The next two meetings are to be January 17→20 in Washington, D.C. and June 12→15 in Salt Lake City. The Washington meeting will feature a trade show hosted by /usr/group and technical conference hosted by USENIX. The Salt Lake City meeting will be a technical meeting and vendor exhibit, similar to this [the Toronto] meeting.

USENIX is an association of people interested in sharing information on UNIX and related topics. It started as a mutual self-help group of academic license holders. In June, 1979, it was registered as an association, and in 1980 it was incorporated.

Several problems have been found with the current Association bylaws, which are based on guidelines from a for-stock corporation. [USENIX is a non-stock, not-for-profit corporation.] Changes under discussion include decoupling licenses from membership and defining four classes of membership. All present were invited to an open meeting with the Board of Directors that evening to discuss the bylaws changes and any other topics of interest.

;login:

## **KEYNOTE ADDRESS: Technology Driven Software vs. Psychology of Users: An Irresistible Force Meets an Immovable Object**

*Michael Lesk*  
Bell Laboratories  
Murray Hill, NJ 07974  
201-582-3000

This talk contained many observations on how programmers can help retain the virtues of UNIX, and described an experiment done at Bell Laboratories to compare menu and command systems.

The experiment was an attempt to test the conventional wisdom that non-programmers prefer menu systems to command systems for retrieving information from a computer. The testing was done with an on-line library catalog, a weather service, and a national news service. The library users consisted of both browsers and people searching for known items. For a library catalog using the Dewey system, it was found that users needed at least four to five levels of menu. The command system, on the other hand, provided a method of entering one or more keywords to search for titles and author names. Here it was found that library users prefer commands and Associated Press Wire users preferred menus. Generally, if you know what to expect, keywords/commands are better; if you do not know what to expect, menus are better.

Some interesting observations were made:

- less documentation is better: not in a desire for terseness but rather because concise, well designed programs do not require a lot of documentation;
- terseness in documentation does not (have to) mean cryptic.
- fast program response is a requirement so that support for spooling, status messages, etc. can be minimal.
- do not ask questions the user views as superfluous to solving the immediate need;
- do not try to understand errors: have the user re-enter the whole thing;
- The UNIX manual used to be small; now some issue masters degrees in *stty*.

Mr. Lesk does not believe in novice modes, although he commented that this requires some sort of training be provided.

There does not have to be a distinction between technology-driven software and the psychology of users, given adequate attention to the needs of the user.

## **UNIX Licensing and New AT&T Product Offerings**

*Larry K. Isley*  
Western Electric  
Guilford Center  
P.O. Box 25000  
Greensboro, NC 27420

Software sales and marketing for UNIX has been assigned to Western Electric and so the Technology Licensing group has been moved from AT&T to Western. The group is now headed by Otis Wilson, and is expected to grow very rapidly. A complete overview of the new structure will be presented at the January meeting.

Source license numbers were presented; Western still refuses to disclose the number of binary UNIX licenses.

Western is offering a total of 16 courses on UNIX at four sites. All but three are available to anyone; the others require a System V source license.

;login:

Three software packages were announced:

S A statistical research package that runs on Systems III and V, 4.1BSD, and V7. S needs at least 15Mb of disk, and 32-bit machines are preferred because they have more memory. Support is not available. Sites that now have S will have to upgrade their license and pay more money in order to receive an updated distribution.

UNIX Writer's Workbench<sup>†</sup> Software

A collection of more than 24 tools to assist writers.

UNIX Instructional Workbench<sup>‡</sup> Software

A package for support of course instructors. It is available in binary form only, and only for CPUs for which Western distributes System V.

No other new programs are available for licensing.

Based on a show of hands, 65% of the conference attendees were from government or commercial installations, 30% were from educational institutions, and 5% were from AT&T or affiliates.

## Programming Tools 1

Wednesday, July 13, 1:30→2:55

Chair: Joseph Yao

Reporter: *Andrew Tannenbaum*

Bell Laboratories

Whippany Road

Whippany, NJ

### Bcc: Runtime Checking for C Programs

*Samuel C. Kendall*

Delft Consulting Corporation

165 West 91st Street, Suite 2A

New York, NY 10024

(212) 624-1149

decvax!genrad!wjh12!kendall

*bcc* is a preprocessor which generates bounds checking code for C programs. It will check array indices, pointers into arrays, and pointers into space allocated by *malloc* ().

*bcc* generates diagnostics that tell the user which subscript or pointer variable overstepped the bounds of which array on which line of your C source program. Quite a bit nicer than the usual core dump.

With *bcc*, your program will take about 3 times longer to compile and ten times longer to run. As this is not a production compiler, speed isn't a primary consideration anyway.

*bcc* can be used with makefiles, but it's a little tricky.

*bcc*'s job in life is to detect bounds check errors, and it's especially valuable, because boundary errors don't always cause a program to fail immediately. *bcc* can be used to test program integrity and uncover errors that would otherwise lay dormant, thereby saving many bleary-eyed hours staring at the CRT screen.

[The speaker has submitted a paper for the conference proceedings...Ed.]

---

<sup>†</sup> Writer's Workbench is a trademark of Western Electric. <sup>‡</sup> Instructional Workbench is a trademark of Western Electric

;login:

## **On Enhancing the Presentation of C Source Code**

*Ronald Baecker, Paul Breslin, Christopher Sturgess*  
Human Computing Resources Corporation  
10 St. Mary Street  
Toronto, Ontario, Canada M4Y 1P9  
decvax!hcr!hcrvax!ron

The authors are involved in a 2 year DARPA project investigating the effect of phototypesetting wizardry on the usefulness of software listings. All sorts of sexy looking variations were presented, involving changes in the font styles, sizes, spacing, shading and formatting of code and comments, as well as use of flow diagrams, timing graphs, snapshots of the authors, etc.

It looks like great fun and such futuristic tools would probably make reading code a lot more pleasant and would be a boon to software packaging. Rob Pike might say that this is what would happen if Berkeley rewrote *cb*.

## **On-line Manual System for Software Development on UNIX**

*Osamu Nakamura, Jun Murai*  
Department of Mathematics  
Keio University  
3-14-1 Hiyoshi  
Kohoku, Yokohama 223 Japan

The On-line Manual System is a set of database manipulation routines that can be integrated into systems which browse through manuals like the UNIX user's manual.

UNIX man pages have always been designed and formatted to be bound and used in book form.

This system allows integration of interactive documentation browsing into software systems. The normal UNIX man command is augmented with keyword searching and cross referencing. Man pages are stored in a hashed database that make for easier and more efficient access.

[The speaker has submitted a paper for the conference proceedings...Ed.]

## **cdb - A C Source Level Debugger**

*Michael Farley, Paul Kunkel, Trevor Thompson*  
Mark Williams Company  
1430 W. Wrightwood  
Chicago, IL 60614  
(312) 472-6659

*cdb* is a screen-oriented debugger for C programs, currently running on the DEC Rainbow 100, though several ports are planned.

Four windows are used, to

1. display the source being executed,
2. inspect data,
3. display output,
4. log trace points.

*cdb* is based on a heavily modified compiler that implements full C. The compiler provides extensive symbol information and the debugger runs as a daemon process separate from the debugged code. This gives the debugging benefits of an interpreter without the heavy speed overhead. You may debug individual functions and jump around fiddling variables and executing statements of your choice.

;login:

It is possible to link *cdb* *cdb* is designed to fit in 64Kb address spaces, so you folks with split I/D 11's can have hopes that *cdb* will eventually run on your machines.

[The speaker has submitted a paper for the conference proceedings...Ed.]

## UNIX Implementation 1

Wednesday, July 13, 1:30—3:00

Chair: Michael Blake-Knox

Reporter: *Tom Strong*

### TUNIS: A Portable, UNIX Compatible Kernel written in Concurrent Euclid

*R. C. Holt, M. P. Mendel, S. G. Perelgut*  
Computer Systems Research Group  
Sandford Fleming Building  
10 King's College Road, Room 2001d  
University of Toronto  
Toronto, Ont. Canada M5S 1A4

TUNIS is a portable kernel that is compatible with UNIX. It is a descendant of Version 7 UNIX and runs Version 7 object modules.

TUNIS was written in a language called Concurrent Euclid (CE), a Pascal-based language that comes from the Euclid language, and was designed for developing verifiable systems software. CE has additional features for systems programming, concurrency, and separate compilation of modules. CE is implemented as a small portable compiler with a replaceable code generator. Code generators are available for the PDP-11, VAX, M68000, M6809, Intel 432, NS16000, and IBM 370.

The TUNIS "nucleus" is equivalent to the UNIX kernel. It consists of a CE program supported by a kernel written mostly in assembly language. The nucleus has layers of CE modules that provide operations that are used by successive layers. Each layer treats the operations provided by a lower layer in terms of an *abstract machine* description of the service performed, not in terms of how the service is implemented. The major layers include user, file, memory, and device managers. As an example of the layering used, the file and user manager use only virtual memory while the device manager uses only physical memory.

The TUNIS nucleus currently runs on PDP-11's and is being tested on a VAX.

[The speaker has submitted a paper for the conference proceedings...Ed.]

### The Sol Operating System

*Michel Gien*  
Pilot Project Sol  
c/o INRIA  
B.P. 105  
78153 — Le Chesnay Cedex France  
(3) 954 90 21  
decvax!mcvax!vmucnam!mg

Sol is a project to build a portable software engineering system. It has a portable kernel and is based on UNIX. Sol was developed in ISO standard Pascal; this was found to be lacking and so some implementation-specific Pascal extensions were defined. (C was very unknown in France in 1979 when the project started.)

;login:

Sol is not simply a translation of the UNIX C kernel to Pascal. It is a complete re-engineering of all the internals with a view towards better modularity and portability, while making efficient use of Pascal.

Sol was designed to be compatible at the system call level with Version 7 UNIX. Some System 3 extensions have also been added. Sol includes all the basic utility programs of UNIX, [many/most/all?] of which are written in Pascal.

Sol is available as a commercial product, including the source code. There is a validation center to coordinate validation efforts.

## **An Implementation of UNIX for the Intel iAPX 286**

*P. L. Barrett*  
Senior S/W Engineer  
Intel Corporation — HF2-1-800  
5200 N.E. Elam Young Parkway  
Hillsboro, OR 97123  
503-640-7335  
icalqa!omsvax!plb

This talk gave an overview of the iAPX 286 microprocessor and told how the UNIX kernel was implemented on it. The iAPX 286 has integrated memory management and protection. Four levels of protection are provided for each task. Up to 1 gigabyte of virtual memory per user are available and up to 8 [16?] megabytes of real memory. The iAPX 286 uses two buses: Multibus for I/O and a local bus for access to memory.

The iAPX 286 uses four segment registers to point to a table which contains the base addresses of the code segment, data segment, stack segment, and extra segment. Virtual address space is controlled by a series of memory-based mapping (descriptor) tables. These contain the base address, length, access rights, and status of every segment in the system.

In the iAPX 286 UNIX kernel implementation, each process has its own local descriptor table and the kernel has a global descriptor table, allowing it access to the whole machine.

Memory is allocated in segments with a granularity that is a multiple of the disk block size - 1Kb. Segment growth can be in place or done by a copy or a swap.

## **UNIX at Data General**

*Wayne McLaren*  
Data General Corporation  
62 T. W. Alexander Drive  
Research Triangle Park, NC 27709  
919-549-8421

Data General has implemented a System III version of UNIX as a "hosted system" on their 32-bit computers. It runs on top of their AOS/VS operating system as a user application and uses its file system and other existing AOS/VS features. Over 100 commands are supplied, including *make*, *lex*, *yacc*, *nroff*, and *scs*. The system was, at the time of the conference, in  $\beta$ -test.

Data General ran into several problems. There were things the native operating system did not allow, such as groups, general file names with the full ASCII character set, and *fork* (although they did implement *vfork* which is adequate except for the shell!).



;login:

## Programming Tools 2

Wednesday, July 13, 3:30→5:00

Chair: Ian Darwin

Reporter: *Andrew Tannenbaum*

### VCHK - A Maintenance Program for UNIX file Hierarchies

*Scott Bryan*  
UniSoft Corporation  
2405 4th Street  
Berkeley, CA 94710  
(415) 644-1230

VCHK is used to keep track of the integrity of UNIX file system directory structures. It makes sure that files are in the right place with the right modes, ownerships, links, and checksums.

VCHK can generate a snapshot of your file hierarchy and then use that snapshot as a basis for comparison to ensure the integrity of your system at a future time, sort of like a makefile. VCHK can fix problems with modes, ownerships and links, but would, of course, have trouble fixing corrupted file contents.

[The speaker has submitted a paper for the conference proceedings...Ed.]

### Enhancing *make*, or Re-inventing a Rounder Wheel

*Edward Hirtel*  
Zehntel, Inc  
2625 Shadelands Drive  
Walnut Creek, CA 94596  
(415) 932-6900, ext. 364  
decvax!sytek!zehntel!zinfandel!ed

*make* has been modified to handle files in separate directories. This is especially useful when a group of people must all use the same source files in various stages of development and testing. %cd, %pushd, and %popd have been implemented, much like in *cs*. A *cs*-like %foreach file directive is also included.

A macro facility has been added that includes conditionals (like the C preprocessor's #if), macros, expressions, and tty input to allow users to control makefiles in progress. There are lots of other frills, like support for multiple shells, temp files, locking and logging.

[The speaker has submitted a paper for the conference proceedings...Ed.]

### Mm4 - *make* with *m4* for Maintaining Makefiles

*Martin J. McGowan III, William L. Anderson, Allen H. Brumm*  
Product Assurance Department  
Computer Consoles, Inc  
97 Humboldt Street  
Rochester, NY 14609  
(716) 482-5000

McGowan "offers us a *make* we can't refuse."

He uses a *make.m4* file which is run through *m4* to produce a makefile for your system. Maintaining the *make.m4* file is allegedly more straightforward than maintaining a makefile. *Make.m4* files

;login:

contains the BITMELD macros (bin include test man ed/etc lib doc) that generate common *make* production rules.

A utility called *minc* (make includes) produces make dependencies for your system (it *greps* your files for #includes).

McGowan also talked about TIPO (Tools, Inputs, Primaries and Outputs). The TIPO program generates lists of these components in your system, which should be very satisfying to structure fiends.

If nothing else, this talk did introduce a few new and cryptic acronyms to a world which is so lacking in that area.

## Using *make* Effectively

*Robert E. Novak*  
Pyramid Technology Corporation  
2471 East Bayshore Road  
Palo Alto, CA 94303  
(415) 494-2700

Novak's important main point was that hackers group all kinds of diverse variables and definitions into one big .h file which is typically included in many .c files because otherwise it's too troublesome to generate the makefile dependencies. He claims that you should generate your makefiles automatically and let your file structure reflect the structure of your problem more closely.

A script that *greps* for #includes accomplishes this task. This also solves the age old problem of modifying a .h file and having to regenerate all the .o files even though they weren't really affected by the change.

## UNIX Implementation 2

Wednesday, July 13, 3:30→4:55

Chair: Phyllis Bregman

[The report on this session was not submitted; the abstracts are reprinted below...Ed.]

## **gprof: A Call Graph Execution Profiler**

*Marshall Kirk McKusick*  
1616 Oxford St.  
Berkeley, CA 94709  
ucbvax!mckusick

Large complex programs are composed of many small routines that implement abstractions for the routines that call them. To be useful, an execution profiler must attribute execution time in a way that is significant for the logical structure of a program as well as its textual decomposition. This data must then be displayed to the user in a convenient and informative way. The *gprof* profiler accounts for the running time of called routines in the running time of the routines that call them. The talk describes the design and use of this profiler, which will be available on the next Berkeley distribution.

;login:

## **File System Considerations in a Multiple Processor UNIX Environment**

*Ed Patriquin*  
Convergent Technologies  
3055 Patrick Henry Drive  
Santa Clara, CA 95051  
(408) 980-0850

Traditionally UNIX has been run on single processor systems like the PDP 11/70. Even when UNIX moved to the world of multiple processors, it only ran on tightly coupled multiple processor systems. The MegaFrame is a loosely coupled multiple processor environment with each processor running its own copy of UNIX. These processors are connected by a high speed system bus. There are also processors in the system running CTOS, a Convergent proprietary operation system.

UNIX, in its original form, keeps a large amount of information concerning the state of the file system in in-core data structures. In a multiple processor environment this is not a feasible solution.

To provide better performance and allow sharing of file resources, the UNIX file system was offloaded and put as service process under CTOS on all File Processors which have UNIX file systems on them. This presented several major architectural problems within UNIX.

The UNIX file system was redesigned and reimplemented to run under CTOS. To make the program interface properly with CTOS, the file system was made into a message based server process. This means that the file system is now single-threaded in a multi-processor environment. A parallel server process which uses the same data structures as the regular file system but does not modify these data structures was added. Any operation which will modify a data structure is sent to the single-threaded file system server for completion. This allows the high volume traffic, such as normal reads and writes, to flow quickly around the single-threaded bottleneck. This also eliminates concurrency problems that arise with multiple threaded operation.

## **A High Performance Implementation of UNIX for the IBM Series/1**

*Michael E. Wilens*  
Computerized Office Services Incorporated (COSI)  
313 North First Street  
Ann Arbor, MI 48103  
(313) 665-8778

COSI undertook the porting of UNIX System III to the IBM Series/1 as part of a joint effort with the CMI Corporation. The resulting system is called SERIX (for Series/1 UNIX) and will be announced within the next few weeks.

This talk focuses upon the capabilities of SERIX in general including several interesting aspects of UNIX on the IBM Series/1:

1. The Series/1 is an unusual machine with a somewhat unusual architecture and instruction set. A lack of translation registers leads to unique concepts such as twilight zone cache management.
2. A 'C' compiler capable of generating extremely efficient Series/1 code was developed. The compiler optimizes usage of hardware stacking instructions, supports inline insertion of assembly code which is subsequently optimized, and generates extremely efficient (in both time and space) object code. As a result, less than 500 lines of assembly code was written for SERIX.
3. The I/O channel orientation of the series/1 was exploited by interleaving at both the controller and/or device levels, significantly increasing both the bandwidth (hence speed) of access to disk and the maximum size of file systems.

;login:

4. Real-time extensions jointly developed by General Motors and COSI were added.

## **The Use of the Z80 I/O Processor by the TRS-XENIX Operating System**

*Jerry Dunietz, Robert Powell*  
Microsoft Corporation  
The Microsoft Building  
10700 Northup Way  
Bellevue, WA 98004  
(206) 828-8080  
decvax!microsoft!jerryd

The Radio Shack Model 16 computer system uses two processors — a Motorola Mc68000 16:32 bit processor and a Z80 8 bit processor. Only the 8 bit processor has access to I/O devices, such as the console/keyboard, built-in serial ports and parallel port, built-in floppy drive(s), and hard disk drive(s). The Z80 can access the Mc68000 memory and can generate three distinct interrupts to the Motorola processor. The TRS-XENIX system was designed to take advantage of this system architecture. The features dependent upon the dual processor design generally improve the system in one of two areas — I/O performance and configurability.

This talk describes the design of these features in some detail. Points of interest include:

- minimizing the Mc68000 interrupt burden in high-speed serial I/O without delaying XOFF processing;
- designing one interface between the Z80 and Mc68000 to handle all character-at-a-time devices, whether serial or parallel;
- designing one interface between the Z80 and Mc68000 to handle multiple sizes and formats of hard and floppy disks;
- making the TRS-XENIX system dynamically adjust to the format of an inserted floppy disk;
- making the TRS-XENIX system auto-configure for the root and swap devices and their sizes;
- detecting the removal of a currently open floppy disk;
- for Microsoft's internal development and debugging use, providing a mechanism by which the Z80 may detect the crash of the Mc68000.

## **Virtual Memory Management in GENIX**

*Laura Neff*  
Microcomputer Systems Division  
National Semiconductor Corporation  
Santa Clara, CA  
menlo70!nsc!lneff

We have recently ported UNIX to an NS16032-based system. The port is based on 4.1BSD UNIX. Our operating system is called GENIX. It supports the virtual memory architecture provided by the NS16082 Memory Management Unit.

A GENIX process has a virtual address space of up to 16 megabytes and is mapped by a two-level hierarchy of page tables. Both process pages and second level page tables can be paged out of an faulted into memory.

The kernel's address space is mapped with its own set of page tables. Because the user and kernel page tables have the same structure, the kernel manages its own address space with the same algorithms it uses to manage user address spaces.

;login:

Sharing of texts is done by mapping executable files. When a file is executed for the first time, the kernel creates a file map for it using level 1 and 2 page tables. The process's page table entries are then set up using the file's map as a template.

## **User Interface 1**

Thursday, July 14, 8:45→10:00

Chair: Barbara Arlow

Reporter: *Andrew Tannenbaum*

### **The Interface Arsenal: Software Tools for User-Interface Development**

*Gary Perlman*  
Bell Labs, 5D-105  
600 Mountain Avenue  
Murray Hill, NJ 07974  
(201) 582-3624  
ucbvax!mhb5b!gsp

Most programs have poorly designed user interface because programmers either don't have resources to implement them, don't have knowledge to implement them, or don't have interest in implementing them.

The Interface Arsenal is a set of C function libraries which make it easier to design systems with good, standard, user interfaces.

Library functions include run-time strings allocation, list processing, function key management, file handling and menu handling.

A library for type and range checking for form filling is planned.

### **A User Interface Management System**

*W. Buxton, M.R. Lamb, D. Sherman, K.C. Smith*  
University of Toronto  
Rm. 2002, Sanford Fleming Building  
10 King's College Road  
Toronto, Ontario M5S 1A4 Canada  
(416) 978-6320

A tool called MENULAY (menu layout) was developed to allow non-programmers to design interactive graphics programs. MENULAY consists of tools used to develop the programs and routines which handle the various esoteric forms of I/O.

A sexy demo showed a designer using MENULAY to create and run a real time demonstration of the chemical titration process, complete with color and sound.

MENULAY generates C code which can be integrated with application specific routines.

;login:

## Talking to UNIX — Some Experience with Speech Input

*Martin Tuori*  
DCIEM, Toronto  
Box 2000  
Downsview, Ontario M3M 3B9 Canada

Tuori presented a videotape which showed what can be expected from a \$2000 Threshold Technologies Auricle-1 voice input device and a \$35000 Verbex Model 3000 voice input device.

The Verbex accepted continuous speech while the Auricle-1 only accepted discrete words, but the presentation emphasised the simpler unit because the more complicated one hadn't been fully integrated into their UNIX system.

The Verbex was shown running stand-alone recognizing streams of five numbers spoken in a continuous phrase.

The Auricle-1 was shown hooked up to a UNIX terminal where the voice input device had access to standard input, parallel with the keyboard.

Voice input has a variety of applications, from input for the physically disabled to input for people who have their hands full on assembly lines. In either of these cases, the limited vocabulary of an inexpensive system can be used to great advantage.

I'll have to get used to the idea of talking to UNIX. I can see my bleary-eyed self yelling at my terminal, "A D B foo! blech dollar B! colon R! break break break dammit!" (dammit: not found)

I just hope I have my own office by the time this technology becomes common.

[The speaker has submitted a paper for the conference proceedings...Ed.]

### UNIX Implementation 3

Thursday, July 14, 8:45→10:00

Chair: David Martindale

[The report on this session was not submitted; the abstracts are reprinted below...Ed.]

### A General-Purpose Object-File Format

*Brian Lucas, Heinz Lycklama*  
Interactive Systems Corporation  
1212 Seventh Street  
Santa Monica, CA 90401  
(213) 450-8363  
decvax!yale!ima!heinz

In porting the UNIX system to various 16- and 32-bit computers and in developing cross-compilers that run on these computers but generate code for other machines, Interactive has found the object-file formats used by UNIX System III and System V, including the new *coff* format, to be inadequate. As a result, Interactive has designed a new general-purpose object-file format meant to achieve the following objectives:

- (1) Support on all currently popular 16-bit and 32-bit computers.
- (2) Low overhead on small machines.
- (3) Common set of utilities that deal with object-file formats.
- (4) Provision for compatible extensions.
- (5) Independence from byte and word order.

;login:

This paper describes the details of the object-file format that achieves these objectives.

## **A User Information Data Base for UNIX**

(What to do when *etc/passwd* just isn't enough)

*Clyde W. Hoover*  
Computation Center  
University of Texas at Austin  
Austin, TX 78712  
(512) 471-3241  
eagle!ut-ngp!clyde

The increasing amount of data desired for the successful administration of UNIX systems in a large academic computing environment has outgrown the standard password file format. This need is best met by supplanting the password file with a user information data base system.

As the usage of UNIX at the University of Texas Computation Center grew, it became necessary to maintain extended accounting and administrative information for each user of our UNIX systems.

A number of methods for storing and maintaining this information were examined. The one 'unused' field in the standard UNIX password file format was found to be inadequate for this purpose; the large number of programs that depend on this format effectively disallowed changing it; the problems involved with maintaining synchronization between multiple accounting information files (*etc/passwd* plus a supplemental data file), precluded that approach.

A User Information Data Base (UDB) was the best solution to this problem. This data base contains all the per-login-user information and replaces *etc/passwd* for system accounting and access purposes.

The standard password file still exists (and is generated from the information in the UDB) to provide compatibility with existing programs, though it is ultimately planned that the UDB will totally replace it for all applications, including user id/login name mapping (most commonly done by programs like *ls*).

Programs that employ the standard UNIX password file access routines can use the UDB without source changes. A library of access and manipulation routines are provided for directly using the UDB. There is also a suite of programs to keep the data base in good working order, and an interactive editor for data base maintenance and account control.

## **Z — A High Performance Raster Graphics Package for UNIX Operating Systems**

*Steve Daniel*  
Microelectronics Center of North Carolina  
P.O. Box 12889  
Research Triangle Park, NC 27709  
(919) 541 7285  
decvax!mcnc!swd

Z is an outgrowth of the VLSI-CAD tools effort of the Microelectronics Center of North Carolina (MCNC). Several of the tools under development at MCNC use high performance raster-graphics processors. When their development was started no high quality, high speed, interactive, raster graphics packages were available for UNIX. A CORE graphics standard system was implemented and rejected because it was too slow and not well suited to the needs of interactive raster-graphics programs. (This CORE system, called LEO was described at the Jan 1982 USENIX conference.)

Z differs from the proposed CORE standard in several ways. First, we view the CORE system as oriented towards vector-graphics devices. CORE's concept of a picture segment maps very well onto

;login:

the display lists of vector devices. The translation, scaling, and rotation of segments is easily accomplished by many vector devices. Raster devices have trouble implementing these operations.

Instead, raster devices provide such features as infinite picture complexity, the ability to erase regions of the screen, the ability to copy portions of the screen from one place to another, and the ability to manipulate the bit planes individually. Z provides access to these hardware capabilities.

Z is designed to provide support for painting on the screen, rather than creating and manipulating a graphical data base. Not only does this approach map easily onto raster-graphics hardware, but it also eliminates the time-consuming and error-prone task of mapping between the user program's graphical data base and the graphics system's data base.

Z is fast. Numerous algorithmic optimizations have been applied to make the code fast and efficient.

## Attaching an Array Processor in the UNIX Environment

*Clement T. Cole*  
Massachusetts Computer Corporation\*  
543 Great Rd.  
Littleton, MA 01460  
genrad!masscomp!clemc

In this report, the work necessary to attach a commercial array processor, the Floating Point Systems FPS-164, to a VAX 11/780 running 4.X BSD UNIX is described. The FPS-164, its surrounding development system, and the interaction with potential application programs are also presented.

\*Work done while a member of the CAD group of UC Berkeley.

## User Interface 2

Thursday, July 14, 10:30→11:40

Chair: Robert Pike

Reporter: *Andrew Tannenbaum*

## The Edit Shell — Combining Screen Editing and the History List

*Joseph L. Steffen, Michael T. Veach*  
Bell Laboratories  
Naperville, IL 60566  
(312) 979-5381  
ucbvax!ihnss!ihuxs!steffen

In the past, the shell has offered less than optimal editing of command lines. Either erase and kill with *sh*, or better, the line edit oriented *csh* history mechanism. *esh* (edit shell) improves upon this mechanism by offering a "what you see is what you get" screen editor style interface to the shell. You can move up and down in the history list and make changes using the movement and change commands of either *vi* or *emacs*.

You could say that *sh* is like editing with *cat*, *csh* is like editing with *ed*, *ksh* is like editing with *vi* or *emacs*.

Considering the amount of time we spend typing at the shell, this seems like a dandy idea. The edit features of *esh* have been incorporated into *ksh*, an *sh* compatible shell which is described in the next report.



;login:

[The speaker has submitted a paper for the conference proceedings...Ed.]

## **KSH – A Shell Programming Language**

*David Korn*  
Bell Laboratories  
Murray Hill, NJ 07974  
(201) 582-7975  
inhp4!mhb5b!dgk

*ksh* (Korn shell) is upward compatible with the Bourne shell (*sh*). It includes almost all the features found in Berkeley's *cs*h and Joe Steffen's *esh*.

Notable additions to *sh* include a history mechanism which continues across login sessions, *esh* screen-oriented editing features as described in the *esh* review above, arithmetic a la *cs*h @, aliases, shell functions (procedures), *cs*h style job control, and menus.

*ksh* is faster than other shells, and it runs on 4.1BSD and kin as well as Systems III and V.

A personal plug: I use *ksh* on my 4.1BSD system, it's a fine piece of work. It has the power of *cs*h, compatibility with *sh*, speed, and some tricks of its own. What more could a hacker want?

[The speaker has submitted a paper for the conference proceedings...Ed.]

## **A Simple Window Management Facility for the UNIX Timesharing System**

*David Mankins, Daniel Franklin*  
Bolt Beranek and Newman, Inc.  
50 Moulton Street  
Cambridge, MA 02138  
(617) 491-1850

A window facility was implemented under the BBN C70's UNIX Version 7. The system may have output in progress in several windows while the user is typing in yet another window. After one person-month of work, a system exists which uses process groups to multiplex a single terminal I/O data structure to talk to the group of windows on a bit mapped terminal like the BBN BitGraph.

The system requires minimal kernel changes and is quite efficient. It does not require additional overhead on terminal I/O until you switch windows. Presumably, this also means that there is no integrity checking on the output and screen control data. A privileged user mode "window manager" handles all window manipulation.

[The speaker has submitted a paper for the conference proceedings...Ed.]

## **Compilers and Languages 1**

Thursday, July 14, 10:30→11:40

Chair: Jean Wood

[The report on this session was not submitted; the abstracts are reprinted below...Ed.]

## **A New Portable Compiler For XENIX**

*Ralph Ryan, Hans Spiller, Dave Weil*  
10700 Northup Way  
Bellevue, WA 98004  
(206) 828-8080 ext. 226  
decvax!microsoft!ralphr

A new portable compiler has been developed which allows compilation of multiple languages to multiple machines. This paper presents the design goals that were set up for the compiler, and discusses the success of the architecture in meeting these goals. The design of the intermediate languages is discussed including:

- support of C, Pascal, FORTRAN, and BASIC front ends.
- language and machine independence.

A case study of the first re-target is presented, including:

- problems in porting from a flat address space to a segmented architecture.
- a comparison of portability between this compiler and the portable C compiler (PCC).

## **Objective C: Programming Smalltalk-80 Methods in C Language**

*Brad J. Cox*  
Productivity Products Incorporated  
Sandy Hook, CT 06482  
(203) 426-1875

The Objective-C[1] compiler provides message/object programming (a la Smalltalk-80[2]) within conventional environments such as UNIX[3]. This compiler and its library, while similar in purpose to an earlier version called OOPC[4], are a totally new design and implementation based on direct experience with Smalltalk-80.

The compiler turns Objective C language, a superset of standard C, into programs consistent with the run-time semantics of Smalltalk-80. The compiler makes full use of the standard C code production chain and the UNIX run-time environment. A program translator pass is added just after the C preprocessor to turn Objective C language into standard C source. It does this by performing a full parse on the incoming language, so it is capable of accurate compile-time diagnostics and can move several speed-critical operations from run-time to compile-time. The result is a language that offers the productivity benefits of Smalltalk-80 while retaining the efficiency/portability benefits of C language.

[1] Objective-C is a trademark of Productivity Products, Incorporated.

[2] Smalltalk-80 is a trademark of Xerox Corporation.

[3] UNIX is a trademark of ATT.

[4] Brad Cox, *The Object Oriented Precompiler*, SIGPLAN Notices, January 1983

;login:

## Compilers on the NS16000

*Jay Zelitzky, Sunil Srivastava*  
National Semiconductor  
Microcomputer Systems Division  
135 Kern Avenue  
M/S 7C-265  
Sunnyvale, CA 94086  
(408) 733-2600  
menlo70!nsc!myunive  
menlo70!nsc!sunil

As part of our support for the NS16000 UNIX system, we have ported the 4.1 BSD C compiler to the NS16000 and written a compiler for EPascal, a Pascal extension developed at National Semiconductor. These compilers have been designed to be used in a systems development environment. While C by nature supports systems programming, Pascal has been extended to do so. The design criteria for these compilers included the following features: good code quality, separate compilation, inter-language module linkability. These continue to be important in our plans for further development. We use the results given by some popular benchmarks to illustrate the strengths and weaknesses of the compilers and the chip set.

There are many similarities in the C and EPascal compilers' output although their internals are quite different. Both compilers generate assembly language files. Both use full 32-bit representation for integers and 64-bit IEEE standard representation for floating point numbers. The two compilers take advantage of the indexed addressing modes of NS16000 to do array indexing. The two compilers follow the same conventions for saving and restoring registers, so routines of the two languages can be linked without risk of destroying the contents of the registers.

The C compiler is an extensively modified version of the VAX C compiler. Functions returning structures are handled differently; return values are moved directly to their final destinations. The compiler efficiently supports array, pointer, and structure manipulation by using the NS16000's indexed addressing mode. It optimizes assignments to a register variable by using the register variable as a temporary register. This saves a final assignment back to the register variable since the result is already there. The C compiler uses a modified version of the VAX c2 optimizer to do some additional peephole and branching optimization.

## UNIX Implementation 4

Thursday, July 14, 1:30→3:00

Chair: A.R. White

Reporter: *Dennis Ritchie*  
Bell Laboratories  
Room 2C517  
Murray Hill, NJ 07974

;login:

## Running the UNIX Kernel in User Mode

*Michael Lutz*  
Computer Systems Consultant  
103 Fox Chapel Road  
Henrietta, NY 14467  
(716) 359-2264  
rochester!ritcu!mjl

*Michael Shon*  
GCA, Tropel Division  
60 O'conner Road  
Fairport, NY 14450

The authors wished to run UNIX on a MC68000 EXORmacs system, but had to solve the following problem: its memory management unit does not support mapping in kernel mode. UNIX is written to expect per-process data to appear at the same place in the address space when each process is running kernel code. On most machines, this can be done by adjusting the kernel map when switching processes, a cheap operation. The authors reject two possibilities for solving their problem: rewrite the kernel to use a global pointer to the per-process data (doesn't solve the problem); copy the per-process data to a fixed location when changing processes (too slow). They opt for running the kernel code in user mode, to take advantage of mapping, and arrange for the relatively few functions that actually require kernel privileges to trap to true kernel mode. They seem to have made the right choice, and their paper discusses the situation well.

[The speaker has submitted a paper for the conference proceedings...Ed.]

## UNIX Support for Guaranteed Real-Time Processing

*Douglas J. Ross*  
ANDYNE Computing Limited  
221 King Street, East  
P.O. Box 1496  
Kingston, Ontario K7M 7A1  
(613) 548-4355  
decvax!utzoo!dciem!rds!djr

*M. Martin Taylor*  
DCIEM  
P.O. Box 2000  
Downsview, Ontario M3M 3B9  
(416) 635-2048  
decvax!utzoo!dciem!mmt

This paper describes a scheduler that provides a guarantee of timely service to user processes that are willing to engage in the necessary negotiations, and to refrain from using more resources than they have contracted for. Their approach is to be recommended, especially for projects like laboratory automation. It is not without cost; it depends on various primitives of MASCOT, for example. Still, the original implementation runs on the PDP-11/34, so the additions cannot be too large. The "reservations desk" to which processes apply for service guarantees is still primitive and ultraconservative.

The authors correctly observe that attempts to provide real-time service based on simple task priorities may unexpectedly fail because of bad luck in timing or unforeseen interference between tasks. On the other hand, a guarantee-based system is no panacea; it can be as disconcerting for a reservation to be denied as for a deadline to be missed. But if the reservations are placed well in advance, there is still a chance for intelligent intervention.

[The speaker has submitted a paper for the conference proceedings...Ed.]

;login:

## **Performance Effects of Disk Subsystem Choices for VAX Systems Running 4.2BSD UNIX**

*Bob Kridle*  
mt. Xinu  
2405 Fourth Street  
Berkeley, CA 94710

*Kirk McKusick*  
Computer Systems Research Group  
EECS  
University of California  
Berkeley, CA 94720

The authors tested the disk throughput of 4.2 BSD with applicable subsets of this cross product: 750 vs. 780 processors, DEC UDA-50 vs. Emulex SC7x0 vs. SI 9900 controllers, DEC RA81 vs. Fujitsu "Eagle" disks. All of the tests were made on otherwise quiet systems with large (4KB and 8KB) block sizes; thus they approach the best possible transfer rates achievable with file system I/O, and the numbers are impressive, ranging from about 200 to 800 Kb's. There are many tables and many numbers, and not all the results are explained, nor point to unequivocal choices. It looks like the SC7x0/Eagle combination was fastest on reading, followed by the UDA50/RA81, and then the SI9900. For writing, the Emulex was still fastest but the SI controller pulled ahead of the UDA50.

The authors conclude that all of the combinations they tested probably provide adequate I/O bandwidth. They observe, but not forcefully enough, that the tests were performed in idealized situations. The large block sizes and concentrated file allocation available in 4.2 BSD, combined with the lack of competition from other programs, must have made seek time much less important than in the usual UNIX system, so the measurements reported may not be particularly applicable in other situations.

[The speaker has submitted a paper for the conference proceedings...Ed.]

## **UNIX System V and 4.1c BSD**

*John Chambers*  
Office of Academic Computing  
449 Admin. Bldg. and Biostatistics  
University of Texas Medical Branch  
Galveston, TX 77550  
(409) 761-1813  
decvax!eagle!ut-ngpljbc

*John Quarterman*  
Computation Center  
University of Texas at Austin  
Austin, TX 78712  
(512) 471-3241, ext.252  
decvax!eagle!ut-ngpljsq

This paper is a telegraphic discussion of features of, and differences between, the cited systems. A novice asked to choose between the two versions of UNIX on the basis of this paper would probably be confused, because it assumes considerable knowledge of what has been available traditionally. The connoisseur will enjoy studying it as a snapshot of evolution in action; like the young Darwin, the authors annotate the differing tortoise's shells and finch's bills that inhabit the separated Galapagos of Murray Hill and Berkeley. Unlike Darwin, the authors have not yet formulated a theory to account for their observations.

The paper's greatest virtues are that it tries to mention all the differences, and that it is fair-minded. Its encyclopedic nature contributes to its greatest fault, which is that it does not distinguish well enough between important and small differences; sometimes it says "this is important," but all topics tend to receive equal space. It is generally accurate, but all statements regarding the Blit terminal and its successor, Teletype's 5620, should be ignored.

[The speaker has submitted a paper for the conference proceedings...Ed.]

;login:

## Compilers and Languages 2

Thursday, July 14, 1:30→2:55

Chair: Jean Wood

Reporter: *Andrew Tannenbaum*

### **A General Purpose Programming Language with an Embedded Data Base Interface**

*Joel Isaacson*  
Sarris Computers  
c/o New York Blood Center  
310 E. 67 Street  
New York, NY 10021  
(212) 229-7425  
harpo!floyd!cmc12!presby!joel

QL is an interactive, interpreted general purpose language with database capabilities. It has a C-like syntax with data types extended for database manipulation. Aside from the usual C data types, QL supports strings (not just character arrays), dates, associative arrays, forms (database tuples), and data base keys. The arrays can be indexed by strange types like string or float.

You can mix modes; for example, you can add an integer to a date and the result will be the date with integer number of days added to it.

There is no type declaration, types are assigned dynamically as in APL. There are generous run-time diagnostics in the interpreter.

### **Turing: A New General Purpose Computer Language Under UNIX**

*J.R. Cordy, R.C. Holt*  
Computer Systems Research Group  
University of Toronto  
Toronto, Ontario, Canada  
(416) 978-8715

Turing is a general purpose language which will be used as the first computer language taught to University of Toronto students in the fall of 1983. It is a generalized form of Pascal with string handling, modules, unions, and dynamic arrays. It has no machine dependent features and is easily ported to new machines. Its features support formal program verification.

Turing seems to have less syntactic trash than even C, the canonical "hello world" program is written as:

```
put "hello world"
```

In Turing, no "main" declaration and no brackets. It has a rich set of operations.

[The speaker has submitted a paper for the conference proceedings...Ed.]

;login:

## A UNIX Tool Kit for Making Portable Compilers

*Andrew S. Tanenbaum, Hans van Staveren, E.G. Keizer*  
Department of Mathematics and Computer Science  
Vrije Universiteit  
Amsterdam, The Netherlands  
31 (20) 548 2410  
decvax!mcvax!vu44!ast

Until recently, new compilers for new computers were usually written from scratch. This talk described a tool kit to bring mass production to compiler writing. The problem of compiler writing is decomposed into machine dependent parts, language dependent parts, and independent parts. The independent intermediate code can be processed by universal assemblers and optimizers, which may be used by all compilers produced by the tool kit. The mass produced parts can be "tweaked" for their own speed and size and for the speed and size of the output code. Writing a compiler for a new language requires one new module for the language front end, adding a new machine requires writing two tables, one for the back end and one for the universal assembler.

Pascal, and C compilers have been produced, with many more languages in the works. Back ends exist for PDP-11, VAX, 8086, 68000 and others. The software currently runs on PDP-11 and VAX; the 68000 is on the way. It looks like Tanenbaum is working on the Model T of compilers.

[The speaker has submitted a paper for the conference proceedings...Ed.]

## UNIX Directions

Thursday, July 14, 3:30→5:15

Chair: Michael Tilson

Reporter: *Andrew Tannenbaum*

## UNIX Style, or *cat -v* Considered Harmful

*Rob Pike*  
Bell Laboratories, 2C-521  
Murray Hill, NJ  
{dequcb}vax!research!rob

It seems that UNIX has become the victim of cancerous growth at the hands of such organizations as UCB. 4.2BSD is an order of magnitude larger than Version 5, but, Pike claims, not ten times better.

The talk reviews reasons for UNIX's popularity and shows, using UCB *cat* as a primary example, how UNIX has grown fat. *cat* isn't for printing files with line numbers, it isn't for compressing multiple blank lines, it's not for looking at non-printing ASCII characters, it's for concatenating files.

We are reminded that *ls* isn't the place for code to break a single column into multiple ones, and that *netnews* shouldn't have its own *more* processing or joke encryption code.

Rob carried the standard well for the "spirit of UNIX," and you can look forward to a deeper look at the philosophy of UNIX in his forthcoming book.

;login:

## Everything You Wanted to Know about System V, and Then Some

*Jim Balter*  
Interactive Systems Corporation  
1212 Seventh Street  
Santa Monica, CA 90401  
(213) 450-8363  
decvax!yale!imaljim

Jim discussed many of the new features of System V. The good old manual has had its sections juggled and the administrative goodies have been put in a separate volume. Many sections have been renumbered, for reasons that aren't clear.

System V is faster than System III, due to 1Kbyte block filesystems (which can run side by side with old 512 byte block filesystems under the same kernel) hashed inodes, and other kernel speedups.

Inter-process communication and locking seem to be partly present but some of the code is just no-ops.

There are some kernel changes; for example, *uname()* returns a larger data structure, which may clobber random data in your programs unless you recompile.

System V still non-supports a wide variety of non-DEC peripherals.

## UNIX System V and 4.1C BSD

*John Chambers*  
Office of Academic Computing  
449 Admin. Bldg. and Biostatistics  
University of Texas Medical Branch  
Galveston, TX 77550  
(409)761-1813  
decvax!eagle!ut-ngp!jbc

*John Quarterman*  
Computation Center  
University of Texas at Austin  
Austin, TX 78712  
(512) 471-3241, ext. 252  
decvax!eagle!ut-ngp!jsq

Comparisons were done running on a VAX 780 and a 750. 4.1c seems to be 25% faster than System V at typical disk file I/O, though Berkeley has been forecasting factor of 5 or so speedups for heavily disk I/O bound tasks in 4.2. Benchmarks showed that 4.1c was generally a bit faster.

Bringing up 4.1c seems easier because of the autoconfiguration code. System V requires hand setting of system parameters. Berkeley has paging, more languages, sockets, network, IPC, different terminal support, more device drivers for non-DEC equipment. Berkeley has RCS, *rogue 5.2*, *netnews*, *sendmail*, *csh*, and *cat -v*.

BTL has KMC drivers with Virtual Protocol Machine software, nice support for system error logging, *nroff* mm macros, *SCCS*, and new "common object file format."

System V has copious documentation, well, relatively copious. It has a manual of system error messages (!). Seems both systems are growing, which might be good, depending on your viewpoint.

There is a memo that goes into these questions in greater detail.

[The speaker has submitted a paper for the conference proceedings...Ed.]



;login:

## **Berkeley UNIX after 4.2BSD: Where Is It Going and Why Do We Want It To Get There?**

*Michael O'Dell*  
Department of Computer Science  
Lawrence Berkeley Laboratory  
University of California  
Berkeley, CA 94720  
(415) 486-5583

The released form of 4.2BSD will not have remote filesystems, some of the intended IPC, and some other features which were described in the original 4.2 preliminary documents. The 4.2BSD release documents were mailed right before Toronto USENIX, and will require signatures on new documents. More work for more lawyers.

Mike discussed the turnover of Berkeley staff, and what will happen for 4.3BSD. 4.3BSD will be a cleaned up version of 4.2 with an emphasis on packaging, compatibility, and simplification. I think Mike's been talking to Rob Pike.

Note: BRL is working on a package to emulate System V under 4.2BSD. Bless their hearts.

### **Networking**

Friday, July 15, 8:45—10:00

Chair: Michael O'Dell

Reporter: *Bill Tuthill*  
2660 Marine Way  
Mountain View, CA 94043  
ucbvax!imagen!tut

### **Local Network with Virtual Ports**

*Gary Gafke, Eric Bergan*  
Johns Hopkins University  
Applied Physics Laboratory  
Johns Hopkins Road  
Laurel, MD 20707  
301-792-7800  
brl-bmd!aplvox!gary

The Johns Hopkins University Physics Laboratory has implemented a local area network using fiber optics and microprocessor-controlled network interface units (NIUs). The NIUs relieve the hosts of most of the overhead associated with network protocols, and also allow terminals to be connected directly to the network. The host computers on the network are: a VAX 780 running 4.1 BSD, a PDP 11/45 running PWB, and a 68000-based system also running UNIX. The network supports about 12 terminals and various peripheral devices, and has been in use since February of 1983.

The PDP and 68000 computers are connected to the network using the traditional "milking machine" approach, in which a separate physical port on the host is required for each user (or other host) conversing with it over the network. However, since tty ports are always at a premium, a different approach was taken on the VAX: a single port is used to support many users. This was done by developing a system of multiplexed virtual ports, which map into a single physical port on the VAX.

;login:

Multiplexed virtual ports require the following additions to the kernel: a virtual device driver (much like the DH/DM driver for dialup ports), a virtual tty driver (with slightly modified *ioctl* routines), and an input finite state automata with message processor. These changes add 15K bytes to the UNIX kernel, but since the virtual tty driver is much like the new tty driver, this figure could be reduced by consolidation.

Very little was said about fiber optic network technology, although this seems interesting as well.

[The speaker has submitted a paper for the conference proceedings...Ed.]

## A UNIX-based Network-using Operating System

*A. Wambecq*  
Bell Telephone Manufacturing Company  
Antwerp, Belgium  
(323) 237-1717

NETIX is an Ethernet-based UNIX network designed and implemented at Bell Telephone Manufacturing Company in Antwerp, Belgium. It requires a number of host computers, disks, terminals or workstations, and network interface units (NIUs). The NIUs, which contain an unspecified processor, and seem to be supplied by Ungermann-Bass, are connected to the Ethernet cable, and are responsible for handling communication protocols. There is no need for FTP, IP, or TCP.

The central idea behind NETIX is that there is one global machine and one global filesystem. Conceptually there is a directory named `/.`  above root on all the constituent machines. You could refer to a file on the "B" machine as `"/./B/usr/guest/file"`, or you could run a program residing only on the "C" machine by invoking `"/./C/usr/bin/prog"`. The protection scheme is completely distributed, and root login across hosts is not allowed. This is all accomplished by means of interkernel communication (IKC).

NETIX was designed to make the attached processors (the NIUs) and the local area network (LAN) work together so that the user of the system is aware only of a single, networked machine, which unifies the files of all the processors, and makes programs run on whatever hardware is most appropriate. The activity is transparent both to the user and to user programs, resulting in uniform access to all resources (processing power, disk storage, printers) by devices located on the network. Nonetheless, each attached processor can work independently from the others, and each processor can grant access by other machines to specific local resources.

[The speaker has submitted a paper for the conference proceedings...Ed.]

## EtherTIP — A Virtual Terminal Interface to Ethernet

*Dick Foster*  
Department of Computing Science  
University of Alberta  
Edmonton, Alberta T6G 2H1 Canada  
403-432-5640  
alberta!dick

EtherTip, which provides a virtual port service, allows terminals to log into any host computer connected to an Ethernet. Although some development was done at the University of Alberta in Edmonton, they are also using UNET software from 3Com.

The EtherTip hardware includes a Sun workstation with no keyboard or screen, but with a 3Com Multibus Ethernet<sup>+</sup> interface, and eight configurable RS-232 ports. Originally, the network host computers were: a VAX 780 running 4.1 BSD, a PDP 11/45 running 2.8 BSD, and two Sun workstations. During the summer of 1983, three VAX 780s and 4 Sun workstations will be added.

;login:

The software is loaded from a host through the Ethernet. (The load program could, but does not, reside on a PROM in the EtherTip hardware.) A password-protected command mode can be used at any time to configure each of the RS-232 ports by setting baud rates, parity, and so forth. The network can also be tuned using the same mechanism by changing such things as high water marks in buffers, and retransmission timing characteristics.

EtherTip uses the TCP/IP communication protocols, and the Telnet protocol for terminal support. The software records certain performance measurements, such as the number of Ethernet collisions, the number of transport layer re-transmits, the average and maximum round trip times, etc. These statistics can be checked to monitor performance. There are minor security problems which were not discussed.

## Applications

Friday, July 15, 8:45→10:00

Chair: Barbara Arlow

Reporter: *F. Arlene Spurlock*

Technical Information Department  
Lawrence Berkeley Laboratory  
Berkeley, CA 94720

### **A Data Base Frontend, Driven By Tables Generated from a Data Dictionary**

*Edward Haenlin*

New York Blood Center  
310 East 67th Street  
New York, NY 10021  
(212) 570-3112  
decvax!harpo!floyd!cmc12!nybcg!haenlin

This talk described an approach used by the New York Blood Center to deal with multiple existing databases, many database programs, and an overabundance of application programming requests. When they started their project there was nothing commercially available.

Two programs now allow their users a degree of flexibility in accessing, editing, modifying, and storing data.

“dbsrq (database screen) — has virtual procedures; it is table driven; detects corruption of a table; and allows its users to customize their requests without programming.

“pgen” (procedure generator) — is directed by the procedure database,; it generates all the tables for “dbs”, it performs consistency checks; and reads the data dictionary which describes every database, itself, and all the known links.

The advantages they feel they gained were a uniform model that is non procedural in nature and the biggest plus, a system which encourages data revision and update.

[The speaker has submitted a paper for the conference proceedings...Ed.]

;login:

## **A Powerful Accounting Package for UNIX-Based Systems**

*Peter Wolfe, Allen Hustler*  
Human Computing Resources Corporation  
10 St. Mary Street  
Toronto, Ontario M4Y 1P9 Canada  
(416) 922-1937

This was a commercial presentation of a UNIX-based accounting package for business applications that use existing UNIX tools such as *C*, *make*, *lint*, *awk*, *nroff*, and the ported database, MISTRESS.

The business applications being addressed were accounting functions: accounts payable and receivables, and general ledger. Others mentioned: inventory, invoicing, sales analysis, order processing, and purchase order procedures.

The projected availability for these applications programs was September 1983, with cost being dependent upon the users hardware.

## **UNIX Writer's Workbench**

*Charles R. Smith*  
Colorado State University  
Department of English  
Fort Collins, CO 80523  
(303) 491-5310

Colorado State University undertook testing of the Writer's Workbench as a research project with Bell Labs.

They tailored the workbench structure specifically for student uses and applications producing a computer-aided instruction composition program which has received positive, enthusiastic response from student and teacher alike.

The final line: Writer's Workbench both improved how the subject is taught and significantly improved their student's writing skills.

## **UNIX Mail**

Friday, July 15, 10:30→11:45

Chair: Michael O'Dell

Reporter: *Tom Strong*

## **Where is Europe?**

*Jim McKie*  
Mathematisch Centrum  
Kruislaan 413  
1098 SJ Amsterdam  
The Netherlands  
+31 20 5924147  
decvax!mcvax!jim  
philabs!mcvax!jim

UNIX users in Europe began to meeting informally in 1977 as a SIG of DECUS UK. In April, 1981 the first meeting of the European UNIX Users Group (EUUG) was held in Amsterdam. At that

;login:

time the group was still composed mainly of members from the United Kingdom. At the Paris EEUG meeting in the Spring of 1982 the Mathematisch Centrum in Amsterdam announced that they had connected to *usenet* in the USA and were willing to call or be called by any EUUG site in Europe. This stimulated a rapid rise in the number of sites in Europe on the network, and a change in the composition of the EUUG. When a new country joins the net, the first site is nominated as a "backbone" site for the country and encouraged to set up a national UNIX group. Representatives of the national groups get to sit in the EUUG committee. Thus the EUUG has become oriented to the network users.

The countries currently involved in the network and their backbone sites are:

Country	Backbone Site	
Belgium	vub	Vrije Universiteit, Brussel
Denmark	diku	University of Copenhagen
England	ukc	University of Kent
France	vmuncnam	CANM, Paris
Germany	ztil	Siemens, Munich
Netherlands	mcvox	Mathematisch Centrum, Amsterdam
Norway	kvport	Kongsberg, Kongsberg
Scotland	edcaad	EdCAAD Studies, Edinburgh
Sweden	enea	ENEA Data Svenska, Taebly
Switzerland	cern45	CERN, Geneva

News and mail from North America comes only from three sources:

decvax → mcvox (mail only)  
philabs → mcvox  
vax135 → edcaad (mail only), ukc

The EUUG and the network are conscious of the need to remain cost effective as the volume on the network increases. This may mean using another network such as X.25.

There does not seem to be much support for a separate user group for commercial interests. Only about 30% of the EUUG members are commercial but the major manufacturers have joined EUUG and, in fact, Philips contributes to the cost of the news link between philabs and mcvox. DEC Europe has provided a VAX 11/750 for network news and mail.

[The speaker has submitted a paper for the conference proceedings...Ed.]

## UNIX and Electronic Mail: Trials, Tribulations, and Proposals

*Michael D. O'Dell*  
CSAM 50B/3238  
Lawrence Berkeley Laboratory  
University of California  
Berkeley, CA 94720  
ucbvax!lbl-csam!mo

There are several problems with electronic mail on UNIX. The biggest is that mail may or may not get there, and the sender may or may not know it. The different address protocols for different networks (e.g., *uucp*, Arpa) also causes problems.

Electronic mail needs to consider recipients as agents, not just logins, so such things as remote execution of a command will be possible.

There needs to be some notion of who is responsible for a message at any particular time. This implies some reliable way of building return addresses.

;login:

*mh* was recommended as the most UNIX-like mailer. It has folders and cross-filing, among other features. *mh* will be part of the 4.2BSD distribution as contributed software from Rand Corporation.

Mike Lesk, the original author of *uucp*, commented that *uucp* was written as a temporary kludge 7 years ago until some better network came along. The replacement for *uucp* is now long overdue.

## Standards, Validation, and Portability

Friday, July 15, 10:30→11:45

Chair: Michael O'Dell

Reporter: *Bill Tuthill*

### Developing a UNIX Validation Suite

*Gary Fostel, Alison Naylor*  
North Carolina State University  
P.O. Box 5972  
Raleigh, NC 27650  
decvax!duke!ncnc!ncsu!fostel

A group at North Carolina State University was involved in porting UNIX to a Data General machine. Their porting effort raised the question: once you have UNIX running on a new architecture, how can you test it to make sure it works correctly? And, with so many types of UNIX, what is correct? There has been no UNIX standard since Version 7. Either 4.1 BSD or System 3 might be taken in its place, or in the future, either 4.2 BSD or System 5. The group at NCSU chose to compare their effort with 4.1 BSD, with occasional reference to the /usr/group Draft Standard.

Validations suites have been done before. There have long been complete test sets for Fortran; the Ada Validation Test Suite developed by Goodenough et. al. at Softech is a good example of recent work in this area. Bell Labs has a UNIX test set, but it has been censored outside the Labs. The Sol and Tunis operating systems, written in Pascal and Concurrent Euclid, were extensively tested with their own validation suites. But for the commercial UNIX world, there are no established guidelines.

The /usr/group Draft Standard is a small subset of most real-world UNIX systems. Each different UNIX system intersects only in part with the set described by any other UNIX system. Even without these problems, a complete validation suite might test only part of the kernel, and an even smaller part of library routines and system utilities.

As an example of the vagaries encountered by the UNIX implementor, Mr. Fostel mentioned the following undocumented properties of his version of UNIX: *unlink(2)* changes the mod time after a link; *open* is allowed twice by two different processes; *errno* is not tied to individual system calls; *lseek* doesn't return error past beginning of file; you can't ever write on a directory.

[The speaker has submitted a paper for the conference proceedings...Ed.]

;login:

## Status Report from the /usr/group Standards Committee

*Heinz Lycklama*  
Interactive Systems Corporation  
1212 Seventh Street  
Santa Monica, CA 90401  
(213) 450-8363

The Standards Committee was formed in 1981 by /usr/group's Board of Directors; they published a preliminary Draft Standard in late 1982. Their work has been made difficult by the proliferation of different UNIX versions: about one a year from AT&T, enhancements of varying qualities to these releases, and UNIX look-alike systems.

The standard they have devised is based only on sections 2 and 3 of the UNIX Programmer's Manual. What they have is for the most part a subset AT&T's System 3. The *mount(2)* and *unmount(2)* calls have been dropped, because they are of no use to application programs. The *dup(2)* call was added from Version 7, although it has been replaced by *fcntl(2)*. The terminal driver is a huge question mark; *ioctl(2)* is a null set. A file locking primitive, *lockf()*, not present in any of the major UNIX incarnations, have been added, because of its importance for database work.

## UNIX on the Gould SEL Concept Computers

*Deborah L. Franke, Thomas R. Truscott*  
Research Triangle Institute  
P.O. Box 12194  
Research Triangle Park, NC 27709  
(919) 541-6830(dlf), (919) 541-7005(trt)  
mcmc!rti!dlf, mncn!rti!trt

Research Triangle Institute was an early user of the Gould Concept 32/8750 computer, running UNIX 32V. This computer benchmarks at 3.7 MIP-Whetstone, which is a lot faster than the VAX 780. However, 4.1 BSD is not yet available for Gould computers. Their idea was to run CPU-intensive jobs on the Gould, while using the VAX for interactive tasks such as software development. For this task the Gould has been more than adequate.

RTI used UNET to drive an Ethernet, which connected the Gould computer to a VAX running 4.1 BSD. Software development of CAD application tools (for VLSI design and graphics) was done on the VAX, and finished programs were then ported to the Gould. For graphics, RTI used UNIX *plot(1)* filters, and ran a SigGraph Core Standard package implemented by Steve Rubin of Fairchild.

Here are some hardware differences between the VAX and the Gould Concept: the Gould doesn't do sign extension, has a left to right byte order (like most of the world), and reference to location zero is illegal. These issues have been discussed at previous conferences. Software differences between 4.1 BSD and Gould's 32V are as follows: flexnames are not available in 32V, the tty driver is less advanced and less complicated, and the order of argument evaluation can be different.

[The speaker has submitted a paper for the conference proceedings...Ed.]

;login:

## **UNIX Version 7 Compatibility Under System 3/5**

*Bob Scheulen*  
Microsoft Corporation  
10700 Northup Way  
Bellevue, WA 98004  
(206) 828-8080  
decvax!microsoft!bobs

Whatever their technical merits or lack thereof, System 3 and 5 are replacing Version 7 as the UNIX standard. Unfortunately, they are not upwardly compatible from V7. Most new UNIX users have neither the source code nor the expertise required to convert between V7 and S3/S5.

Therefore, Microsoft has undertaken a project to provide binary- and source-level compatibility between Version 7 and System 3 and 5. Their new release of Xenix will be based on System 3, with system calls and library routines included to provide backward compatibility with Version 7. The 'x.out' executable header will replace the 'a.out' format, and will run on either V7-based Xenix or S3-based Xenix. There will be compatibility libraries for V7-S3 interchange.

Here are some unresolved kernel differences, however: the null pathname resolves to '.' in S3 (? - as it does in V7); *mount*(8) used to be allowed but is reserved for the superuser in S3; *chown*(1) is no longer reserved for the superuser; and the superblock formats are different. In cases like these, Microsoft has gone with System 3.

[The speaker has submitted a paper for the conference proceedings...Ed.]

### **Open Session**

Friday, July 15, 1:30→3:00

Chair: Ian Darwin

The following talks were presented...Ed.

#### **The N.Y.U. Ada System**

*Michael N. Condict*  
New York University

#### **A Multiplexed Interactive System PWB/II**

*Tomihiko Kojima*  
Hitachi Ltd.

[The speaker has submitted a paper for the conference proceedings...Ed.]



;login:

### **A Virtual Machine - UNIX**

*Tomihiko Kojima*  
Hitachi Ltd.

[The speaker has submitted a paper for the conference proceedings...Ed.]

### **CSNET: A Computer Science Research Network**

*Laura Breeden*  
BBN

*Mike O'Brien*  
Rand

[The speaker has submitted a paper for the conference proceedings...Ed.]

### **Dynamic UNIX Re-Configuration**

*Joseph S. D. Yao*  
Hudson, Inc.

### **X25 and CCITT protocols for UNIX**

*Gerald Neufold*  
University of British Columbia

### **The Newcastle Connection, or UNIXes of the World Unite!**

*J. P. Black*  
University of Newcastle Upon Tyne

[The speaker has submitted a paper for the conference proceedings...Ed.]

### **MIRAGE - An Assembler Generator and Relocatable Linker**

*Daniel Klein*  
Mellon Institute, PA

## **Information on Vendor Exhibits at the Toronto Conference**

Each of the 45 exhibitors at the Toronto Conference was personally invited by the editor to submit 100 words describing their products or services for inclusion in this issue of *;login:*. All material presented to the editor has been reprinted below with only minor editorial changes.

## **Callan Data Systems**

The product line consists of the unistar 100 and 200 systems which are 68000/UNIX-based workstations. The unistar 100 is a single user workstation with 256Kb ram, 10Mb Winchester (5-1/4") 1Mb floppy and 8-slot multibus. The unistar 200 is a multiuser (4 users) workstation with 512Kb RAM, 21Mb Winchester (5-1/4"), 1Mb floppy and 8 slot multibus. Both systems come with UNIX C and assembler and can expand memory to 2Mb. Available options include graphics, Ada, FORTRAN, Pascal, BASIC, COBOL, word processing, spreadsheet and microingres.

For further information contact:

Kenn G. Morris  
Director of Marketing Communications  
Callan Data Systems  
2645 Townsgate Road  
Westlake Village, CA 91361  
805-497-6837

Lauren Schiro  
Manager, Marketing Programs  
Callan Data Systems  
2645 Townsgate Road  
Westlake Village, CA 91361  
805-497-6837

## **Cambridge Digital Systems**

Cambridge Digital has long been considered the system integrator's system integrator when it comes to DEC and DEC-compatible systems.

Cambridge Digital is expanding its product line by offering more advanced technology products for DEC based systems including a new 68000-based processor board that can give a Qbus system the performance of a VAX.

Cambridge Digital is also expanding into the fast growing Multibus market by offering the well known NCR Tower system.

These expansions are complimented by Cambridge Digital's distribution and support of UNIX and UNIX-based software for DEC, NCR, IBM-PC and the Qbus 68000.

For further information contact:

Dick Churchill  
Cambridge Digital Systems  
Division of Compumart  
65 Bent Street  
Cambridge, MA 02139  
617-491-2700

## **Computer Technology Group**

The Computer Technology Group (CTG) demonstrated its video-based training courses on UNIX and C programming. The courses presently available on video are:

UNIX overview (6 tapes)  
UNIX fundamentals (15 tapes)  
C programming (16 tapes)

The courses also include extensive textual material with mastery tests and hands-on exercises.

CTG also displayed material from its full curriculum of live courses which are taught in public seminars or on-site.

For further information contact:

John De Ano  
Computer Technology Group  
310 South Michigan

;login:

Chicago, IL 60604  
312-987-4084

### **Computerized Office Services, Inc.**

COSI, a sizable UNIX house (22 people), has just completed porting UNIX (SERIX) onto the IBM Series/1 computer. To bring UNIX to the commercial environment COSI provides custom software solutions.

SERIX is AT&T-licensed UNIX System III. The included C compiler takes full advantage of the Series/1 instruction set. By putting UNIX on the Series/1 COSI has forged a path for UNIX OEMs to sell into all-IBM DP shops. The estimated Series/1 installed base is over 30,000 CPU's.

Also introduced by COSI was a visual shell, or menu system, which permits: menu creation, data entry, and file maintenance.

For further information contact:

Jeff Spencer  
Computerized Office Services, Inc.  
313 North First Street  
Ann Arbor, MI 48103  
313-665-8778

### **Data General**

Data General introduced SPHINX, and UNIX System III implementation, on their proprietary MV/family hardware, at the USENIX show. SPHINX is integrated with the native operating system, AOS/VS, and thus allows access to other DG software such as Xodiac (networking), CEO (comprehensive electronic office), Ada, etc. simultaneously with SPHINX. SPHINX is a comprehensive System III implementation—it includes all of PWB, most system calls, the shell, a C compiler, and most UNIX commands. SPHINX is available for all Data General MV processors—MV/4000, MV/6000, MV/8000 models 9300 and 9600, MV/8000 II and MV/10000—for 6, 16, 32, 64 and 64+ simultaneous SPHINX users.

For further information contact:

Andrea Curtis  
Data General  
4400 Computer Drive  
Westboro, MA 01581  
617-366-8911

### **Gould, Inc.**

Gould Inc., Computer Systems Division exhibited a variety of UNIX-based products at USENIX. The center piece was the Gould CONCEPT 32/6750 computer running under UTX Goulds' authorized implementation of the Bell Laboratories UNIX. Other UNIX-based products demonstrated included: Gould's new Ada R Learning Environment (ALE), an APL Compiler from Dyalog in the UK, and the MISTRESS relational data base from Rhodnius Ltd. of Canada.

The new Gould ALE Package provides Ada training and Ada program development capabilities under the UNIX-based UTX Operating System. It includes hardware, software, on-site training, installation and documentation.

For further information contact:

Bob Bergman  
Director UNIX Marketing/Planning  
Gould, Inc.

;login:

Computer Systems Division  
6901 West Sunrise Boulevard  
Fort Lauderdale, FL 33310  
305-587-2900

## **Handle Corp.**

Handle Corp. has just announced a new Handle Writer word processor feature. The Off Line Document Storage feature which enables a user to off-load and store selected documents while Handle Writer keeps a complete catalog of all off-line documents. The catalog includes a listing of who wrote the document, when and where the document was stored, how many times and who has accessed the document.

Handle Writer is part of the "Handle family of products" which include Handle Merge, Handle Calc, Handle Graphics, Handle Access, Handle Draw, and Handle Speller. Handle Writer is a 2nd generation word processor and includes full word processing features as well as sophisticated document management and security features. Handle products also make extensive use of a proprietary "Import/Export" feature which allows any Handle product to interact with data in existing system files.

For further information contact:

Karan Kauppila  
Handle Corp.  
140 Mackinaw Road  
P.O. Box 7018  
Tahoe City, CA 96730  
916-583-7283

## **Human Computing Resources Corp.**

### *Products*

1. Unity (UNIX) product line on the NS 16032, MC 68000
2. Unity (UNIX) system on the PDP-11 (11/23 up to 11/70)
3. Unity (UNIX) on the DEC VAX
  - a. standalone
  - b. under VMS (emulation)
4. Pascal, BASIC, Editor, menu shell, DBMS under UNIX

### *Services*

1. Porting services for hardware manufacturers of UNIX to their CPU's in standalone mode and as an emulation on the top of their existing operating system.
2. Training services - UNIX courses (hands on training)

For further information contact:

Sylvie Trepanier  
Human Computing Resources Corp.  
10 St. Mary Street  
Toronto, Ontario, Canada M4Y 1P9  
416-922-1937

;login:

## **Intel Corp.**

Intel Corp. is demonstrating two new UNIX products.

The Xenix 286 Operating System is the first port of the UNIX Operating System to the iAPX 286 Family. Xenix 286 takes full advantage of the following 286 features:

- Onchip memory protection
- 80287 Fast Floating point support
- 16 Mb addressing

The combination of the iAPX 286 family and the Xenix Operating System yields the fastest implementation of UNIX on a microprocessor. Xenix 286 will first be shipped to customers in September.

The iDIS System 86/735 is a Xenix 86 based product designed to provide a vehicle to extract data from a mainframe database to be used at a local workstation. The software contains a relational database, text processor, menu driver, and spreadsheet. These tools are integrated to provide a seamless software solution.

For further information contact:

Intel Corp.  
3065 Bowers Avenue  
Santa Clara, CA 95051  
408-987-8080

## **Interlan, Inc.**

Interlan, Inc. designs, manufactures, and sells Ethernet hardware, software and systems that provide data communications between systems with dissimilar system architectures. Specifically BSD 4.1 and 4.2 contain device drivers for Interlan Ethernet controllers. Interlan has also developed the Xerox WS Internet Transport Protocols to run under UNIX, VMS, and RSY.

For further information contact:

Jerry Wesel  
Interlan, Inc.  
3 Lyberty Way  
Westford, MA 01886  
617-692-3900 ext. 214

## **Logical Software, Inc.**

LOGIX is a relational database management system uniquely configured for the UNIX operating system. LOGIX runs at the shell level allowing users to take full advantage of UNIX facilities. LOGIX provides a powerful interactive command language, a query language/report writer, a full-screen relation editor and a C-interface.

SoftShell is a full-screen user interface to UNIX with command templates for help with command options and help screens for finding the commands a user wants. In addition, SoftShell amplifies the power of the usual Bourne Shell with scrolling, editing and re-execution of commands and offers a convenient and powerful forms interface for applications programs.

For further information contact:

Kent M. Brown  
Logical Software, Inc.  
55 Wheeler Street  
Cambridge, MA 02138

;login:

617-864-0137

## **Masscomp**

The MC-500 system is a superset of UNIX System III coupled with Berkeley virtual memory enhancements plus Masscomp Real-Time enhancements. It has a 32-bit VLSI CPU that has 16 megabytes virtual address space and a Multibus system peripheral bus. The Data Acquisition front end incorporates bit slice technology for 2 megabyte analog acquisition throughout.

An easy-to-use menu driver interface provides high level capabilities for data acquisition, analogs, and presentation.

The independent graphics processor off-loads the main CPU for graphics-intensive tasks.

For further information contact:

Phil Despo  
Marketing Communications Manager  
Masscomp  
543 Great Road  
Littleton, MA 01460  
617-486-9425

## **National Semiconductor Corp.**

National Semiconductor demonstrated GENIX, their port of the Berkeley 4.1 BSD UNIX for National's NS16032 microprocessor family. The demonstration, including C and Pascal compilers, was done on SYS16, an NS16032-based development system. The system, called SYS16, is slated for formal introduction and production the end of this year.

The NS16032 is an internal 32-bit external 16-bit microprocessor which is available now. The 32032, available later this year, brings out the full 32-bit bus. Slave components include a memory management unit (MMU) for support of demand paged virtual memory and floating point unit (FPU) slave processor. The architecture is optimized for high level language efficiency.

For further information contact:

Pam Flournoy  
Communications Manager  
National Semiconductor Corp.  
Microcomputer Systems Division  
2900 Semiconductor Drive  
Santa Clara, CA 95051  
408-721-5000

## **NCR Canada, Ltd.**

NCR demonstrated its Tower 1632, a high performance super micro. The Tower utilizes a 16-bit Motorola 68000 processor; an operating system derived from UNIX Version 7 for maximum portability and compatibility; a Multibus input/output system; an Ethernet-based local area network; and IBM-compatible high order communications. All peripheral interfaces are industry standard.

For further information contact:

Virve Tremblay  
NCR Canada, Ltd.  
6865 Century Avenue  
Mississauga, Ontario, Canada L5N 2E2  
416-826-9000

;login:

## **New Age Software, Ltd.**

New Age Software is offering a version of BASIC that runs on UNIX. It is functionally compatible with Oasis BASIC. A NABASIC program can temporarily escape to the shell and execute a UNIX command through a *cs*h command that is equivalent to CSI under Oasis. Development of NABASIC programs is facilitated by the fact that all terminal input and output can be redirected through UNIX standard I/O.

For further information contact:

Joseph Novak  
New Age Software, Ltd.  
122 St. Patrick Street  
Toronto, Ontario, Canada M5T 2X8  
416-977-5585, Telex 06-986766

## **Officesmiths, Inc.**

The Officesmith is a UNIX-based office application development system. It has a powerful multi-window operating environment and document-based management facilities. The Officesmith is licensed directly and through micro hardware vendors and system integrators. It has been designed for office system developers who build departmental administrative support systems.

OfficePolicy is the first in a series of application products. It provides administrative offices with a management guide, training courses and software for policy and procedure systems.

For further information contact:

Glenn McInnes  
President  
Officesmiths, Inc.  
331 Cooper Street  
Ottawa, Ontario, Canada K2P 0G5  
613-235-6749

## **Oregon Software**

Oregon Software offers a high-performance Pascal compiler, Pascal-2 and source-level debugger for use on UNIX computers. Pascal-2 is provided for software developers whose applications require the fastest, most compact code and which offer the greatest possible portability.

The Oregon Software Pascal-2 compiler for UNIX MC68000 systems including Masscomp MC500 and UniSoft Uniplus was introduced during the USENIX conference. Benchmarks contrasting performance of code generated by Pascal-2, Silicon Valley Pascal, Motorola Pascal and UniSoft C were provided to attendees.

Pricing for Pascal-2 on MC68000/UNIX Systems begins at \$600.00 (U.S.) list price. OEM discounts are available.

For further information contact:

David Cloutier  
VP Marketing and Sales  
Oregon Software  
2340 SW Canyon Road  
Portland, OR 97201  
503-226-7760

;login:

## **Quadratron Systems, Inc.**

Quadratron Systems exhibited its complete office automation series, featuring Q-one, the industry standard in word processing. Also available are electronic calendar, notepad mail, calculator, phone directory, and forms generator. All are integrated through Q-menu, the menu manager which can process, create, and execute any menu, program, process, or command. All programs operate on all UNIX and compatible operating systems, and are processor and terminal independent.

For further information contact:

Karl K. Klessig  
Quadratron Systems, Inc.  
15760 Ventura Boulevard  
Encino, CA 91436  
213-789-8588

## **Quality Micro Systems**

*Complete turnkey system:* from camera capture of images to versatile screen handling of both graphics and text to reproduction by a high resolution laser printer.

Package includes the AUDRE minicomputer which is designed to speed and simplify computer input of pictorial information, such as maps, engineering drawings, text documents, typesetting, photographs, and charts.

AUDRE screen displays are easily transmitted to the Lasergrafix 1200. This laser printer contains a sophisticated intelligent controller which allows the merging of text and graphics on a single page at the resolution of 300 dots per inch. In addition, typesetting formatting software is available to support *troff* and TEX.

For further information contact:

James Carter  
Quality Micro Systems  
P.O. Box 81250  
Mobile, AL 36689  
205-633-4300

Lee Phillips  
Leerick Limited  
P.O. Box 1599  
Warren, MI 48090  
313-268-4800

## **Ryan-McFarland Corp.**

RM/COBOL is a reliable, industry standard COBOL compiler that creates portable application software.

In use since 1976, now in more than 150,000 installations, RM/COBOL is available on more than a hundred specific computer systems, from 8- and 16-bit micros, to minicomputers, to IBM mainframes, supported by more than 35 operating systems. An RM/COBOL-based application can be executed on any of these systems, regardless of where it was developed.

Designed for serious business computing, RM/COBOL provides the features needed for the efficient development and execution of COBOL applications. An expanding resource of hundreds of business applications is immediately available to all systems supported by the compiler.

RM/COBOL is a GSA certified (exception free) implementation of the ANSI 74 COBOL standard.

For further information contact:

Ryan-McFarland Corp.  
Corporate Sale Headquarters  
609 Deep Valley Drive  
Rolling Hills Estates, CA 90274



;login:

213-541-4828, TWX 910-344-6353 RMC ROHE

## **The Santa Cruz Operation**

The Santa Cruz Operation offers quality products designed for the UNIX environment. We currently feature: Xenix ports, Micro Focus COBOL Products, Informix Relational Database Systems, Uniplex Word Processor, Microsoft Multiplan for UNIX, a family of high quality cross assembler tools, and the UNIX system tutorials. Our products are available for most of the popular UNIX-based systems.

For further information contact:

Kam Bargert  
The Santa Cruz Operation  
500 Chestnut Street  
Santa Cruz, CA 95060  
408-425-7222

## **SofTest, Inc.**

SofTest, Inc. exhibited three products at USENIX. Lex—a word processor—is the most mature, multifunctional word processor available for the UNIX market. SofGram—a communications package—lets you create and send messages via the Telex, Twx, and DDD networks. The MENU SYSTEM—a user interface package—lets you build menus using English as a front-end to the shell and any application under UNIX.

SofTest is also available for in-house consulting projects.

For further information contact:

Michael Heffler  
SofTest, Inc.  
555 Goffle Road  
Ridgewood, NJ 07450  
201-447-3901

## **UniSoft Corp.**

The highlight of the UniSoft display at USENIX was the demonstration of the UNIX System V operating system running on Motorola MC68000-based microcomputers. This new version of UniPlus was demonstrated on machines from Callan, Codata, and Cosmos. The System V UniPlus operating system was available on both SUN and 68451 memory management units.

UniSoft Systems, located in Berkeley, distributes a variety of languages and applications to run under UniPlus. These include FORTRAN 77, Pascal, BASIC FOUR, BASIC-PLUS, COBOL, ADM68, ViewComp, UltraCalc, Ada, and B-NET Berkeley networking software.

For further information contact:

Charles C. Valente  
UniSoft Corp.  
2405 Fourth Street  
Berkeley, CA 94710  
415-644-1230

;login:

## Cross Compiler List

The following is a list of cross compilers that appeared on *usenet*. It is reproduced courtesy of the person who put it together, Tim Curry. Updates, corrections, and additions should be sent to him at `decvax!ucf-cs!tim` on *uucp* or `tim.ucf-cs@rand-relay` on Arpanet.

---

- (1) The Source Machine, Operating System and Configuration  
Any UNIX machine, or other machine capable of running the Portable C Compiler.
  - (2) The Destination Machine, Operating System and Configuration  
Any 8086 or 8088 system with an assembler accepting Intel mnemonics. There is no run-time support supplied with the compiler.
  - (3) Availability/Terms  
Send me a photocopy of your UNIX source license (signature page is fine), a blank tape, and a self-addressed, stamped return mailer. I can also supply it via Arpanet.
  - (4) Who to contact about it  
Jim Rees  
Computer Science Dept. FR-35  
University of Washington  
Seattle, WA 98115  
(206) 545-0912
  - (5) Comments / Caveats  
I originally got this compiler from Chris Terman at MIT, who was using it with his own (non-Intel) assembler. I fixed it to produce Intel compatible assembly code, and to use the segment registers. It uses the Intel "long" model, so is not restricted to 64K. The port is far from complete, and I won't be doing any more work on it, but it will compile simple programs. It won't compile itself.
  - (6) Information Supplied By:  
`decvax!microsof!uw-beaver!jim`  
Jim Rees <jim at uw-beaver>
- 

- (1) The Source Machine, Operating System and Configuration  
The distribution of DECUS C (Public-domain C system for DEC operating systems such as RSTS/E, RSX-11M, RT-11, and VMS) contains several cross assemblers for machines such as the Z80 and 8085 (I think). Source code is in C, it should be possible to get it running on any system that supports C.
  - (3) Availability/Terms  
It is distributed in source code and is in the public domain.
  - (4) Who to contact about it  
DECUS C is distributed by the Digital Equipment Users Society as DECUS 11-SP-18. Contact them for more details.
  - (5) Comments / Caveats  
There is no support available whatsoever. (I don't think I've even compiled it for about three years.)
  - (6) Information Supplied By:  
Martin Minow  
`decvax!minow`
-

;login:

- (1) The Source Machine, Operating System and Configuration  
The source machine can be VAX/VMS, RSX-11, (and possibly in the near future) PDP-11 UNIX.
  - (2) The Destination Machine, Operating System and Configuration  
The destination machines can be a 68K, or a PDP-11. Much of their work is oriented to doing small stand alone systems, (e.g., DEC's Falcon board, single board 68K's).
  - (3) Availability/Terms  
Contact Oregon Software, 2340 SW Canyon Road, Portland OR, 97201. (phone is 503-226-7760). Cost is just under \$4k, but there is a good educational discount if the system is used for instructional used.
  - (4) Who to contact about it  
(same as above)
  - (5) Comments / Caveats  
This product is well supported, and comes with a wide variety of tools and available options. Included with the system is a full symbolic debugger, a couple source formatting tools, and complete documentation. Available as options are a concurrent programming package for developing, debugging & testing concurrent systems, and a "Stand alone" package for developing systems on target machines without an operating system. Also available (pascal not required) is a source control system for DEC operating system that gives the same functionality as scs and make, with additional programs for downloading systems to target machines. The base price also includes a free year of technical-phone help and updates. Oregon Software will likely get into C in the future, but they don't have any products in that field yet. I would be willing to answer questions net people have about them until they get on the net (I work for them occasionally).
  - (6) Information Supplied By:  
decvax!harpo!utah-cs!jwp
- 

- (1) The Source Machine, Operating System and Configuration  
It should run on any UNIX system, but it now works on PDP-11, moving to VAX
- (2) The Destination Machine, Operating System and Configuration  
target machines now: PDP-11, VAX, 68000, 8086; shortly Z80, 8080 and others
- (3) Availability/Terms  
Availability expected earlier in the summer. University price probably about \$500 for sources. Commercial price negotiated on a case by case basis, depending what rights the company wants (single/multiple cpu, binary/source right to sublicense etc.)
- (4) Who to contact about it  
Contact is Andy Tanenbaum, at decvax!mcvax!vu44!ast (Amsterdam)
- (5) Comments / Caveats  
The Pascal front end is based on our PDP-11 UNIX Pascal compiler which is widely used in about 20 countries around the world. The object code quality is pretty good, although compiling on a badly overloaded PDP-11 is not real speedy. The thing is very flexible, i.e. adding a new target machine just requires writing two tables, one for the code generator, one for the universal assembler. The kit has many options, e.g. you can choose 16 or 32 bit integers for C on the 68000. It even comes with some documentation.  
[Editor's note:] Some references to look up: CACM, March 1978 p. 237, TOPLAS Jan. 1982 p. 221. A more recent CACM publication soon to appear.
- (6) Information Supplied By:  
Andy Tanenbaum

;login:

decvax!mcvax!vu44!ast

---

- (1) The Source Machine, Operating System and Configuration  
CROSS is a cross assembler that runs on the DECsystem-10/20.
  - (2) The Destination Machine, Operating System and Configuration  
Translating source code for 6502, 6800, 6805, 6809, 8080, 8085, MCS-48, z80, 1802, 3870/F8[future], and 8008 into binary files suitable for absolute load.
  - (4) Who to contact about it  
The documentation I have lists Ted Hess, Digital Equipment Corporation as the author and Joseph M. Newcomer, Carnegie-Mellon University as doing this documentation.
  - (5) Comments / Caveats  
It implements the features of the PDP-11 macro assembler for RSX-11D. It does not conform to input standards of other assemblers (eg, the manufacturers assembler), but instead uses the same input for all the microprocessors (ie, macros, labels, etc all look the same for the various target machines). I've used it quite a bit and found it fairly easy to use with only a few bugs (none that I couldn't get around - mostly having to do with phasing).  
At least for the 6502 (and others???) it produces either a binary file output or a ASCII encoded file of lines of Hex with checksums. My version of the documentation is dated Feb 7 1981. I believe [RUTGERS]<MICRO> has the documentation and executable code.
  - (6) Information Supplied By:  
Tyson@SRI-AI  
Mabry Tyson <Tyson@SRI-AI>
- 

- (1) The Source Machine, Operating System and Configuration  
To run on 11/70 and VAX. Based on Pcc.
  - (3) Availability/Terms  
Source available for any UNIX licensee. Free, of course, just send tape.
  - (4) Who to contact about it  
Gil Berglass at MITRE in Virginia - 703-827-6087, or Berglass@Mitre on Arpa.
  - (5) Comments / Caveats  
Creates analogues of cc, nm, ld, and adb. It works well, does not support floats. The Z8000 is similar to an 11, so they are nice micros for some applications. We are using the non-segmented version (only 64K), so I don't know if the C compiler is available for the Z8000 (16Mb address space).
  - (6) Information Supplied By:  
sun!megatest!fortune!dsd!atd!somewhere!nelson
- 

- (1) The Source Machine, Operating System and Configuration  
On our VAX 750 4.1BSD system we have the following:  
Packages: Unidot Cross Assemblers, Linker, and Librarian  
Runs On: VAX 4.1BSD, PDP-11 UNIX, ONYX UNIX
- (2) The Destination Machine, Operating System and Configuration  
8021, 8022, 8041, 8041A, 8048, 8051, 8080, 8085, 8086, Z80, Z8000
- (3) Availability/Terms  
Availability: Anyone can receive (commercial software)

;login:

Costs: Assemblers:

8080, 8085, Z80: \$600 each

8048, 8021, 8022, 8041, 8041A, 8058, 8051: \$800 each

8086: \$1000

Z8000: ???

Linker: \$600 (covers all of the processors)

Librarian: \$600 (covers all of the processors)

Note: ALL of the above costs must be doubled for VAX use.

(4) Who to contact about it

Santa Cruz Operation, (408) 425-7222

(6) Information Supplied By:

decvax!microsof!uw-beaver!teltonelira

---

(1) The Source Machine, Operating System and Configuration

Runs on: VAX 4.1BSD, PDP-11 UNIX

(2) The Destination Machine, Operating System and Configuration

Packages: 8086 C Compiler and Assembler

Supports: 8086, 8088

(3) Availability/Terms

Availability: Anyone can receive (commercial software)

Costs: 8086 C compiler: \$1750

8086 Assembler, linker: \$1750

(4) Who to contact about it

Darold Foster, Advanced Digital Products, (615) 383-7520

(5) Comments / Caveats

The C compiler is not very efficient, no register variables. We are using the assembler as an assembler and have had difficulties due to poor documentation and missing assembler features (such as no string constants). Note that the assembler is not very Intel compatible.

(6) Information Supplied By:

decvax!microsof!uw-beaver!teltonelira

---

(1) The Source Machine, Operating System and Configuration

Vandata Co. produces a Z80 cross compiler which presently runs on/under: VAX UNIX (4.1BSD), VAX VMS, PDP-11 UNIX, PDP-11 RSX and RSTS, Z8000 ZEUS (Zilog System 8000), Z8000 ONYX (UNIX), Z8000 ZENIX, Z8000 Plexus (UNIX), 68000 Unisoft port UNIX (Pixel), IBM-370 UNIX. And may soon be available to run under CP/M. Others by special arrangement.

(3) Availability/Terms

The compiler is good but not cheap. Presently prices are \$3200-\$3800 possibly less if you already have a Whitesmiths license. Distribution is made on magnetic media appropriate to the usual hardware configuration.

(4) Who to contact about it

Vandata

17544 Midvale Ave. N. Suite 107

Seattle, WA 98133

(206) 542-7611, (800) 426-5248

;login:

(5) Comments / Caveats

The compiler is a Whitesmiths Derivative. The full C language is supported, including bitfields and floating point. Machine level support library (source) is purchasable from Vandata, and the Whitesmiths portable C runtime environment for CP/M is available and supported. The usual: assembler, linker, loader, librarian, and rel-file utility are provided along with a number of useful tools including a multi-line regular expression based stream editor. A provision has been made for optional ROM/RAM environments, more compact calling conventions and 8-bit expression evaluation. If used, these options generate better code, but are not standard C. IEEE floating point may also be available soon.

[some comments on this package were submitted by another netter]

We use the Vandata C cross compiler, target is Z80. We are trying to bring up a BSO Z80 cross assembler, but are doing file transfers over our network to a CP/M running Microsoft asm/linker in the meantime. Vandata's Asharp/linker package isn't useful to us because of the limitations of the linker.

sun!megatest!ubvax!george

(6) Information Supplied By:

decvax!microsof!fluke!corey

---

(1) The Source Machine, Operating System and Configuration

VAX + 4.1BSD with about 2000 disk blocks (2MB)

(2) The Destination Machine, Operating System and Configuration

Product: cross-development system for NS16000 chip set

- cross-compiler for C
- cross-assembler for NS16000
- cross-loader
- cross-debugger (adb)
- data transfer programs
- stand-alone ROM monitor
- PROM-burner programs
- 'virtual terminal' connect program

You connect a NS16000 development system of your choice to the VAX via a terminal line. Then you can work with the development system from any terminal on the VAX; this allows convenient sharing of the system.

NS16000 development system (several sources possible, we use a hacked-up DB16000 on Multibus) with 16Kbyte EPROM, 8Kb RAM, one async line.

(3) Availability/Terms

We have used this package in-house for our V7 port; it will be available when marketing and documentation are ready late April / early May. Price yet to be announced.

(4) Who to contact about it

Tom Linden  
Translation Systems, Inc.  
530 Atlantic Ave.  
Boston, MA  
617-357-9433

(5) Comments / Caveats

The assembler uses a simplified notation (relative to that proposed by NS) that is similar to NS' notation but looks a lot more like the usual minimalist UNIX style.

The assembler supports ALL NS16000 CPU, MMU, FPU, and CSP (Custom Slave Processor) instructions.

;login:

At this time, floating point support in the C compiler is not completed. The C compiler is a modified Pcc.

You need an AT&T UNIX license to receive this product.

A subset of the BSD a.out format is used. In general, related BSD utilities are useful (i.e., nm, ar, size, etc.)

You customize the sources for the ROM monitor, primarily to accommodate your development system's async hardware. Note that you only need ONE async line on the development system; we run that line at 9600 baud.

TSI will also be selling source for its V7 port, later to be upgraded to System V with demand paging. You will need an AT&T UNIX source license to buy that package.

(6) Information Supplied By:

decvax!cca!t-krgr

Mike Krueger

Translation Systems, Inc.

---

(1) The Source Machine, Operating System and Configuration  
PDP-11, RT-11

(2) The Destination Machine, Operating System and Configuration  
Intel 8085, Intel 8048/49

(3) Availability/Terms  
Available to anyone interested.

(4) Who to contact about it  
Tom Miller  
North Carolina State University  
919-737-2336  
tkm.unc@udel-relay

(5) Comments / Caveats  
The assemblers were written mostly in FORTRAN under RT-11. I will probably convert the sources to C unless I can get assemblers from someone on the network.

(6) Information Supplied By:

duke!unc!tkm

Tom Miller

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

First Class Mail

