UNIX™ REVIEW

THE PUBLICATION FOR THE UNIX™ COMMUNITY

December 1984 $3.95

LOW COST PERSONAL UNIX

MODULAR.
INTEGRATED.
NOW.

Handle™ presents a modular software series for office automation.

Available now for several UNIX™ and XENIX™ based systems, the **Handle** family of office automation products may be purchased as individual modules, in combinations, or as a fully integrated system. The **Handle** product series is a powerful set of software tools designed for today's multi-user office environment. **Handle** integrated software can easily share data between modules in the series as well as with outside applications and databases. **Handle's** open architecture is built on a database foundation and is available to outside developers.

Handle Office Automation Software Series product modules include:

- **Handle Writer/Spell™.** Word processing with automatic spelling correction and verification.
- **Handle Calc™.** Virtual spreadsheet with up to 32,000 rows and columns.
- **Handle Graphics™.** Advanced business graphics.
- **Handle List™.** List processing, management, and forms.

AT&T 3B2/300 & 3B5

CONVERGENT TECHNOLOGIES MINIFRAME

DURANGO POPPY II

HANDLE TECHNOLOGIES, INC.

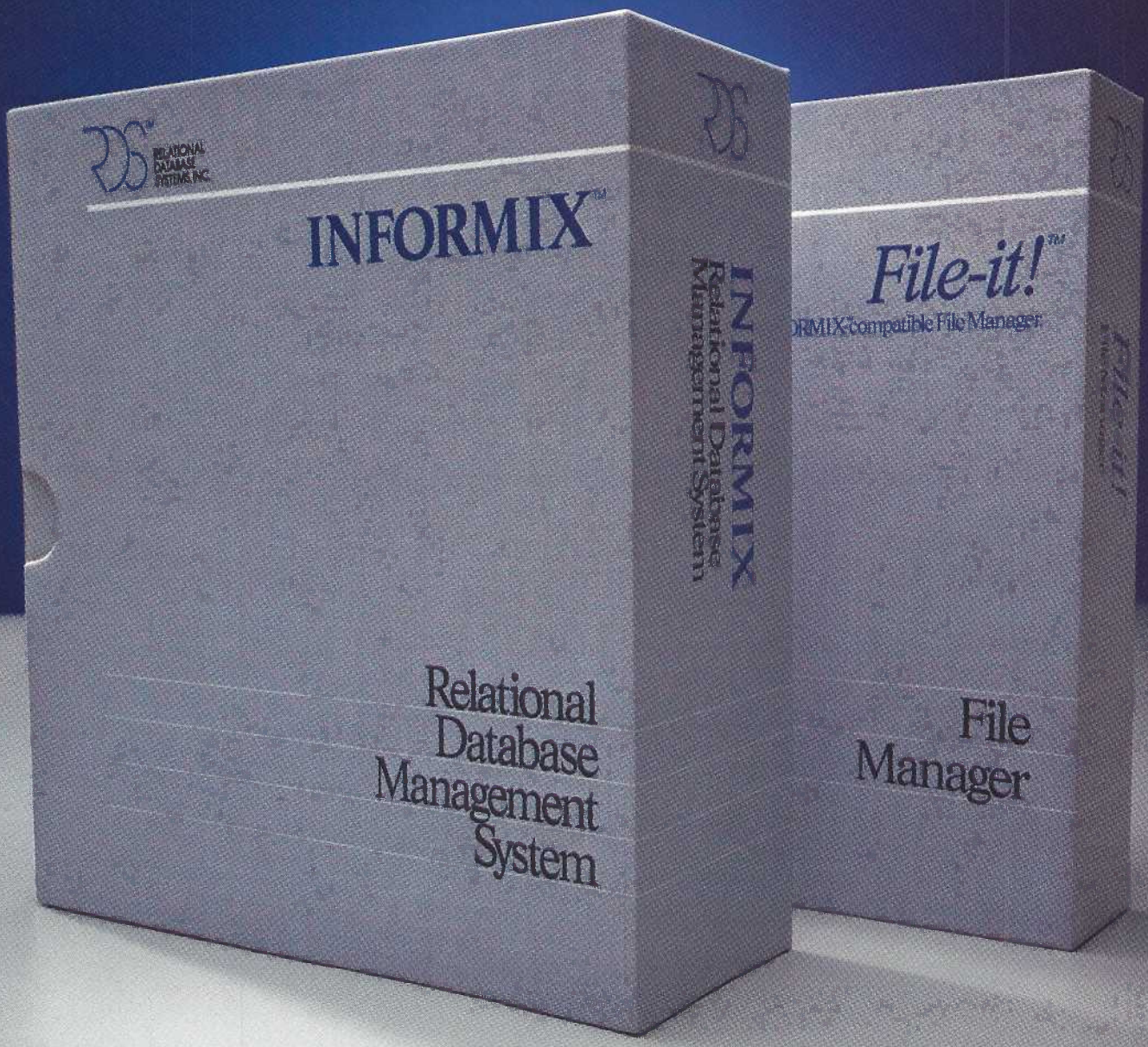
850 NORTH LAKE BOULEVARD / PO BOX 1913 / TAHOE CITY, CALIFORNIA 95730 / 916-581-5227 **Circle No. 300 on Inquiry Card**

YOU CAN'T PREDICT THE FUTURE.

BUT YOU CAN BE PREPARED FOR IT.

WITH INFORMIX AND File-it!

THE FIRST DATABASE SOFTWARE FAMILY FOR UNIX AND MS-DOS.



Now OEMs and systems integrators can sleep better at night. Because one company has taken the worry out of buying the right software.

RDS.

The company that produces a family of database software designed to take on the future.

Incompatibility is a thing of the past.

INFORMIX®and File-it!™ are compatible with UNIX,™ MS™-DOS, PC-DOS,™ and PC/IX systems (over 60 micros and minis* at last count).

INFORMIX is a true relational database system designed to take full advantage of the power of UNIX. It includes the most widely used report writer on the market.

Then there's File-it! The first easy-to-use UNIX file manager. Together, they have the flexibility to accommodate novices and experts alike.

INFORMIX and File-it! are fully integrated. Users can upgrade from File-it! to INFORMIX or access data from one program or the other without re-entering data, retraining employees or reprogramming.

Applications can also be moved from MS-DOS to UNIX and vice versa without having to rewrite the application.

Simplify program development.

RDS offers C-ISAM,™ the de facto standard ISAM for UNIX. It's a library of C subroutines with a B^+-Tree based access method that stores, retrieves and modifies data from indexed files. It's embedded in INFORMIX and File-it! Or is available as a standalone product.

Software good enough for AT&T.

AT&T, inventor of UNIX, has co-labeled INFORMIX, File-it! and C-ISAM to run on their full AT&T 3B Computer line (from micros to minis).

Hewlett-Packard, Altos, Zilog, Siemens, Cromemco, Perkin-Elmer, Sydis and General Automation have selected RDS as well.

In fact, INFORMIX has an installed base of over 6,000 copies. And RDS has sold over 35,000 licenses for all their products to date.

But before you make up your mind, check the facts one more time.

There's only one database software family that's UNIX-, PC-DOS-, MS-DOS- and PC/IX-based. It runs on more than 60 systems. And it's ideal for both novice and expert.

Now it doesn't matter where the future's headed. You're already there.

*RDS products are available for the following systems:

- Altos 586, 986, 8600, 68000
- Apollo DN300
- AT&T 3B2, 3B5, 3B20, AT&T Personal Computer
- BBN C machine (all models)
- Bunker Ramo Aladdin 20
- Charles River Data Systems Universe 68
- Convergent Technologies Miniframe and Megaframe
- Corvus Systems Uniplex
- Cromemco System 1
- DEC 11/23, 11/34, 11/44, 11/60, 11/70, VAX 11/730, 11/750, 11/780
- Dual Systems System 83
- Fortune 32:16
- Forward Technology 320
- General Automation Zebra (all models)
- Hewlett Packard 150, 9000 Series 200, 9000 Series 500
- IBM PC, PC-AT, PC-XT
- Intel System 86/380, 286/310
- Masscomp NC 500
- Momentum Hawk 32
- NCR Tower
- Onyx C8002, C8002A
- Pacific Micro Systems PM200
- Perkin-Elmer 32 Series, 7350
- Pixel 100/AP, 80 Supermicro
- Plexus P/25, P/35, P/40, P/60
- Pyramid Technology 90X
- Radio Shack Model 16
- SCI Systems IN/ix
- Silicon Graphics IRIS 1400
- Visual Technology 2000
- Wicat Systems
- Zilog System 8000 (all models)

Demos of INFORMIX and File-it! are available. Demonstration software and complete manuals included.

RDS™ RELATIONAL DATABASE SYSTEMS, INC.

2471 East Bayshore Road, Suite 600, Palo Alto, California 94303
(415) 424-1300 TELEX 467687

UNIX™ REVIEW

THE PUBLICATION FOR THE UNIX COMMUNITY

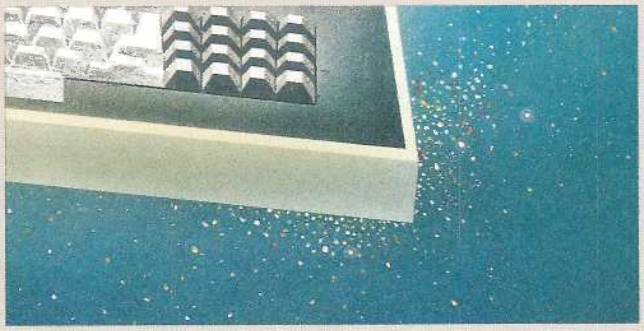
Volume 2,
Number 9
December 1984

DEPARTMENTS:

Cover art was computer generated on the Quantel Paint Box system at Artronic, NW, Seattle, WA.

FEATURES:

LOW COST PERSONAL UNIX



UNIX REVIEW (ISSN 0742:3136) is published monthly by REVIEW Publications Co. It is a publication dedicated exclusively to the needs of the UNIX community. Subscriptions are $28.00 per year (12 issues). Canada and Mexico add $7/year. Overseas add $20/year for surface mail. Address correspondence regarding editorial, press releases, product announcements to 520 Waller Street, San Francisco, CA 94117. Correspondence regarding subscriptions, change of address, USPS Form 2579, fulfillment and dealer sales should be addressed to 901 S. 3rd Street, Renton, WA 98055. Letters to UNIX REVIEW or its editors become the property of the magazine and are assumed intended for publication and may so be used. They should include the writer's full name, address, and home telephone. Letters may be edited for the purpose of clarity or space. Opinions expressed by the authors are not necessarily those of UNIX REVIEW.

Editorial telephone: 415/621-6415. Subscription/Dealer Sales telephone: 206/271-9605.

Application to mail at second class postage rate is pending at Renton, Washington 98055.

PUBLISHER:
Pamela J. McKee
EDITOR:
Mark Compton
ASSISTANT EDITOR:
David Chandler
ASSOCIATE EDITORS:
Ken Roberts, Scott Robin
PRODUCTION DIRECTOR:
Nancy Jorgensen
PRODUCTION STAFF:
James Allen, Dan Arthur, Tom Burrill, Cynthia Grant, Corey Nelson, Florence O'Brien, Denise Wertzler, S.T. Wilson
CIRCULATION/DEALER SALES:
Tracey McKee, Barbara Perry
BUSINESS MANAGER:
Ron King
ADVERTISING REPRESENTATIVES:
Jules E. Thompson, Inc.
1290 Howard Avenue Suite 303
Burlingame, CA 94010
Lucille Dennis 415/348-8222
303/595-9299 - Colorado
312/726-6047 - Illinois
617/720-1888 - Massachusetts
713/731-2605 - Texas
Jules E. Thompson, Inc.
2560 Via Tejon
Palos Verdes Estates, CA 90274
Ed Winchell 213/378-8361
212/772-0933 - New York
TELE-TYPESETTING:
Data & Staff Service Company
PRINTING:
Thomas Ogle

VIEWPOINT

UNIX gets personal

No consideration of "Low Cost Personal UNIX" would be complete without addressing at least two non-trivial questions: 1) what is low cost? and 2) what is UNIX? Aha, you say, that's obvious. But, sadly, it's not. The fact is opinions are divided on both questions.

A general consensus about the nature of UNIX did exist until AT&T's decision last year to unbundle System V.2. Since then, the issue has become muddled. Traditionalists argue that UNIX never was *merely* an operating system. It was also an environment stocked with tools and conveniences for the software developer.

This robust brand of UNIX can still be found. Call it System III, Version 7 or 4.2 BSD. As for System V.2 ("consider it standard") and Microsoft's Xenix for the IBM PC-AT, only core UNIX is offered in a base package. Tools for software development and document production are sold separately. Since this is the tack AT&T and IBM have chosen to take, you can look for others to follow suit.

This all casts fresh light over the question of "low cost." Under old assumptions, a low cost UNIX box still carried a price tag of $10,000 or more because of the minimum hardware demands posed by the system's size. Even a minimal system containing just 256K of RAM and a 10 MB hard disk was straining the definition of a "personal computer." With unbundled UNIX, though, even floppy-only systems offering just 128K of RAM are feasible as minimal systems — though they certainly wouldn't be recommended as candidates for kernel hacking.

Thus, the question of "low cost UNIX" leads inevitably to the gray juncture where software development and personal computing meet. I'm pleased to report that finer minds than my own serve in the pages that follow as guides through this twilight zone.

Taking first crack at the topic is Gig Graham, executive vice president of VenturCom. The argument he makes for general availability of UNIX on personal computers betrays Gig's solid grounding in economic thought.

Rob Warnock, the architect behind Fortune Systems' 32:16, follows with an article focusing on the feasibility of achieving full-bodied (i.e., un-unbundled) UNIX on personal computers. As you might guess, the piece is both forward looking and hardware oriented.

Cost-cutting approaches for people of initiative are proposed in a third article, co-authored by John Bass and Walter Zintz. John, who helped get both Onyx Systems and Fortune Systems going, has never made a secret of his enthusiasm for kits and home-brewed systems. Walter, meanwhile, is simply an advocate of UNIX in any form. He is also the founder and executive director of the Uni-Ops users' group.

UNIX REVIEW's focus on personal UNIX concludes with an interview featuring Bob Marsh, chairman of Plexus Computers. Bob made the notion of personal UNIX possible in 1980, when as president of Onyx Systems, he introduced UNIX to the world of micros. Dick Karpinski, manager of UNIX services at UC San Francisco, delivers the questions.

With all this talk about low cost UNIX, you may already be studying your checking balance. To think, here we are only a few scant days from Christmas. Timing is everything, isn't it? We hope you appreciate the thought.

Mark Compton

Available for IBM PC

What C did for Programming
Mark Williams has done for C Programming

The C Programming System from Mark Williams

MWC86 gets your C programs running faster and uses less memory space than any other compiler on the market. Then *csd*, Mark Williams' revolutionary C Source Debugger, helps you debug faster. That's The C Programming System from Mark Williams Company.

MWC86

MWC86 is the most highly optimized C compiler available anywhere for the DOS and 8086 environment. The benchmarks prove it! They show MWC86 is unmatched in speed and code density.

MWC86 supports large and small models of compilation, the 8087 math coprocessor and DOS 2.0 pathnames. The compiler features common code elimination, peephole optimization and register variables. It includes the most complete libraries. Unlike its competition, MWC86 supports the full C language including recent extensions such as the Berkeley structure rules, voids, enumerated data types, UNIX* I/O calls and structure assignments.

Quality is why Intel, DEC and Wang chose to distribute MWC86. These industry leaders looked and compared and found Mark Williams to be best.

User Friendly

MWC86 is the easiest to use of all compilers. One command runs all phases from pre-processor to assembler and linker. MWC86 eliminates the need to search for error messages in the back of a manual. All error messages appear on the screen in English.

A recent review of MWC86 in *PC World,* June, 1984, summed it up:

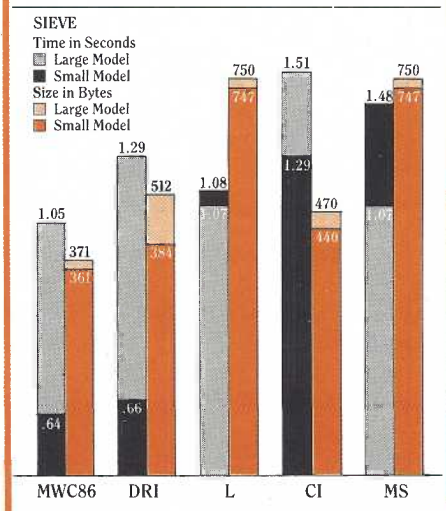
"Of all the compilers reviewed, MWC86 would be my first choice for product development. It compiles quickly, produces superior error messages, and generates quick, compact object code. The library is small and fast and closely follows the industry standard for C libraries."

csd C Source Debugger

Mark Williams was not content to write the best C compiler on the market. To advance the state of the art in software development, Mark Williams wrote *csd.*

csd C Source Debugger serves as a microscope on the program. Any C expression can be entered and evaluated. With *csd* a programmer can set tracepoints on variables and expressions with full history capability and can single step a program to find bugs. The debugger does not affect either code size or execution time. *csd* features online help instructions; the ability to walk through the stack; the debugging of graphics programs without disturbing the program under test; and evaluation, source, program and history windows.

csd eases the most difficult part of development — debugging. Because *csd* debugs in C, not assembler, a programmer no longer has to rely on old-fashioned assembler tools, but can work as if using a C interpreter — in real time.



The C Programming System from Mark Williams now supports the following libraries:

Library	Company
Windows for C	Creative Solutions
Halo	Media Cybernetics
PHACT	PHACT Associates
The Greenleaf Functions	Greenleaf Software
Btrieve	SoftCraft

The C Programming System from Mark Williams

The C Programming System from Mark Williams delivers not only the best C compiler for the 8086 but also the only C source level debugger. That's why it does for C programming what C did for programming. The Mark Williams C Programming System gives the programmer the MWC86 C compiler and the *csd* C Source Debugger for only $495. Order today by calling 1-800-MWC-1700. Major credit cards accepted.

Technical support for The Mark Williams C Programming System is provided free of charge by the team that developed it.

Mark Williams Company
1430 W. Wrightwood Ave.
Chicago, IL 60614

*Unix is a Trademark of Bell Laboratories.

Circle No. 298 on Inquiry Card

DEVIL'S ADVOCATE

<MRY XMS>; <HPY NU YR>.

by Stan Kelly-Bootle

I could not resist adding the Bellsoft (no relation to you-know-who's-Ma, as far as I know) Pop-Up software package to my IBM PC. The general idea of popping-up has always appealed to me. Indeed, before I gave up alcohol my *whole life* was a vague sequence of popping-up, propping-up and popping-off, and not always in that order, if memory serves. My program also reflected this bias. My psychiatrist, Siggy Kashflau, used to say, "*Schtan, you haff pusch-aversion und pop-fixation in expensively abnormal proportions*!" I should *pay* to hear this? I knew already that I had the untidiest *schtack in ze block*, my pull-offs and push-ons never in balance...until I discovered Pop-Up.

PC-DOS Version 2, as you know, offers all the inconveniences of UNIX and none of the advantages. As a friend of mine once put it, "Who wants to come home to a mess of hierarchical file directories — I get enough of *that* at work!"

The Bellsoft Pop-Up offers a low-cost migration from PC-DOS to some of the more useful features of UNIX — and beyond that to MacUNIX! Pop-Up is a whole kit of modular tools that you can invoke (that is to say, "cause to pop-up") with a single well-chosen keystroke while you are running WordStar, Lotus

1-2-3, or whatever. Your interrupted job is resumed when you pop-down.

The selected diversion unfolds in a *window*, that *sine qua non* of contemporary software. Actually, the use of the term "window" is confined to your lower class Cupertino Hacker; for the yuppie Mill Valley Software Engineering Consultant I have coined the phrase *conspicuous concurrency*, if only to escape the Window Tax promised by the forthcoming Reagan administration. (Remember, you read it first in UNIX REVIEW!)

A great deal has been written on screen metaphors and icons, and on how much can be safely revealed to the user without losing a friend. It has been pointed out that the icons currently in use portray a Victorian, anachronistic office full of strange devices called *filing cabinets* — objects completely unknown to recent school leavers to whom a file is simply a related set of records on a disk. So the first week of training is spent explaining how a filing cabinet used to work, and how to imagine disks cut up into rectangular sheets with little colored index tags clipped at the corners. All very confusing. At what point does a sacred icon become a profane rebus? The desk metaphor, too, is a mess. I call on both surviving Professors of Rhetoric to join me in condemning this bum trope. The screen *qua desk* is no metaphor, merely an empty metonymy, a simplistic synecdoche. See what happens when two solid semesters of "Speech, Figures Of" are dropped from the curriculum and replaced by "Intermediate Turtle Doodling." When the San Francisco Chronicle's Question Man asked 50 people on Market Street, "Would you rather have a zeugma or a syllepsis?" the most common reply was "I hate Gypsy Music, so I'll go for the disease."

The perfect analog of the President's clean, unblotted desk, of course, is the President's blank, unused CRT. For the overworked slobs down the line, however, an infinite regression of nested windows offering contradictory spread sheets is more appropriate. The real problem lurking behind

"Nothing works together," they cried out.

"It will now," said Alis.

With Alis™ everything works together. Text, spreadsheets, drawings, business graphics and database information work together in a single, always editable document.

Alis combines the advantages of integrated PC applications with the information-sharing benefits of communications-based OA systems. It offers the most advanced total office solution ever.

Alis makes it easy for people to work together. It provides integrated electronic mail, calendar and meeting scheduling, and revolutionary Automatic Office Assistants™ to aid management information monitoring and decision making.

Written in "C," Alis is initially available on UNIX* to large OEMs. It's destined to bring sanity to the mad tea party world of office automation.

*UNIX is a trademark of Bell Laboratories. Applix, Alis, and Automatic Office Assistants are trademarks of Applix, Inc.

Alis™

The next-generation office software system from APPLiX

Finally, some answers in Wonderland.

APPLiX, INC. 302 TURNPIKE ROAD, SOUTHBORO, MASSACHUSETTS 01772 (617) 481-4721

Circle No. 297 on Inquiry Card

See us at UniForum Booth #2452

many of the so-called metaphors is to what extent are they useful. A 9-inch screen (especially when measured diagonally) is no match for an honestly measured 6-foot desk when it comes to fruitful, concurrent clutter. Was it not Racine who wrote:

> *"Les Pixelles infinies de mon bureau*
> *"Surpassent celles de mon pauvr'ecran?"*

("My desk has infinitely more pixels than my poor screen.") [Editor's Note: No, it wasn't Racine; far too iambic for Racine. Try Verlaine.]

Is the answer bigger screens or smaller desks? Or no desk at all? Will trope shade into reality? Will my new 6-foot screen-desk combo be an honest 6-foot, or a sneaky diagonal 6-foot? No, I'm not quibbling — ask any Pythagorean. We are looking at something like the reciprocal of the square root of two! The screen makers have been pulling a recurring number like that since Yogi Baird invented TV in 1926. Before they invade the traditional desk market, I seek their assurances of exactitude and open disclosure in the mensural environment.

However you measure your desk, its relative size and the gaudiness of its ancillaries are vital parameters in establishing your place in the company's pecking sequence. Increasing size and gaud in suitably tasteless proportions is the pivotal challenge to contemporary desk designers. The phrase "Room at the Top," I believe, specially relates to desk-area and drawer space. I also maintain that there is a standard, hideously cozy, framed family portrait leased to all CEOs regardless of marital status. Sneak a peep next time you are on the carpet. Didn't you see those three complacent kids in your last job?

The screen people will need to perpetuate these follies if they seek to replace the corporate desk. Perhaps some mergers are called for — Thomasville and Silicon Graphics, for instance? Doubling the palette range and image resolution at each rung of the ladder would provide strong incentives for promotion. I suggest that that nasty color Cyan should be replaced by a regal, simulated leathery Red for all those earning over 100K. In order to gloat scientifically over your desktop bitmap densities, you must first urge

However you measure your desk, its relative size and the gaudiness of its ancillaries are vital parameters in establishing your place in the company's pecking sequence.

ANSI, ISO, ECMA (or any of that crowd) to adopt my longstanding proposals for standardized units for graphical granularity:

4 pixels	=	1 blyb
4 blybs	=	1 smydge
4 smydges	=	1 seurat

The new screen-desk will have to offer the diverse functions associated with both artifacts. In my case, I have always enjoyed a secret compartment for highly personal objects which I am not prepared to catalog at this time except to admit that, like you, I own a specially deformed paper-clip for ear-wax removal. My illustrated monograph, "Some Observations on the Ototopological Aspects of Paper-Clip Deformation," has yet to find a publisher, but I digress.

When an employee is fired, will the boss shout, "You've got three seconds to clear your screen!"? I see the employee scrambling through the manual. "Is it CLS or UC + ESC + ALT + XY = 0? Damn, damn!"

Bellsoft has just interrupted this essay. It has a Pop-Up Alarm Clock which pops up automatically at pre-set times with what in UNIX would be called a Self-Mailing. It has reminded me that Christmas is Coming, and UNIX REVIEW readers want a more *seasonable* column. People getting fired and cleaning out their ears and desks...where's the fun in that?

Sending mail to oneself for delivery at a predetermined date, and receiving it *on time* without scanning the mailbox is multijoy unbounded. The true narcissist (and aren't we all?) can peek ahead of time and build up a wonderful glow of anticipation which makes each day a winged chariot of delight. Between the "Don't forget AGM on 23rd," and "3 kilos Belgian Endive, 2 lbs. salt," you can receive regular back-pats: *You're doing great, Stan, or may I call you Mr. Wonderful?* I set this one to pop-up every five minutes, but depending on your *angst-du-jour* you can vary the frequency, or even make the interval random — *One Person's Meat is Another Person's Poisson!*

There are signs that self-mailing will have a profound effect on the entire US Postal Service. *No Man May Be an Island* —let's rephrase that — it may be true that *No Man Is an Island*, yet once

INTEGRATED OFFICE SOFTWARE

- Data Base & Forms Management
- Full Screen Interaction Document Processing
- Spreadsheet
- Telephone Directory
- Legal & Accounting Applications
- Calendar Scheduling
- XED PostMaster™ Executive Mail
- Type-setting

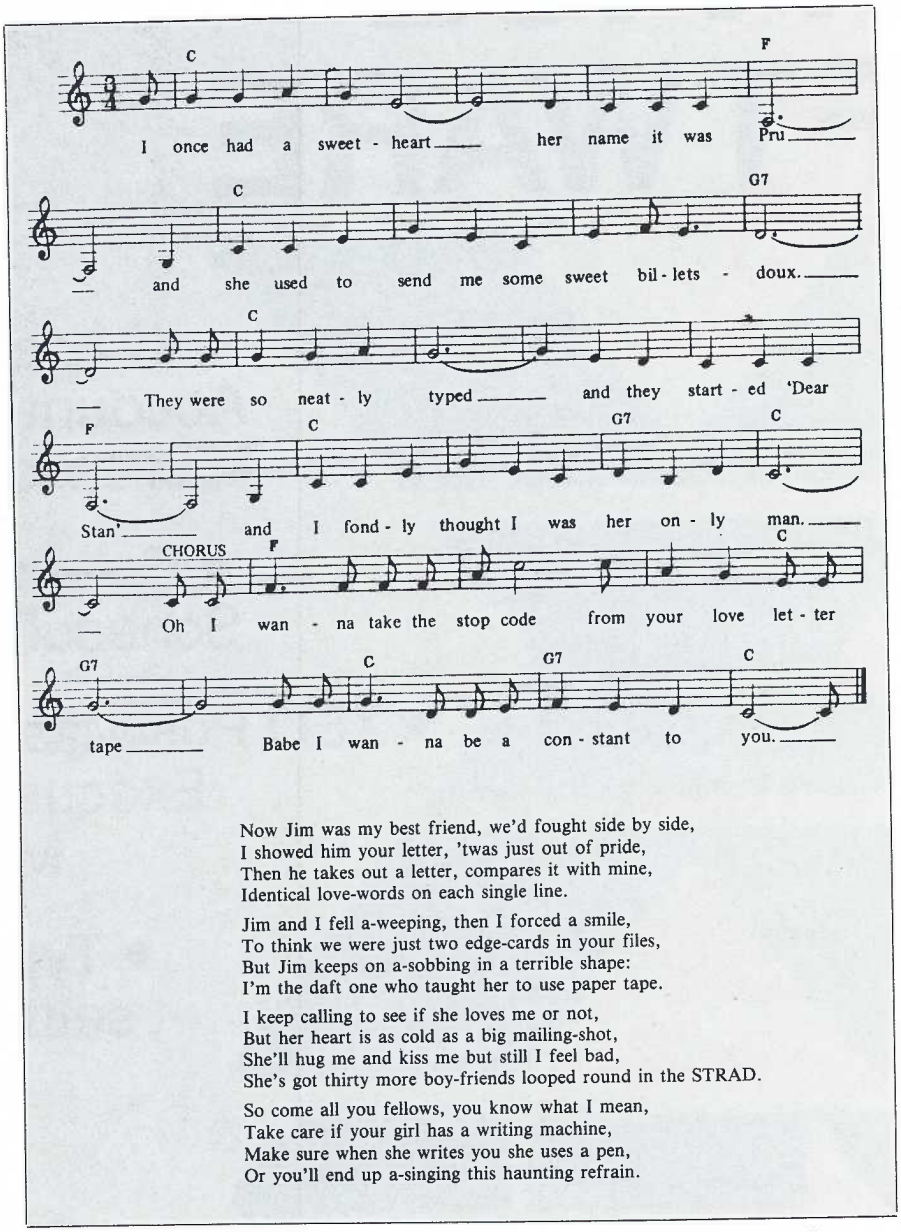
XED®

XED®

INTEGRATED OFFICE SOFTWARE

Box 3938 • Chatsworth, CA 91313 U.S.A. • (818) 884-2000 • FAX (818) 884-3870 • TWX 910-494-1716 • Intl. Telex 292 662 XED UR

Circle No. 296 on Inquiry Card



Now Jim was my best friend, we'd fought side by side,
I showed him your letter, 'twas just out of pride,
Then he takes out a letter, compares it with mine,
Identical love-words on each single line.

Jim and I fell a-weeping, then I forced a smile,
To think we were just two edge-cards in your files,
But Jim keeps on a-sobbing in a terrible shape:
I'm the daft one who taught her to use paper tape.

I keep calling to see if she loves me or not,
But her heart is as cold as a big mailing-shot,
She'll hug me and kiss me but still I feel bad,
She's got thirty more boy-friends looped round in the STRAD.

So come all you fellows, you know what I mean,
Take care if your girl has a writing machine,
Make sure when she writes you she uses a pen,
Or you'll end up a-singing this haunting refrain.

Figure 1 — *"Repetitive Love Letter Blues."* ©*Logo Music/ASCAP 1968.*

people experience the pleasures of junk-free, non-begging, unthreatening, well-composed, informative correspondence which can be answered or not as the mood dictates, I predict a massive swing towards insularity.

The worst kink of incoming mail from *others* is the mock personal, or what I call the *Dear Stop Code*, letter. I used to have fun guessing which WP package was responsible, and whether it had used a hard or soft hyphen in Kelly-Bootle, but that soon wore off — soon to be replaced by feelings of resentment. That people should go to all that trouble just to tell me I've already won a million dollars....wait a minute....let me read that again.

In the early days of Flexowriter automatic, paper-tape letter writing I composed the song shown in Figure 1. I think it retains a smydge of relevancy in spite of the shifts in technology.

The final Come-all-ye warning, of course, does not apply if you write to yourself. Fats Waller, were he alive today, would sing:

"I'm gonna sit right
down and interface
with WordStar,"
"And make believe it
came from you,"
"I'm gonna process
words so sweet,"
"Gonna scroll me off my
feet,"
"Control-X to the
bottom,"
"I'll be glad I gottom!"

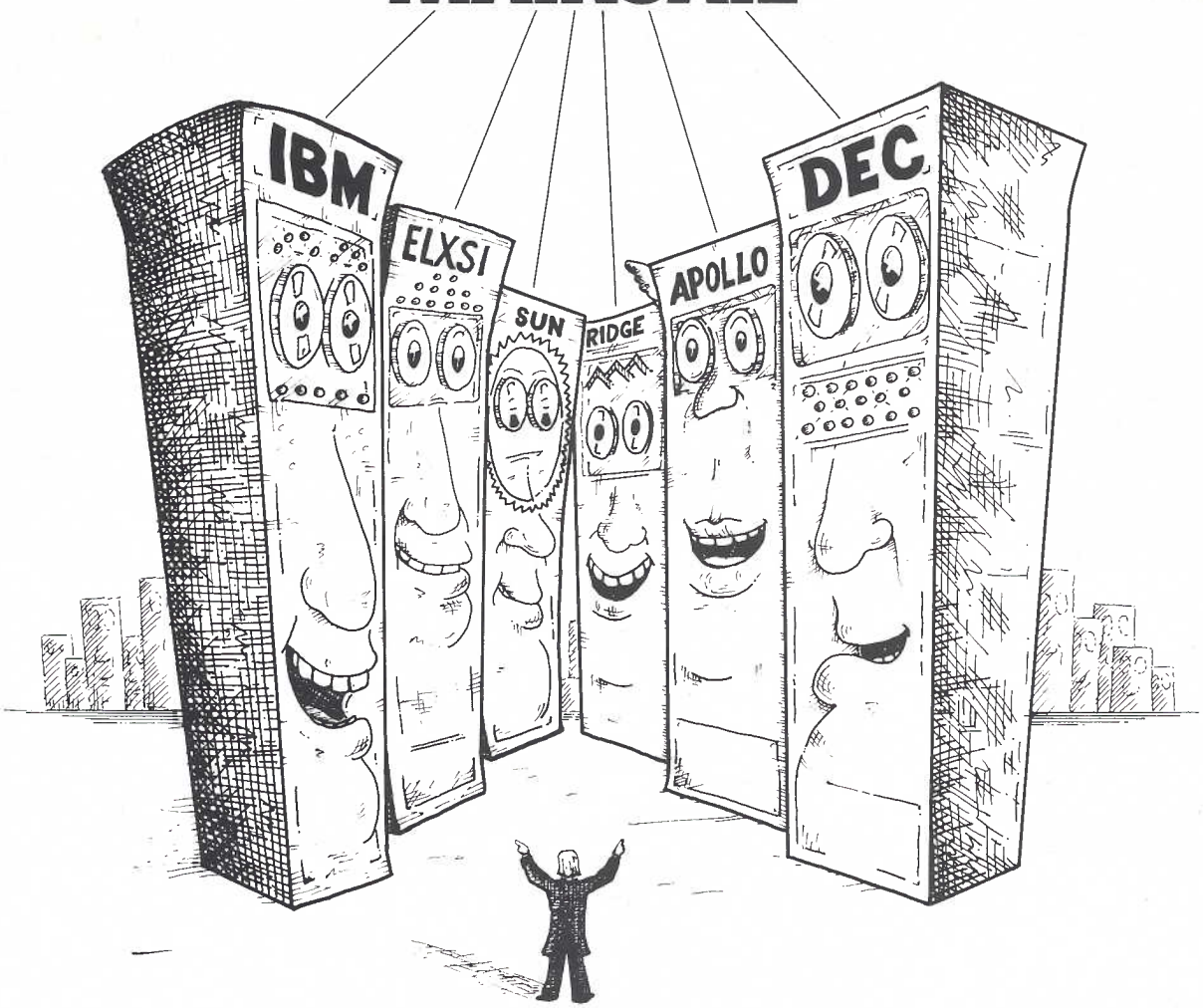
Would Fats Waller's SpellStar allow "gonna" and "gottom"? I know that mine rejects simple words such as "AI" and "SpellStar" — the latter being an example of recursive intolerance.

Since this is my first Christmas column for UNIX REVIEW, I hunted far and wide for a closing piece, a brief, timely emetic to throw up my feelings of goodwill and friendship towards the far-flung UNIX community. Who better to enlist than Robbie Burns of Immortal Memory, whose prophetic words are now imbued with new meaning:

"Noo awk that, an' awk
that,
"Gin foo' the bar, an'
awk that.
"Fu' mony greps oor
lang strings gang,
"God gie us grace tae
awk that."

Stan Kelly-Bootle has diluted his computer career by writing contemptuous folk songs for Judy Collins ("In My Life," Elektra K42009), The Dubliners and others. He is currently writing, with Bob Fowler, "The 68000 Primer" for the Waite Group, to be published by Howard W. Sams in the Spring of 1985. ■

MAINSAIL™



SPEAK MAINSAIL. BECAUSE NOT EVERYONE SPEAKS UNIX.

Most computer programmers like the UNIX environment. It gives them a convenient set of software development tools. There's one problem, however. Applications must often run on machines that don't have UNIX.

That's where MAINSAIL comes in.

MAINSAIL is a powerful programming language that can help cut your development time and eliminate software conversion costs. You'll be able to take advantage of the development power of UNIX, while retaining the ability to move to other systems. And you'll be amazed at how easy it is to learn this proven, versatile language.

If you like UNIX, but need to keep your portability options open, just let us know. We'll show you how MAINSAIL can accelerate your development of portable, sophisticated application programs—not only under UNIX, but under VAX/VMS®, VM/CMS®, and a variety of other operating systems. For details, contact us at: XIDAK, Inc., 530 Oak Grove Ave., Suite 101, Menlo Park, CA 94025, (415) 324-8745.

UNIX is a trademark of Bell Laboratories.
VAX/VMS is a trademark of Digital Equip. Corp.
VM/CMS is a trademark of IBM Corp.

xidak

Circle No. 295 on Inquiry Card

THE HUMAN FACTOR

What to expect in Dallas

by Richard Morin

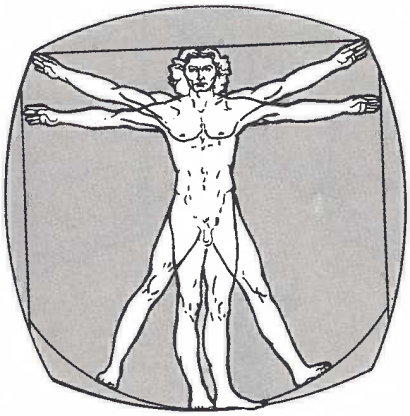
Several thousand people will attend the UNIX conferences held in Dallas next month. While some will be seasoned veterans, others will be neophytes, unaware of the folklore and social conventions surrounding UNIX. There is thus a crying need for a guide through the maze of UNIX etiquette and history.

Tell me if you find one. This column has a simpler objective. It merely attempts to describe the conferences and the organizations involved, offering some trade show pointers along the way.

THE ORGANIZATIONS

Cooperation between UNIX organizations has a checkered history, and the conferences reflect this. In July of 1982, the Summer Usenix conference (Boston) was informally co-sponsored by /usr/group and the Software Tools Users Group. The January 1983 UNICOM (San Diego) was sponsored by /usr/group, the Usenix Association and the Software Tools Users Group.

The January 1984 UniForum (Washington, DC) was sponsored by /usr/group and the Usenix Association. Next month, /usr/group will sponsor a conference and trade show. The Usenix Association will hold its own conference nearby, possibly in conjunction with the Software Tools



Users Group.

The sociology behind these events would more than fill an entire column. It might also involve UNIX REVIEW in a lawsuit. A mere description of the organizations involved will therefore have to suffice.

The needs of UNIX vendors serve as the principal focus of /usr/group ("user group"). It engages in standards activities, AT&T lobbying efforts and other functions that might be expected of a UNIX trade association. The /usr/group catalog, for instance, offers broad coverage of the UNIX marketplace.

The /usr/group talks tend to reflect the commercial nature of the /usr/group membership. Product announcement and description talks, disallowed by Usenix, are one example. Members of /usr/group tend to wear suits and ties and display a keen interest in the "bottom line."

Usenix ("use-a-nix") is more oriented toward the needs of the academic and research crowds, although a number of vendors also belong. Members generally wear T-shirts and jeans, and tend to get excited about the latest kernel hacks or a new version of **rogue**.

The group distributes tapes containing a wide range of contributed UNIX software, including everything from device drivers to games. The various levels of UNIX licensing restrictions are enforced, however, and the tapes are unfortunately available only to institutional members.

Usenix holds two conferences each year, and the talks tend to cover the experimental, theoretical, and, occasionally, sociological aspects of UNIX. Don't be scared off; the talks are often interesting and thought-provoking, and can generally be understood by mere mortals.

Some of the talks are humorous as well. Rob Pike of Bell Labs is one of the most entertaining speakers in the Usenix community. Dr. Pike is a self-proclaimed expert on hydroponic gardening in both sub-floor environments and areas above raised ceilings.

He is also recognized as the co-author (with Brian Kernighan) of *The Unix Programming Environment*. His talks are

4.2BSD

FROM NOW ON, CONSIDER IT SUPPORTED.

When it comes to Unix™ systems, "standard" isn't always good enough.

Experts agree that the most powerful and most technically advanced **Unix** system is the Berkeley version. That's why 4.2BSD from Berkeley is the operating system of choice for software development, networking, engineering, CAD/CAM and demanding scientific applications. Other **Unix** systems don't have the features advanced users require.

But 4BSD was developed at a university, so it has never had real-world support. User assistance, bug fixes, updates and enhancements have not been provided.

Now that's changed.
MT XINU, the 4BSD specialist, supplies:

- Fully supported 4.2BSD-based binary licenses (MORE/bsd) for VAX™ computers.
- 4.2BSD source support and source updates for current 4.2BSD source licensees.
- Enhanced 4.2BSD-based source software for new sites, with or without redistribution rights.
- Full support for a wide variety of DEC™ and non-DEC peripherals.
- Assistance for OEM's and hardware manufacturers developing 4.2BSD-based products.

MT XINU personnel have been involved with 4BSD development from the beginning. Now we are producing 4BSD performance enhancements, advanced networking, other **Unix** system extensions, and support for new peripherals and architectures. As a service, we distribute 4BSD bug reports and proposed bug fixes to the community. Our years of experience can speed and improve your 4BSD implementations.

4.2BSD. It's always been better than just "standard." Now, with MT XINU, consider it supported.

"We know UNIX™ Backwards and Forwards"

MT XINU™

UNIX™ SUPPORT FROM BERKELEY

Circle No. 294 on Inquiry Card

See us at UniForum Booth #2008

739 Allston Way, Berkeley, CA 94710 ■ 415/644-0146 ■ ucbvax!mtxinu!mtxinu

guaranteed to be both amusing and insightful. If he gives one in Dallas, don't miss it.

The Software Tools Users Group (STUG) was inspired by the Kernighan and Plauger book *Software Tools*, which shows how a portable user interface can be layered onto almost any operating system. The group has produced a relatively complete "Virtual Operating System" that provides a UNIX-like user interface on a variety of machines.

STUG serves the interests of folks interested in the existence of public domain software with a UNIX flavor. Although most STUG software is written in RATFOR (RATional FORtran), other languages are starting to appear on STUG distribution tapes. STUG members have rewritten a number of UNIX utilities in public domain versions. Thus, even UNIX users should look at the STUG distribution tapes.

One group will unfortunately be conspicuous only by its absence. This organization serves the interests of UNIX end users. Lacking large corporate or institutional backing, these users typically do not have source licenses.

End users constitute the largest and fastest growing segment of the UNIX user population. They need their own organization to lobby the industry, provide channels of communication and otherwise serve their needs. Unfortunately, no such organization exists at this time. Perhaps it will surface at some future conference.

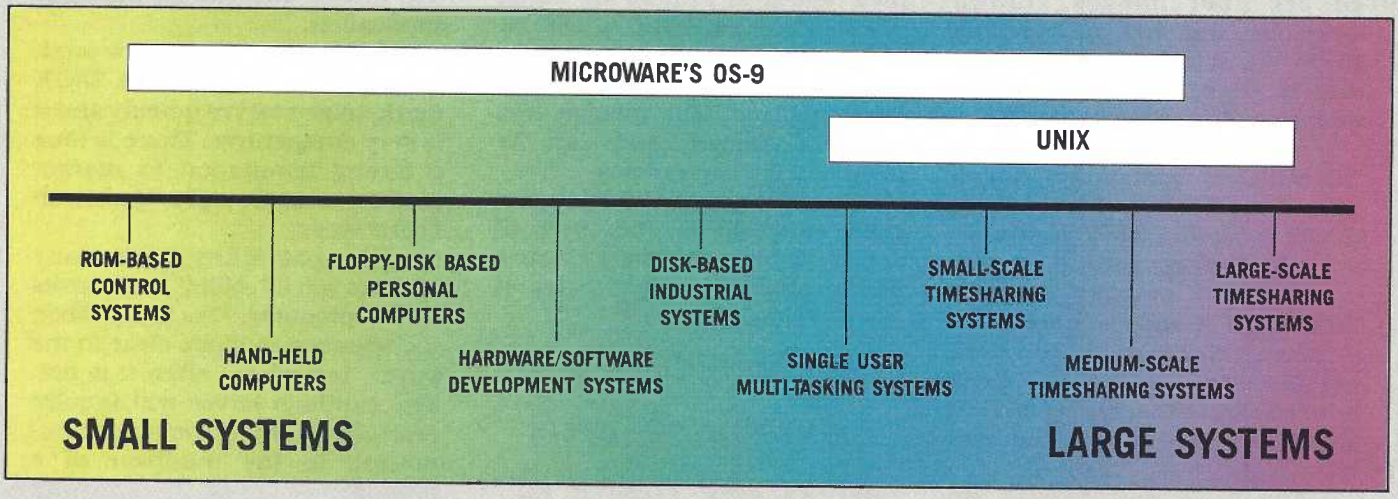
THE VENDOR EXHIBITS

The UNIX marketplace has matured dramatically in the last few years. Exhibit space first exploded from only a few booths to dozens and now it's counted in the hundreds. Around 1000 vendor booths are expected at the coming /usr/group trade show. The vendors will be hawking hardware, software, literature and a variety of services.

While many of the exhibits will be ho-hum, a few will be well worth examining. The following list is a very subjective guide to this columnist's favorite UNIX products. Probably the snazziest is the Silicon Graphics "Iris" workstation. Like most of the other UNIX workstations, it uses a Motorola 68010 microprocessor.

Unlike the others, however, it has a set of 12 proprietary VLSI chips that function as a "geometry engine." This allows the Iris to

Only Microware's OS-9 Operating System Covers the Entire 68000 Spectrum



Is complicated software and expensive hardware keeping you back from Unix? Look into OS-9, the operating system from Microware that gives 68000 systems a Unix-style environment with much less overhead and complexity.

OS-9 is versatile, inexpensive, and delivers outstanding performance on any size system. The OS-9 executive is much smaller and far more efficient than Unix because it's written in fast, compact assembly language, making it ideal for critical real-time applications. OS-9 can run on a broad range of 8 to 32 bit systems based on the 68000 or 6809 family MPUs from ROM-based industrial controllers up to large multiuser systems.

OS-9'S OUTSTANDING C COMPILER IS YOUR BRIDGE TO UNIX

Microware's C compiler technology is another OS-9 advantage. The compiler produces extremely fast, compact, and ROMable code. You can easily develop and port system or application software back and forth to standard Unix systems. Cross-compiler versions for VAX and PDP-11 make coordinated Unix/OS-9 software development a pleasure.

SUPPORT FOR MODULAR SOFTWARE — AN OS-9 EXCLUSIVE

Comprehensive support for modular software puts OS-9 a generation ahead of other operating systems. It multiplies programmer productivity and memory efficiency. Application software can be built from individually testable software modules including standard "library" modules. The modular structure lets you customize and reconfigure OS-9 for specific hardware easily and quickly.

A SYSTEM WITH A PROVEN TRACK RECORD

Once an underground classic, OS-9 is now a solid hit. Since 1980 OS-9 has been ported to over a hundred 6809 and 68000 systems under license to some of the biggest names in the business. OS-9 has been imbedded in numerous consumer, industrial, and OEM products, and is supported by many independent software suppliers.

Key OS-9 Features At A Glance

- Compact (16K) ROMable executive written in assembly language
- User "shell" and complete utility set written in C
- C-source code level compatibility with Unix
- Full Multitasking/multiuser capabilities
- Modular design - extremely easy to adapt, modify, or expand
- Unix-type tree structured file system
- Rugged "crash-proof" file structure with record locking
- Works well with floppy disk or ROM-based systems
- Uses hardware or software memory management
- High performance C, Pascal, Basic and Cobol compilers

microware® OS-9™

MICROWARE SYSTEMS CORPORATION
1866 NW 114th Street
Des Moines, Iowa 50322
Phone 515-224-1929
Telex 910-520-2535

Microware Japan, Ltd
3-8-9 Baraki, Ichikawa City
Chiba 272-01, Japan
Phone 0473(28)4493
Telex 299-3122

produce real-time dynamic shaded 3D color images. Silicon Graphics also has a tumbling Rubik's Cube that's really nifty and a flight simulator that's guaranteed to produce vertigo.

Another exciting machine is the MOSAIC workstation, which starts out with a National Semiconductor 32032 processor, a 32081 floating point processor and a 32082 memory management unit. It adds a proprietary graphics processor and ties it all together with a pair of high speed buses. With all that horsepower, it should be pretty powerful, so look for it.

The SUN workstations are notable for the fact that they can run without local disks, sharing files across a network. This distributed file system is implemented by a "remote procedure call" protocol. While its implementation is proprietary, SUN has released the specifications. The protocol is not specific to SUN, 4.2 BSD or even UNIX, and thus could be used to tie an arbitrary range of hardware together.

The Masscomp workstation has multiple-bit planes, allowing it to change very rapidly between screen images. In addition, its UNIX has been optimized for use in real-time data acquisition and process control applications. Anyone needing continuous sampling rates of up to 350K samples per second should talk to Masscomp.

A variety of other graphics workstations will also be present in Dallas. Most of these are black and white, do windowing and have around 1000x1000 pixel resolution. Several allow the addition of a color monitor, at around 500x800 resolution. The squeaking sounds of mice will lead you in the right direction.

A few raster graphics terminals will also be present. While these are able to achieve the same resolution as the workstations, their speed is limited by serial ASCII interfaces. A 1000x1000 bit image takes about a minute to transfer at 19,200 baud. Vector drawings and text involve less data, of course, and can be transferred more quickly.

The Teletype 5620 DMD is a good example of this sort of device. It has a Western Electric 32000 processor, 256K bytes of RAM, 64K bytes of EPROM and a three-button mouse. It also has a 800x1024 pixel bitmapped screen in a scintillatingly ugly shade of green. The 5620 does its own window management and has a number of utility routines for both the terminal and its host machine. All of this comes in a terminal available for a few thousand dollars.

Excitement, after all, is the name of the game.

Several low cost laser printers will be on display at the trade show. A cautious buyer will check the print quality, data transfer rates and graphics capabilities before making a decision. Not everybody needs full page bit-image printing, but more people could use it than realize that fact.

First, however, one needs a good source for such images. The Interleaf text composition software is one. It is a powerful, elegant and (sigh) expensive tool for interactively producing typeset copy. Be sure to see it (at the SUN booth) if you are tired of **troff** or simply need to produce good looking documentation quickly. Although it's a bit pricy, it could well be cost-effective for the right application.

Keep an eye out for new products, but be careful. The UNIX marketplace moves quickly and it is very competitive. There is thus a strong temptation to market products that are still in the breadboard stage.

If the past is any guide, many vendors will be selling promises as well as products. This is fine when the situation is made clear to the buyer, but all too often it is not. The cautious buyer will inquire carefully before planning or committing to the purchase of a promise.

Look for a demonstration of multivendor TCP/IP networking. The /usr/group trade show may repeat an experiment seen at the Summer 1984 Usenix trade show. Usenix set up an Ethernet and invited vendors to attach their products. This was more novel than some folks realized.

Previous Ethernet demonstrations have generally been limited to multiple machines talking simultaneously with their own devices. At this show, different machines busily conversed with each other. The fact that this was done on an ad hoc basis merely added to the excitement.

Excitement, after all, is the name of the game at conferences and trade shows. Stress levels reach dizzying heights, ideas spark other ideas, deals are attempted or made, and products are hyped. The UNIX phenomenon continues, and we watch, participate and enjoy.

Richard Morin is an independent computer consultant specializing in the design, development and documentation of software for engineering, scientific and operating systems applications. He operates the Canta Forda Computer Lab in Walnut Creek, CA. ■

The Profitable Way To Program In UNIX™

UX-Basic+®

Program in UX-Basic+ . . .

Before UX-Basic+ programming in UNIX meant programming in C. C gave you power and portability. Now, . . . go beyond C to add productivity . . . and profit to UNIX.

Program in UX-Basic+ . . . to be powerful

Ux-Basic+ provides high level functions, all with error checking syntax that access UNIX system calls and libraries including UNIX graphics and C-ISAM.™ This gives UX-Basic+ programmers the ability to tap the power of UNIX.

Program in UX-Basic+ . . . to be portable

Programs written in UX-Basic+ will execute unchanged on any UNIX machine and are completely device independent. UX-Basic+ has been ported to all popular UNIX machines.

Program in UX-Basic+ . . . to be productive

The UX-Basic+ Development System provides an interpreter with full featured editor, development and debugging tools, a pseudo-code compiler and runtime module. UX-Basic+ includes C-ISAM, BCD arithmetic and the easily readable structured, modular code is compatible with OASIS™ Basic.

Program in UX-Basic+ . . . to be profitable

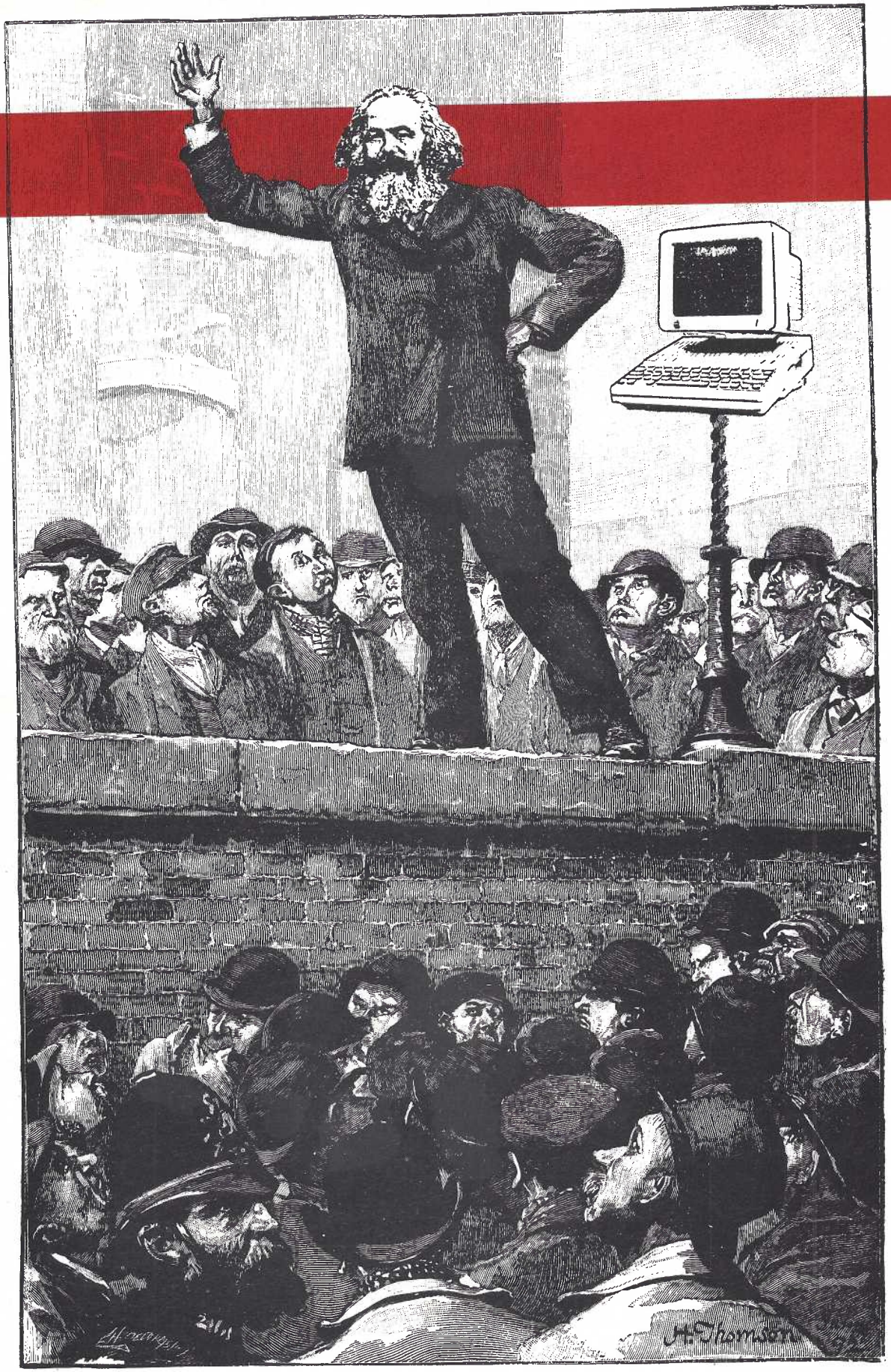
UX-Basic+ is the choice of leading UNIX software developers. Now, . . . the profitable way is yours.

UX-Basic+ is available through licensed hardware manufacturers and selected distributors, or call:

UX Software, Inc.

10 St. Mary Street, Toronto, Canada M4Y 1P9

TWX: 610 491 2138 **(416) 964-6909** TLX: 06-22492



UNIX FOR THE PEOPLE!

A manifesto for appealing to the masses

by Gig Graham

Many services and technical products that have become part of our lives during the last 50 years have succeeded because of the timely union of two key events: a reasonably mature, useful service or technology, and a dramatic change in the economies of scale for delivering the product. Typically, the products offered significant new capabilities that became commercially viable when they moved within the reach of a large number of people.

For those who hope that UNIX will take its position as a classically successful product, we're now at a crossroads. UNIX comes close to satisfying several of the criteria for economic success, but it's missing other important ingredients.

There are many historical models that illustrate the ingredients of a successful product or service. During the 1920s and '30s, Henry Ford pioneered mass production of automotive technology and transformed a product which had previously been a luxury item into a commodity for the middle class. Starting in the '40s and '50s, life insurance and other types of insurance allowed the middle class to share risks among a large number of people to guard against catastrophic losses. Relatively inexpensive hotels and air travel appeared in the '60s, thus opening the world to travelers who didn't own airplanes or villas in fashionable regions of the globe. During the '70s, calculators served as the forerunners of the microelectronic revolution and the explosion in demand for personal computers and other electronic products of the '80s.

In most cases, the technologies or concepts were reasonably mature well before they gained widespread commercial acceptance. An interested customer base and the economies of mass delivery permitted the products to flourish.

The same economic principles that determined the success of other technologies also apply to the software industry. In particular, a careful assessment of the industry's economics may help explain why UNIX has not been a wide commercial success and what must happen for it to fulfill its potential.

If economic history is a guide, UNIX must evolve from a product for expensive, luxury computers — such as the minicomputers of its origin — to a product that is available on the phenomenally successful generation of low cost personal computers. Widespread availability of UNIX machines is necessary to provide independent software developers with a strong incentive to write programs for UNIX. Developers will always evaluate the total population of machines over which their development and marketing efforts can be amortized. The larger the population, the more willing a software company is to assume risk in bringing new products to market.

Further, UNIX must capitalize on its mature technical position by clearly illustrating its superior capabilities to the mass market, and explaining why its features are unmatched by "substitute" products. An ideal UNIX product that cannot be easily duplicated in a non-UNIX environment must emerge which has widely recognized benefits.

HISTORY OF UNIX

Today, UNIX and its base of applications strongly reflect the system's initial orientation toward shared logic systems — systems where several users share an expensive CPU via terminals connected to the host by a serial link. The most popular examples of this type of system are the VAX and PDP-11, TRS 80 Model 16 and the Altos 586.

The configuration of these systems had a significant influence on the design of the software. UNIX software developers typically wrote either relatively small programs which did not monopolize the CPU, or programs which could be used with sophisticated paging routines. Programs that compromised the purpose of shared logic were obviously not appreciated. So full screen editors were kept only moderately interactive to economize on CPU cycles, and graphics applications were never seriously considered.

In contrast, non-UNIX software developers who wrote programs for 8 and 16-bit microcomputers assumed they had ex-

clusive use of the CPU. No one considered that other users might suffer serious degradations in performance because of a spreadsheet capable of consuming all of core memory. Further, these users had better graphics and interactive text processing at their disposal because, even on the early microcomputers, the bandwidth between video memory and the console terminal was faster than the serial interconnect of shared logic systems.

These differences in the environments used for software development are the basis of one of the major problems that bedevils UNIX to this day.

NEW UNIX WORKSTATIONS

During the last 18 months a new generation of UNIX machines has emerged. Unlike their shared logic counterparts, these new "supermicros" have dedicated processors and co-processors. Typically, they're described as "engineering workstations." Examples are the DEC Professional 380, and the Sun, Apollo, and MASSCOMP workstations.

By all accounts, the hardware is superb. However, the UNIX software which migrated to these machines was designed to accommodate limitations that no longer exist. The new machines are used by a single individual, and have high bandwidth, bitmapped video. They are not timeshared CPUs with low bandwidth monitors.

More significantly, the market acceptance of these machines has been sluggish. As a result, the current and anticipated base of machines is too small to justify a major marketing and development effort by a software company on behalf of a new mainstream UNIX product. The new products that do emerge are generally highly specialized with lofty prices — in the luxury class of software rather than in the line of products for the mass market.

UNIX clearly cannot look to the supermicro class of machines as a means to gain market share and demonstrate its unique capabilities to a mass marketplace.

UNIX ON PERSONAL COMPUTERS

In contrast with the market for supermicros, the personal computer market has been a success, largely because: (a) PC prices are affordable to individuals due to economies of scale in producing the components; (b) the machines are generally reliable and require little service; and (c) the machines are personal and dedicated in nature. In short, the PCs have become classic commodity items while the supermicros have appealed only to the technical elite.

An interested customer base and the economies of mass delivery permitted the products to flourish.

Perhaps most importantly, the independent software industry has responded with incredible enthusiasm to the introduction of PCs. Developers have shown they believe that they have a good chance of making a reasonable return on their investment, and indeed many software companies have flourished.

Unfortunately for UNIX, it was not available at a sufficiently early stage in the PC growth cycle to become an important factor in the design considerations of software development companies.

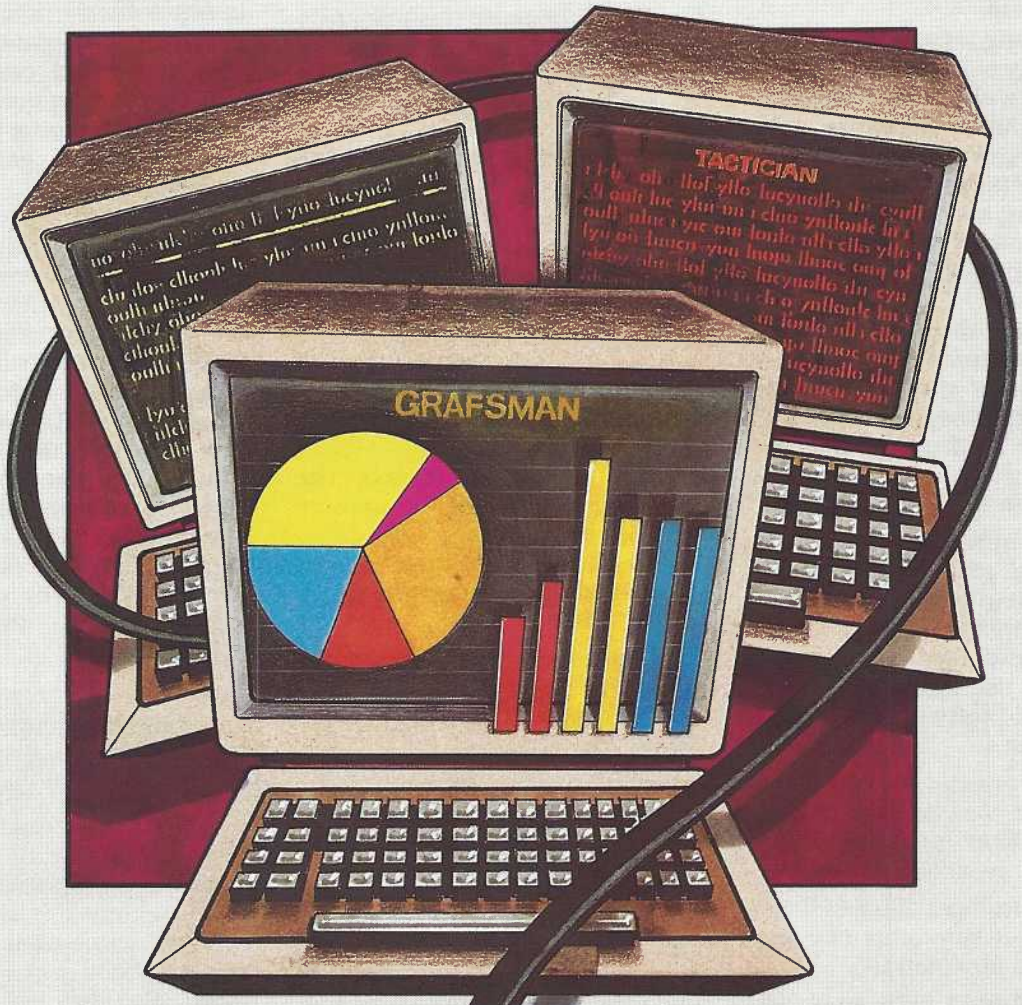
In fact, a dangerous trend developed in the evolution of microcomputer software due to the lack of a sophisticated operating system such as UNIX. With very few exceptions, PC-based application programs ignore the concept of an operating system standard; nearly all popular programs are tightly linked to the hardware features of IBM/Intel architecture. Typically, they are written in 8086 Assembler and generally address video memory directly. Other less obvious tricks are also employed to squeeze maximum performance from the PC architecture.

For UNIX to be successful in the PC marketplace, it must exploit the economic and technical weaknesses of the current generation of software and clearly demonstrate its superior features for long term growth. UNIX needs an action strategy over three time periods.

In the short term, UNIX vendors must sell PC customers on the basic capabilities of UNIX, such as multitasking and communications. Further, since UNIX works on many different computers, it should be the clear choice for managers who are most concerned with the costs of training employees in the use of different machines.

In the medium term, software developers need to be convinced of the superior economics of UNIX. Developers can write in a high level language and amortize their initial investment more rapidly than with Assembler/hardware-oriented programs. Further, the marginal cost of porting software

Fast...Versatile ...Powerful Tools



Integrated Productivity Tools from SouthWind Software provide today's professional with fast, versatile tools to organize, manipulate and display information with clarity and effectiveness.

TACTICIAN is an enhanced 1024 x 1024 spreadsheet which interacts with popular UNIX DBMS ie. UNIFY, INFORMIX and MICROINGRES.

GRAFSMAN's panel driven graphic editor helps create brilliant color graphics for reports and presentations using data from TACTICIAN, DBMS, or application files.

Integrated Productivity Tools were designed with the OEM in mind to allow for quick, easy incorporation into problem solving applications. IPT is available today for a wide variety of UNIX/XENIX based systems.

Call for further information about the fast, versatile and powerful Integrated Productivity Tools family.

See us at UniForum Booth #3337

SOUTHWIND SOFTWARE, INC.
4520 E. 47th St. So.
Wichita, Kansas 67210
316-788-5537
1-800-346-3025 EXT 234

SOUTHWIND SOFTWARE

Circle No. 290 on Inquiry Card

to new machines is lower, allowing developers to focus on marketing programs, instead of porting efforts.

In the long term, the stability and maturity of UNIX is essential. Customers want a software environment that is backward looking as well as forward looking. The multiple facets of software acquisition, training and use are too expensive to simply chuck out as new, more powerful hardware is introduced.

To meet these objectives, UNIX vendors must coordinate on the definition of standards. Standards for dedicated UNIX workstations involve a different set of problems than those for shared logic systems. The key issues follow:

1. A uniform procedure for incorporating graphics programs. The preprocessor PIC approach in System V is inadequate and does not accommodate the interactive capabilities of software such MacPaint does.
2. Interprocess communications, so that interdependent application programs can operate more efficiently.
3. Real time capabilities for high speed data throughput for networking applications.
4. A distributed file system for effective linkage of multiple micros.

Who is to lead the standards process? This author believes the leaders should be:

1. AT&T, because it is primarily responsible for new product development and is committed to System V.

UNIX needs an action strategy over three time periods.

2. DEC, because it has the largest base of innovative customers, a large and capable technical development team, and a commitment to 4.2 BSD.
3. /usr/group, to moderate the more focused objectives of DEC and AT&T.

SUMMARY

UNIX on PCs provides a valuable service to the marketplace. Many PC users will soon experience the limitations of a nonportable generation of software and will seek a better solution for shared resources, multitasking and other capabilities acknowledged as UNIX strengths.

But as the last two years have demonstrated, the commodity PC products will be the focal point for how the software industry invests in new software programs. If UNIX is able to demonstrate its value to developers and customers in a market that requires its capabilities, it will become a major force in the next generation of hardware.

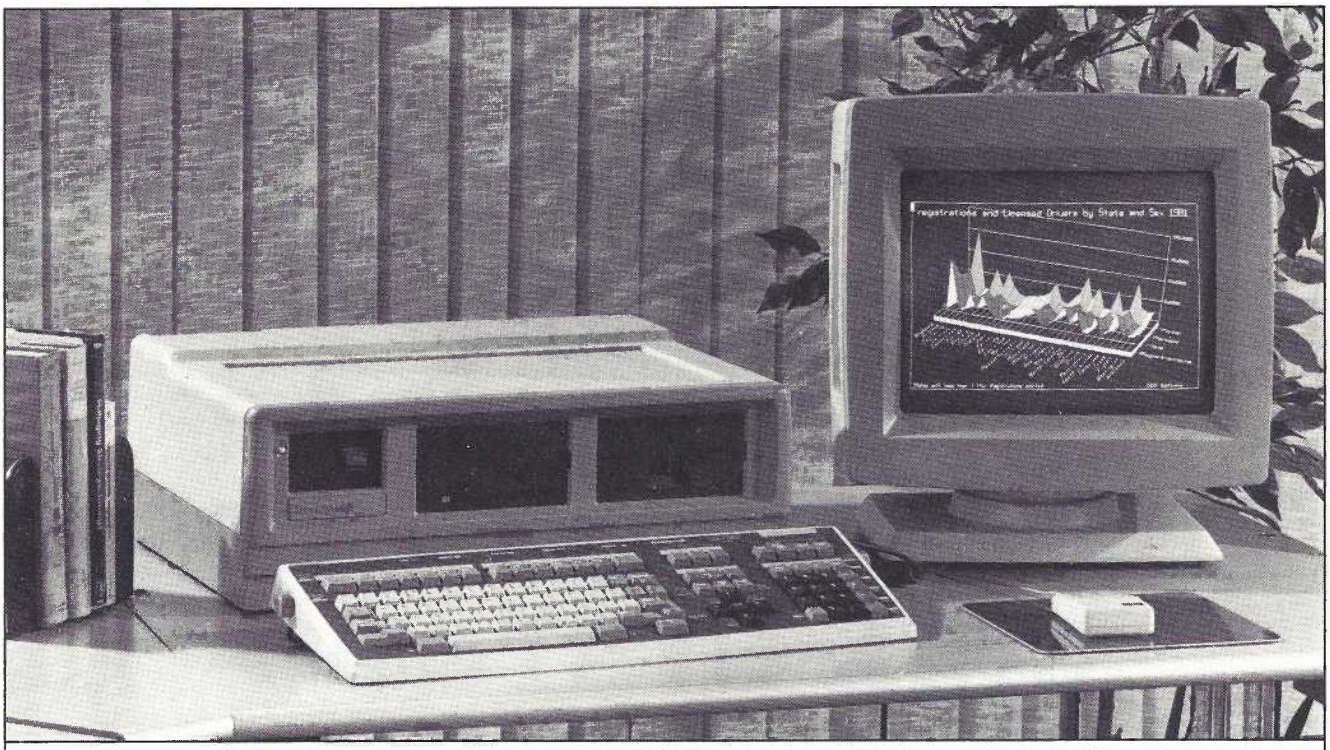
Today there are good implementations of genuine UNIX on personal computers, so at least half the ingredients from the cookbook for success are in place. UNIX works well on low cost computers and is within reach of the mass market. The second half of the solution is still missing. The UNIX market collectively has failed to demonstrate why it is superior to alternative environments. This is the challenge to be met by those who wish UNIX to succeed. The PCs are the battleground.

THE APPROPRIATE CONFIGURATION FOR LOW COST UNIX

So you are ready to buy and want to know what to get. An excellent configuration for a low cost UNIX machine is an IBM PC-XT, or a DEC Professional or Rainbow for graphics intensive applications. The major available alternatives are VENIX from VenturCom, PC/IX from IBM, and XENIX, from SCO.

A minimum configuration for VENIX, for example, requires at least 192K bytes of core memory and a 10 MB hard disk. Such a configuration for an IBM XT should cost less than $5000.

Gig Graham is co-founder and executive vice president of VenturCom, where he is responsible for marketing and business development. He holds an Economics degree from MIT and is a graduate of the Sloan School. ■



The microSystem NX from Honeywell. Quite possibly the most cost-efficient, full-featured UNIX*-based workstation.

Power, versatility, networking, and performance are all a part of the microSystem NX UNIX-based workstation from Honeywell. Featuring the MC 68000** processor, the NX is a super-micro engineered to compete with larger, more expensive multi-user systems.

The Professional's Workstation

microSystem NX is designed for engineers and other professionals who require sophisticated applications and a powerful tool for software application design and development, documentation support and technical research support.

Extensive System Development Tools

microSystem NX features a rich set of development tools that includes compilers, linkers, editors, formatters and debuggers. In addition, the system provides a "C" compiler, font editor, Fortran 77, and Pascal for enhancing commercial software packages and developing customized applications.

Powerful Applications Software

microSystem NX also provides application packages which take full advantage of the power of UNIX and include: a menu processor; a highly versatile word processor; extensive spreadsheet capabilities; a 3-dimensional graphics package; and a window manager that allows you to run up to 6 applications concurrently.

Advanced Networking

microSystem NX incorporates an advanced networking architecture that employs a local area network to provide a versatile UNIX-to-UNIX communications link between workstations.

UNIX-based Operating System

microSystem NX is driven by an enhanced version of the UNIPLUS+ operating system, with a complete set of utilities, development tools, networking options, and the ability to support a wide range of 3rd party applications software.

The new Honeywell microSystem NX. A sophisticated hardware/software system that gives the professional computer user more computer power for less money.

For complete information call 1-800-328-5111 ext. 2743 (in Minnesota call collect 612-870-2142 ext. 2743) or write: Honeywell Information Systems, Inc. MS 810, 300 Concord Road, Billerica, MA 01821.

Together, we can find the answers.

See us at UniForum Booth #807

Honeywell

Circle No. 289 on Inquiry Card

*Trademark of Bell Laboratories
**Trademark of Motorola Inc.

PROSPECTS FOR PERSONAL LOW COST UNIX SYSTEMS

The shape of things to come

by Rob Warnock

Every year, it seems, new faster chips come out while older chips get cheaper. Disks gain higher capacity and yet grow physically smaller. LCD displays emerge to challenge the venerable CRT. But how do these new hardware pieces really affect the price of a personal UNIX system? How low can it go? Will "everyone have one" — like calculators — or will "only the rich" professional computerists benefit?

With prices changing every day as new products are announced and old products respond, no one can predict exact figures. Still, a look at some of the tradeoffs affecting hardware system design of personal single-user UNIX systems can help prospective buyers and users navigate through the maze of prices, options and configurations.

The key tradeoffs that drive price are the machine's intended use or functionality (how it looks to the user), the basic architecture of the machine (how it looks to itself), the components, enclosure and possible configurations (how it stands in the world), and marketing (how the machine is sold and bought). We will look at each of these considerations, but remember that highly-optimized designs will make tradeoffs across these boundaries.

UNIX — A "PERSONAL" OPERATING SYSTEM

Let us assume that personal systems will be used, for the most part, for the same things that people usually do with larger UNIX systems: write (including edit), compile and execute programs; write, "compile" (format) and print documents; and write, read, send and receive mail.

As the prices of CPUs and memories drop, the trend is towards ever more sophisticated software that makes the user's interface with the system simpler and more powerful. The UNIX system has been accused of being large and complex, but for the range of facilities it provides, it is actually fairly compact. The UNIX system has grown, along with the times, adding such features as screen editors, job control and windows, but since many of the basic facilities to support these new functions were already in UNIX, the rate of growth has not been so drastic as with systems offering less to start with. A full-functioned system with a "modern" user interface using UNIX need not be much bigger than comparable non-UNIX packages. Still, even a minimal personal UNIX system will need several megabytes of kernel and utility code, and a CPU capable of giving crisp response to screen-oriented interaction.

Such a CPU is fairly easy to provide, with a 16-bit microprocessor. In fact, the main barrier to low cost UNIX may be that the system is *too* powerful!

COMPUTER ARCHITECTURE: "THE WAY IT WAS"

For the most part, UNIX has been run on multiuser systems of the minicomputer and supermini class. Even when run on micros and supermicros, it has been oriented towards multiuser operation. In this regard, the design of the usual hardware system to support UNIX has not been much different than the design of other

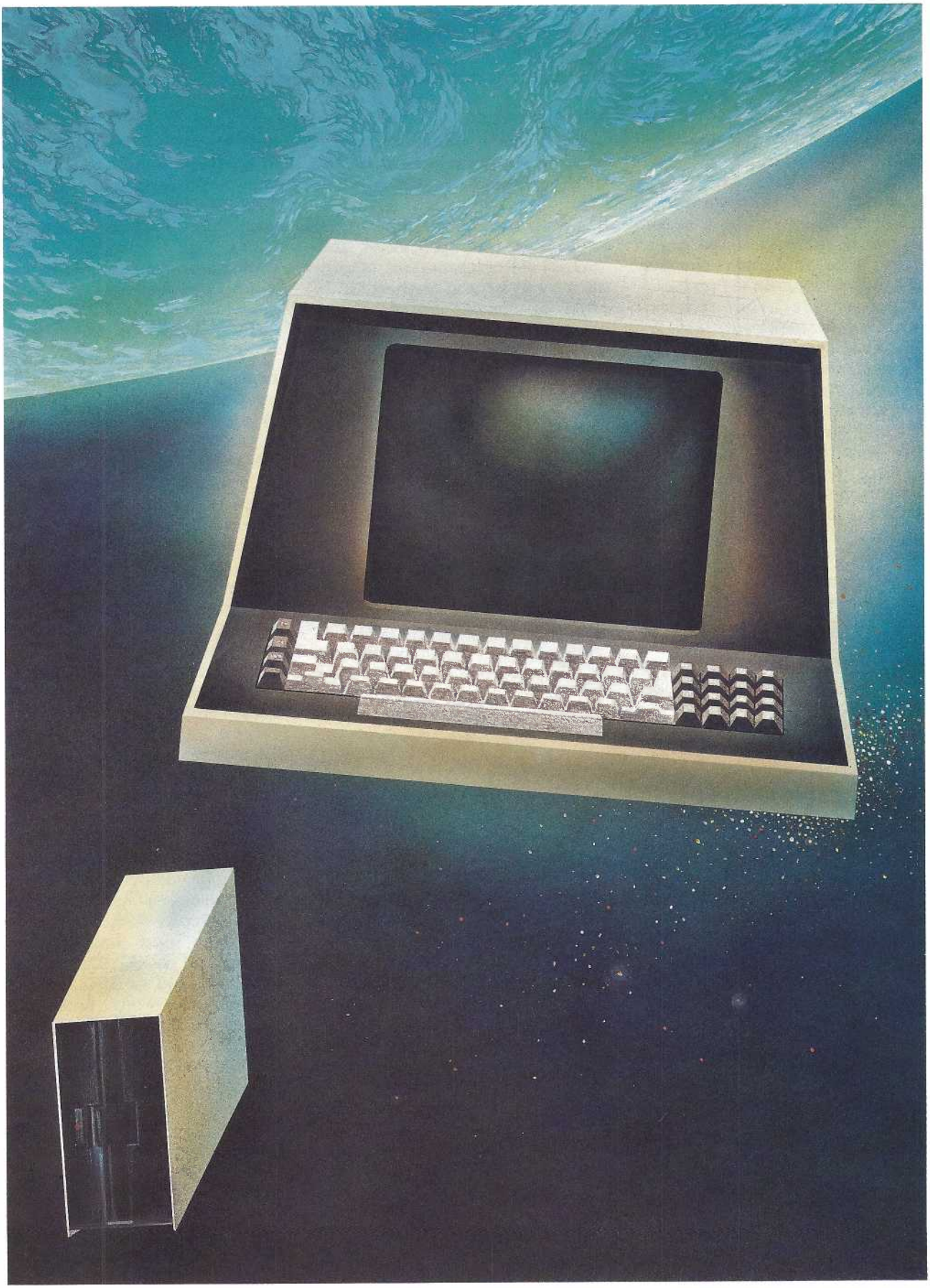


Illustration by Hyon Kim

general purpose timesharing systems.

Back in the days of the room-filling mainframe, the processor or CPU of a computer was a major expense of the system, so high-performance multiuser designs went to great lengths to "protect" the processor from under-utilization or inefficient utilization. Cache memory protected the CPU from waiting on slow main memory (since fast main memory was unavailable or unaffordable); DMA I/O controllers and even co-processors removed the load of handling data transfers to and from peripheral devices; and video controllers and terminals handled "painting" characters and cursor addressing operations.

The invention of the minicomputer provided a revolution in the cost balance of computing systems. Suddenly, a small system might cost tens of thousands of dollars instead of hundreds of thousands. Early minicomputer designs made quick use of these new economies to lower costs even further, letting the CPU do as much of the drudge work as possible. After all, the user's task couldn't continue until the I/O was done, anyway. Small minicomputers of the late 1960s did nearly everything with the CPU: assemble and disassemble RS-232 serial data into bytes; paint characters on a video screen a dot at a time; and even handle data transfers byte-at-a-time from mass storage.

But as these costs became the norm, minicomputers began to be used the way that only large computers had previously. Before long, multiuser operating systems appeared for these machines, such as TSS-8 for the DEC PDP-8, and later, UNIX for the PDP-11. Suddenly, the processor needed to be "protected" again. Not only did it comprise a substantial fraction of the cost of the system, but it became a performance bottleneck. Nobody wanted to wait while the CPU was doing somebody else's I/O.

With the introduction of the single-chip microprocessor, this "Wheel of Reincarnation" (a term used by Myer and Sutherland in 1968 to discuss graphics processors) turned again. As before, the low cost of the CPU allowed it to take on the work of I/O and display functions to form extremely cheap systems, and once again the expectations of what could be done with CPUs grew until they needed "protection" again. Current "megamicros" scarcely differ from the mainframes of yore in terms of cache, co-processors and "intelligent" I/O.

As the prices of CPUs and memories drop, the trend is toward ever more sophisticated software.

The 16-bit micro chips opened another chapter in price/performance, so it is not surprising that machines such as the Apple Macintosh have returned to the CPU-intensive style, including the Apple II's "Woz Chip" instead of a floppy controller. Single-user personal UNIX systems could be designed with the same tricks, and indeed the 512K byte Mac comes close to being such a machine, apart from its regrettable lack of an MMU and an integral hard disk. (The MMU is not actually necessary for UNIX, except to protect one process from another — which is less critical in single-user systems; one of the 68000's index registers can be used for process relocation. Rumor has it that integral hard disks may soon be available, even if not from Apple.)

NEW TRICKS FOR NEW DOGS

Of course, there are various new devices and design techniques which will help with low cost systems (though most of these are applicable to multiuser systems as well). Large liquid crystal displays (LCDs) are just beginning their life cycle, and are already competitive in certain applications where portability and power are important. Since they are being made with integrated VLSI circuits to drive and control them, their price should quickly slide under the "mature" CRT technology as soon as volume builds up. Vendors such as Data General are already offering portable computers with "full screen" (80 x 24) displays. (Should we look for a Flat Mac"?)

As noted above, a basic UNIX system will need several megabytes of program storage. While small winchester disks are getting cheaper, at least one alternative is likely to be competitive in the portable market (where winchesters have difficulty due to power consumption and sensitivity to mechanical shock). With all of the attention paid to 256K dynamic RAM chips, few have noticed that 256K EPROMs (Electrically Programmable Read-Only Memories), EAROMs (Electrically-Alterable ROMs), and still larger one-megabit "masked" ROMs (programmed at the time the chip is made) are readily available.

Even at current prices, a megabyte of ROM (eight chips) costs little more than an equivalent floppy disk, has "instant seek" access, doesn't need a controller and never wears out. By

How To Make Profitable UNIX* Systems Connections

System Integrators

Dealers and Retailers

Corporate Managers DP/MIS

Scientific/ Engineering

Value Added Resellers

Distributors

Small Business Managers

Manufacturing/ Production

OEMs

Software Developers

UNIX system installations increased from 20,000 in 1982 to 100,000 in 1984. And as many as 500,000 installations are expected in 1985! This expanding and profitable computer market demands a forum that's conducive to sales in the hotbeds of UNIX systems activity — San Francisco and Boston.

If you're marketing:

- multi-user, multi-tasking systems;
- networking products;
- applications software;
- programming aids;
- add-ons and supplies;

Then don't miss connecting at

• UNIX*Systems Expo/85 Spring
April 24-26, San Francisco.
• UNIX*Systems Expo/85 Fall
October 2-5, Boston.

Attendees at both shows will hail from the ranks of value-added resellers, systems integrators, software developers, distributors, dealers, retailers, and OEMs.

You can also count on volume end-user prospects from large corporations, small businesses, government and education. **They'll all be there!**

These highly-qualified attendees have heard much about the portability and potential cost savings of UNIX systems. They'll pay close attention to your products and services. They'll be most interested in the creative ways in which you can help them connect to the most cost efficient UNIX systems.

Make sure that the UNIX systems needs of these prospective customers contribute to your bottom line in 1985! Exhibit at both UNIX Systems Expos/85. You'll find a solid marketplace for your UNIX system products and services.

You'll connect at the UNIX System Expos/85!

Exclusive productions of
Computer Faire, Inc./
A Prentice-Hall Company.
181 Wells Avenue
Newton, MA 02159
617/965-8350
611 Veterans Boulevard
Redwood City, CA 94063
415/364-4294

Yes, I'm interested in hearing more about UNIX* Systems Expo/85

- ☐ San Francisco
- ☐ Boston
- ☐ Both

(UR)

Name _____

Title _____

Company _____

Address _____

City/State/Zip _____

Phone _____

Computer Faire, Inc. 181 Wells Avenue Newton, MA 02159

next year, it is likely that the several megabytes needed for a personal UNIX system will cost a good deal less than an equivalent small winchester, especially when the savings in power, space and controller are included in calculations.

As an additional benefit, when used with a microprocessor chip and a "paging" MMU, the entire C run-time library could be placed in each process's virtual memory using "shared libraries" (which some vendors such as Convergent Technologies are already offering) — thus saving the RAM space needed for library routines. RAM usage could also be avoided for any program code that is part of the "basic" ROM-based UNIX set. (For the nitty-gritty VMUNIX reader, ponder the use of "copy-on-write" for the data pages.) Of course, to put UNIX in ROM requires a higher level of confidence in the system software than has been typical in the past — considering that changing one's entire ROM pack for each bug found and fixed would be quite uneconomical, to say the least. (The use of EAROMs may alleviate some of this problem.)

For still larger savings in program storage, look to the emerging use of laser video and/or audio disks for software distribution. Because of the high retail volume, players for these disks are already low cost, and several vendors are discussing formats for storing digital data for computer software on them. The electronics to connect the computer will be quite low in cost, since video and audio programs are already being stored on the disks in digital form. Initial estimates of storage capacity range from several hundreds of megabytes to several gigabytes (thousands of megabytes) per disk. [Editor's Note — This is not a misprint.]

The main barriers to immediate introduction are likely to be the selection of a standard recording format and issues of software protection. The standards issue will surely be settled sharply if IBM gets into the act as rumored, while data protection concerns will probably give rise to unique encryption schemes for each program, with separate distribution of keys (probably on floppies or via data networks).

The feeling arose that the CPU should do as much of the drudge work as possible.

Similar techniques have already been used with video cassette recorders, but have not been popular due to lack of support by major manufacturers, the slowness of random access to files on tapes, and the fact that VCRs use analog recording techniques, requiring external coding and error correcting circuitry that is already an inherent part of the laser disks.

Finally, if — as some disk player manufacturers are promising — writable laser disks appear at affordable prices (say, the same as a winchester disk), the problem of backup files will be solved! You'll never have to throw anything out again. (Of course, you may have trouble *finding* it...)

THE COMPUTER THAT LIVED IN A SHOE(BOX)

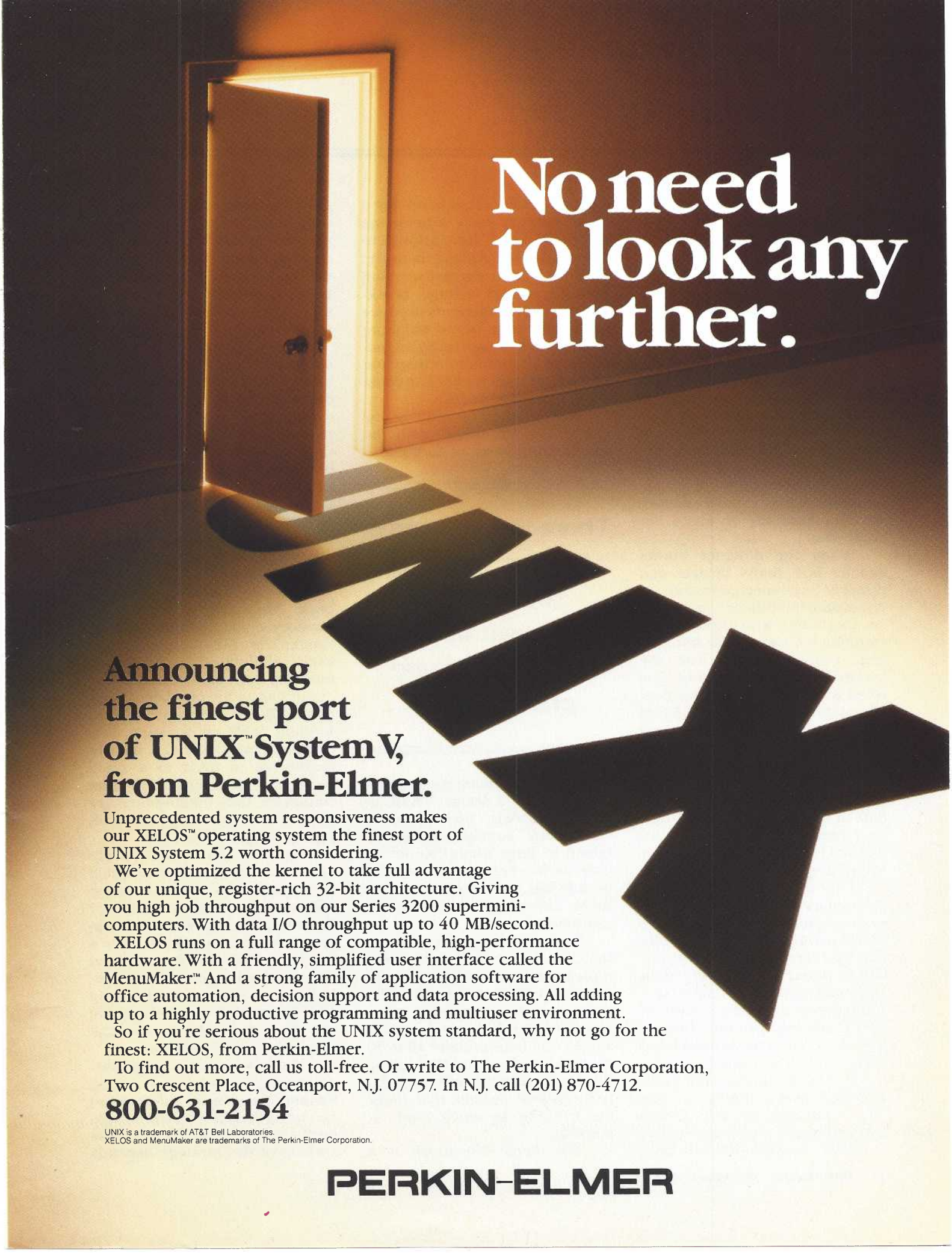
By now, you may have gotten a glowing feeling that all this will be yours for a mere pittance. Well, there are still some "minor" realities that stand between us and the continued rapid price declines we have come to expect. One, mentioned above, is that we now expect more from a system. Where we were once happy with 8 or 10 kilobytes of system software ("ROM Basic"), we now demand megabytes of high-quality code.

Another is that CPU and memory chips and disks still occupy physical space, they still have to be mounted somewhere, they still have to be connected to each other and the outside world, and they still have to be supplied with power, cooling and shielding.

Connectors and cabling are mechanical rather than electronic components, so as electronics costs drop, the cabling (which has a *rising* labor cost) becomes an increasing component of the cost of a total system. The new FCC restrictions on spurious emissions by computing equipment have accelerated this trend, particularly for systems which are used in a "home" environment (FCC Part 15 Subpart J "Class B Computing Devices").

These regulations really are needed, since early home computers were notorious for tearing up neighborhood radio and television reception. When we were still in the "Wild West" of home computing, an occasional "gunshot" from a rambunctious computer on a hot Saturday night was tolerable (unless *you* were the one trying to watch football!). Now that things are "civilized" and nearly every home has some kind of computer (even if it's hidden in a video game), that sort of anti-social behavior isn't allowed. Still, it's a cost that we haven't been used to in the past.

It's not only cables that cost



Announcing the finest port of UNIX™ System V, from Perkin-Elmer.

Unprecedented system responsiveness makes our XELOS™ operating system the finest port of UNIX System 5.2 worth considering.

We've optimized the kernel to take full advantage of our unique, register-rich 32-bit architecture. Giving you high job throughput on our Series 3200 supermini-computers. With data I/O throughput up to 40 MB/second.

XELOS runs on a full range of compatible, high-performance hardware. With a friendly, simplified user interface called the MenuMaker™. And a strong family of application software for office automation, decision support and data processing. All adding up to a highly productive programming and multiuser environment.

So if you're serious about the UNIX system standard, why not go for the finest: XELOS, from Perkin-Elmer.

To find out more, call us toll-free. Or write to The Perkin-Elmer Corporation, Two Crescent Place, Oceanport, N.J. 07757. In N.J. call (201) 870-4712.

800-631-2154

UNIX is a trademark of AT&T Bell Laboratories.
XELOS and MenuMaker are trademarks of The Perkin-Elmer Corporation.

PERKIN-ELMER

more, but brackets, screws, boards, even the little rubber bumper feet under the keyboard. In fact, each additional component or piece that must be manipulated brings with it the cost of "handling." This is why factory automation, or "computers building computers," has become such a key issue.

One of the main ways to hold down such costs is to simply not incur them. Thus more and more systems are being supplied in bundled or packaged configurations with everything included in the main cabinet — and no options offered (and therefore no connectors, brackets or space for additions).

When the designer knows exactly how many devices can possibly be connected to each internal data bus (that is, exactly the set that come with the system), components can be left out that were needed to deal with the possibility of external options. Bus terminators can be lightened or eliminated; lower-powered driver chips can be used; and in some cases bus drivers can be eliminated altogether. Each of these saves space, power, parts cost and handling cost. "The best part of a design is the part that isn't there" is a quote often attributed to Gordon Bell, the former V/P of Engineering at DEC.

The use of CMOS (Complementary Metal-Oxide Semiconductors) can help somewhat, since CMOS parts have virtually no idle current except static leakage, but CMOS power rises rapidly with switching speed (each "state transition" costs a certain amount of energy; energy per unit time is power). At the speeds associated with high-performance computers, CMOS may use the same or more power as NMOS or even TTL. And unfortunately, CMOS parts still carry a premium price.

Power supplies deal with bulk electrical phenomena involving a lot of energy. Such subsystems have physical limitations to miniaturization. Among them are: heat dissipation, current carrying limitations and voltage breakdowns. The components used are physically large (some weighing pounds), and are not easily accommodated by automatic insertion equipment used for ICs and other small electronic parts.

Power supplies tend to be priced "by the watt," with the rule-of-thumb cost of "a dollar a watt" having remained nearly unchanged over a decade. (Well, to be fair, modern high-efficiency "switching" supplies can be obtained in large quantities for as little as 50 cents-a-watt, but the decline has certainly not tracked RAM chips, just to cite an example.)

There are still some "minor" realities that stand between us and continued rapid price declines.

Each TTL chip removed from the CPU, I/O or memory system represents that much less power needed. A 25-cent Schottky TTL high-speed gate can easily mean an additional 25 cents in power supply cost (plus perhaps 10 to 50 cents in handling and testing). So this is yet another reason for tightly integrated designs that make the CPU do as much work as possible.

The Apple Macintosh is a good example of a high level integration of package, architecture and power supply. Those who complain about the lack of internal "option slots" have missed the point: those option slots cost a lot more than the bare price of the connector.

There is a valid side to the complaint as well. Low cost achieved at the expense of customer satisfaction wins no awards, and one of the things people often want to do is expand their systems. A tight level of integration that includes soldering the RAM chips to the CPU board (allowing auto-insertion, saved sockets and increased reliability) means that upgrading memory size requires replacing the entire board.

Recently, manufacturers have been supplying a number of memory chips on a single little "stick," or SIP (Single In-Line Package). Each SIP may contain from 64 to 512K bytes of RAM, depending on the number of chips on the SIP and their size. With one megabit RAM chips, such SIPs could contain as much as two megabytes of RAM in a small package with a single low cost connector, thus making memory upgrades economical even in tightly integrated designs.

On the other hand, a high-speed bit-serial local network such as AppleBus or Ethernet is an ideal way to expand a system for those applications where it is appropriate. The cost is low: only a few bits of I/O need be protected and shielded; one internal controller function is shared among all of the possible external options; and the power consumption and bus load is fixed and low.

The power supply, additional processing power and buffer memory (such as for a disk drive) can be packaged with the option, itself a tightly integrated unit. The success of this strategy depends

on the ability of the manufacturer(s) to predict the combination of features users will want in each subsystem box, and to make them in large enough quantities to recover the design costs of such tightly integrated designs.

"HURRY, HURRY, ONLY ONE PER CUSTOMER!"

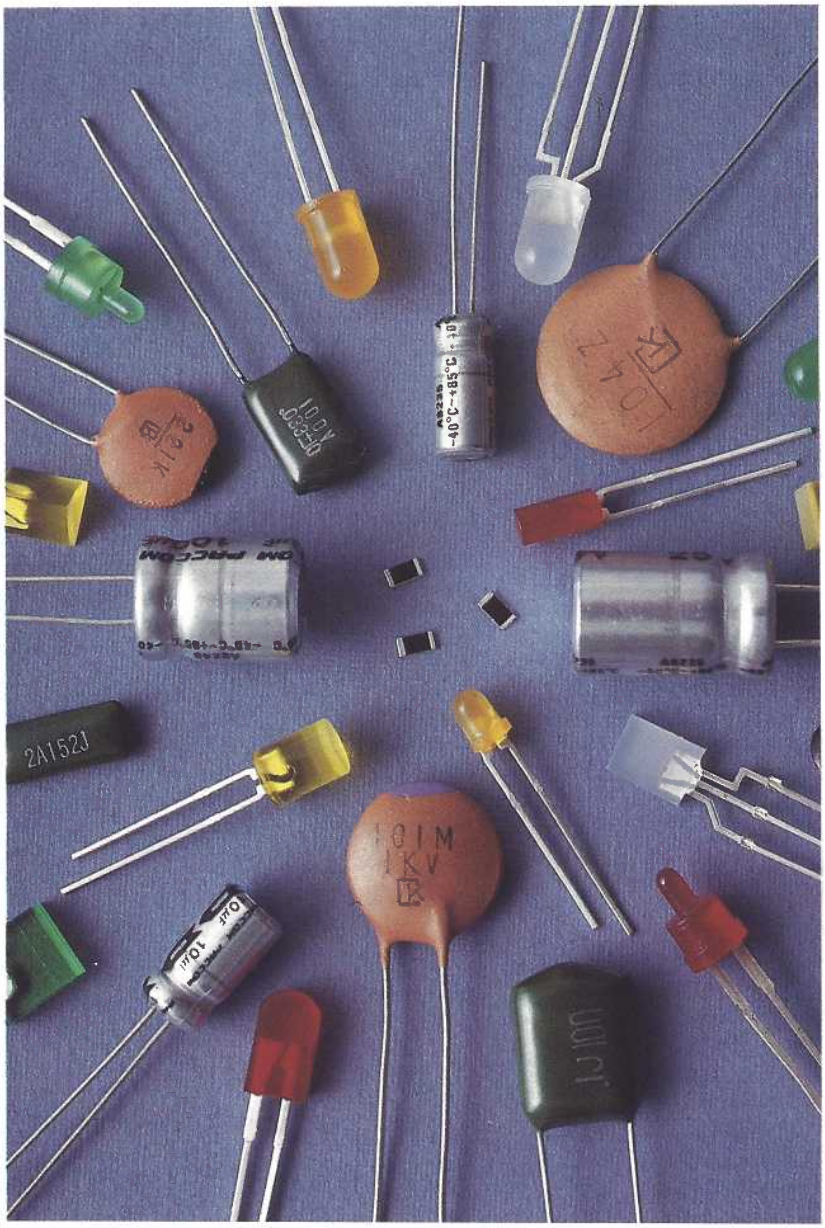
In order to reap the benefits listed above for tightly integrated packaged designs, systems must be sold in fairly high quantities. As noted, this means a limited number of options, themselves in separate packaged option subsystems. Local network techniques can preserve a degree of flexibility, but as software becomes ever firmer (whether in ROM or video disk), even this will be limited unless higher-level standards arise which define "network peripheral objects." (Smalltalk, anyone?)

The critical factor will be whether or not some manufacturer is willing to take the risk that the market for low cost personal UNIX systems is large enough, and active enough, to make money.

The home computer market, until now, has not flourished beyond the amateur computerist and the professional dabbling at home. Part of the reason, surely, is that systems offered for home use have, until now, been too limited in capability to support the very sophisticated user interfaces that would make the computer a true home "appliance."

But rather than being "too complex" for the home user, a UNIX system as hinted above, with megabytes of memory and gigabytes of packaged software, might just begin to be adequate for the task.

Rob Warnock is an independent computer architecture consultant with nearly 20 years experience in data communications hardware, and real-time and time-sharing operating systems. He has worked for Digital Communications Associates, Xerox XTEN and was the system architect for the Fortune Systems 32:16. Mr. Warnock currently resides in Foster City, CA. ■



Paul Kotz/West Stock

PUTTING THE PIECES TOGETHER

Strategies for affordable UNIX

by Walter Zintz and John Bass

The UNIX system boom so far has bypassed computer hobbyists and very small businesses. Price is the only reason. Truly inexpensive UNIX systems can't be found at Main Street dealerships or ads in the back of Byte, so the impression has gotten around that UNIX is strictly for well-heeled businesses and universities. But, yes, there *are* ways to acquire low cost UNIX systems. That's real UNIX, right now, running on serious hardware — and not all the ways for getting it require technical know-how.

The route for non-technical people leads to the IBM PC-XT. Used PCs are readily available at very reasonable prices: hard disk models go for between $1800 and $3000. UNIX tailored for the system can be had from several sources, typically for around $800. That's all you need to have a complete UNIX system — for as little as $2500.

For people on a tight budget, a dual 400-800K byte floppy-only system is usable with some tinkering. Typically, such a system will perform nearly as well as an MS-DOS floppy-only system. The chief inconvenience is that the root floppy is not dismountable, thus requiring a minimum of two drives. Systems such as this are generally used as startup systems until a cheap hard disk upgrade can be purchased.

The PC performs reasonably well as a single-user system in most applications, even without a hard disk. Most of the UNIX implementations for the PC are reconfigurable, making it rather easy for even a non-technical user to start with a floppy-only system and add hard disk capacity according to need. The main hazard for the non-technical buyer is avoiding software vendors who sell "UNIX systems" offering little more than a kernel, which in effect ensures that needed utilities must be purchased later as expensive extras.

For people with a little technical skill, DEC Q-bus may well be the way to go. Used Q-bus boards are widely available at low and stable prices. An 11/23 system can be cobbled together this way with 256K byte, 8-inch floppies, and a 10 to 20 MB, 5 1/4-inch hard disk for a sum total of $2000 to $4000. Single-user and multiuser UNIX is available from a number of DEC value added resellers and DEC itself. The cost ranges from $800 to $2400, and DEC's resellers will often sell it for less. This means you can build a system with substantially more performance than a comparable IBM PC system — for essentially the same price.

If you want the whole system built into the terminal — as with the IBM PC — DEC can still be the system of choice. An old series of DEC terminals (VT71, VT72 and VT1XX) had a Q-bus cage built into the terminal case. Used units are available for as little as $75 to $500, and as a bonus you get what the terminals were built for: high speed wordprocessing. A good keyboard and a memory-mapped display allow you to work at far higher speeds than RS-232 could ever hope to keep up with.

An extra $1000 or so will get you an 11/73 CPU board in place of the 11/23 — with performance that has to be the sleeper among low cost UNIX systems. The 11/73 uses DEC's single-IC version of the PDP 11/70 minicomputer. The 11/70 was the biggest and fastest of the PDP-11 systems; UC Berkeley used 11/70s to handle 40 or more users at a time. Various benchmark tests of the 11/73 show it to be a little faster/slower than the 11/70, primarily because it has nice extras like memory management and cache memory built into the IC. The 11/73 will also run 2.9 BSD UNIX, which is almost as powerful as 4.2 BSD and considerably more solid than earlier versions of 2 BSD.

A breakdown of the costs associated with building an 11/23 is provided in Figure 1.

Even if you don't wish to resort to components, complete

	USED	NEW
System chassis & power	$ 250	$1200
11/23, mmu, fpu and 256K byte	900	1800
RX02 type floppy and ctlr	500	1700
20 MB hard disk and ctlr	700	3500
Serial, parallel, misc	300	700
Terminal & printer	500	1400
Total hardware in parts	$3150	$10300

Figure 1 — *An itemized list of costs associated with building an 11/23.*

systems can often be purchased for a discount of 25 to 40 percent. This is particularly true of used DEC systems. Processor boards for the 11/73 are available on the market for about $1700, while 11/23 boards can generally be had for about $700. For the extra $1000, you get a system that's about three times as fast and runs most of the UNIX software released. Software for the 11/23 is much more limited. It's also interesting to note that the 11/73 is faster than most MC 68000 systems.

Hardware experts can create extremely flexible low cost UNIX systems by using the S-100 bus. Almost every board imaginable has been built for the S-100 bus, and there are plenty of used boards available. Dual Systems and Cromemco both build S-100 UNIX systems using the 68000 CPU. A CPU board and a port of UNIX from either company could serve as the core of a complex system. Using this approach, you should be able to create a minimum UNIX system for less than $3000.

A PROPOSED KIT

Lower cost UNIX should be available early in 1985, in kit form, from a California startup venture. This group has promised board-level kits (with components already soldered into the boards, leaving the user to simply assemble the boards) in both floppy and hard disk versions. A kit to build a system with 256K bytes of dynamic RAM, a pair of RS-232 ports and a 1 MB floppy drive will sell to end users for $1500, complete with a single/multiuser UNIX license and software. An additional 33 MB of hard disk will bump up the user price up to $2750.

The impression has gotten around that UNIX is strictly for well-heeled businesses and universities.

These prices are the result of several careful design decisions and a bit of hard buying. The first key decision was to go with the 68008 CPU. This is the familiar 68000 with an 8-bit I/O bus, which cuts the complexity and cost of the system without seriously degrading 68000 performance. The second key is a single-board design that's been field-proven. A CPU board that's similar, apart from using smaller RAM chips and a SASI interface to the disk instead of SCSI, has been on the market for some time now.

The hard disk capacity has been increased by more than 50 percent above the disk manufacturer's nominal rating by going to 1K sector size and breaking away from traditional track encoding. The standard, almost universal, encoding scheme for both floppies and hard disks on small systems is MFM, long used by IBM. The industry originally went with MFM because of IBM's backing and stayed with it for floppies because a standard made it possible for floppies written on one machine to be read on another. However, two micro manufacturers, Apple and Micropolis, have sold systems with non-MFM floppy encoding, and the only noticeable result has been some customer mutterings about incompatibility. Hard disks on small systems have also drifted into MFM, even though there was no need for a common standard in this area since MFM is familiar and easy to implement in hardware. But with MFM encoding, only half the bits on the disk tracks hold data; the other half hold clock pulses.

These kits will use RLL2,7 encoding, which needs far fewer clock pulses. RLL2,7 is more difficult to put into hardware — the problems being signal-to-noise ratio, bandwidth and frequency stability. The hi-fi industry had these same problems more than a decade ago, when moving-coil phonograph cartridges were developed. It overcame the problems by putting a small pre-preamp right on the turntable at the base of the tone arm. These disk drives use an analogous solution, in the form of a very small chip

that mounts on the disk's head arm itself and amplifies the pulses from the disk before they're carried by wire to the disk controller. This chip was developed to handle the output from 14-inch disks, which need about twice the bandwidth of the 5 1/4-inch disks included in these kits, so there's plenty of what the hi-fi industry calls "headroom."

The most surprising aspect of these systems is that the board itself is completely assembled. The user needs only to take the last few steps leading to a fully assembled system. The engineer of the startup manufacturing the kits indicates that the savings his firm realizes on those last few steps is just the tip of the iceberg. It also seems that the boards do not need to be certified by the FCC for meeting noise restrictions protecting radio transmissions — as assembled computers do. Neither are the boards bound by state laws requiring assembled computers to be accepted or listed by Underwriters' Laboratory. Eliminating the elaborate testing routines for these certifications has cut costs for the startup company by well over $100,000. People who buy kits, moreover, don't usually demand the sort of hand holding support that buyers of assembled products often insist on. These savings apparently get passed on to consumers.

Advertising by the startup is expected to begin in January, with first production shipments in February.

CUTTING SOFTWARE COSTS

People who want low cost UNIX systems will undoubtedly also want low cost software. The UNIX community doesn't yet have a tradition of cheap semi-commercial software packages, as CP/M has had for so long. Despite the fact that much of the software was developed at public universities or with grant dollars, it is essentially restricted to UNIX source license holders because the principal software distributors, UC Berkeley and Usenix, are worried that anything that even *runs* on UNIX might be the property of AT&T.

The poor man's best hope is thus Mike Myers of Ann Arbor, MI. In the finest tradition of philanthropy, Mike has thrown open the use of his System III Altos 68000 system to anyone who cares to dial in, free of charge, 24 hours a day. Mike has quite a bit of software online that is in the public domain and free to anyone who cares to take it. Much of the software comes from the C Users Group in Kansas. Mike has also installed a powerful conferencing package that's used to maintain an ongoing dialog among users across the country — sort of a poor man's substitute for Usenet. To join the festivities, just dial up Mike's system at 313/994-6333; prompts will tell you how to set up an account. The system will accept AT&T and Racal-Vadic type modems at speeds of up to 1200 baud. Of course, with five ports and 559 accounts, don't expect to get on the first time you dial in.

Walter Zintz, as founder and executive director of Uni-Ops, sponsors several UNIX-oriented conferences and education programs annually. Articles he has written have appeared in Byte, CommUNIXations and Hard Copy. Mr. Zintz is also a member of ***UNIX REVIEW****'s Editorial Review Board.*

John Bass has worked as a UNIX systems engineer since Version 6 was first released in 1975. His technical assistance was key to getting both Onyx Systems and Fortune Systems off the ground. ■

Photos by Randy Becker

KEEPING UNIX IN ITS PLACE

An interview with Bob Marsh

Many factors have contributed to the birth of a personal UNIX market, but none has been more important than Onyx System's decision to introduce a UNIX-based micro in 1980. Bob Marsh, now chairman of Plexus Computers, made that decision.

Chances are another company would have done the job sooner or later. But Marsh's timing was critical. The success of the Onyx product showed not only that a UNIX micro port was technically feasible but commercially viable. The object lesson was not lost on OEMs, who were casting about at the time for alternatives to expensive minicomputer systems.

The rest, as they say, is history.

Bearing this in mind, UNIX REVIEW asked Dick Karpinski,

manager of UNIX services at UC San Francisco, to solicit Marsh's thoughts on UNIX's role in the PC marketplace. Karpinski succeeded in digging that up — and much more.

REVIEW: *What does low cost UNIX mean to you?*
MARSH: I've been thinking about that lately — particularly about where UNIX fits in the world, what AT&T ought to do, and how PC-class machines fit into the picture. Is UNIX even appropriate in the PC market? What is appropriate if UNIX isn't? I think I've pretty much come to the conclusion that UNIX isn't appropriate on PC class machines. MS-DOS is, though.
REVIEW: *Does it make sense to use UNIX on an IBM PC-AT?*
MARSH: As a small multiuser machine, yes. But the bottom line is that UNIX came out of the minicomputer world, where resources were scarce and expensive. The idea in that world was to share resources. That was fundamentally a different kind of environment — one designed for multiple users and teletype terminals. You can't be further away from what makes a PC successful than a teletype terminal. If you look at the sort of packages that are successful on PCs, the things that have made PCs explode in the marketplace, you'll find that they're video oriented, bitmap display oriented, keyboard oriented, and loaded with interactive tools.

UNIX is appropriate in multiuser environments with shared data and in instances where you're not particular about the graphic appeal of the user interface. UNIX is essentially an interface between the kernel and your applications. There's a lot of code in there for the file system and its performance. A lot of attention is paid to isolating users and UNIX processes from the hardware underneath. That's a lot of protection, which makes it a time-expensive system to get through. Getting back and forth through the protections and the system call interface costs you something. In addition, there is nothing in UNIX that defines an environment accommodating a window bitmap display and mouse.

You can't be further away from what makes a PC successful than a teletype terminal.

REVIEW: *There's not a system call library for graphics?*
MARSH: It's not really in the kernel at all. This is because the designers of UNIX wanted to provide an optimal system for multiple users using teletypes. But PCs are completely at the other end of the spectrum. In that environment, protection from other users just isn't an issue. You can turn UNIX into the kind of environment that would be appropriate on a PC. But by the time you did that, it wouldn't be UNIX anymore. It's the same argument that can be used against MS-DOS. MS-DOS is not going to become a multiuser system. If you tried to make it into one by adding what you need, it simply wouldn't be MS-DOS anymore.

UNIX is not a single-user system because you've got an RS232 pipe between you and the CPU. You can take a system like the Sun and throw enough hardware at it to get reasonable performance, but even the Sun system is not mind-blowing in terms of performance of office applications.

In fact, I just got back from a conference where we talked about office systems. One of the questions that was posed was: will MS-DOS or UNIX be the system for office systems? Basically, I don't think either one should be. Neither is ideal. In fact, something like the Mac operating system is a much, much better system for providing the productivity tools needed for office applications.

REVIEW: *What I'm hearing you say is that a bitmap display and a windowing system are fundamental tools for building a good user interface.*

MARSH: Certainly bitmap displays are. I'm not so sure that window presentation is optimal. I take exception to the way some of the Mac stuff is done. I like pop-up menus instead of pull-down menus, and I like mice with more than one button. But I like the attention paid to detail in the Mac.

But what I'm really after is an operating system that presumes a bitmap display, presumes a mouse, presumes a keyboard, and provides an interface that makes use of all those entities — while still providing the sort of distance you have under UNIX, so that the hardware can change underneath you.

REVIEW: *It used to be expensive to get 64K of memory, so we used to strive for systems that could run in 12K or even 6 or 2K. Well, now it's on the order of $150 to buy a megabyte of RAM chips. That's just a small fraction of the price of a machine of any capacity. A 32032 is roughly a $400 chip now, so even that doesn't represent a very big fraction either. Does this mean that the multiprogramming facilities of UNIX could be offered in personal-sized machines?*
MARSH: Sure. The multiuser capabilities and the file system and all that are absolutely justifiable on a very small machine — if you look downstream a year or two. Memory is going to be cheaper, the pro-

cessors will be faster, and the storage capacity and low cost peripherals will be there.

REVIEW: *You made reference earlier to the barrier between application code and kernel code. Doesn't that result in a 20 to 30 percent overhead cost?*

MARSH: The cost isn't as important as the style of the interface. Currently, there's a real need for some sort of window environment and some sort of bitmap display. High responsiveness, a mouse orientation and a standard way to interact with UNIX systems in a bitmap windowing environment are also needed.

REVIEW: *Is that a problem no one is addressing?*

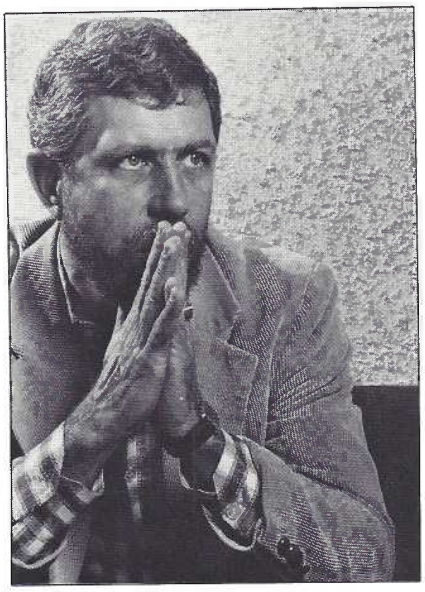
MARSH: AT&T might be. They talk about it. When I was at Cambridge a couple of weeks ago, they said that was one of the things they were going to be offering. But when? It's a fast moving market. If Microsoft came out and put that in Xenix right now, they could probably have an impact.

REVIEW: *Was Onyx the first UNIX vendor on micro hardware?*

MARSH: I think so. I signed the distribution license in November of 1979.

REVIEW: *Since then there have been a lot of trade shows, a lot of new machines, and you might say that the explosion has been almost entirely in micros and microprocessor-based systems.*

MARSH: There are other vendors now that have strategies based on proprietary processors that can offer UNIX on high end machines. That's okay. But the reason UNIX got established in 1979 in the first place was that you had 16K memories, 8-inch Winchesters, cartridge tapes and 16-bit processors just becoming available for the first time. Add to that the fact that the 8-bit world was playing itself out. People were trying to bank switch Z-80s, install hard disks and stretch out to 128K memories. Unfortunately, there was nothing to use this raft of new hardware with. The semiconductor people didn't provide any tools or high level languages, really.



I think AT&T is trying to displace MS-DOS on PCs. Its efforts are wasted.

They were still providing hardware emulators and Assembly language tools.

At the time we needed something that was going to take advantage of the new hardware and thus provide an alternative to minicomputers. But you didn't have any software available. None of the software vendors were designing for multiuser environments anyway. So UNIX just dropped into place. AT&T was not an obstacle. It wasn't particularly helpful either, but it wasn't an obstacle.

It's funny. When you consider what's appropriate and what might happen over the long term with hardware, UNIX makes sense — provided that some sort of window is utilized in the user interface.

But there's already a large installed base of PCs. People aren't going to replace those overnight. Neither are they going to throw out all of their MS-DOS software.

My sense of where UNIX fits in is that it shouldn't be running on the PC itself. What makes sense is to run it on another machine that the PC is connected to so that you can continue to run MS-DOS on the PC and thus make use of the applications you like, without sacrificing the functions the UNIX machine can perform. UNIX in this way can become an additional resource for the PC users.

I think AT&T is trying to displace MS-DOS on PCs. Its efforts are wasted. We might just as well settle for finding ways for coexisting with the PC, running PC-DOS on it plus something else — probably a windowing environment.

REVIEW: *Do you see UNIX being used strictly to serve networks of PC and Macintosh operators?*

MARSH: Sure. It all revolves around the same argument I made before. If you take MS-DOS and add to it all the things it needs to be an effective multiuser, multitasking environment, it's not MS-DOS anymore. Likewise, if you take enough out of UNIX to make it run fast and small and offer windowing capabilities, you don't have UNIX anymore. That's all there is to it. Joining the two systems allows each to do what it does best.

REVIEW: *So are you saying then that Lotus 1-2-3 and MacPaint are just the beginning of a broad*

range of user accessible software that can offer a better human/computer interface?

MARSH: UNIX certainly needs help in that area. I mean, the UNIX process structure, its hierarchy of files, its interprocess communications, and the simplicity of the kernel are nice ideas. But they're only part of the overall environment necessary for making the development of applications easy. And that after all is what it's really all about. Nobody buys an operating system, per se. And no one solves any problems with an operating system alone. It's just a tool that allows you to develop a certain class of applications well.

If I were working to develop the next generation of standards for PCs, I would demand software that from day one was a) independently offered; and b) designed for bitmap displays, pointing devices and keyboards. The closest you come to that right now is Smalltalk. I don't know that much about Smalltalk as a programming environment, but it is the kind of environment that presumes both graphics and multitasking.

REVIEW: *Doesn't Smalltalk also presume a large amount of processing power?*

MARSH: Yes, but so does UNIX. I could make the same argument about Smalltalk that we just made about UNIX in the sense that hardware costs are coming down rapidly, making Smalltalk on PCs feasible in the next couple of years. Any industry standard is an interface. It's a line drawn between a couple of domains. On one hand is a domain of applications and application development tools. Below that is a combination of resource management and abstracting mechanisms. This, in effect, takes some hardware technology and presents an abstract machine to the body of application development tools. Done properly, you can be isolated from the implementation, and you can also isolate the applications. UNIX presents one class of abstract machine, heavily oriented toward ASCII terminal devices, ASCII character devices, and presuming a disk drive, file system and multiple users.

REVIEW: *You've expressed some serious reservations about UNIX and yet you're chairman of a company selling hardware running UNIX. What can we expect to see out of Plexus?*

In the disk-intensive, multiuser, supermicro market, UNIX will be the dominant force for years to come.

MARSH: Everything we do is UNIX-based. It always has been and will continue to be. All the arguments I've just given you don't indicate the fundamental value UNIX has in the marketplace. UNIX offers a multiuser environment where vendor independence and technology independence are possible. UNIX will continue to reign supreme in this regard as far as I'm concerned. In the disk-intensive, multiuser, supermicro market, UNIX will be the dominant force for years to come.

REVIEW: *Isn't this specifically for folks who have already been into computing for the last 10 to 15 years — not the new wave of folks who have just gotten computers on their desks?*

MARSH: I don't think they're mutually exclusive. It is true that people buying UNIX micros for the most part today are people who have come out of the minicomputer world, because they appreciate all the characteristics of supermicros. In fact, they're the ones who are shaping this market.

But I don't think that necessarily excludes new users. My sense is that we're going to find ways to treat PCs as peripherals to the UNIX system, thus marrying those two sets of technologies and using each for what it's best at. Why try to replace MS-DOS? I mean, you're just swimming against the current.

REVIEW: *How are the folks who have PCs at home going to get access to UNIX systems?*

MARSH: I'm not sure they want connections.

REVIEW: *They want electronic mail. Some of them want NetNews.*

MARSH: You can do that over phone lines. I guess I've come to the point of view that UNIX won't be an important issue to people who have a computer at home. But there is a certain class of people — software people — that appreciate UNIX for what it is. Everybody else buys a computer to do something, like balance checkbooks, generate graphics or play games. In that environment, who cares whether UNIX is a part of it? In fact, it's a rather expensive toy, and it's not clear that it gives you enough functionality to pay for the overhead.

REVIEW: *What system does offer that functionality? PICK?*

MARSH: There are substantial advantages of UNIX over the PICK system — its general purpose and openness, for instance. The PICK system may be a highly optimal environment for certain limited purposes, but you can take that environment and put it on top of UNIX. Fundamentally you're still

dealing with disks and ASCII terminals. It's very hard to put something like a windowing system on top of UNIX — it's awkward and kludgey. Even when it's done efficiently, it's typically ad hoc. Putting the PICK system on top of UNIX, though, is not ad hoc at all. It costs you some performance, but in the meantime it also buys you the option of running a word processor or a communications package. UNIX, meanwhile, brings one very important thing to the table — it's already virtually an industry standard. It's not difficult to see why UNIX is important in the marketplace. The fact is that it's good enough to suffice.

REVIEW: *And there is already multivendor support for it.*

MARSH: That's right. That is what keeps attracting more application software to it, thus continuing the self-fulfilling prophecy. It is literally true that UNIX is available on lots of different machines from different vendors and that people can port software from PCs to the Cray. That's important. That's never happened before. That isn't to say you couldn't do it with Smalltalk. But nobody's done it so far.

UNIX looks pretty conventional to people who are developing compilers and the kinds of tools and applications people want. By comparison, an environment like the Smalltalk is pretty radical. You've got to get people to rethink on an object-oriented system. They have to alter their style and it's difficult. The fact that UNIX is pretty conventional, I think, is one of the reasons it is so acceptable in the commercial marketplace. People understand it.

REVIEW: *Speaking of standards in UNIX, there is an effort in /usr/group to create a "UNIX standard." How is that progressing, and what do you think its importance is?*

MARSH: Well, the standards have been published, voted on and accepted. The real question is: what's next? The Standards Committee needs to extend the document to incorporate System V extensions that didn't exist when work on the standards began. I know the committee is looking at interprocess calls and basic operating facilities for networking. They have a number of target areas — about 10 in all — that they're looking at.

My sense is that we're going to find ways to treat PCs as peripherals to the UNIX system.

REVIEW: *Are they looking at graphics interfaces?*

MARSH: I'm sure they are. But whenever you stumble across something new that has no momentum behind it, standards efforts can stagnate for a long time while technical people argue over the merits of one approach versus another.

When the standards effort first got going, everyone was behind it. Even the look-alike vendors wanted the world to settle down. System III seemed okay, so everybody was willing to put aside their technical arguments since there was already a lot of software using System III. But when you start venturing out into new territory, like graphics, I can't imagine a committee actually being able to pull that together.

What the committee has done effectively is abstract down from UNIX what could be standardized. Basically, they took out the things that were ambiguous and implementation dependent. In doing that, they came up with an abstraction of an implementation-independent, unambiguous machine.

REVIEW: *They sure didn't throw out much of System III.*

MARSH: The areas they had to throw out basically concerned terminal I/O. There were certain areas they could see they could never agree on, so they stayed away from those areas.

The question is, where do we go now? While the standards were being developed, AT&T introduced System V. AT&T is pushing that very hard now — along with an applications library based on System V. The next version of UNIX apparently will support the /usr/group form of record locking.

So, on one hand, AT&T says it's committed to being consistent with the standard, while on the other, it's pushing System V. My sense is that System V will ultimately be the standard. I also think, though, that the /usr/group Standards Committee will have some influence on what goes into System V and how it evolves.

REVIEW: *It would seem that UNIX*

lends itself to standards since it itself has been an effective standard for the industry.

MARSH: Exactly. Where people have intentionally made different versions, it disturbs customers. Actually, there are more differences in C compilers than in UNIX versions. And most of the differences are annoying. They're not the kinds of things that involve man-years of effort to get around, but they're annoying. So, in one sense, UNIX has been standard for a long time. If you look at only the system call interface over time, you'll see that it hasn't changed a whole lot. The implementations have changed a lot, and the utilities have changed a lot. But it doesn't make that much difference to a guy writing in the commercial world.

REVIEW: *Is there anything that's even close to UNIX in terms of acceptance in the multiuser marketplace?*

MARSH: The PICK system is as close as we've got.

REVIEW: *And that's at least an order of magnitude away.*

MARSH: In terms of acceptance, yes. Even IBM thinks UNIX is important. You don't hear those people talk about the PICK system, do you? There just isn't the same sort of vested interest in the PICK system. There are a lot of people committed to the UNIX system who have millions of dollars of venture capital and large bases of computers. That, in itself, is part of the self-fulfilling prophecy. The more people that commit to it, the more solid the commitment becomes. More and more new people come into the fold. You even have government contractors — like ITT, Bunker Ramo, CDC and all the Beltway OEMs in Washington DC — now learning how the UNIX system operates. They're all bidding UNIX-based systems to the federal government. It's the result of a movement that I think actually started in the 1980-82 timeframe.

A lot of people are still trying to understand it. The biggest obstacle to understanding the UNIX phenomenon is that so many people are still approaching it from a PC point of view — evaluating it in terms of sheer volume. I sat on a panel the other day with a fellow who was downplaying UNIX, saying there's only a hundred thousand installations. I said, "It's a billion dollar market, what do you mean, 'only a hundred thousand installations?' " He was talking from a PC perspective, where machine volume is the important issue, rather than dollar volume. But UNIX systems cost 10 to 20 times what the PCs cost, and you can generally count 10 to 20 times more users on them. You can't argue that the volumes in the UNIX marketplace aren't significant. People using UNIX to develop applications are serious commercial users who used to buy minicomputers to accomplish the same tasks they can do today on supermicros. There is no question in the minds of most OEMs that UNIX makes sense for them.

Standards efforts can stagnate for a long time while technical people argue.

REVIEW: *They selected it because it's a development environment that can be used to move applications out to real customers.*

MARSH: Yes. It's good enough as a development environment, but it's also a sufficient execution environment. And don't forget that it gives them vendor independence.

REVIEW: *As an execution environment, can UNIX be easily hidden?*

MARSH: Most of our customers don't use the shell. The shell isn't part of the interface to them because they use a database system like Unify or Informix or they use a programming language to create their applications.

REVIEW: *How does UNIX compare with other operating systems in terms of being hidable?*

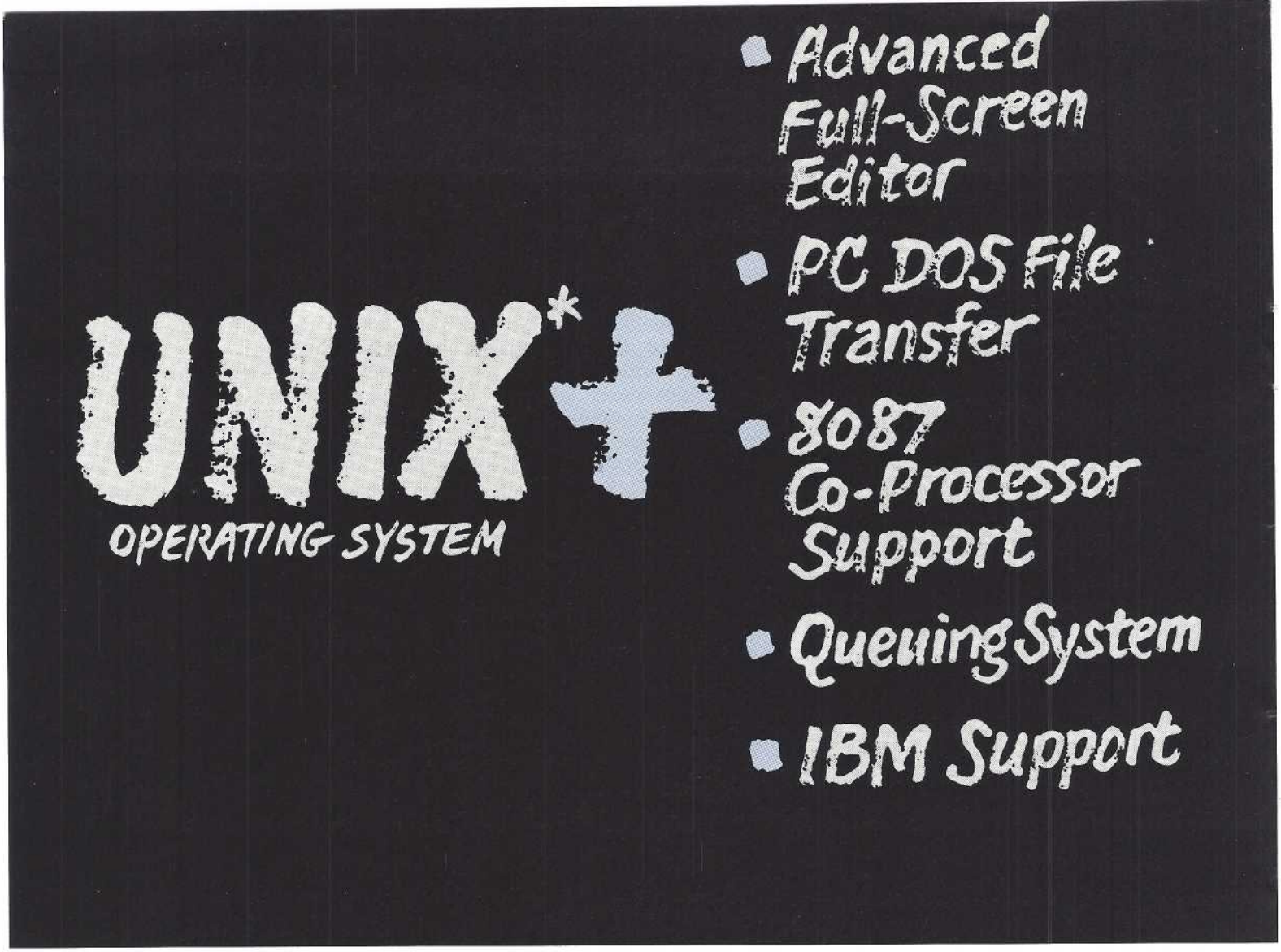
MARSH: Oh, it's eminently more hidable — and in a cleaner way, too. That's where UNIX shines, as far as I'm concerned.

REVIEW: *You mean in terms of getting out of the way?*

MARSH: Yes. The command interface in particular is so separate from the kernel that it's easy to just take it away and put a whole new look on it, provided you stay within the domain of ASCII character devices. It's also very easy to replace it altogether.

We designed a newspaper system in 1975 that was based on UNIX, originally Version 5 and then Version 6. It was a dual 11-70 failsafe system. The editors never knew UNIX lived underneath it. We went around the file system and created a fixed database environment. We scrapped the shell and all the utilities, and put our own front end on it. As far as the editors of the paper were concerned, this was just an editorial system. When they logged on, they got a different look than most UNIX users do. They could move around in desks and folders and edit copy — without ever knowing they were running on top of UNIX.

In putting that system together, we found UNIX to be a beautiful development environment and a good execution environment in the sense that we had the option of doing whatever



Now you can have a complete UNIX System III implementation on most models of the IBM PC.

The IBM Personal Computer Interactive Executive† (PC/IX) offers you tools like the C language, programmer's workbench, communication facilities, a text processing system and much more. It's also a multitasking system that offers you the same facilities found on larger UNIX operating systems.

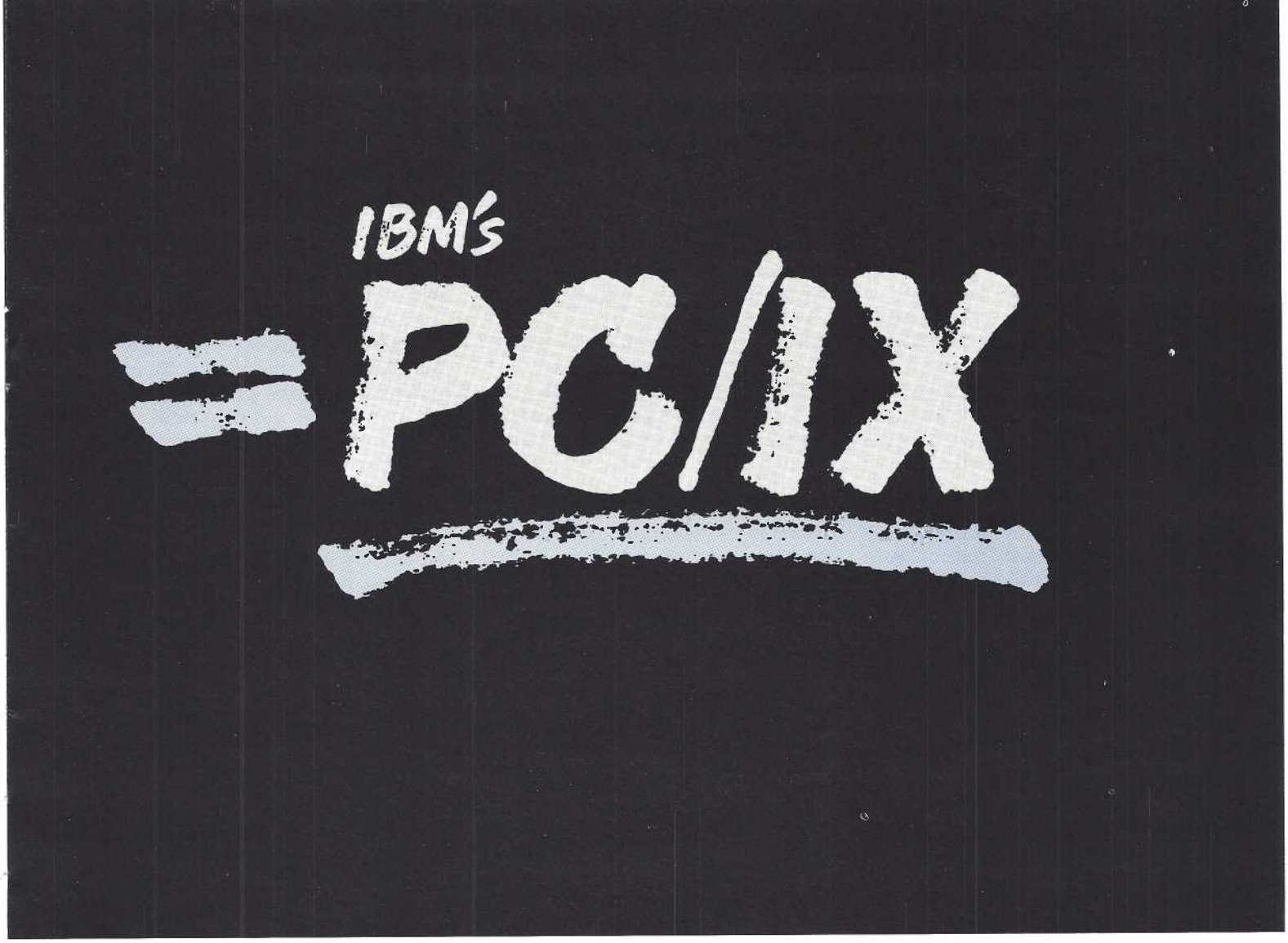
And PC/IX incorporates many significant enhancements. For example, a new full-screen editor helps you program more effectively. It offers such advanced features as windowing, function-key editing, the ability to execute commands from the editor, and automatic file backup. Beginning users of the editor can rely on a variety of "help" screens.

If you've been using PC DOS, you'll be glad to hear that it can co-reside with PC/IX. And that you can transfer files between the two systems.

PC/IX is specifically designed to take full advantage of the architecture of the IBM PC.

It transparently supports the 8087 Math Co-Processor. And PC/IX automates the management of input/output streams through a multipurpose queuing and spooling system.

IBM has other software for PC/IX



available, too, including INfort (a FORTRAN compiler with programming tools) and INmail/INnet/FTP (an electronic mail and file transfer facility)†.

And because PC/IX is from IBM, you get IBM's quality documentation, as well as IBM service and support.

PC/IX requires 256KB of memory on an IBM PC with fixed disk or an IBM PC XT, PC XT/370 or PC AT.

You'll find PC/IX at your nearest IBM Product Center or your Authorized IBM Personal Computer Dealer. To order, or for more information and the latest list of PC/IX application software, call IBM toll free at 1 800 IBM-2468, and ask for the PC Software Department, Ext. 393. Or call your IBM marketing representative.

IBM®

IBM Direct PC/IX 12-84
PC Software Dept. 3E/393
1 Culver Road, Dayton, NJ 08810

Please send me more information on PC/IX.

Name_____

Title_____

Company_____

Address_____

City_____ State_____ Zip_____

Phone_____

*UNIX is a trademark of AT&T Bell Laboratories. PC/IX is based on UNIX System III, which is licensed to IBM by AT&T Technologies, Inc.
†Developed for IBM by INTERACTIVE Systems Corp.

Circle No. 237 on Inquiry Card

we wanted. We made the editing system failsafe. We developed a high-performance, robust database for it and piped wire services in with a communications front end. All of that was easy in the context of the UNIX architecture.

Graphics, in the same way, can be incorporated fairly easily, but somebody has got to do it and make it standard before it'll have much impact.

REVIEW: *The first microbased UNIX system on the market was the Onyx machine with the Z-8000 in it. How long did it take you to do that port?*

MARSH: We had UNIX online three months after we had working chips.

REVIEW: *That's moving fast.*

MARSH: We did a lot of things fast. We didn't have much money.

REVIEW: *What have been the significant events in the world of microbased UNIX systems since then?*

MARSH: Dozens of new supermicro vendors started supplying UNIX systems, and so did all the major computer companies. I felt originally that we had an opportunity because the large companies were kept occupied by their major investments in proprietary systems. This whole idea of being standard and independent was given a bit of lip service on the compiler level, but not on the operating system level.

REVIEW: *Is the notion that buyers can avoid being locked in the key development of the last five years?*

MARSH: That's right. That's the thing that surprised us the most. When we first did the UNIX system at Onyx, we were looking for a multiuser system because that's what our customers wanted. From my standpoint, the easiest way to do that was with UNIX because it was inherently multiuser and all I had to add was record locking. I figured that would be good enough for the dealer environment, the retail environment — in essence, the whole micro world.

REVIEW: *And you were right about that, for the most part.*

MARSH: Yes. But what we underestimated was how great the demand would be in the more sophisticated world. The truth was that when we introduced it in the micro world, the micro world yawned — not the software developers but the dealers. They wanted applications. It took a long time to get applications out. So we were running two or three years ahead of our time in that domain. Unfortunately, the dealers made up our whole customer base at the time.

My sense is that System V will ultimately be the standard.

But when we introduced the UNIX micro, we found a whole different class of people showing up — big, big OEMs. They had all previously built minicomputer-based systems and were fed up with being locked into proprietary technology and not getting the kind of support they wanted. Basically, they had all been at the mercy of a single vendor, and they had each made a strategic decision never to do that again.

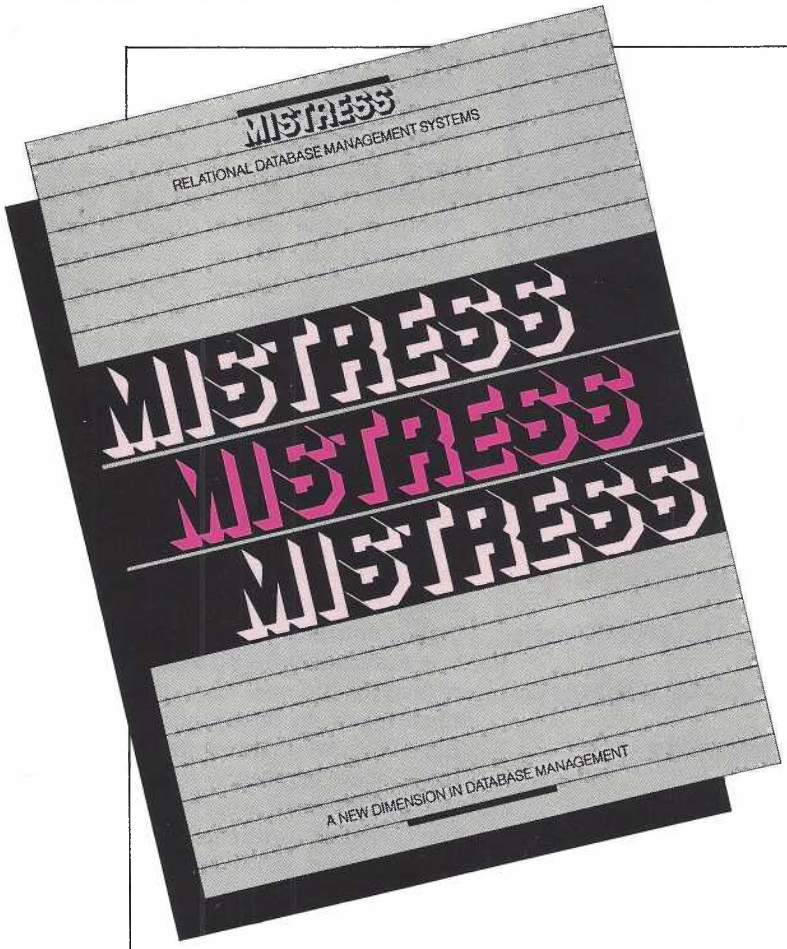
UNIX was a unique combination of circumstances at the time. AT&T was prohibited from playing in the marketplace, but it was required to license the technology, which had been designed 10 years earlier and so was really stable and portable. Either by accident or intention, AT&T had made it available, so lots of people knew something about it. UNIX was tailor-made for 16-bit and 32-bit machines, and the peripherals at the time were getting close to what you needed to be able to run it. Memory was getting cheap enough. So, the forward thinking OEM saw where this was all headed and strategically chose UNIX as the vehicle. Mind you, they knew early on that it wasn't perfect. But it was good enough, and the fact that it was standard, vendor independent and technology independent appealed to them.

Three years in this business is a lifetime. You can go through two generations of processors and a whole generation of peripheral storage devices. You need to be isolated from that. You need to be able to take advantage of what comes up. That is why UNIX is successful in this market — not because it's a nifty little operating system.

There's still another customer that we haven't talked about yet — the end users that act like OEMs. That is, the ones that purchase computers over a long period of time, and spread their investments over a lot of computers rather than make a major investment in one computer. Anybody who buys in volume, basically, has an inherent long term interest in standards and the flexibility they afford. But one of the characteristics of an OEM-like market is that it doesn't explode like a PC market.

The PC filled a vacuum. There was no equivalent product before, and it didn't really threaten too much of any manufacturer's existing business, so it was easy to play by some new rules. UNIX micros, by comparison, were directly competitive with minis and superminis, and so they represented a threat to the bread

A NEW DIMENSION IN DATABASE MANAGEMENT



MISTRESS is the fully relational database management system (RDBMS) for UNIX.* It features the Structured Query Language (SQL*) for the end user as well as standard programming interfaces to the C language for the DP professional. Advanced concepts include variable-length character fields, dynamic storage allocation, and B+ Tree indexing. MISTRESS has been designed exclusively for the UNIX environment and is totally written in C.

MISTRESS/32 is the advanced relational database management system for extended addressing UNIX products. MISTRESS/32 features enhanced capabilities for security, recovery and data integrity, as well as a fully integrated report writer and screen interface. MISTRESS/32 is the recommended system for more demanding applications.

*UNIX is a trademark of Bell Labs. IBM and SQL are trademarks of International Business Machines.

See us at UniForum Booth #1070

RHODNIUS Incorporated

10 St. Mary Street, Toronto, Ontario, Canada M4Y 1P9 (416) 922-1743 Telex: 06-986766 TOR.

and butter business of DEC, Data General, Hewlett-Packard and IBM. That's one aspect. Another is that the OEMs, if they're successful, make an investment and then spend a few years getting some return out of it. They can't afford to keep changing the technology or rewriting their applications, so there are only certain points in an OEM's life cycle where the decision to change can be made. In the course of a given year, not all of them will choose to do much different, but a third of them will look at it. Some percentage of that third will decide to change.

If you're an OEM at that point, what alternative do you really see out there? What's your safest decision? UNIX. If you're a minicomputer OEM, that's obvious. But OEMs that look at UNIX through a microscope can get nervous because UNIX doesn't have the concept of records, nor does it offer record locking, good error messages, power fail, restart, and a lot of things that OEMs have come to take for granted. But they've got to get over those technical hurdles. It's a question of how much emphasis they put on technology versus independence. My argument is that independence is a large factor, and that the technical blemishes of UNIX get washed out when people look at the alternatives. The strategic value of UNIX is much more important.

REVIEW: *What fraction of the sales of these boxes are going to end users versus large companies or OEMs, which are really a different breed of purchaser?*

MARSH: Well, I'll tell you what our numbers are: 70 to 80 percent of our customers are OEMs. The balance are end users. For the most part, they are technical end users.

REVIEW: *You've said a lot about the advantages of UNIX for the OEM, but why would end users be interested?*

MARSH: When you buy a computer, you want to be able to do as much with it as possible. A vendor-independent operating system, by definition, is going to provide you with access to lots more packages over time than a dedicated system. Thus, it's more valuable than a system that doesn't have that kind of open-endedness.

The more people that commit to it, the more solid the commitment becomes.

We've seen the advantages of standards elsewhere. It's happened in retail consumer environments. Sony and Phillips and others got together to define the compact disk format standard. The consumer electronics people have figured out that it's in their best interest to arrive at standards and then compete in the open marketplace. The commercial side of the systems business hasn't figured that out yet. But the semiconductor people have. They know that without second sources, it's very hard to get people to use their chips.

REVIEW: *So a merchant market is being formed?*

MARSH: It has a lot of the characteristics of a merchant market. Customers are demanding it now because it's possible to demand it. It used to be that a customer would listen to a minicomputer salesman give three months worth of presentations, and then go out and compare that information to the alternatives. Once he started writing code, though, it was very painful to change.

Today, the customer does a lot more independent market research. He may even do some benchmarks, or read UNIX REVIEW and UNIX/World to sort it out for himself. No longer does he have to make a long-term commitment. The customer can begin his development, and get all the support he would get from a minicomputer company, by buying just one machine with no volume commitment. He can then spend six months or so developing his application. Meantime, there might be 10 new announcements. Five companies form and three fold. By the time the application is ready, he's learned what his vendor is like. In the meantime, the competitor's salesmen have been calling. So maybe he moves code to those machines to see how hard it is. Then, when it's time to make a decision, he calls all the vendors together and runs a demo of the application package on each machine. After that, he's ready to talk contract terms, and he knows, as customer, he's in the driver's seat. He didn't have that kind of option before. Vendors in this market thus cannot stop sell-

ing or stop supporting. They can't stay behind the technology. It's suddenly become a very demanding environment.

REVIEW: *When you started putting UNIX on micros, a lot of this wasn't clear. What impressions did you have at that time?*

MARSH: I was a DEC OEM running UNIX applications. I had been an end user and I'd been jerked around by both IBM and DEC. I knew what it felt like. UNIX made so much sense from so many standpoints that I was willing to look past the technical shortcomings. If I was willing to do that, I figured a lot of other people would be too. The speed at which the market developed blew my mind. We were talking to minicomputer people who understood minicomputer architectures and wanted minicomputer class performance. So that's what we built. Architecturally, people recognized it. It looked like what they were used to, it performed like what they were used to. What's more, it gave them a tremendous price advantage.

I'm wondering now how to take these underlying principles and reapply them to the same market to come up with a successor. The successor we talked about earlier was Smalltalk. My sense is that there is an opportunity for another UNIX phenomenon in this market. It might come from Microsoft, but it probably won't. They have a big MS-DOS installed base, and it's hard for them to shift. But there is an opportunity for another set of tools to take off.

REVIEW: *You indicated earlier that if somebody were to develop a system that could be easily ported to various microprocessors and could provide access to graphics screens and pointing devices, it would develop a new marketplace. But would that wipe this one out?*

MARSH: No. This market is going to be around for a long time.

REVIEW: *Is this a marketplace of niches?*

MARSH: No. In fact, that's a problem of the UNIX market; it's too broad. The market has matured so quickly that you can no longer distinguish yourself by saying you supply UNIX systems. So what? Everybody else does, too.

REVIEW: *But the OEMs still have markets of niches?*

MARSH: Yes, basically people are doing database-intensive multiuser applications. Actually UNIX is being used in too many different ways. MassComp is doing real time data acquisition. Sun is generating bitmap displays for technical applications. These are completely different markets with completely different sets of requirements. AT&T has people from all different segments telling it what to do with UNIX. PC people want multitasking, They also want it smaller and cheaper, and they want windowing, graphics, MS-DOS compatibility, networking, object code standards, media standards, retail image and PC ports. Basically they could care less about the 68000 and National's 32016. They want 8086 family ports. On the other end of the spectrum is a big academic community that wants UNIX put in the public domain to let people hack on it.

You can't argue that the volumes in the UNIX marketplace aren't significant.

There are also people who would be happy if UNIX went away, or were limited to PCs. These people, typically, have a bread and butter product that they feel UNIX threatens.

So AT&T is pulled in all these different directions. Nobody makes money on operating systems — certainly not on a scale that makes sense to AT&T. I think they ought to drop the price of the kernel, unbundle all the utilities, and compete in the utilities arena. You can charge twice as much for the utilities as you can for the kernel itself, and in the process you can eliminate a lot of hassle.

REVIEW: *Do end users care about which operating system drives these applications?*

MARSH: No. They don't care at all. Actually my feeling is that AT&T should stop advertising operating systems in end user trade magazines. The focus should be on applications. AT&T should set up an authorized distributor program giving other vendors co-op advertising and some incentives to talk OEMs into moving to the

Continued to Page 76

/usr/lib

The good book

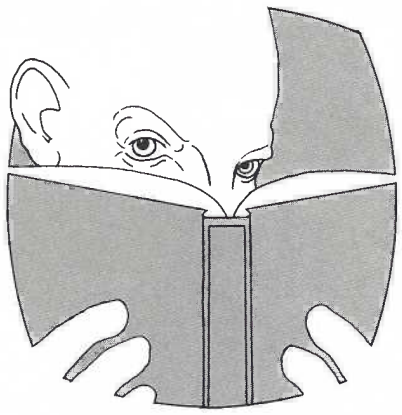
by Jim Joyce

This column is about a book that should be read and reread by everyone interested in UNIX. Given the season, winter days may provide just the proper opportunity.

The book? Why, the *UNIX Programmer's Manual*, of course — both in its familiar, Version 7 format and its repackaged System V format.

John Mashey, in his "Small is Beautiful" talk, reminisces about the early days at Bell Labs, where the practice was to reread the *UNIX Programmer's Manual* once a month. That, of course, was when it was less than an inch thick. Now the manual, at minimum, comes in two inch-thick volumes (from CBS Educational Publishers) or in nine volumes of varying sizes (from AT&T Bell Laboratories). People at Bell Labs reread the manual once a month to reacquaint themselves with commands, options and examples forgotten in the press of daily work.

Rather than discuss the various flavors of the manual, my comments will be based on the Version 7 two-volume iteration.



Generally, most users use a subset of the commands, learning them well but forgetting about the others. More pointedly, most users use a subset of a particular command's options — learning them well, but forgetting about the other options — ones that may help them in new situations.

Section 1 of Volume I contains the basic set of user commands. The terseness of the writeups has been a source of vexation for many a reader. Although I certainly do not defend the manual pages for their lack of examples and obtuse prose, it is true that most often the information *is* there for the reader who is patient and attentive.

An often-neglected part of the manual is the papers in Volume II. The papers on **tbl** and **nroff** contain formatting gems. Programmers should read Steve Bourne's writeup on the shell, as well as the paper on **awk**.

All of this is not to say that the manual could not be better. My approach to the manual is to treat it as a collection of intra-office memos among like-thinking individuals. The command pages, especially, show this in-crowd attitude: "Very long lines are silently truncated" reads the message in the **sort** writeup, with the clear assumption that anyone wanting to know what a "very long line" is will look at the source code. Irritating, yes. But rarely dull.

Jim Joyce is President of International Technical Seminars, Inc., a firm committed to UNIX training, and founder of the Independent UNIX Bookstore. For answers to your questions about books, call 415/621-1593. ■

THE INDEPENDENT UNIX* BOOKSTORE

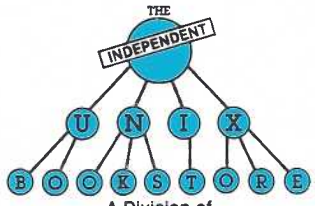
OVER 60 UNIX AND C ITEMS IN STOCK INCLUDING

- UNIX BOOKS
  - The UNIX System by Stephen R. Bourne (Addison-Wesley)
  - Operating System Design: The XINU Approach by Douglas Comer (Prentice-Hall)
  - The UNIX Programming Environment by Brian Kernighan and Rob Pike (Prentice-Hall)
  - A Practical Guide to the UNIX System by Mark G. Sobell (Benjamin/Cummings)
- C BOOKS
  - The C Puzzle Book by Alan R. Feuer (Prentice-Hall)
  - The C Programming Language by Brian W. Kernighan and Dennis M. Ritchie (Prentice-Hall)
  - Learning to Program in C by Thomas Plum (Plum-Hall)
  - C Programming Guide by Jack Purdum (Que Corp.)
- C and vi REFERENCE CARDS
- PERIODICALS
  - UNIQUE
  - UNIX Review
  - World UNIX & C
- vi POSTER, UNIX SHELL POSTER
- T-SHIRTS
  - "UNIX is a Trademark of Bell Laboratories"
  - "-rwxrwxrwx"
  - "grep for it"
  - "awk: bailing out near line 1"
- NEW ITEM
  - UNIX Coffee Mug
  - "UNIX" in blue on outside
  - "is a Trademark of Bell Laboratories" inside lip imprint
  - Porcelain mug, kiln-fired permanent color

Call or write for a complete catalog

Mail and phone orders only.
We ship anywhere in the U.S.
Contact us for shipping elsewhere.

* UNIX is a Trademark of Bell Laboratories



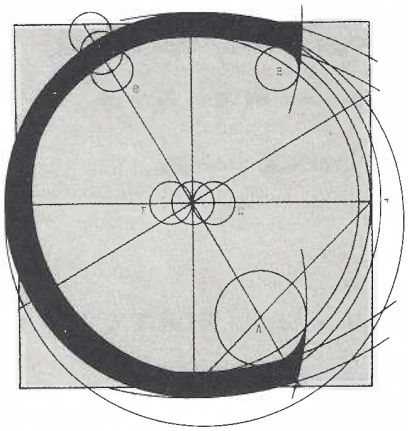
A Division of
International Technical Seminars
520 Waller Street
San Francisco, CA 94117
(415) 621-1593

Circle No. 283 on Inquiry Card

C ADVISOR

Pipe programming

by Bill Tuthill



One major reason for the success of UNIX, aside from portability, is the pipe. UNIX was not the first operating system to make use of pipes, but it certainly broke ground by offering a clean and easy-to-use implementation of the idea. A pipe is an I/O channel between two related processes that allows data to move in a single direction only; one process writes while the other reads. The processes can be related as parent and child, or as two children of the same parent.

Pipes permit programs to pass data to other programs, without first routing it through a temporary file. Much software built on large operating systems requires huge amounts of disk capacity for storing temporary files. Pipes require no disk space, and they also speed processing since the second program can start working before the first program has finished. In this way, pipes help reduce the cost of computing.

This month's column shows how to use pipes in C. The first sample program reads input from one process, and sends output to another. The second sample program writes to and reads from the same process, which is somewhat more difficult. For more information on pipes, read the article by Kernighan and Ritchie, *UNIX Programming-Second Edition*, which first appeared in the Version 7 manual.

LIBRARY ROUTINES

The Standard I/O Library makes available the **popen( )** routine, similar to **fopen( )**, but used for opening pipes instead of files. Pipes may be established for reading or writing. Instead of specifying a filename, as you would for **fopen( )**, you simply give the name of a program. Figure 1 shows a program that opens a pipe to the **who** command for reading, and a pipe to the **sort** command for writing. It should be noted that this sample does not show the most efficient means of accomplishing the task. It is also possible to pipe between two child processes, but this necessitates programming at the system call level. More about this later.

The **popen( )** routine consults the PATH environment variable to find the command specified in the first argument. In most implementations, **popen( )** invokes a shell to do this. The second argument may be either "**r**" for reading, or "**w**" for writing. Note how **popen( )** returns a FILE pointer, as does **fopen( )**. We have two such pointers here, one for input and the other for output.

While reading input with **fgets( )** and writing output with **fputs( )**, the number of users is counted, so we can supply this number later on. Just as you should close files with **fclose( )**, you also should close pipes with **pclose( )** when you're finished. Otherwise the count of users might appear in the middle of user names. Not only that, each open pipe counts as an open file, counted against the system limit of 20 simultaneously open files. After closing the pipes, the sample program first prints the number of users, then the date using the **system( )** function, and finally exits.

SYSTEM CALLS

The two library routines for establishing pipes, **popen( )** and **pclose( )**, will not suffice for writing to and reading from the same process. For example, creating two pipes to the **sort** program, one for reading and one for writing, would result in two

```
#include <stdio.h>

main()          /* nwho - who program, sorted, with # of users */
{
        FILE    *ifp, *ofp, *popen();
        char    str[BUFSIZ];
        int     users = 0;

        if ((ifp = popen("who", "r")) == NULL) {
                perror("who");
                exit(1);
        }
        if ((ofp = popen("sort", "w")) == NULL) {
                perror("sort");
                exit(1);
        }
        while (fgets(str, BUFSIZ, ifp)) {
                fputs(str, ofp);
                users++;
        }
        pclose(ifp);
        pclose(ofp);
        printf("%d users at ", users);
        system("date");
        exit(0);
}
```

Figure 1 — *An illustration of piping using the* ***popen( )*** *routine.*

separate **sort** processes, which is not what you want. You need to use the low-level system call **pipe( )**, along with **fork( )** and **dup( )**. The idea is to set up two cooperating processes with two pipes between them. You make each pipe one-way by closing opposite directions for each process. Figure 2 diagrams how this is done.

Suppose we were writing a visually-oriented desk calculator, using **curses**, say, or better yet, a bitmap display terminal with a mouse. We would paint a calculator keyboard on the screen, and then allow the user to select appropriate keys by clicking the mouse. The program would build up arithmetic expressions and send them through a pipe to **bc** when they were complete. The sample program in Figure 3 does not handle the bitmap display or the mouse, but it does demonstrate how to set up a two-way pipe to **bc**. After creating its plumbing, the program reads a line from standard input, writes the line through a pipe to **bc**, reads the answer back from a second pipe and then writes the answer to standard output.

We use **fdopen( )** because we want to obtain a FILE structure, rather than a simple file descriptor. This allows us to use Standard I/O Library routines in **main( )**. Note how the parent and child processes are, in many ways, mirror images of each other. When the parent closes for reading, the child closes

One major reason for the success of UNIX, aside from portability, is the pipe.

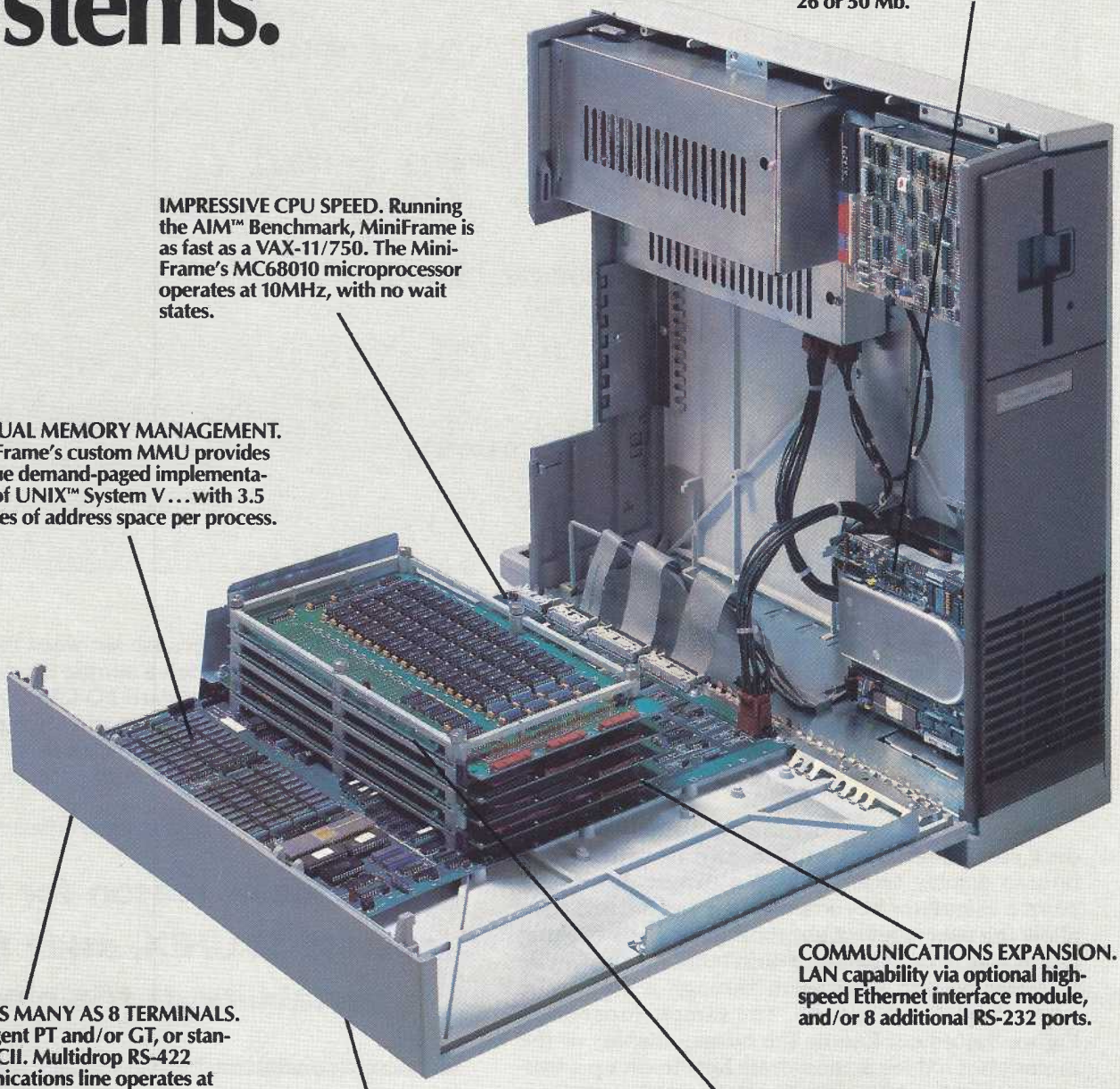
for writing, and vice-versa. When the parent opens for writing, the child opens for reading, and vice-versa. We call **setbuf( )** so the pipe will be unbuffered, and so transactions between our program and

An open-and-shut case for making MiniFrame™ your first choice in low-cost OEM systems.

ONE TO EIGHT USERS. MiniFrame can be configured for up to 50 Megabytes of integral mass storage: 5¼" Winchester fixed disks of 13, 26 or 50 Mb.

IMPRESSIVE CPU SPEED. Running the AIM™ Benchmark, MiniFrame is as fast as a VAX-11/750. The MiniFrame's MC68010 microprocessor operates at 10MHz, with no wait states.

VIRTUAL MEMORY MANAGEMENT. MiniFrame's custom MMU provides unique demand-paged implementation of UNIX™ System V . . . with 3.5 Mbytes of address space per process.



RUNS AS MANY AS 8 TERMINALS. Convergent PT and/or GT, or standard ASCII. Multidrop RS-422 communications line operates at 307 Kbit/sec.

COMMUNICATIONS EXPANSION. LAN capability via optional high-speed Ethernet interface module, and/or 8 additional RS-232 ports.

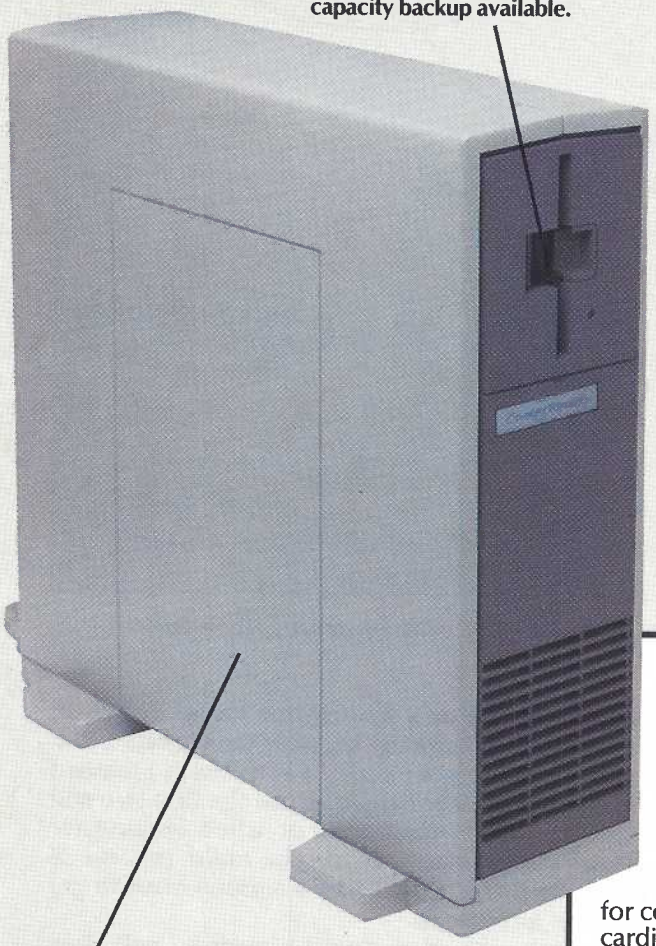
ECONOMICAL MEMORY EXPANSION. MiniFrame provides ½ Megabyte of RAM standard; up to three boards can be added for system expansion to 2 Mb.

FOR HIGH-SPEED PRINTERS, a Centronics-compatible parallel printer port is standard in the MiniFrame processor.

MiniFrame: another first from Convergent Technologies. Super-minicomputer power at PC prices.

OEMs can now meet the needs of small to medium-sized organizations for low-cost, high-performance systems capable of handling large UNIX-based applications.

The MiniFrame system—which effectively opens a whole new market for you—is unique not just for its price/performance advantages. It also provides complete flexibility in meeting requirements ranging from single-user "personal UNIX systems" to eight-terminal data-processing installations.



BACKUP STORAGE with integral 5¼", 640 Kbyte floppy disk. Optional high-capacity backup available.

Convergent Technologies achieved this breakthrough by adapting the innovative hardware and software developed for its MegaFrame multiprocessor.

The result is a powerful, compact and expandable unit priced to give you the maximum opportunity of capitalizing on a rapidly growing UNIX market.

There are, of course, many other unique benefits of the MiniFrame system.

Convergent provides foundation software for office applications—including a powerful, Wang keystroke-compatible word processor, an advanced financial spreadsheet and complete electronic mail facility.

The Window Manager permits viewing and manipulating of up to four applications running simultaneously on Convergent's PT or GT terminal screens.

Major performance increases result from utilizing these terminals with the MiniFrame. At only slightly higher cost than standard "dumb tubes", our PT or GT terminals offer high-speed communications plus built-in processor with enough memory to execute key portions of the system code.

Programming languages include industry-standard High Level COBOL and BASIC, full FORTRAN-77, Pascal and C.

OEM prices for the MiniFrame start at less than $5,000; an eight-user MiniFrame can be configured for under $10,000. Prices like these make it an open-and-shut case for choosing the MiniFrame.

WE'VE CLOSED OUR CASE... now send for a complete information package. Write to Convergent Technologies, Data Systems Division, 3055 Patrick Henry Drive, Santa Clara, CA 95050. Phone: 408/980-0850. Telex: 176-825.

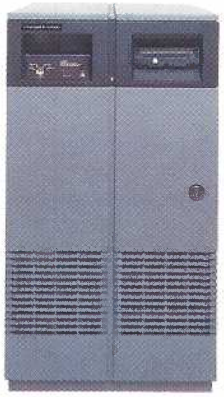
Convergent Technologies

Where great ideas come together

Circle No. 282 on Inquiry Card

Graceful upgrade path to the MegaFrame.™

The MegaFrame, Convergent Technologies' revolutionary super-minicomputer system, utilizes multiple processors; has expansion potential to 128 users, 8 MIPS and gigabytes of disk storage. It enables OEMs to handle today's growing demand for computing services without discarding a single piece of hardware ... or being forced into expensive CPU upgrades.



Applications software can be transported—unchanged—from the MiniFrame to the MegaFrame whenever the workload requires it. The two systems are object-code compatible, allowing OEMs to provide a complete family of systems.

MegaFrame: proof that if anyone can build a UNIX system the way it should be built—it's Convergent Technologies.

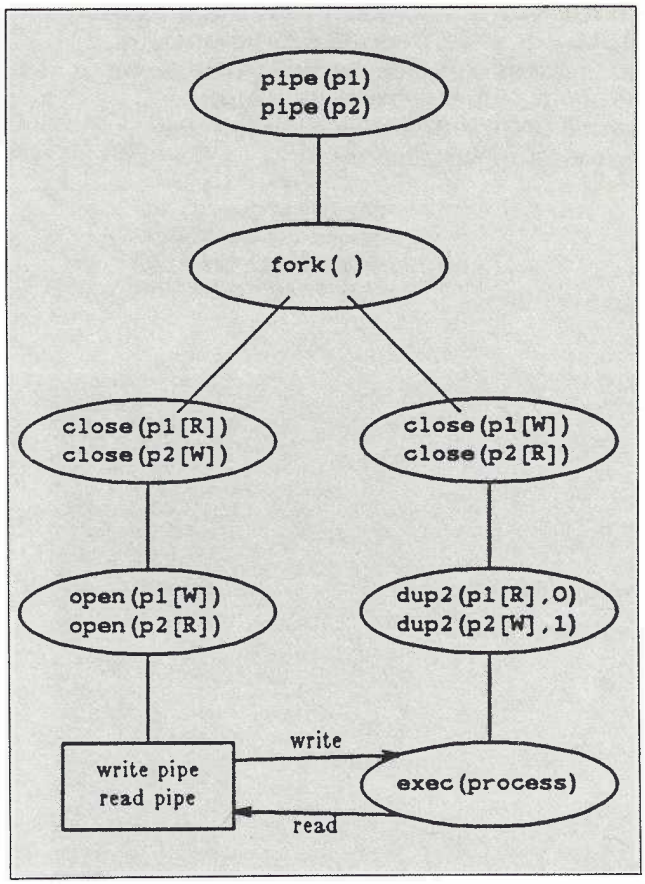


Figure 2 — *Piping with low-level system calls.*

bc will take place on a line-by-line basis. The child process will be overlaid by **bc**, after the **execlp( )** call. It's wise to use either **execlp( )** or **execvp( )** instead of other kinds of **exec** calls because the first two will consult the PATH environment when looking for a program to execute. Since the child process is replaced by **prog**, it would be a serious error to get to the line afterwards.

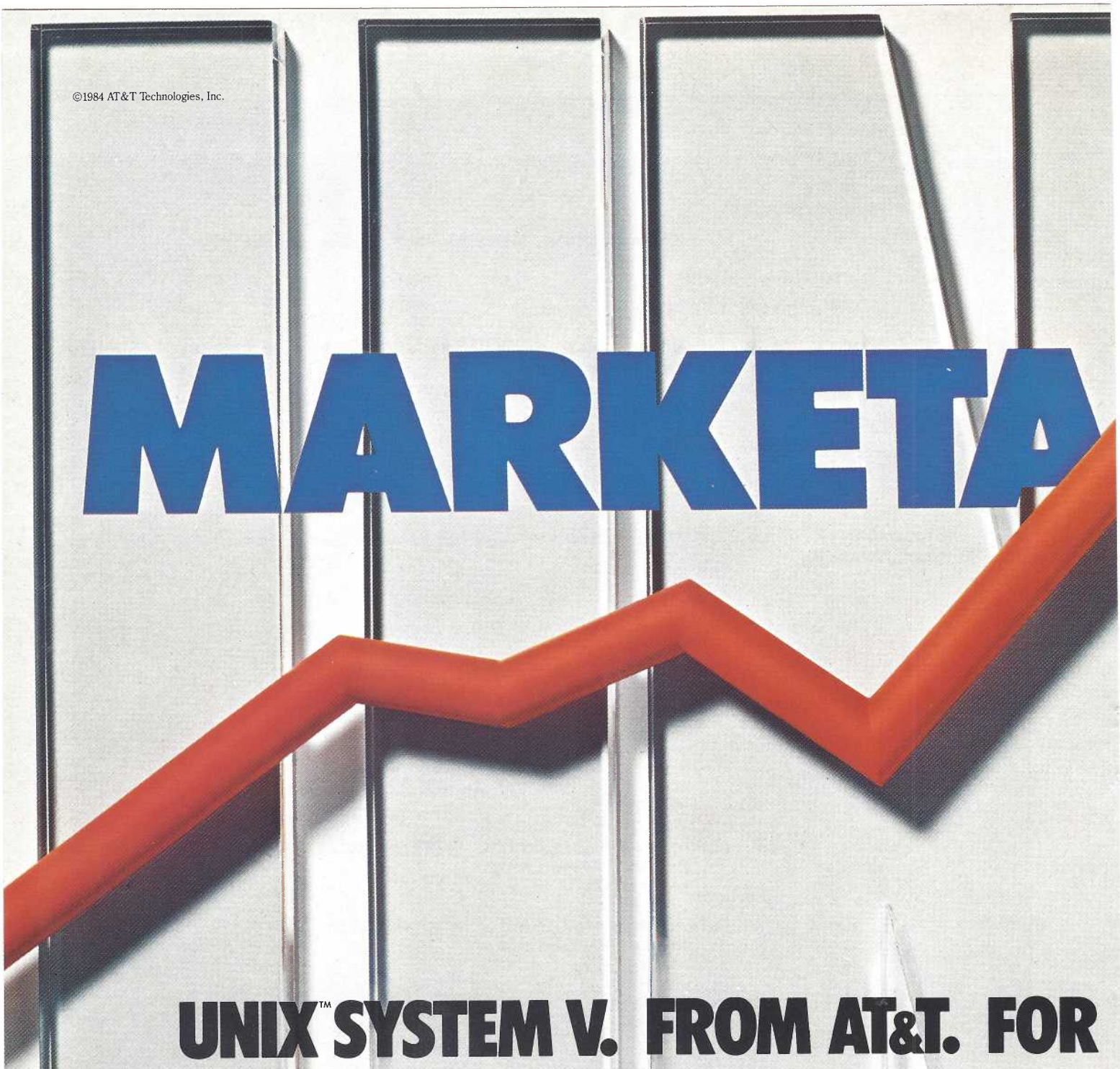
Even though the **dup2( )** system call was available in Version 7, it is strangely absent from System V. There, you have to make this haphazard collection of calls instead:

```
close(0);                    /* stdin */
dup(p1[READ]);
close(p1[READ]);
close(1);                    /* stdout */
dup(p2[WRITE]);
close(p2[WRITE]);
```

Of course, we would want to add error checking, as in the sample program in Figure 3. It is better to

```
#include <stdio.h>
#include <signal.h>
#define READ    0
#define WRITE   1
FILE *fpW, *fpR;

main()                  /* read stdin, send to "bc", write stdout */
{
        char line[BUFSIZ];

        if (pipeto("bc") < 0)
                exit(1);
        while (fgets(line, sizeof(line), stdin)) {
                fputs(line, fpW);
                fgets(line, sizeof(line), fpR);
                fputs(line, stdout);
        }
        fclose(fpW);
        fclose(fpR);
        exit(0);
}

pipeto(prog)            /* open two way pipe to process named "prog" */
char *prog;
{
        int p1[2], p2[2], pid;

        if (pipe(p1) < 0 || pipe(p2) < 0 ) {
                perror("pipe");
                return(-1);
        }
        if ((pid = fork()) < 0) {
                perror("fork");
                return(-1);
        }
        if (pid > 0) {                  /* parent */
                close(p1[READ]);
                close(p2[WRITE]);
                if ((fpW = fdopen(p1[WRITE], "w")) == NULL) {
                        perror("fdopen w");
                        return(-1);
                }
                if ((fpR = fdopen(p2[READ], "r")) == NULL) {
                        perror("fdopen r");
                        return(-1);
                }
                setbuf(fpW, NULL);              /* unbuffered */
                setbuf(fpR, NULL);
        } else {                        /* child */
                close(p1[WRITE]);
                close(p2[READ]);
                if (dup2(p1[READ], 0) < 0) {    /* stdin */
                        perror("dup2 r");
                        return(-1);
                }
                if (dup2(p2[WRITE], 1) < 0) {   /* stdout */
                        perror("dup2 w");
                        return(-1);
                }
                execlp(prog, prog, NULL);
                perror("execlp");               /* shouldn't get here */
                return(-1);
        }
        return(0);
}
```

Figure 3 — *An illustration of two-way pipes.*



MARKETA

UNIX™ SYSTEM V. FROM AT&T. FOR

More than 100 major computer manufacturers and resellers are licensed for UNIX System V. And each one is out to stake a claim in the growing multi-user market. They're looking to you for applications.

It's another reason why good business decisions are based on UNIX System V.

Now is the time to start writing software for UNIX System V from AT&T. Many of your customers are already asking for programs. As more and more companies turn to UNIX System-based machines to solve their computing needs, the market for applications will explode. In addition, the sooner you act, the better able you'll be to cash in on the industry-wide trend to UNIX System V from AT&T.

You'll find a receptive market. It's a golden opportunity to get in on a business with a bright and profitable future.

The profits of portability

Because applications software written for UNIX System V is essentially portable, you'll be able to sell your packages for a wide variety of machines—without expensive recoding. You'll also be able to develop applications faster, thanks to the powerful software development tools built into UNIX System V.

That means hitting your market window and taking advantage of a major source of new business.

AT&T offers a number of programs designed to help you develop and sell the most marketable software for micros and

SYSTEM V

BILITY

SOFTWARE THAT SELLS.

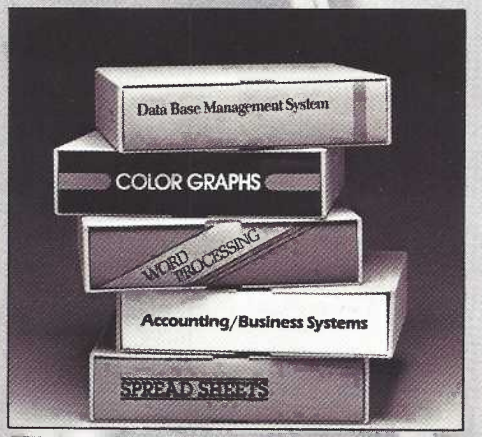
minicomputers. Our Independent Software Vendor Support Program offers training, evaluation, technical assistance, even AT&T 3B Computers to get you started.

UNIX System V. From AT&T. From now on, consider it standard.

Support from the source

The market for applications for UNIX System V from AT&T is heating up. And with our complete support program, entry has never been easier. For more information on the profits of writing software for UNIX System V, send in the coupon.

AT&T



Please send the free booklet on the AT&T Independent Software Vendor Support Program
Mail to: AT&T, P.O. Box 967, Madison Sq. Station
New York, N.Y. 10159 AT&T UR-1200-DE

Name_____

Title_____

Company_____

Address_____

City_____ State_____ Zip_____

My business category (check one):
☐ OEM/VAR ☐ Software House
☐ MIS/DP Mgt. ☐ General Business Mgt.

Other_____

UNIX System Licensee ☐ Yes ☐ No
☐ Don't know

use **dup2( )** if possible, because it guarantees that you are closing and duplicating the correct file descriptors.

EXERCISES

The desk calculator **bc** works on a line-by-line basis. That is, we send a line and it sends a line back.

Pipes require no disk space, and they also speed processing.

If we were establishing a two-way pipe to another program, such as **tbl**, input would not correspond line-for-line with output. In fact, we could not be sure that we would get the same number of lines as we sent. Consequently, we would have to make arrangements for reading until there was nothing left to read.

The first example offered in this column (Figure 1) is not as efficient as it might be. The parent process reads from one child (**who**), and writes to another (**sort**). It would be better if the first child wrote directly to the second. This could be done by extending **pipeto** so that it would do another **fork( )** in the parent process (**if pid**), and set up the pipe between that child and the older child. If you work through this exercise, replace the two calls to **fdopen( )** with **open( )**.

Bill Tuthill is a member of the technical staff at Sun Microsystems. He was formerly a leading UNIX and C consultant at UC Berkeley, where he contributed software to BSD 4.2. ■

RULES OF THE GAME

Babel revisited

by Glenn Groenewold

Why does it have to be so doggone complicated? And why can't the United States cover an area such as computer law with a uniform set of rules?

Who hasn't had similar thoughts when confronted with incomprehensible or seemingly inconsistent legal requirements?

The American legal system *is* complicated, no denying that, and computer law often demonstrates this in spades. But easy as it is to blame this state of affairs on some conspiracy of lawyers, the legal profession in reality has expended considerable energy in attempts to simplify the law. To understand why these efforts seem to have had so little result, we must consider where law in general, and computer law in particular, comes from.

In the beginning, there was the common law.

Common law is a collection of rules for the regulation of personal and business conduct which evolved gradually in England over several centuries. At first, it consisted of nothing more than the shared experiences of magistrates who toured the country settling disputes. When these officials met up with one another, over a tankard of mead or two, they'd talk shop — what else? — exchanging stories of interesting controversies and the ways they had been resolved. Later on, when

one of these magistrates encountered a similar case, he would be likely to draw on one of these precedents in making his decision.

From this beginning, the notion took hold that controversies involving comparable facts ought to be decided in the same way that they were previously. This is the legal principle called *stare decisis*, which simply means that a court may be obliged to follow an earlier decision when it decides a case.

As the common law system continued to evolve, uniformity was no longer dependent on the swapping of stories: decisions came to be written, and ultimately, printed. The time arrived where people buried their noses in books to find the appropriate precedent governing the determination of a dispute.

In America, the English common law formed the basis for the law of each of the 13 original colonies. Following independence, most of the newly-admitted states also adopted the common law, and even those that did not generally patterned their laws on it.

It's tempting to shrug this off as quaint history that has little to do with databases or object code, but our legal concepts of property rights and the conduct of fair business largely come out of the common law and continue to be influenced by it. Whenever a new situation develops in our perennially restless, innovative society, the common law comprises a ready-made body of general legal principles that can be applied, if nothing else, as a stopgap until more specific laws can be devised.

So it was when the computer came upon the scene. A distinguished jurist such as Sir Edward Coke, who could have seen the first production of *Hamlet*, wouldn't have known a byte from a bit, but he would have had little trouble recognizing the concepts of property rights utilized in assembling the body of law applied to computer technology. In particular, the laws of trade secrets and unfair competition depend upon common law concepts, and until recently common law

was an important source for even copyright protection.

OTHER INFLUENCES

But common law is just one of several sources for American computer law. Next, there is the Constitution, which, in spite of the framers' ignorance of devices such as computers, contains a provision of considerable importance for the computer world. It's stated in very simple language that Congress is given the power to secure "for limited Times to Authors and Inventors exclusive Right to their respective Writings and Discoveries." This is the entire basis for our present copyright and patent laws. (Of course the specific laws derived from this provision and enacted by Congress run for pages and pages.) You'll notice that the Constitution seems to link "authors" with "writings;" it was quite a while before it was clear that a computer program, so unlike the "writings" of the framers' experience, could be protected by copyrights deriving from this language in the Constitution.

Difficulties of this sort are almost unavoidable when established legal principles are applied to new technologies. The infant motion picture industry ran into problems obtaining copyright protection for its films, because existing concepts of copyright law couldn't encompass this novel form of expression. So, until the law was amended in 1912, moving pictures could be copyrighted only as a series of still photographs.

BABEL, INC.

Thus far we've been talking about sources of the law that at least have the virtue of uniformity, however antique they may appear. Now, though, we come up against something designed to toss uniformity out the window — the American federal system.

Europeans and others accustomed to centralized government tend to view American federalism as incomprehensible at best — and at worst, insane. After all, it was the proud achievement of the Napoleonic Code to have ended a situation where one changed laws each time one changed trains. But in the United States many people routinely change to a whole new set of laws merely by commuting to work.

We can have laws that incorporate different and often conflicting public policies.

Most Americans have demonstrated the adaptability of the human species by largely ignoring the multiplicity of legal jurisdictions that surround them. We usually think of this — if at all — in terms of whether we can lawfully get a cocktail on Sunday in a place we are visiting. But the fact remains: we have 50 sometimes highly individualistic state legislatures, who can change the law as they please, subject only to the limits imposed by federal and state constitutions. Add to this the federal Congress, plus assorted territories, plus counties and cities and towns, plus various regulatory agencies — and consider that all of them are busy making laws.

Often the differences in the laws of various states or counties or whatever have simply reflected what the inhabitants wanted. It's hard to imagine the blue laws that have been commonplace in some parts of the country being accepted in freewheeling California, for example.

Lack of uniformity among state laws may constitute little more than a nuisance so far as the traveler is concerned, but this can be a hazard of major proportions for anyone doing business across state lines. Fortunately, some factors do exert a push toward uniformity: most states want to encourage business activity and recognize that oddball laws can be counterproductive.

In an attempt to standardize state laws on various subjects, model uniform laws have been drawn up. Some of these have been widely adopted. One that's significant for the computer community is the Uniform Commercial Code, which we discussed several columns ago.

Even with these efforts, though, state laws still differ widely. Anyone doing business across state lines, or who distributes a product — like software — that can end up being used in another state, has to keep this in mind. The rules that determine liability for defective products, for instance, may be vastly different in another state.

If this weren't confusion enough, keep in mind that each state not only has its own legislature, but also its own court system. Using the principle of *stare decisis*, courts not only decide controversies but also interpret the laws passed by the legislature. Therefore they constitute still another source of legal rulings — and one that has played a major role in the development of computer law.

By now it should be evident why there's no single place you can look to find the laws applicable to the computer industry. In general, you must consult

federal sources for copyright and patent laws; check the laws of the appropriate state for such things as trade secret, employer-employee and unfair competition laws; and search both federal and state enactments for the laws applicable to trademarks.

EATING OUR CAKE AND HAVING IT TOO

There's yet another reason why this multiplicity of sources for our laws has, by and large, suited us. It's simply that we can have laws that incorporate different and often conflicting public policies.

For instance, we don't want to be bound either to complete freedom of access to ideas — which can lead to cutthroat competition — or to a rigid theory of protection in intellectual property. So we try to have the best of both worlds, with antitrust laws existing side by side with trade secret, copyright and other quasi-monopolistic forms of property protection. Most people feel that this arrangement works. But there is a price — complexity. It's just not realistic to expect it to go away. Sorry.

Questions from readers on legal subjects are welcomed. Even if Mr. Groenewold is not able to respond individually, questions dealing with subjects of general interest to the UNIX community will form the basis for future columns. Mail can be sent to Mr. Groenewold at 520 Waller Street, San Francisco, CA, 94117. Electronic mail can be sent to ucbvax!olympus!its!glenn.

Glenn Groenewold is a California attorney who devotes his time to computer law. He has served as an administrative law judge, has been active in trial and appellate work and has argued cases before the state Supreme Court. ■

THE UNIX GLOSSARY

Text processing utilities

by Steve Rosenthal

Note: where terms have several definitions, only the ones most directly connected with text processing are listed.

$ — in the regular expressions used to specify patterns for **grep**, **awk** and other pattern-matching utilities, the **$** sign stands for the end of a line (or, in **awk**, the end of a field). A pattern ending with a **$** sign is said to be anchored.

$ — in the **awk** pattern-matching report generator, this is used to precede a digit indicating the field number. For example, **$3** refers to the contents of the third field in each input line.

&& — used in the **awk** pattern-matching report generator as the boolean **AND** operator, meaning both conditions in a pattern must be matched in order to invoke the accompanying action.

***** — in the patterns checked for matches in **grep**, **awk** and other text programs using regular expressions, * stands for closure (the repeat factor) {0, }, meaning zero or more repetitions of the previous character or pattern. For example, the pattern **[0-9]*[a-z]** matches a letter with an optional numeric prefix.

+ — in the patterns checked for matches in **egrep**, **awk** and other text programs using regular expressions, + stands for closure (repeat factor) {1, }, meaning one

or more repetitions. For example, the pattern **[0-9]+[a-z]** matches a letter with a required numeric prefix.

. — in the patterns used for matching in **grep**, **awk** and other programs using regular expressions, the dot (as the period is usually called) matches any character.

/ — used to enclose the pattern to be matched in the **awk** report generator. One slash is used before and one after a pattern, which if matched invokes the accompanying action. For example, **awk '$2 ^ /194[0-9]/ {print $0}'** prints any line with a date in the 1940s in the second field.

? — in most implementations of **grep** expressions, **?** stands for closure (repeat factor) {0,1), meaning zero or one repetition. For example, the pattern **[a-z]{3}s?** matches all three-letter words and their plurals formed by adding "s".

^ — in **grep** or **awk** patterns, the caret matches the beginning of a line (except used inside square brackets to mark classes).

^ — in most implementations, if the **^** appears in a **grep** or **awk** expression inside a class (a range marked by [ ]), it inverts the class to mean everything outside the range. For example, **[^0-3]** matches anything except the digits 0,1,2 and 3.

{ } — used to enclose the action part of an **awk** pattern-matching report generator specification. Usually, the action is preceded by a pattern, which is enclosed in slashes (/pattern/). See / above.

| — in **grep** or **awk** regular expressions, the | symbol represents alternation (the logical "or" operation).

|| — used in the **awk** pattern-matching report generator as the boolean **OR** operator, meaning that either one or both conditions in a pattern must be matched in order to invoke the accompanying action.

action — in the **awk** pattern-matching language, a set of operations to perform if a match is found for the accompanying pattern.

awk — a pattern-matching report

Whatever System You Use

. . . there's plenty of UniPress Software from our product library

PRODUCTIVITY AIDS

- Emacs Multi-window extensible editor
- Minimacs—Emacs without MLISP
- Phact Isam File Manager
- /RDB — Relational Database
- The Menu System

APPLICATIONS

- Uniplex II Integrated Software System
- Q-Calc Spreadsheet
- Leverage List Processing

LANGUAGES

- Lattice C Native
- Lattice C Cross to 8086
- Amsterdam Compiler Kit

SUPPORTED O.S.

Unix 4.1, 4.2, System III/V, V7
VMS
MS-DOS
Xenix

In addition, we carry the UniPlus + Unix System V for the Apple Lisa and a full line of application software dedicated to the Lisa.

Call or write our trained staff for more product information

UniPress Software, Inc.

2025 Lincoln Highway, Edison, NJ 08817
201-985-8000 • Order Desk: 800-222-0550 (outside NJ) • Telex 709418
Mastercard and Visa
Japanese Distributor: SofTec Tel.: 0480 (85) 6565

generator that compares lines in an input file against a set of patterns, taking the action associated with any of the matches. For simple uses, the patterns are often given in the command line, but they can also be specified as a file. The **awk** program is similar to **grep**, but more powerful, with more flexible patterns and the ability to specify a variety of actions as the consequence of a match.

BEGIN — a pattern built into **awk** that matches the beginning of a file. It's used mostly with already prepared **awk** programs to specify actions to take when processing the beginning of a file.

cat — the command to *concatenate* files (or *catenate* them, as some people prefer). The theoretical purpose of this command is to combine several files, but its most common use is to list file contents on the terminal. It can also be used to create a new file from terminal input.

closure — a repeat factor used after elements of regular expressions to indicate that the element should be used a specified number of times. The general form is: **hb0{l,m}**, where **l** is the least number of times and **m** the most number to repeat the element. If **l** is missing, it is assumed the value equals 1; a missing **m** is assumed to signify an infinite quantity. In addition, * stands for {0, }, + for {1, }, and **?** for {0,1}.

cmp — a utility program that checks two files for differences, comparing them byte by byte. Unless used with the **-l** option, it stops at the first difference. The command is mostly used with textfiles to see if they are absolutely identical; if not, differences are searched for with **diff**, **comm** or another utility.

comm — the command to print lines that are common to two files. The input files must be sorted in the same order. By default, **comm** prints lines in each file individually but not in both, plus lines in both. However, any one of these can be suppressed using the appropriate flags.

csplit — a variation on the **split** command that outputs smaller

USENIX

1985 Winter Technical Conference
Fairmont Hotel, Dallas, Texas
January 23 - 25, 1985

The 1985 Winter USENIX Technical Conference,

sponsored by the USENIX Association, is designed to promote the exchange of technical information and ideas among the UNIX* Community.

Technical Sessions

The technical sessions will cover: Networking, Distributed Systems, Software Tools and Applications, Languages, Environments, USENET and Future Directions, and other topics. Current research and development within the UNIX Community will be presented. Birds of a Feather sessions will be structured to provide a forum for technical exchange.

Tutorials

Also scheduled are advanced full day tutorial sessions covering relevant UNIX technical subjects including:

- 4.2BSD Internals
- The ANSI C Standard
- UNIX Networking
- Advanced C Programming, illustrated by Classical Algorithms
- Driver Writing
- UNIX Language Tools
- UNIX Interprocess Communication
- UUCP, Mail and News
- UNIX Systems Administration

Also ...

Concurrent with the USENIX Technical Conference, /usr/group will be holding their UNIFORUM Tradeshow/Conference at the Dallas Infomart, January 21 - 25. Continuous free shuttle bus service will be provided between USENIX hotels, /usr/group hotels and the Infomart.

For more information ...

If you have not received the pre-registration packet and wish additional conference information, please contact:

USENIX Conference Office
P.O. Box 385
Sunset Beach, CA 90742
(213) 592-1381 (213) 592-3243

*UNIX is a trademark of AT&T Bell Laboratories

Circle No. 274 on Inquiry Card

files made from sections of a large one. The **csplit** (contextual split) version divides the file where it finds matches to specified patterns rather than at fixed line intervals. It's normally found only in System III or System V.

cut — a command that selects only specified columns from a file containing tabular material. It is often used in combination with **sort** and **join** to make new tables combining information from several existing files. It's normally available only on System III or System V.

diff — the command to compare two textfiles, outputting a list of the differences between the two. When used with the **-e** option, the list of differences will be listed in a form suitable for use as input to the **ed** editor, where it can be used to transform the first file to match the second. The **diff** command is often used to store just the differences between versions of a revised program or document, rather than keeping the full text of all past versions.

dot — the common name for the period. See . above.

egrep — an extended version of the **grep** utility that outputs the lines in a file matching a specified pattern. The **egrep** version allows various patterns to be combined with | (the alternation sign, meaning "or"), so that lines matching any of the patterns are output. It requires large amounts of memory, so it may not run on all systems.

END — a pattern built into **awk** that matches the end of a file. It's often used to print summaries at the end of a file.

fgrep — another version of the **grep** program used to print lines in a file matching a specified input pattern. The pattern for **fgrep** must be a literal string rather than a general pattern, allowing for faster processing.

field separator — a character that separates the input lines into sections that can be referred to by positional parameters in the pattern and action specifications used by the **awk** report generator. Space and tab are the default field separators, but the list can be set to use other characters.

grep — a command to print to the

PC*works*.
The Right Connection.

With TouchStone's PCworks™ software it's easy to connect industry-standard PCs to a variety of other computers and information services. Now previously isolated PCs can share information with other computer systems.

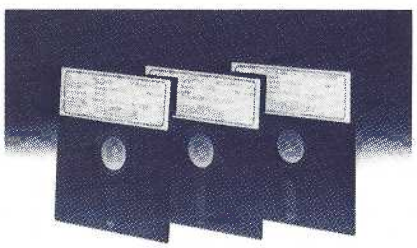
It's So Simple

Just load the PCworks program into your PC, plug in a standard cable or telephone line, and your PCs are ready for work as part of a network. PCworks will do the rest – dial the right numbers, log onto another computer, and start a program. When you want your PC as a PC, no problem. PCworks uses the MS-DOS™ operating system, and will run with all your favorite PC programs.

PCworks and UniHost. A UNIX Software Connection.

PCworks in combination with TouchStone's UniHost™ software provides a special connection from your PC to over 50 kinds of UNIX™ computers. With the PCworks/UniHost combination the UNIX (or Xenix™) system acts as a network manager, allowing PCs and UNIX system users to communicate with each other as well as take advantage of more powerful UNIX system hardware and software.

Now PCs can share electronic mail, exchange all types of files, and use the UNIX system disk and printer.



THE CONNECTABLES™:
FAMILY OF COMMUNICATION SOFTWARE PACKAGES.

Move a spreadsheet from your PC to the UNIX system. Transfer a text file from the UNIX system to your PC for editing. Print the file later on the UNIX system printer, or use the UNIX system disk for backup.

The Connectables Software from TouchStone.

The Connectables™ family of communication software packages from TouchStone offers a simple solution to networking problems. Look for MacLine,™ the newest member of the family, which provides connections to PCworks and UniHost for Macintosh™ computers.

TouchStone
Software Corporation
909 Electric Avenue, Suite 207, Seal Beach,
California 90740 213/598-7746

Circle No. 236 on Inquiry Card

standard output all lines in an input file matching a specified pattern. The name stands for **g**lobal **r**egular **e**xpression **p**rinter. The command is often used as a quick way of pulling lines out of a list, or as a filter in a pipeline.

head — a command in the Berkeley versions of UNIX that outputs the initial lines of a file (defaults to 10 lines). It's most often used to check whether a file is indeed the one a user wants, or whether the right type of output was created by some program.

join — a command used to combine tables from two files by concatenating specified fields from the rows in the second file to the corresponding rows in the first file. It links the lines where the values of a specified column in the second file match the values in a column in the first file. The **join** command can be used to combine two tabular files that share at least one field containing common information.

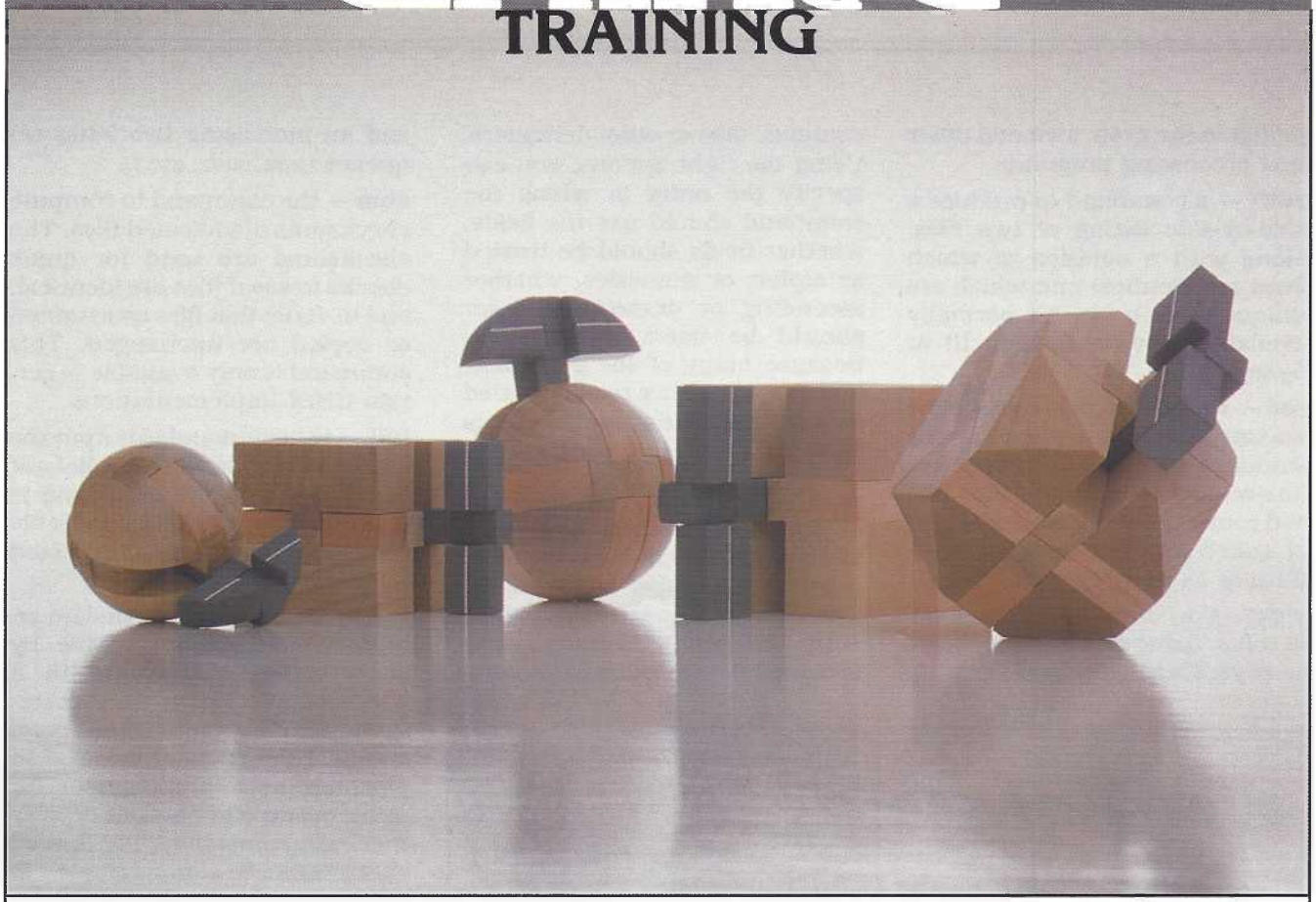
NR — a built-in variable in **awk** that holds the number of records (the line number, where each record exists is considered a line). It's often used for computing averages or making counts.

paste — a command to combine two files of tabular material by concatenating each set of corresponding lines. The resulting table has the same length as the input files, but its width is the sum of the two original widths. It's normally available only on System III or System V.

pr — the command to prepare a file for printing. It separates files into pages with headers, footer margins and page numbers. It can also produce multicolumn output. It's assumed output will go to the screen but, in fact, it is normally redirected to a file or line printer.

regular expression — a pattern or template produced by combining literals, metacharacters, wildcards, ranges, repeat factors and other similar elements that stand for text strings. Regular expressions are used as the patterns to

UNIX™ & 'C' TRAINING



MANY UNIX-BASED SYSTEMS ONE UNIX TRAINING COMPANY

The Computer Technology Group provides the UNIX training solution. Training to fit the complexities of your UNIX-based system.

Three factors make the Computer Technology Group the experts in UNIX and 'C' language training:

- Experience, through training thousands of students worldwide in live seminars, with thousands more using our video training at their own locations.
- Extensive Curricula Supporting All UNIX Versions, creating a client base of manufacturers, software developers and end users.
- Quality of Instruction, with instructors and course developers who are experts in teaching UNIX and 'C', as well as in designing and implementing a variety of UNIX-based systems.

ONE UNIX TRAINING COMPANY MULTIPLE DELIVERY SYSTEMS

Whether you're training two, 200, 2000...you can select the most efficient and economical training solution for your unique environment:

- Public Seminars offered in major cities throughout the world.
- On-Site Seminars for training customzied to your system and to specific groups within your organization.
- Video-Based Training for consistent training that is always available at your location.
- Interactive Videodisc Training, which dynamically tailors courses to the individual—from novice to expert programmer.

ASK FOR OUR 48-PAGE COURSE CATALOG, WHICH PROVIDES:

- Comprehensive course outlines
- Course prerequisites
- Curriculum recommendation for multiple audiences
- Guidelines for cost-effective training media selection
- Current seminar schedule

CALL (800) 323-UNIX or
(312) 987-4082 in Illinois

™ UNIX is a trademark of Bell Laboratories.

COMPUTER TECHNOLOGY GROUP

Telemedia, Inc.

310 S. Michigan Ave., Chicago, IL 60604

Circle No. 266 on Inquiry Card

match in the **grep**, **awk** and other text processing programs.

sdiff — a command to produce a side-by-side listing of two files, along with a notation of which lines are identical and which are unique to each file. It's normally available only on System III or System V.

sed — the command to invoke the stream editor, a version of the standard **ed** editor that operates line-by-line on an input file. The **sed** command is often used as part of a shell script for updating text-files, or as part of a pipeline.

sort — the command to sort lines in a file, using one or more fields as keys. Fields can be defined by commas, tabs or other delimiters. Using the right options, you can specify the order in which the command should use the fields, whether fields should be treated as alphas or numerics, whether ascending or descending order should be used, and so on. Because many of the other text processing utilities require sorted files as input, the output of **sort** is often piped to other programs.

split — the command to make smaller files from a large one. Each of the new files can be up to a specified number of lines long. The smaller files are each named with the input filename or a specified name, followed by an x and an increasing two letter sequence (xaa, xab, etc.).

sum — the command to compute checksums of indicated files. The checksums are used for quick checks to see if files are identical, and to verify that files transmitted or copied are unchanged. This command is only available in certain UNIX implementations.

tail — the command to output the last lines of a file (with the default being 10). The **tail** command is most often used to check that a file contains the complete expected text.

tr — a command to translate an input file to an output file by substituting characters in a specified list with the corresponding characters in a second specified list. It is often used in pipelines as a preprocessor for printers and other output devices that have non-standard or limited character sets (for example, to translate all small letters to capitals).

uniq — the command to delete identical successive lines in a file. The same operation can be done with the **-u** option on the **sort** command if **uniq** is not available.

wc — the command to count characters, words and lines. It is most often used with the **-l** option to count lines only, and typically follows **grep** or another utility that produces a line for each item to be included in a group. The default for **wc** is standard input, but it's most often used with redirected input.

If you have comments, please send them to Rosenthal's UNIX Glossary, Box 9291, Berkeley, CA 94709.

Steve Rosenthal is a lexicographer and writer living in Berkeley. His columns regularly appear in six microcomputer magazines. ■

OFFICE SYSTEMS

EXPERIENCE THE WORLDS OF UNIX*

UniForum '85 is your passport to the fascinating "Worlds of UNIX." You'll examine the growing impact of UNIX in Office Systems, Personal Computers, Engineering/Programming, and Market Trends at UniForum '85, the largest UNIX event ever held.

PERSONAL COMPUTERS

More than 200 major vendors, in 850 booths, will display and demonstrate all that's new in UNIX products and applications.

An extensive conference and tutorial program will expand your UNIX database. This program, organized by /usr/group, will include 14 all-day tutorials on user-specific aspects of UNIX...40 in-depth and informative conference sessions...four nationally-known plenary speakers.

In addition, a number of introductory courses in UNIX will be presented throughout the event.

UniForum '85 will be your total UNIX experience. Whether you're just getting started... or are a seasoned UNIX veteran...UniForum '85 is *the* UNIX event of the year.

Sponsored by

/usr/group
The International Network of UNIX Users

For Complete Information, Call:

1-800-323-5155

(In Illinois, Call: 1-312-299-3131)

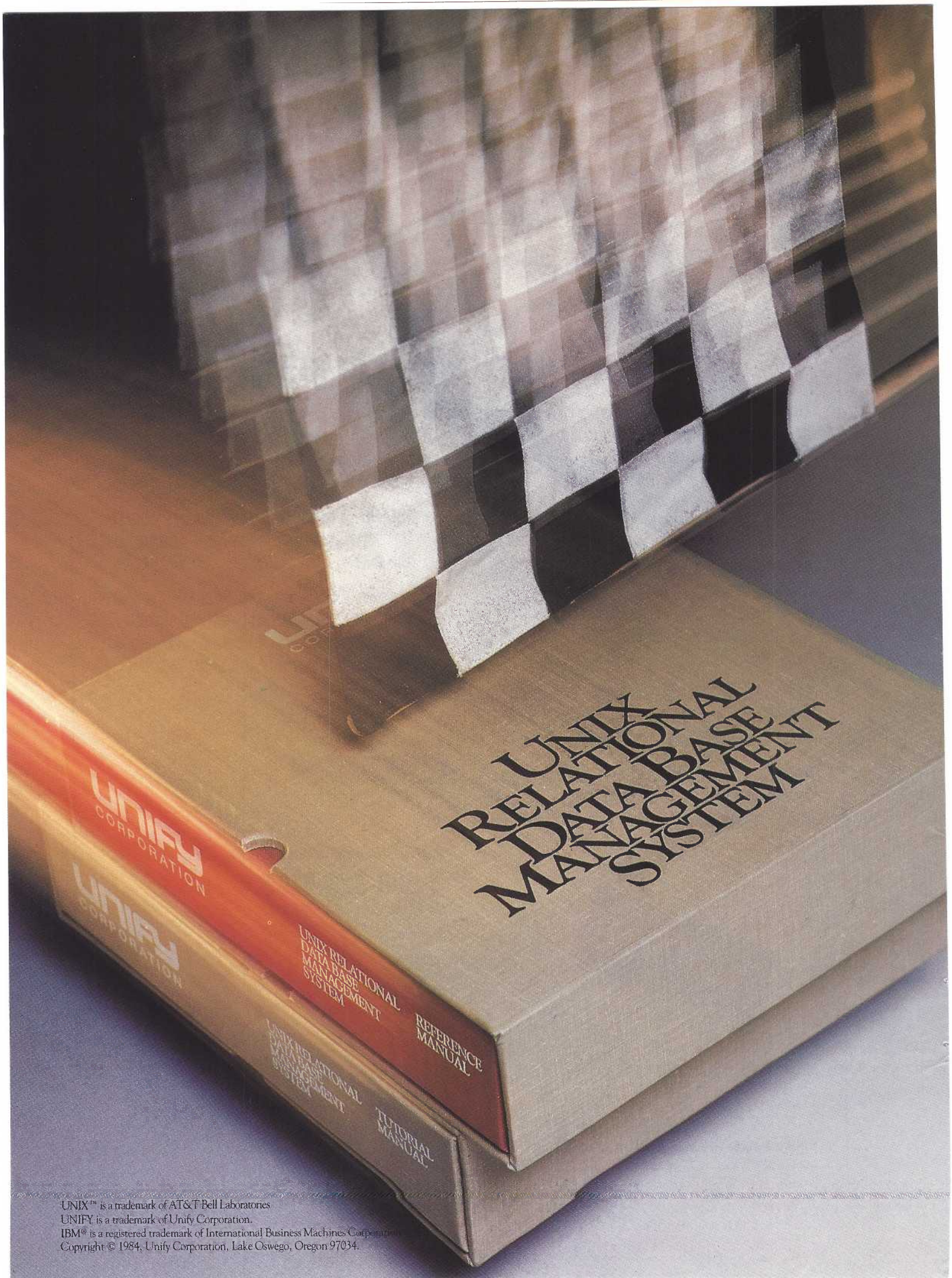
Or Write:

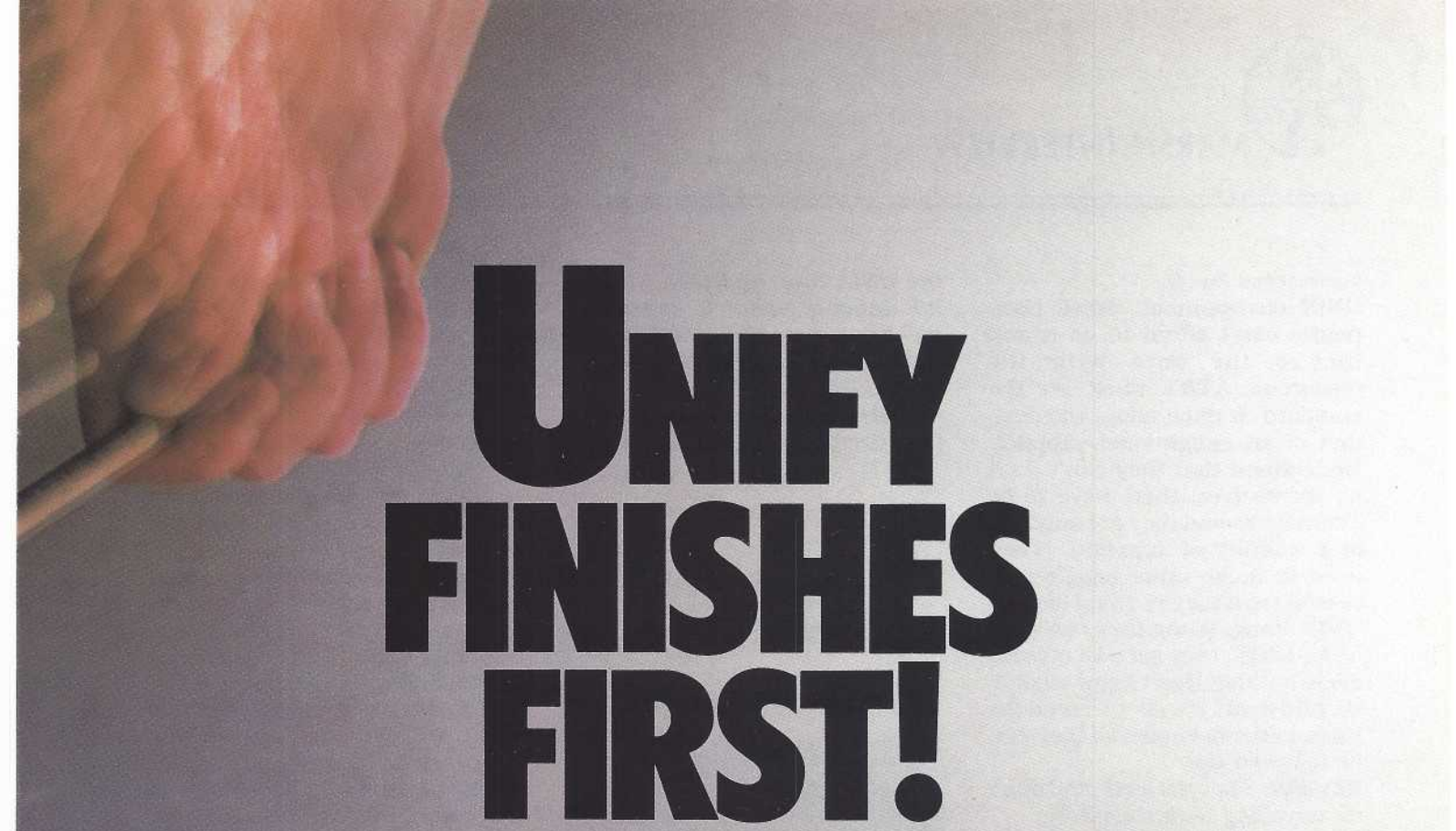
UniForum
Suite 205
2400 East Devon Avenue
Des Plaines, IL 60018

MARKET TRENDS

ENGINEERING/ PROGRAMMING

*UNIX is a registered trademark of AT&T Bell Laboratories.





UNIFY FINISHES FIRST!

In one independent competition after another, UNIFY has proved itself the fastest UNIX data base management system. No wonder it's been selected by more computer manufacturers than all other UNIX data bases combined.

UNIFY speeds you through development and expedites program execution with some of the most powerful utilities of all, including:

Fully menu-driven design.

A powerful screen handling package that helps you format screens quickly, with no coding required.

Raw I/O, that lets you bypass the UNIX file system for up to 40% faster performance in large data bases.

Built-in optimizers that select the fastest of four data access methods.

Industry standard IBM SQL query language, plus our powerful report writer, for easy access by end-users.

Ninety subroutines for advanced program development... the most complete package of its kind.

UNIFY's integrated design links program modules like screens, query language and report writer to help you quickly create complete, friendly, easily expandable applications.

Horsepower for the long run. Unlike other data bases, UNIFY won't slow down under the weight of additional data or multiple users. It's built with the power to support new features later.

Judge for yourself. Send for our 300-page tutorial and 500-page reference manual—yours for only $95—that show you how to build virtually any application. Contact UNIFY, Dept. MMS-11, 4000 Kruse Way Place, Bldg. Two, Suite #255, Lake Oswego, OR 97034, (503) 635-6265, TELEX 469220.

UNIFY®

THE PREFERRED DBMS.

Circle No. 263 on Inquiry Card

Continued from Page 49

UNIX environment. Small companies can't afford to do it and they're the ones with the resources. AT&T must set the standard. It must adopt the position of an enlightened supplier. Understand that they can't do it by themselves; there have to be alternatives and they are only one of a number of suppliers. They need to make other people successful too if they're going to push UNIX along. When they run a big ad for UNIX, they get a lot of leads back, but they don't know what to do with them. It's simple: send the leads to the licensees so they can be followed up.

REVIEW: *They get leads and don't do anything with them?*

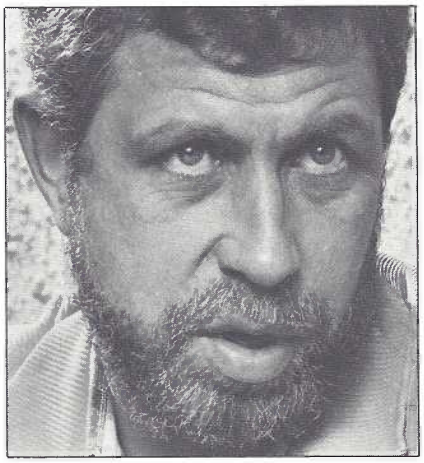
MARSH: I don't know what they do with them, but they don't send them to us. They should be realistic and enlightened about the marketplace. Forget about the PCs. PC-DOS is it on the PCs. Find ways to co-exist with it. The MIS managers have figured out the PC world. And now they have to figure out what to do with UNIX. Where does UNIX fit in their world? They need help, and AT&T is in a position to give it. But I think they view themselves in a struggle with MS-DOS and other systems. A better approach would be to understand where UNIX fits and help people see that.

This stupid business of sending vendors nasty letters when they use UNIX as a noun instead of an adjective is an irritation to licensees. What I'd rather see is some positive incentives. Give me a rebate on this stuff and encourage me to do some training and support. Then, if I don't do it well enough, remove my discount. That sort of positive incentive program would be a lot more effective than a legal contractual enforcement program.

REVIEW: *While AT&T has been sorting things out, IBM has joined the UNIX fray, and with the PC-AT announcement it appears it will be a major player in low cost UNIX. Any thoughts on where it will go from here?*

MARSH: IBM's recent strategy has been pretty clear. They're going to use the PC all over the place. They had the mainframe market nailed on one hand, and then almost by accident, I think, they ended up with control of the desktop market. I think their strategy will be to serve as many different needs out there on desks as possible and just squeeze out anybody else in between. The interesting thing is that they've got UNIX on the 4300, the PC, the Series 1, the 9000, and now they're talking about XENIX on the PC-AT. That's UNIX of a lot of different machines.

Three years in this business is a lifetime.



We did a lot of things fast. We didn't have much money.

They may not have an overall strategy, but before very long they'll be in a position to show a consistent operating system across all of their hardware, which is something they've never been able to do before. That's mind-boggling.

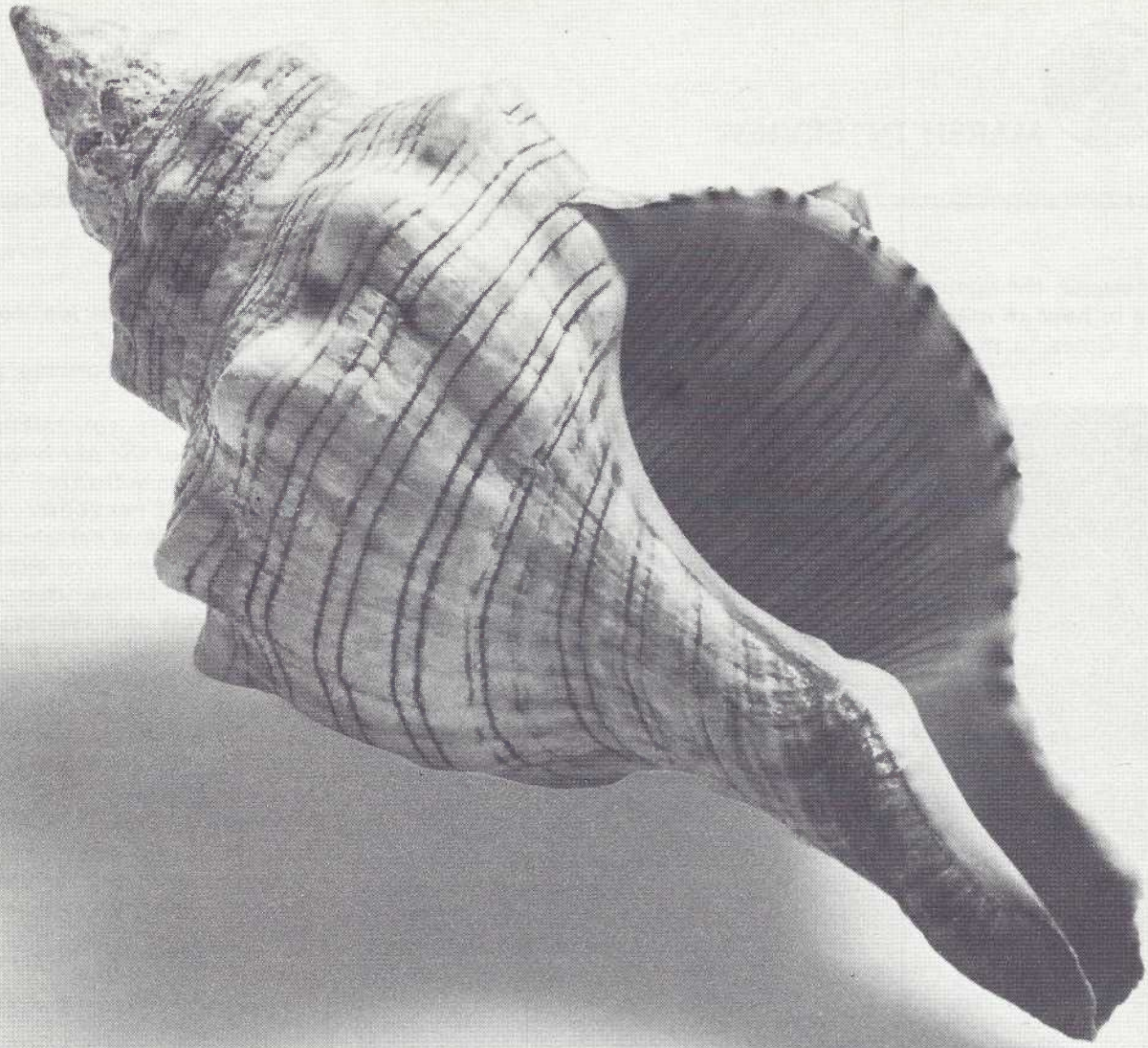
REVIEW: *One of the purposes of this interview is to explore low cost UNIX. Why don't you take a stab at defining low cost?*

MARSH: I guess when I think of low cost, I think of the smallest, most serious business computers — something like the Macintosh. That's a low cost machine, but it's not a toy — it's a high quality machine that's acceptable in a business environment with packages running on it costing $150 each. But the operating system is taken for granted, since it's bundled in with the Mac's low cost processor.

As for low cost UNIX, as long as it's running in its current form, it'll be running on machines that cost $6000 to $7000 — until the disk prices come down, or until a new generation of disk technology comes along.

REVIEW: *It used to be that CPUs were expensive, but micros took that out of the picture. They're down to a few hundred dollars now.*

MARSH: Well, a few hundred dollars is still too expensive, if you're talking about material costs. If you sell a computer for $3000, you can bet the material costs were very low, because if you sell it retail, it's going out the door of the manufacturer, at say, 40 percent off list, maybe even 50 percent off list. So, if you're talking about a box that costs $1500



Try listening to UNIX

That's what we do in UNIX REVIEW
We listen, and look, and probe
It's surprising how much there is that's fresh and new.
Just waiting to be shared
It's often that way with things that are simply elegant.
Like a shell. Like UNIX

UNIX™ REVIEW

THE MAGAZINE FOR THE UNIX COMMUNITY

wholesale, the manufacturer is going to have to mark it up four or five times to make any money. When you're talking about material costs, a couple hundred dollars here and there is expensive.

REVIEW: *Why is low cost UNIX hard to achieve?*

MARSH: It's just pricing, pure and simple.

REVIEW: *Are you talking about the price of the hardware, or the price of the software?*

MARSH: Everybody in this business prices certain hardware configurations very aggressively and then builds margins into peripherals and add-ons, so that when they sell a typical system or a fully-loaded system, they make money.

You need to do the same thing in the software world. AT&T needs to figure that out. The strategy I would use is to make the

Nobody buys an operating system, per se.

operating system free. Of course, the operating system is little more than the kernel and a few utilities.

I'd even throw in a simple shell. Then I would sell the utilities and communication packages. People in typical configurations are going to end up buying those things. I'd have to have good utilities because I'd have to be able to compete and it's a tough game, but you can make money there in the end.

I would price the operating system very aggressively just to guarantee it was going to be the standard. Forget about making money there. This whole business of the number of users, and what the royalty amounts to is crazy. It makes for just too much record keeping, for I would much rather

EMERALD ONE™

SOFTWARE WITH COURAGE, BRAINS AND HEART

WHAT IS EMERALD ONE?

The most complete integrated office system available today, EMERALD ONE combines the most essential office tasks through six fully compatible and seamless sets of tools. EMERALD ONE runs on a broad range of mainframe, mini, super-micro and personal computers which use the UNIX™ operating system–the emerging standard for the office.

THE TOOLS OF EMERALD ONE

EMERALD ONE integrates your office tasks through:

1. *COMMUNICATIONS*, including Telephone Messaging and Electronic Mail systems,
2. *INFORMATION HANDLING* with EMERALD ONE's powerful Relational Database system,
3. *DECISION SUPPORT* features such as Business Graphics and the Electronic Spreadsheet,
4. *DOCUMENT PREPARATION* with Word Processing and a Cabinet, Document and File Folder system,
5. *TIME MANAGEMENT* tools such as the Personal Diary system and Meeting Scheduler and
6. *SYSTEM ADMINISTRATION* functions that allow a non-technical user to customize EMERALD ONE for the individual, work group and organization with ease.

SOFTWARE FOR THE WORK GROUP

EMERALD ONE goes far beyond stand-alone personal computer software by linking individuals and their work groups. With EMERALD ONE, users work as a communicating group, not as isolated individuals. Whether it be a document, spreadsheet or personal diary entry, everything created with EMERALD ONE can be exchanged easily between individuals, work groups and beyond.

EMERALD CITY, THE PEOPLE BEHIND EMERALD ONE

EMERALD ONE is the result of an intensive, multi-year research commitment by Emerald City and its sister company Trigon Systems Group, one of the most respected consulting companies in office integration.
Attractively priced for distribution by hardware manufacturers, system integrators and OEM's, EMERALD ONE is fully supported by an extensive marketing program designed to assist distributors in penetrating the integrated office market. Emerald City offers the reality of a complete business solution, not just technology.

Emerald City, the company with courage, brains and heart.

EMERALD CITY

Emerald City Inc.
20 Richmond Street East, Suite 700
Toronto, Canada M5C 2R9
(416) 863-9923

See us at UniForum Booth #3038

Circle No. 260 on Inquiry Card

charge a lower fee for the kernel and more for the utilities. Let the customers sort it out. The truth is my customers wouldn't buy 98 percent of what AT&T provides. They just don't supply very much that commercial customers use.

UNIX is not a single-user system.

REVIEW: *So we're talking about a virtual UNIX that is just the line between the parts you don't use below and the parts that you don't use above? You throw the rest away?*

MARSH: That's exactly it. But it's a good enough line that people see a standard in it — UNIX. UNIX serves as a playing field — an imaginary surface that separates two different domains. It provides enough stability, with different alternatives in terms of engines, that people are willing to make investments in it. On the other side is the investment in the applications they are using. That's all an industry standard is.

Everybody gets all wrapped up talking about how UNIX is unfriendly, but every operating system is unfriendly. The operating system is unfriendly in the sense that people don't like talking to it. Well, nobody likes talking to operating systems that I know of — except programmers. And programmers love UNIX, because it's clean, simple and consistent. ■

UNIX just dropped into place.

RECENT RELEASES

SUN WORKSTATION UNDER $10,000

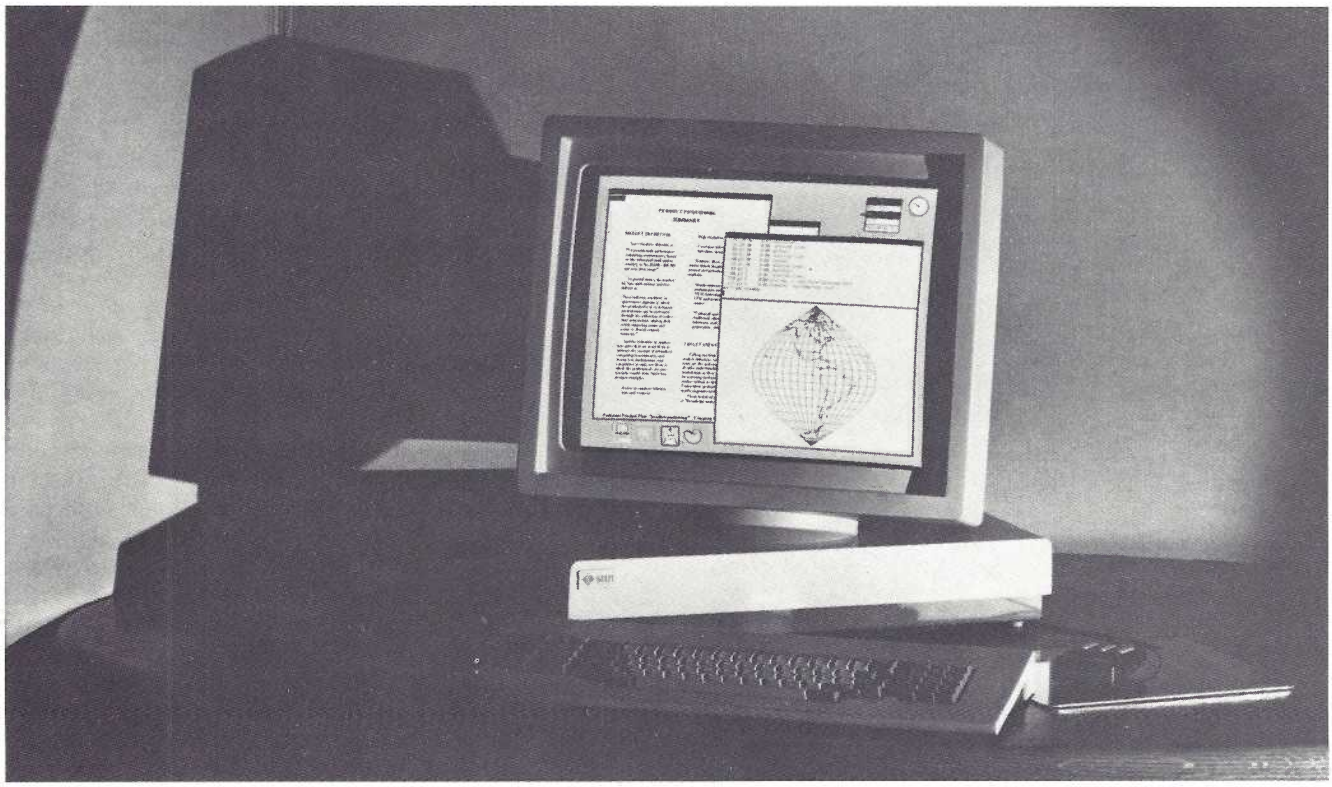
Sun Microsystems has introduced the Sun-2/50 Desktop SunStation, a full-function professional workstation with a list price of under $10,000. The Sun-2/50 is designed specifically for technical professionals who do not require the expansion flexibility of a card cage but who desire the same computational and graphics performance as provided by the existing Sun-2 family of graphics workstations.

Sun's enhancements to UNIX and its Ethernet local area network interface allow the Sun-2/50 to operate without a local disk by demand paging at high speed over the network to mass storage on the fileserver. Several diskless machines can share the disks of a Sun-2/120FS or Sun-2/170 fileserver, thus lowering the cost per user. Users can be added to a network incrementally while maintaining consistent performance. In addition, the single-board design puts the power of the Sun-2/50 in a quiet, compact, desktop package.

The Sun-2/50 CPU is based on the MC68010 microprocessor operating at 10 MHz with no wait states. The Sun patented memory management design allows for 1M byte to 4M byte of physical memory and up to 16M byte of virtual address space per process. A hardware floating point accelerator is optionally available for compute-intensive applications.

The Sun-2/50 has a 19-inch, high-resolution (1152x900), 66 Hz non-interlaced display, with tilt and swivel.

Sun demonstrated the Sun-2/50 Desktop SunStation at the second annual Sun User Group meeting in Boston. List price for a 1M byte system is $9,900, with volume discounts available. The Sun-2/50 is also



available in 2M byte and 4M byte configurations.

Standard on all machines are UNIX 4.2 BSD, networking, SunCore graphics, SunWindows window manager, and C, Fortran and Pascal programming languages. First shipments have begun and customers can expect delivery 60-90 days after receipt of order. Sales are handled through Sun's direct sales offices and through distributors worldwide.

In addition to the Sun-2/50, Sun announced the Sun-2/160 Color SunStation and the Sun Network File System (NFS). The Sun-2/160 Color SunStation is an integrated color workstation with high-resolution display and graphics capabilities. The NFS supports transparent network-wide read and write access to directories and files. As part of its commitment to open systems and industry standards, Sun is publishing the NFS and supporting protocol specifications with the intent of providing a standard for network file system. The NFS protocols are designed to allow heterogeneous systems running different operating systems to share files in a local area network.

Circle No. 254 on Inquiry Card

INTERLEAF INTRODUCES PUBLISHING SYSTEM

Interleaf, Inc., has introduced the TPS-2000 Technical Publishing System, a graphics arts and text publishing system that is an advance in the growing field of electronic publishing.

A fully integrated graphics imaging and text processing system, the TPS-2000 supports a variety of digital typesetters and provides graphic scanning and editing capabilities.

The scanner allows a user to incorporate existing hardcopy artwork and photographs into documents created on the TPS-2000. In addition to accepting scanned images, the TPS-2000 also enables a user to create diagrams and business charts directly on the screen. The TPS-2000 also accepts graphics from CAD systems.

The TPS-2000 complements Interleaf's existing OPS-2000 Office Publishing System, introduced in September, 1983.

The power of the TPS-2000 is derived from a 32-bit Motorola 68010-based workstation and Interleaf's proprietary software, which fully integrates advanced multiple-font text processing, business graphics and interactive diagramming.

The new image scanner gives the Technical Publishing System the ability to produce illustrated documents containing complex line art and black-and-white photographs, along with the text and graphics generated directly on the system.

The image of the object scanned into the system can be positioned on a page and edited directly on the screen. It can also be

Circle No. 257 on Inquiry Card ►

Gould...Innovation and Quality in UNIX-based Systems

The Firebreathers from Gould blast the competition into oblivion.

With blazing performance.

Great leaps in raw performance are rare in the computer world. Usually, changes occur in increments of half-a-MIP or so. Now real Firebreathers roar into the arena, quadrupling the best the competition has to offer.

With scorching speed.

These creatures don't run a little faster. Running real production code supplied by VAX™ users, our PowerNode™ 9000 benchmarks at 4.5 times faster than the VAX 11/780...at a comparable price! Even our second-in-line PowerNode 6000 runs 1.5 times faster. But with all this power and speed, the PN6000 has a dainty footprint...60% less than the VAX 11/780.

With sizzling simplicity.

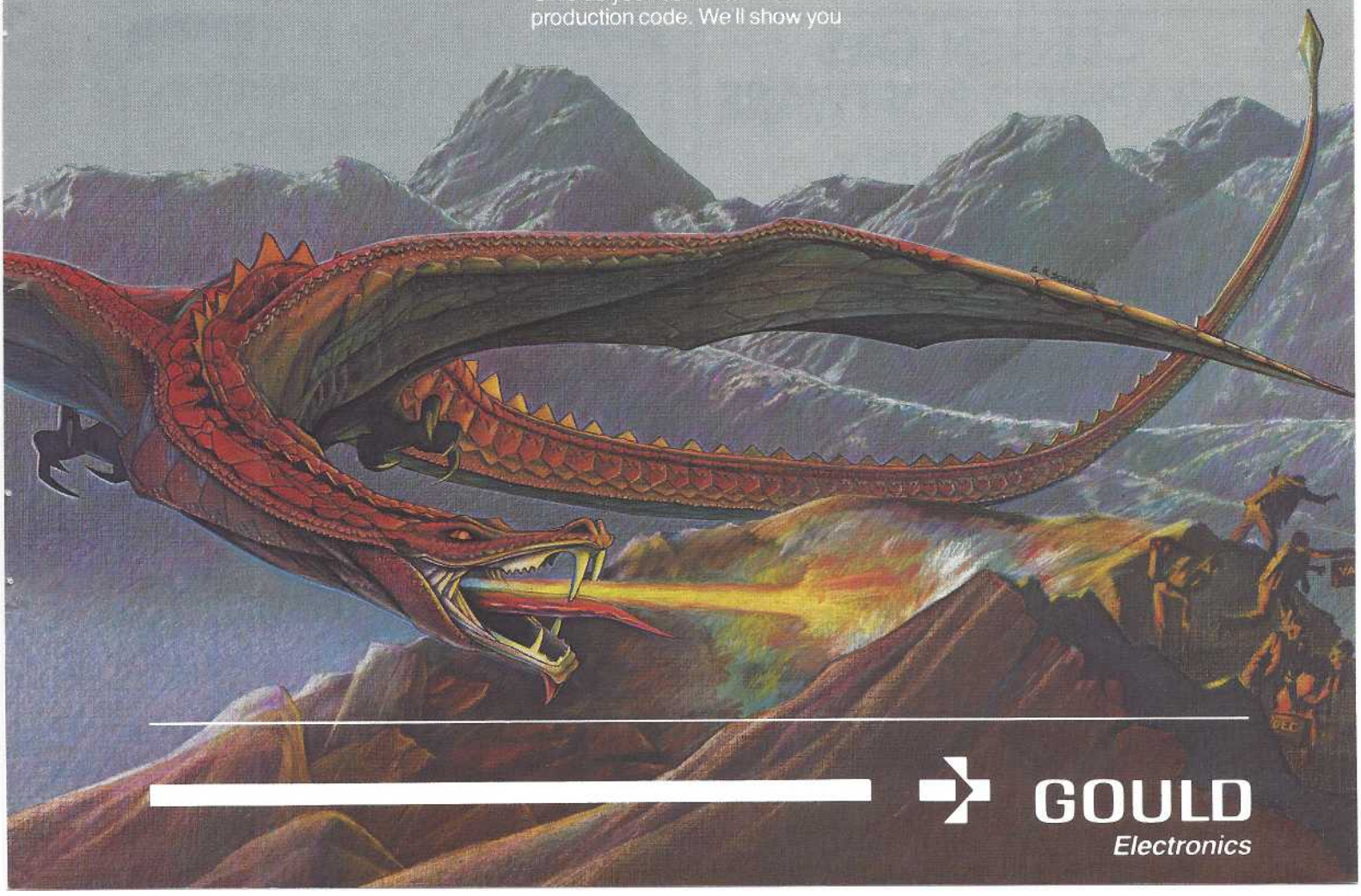
Even the hottest firebreather won't fly if it's hard to run. Ours are UNIX*-based, so ease of use and compatibility become part of the power. Plus we offer the "Compatibility Suite", application software packages which are consistent across our entire Gould PowerSeries™ product line...the widest range of UNIX-based systems in the world.

More than a myth.

We know these claims sound outrageous. Make us prove them. Give us your benchmarks and production code. We'll show you numbers that surpass all expectations. Call 305-797-5459 for a test flight on the Firebreathers. Mainframe performance disguised as a supermini. The scorching performance you expect from Gould.

Gould Inc.
Computer Systems Division
Distributed Systems Operation
6901 West Sunrise Boulevard
Ft. Lauderdale, FL 33313

*UNIX is a trademark of AT&T Bell Laboratories
™PowerNode and PowerSeries are trademarks of Gould Inc.
™VAX is a trademark of Digital Equipment Corp.



GOULD
Electronics

cropped, sized, moved, rotated and "pixel-edited."

As an additional option to the TPS-2000, Interleaf extends the output capabilities of the system by offering interfaces to several digital typesetters. Initially, the TPS-2000 will support interfaces to typesetters manufactured by Monotype, Inc., Compugraphic Corp., Autologic, Inc. and Information International, Inc. Together the image scanner and typesetter interface give the TPS-2000 the ability to produce fully composed typeset pages including photographs and other sophisticated graphics.

Prices for the minimum one-workstation configuration of the TPS-2000 begin at $37,500. The scanner and image editing software cost $40,000, while the typesetter interfaces begin at $7,500. Laser printers, which can be used as either final output device or proof printers for typesetting, can be purchased separately, with prices beginning at $14,500. Availability is 30 to 60 days after receipt of order, depending upon configuration.

Circle No. 253 on Inquiry Card

I/O PROCESSOR HANDLES HIGH DATA RATES

Relief from input/output congestion in a 68000/UNIX environment is provided by a 68000-based I/O processor from Alcyon Corporation. Data bursts up to 9600 BAUD on each of 16 full duplex asynchronous serial data channels are accommodated. Synchronous data rates up to 1.5 MBaud are possible on each of two on-board full duplex channels.

The I/O processor's hardware facilities include, in addition to its 68000, a 4-channel DMA controller, high-speed buffer memory and two programmable synchronous communication channels. The I/O processor occupies a dual-width Q-bus circuit board which can be coupled via a high-speed expansion bus to dual-width option cards. Options include 8-channel and 16-channel programmable asynchronous multiplexers. Asynchronous lines are RS423/RS232. Synchronous lines are RS422 or RS423/RS232.

The 68000 I/O processor joins the line of peripheral controllers for Alcyon's family of 68000-based supermicros. It is fully integrated with REGULUS, Alcyon's

Circle No. 256 on Inquiry Card

Circle No. 255 on Inquiry Card

real-time, UNIX-compatible operating system. Alcyon products target real time and high data rate applications.

Quantity one price for the I/O processor is $2,345; the 8-channel multiplexer is $950, the 16-channel multiplexer $1,555. Delivery is 30 days ARO.

Circle No. 252 on Inquiry Card

SOFTWARE DEVELOPER'S TOOL KIT FOR THE IBM-PC

A software developer's tool kit, designed specifically to allow the IBM PC to be used as a powerful development workstation in mainframe, mini and microcomputer environments, is now available from Lantech Systems, Inc.

Retailing for less than $400, the Tool Kit contains a choice of text editors, a terminal emulator and windowing capability.

The Tool Kit provides the user with a choice of four popular editors: The Bell Editor (an implementation of the Rand Editor), **vi**, **ed** and an EMACS-like editor called "Epsilon."

The Tool Kit also includes uNETix, the UNIX-compatible operating system developed by Lantech. The Tool Kit may be operated in a standalone mode or in conjunction with a host system.

Another product option is the Lattice C Compiler. This option permits development of applications in a local mode without the need for an expensive UNIX host machine. Lattice C Cross Compiler tools can also be applied.

Circle No. 251 on Inquiry Card

BRIDGE TERMINAL SERVER IMPLEMENTING NETWORK PROTOCOLS

An Ethernet local area network (LAN) based terminal server implementing the TCP/IP network protocols, which have been standardized by the Department of Defense and are an integral part of UNIX 4.2 BSD, has been introduced by Bridge Communications, Inc.

Bridge's CS/1 server with TCP/IP software performs the function of a terminal or host server, allowing up to 32 asynchronous devices (e.g., terminals, printers, computers) to access host computers that support TCP/IP and are attached to an Ethernet LAN.

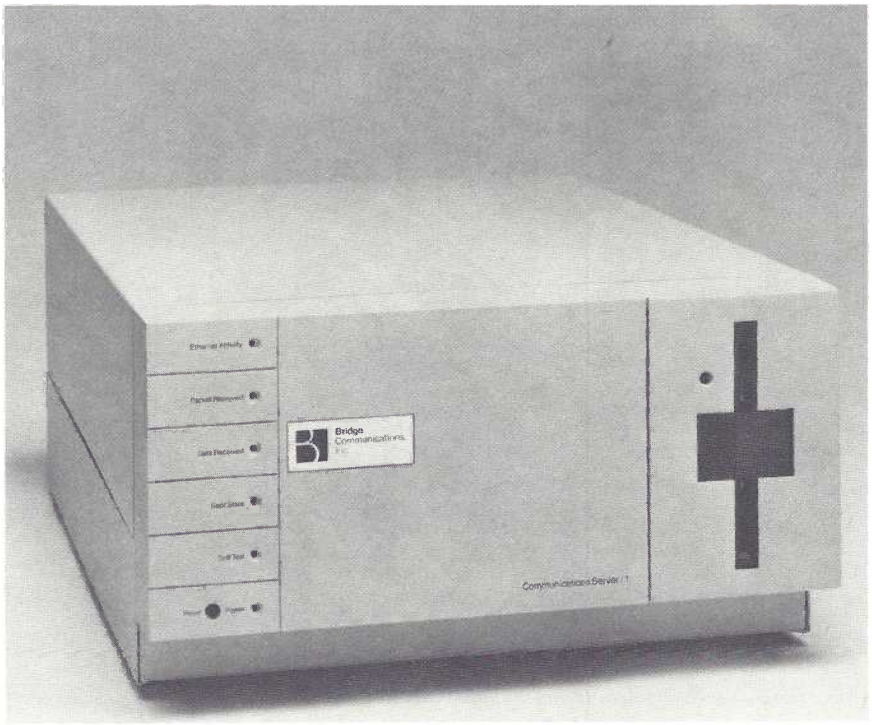
Initial beta site is Contel Information Systems, Great Neck, NY, which in June 1984 agreed to purchase up to $10 million of Bridge products during a three-year period.

The CS/1 with TCP/IP software can be used to connect asynchronous terminals to host computers that support TCP/IP and are attached directly to an Ethernet via an embedded Ethernet controller card such as Digital Equipment Corporation's DEUNA or 3Com Corporation's Multibus controller. In these applications, the CS/1 is needed only on the terminal side of the terminal-to-host switch. However, asynchronous devices also can be connected via Ethernet to host computers that do not support TCP/IP by using CS/1 units running TCP/IP software on both the terminal and host sides of the network.

On the host side Bridge has offered connections to the X.25, DEC and IBM SNA environments. The new CS/1 with TCP/IP joins our standard CS/1 and CS/100 servers on the terminal side.

The TCP/IP (Transmission Control Protocol and Internet Protocol) are higher-level protocols (ISO Reference Model layers 3-5) which provide network and transport services above the physical and datalink protocols defined by the Ethernet specification.

The CS/1 with TCP/IP soft-

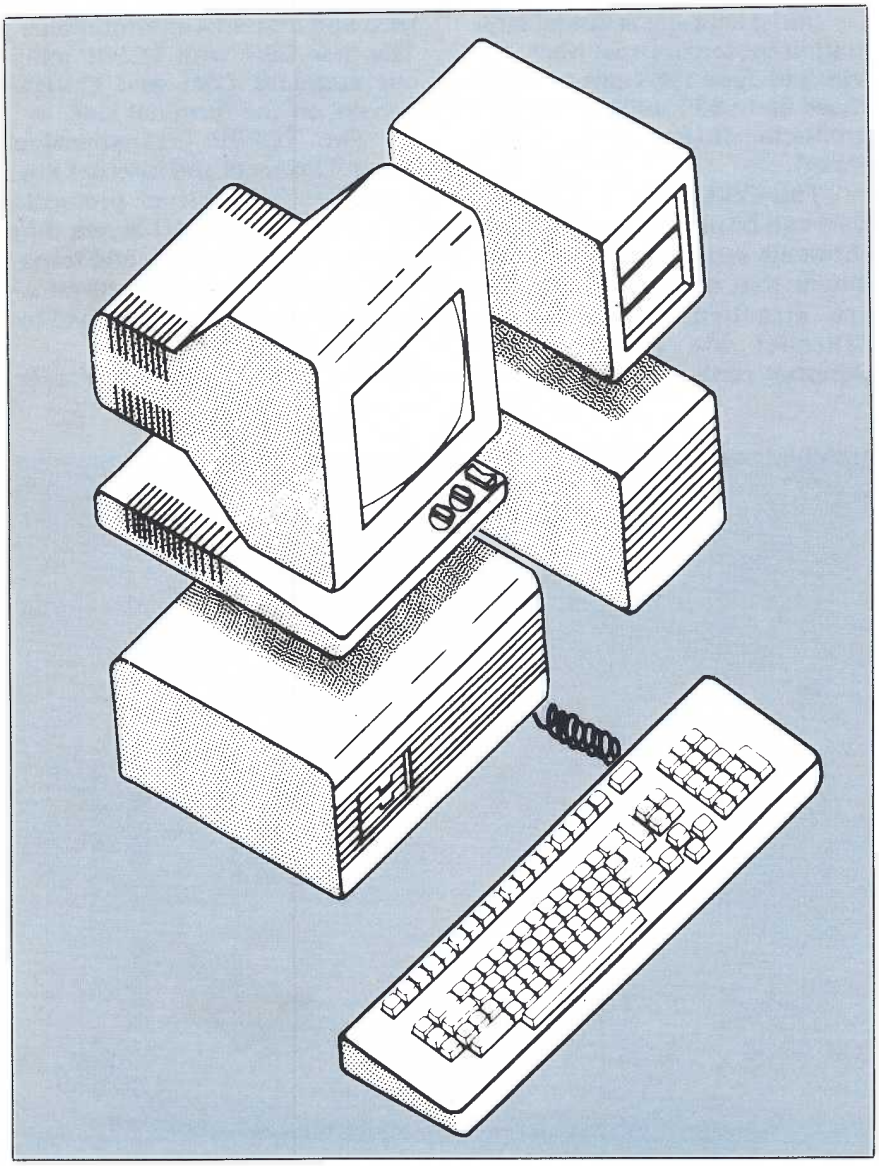


ware uses a standard Telnet protocol to support virtual circuits between terminals and host computers; both User and Server Telnet are supported. Also implemented on the unit to supplement the TCP/IP protocols are the DOD universal datagram protocol (UDP) and the Ethernet address resolution protocol (ARP).

The CS/1 with TCP/IP software offers network management services including usage statistics and Ethernet and server performance statistics accessible from a terminal connected to the unit.

The CS/1 with TCP/IP is available 60 days after receipt of order. The unit is priced at $9,900 with eight serial ports; additional eight-port expansion boards (for up to 32 ports) are $1,900 each. A one-year TCP/IP software license (including maintenance) is $250; the software comes on a standard 5 1/4-inch diskette.

Circle No. 223 on Inquiry Card



FORTUNE FAMILY OF WORKSTATIONS

Fortune Systems Corporation announced a family of workstations that runs UNIX and PC-DOS, and offers users a variety of upgrade possibilities. The options range from a single ASCII terminal to an intelligent workstation or a graphics workstation, to a complete UNIX and PC-DOS networked personal computer with color graphics.

Working in conjunction with Fortune's existing 32:16 central processing unit, the Fortune 1000 desktop workstation family also permits networking between IBM PCs and various Fortune products. And, users have a number of optional resources to choose from: upgradeable ASCII terminal with a 14-inch monochromatic monitor; color monitor; hard or floppy disks, or both, for local-storage capabilities; networking or standalone capabilities; high-speed data link to provide intelligent workstation capabilities with PC-DOS, UNIX or combined PC-DOS/UNIX operating systems; fully configured UNIX/PC-DOS personal computers.

The Fortune 1000's operating systems provide a range of business applications. If users select PC-DOS, they have access to the thousands of software programs designed for the IBM Personal Computer family. Also, UNIX provides users with a wide variety of software including Fortune:Word, the company's recently enhanced word processing system.

Circle No. 250 on Inquiry Card

MASSCOMP HAS RIGHT STUFF

With the recent launch of Challenger Space Shuttle Flight 41-G, MASSCOMP of Westford, MA, and UX SOFTWARE, Inc., of Toronto, Canada, have completed

a project to provide Payload Flight Engineers in the National Aeronautics and Space Administration (NASA), Johnson Space Center (JSC), Houston, with advanced UNIX computer technology to help them plan and ground-control the primary flight 41-G attached payload.

This marks the first time any computer system has been applied by NASA's Johnson Space Center to the task of planning and generating payload commands for ground control of experiments run aboard the shuttle. It is also a first use of UNIX technology by NASA.

NASA chose the MASSCOMP MC-500 supermicro and associated operating system based on its design and applicability to engineering and scientific tasks. To handle the unique requirements associated with flight 41-G payload planning and control, NASA selected UX-Basic, Canadian-designed software. UX-Basic was selected because it offered NASA payload engineers the convenience and power of using C language with the simplicity of programming in BASIC.

While in use on the MASSCOMP MC-500, UX-Basic is helping to manage some 6,000 files of data variables that are used to control and activate four scientific experiments, termed "mission successful" by NASA. These experiments involve the operation of the Shuttle Imaging Radar (SIR-B); Large Format Camera (LFC); Measurement of Air Pollution from Satellites (MAPS); and Feature Identification and Location Experiments (FILE).

The team of NASA payload engineers that programmed UX-Basic for use during Flight 41-G is responsible for processing some 50 pages of computer script and commands each day of the flight. The high performance interpreter and compiler in UX-Basic allowed 41-G payload engineers to successfully reduce the run time of an average program from 20 to 8 minutes.

ANALOG OFFERS FILTER SYNTHESIS SOFTWARE

A computer aided design software package created to facilitate the synthesis of analog filters is being offered by Analog Design Tools, Inc., as an option to its Analog Workbench, an engineering workstation for the design of analog circuits.

The S/FILSYN program will be offered under the terms of a licensing arrangement with the program's developer, George Szxentirmai of DGS Associates, Santa Clara, CA, which gives Analog Design Tools exclusive rights to market S/FILSYN on workstation hardware. The product, previously available only for mainframes and superminis, has recently been adapted by DGS to run on the Sun Microsystems hardware use by the Analog Workbench.

S/FILSYN will permit the Analog Workbench user to design a wide range of filter types, including passive (LC, crystal and mechanical), active, microwave, switched capacitor and digital (FIR and IIR) filters. The program can be used to design lowpass, highpass, linear-phase lowpass, bandpass, band-reject, delay equalizer and bandpass impedance matching filters.

Structured to accommodate any design procedure, S/FILSYN includes: a PLACER segment which specifies filter requirements and determines transfer function parameters; a FILSYN segment which determines transfer functions from PLACER or from data entered at the terminal; a DELAY EQUALIZER segment which is used to design equalizers; a LADDER segment which performs frequency domain analysis and other critical operations; five SYNTHESIS and DESIGN segments for various stages of filter design; and three ANALYSIS segments for manipulation, analysis and reporting of performance criteria.

Priced at $17,500, the S/FILSYN option for the Analog Workbench will be available in the third quarter of 1985.

Circle No. 238 on Inquiry Card

UNISOURCE OFFERS EDUCATIONAL DISCOUNT

Unisource Software Corporation is offering a 44% educational discount on its licensed UNIX software for personal computers. By special arrangement with AT&T and independent software developers, Unisource now offers substantial discounts to qualified schools and universities on the VENIX operating system and UNIX application software.

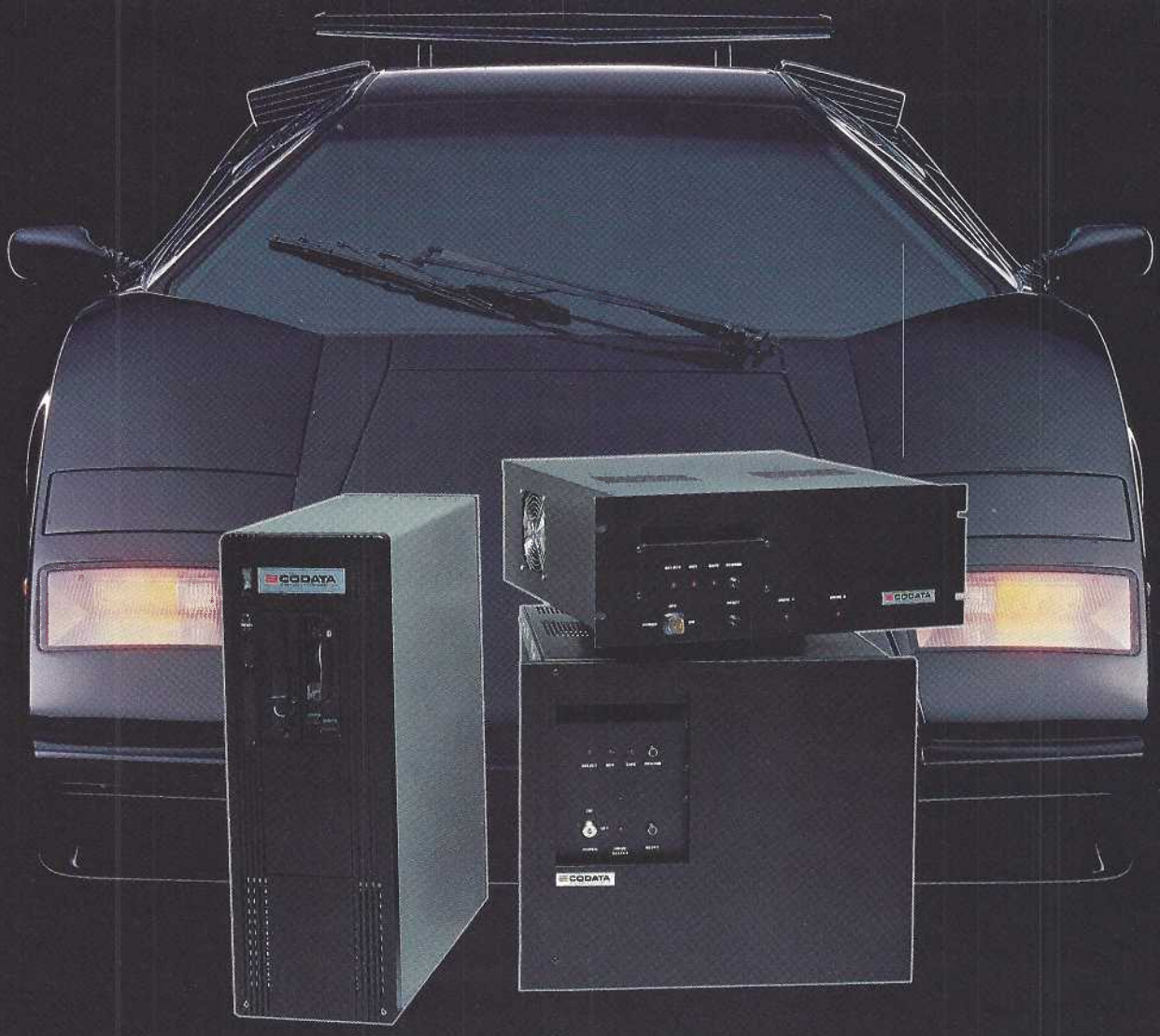
Under the Preferred Educational Discount Program Unisource offers, the recently released ENCORE version of VENIX/86 retailing for $800, is delivered with a System V UNIX license. It is also available for the IBM PC/XT, AT&T 6300, Compaq Plus and DeskPro, Eagle Turbo, MAD 1 and the Leading Edge PC. VENIX/Pro for the DEC Professional/350 is also marketed by Unisource.

Circle No. 233 on Inquiry Card

UNIX SOFTWARE FOR BURROUGHS SUPERMICRO

A line of integrated, full-featured mainframe-quality UNIX-based business application software modules for the new Burroughs XE-550 supermicro has been announced by Precision Software, Inc.

The line — called the UniCentre Series — includes seven modules: Accounts Receivable, Accounts Payable, General Ledger, Inventory Management, Order Processing, Payroll, and



High Performance Machines.

Now, from the same people who brought you the industry-leading price/performance champion 3300-Series supermicro comes a totally new dimension in high performance machines—the Contel-Codata 3400-Series. We've designed the 3400-Series specifically for the OEM marketplace and proudly bring you more versatility, dependability, and expandability than any other manufacturer. It's the Codata Difference and another major improvement in OEM microcomputers.

Whether your application is scientific or desktop, whether you need a graphics engine or a free-standing floor model, the Codata 3400-Series gives you the industry's broadest range of capabilities.

3400-Series features include 8-megabyte RAM addressability, expanded mass storage with 12, 47, 84, 168 or 320-megabytes of high-speed Winchester efficiency, 9-track 800/1600 BPI magnetic tape, hardware floating point accelerator for Fortran/Pascal, Ethernet LAN, and, of course, the M68000 running the Contel-Codata autoconfigurating UNIX™ operating system. All 3400-Series systems are based on Multibus™ architecture giving you nearly unlimited versatility of applications.

Let Contel-Codata put you on the fast track! Test drive the 3400-Series supermicros TODAY. For details contact us at: CODATA SYSTEMS CORPORATION, 285 N. Wolfe Road, Sunnyvale, CA 94086, 408/735-1744, 1-800-521-6543. Telex 172869 CODATA SUVL. In Europe, CONTEL-CODATA, 250 Avenue Louise, Box 101, 1050 Brussels, Belgium. Telex 65942 CADO-B.

MULTIBUS is a trademark of Intel Corp. UNIX is a trademark of AT&T Labs

CONTEL CODATA

CODATA SYSTEMS CORPORATION

Circle No. 269 on Inquiry Card

See us at UniForum Booth #1679

Purchase Management.

PSI's UniCentre line also includes a utility program called "The Shuttle," which takes advantage of UNIX features to provide fast file transfer to and from a Burroughs mainframe.

The UniCentre software operates under UNIX System V, which is provided with the Burroughs XE-550. The software is written in R-M COBOL, a widely adopted version of the common business applications computer language.

The average price of a UniCentre software module is approximately $1500. The primary target market for the software is small to mid-size distributors, manufacturers, and other commercial users.

Precision Software, Inc., is an approved Burroughs third-party software vendor, classified as an ISO in the company's APEX/ISO program.

Circle No. 232 on Inquiry Card

IMAGEN TYPEFACE FOR LASER PRINTERS

Imagen Corporation, a manufacturer of intelligent, non-impact page printing systems, is manufacturing and marketing Lucida, a typeface specifically designed for non-impact laser printing systems. Imagen signed a licensing agreement in September with Bigelow and Holmes, the designers of Lucida. The typefaces were developed to run on Imagen's printing systems.

Typefaces on electronic or laser printing systems have previously been adaptations of fonts used in metal printing technology. Lucida is designed to work with laser and CRT technologies to give maximum legibility on low and medium-resolution devices. The quality and clarity of type producible with Lucida makes it an alternative to phototypesetting.

This typeface is designed to

THE MAGAZINE FOR THE FULL-OUT BRAINO.

Computer Language is written for people who can program in two or *more* computer languages.

Let's face it, that leaves out most people. Programming is a rigorous, intellectual discipline and Computer Language magazine is the first and only publication dedicated exclusively to this field.

Your source for the latest technical skills and methods used by software specialists.

We cover the major developments in the software design field, from theory to implementation. Computer Language focuses on the most important and useful language design information available in the fast-moving microcomputer industry.

Written for the person who takes computing seriously.

We're talking about you — the experienced software author, programmer, or engineer who routinely programs in two or more high-level languages.

A person who understands the creative nature of programming and appreciates the beauty of efficient code in action.

COMPUTER LANGUAGE will constantly challenge your abilities.

The foremost industry experts will discuss: · Algorithmic Approaches to Problem Solving · Language Portability Features · Compiler Designs · Utilities · Artificial Intelligence · Editors · New Language Syntax · Telecommunications · Language Selection Criteria · Marketing Your Own Software · Critical Software & Hardware Reviews

Send to:

COMPUTER LANGUAGE

2443 Fillmore Street Suite #346 San Francisco, CA 94115

YES! Start my subscription to Computer Language. My 1 year (12 issues) subscription is just $24.00, a 33% savings under the single copy price. Guarantee: I can cancel my subscription at any time for a full refund.

☐ $24.00 ☐ Bill me. Payment enclosed

Name

Address

City State Zip U

Circle No. 235 on Inquiry Card

meet the requirements of the workstation marketplace requiring serif, sanserif, display and matching CRT screen fonts. Lucida can be used with the formatting programs supported by Imagen's printing systems such as troff and TeX. Its screen fonts have been designed for maximum legibility and reader comfort, while remaining compatible in appearance with the printer fonts. The digitizations of Lucida were produced by Imagen using outline representations that can be easily converted to new resolutions. The design is the result of research by Bigelow and Holmes into legibility requirements of the printing technologies.

Circle No. 230 on Inquiry Card

RM COBOL ON MASSCOMP SYSTEMS

Potomac Systems Resources, Inc., a UNIX systems integrator, announced the availability of RM COBOL for the MASSCOMP computer systems line of real-time, multiuser microcomputers.

Potomac Systems is supplying the US Department of Justice with a network configuration of MASSCOMP workstation units, along with a supermicro of enhanced capacity. The network features both standalone and distributed systems. The application environment includes updating and querying a central database as well as providing local logistical support.

RM COBOL for the MASSCOMP line is available from Potomac Systems at a list price of $3750 per use; both quantity and dealer discounts are available.

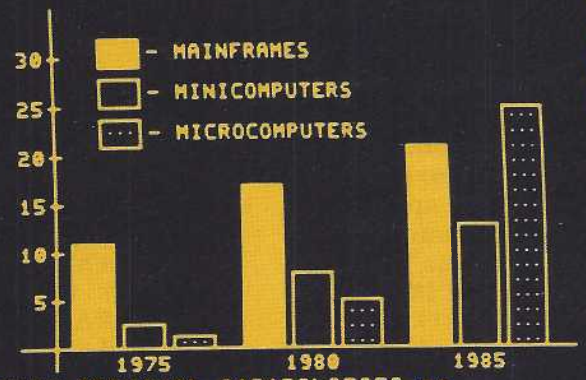
Circle No. 229 on Inquiry Card

RELATIONAL TECHNOLOGY AND BURROUGHS SIGN AGREEMENT

Relational Technology, Inc., (RTI) and Burroughs Corporation have signed an OEM pact granting Burroughs the right to distribute the INGRES relational database management system (RDBMS) with its new supermicro.

The announcement of the multiyear, multimillion dollar

A Picture's Worth:



** GRAPHICS CAPABILITIES **

- HIGH PERFORMANCE GRAPHICS
- TEKTRONIX 4010 COMPATIBILITY
- AUTOMATIC SCALING (1023x1023) WITH 250x512 RESOLUTION
- ALPHA MODE (35 LINES x 73 COLS)
- OPTIONAL JOYSTICK
- CONNECTION TO LOW COST PRINTER FOR GRAPHICS HARDCOPY
- ASCII AND APL CHARACTER SETS
- BLOCK FILL, DOTTED/DASHED LINES

** DISPLAY TERMINAL CAPABILITIES **

- ANSI STANDARD CONFORMANCE
- DEC SOFTWARE COMPATIBILITY
- 80/132 COLUMNS, WINDOWING
- 4 PAGES OF MEMORY STANDARD(UP TO 8)
- 46 PROGRAMMABLE FUNCTIONS
- ASCII AND APL MODELS
- COMPACT ULTRA-THIN KEYBOARD
- HIGH RESOLUTION AMBER PHOSPHOR

High performance graphics at a new low price!

$1190*

CONCEPT GVT+™ Graphics Display Terminal

*Small quantity price

human designed systems, inc.

Whether used in video display mode or in its high-performance graphics mode, Human Designed Systems' GVT+™ Graphics Display Terminal offers more user friendliness, more design features, and more advanced functionality to optimize productivity — and encourage creativity — for the terminal operator, interactive user, and applications developer than any other terminal available today.

Atlanta — (404) 391-9763; Boston — (617) 449-6446; Chicago — (312) 825-2960; Dallas — (214) 437-1888; Delaware — Infocon: (302) 239-2942; Denver — (303) 469-1953; Detroit — (313) 471-2807; Hawaii — Gray Associates: (808) 261-3751; Houston — (713) 952-1403; Los Angeles — (213) 410-9454; Northern New Jersey — Infocon: (201) 624-1372; New York City Area — Infocon: (212) 689-8833; New York State — Naco Electronics: Rochester: (716) 223-4490; Syracuse: (315) 699-2651; San Francisco — (415) 692-4184; Washington, DC — International Systems Marketing: (301) 279-5775; Argentina — Itron SA: (01) 774-9369; Australia — Computer Clarity Pty. Ltd.: (02) 241 3385; Belgium — BELCOMP: 091-31.52.22; Canada — CAIL Systems: Toronto: (416) 362-1063; Denmark — ADCOM Data Aps: 1-19 44 66; Finland — Valtamatic Oy: 0-742 011; France — Walton: (1) 226.06.90; Japan — Ampere: 03 (365) 0825; Portugal — Soc. Com. Crocker, Deliaforce Co. Ltd.: 1-6801 41; Singapore — DTS Singapore: 33-88-566; South Africa — Psitec (Pty.) Ltd.: (11) 836-9181; Switzerland — Mitek ag: 01/461 22 52; United Kingdom — Shandell Systems Ltd.: 2407-2027; Venezuela — H. Blohm SA: 2 541.21.22; West Germany —COMKO Computersystemges, mbH: 221-48 30 51. INTERNATIONAL DISTRIBUTORSHIP INQUIRIES INVITED.

GVT+ is a trademark of Human Designed Systems, Inc.

Circle No. 234 on Inquiry Card

agreement coincides with Burroughs' unveiling of the XE 550, a new MC68000, UNIX-based supermicro designed for multiuser office environments.

Under the exclusive agreement, INGRES will be the only third-party relational database management system that Burroughs will market directly on the XE 550. Burroughs has also assumed responsibility for supporting the software. Pricing will be $6000 for a single application processor version, and $9000 for a multiple application processor version.

This agreement follows the recent announcement by AT&T to market and support INGRES on the 3B computer series. In addition, RTI will maintain agreements with Computervision, Sun Microsystems, NCR and several other hardware manufacturers and distributors who currently market and support INGRES on their equipment. RTI also will continue to market INGRES directly to the VAX/VMS and VAX/UNIX marketplace.

Circle No. 228 on Inquiry Card

MORROW SETS PRICES ON PIVOT PORTABLE

Morrow has begun shipments of its Pivot portable computer, a 10-pound, battery-operated system that runs many popular IBM PC programs. Initial production units are dual-floppy models with 256K bytes of RAM memory,priced at $2895 retail.

Morrow also announced that a dual-floppy version with 640K bytes of RAM is available for $3695, and a single-floppy, 128K byte version priced at $1995 will begin shipping in January.

Besides the Pivot's originally announced standard features — clock, built-in 300 baud modem, four-function calculator and word processing software — the system now includes previously unannounced features. A personal ap-

Circle No. 226 on Inquiry Card

pointment scheduler allows the user to enter appointments for any date, setting an alarm that will remind him of the appointment five minutes before it begins. Also, a personal phone directory with auto dialer software stores and dials more than 100 telephone numbers.

For users interested in high-speed performance, the 640K, two-drive version will incorporate a RAM disk feature, in which ROM-based software permits the user to have part of his RAM memory appear to the system as if it were floppy-disk memory.

Pivot options include a rechargeable battery which provides five hours of additional power ($79), 128K memory expansion ($399), 512K memory expansion ($1299), factory-installed disk drive ($549), Cordura nylon carrying case ($69) and leather carrying case ($129).

Other planned options are a battery-operated printer and an acoustic coupler.

Morrow's Pivot, first announced in May 1984, is a full-featured 16-bit microcomputer system which is based on the low-power CMOS 80C86 microprocessor and uses the industry-standard MS-DOS 2.11 operating system.

The 13 x 5.6 x 9.5-inch unit has a 16-line x 80-column LCD display with 480 x 128-pixel bit-mapped graphics and a 25-line virtual screen. Its keyboard has 63 programmable keys, 10 membrane function keys and a 22-key calculator overlay.

Pivot features an IBM PC-compatible parallel port, an RS232C serial port with programmable baud rates from 300 to 19.2K, and an RJ11C port to connect the built-in modem to ordinary telephone lines. The system's double-sided, double-density 5 1/4-inch floppy drives use IBM PC nine-sector format and store up to 360K bytes (formatted) each.

Circle No. 224 on Inquiry Card

SIR/DBMS NOW RUNS UNDER UNIX-BASED OPERATING SYSTEMS

SIR, Inc., has made SIR/DBMS, its fully relational database management system, available for four UNIX-based operating systems.

SIR/DBMS now operates under the following UNIX 4.2 BSD, the DEC VAX, Hewlett-Packard's HP-UX and Data General's DG MV/UX. SIR/DBMS also operates Apollo's AUX. Conversion for the Sun is underway.

Designed as a set of tools for information analysts in business, industry, and government, SIR/DBMS is especially useful for handling large volumes of complex data. It supports relational, hierarchical and network views of data.

Included in the SIR/DBMS package is SQL+, an expansion of IBM's Structured Query Language. SQL+ is an interactive relational query system that lets users of SIR/DBMS interrogate their databases using English

language commands. It automatically displays the information requested. And because it takes advantage of the existing structure of the database, it performs retrievals very efficiently.

The high-level programming language of SIR/DBMS enables development of tailor-made applications. Other features include: an active data dictionary for comprehensive data integrity and quality control; direct interfaces with BMDP, SAS and SPSS statistical software packages; and flexible report generation including publication-quality tabular display.

SIR/DBMS also includes the following components: FORMS for interactive screen-oriented data entry and query-by-forms; HOST, a language interface for access to one or more SIR DBMS databases; HELP for on-line documentation and user assistance; and GRAPH for production of scientific and business graphics.

Circle No. 220 on Inquiry Card

Announcing . . .
PROFLEX™

A comprehensive integrated multi-user accounting and management information system. Written entirely in the Progress™ database management language from Data Language Corporation. Easily modified source code provided.

Accounting Modules: Cash Management, General Ledger, Accounts Payable, Accounts Receivable, Job Costing, Fixed Assets, Orders and Inventory, Payroll.

Management Modules: Administration and Operations (including tasks, documents, scheduling, dispatching, activities, communications), Sales, Rentals, Personnel, Investments, Trusts, Taxes.

Professional Modules: Medical, Legal, Accountant.

Look for engineering, scientific, and statistical modules to come.

Also available: NIAL AI language on Fortune 32:16. Better than LISP or Prolog.

STARWOOD GROUP, 17707 Blanco, San Antonio, TX 78232, 512/592-7735

Circle No. 221 on Inquiry Card

Fact:
Cardiovascular diseases will cause half of all deaths this year.

American Heart Association
WE'RE FIGHTING FOR YOUR LIFE

UNIX™ VALIDATION

formerly **THE SYSTEMS GROUP**
PERENNIAL

Validation Suite versions available

- UNIX System III and System V (tm Bell Labs)
- BSD 4.1 and BSD 4.2
- /usr/group Standard
- XENIX (tm Microsoft)

Validation Suites Include

- Source code for entire suite
- Template for adding tests
- Installation support service
- User documentation
- Audit trail for all tests
- Test driver control program
- System exercisor—stress tests
- Automatic and manual tests
- Touch tests of all utilities
- Functional tests for all system calls

***PERENNIAL* SOFTWARE SERVICES GROUP**
3130 De La Cruz Blvd. • Santa Clara, California 95054 • (408) 727-2255
SOFTWARE DEVELOPMENT, TESTING, MAINTENANCE AND SUPPORT SERVICES

Circle No. 222 on Inquiry Card

CALENDAR

EVENTS

JANUARY 1985

January 21-25 UniForum, The International Conference of UNIX Users, Dallas, TX. Contact: UniForum/Professional Exposition Management Co., 2400 East Devon Avenue, Des Plaines, IL 60018. 800/323-5155, or in IL, 312/299-3131.

APRIL 1985

April 24-26 UNIX Systems Expo/85, San Francisco, CA. Contact: Computer Faire, Inc., 181 Wells Avenue, Newton, MA 02159. 617/965-8350.

TRAINING

DECEMBER

December 3 AT&T Technologies, Lisle, IL & Sunnyvale, CA: "UNIX System V Internals." Contact: AT&T Technologies, Corporate Education & Training, PO Box 2000, Hopewell, NJ 08525. 800/221-1647.

December 3 AT&T Technologies, Sunnyvale, CA: "C Language for Programmers." Contact: AT&T (see December 3).

December 3 AT&T Technologies, Sunnyvale, CA: "Shell Command Language for Programmers." Contact: AT&T (see December 3).

December 3 NCR Education Seminar, Chicago, IL, Dallas, TX & Dayton, OH: "UNIX Operating System." Contact: NCR Customer and Support Education, 101 W. Schantz Ave., Dayton, OH 45479. 513/445-3905.

December 3-4 Computer Technology Group, Los Angeles, CA: "Shell Programming." Contact: CTG, Telemedia, Inc., 310 S. Michigan Ave., Chicago, IL 60604. 800/323-UNIX, or in Illinois, 312/987-4082.

December 3-7 Bunker Ramo Information Systems, Trumbull, CT: "Advanced UNIX." Contact: Bunker Ramo, Trumbull Industrial Park, Trumbull, CT 06609. 203/386-2223.

December 3-7 Computer Technology Group, Boston, MA & Washington, DC: "C Language Programming." Contact: CTG (see December 3-4).

December 3-7 Plum Hall Training, Maui, HI: "C Programming Workshop." Contact: Plum Hall, 1 Spruce Avenue, Cardiff, NJ 08232. 609/927-3770.

December 4 Office Computer Interface, Scotts Valley, CA: "Survivor's Guide to **vi**." Contact: OCI, 805 S. Fremont Avenue, Alhambra, CA 91803. 818/289-4433.

December 4 Productivity Products International, Boston, MA: "Managers' Introduction to Object Oriented C Programming." Contact: PPI, 27 Glen Road, Sandyhook, CT 06482. 203/426-1875.

December 4-7 Integrated Computer Systems Seminar, Palo Alto, CA: "Hands-On UNIX Workshop." Contact: ICS, 6305 Arizona Place, Los Angeles, CA 90045. 800/421-8166, or in CA, 800/352-8251.

December 4-7 Integrated Computer Systems Seminar, Washington, DC: "Programming in C: A Hands-On Workshop." Contact: ICS (see December 4-7).

December 5-7 CAPE Seminar, Chicago, IL: "A User-Oriented Evaluation Three-Day Seminar: UNIX." Contact: Center for Advanced Professional Education, 1820 East Garry St., Suite 110, Santa Ana, CA 92705. 714/261-0240.

December 5-7 Computer Technology Group, Los Angeles, CA: "Using Advanced UNIX Commands." Contact: CTG (see December 3-4).

December 5-7 Digital Seminar Program, Los Angeles, CA: "Comprehensive Overview of the UNIX Operating System." Contact: Digital Educational Services, 12 Crosby Drive, Bedford, MA 01730. 617/276-4949.

December 5-7 Productivity Products International, Boston, MA: "Programming with Objective C." Contact: PPI (see December 4).

December 8 International Technical Seminars, San Francisco, CA: "User Friendly **awk**" with Jim Joyce; "Using the C Shell Effectively" with Jim Joyce; "UNIX for Programmers" with Steve Bourne. Contact: ITS, 520 Waller Street, San Francisco, CA 94117. 415/621-6415.

December 9 International Technical Seminars, San Francisco, CA: "C by Example" with Jim Joyce; "Problem Solving with the Shell" with Mark Sobell. Contact: ITS (see December 8).

December 10 AT&T Technologies, Lisle, IL & Princeton, NJ: "C Language for Programmers." Contact: AT&T (see December 3).

CALENDAR

December 10 AT&T Technologies, Sunnyvale, CA: "Fundamentals of the UNIX Operating System for Programmers." Contact: AT&T (see December 3).

December 10 AT&T Technologies, Sunnyvale, CA: "UNIX System V Device Drivers." Contact: AT&T (see December 3).

December 10 NCR Education Seminar, Dallas, TX: "UNIX System Administration." Contact: NCR (see December 3).

December 10 NCR Education Seminar, New York, NY: "C Programming." Contact: NCR (see December 3).

December 10-11 Computer Technology Group, Boston, MA & Washington, DC: "Shell Programming." Contact: CTG (see December 3-4).

December 10-11 Intelligent Solution Seminar, Boston, MA: "UNIX Concepts." Contact: Intelligent Solution, 849 22nd Street, Santa Monica, CA 90403. 800/367-0948 or in CA, 213/207-5356.

December 10-12 CAPE Seminar, Somerset, NJ: "A User-Oriented Evaluation Three-Day Seminar: UNIX." Contact: CAPE (see December 5-7).

December 10-14 Bunker Ramo Information Systems, Trumbull, CT: "Advanced C." Contact: Bunker Ramo (see December 3-7).

December 10-14 Computer Technology Group, Los Angeles, CA: "UNIX Internals." Contact: CTG (see December 3-4).

December 10-14 Plum Hall Training, Grand Canyon, AZ: "Advanced C Topics Seminar." Contact: Plum Hall (see December 3-7).

December 10-14 Structured Methods Seminar, New York, NY: "UNIX System Workshop." Contact: Structured Methods, 7 West 18th St., New York, NY 10011. 800/221-9274, or in NY, 212/741-7720.

December 11 Office Computer Interface, Scotts Valley, CA: "Document Preparation Covering **nroff -ms**." Contact: OCI (see December 4).

December 11-14 Integrated Computer Systems Seminar, Palo Alto, CA: Programming in C: A Hands-On Workshop." Contact: ICS (see December 4-7).

December 12-13 Intelligent Solution Seminar, Boston, MA: "Programming in C." Contact: Intelligent Solution (see December 10-11).

December 12-14 Computer Technology Group, Boston, MA & Washington, DC: "Using Advanced UNIX Commands." Contact: CTG (see December 3-4).

December 12-14 Digital Seminar Program, Los Angeles, CA: "The C Programming Language." Contact: Digital Educational Services (see December 5-7).

December 14 Intelligent Solution Seminar, Boston, MA: "UNIX Overview." Contact: Intelligent Solution (see December 10-11).

December 17 AT&T Technologies, Dublin, OH:

CROMEMCO COMPUTERS: DESIGNED TO MAKE UNIX SYSTEM V EVEN BETTER...

UNIX System V, the new standard in multi-user microcomputer operating systems, gives you high performance features along with the portability and flexibility of a standard.

Cromemco computers can make UNIX System V even better. Because our systems are designed with UNIX in mind. First of all, we offer UNIX System V with Berkeley enhancements. Then, our hardware uses advanced features like 64K of on-board cache memory and our high speed STDC controller to speed up disk operations – very important with UNIX.

More capability and expandability

We have a high-speed, 68000-based CPU that runs at 10 MHz, coupled with a memory manager that uses demand-paging and scatter loading to work *with* UNIX, not for it.

We provide room for expanding RAM to 16 megabytes – with error detection and correction – for running even the most sophisticated and advanced microcomputer programs. And the power to accommodate up to 16 users – all with plenty of memory.

But we give you even more.

A complete solution

We give you a choice in systems: the System 100 series, expandable up to 4 megabytes of RAM, and the System 300 series, expandable to 16 megabytes. A high speed 50 megabyte hard disk drive is standard on the systems. And you can expand the hard disk capacity up to 1200 megabytes using standard SMD drives. You can add floating point processing. High resolution graphics. Video digitizing and imaging. Communications through standard protocols. Mainframe interface.

And software support is here to meet your needs. We offer major programming languages, database management systems, communications software, including SNA architecture, X.25 protocol, and Ethernet; even a program to interface to an IBM PC if you need to. And, of course, access to the broad range of standard UNIX applications programs that is growing dramatically every day.

Easy to use.

We also make our systems easier to use, because we install the operating system before we ship your computer. No complicated installation procedures. And the Berkeley enhancements give you the standard UNIX System V operating system, but with the added convenience of these widely acclaimed improvements.

Cromemco's System 100 and System 300 computers: designed to be the highest performance UNIX systems available anywhere.

Just call or visit one of our UNIX System V Official System Centers to see for yourself. They'll also give you a copy of our new publication, "What you should know before you buy a UNIX system." Or contact us directly.

We'll be glad to show you how to get a better UNIX system.

Corporate Headquarters: Cromemco, Inc., 280 Bernardo Avenue, P.O. Box 7400, Mountain View, CA 94039. (415) 969-4710. In Europe: Cromemco GmbH, 6236 Eschborn 1, Frankfurter Str. 33-35, P.O. 5267, Frankfurt Main, Germany.

UNIX is a trademark of Bell Laboratories.
IBM is a trademark of International Business Machines Corp.

Cromemco®

Visit us at UniForum Booth #2236

Circle No. 248 on Inquiry Card

"UNIX System V Administration." Contact: AT&T (see December 3).
December 17 AT&T Technologies, Sunnyvale, CA: "Internal UNIX System Calls and Libraries Using C Programming." Contact: AT&T (see December 3).
December 17 NCR Education Seminar, Dayton, OH: "UNIX System Administration." Contact: NCR (see December 3).
December 17-19 Structured Methods Seminar, New York, NY: "UNIX System Internals." Contact: Structured Methods (see December 10-14).
December 17-21 Bunker Ramo Information Systems, Trumbull, CT: "C Programming." Contact: Bunker Ramo (see December 3-7).
December 17-21 Computer Technology Group, Boston, MA & Washington, DC: "UNIX Internals." Contact: CTG (see December 3-4).
December 18 Office Computer Interface, Scotts Valley, CA: "Practical UNIX for Secretaries." Contact: OCI (see December 4).
December 18-20 Computer Technology Group, Los Angeles, CA: "UNIX Administration." Contact: CTG (see December 3-4).

December 19-21 CAPE Seminar, Tampa, FL: "A User-Oriented Evaluation Three-Day Seminar: UNIX." Contact: CAPE (see December 5-7).
December 20 AT&T Technologies, Lisle, IL: "Overview of the UNIX System." Contact: AT&T (see December 3).
December 20-21 Structured Methods Seminar, New York, NY: "Using **Lex** and **yacc**." Contact: Structured Methods (see December 10-14).

JANUARY 1985

January 7 AT&T Technologies, Princeton, NJ: "Shell Command Language for Programmers." Contact: AT&T (see December 3).
January 7 AT&T Technologies, Sunnyvale, CA: "Internal UNIX System Calls and Libraries Using C Programming." Contact: AT&T (see December 3).
January 7 AT&T Technologies, Lisle, IL: "UNIX System V Internals." Contact: AT&T (see December 3).
January 7-11 Plum Hall Training, Raleigh, NC: "UNIX Workshop." Contact: Plum Hall (see December 3-7).
January 9 AT&T Technologies, Sunnyvale, CA: "Fundamentals of the UNIX Operating System for Users." Contact: AT&T (see December 3).
January 9-11 Digital Seminar Program, Boston, MA: "Comprehensive Overview of the UNIX Operating System." Contact: Digital Educational Services (see December 5-7).
January 14 AT&T Technologies, Princeton, NJ & Sunnyvale, CA: "C Language for Programmers." Contact: AT&T (see December 3).
January 14 AT&T Technologies, Lisle, IL & Sunnyvale, CA: "Fundamentals of the UNIX Operating System for Programmers." Contact: AT&T (see December 3).
January 14 AT&T Technologies, Sunnyvale, CA: "UNIX System V Device Drivers." Contact: AT&T (see December 3).
January 14 AT&T Technologies, Sunnyvale, CA: "UNIX System V Internals." Contact: AT&T (see December 3).
January 14-18 Structured Methods Seminar, New York, NY: "UNIX System Workshop." Contact: Structured Methods (see December 10-14).
January 16-18 Digital Seminar Program, Boston, MA: "The C Programming Language." Contact: Digital Educational Services (see December 5-7).
January 16 AT&T Technologies, Lisle, IL: "Shell Command Language for Users." Contact: AT&T (see December 3).
January 21 AT&T Technologies, Lisle, IL: "C Language for Programmers." Contact: AT&T (see December 3).
January 21 AT&T Technologies, Princeton, NJ:

"UNIX System V Internals." Contact: AT&T (see December 3).

January 21-22 Intelligent Solution Seminar, Baltimore, MD: "UNIX Concepts." Contact: Intelligent Solution (see December 10-11).

January 21-25 Digital Seminar Program, Bedford, MA: "Programming in C." Contact: Digital Educational Services (see December 5-7).

January 21-25 Plum Hall Training, Raleigh, NC: "C Programming Workshop." Contact: Plum Hall (see December 3-7).

January 21-25 Structured Methods Seminar, New York, NY: "C Programming Workshop." Contact: Structured Methods (see December 10-14).

January 21-25 Structured Methods Seminar, New York, NY: "Advanced Topics in The C Language." Contact: Structured Methods (see December 10-14).

January 22-25 Integrated Computer Systems Seminar, Baltimore, MD: "Hands-On UNIX Workshop." Contact: ICS (see December 4-7).

January 23-24 Intelligent Solution Seminar, Baltimore, MD: "Programming in C." Contact: Intelligent Solution (see December 10-11).

January 25 Intelligent Solution Seminar, Baltimore, MD: "UNIX Overview." Contact: Intelligent Solution (see December 10-11).

January 28 AT&T Technologies, Sunnyvale, CA: "Shell Command Language for Users." Contact: AT&T (see December 3).

January 28 AT&T Technologies, Lisle, IL & Sunnyvale, CA: "UNIX System Document Preparation Utilities." Contact: AT&T (see December 3).

January 28-February 1 Digital Seminar Program, Dallas, TX: "Programming in C." Contact: Digital Educational Services (see December 5-7).

January 28-February 1 Plum Hall Training, Raleigh, NC: "Advanced C Topics Seminar." Contact: Plum Hall (see December 3-7).

January 29-February 1 Integrated Computer Systems Seminar, Baltimore, MD: "Programming in C: A Hands-On Workshop." Contact: ICS (see December 4-7).

January 30 AT&T Technologies, Princeton, NJ: "Fundamentals of the UNIX Operating System for Programmers." Contact: AT&T (see December 3).

January 31 AT&T Technologies, Sunnyvale, CA: "UNIX System V Screen Editor **vi**." Contact: AT&T (see December 3).

Please send announcements about training or events of interest to: UNIX Review Calendar 520 Waller Street San Francisco, CA 94117.

Electronic mail can be sent to ucbvax!olympus!its!dave. Please include sponsor, date and location of event, address of contact and relevant background information.

THE LAST WORD

Letters to the Editor

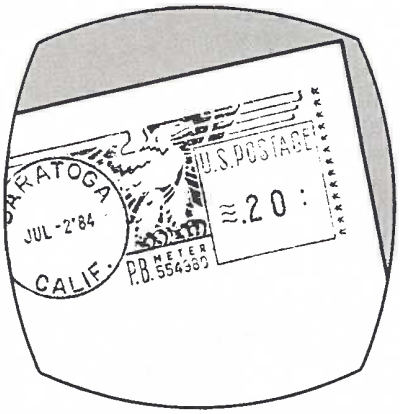
Dear UNIX REVIEW,

Michael Donohue ("Vexations of the Sexist Filter," August 1984) writes of linguistic sexism with the informed sensitivity of an untutored and neutered deaf-mute.

Mr. Donohue senses the current changes in American English (e.g. "policeperson") to be "painful," "unfamiliar" and "downright uncomfortable." These are the same sort of sentiments, actually, that a woman experiences upon being addressed as "chairman," "salesman," "policeman" and — I'll grant him — "host." There is nothing offensive in a "gender-specific turn of phrase" if the gender specified is TRUE. One need not *point* to the "daughter" among the sons, for example. However, the thoughtless comfort we experience with such structures as: "The Chairman of the Board," "The student...he..." and "The Vice-President and Mrs...," is rooted deeply in sexism. It is historic. It is blatant and easily researched in early English Grammar textbooks:

> ...let us kepe (sic) a natural order, and set the man before the woman...
>
> Wilson, 1560

The Masculine gender is



> more worthy than the Feminine.
>
> Poole, 1646

> ...man is the larger, the stronger, the more individually important...
>
> White, 1886

Mr. Donohue recommends some type of tormented legislation to assuage his linguistic hesitation and rash. Such stuff has been attempted: in 1850, an Act of Parliament legally replaced "he or she" with "he." It seems not to have caught on.

American English, in a flux and interwoven, will evolve to reflect our cultural truths. No offense can be found in "programmer/analyst" without adding an adjective. But, as my "mailman" wears lipstick, I find myself seeking an honest alternative.

Yours,

N.P. Evers
Dept. of English
University of Nebraska
Lincoln, NE

Dear UNIX REVIEW,

In the article "Keeping Your System Healthy" (Joyce and Merritt, September 1984), you offer a shell script for finding old "moldy" files named *moldy.oldies*. It looked like a useful script, so I entered it on my system. Imagine my surprise when I found it didn't work. What gives?

Also, can you point me in the direction of a good users' group?

Yours,

Gary Falsken
San Francisco, CA

The horror. The shame.

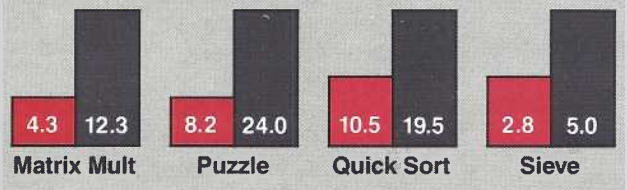
Yes, you're quite right. Two lines were inadvertantly left out of that script. The corrected shell script follows.

As for user groups, the best one I can think of in the Bay Area operates under the name of "Silicon Valley Net." Meeting times for the group are published monthly in the UNIX REVIEW

The TARTAN C Compiler for VAX/UNIX. It's better than free.

TARTAN's C out-performs the portable C compiler you get free with UNIX™. With TARTAN C, you'll see: Faster execution speed, smaller-size object code, unsurpassed error diagnostics, and complete plug-compatibility with portable C. TARTAN C Compilers are part of a new generation of compilers that improve application performance and make programmers more productive.

TARTAN C Benchmarks



Standard benchmark tests* compare TARTAN C (in red) and Portable C compiler object code performance (measured in seconds) for various computational profiles.

TARTAN C Error Messages

C Program with Syntax Errors:

```
1  bubble(a) int a[100]; {int tmp,last,i;
2   for (last=100; last >= 2; last--)
3      {for (i=1; i <= (last-1; i++)
4          if (a[i] > a[i+1])
5             {tmp=a[i] a[i]=a[i+1];
6              a[i+1]=tmp;};
7      }
8  }
```

Portable C Compiler Error Messages:

```
"bubble.c", line 3: syntax error
"bubble.c", line 3: syntax error
"bubble.c", line 8: syntax error
```

TARTAN C accurately pinpoints errors and recovers:

```
   2| for (last=100; last >= 2; last--)
   3|    {for (i=1; i <= (last-1; i++)
                              ^1
***  1 Error 101: Parse error; token ")" inserted.
   4|         if (a[i] > a[i+1])
   5|            {tmp=a[i] a[i]=a[i+1];
                         ^1
***  1 Error 101: Parse error; token ";" inserted.
   6|             a[i+1]=tmp;};
```

*A complete report on these and other benchmarks is available on request.

UNIX is a trademark of AT&T Bell Laboratories

VAX is a trademark of Digital Equipment Corporation.

Get More Performance Into Your Applications.

TARTAN compilers make your programs more competitive. With TARTAN C, you'll see:

- Optimized object code that makes your programs run up to three times faster
- Object code that's up to 55% smaller

Get More Productivity Out of Your Programmers.

TARTAN's object code is optimized by the compiler, not the programmer. Our advanced diagnostics catch errors the first time—in context and with complete, concise messages. With TARTAN C, you'll see:

- No more hand optimizing
- Fewer recompilations
- Error-free code before you know it
- Optional runtime array bounds checking

There's More.

With TARTAN C, you'll see responsive product support and detailed documentation. We also include a copy of the brand new *C: A Reference Manual* by TARTAN's Sam Harbison and Guy Steele Jr.

Call Today.
Order The C Compiler That's Better Than Free.

TARTAN C Compilers for VAX™ systems running Berkeley UNIX have successfully completed customer-site testing and are ready for immediate delivery. To place your order, call our Sales Department at **412-621-2212**, or write us at the address below.

The world's best-known C compiler is free. The world's best-performing C compiler isn't, but it's worth a lot more. The TARTAN C Compiler for VAX/UNIX.

We want to be your favorite software company.

TARTAN

TARTAN Laboratories
Incorporated
477 Melwood Avenue
Pittsburgh, PA 15213

Circle No. 262 on Inquiry Card

```
:
# moldy.oldies -- an administrative script to be run weekly to
# search each user's file space for files that have not been ac-
# cessed in over 90 days. (c) 1984 by International Technical Sem-
# inars, Inc. 520 Waller St., San Francisco, CA 94117 (415) 621-
# 6415. Permission to copy is granted if this copyright notice is
# included in its entirety. Adapted from the /bin/calendar script
# by Doug McIlroy of Bell Laboratories.
#
PATH=/bin:/usr/bin
sed '
        s/\([^:]*\):.*:\(.*\):[^:]*$/y=\2 z=\1/
' /etc/passwd \
| while
    read x
do
    eval $x
    trap "/bin/rm /tmp/$z.$$; trap '' 0; exit" 0 1 2 3 13 15
    case $z in
    adm)        ;;
    bin)        ;;
    check)      ;;
    daemon)     ;;
    root)       ;;
    rootsh)     ;;
    sys)        ;;
    uucp)       ;;
    uucpadm)    ;;
    who)        ;;
    *)
        if (test -s $y) then
            cd $y
            find . -atime +90 -print 2>/dev/null > /tmp/$z.$$
            if (test -s /tmp/$z.$$) then
                mail $z < /tmp/$z.$$
                mail $z <<++
                Dear $z :
                The next mail message you see will contain
                names of moldy oldies you have not accessed
                in 90 days. Please examine these files to see
                if they can be archived.
                   -- Your Friendly System Administrator
++
            fi
            /bin/rm /tmp/$z.$$
        fi
        ;;
    esac
done
```

The genuine article—The **moldy oldies** *shell script, as corrected from the September issue.*

Calendar. We also invite other groups to submit their meeting schedules to the address listed at the end of the Calendar so that those, too, may be published.

Editor

Dear Editor:

In September UNIX REVIEW, Richard Morin's "The Human Factor" complains about the lack of uniformity in UNIX error messages. That's not the half of it! UNIX diagnostics stink. They are terse, intimidating and obscure.

Naturally, this hits Herman User just when he has demonstrated that he is a little confused about what to type. There may be a manual page online, but UNIX won't say so. There may be a specialist available by phone or mail, but UNIX won't tell. This is a cruel fact of life for UNIX users today. Is there no salvation?

There is. By reasonably well observed convention, error messages appear on **stderr** (file descriptor 2). You can (mostly) count on it. It is enough.

A straightforward modification to your favorite shell would permit an error filter to improve this dismal situation. A sample of such a mod (to the Berkeley C shell) was coded by D. L. Wasley and is available on request. After "set error_filter = command" each command invoked gets its **stderr** piped to the named command. The error filter can then capture each message generated on the machine. You can use the data to select messages to be reworded or made more uniform.

The filter could always offer more information, perhaps as briefly as "More info? (y or n)". More sophisticated filters might use an environment (or shell) variable to decide whether to offer a manual page immediately or to display a consultant's phone number. Indeed, if the particular message had already been entered into the appropriate file, replacement text (or action) could be supplied by the filter from the file.

This behavior is non-trivial to achieve, but the filter only needs to be written once. It applies to every program in your system. You do not need the sources. A simple file of error messages and their replacements is sufficient to translate error messages into French (or English). Other useful behaviors are possible. For exam-

WE'RE NOT JUST LOOKING FOR GENIUS

WE'RE MADE OF IT.

Genius begins with a vision. Convergent was founded on an ingenious idea: new 16-bit microprocessors could be combined with emerging disk technologies to create a unique set of products — multifunction workstations.

It took a lot of gumption to quickly bring out a product no one had seen before. To do it in half the time that most other companies would take. To sell it to the leading OEMs. And to follow it up with seven more in rapid succession. Sales for the first six months of 1984 topped $141 million, already 85% more than for all of 1983. It began with a vision.

Convergent Technologies is looking for **senior level hardware and software professionals** to turn their visions into realities. If you have a background in UNIX, related degree with three plus years experience and can work on a wide range of office automation products... intelligent terminals to super minis, then look to Convergent.

We offer significant equity participation, competitive compensation and a 13-week sabbatical leave after five years. There's a premium for genius.

For immediate consideration send your resume to **Joy Mar, Convergent Technologies, Dept. UR/D, 30 East Plumeria Drive, San Jose, CA 95134.** We are an equal opportunity employer. Principals only, please.

Convergent Technologies

ple, first occurrences of particular messages could cause inquiry into the intent of the user (after congratulating them for their accomplishment).

Readers can write me for the C shell mods. They are free on paper or by way of Bitnet or Usenet. Send $15 for a copy on IBM PC disk. Then pass it around, post it, use it, sell it or ignore it.

I am personally offering a $500 prize for the best simple error filter offered by a reader of this letter. The winning program in its entirety will be considered to be in the public domain. Entries must be received at the address below by February 28, 1985.

Dick Karpinski
6521 Raymond Street
Oakland, CA 94609
415/658-3797

ucbvax!ucsfcgl!ucsfcca!dick
ucsfcca.BITNET

To the Editor:

I am writing in response to the review of *A Book on C*, by Al Kelley and Ira Pohl (Benjamin Cummings 1984). First, I would like to thank the reviewer for his kind remarks in regard to the overall quality of the book. However, crucial points about the book were overlooked.

A Book on C is a comprehensive professional and self-study guide to the language. It includes thorough coverage of such topics as enumeration types and structure assignment, unavailable in other books published before it. Further, it is intended for interactive use and self-study with over 250 exercises.

The reviewer comments that the book "leans a bit heavily in the direction of number crunching." We treat C as a major general purpose language and take examples from all domains, including list processing (17 indexed functions) and an interactive game program using enumeration types. There are over 70 actual working programs, with the majority of these non-numerical. The number and breadth of examples of working code, including scientific algorithms, is one of its features.

While the review is appreciative of the overall quality of the code and programming style, and the specially developed dissection technique, it is explicitly upset with two minor programs. The first, on page 3, is in the introductory remarks and is clearly explained to be obscure and system dependent. Of course it is not intended as the introductory example. It appears as a puzzle to be understood after mastering the text and is later explained in Chapter 9. Without giving away its secrets, I suspect most programmers will be intrigued by this puzzle and challenged to read on.

The review also is unhappy with our first example of recursion, namely:

```
main( )
{
 printf(" The Universe is never
        ending! ");
 main( );
}
```

because it must be terminated by programmer intervention. The use of CTRL-C to interrupt a program is clearly explained in the text. Our attitude is that this is quite proper given modern interactive environments, which is a chief premise in the development of many examples. The remaining section on recursion is comprehensive and includes a detailed explanation of **quicksort** that's fully coded.

While appreciating the kind remarks in the review, it nevertheless falls short in providing an overview of the book's major assets or a critique of its aims to be a comprehensive professional reference and a quality college text.

Yours truly,

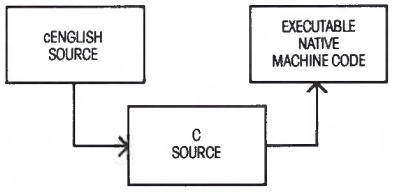
Ira Pohl
Professor of Computer and Information Science
University of California,
Santa Cruz

Letters to the Editor should be addressed to: UNIX REVIEW, 520 Waller Street, San Francisco, CA 94117

c ENGLISH.™
The C Generation Language.

What is cENGLISH? cENGLISH is a comprehensive fourth generation procedural language based on dBASE II™ syntax. It is portable to a wide range of micros and minis. The language features user-transparent interfaces to a wide range of popular C compilers, operating systems, and data base managers.

How is portability achieved? cENGLISH through its compiler interface translates cENGLISH into documented C source and uses a host C compiler to produce native machine code.



C source can be embedded in cENGLISH source.

Differences in the operating system and data base manager are handled by the runtime libraries.

The result is that cENGLISH source can be compiled without modification on any micro or mini configuration supporting cENGLISH.

What about performance? cENGLISH executes FAST, just like any compiled C program.

How easy is cENGLISH to use? While cENGLISH is a powerful high level language that can accommodate complex software development, it remains simple and straightforward to use.

Call or write for availability of cENGLISH for the following configurations—

Compilers:
Standard O/S compilers: Lattice C™ for MS/DOS™

Operating Systems:
UNIX,™ UNIX-like, MS/DOS,™ Coherent,™ VMS™

Data Base Managers:
C-ISAM™ and INFORMIX,™ UNIFY,™ ORACLE,™ PHACT,™ Logix™

Foreign Language Versions:
German, French, Spanish

Attention MS/DOS users. Demo version and special introductory offer available for IBM PC,™ XT,™ AT,™ and other MS/DOS systems. Requirements: 256K, hard disk or two floppy disk drives, and MS/DOS 2.1 or higher.

Attention dBASE II and dBASE III users. dBASE II to cENGLISH Converter now available; dBASE III Converter available later this quarter. Converted code is portable to micros or minis and executes as fast as original cENGLISH source.

SAMPLE cENGLISH PROGRAM

```
IDENTIFICATIONS
  MODULE:  Mininame
  AUTHOR:  bcs
  DATE:    8/29/84
  REMARKS: Sample cENGLISH program that adds first
           names to a file
END IDENTIFICATIONS

GLOBALS
  FIXED LENGTH 1 ans
  FIXED LENGTH 15 Fname
END GLOBALS

MAIN PROGRAM
  BEGIN
    CLEAR SCREEN
    SET ECHO OFF

    USE "NAMES"
    VIEW BY "ID_FNAME" ASCENDING

    AT 23,1 SAY "Add a record? Y or N"
    AT 23,25 ENTER ans USING "!"

    WHILE ans EQ "Y"
      CLEAR GETS
      AT 6,1 SAY "Enter first name"
      AT 6,20 GET Fname
      READ SCREEN

      INSERT
        Fname = Fname
      END INSERT

      AT 12,10 SAY "Welcome to cENGLISH," & Fname
      WAIT
      AT 14,10 SAY "HIT ANY KEY TO CONTINUE"
      STORE "                    " TO Fname
      STORE "  " TO ans
      AT 23,1 SAY "Add another record? Y or N"
      AT 23,30 ENTER ans USING "!"
      CLEAR ROW 1 THRU 23

    END WHILE

    AT 12,10 SAY "That's all for now!"
    UNUSE "NAMES"
    SET ECHO ON

END PROGRAM
```


cLINE™ Inc.

I'd like to know more about cENGLISH. Please send further information.

Your Name _____ Title _____

Company _____ Telephone _____

Address _____

City _____ State _____ Zip _____

Check one: ☐ End User ☐ System House ☐ Dealer ☐ Distributor

Send to: cLINE Inc., 20 West Ontario, Chicago, IL 60610-3809
Telex 516315 Phone (312) 944-4510

In Canada: cLINE Canada, Inc. Complexe La Laurentienne,
425 St. Amable, Suite 165, Quebec, Canada G1R5E4
Phone (418) 524-4641

04Q8

ADVERTISER'S INDEX

COMING UP IN JANUARY

- **Evolution of UNIX**
- **UNIX Origins**
- **The Road Leading from Multix**
- **UNIX Evolution as a Product**
- **The Berkeley to Bell Labs Connection**
- **Workstations in the Future**

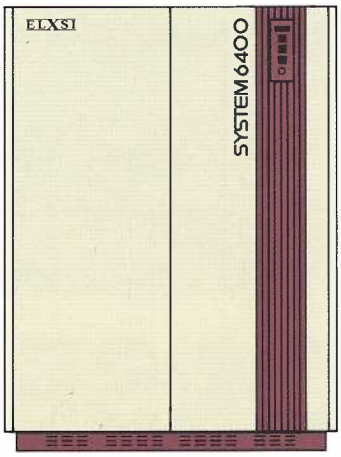
BACK ISSUES AVAILABLE

June/July 1983 . . . . . ☐
August/September 1983 . . . . . ☐
October/November 1983 . . . . . ☐
December/January 1984 . . . . . ☐
February/March 1984 . . . . . ☐
April/May 1984 . . . . . ☐
June 1984 . . . . . ☐
July 1984 . . . . . ☐
August 1984 . . . . . ☐
September 1984 . . . . . ☐
October 1984 . . . . . ☐
November 1984 . . . . . ☐

All back issues are $4.95, including postage and handling. Enclose payment or credit card information or call 206/271-9605.

Name _____
Company _____
Address _____
City _____ State _____ Zip _____
M/C or VISA _____
Exp. Date _____

UNIX MULTIPROCESSOR PERFORMANCE AND EXPANDABILITY IN ONE BOX



NOW

ELXSI System 6400 has a true native port of UNIX System V, not an emulation. With a unique combination of Berkeley enhancements. Virtual memory. Super fast I/O. Ethernet with TCP/IP. Plus the largest physical memory in the business. ELXSI System 6400 will run either UNIX or our proprietary, message-based EMBOS operating system. Or both, concurrently.

ELXSI System 6400: The first true multiprocessor UNIX system. Up to 5 CPU's operating at 20 MIPS in a single cabinet, through the ELXSI Gigabus—our 320 mByte/second, 64-bit wide system bus that can accommodate further expansion up to 10 CPU's and 40 MIPS of computing power. More power for more system users, expandable as your needs grow, with the highest power to footprint ratio in the business.

ELXSI System 6400. No one can match our performance or our expandability, deliverable now. Contact ELXSI today for complete information. Sales offices in most major cities.

THE ELXSI GUARANTEE: find a computer system that can match our performance and our potential for expansion, and we'll raise our price to match theirs.

ELXSI

ELXSI, 2334 Lundy Place, San Jose, California 95131.
408/942-1111, Telex 172-320.

UNIX is a trademark of Bell Laboratories. Ethernet is a trademark of Xerox Corporation. Gigabus and EMBOS are trademarks of ELXSI.

See us at UniForum Booth #2220

Circle No. 242 on Inquiry Card

UNIX™ HORSEPOWER!

There are a lot of UNIX based systems on the market today claiming to be "SUPERMICROS". But do they really have what it takes to run multi-user UNIX well? The IBC ENSIGN™ does and here's why:

FAST MEMORY: No computer running at **any** clock speed can run faster than it's overall memory design. The ENSIGN has up to 8MB of 120nsec memory with dual bit error correction. With IBC's proprietary memory management, **all** of this memory runs with no wait states as fast as the 68000 CPU will go. Compare this to other systems running only small cache memories at full speed. Other multiple user systems cannot load all their programs into a small cache memory. Their systems slow down considerably under a heavy multi-user load.

INTELLIGENT SERIAL I/O CONTROLLER: Even the fastest CPU will slow down when it's trying to handle interruptions from multiple on-line users. The ENSIGN provides slave serial I/O CPU's **and** FIFO buffering for both input and output. The result is the ENSIGN's ability to support up to 32 users, with heavy serial I/O demand, while leaving the main 68000 CPU free to run with little serial I/O overhead.

INTELLIGENT DISK CONTROLLER AND HIGH PERFORMANCE DISK DRIVES: The ENSIGN has a slave CPU to handle all disk operations, **plus** 16K of disk buffering. IBC's proprietary disk DMA allows high speed data transfer to main memory without slowing down the main CPU. Further, the ENSIGN supports SMD type 8" hard disks with much faster seek times and transfer rates than 5¼" hard disks usually found in personal desk top computers.

THE RESULTS: The IBC ENSIGN runs multi-user UNIX at performance levels not attainable by other supermicros.

Call IBC and get a copy of IBC's multi-user benchmarks—benchmarks that test 8 users running large CPU programs, with heavy disk I/O **and** heavy serial I/O simultaneously. You'll find that nothing can compare to the ENSIGN.

If you want to run multi-user UNIX on a high performance system with up to 32 users, 8MB memory, and over 1,000MB disk storage, see the IBC ENSIGN.

IBC/Integrated Business Computers
21621 Nordhoff Street
Chatsworth, CA 91311
(818) 882-9007
Telex No. 215349

UNIX is a trademark of Bell Laboratories

Circle No. 241 on Inquiry Card