THE ABCs OF BERKELEY UNIX SYSTEMS

UNIX*/WORLD

YOUR GUIDE TO THE FUTURE OF MULTIUSER COMPUTING

MAY 1985 $3 IN USA UK £2.20 A TECH VALLEY PUBLICATION

HORIZON'S ANSWER TO LOTUS' SYMPHONY

SYSTEM ADMINISTRATION

CURES FOR AILING BUSINESSES

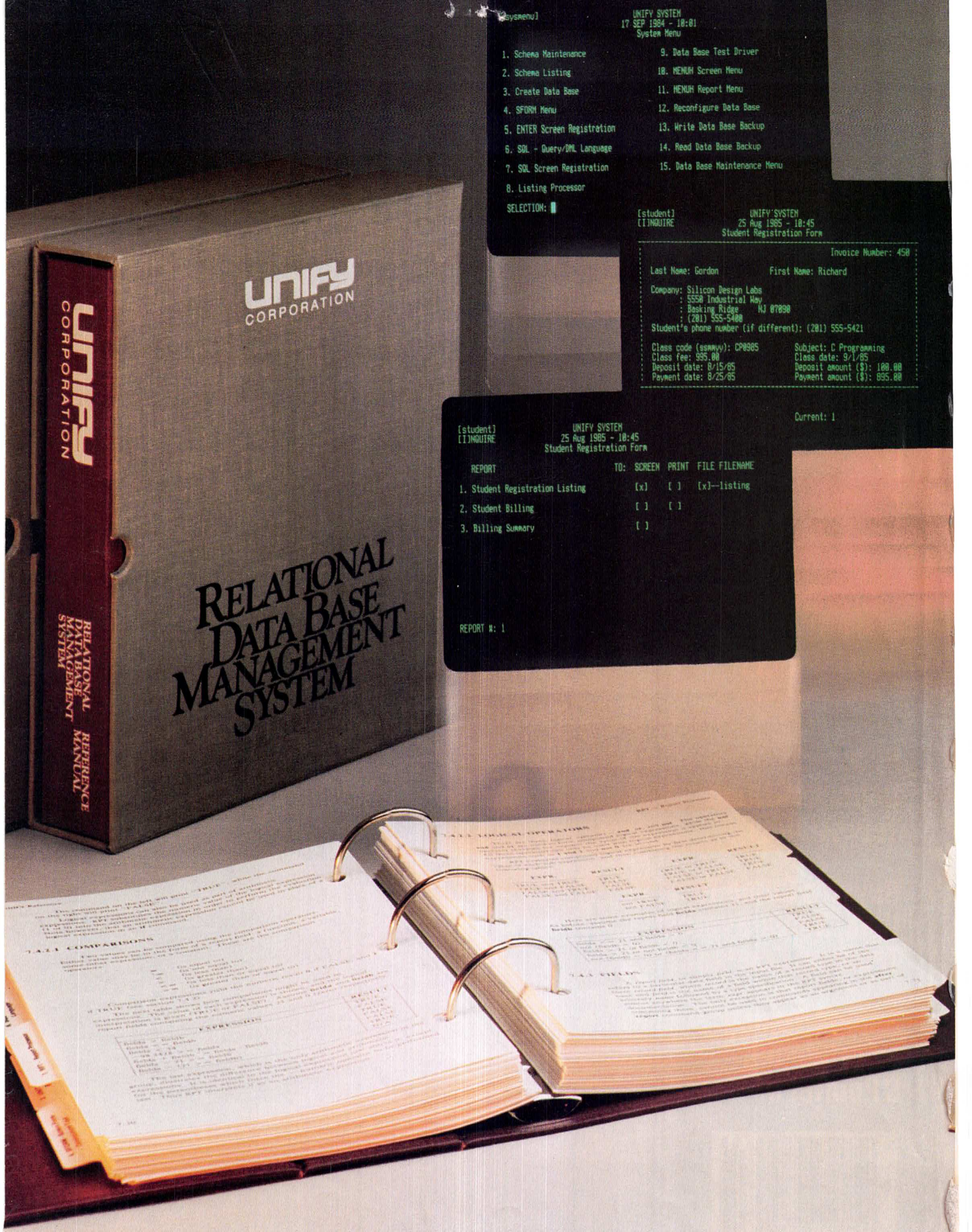
PCs: Consider Them Terminal(s)

Buyer's Guide To Unix, Xenix System Training Courses

Spring-Tuning Your Kernel, C Pointers

0 71572 04782 05

*UNIX is a trademark of AT&T Bell Laboratories. UNIX/WORLD Magazine is not affiliated with AT&T Bell Laboratories.

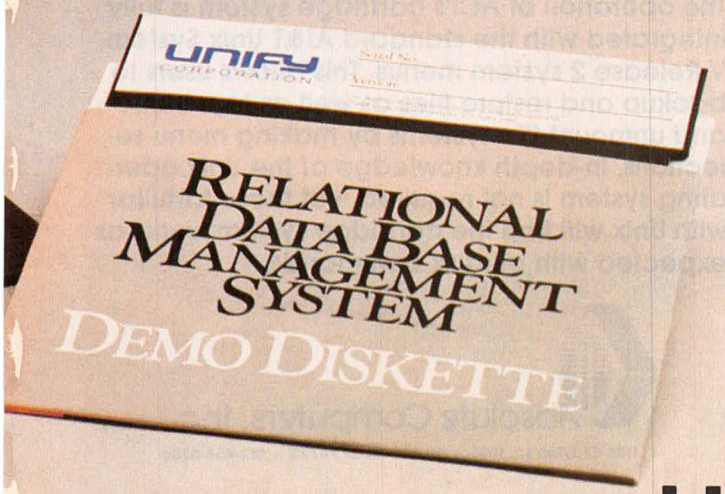


*UNIX is a trademark of AT&T Bell Laboratories.
UNIFY is a trademark of Unify Corporation.

UNIFY. THE MIGHTY EASY DBMS.

UNIFY isn't only the fastest and most powerful of all UNIX*-based data base management systems—it's also one of the simplest to use. Which is why some 75 percent of those who see our manuals and buy a DBMS, buy UNIFY.

UNIFY guides the nonprogrammer through data base development with minimal steps, comprehensive menus, on-line HELP, elementary Query By Forms capability, and clear documentation.



UNIFY expedites applications development for users of all skill levels with features like PAINT, for effortless forms design; SQL, the powerful, English-like query language; and a menu-handler that lets you quickly compile screens, queries and reports into menus customized to each user.

And for the skilled programmer, UNIFY offers the most extensive host language interface, for limitless flexibility when you need it.

All of which makes it mighty easy to see why UNIFY has been selected by more computer manufacturers than any other UNIX-based DBMS.

Judge for yourself. Send for our demo kit—disks, tutorial and reference manuals, all for only $150—that shows you how to build virtually any application. Contact UNIFY, 4000 Kruse Way Place, Lake Oswego, OR 97034, 503/635-6265.

Please circle Ad No. 63 on inquiry card.

unify®

THE PREFERRED DBMS.

High Speed Backup/Mass Storage for the AT&T 3B2 Computer

ACI's 10 megabyte cartridge system provides dual purpose expansion for the AT&T 3B2. This cartridge system provides the fastest 3B2 backup available and also, unlike streaming tape drive storage systems, provides extra mass storage as a mountable disk drive.

As a backup device, ACI's cartridge system allows complete multi-volume backup. Single or multiple files may be saved to, and restored from, removable cartridges. Individual files are easily restored from multi-volume backups.

As a mountable disk under the Unix operating system, the cartridge system allows users to create directories on removable cartridges as desired. In a multi-user environment, administrators may find it useful to allow users their own cartridges, enabling them to have as much disk space as they need in 10MB increments.

In certain situations, the 3B2 user may desire greater storage capacity than the standard 32 MB of the 3B2. ACI's expansion system has a capacity of up to 240 MB of unformatted 24 ms fixed disk storage.

The operation of ACI's cartridge system is fully integrated with the standard AT&T Unix System V Release 2 system menus. This allows users to backup and restore files as well as to mount and unmount file systems by making menu selections. In-depth knowledge of the Unix operating system is not required, yet those familiar with Unix will find the cartridge system works as expected with all disk commands.

Absolute Computers, Inc.

1106 Clayton Ln. 108E • Austin, Texas 78723 • 512-458-5206

3B2 is a trademark of AT&T Technologies
UNIX is a trademark of AT&T Bell Laboratories

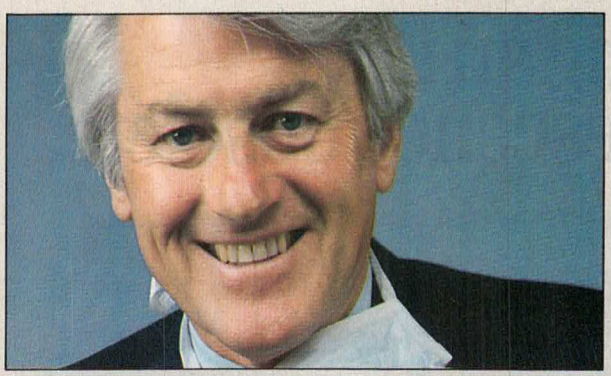
Please circle Ad No. 41 on inquiry card.

UNIX/WORLD

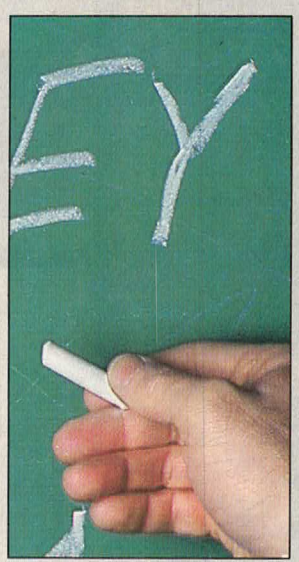
VOLUME II, NUMBER 4

CONTENTS

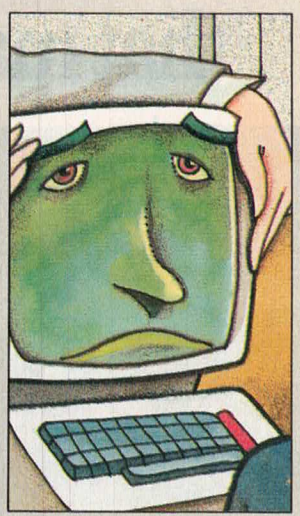
A TECH VALLEY PUBLICATION



THEME



FEATURES



Only Sperry can make the following four statements.

Our PC runs the XENIX™ system, as well as MS-DOS™.

Our 4 new microcomputers run the UNIX system.

Our new minicomputer runs the UNIX system.

Our Series 1100 mainframes run the UNIX system.

All of which means there is a great deal we can do for you.

For instance, our family of computers based on UNIX systems has incredible transportability for all your software.

And being able to accommodate from two to hundreds of users, it's impossible to outgrow our hardware.

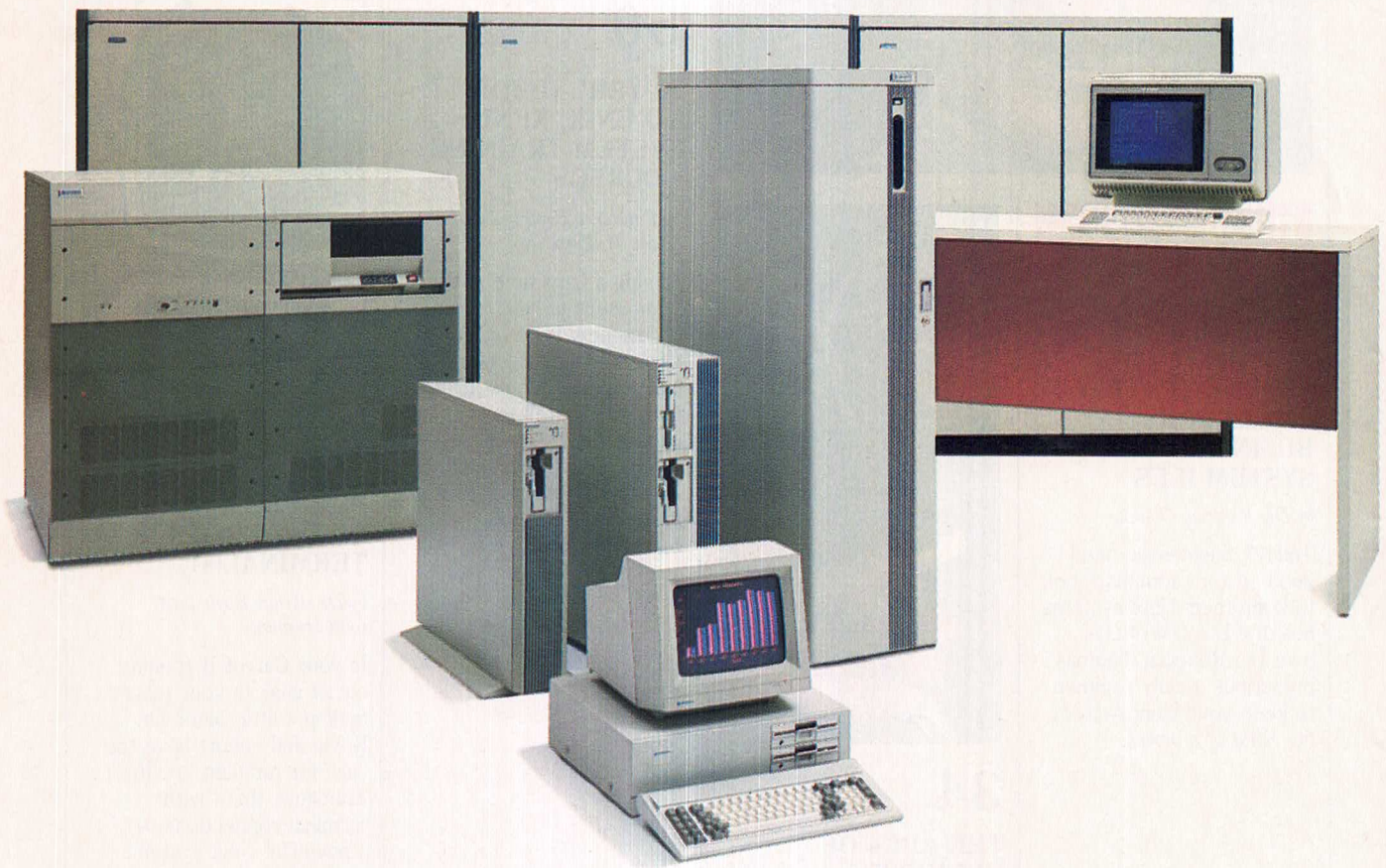
Or course, this linking of all your computer systems can add measurably to your productivity.

And a fast way to find out more is to get a copy of our Sperry Information kit. For yours, or to arrange a demonstration at one of our Productivity Centers, call **1-800-547-8362.**

*UNIX is a trademark of AT&T Bell Laboratories
XENIX and MS-DOS are trademarks of Microsoft Corporation

SPERRY

Introducing an idea that makes obsolescence obsolete.



The UNIX* operating system from PC to mainframe.

Please circle Ad No. 50 on inquiry card.

REVIEWS

JOURNALS

TRENDS

TRAINING

/etc

UNIX/WORLD (ISSN 0739-5922) is published monthly by Tech Valley Publishing, with offices at 444 Castro Street, 12th Floor, Mountain View, CA 94041, telephone 415/964-0900. Third-class permit #1, Sparta, IL 62286. RETURN POSTAGE GUARANTEED. Postmaster, send Form 3579 to UNIX/WORLD, 10 S. Morris St., Dover, NJ 07801.

Subscriptions cost $18 for one year within the U.S.A. and its possessions. For Canadian subscriptions, add $6 for each year. For foreign subscriptions, add $20 each year for surface mail. Remit in U.S. funds drawn on a U.S. bank only. *Subscription questions and problems:* Contact UNIX/WORLD Subscription Department, P.O. Box 305, Dover, NJ 07801. Printed in United States of America. Requests for reprints or bulk orders should be addressed to the Merchandising Manager.

Address all editorial correspondence to the editor at UNIX/WORLD, 444 Castro St., 12th Floor, Mountain View, CA 94041. Unsolicited manuscripts can be returned only if accompanied by a self-addressed, stamped envelope. UNIX/WORLD is not responsible for lost or damaged manuscripts, photos, or artwork.

FROM THE PUBLISHER'S DESK

What's wrong with the Unix system market? Unix system marketeers. The ranting and raving at trade shows about inflated market predictions is just so much hooey. The market *is* there. This magazine proves that: More than 2000 new subscribers join us every month.

Then, why aren't we rich yet?

For starters, purveyors of Unix systems fall into two main groups: the rag-taggle army of startups who saw in the Unix system a weapon in their revolution against IBM; and IBM and the other Big Guys themselves, who are only in because they're afraid they'll miss out as they did on PCs. (Witness IBM's sullen announcement of IX/370 for the Sierra line "for those who consider it a requirement.")

The problem is that neither group is marketing.

Pick a random sampling of startup Unix system vendors. Quiz each on its marketing strategies. Each will answer sagely, "Oh, we don't have the resources to market our products. We're counting on the OEMs to do that." Unfortunately, the mythical OEM (or the even rarer species, the VAR) has yet to hit the airwaves or the print page with a concerted marketing strategy. These guys are in *narrow* established markets—they don't use mass media to create markets.

And don't look to Big Blue and the Seven Dwarfs to dip into the red ink to burn up the airwaves with Super Bowl spots. They never liked the Unix system in the first place; they'll only make it available if you absolutely demand it.

The bottom line is this: Everyone's making the Unix system available, but no one's telling buyers why they need it.

This magazine is going to change all that. We're going to take a leaf from PC magazine publishers and actively kick-start this market, much as they did for the PC market. We're taking the Unix system to the streets.

Our opening salvo was putting UNIX/WORLD on the newsstand. Hundreds of thousands of computer users are being exposed each month to the reality of business Unix systems. And in the coming months our editors will present articles that clearly define the problems with single-user PCs, define the need for multiuser systems, and give clear implementation strategies for those accustomed to PC systems.

Next month in this space I'll delve further into the almost universal, arrogant theme of Unix system marketing campaigns: "Of course you want to buy it, it's Unix!" But this month I want to enlist the support of readers and marketers alike. If we as advocates of the Unix system give you the story, you can make an informed choice either for it, or agin' it.

Let's get the issues on the table and get this sucker rolling. □

John M. Knapp
President & Publisher

UNIX/WORLD

STAFF

Publisher
John M. Knapp
Editor-in-Chief
Philip J. Gill
Editor Emeritus
Dr. Rebecca Thomas
Art/Design Consultant
Robert G. Bryant
Copy Editor
Andrew Ould
Production Editor
Monica E. Berg
Editors-at-Large
Bruce Mackinlay, *System Reviews*
Omri Serlin, *Inside Edge*
Ray Swartz, *Book Reviews*
Lauren Weinstein, *sync*
Associate Editor, Features
Vanessa Schnatmeier
Production Manager
Gerald A. McEwan
Production Artist
James McEwan
Staff Photographer
Ralph Cooksey-Talbott
Cover Art
Paul Fairchild/Judson Allen

CONTRIBUTING EDITORS

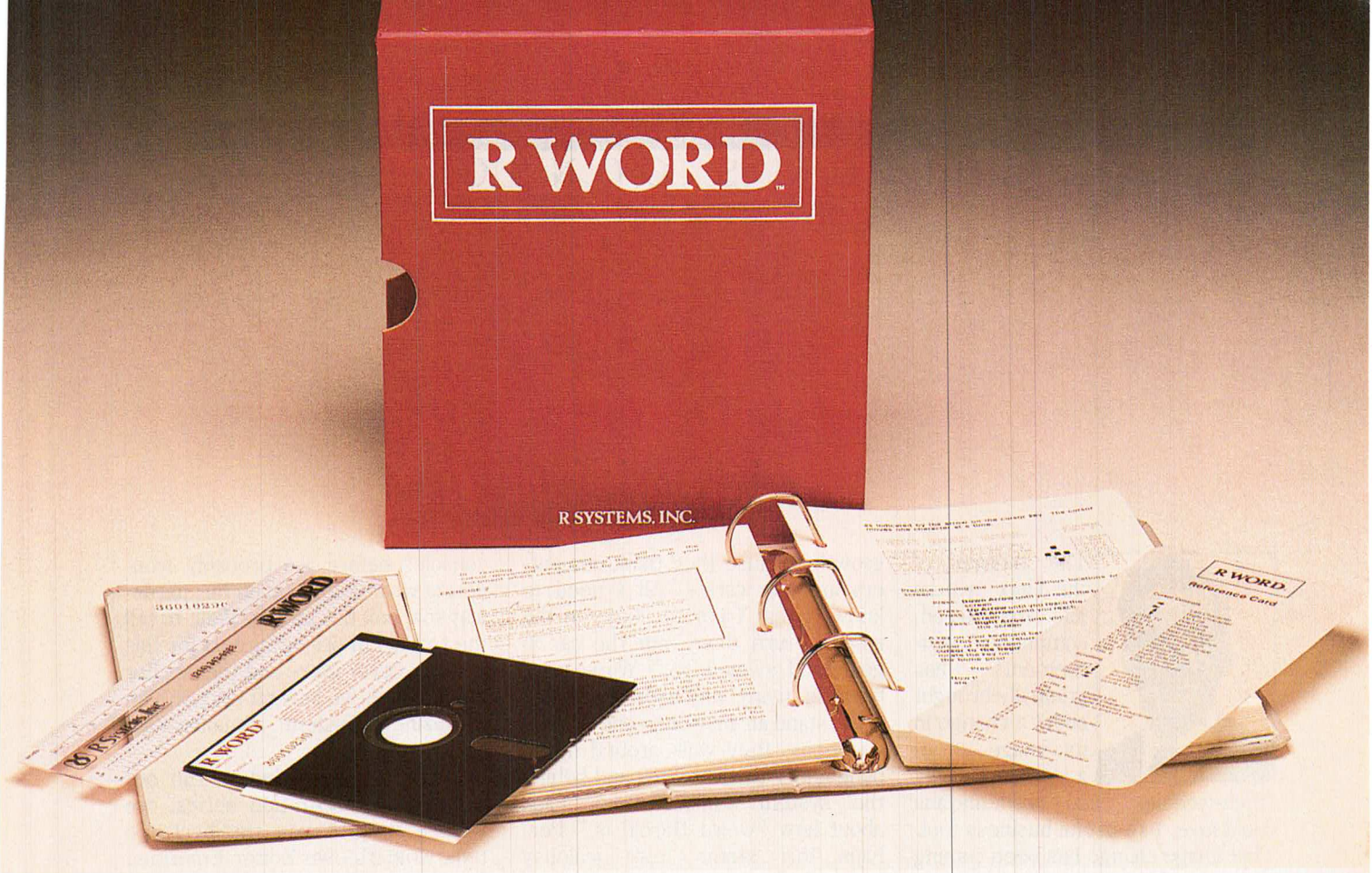
Dr. Brian Boyle, Novon Research Corp.: Market Research; **Thomas R. Billadeau,** The Gartner Group: Office Automation; **Gene Dronek,** Aim Technology: System Optimization/Benchmarking; **Dr. Bill King,** UC Davis: Writer's Workbench; **Bob Marsh,** Plexus Computers: Supermicros; **P.J. Plauger,** Whitesmiths: Work-alikes; **Omri Serlin,** ITOM International: Market Analysis; **Deborah K. Scherrer,** Mt. Xinu: Software Tools; **Ray Swartz,** Berkeley Decision Systems: Book Reviews; **Lauren Weinstein,** Vortex Technology: Networking/Communications.

MARKETING

Marketing Director
Cathy Souza
Circulation Manager
Heidi Spiegelman
Circulation Consultants
Brandt/Klingel Associates
Assistant Advertising Manager
Paul Fertal
Advertising Assistant
Jeanne Ketchum
Advertising Production Coordinator
Beth Doolin
Merchandising Manager
Cheryl Hogan

ADVERTISING SALES

New England, New York, New Jersey, Pennsylvania, and Eastern Canada: Charles E. Lynch, Nancy Wood, Merrie Lynch, 61 Adams St., Braintree, MA 02184, 617/848-9306. **Midwest:** Jeff Edman, Biff Fairclough, The Pattis Group, 4761 W. Touhy Ave., Lincolnwood, IL 60646, 312/478-4000. **Denver:** R.H. "Bud" Foster, 14 Inverness Dr. East, Bldg. A, #232, Englewood, CO 80112, 303/790-8181. **Dallas:** Pam Watkins, 5801 Marvin D. Love Freeway, Suite 303, Dallas, TX 75237, 214/941-4461. **Los Angeles:** Gordon T. Sutton, Carole Ann Philips, 524 W. Commonwealth, Suite G, Fullerton, CA 92632, 714/525-6686. **Miami:** Ray Rickles, Ray Rickles and Company, 3783 Pinetree Dr., Miami Beach, FL 33140, 305/532-7301. **Atlanta:** Scott Rickles, Ray Rickles and Company, 699 Carriage Dr., N.E., Atlanta, GA 30328, 404/252-4567. **San Francisco:** Morton McDonald, Ginger McDonald, Ian McDonald, 265 Baybridge Office Plaza, 5801 Christie Ave., Emeryville, CA 94608, 415/653-2122; 415/967-6400. **Oregon:** Frank M. Eaton, 510 S. First, P.O. Box 696, Hillsboro, OR 97123, 503/640-2011.



"R WORD is an excellent program. It is one of the easiest, yet full-featured, word processors I have ever used."

Bob Woodard of the University of Iowa continues in his letter to us about R WORD, "It beats the heck out of the word processing package available with our machine. R WORD is definitely the favorite of the people who have tried it."

We've received many letters like Bob Woodard's, praising the R Family of Office Automation Software. Now, we'd like **you** to meet R Family — three programs which are powerful, easy-to-use, well-documented, cost-effective, and supportable. And they're designed for single-users, multi-users, and multi-computer users.

R Office Manager

Suggested List $295

R Office Manager is the most powerful organizational and managerial package available. It consolidates and streamlines the basic office functions of today's complex corporate environment. R Office Manager coordinates phone messages, calendars, calculations, things-to-do lists, messages, names and addresses, and more.

R WORD II

Suggested List $895

R WORD II is designed for a production word processing environment which requires full-featured word processing, yet R WORD II offers amazing simplicity of use. Features include Help Menus, Cursor Controls, Editing Functions, Formatting, Three-Level File Structure, Printing Features, Mail Merge, Default Set Up, and Utility Functions.

The program also offers Global Search and Replace, Directories, Spelling Checker, Table Math, Headers and Footers, Automatic Table of Contents, Automatic Paragraph Numbering, Document Assembly, and much more.

R WORD III

Suggested List $1295

R WORD III offers all the features of R WORD II and R Office Manager, plus complete data base, file management, and report generation features which allow you to integrate any information found in other programs on your computer.

1-(800) 527-7610

Call us toll-free for more information about R Family. We are currently ported to and shipping R Family programs on the Tandy Model 16, NCR Tower + XP, Apple Lisa, Pixel, Plexus, and Altos 68000. Additional UNIX and XENIX ports are occurring weekly. We are also shipping for all MS-DOS and PC-DOS operating systems. Manufacture, distributor, dealer, national account, and porting inquiries are invited.

When you select word processing, office management, and office automation software, consider the benefits of R Family. This one set of programs can satisfy single-user, multi-user, and multi-computer requirements. R Family offers you consistency, versatility, and supportability, as well as the benefits of transportability and connectivity.

The R Family - single and multi-user office automation software:
DOS • UNIX • XENIX
COS • DX10 • DNOS

R SYSTEMS, INC. **11450 Pagemill Road, Dallas, TX 75243 (214) 343-9188 (800)527-7610**

Please circle Ad No. 95 on inquiry card.

EDITOR'S CONSOLE

The merger of the Unix system into mainstream computing environments has gradually brought about a change in its image—from a product only "tech-weenies" and academicians could love, to a useful business tool. That image change has been coming on strong in the last year, particularly because of AT&T's massive newspaper, magazine, and television advertising campaign that has made "Unix" a household word.

That same image change, however, has yet to reach into one special and significant area of the Unix system world—system administration. Today, literally hundreds of thousands of business users rely on Unix system-based computers to do their daily work. Yet these same users still regard those who keep the system going as nefarious wizards capable of powers far beyond those of mere mortals. This situation, fostered by actions on both sides, is unfortunate. Some—but certainly far from all—Unix system administrators jealously guard their secrets out of fear or a need for superiority. Others casually dismiss any signs of interest in computers and in the Unix system on the part of their users as the febrile murmurings of idiots.

On the other side are the users, who are not by any means the innocents in this matter. Often they feel left out of an important side of their business lives, one they can't understand, or cut off from further career growth because they think computers are like cars—"All I need to know is that I put the keys in the ignition, turn them, put the car in gear, and away I go." Worse yet, because many end-users may not understand an answer to one of their questions, they walk around mumbling under their breaths and behind the system administrator's back about how "weird Harold is." Perhaps this stems from jealousy because the system administrator *actually understands* all this computer mumbo jumbo.

It's time, then, for both sides to sit down at the peace table and start talking. Remember, you both couldn't get along without each other, despite your earnest yearnings for the contrary. For you few remaining Unix system administrators who thrive on the wizardry image, let me tell you that your users are a potential source of new ideas that can simplify your jobs and boost your careers at the same time.

To the end-user, I say that the problem is this: Despite the "wizards'" knowledge of computers in general and of the Unix system in particular, they are indeed just mortals like you and I. Remember that. Therefore, they make mistakes. More importantly, they are humans attempting to deal with machines, and we all know that machines sometimes have minds of their own.

It's time to declare a truce, take off the wizard's frock, and get down to business by using a little common sense and interpersonal communication. Both sides will be better off; I guarantee it.

Those of you who have read my consoles before are probably wondering what got me up on this particular soapbox this month. Well, to tell the truth, I'm preparing you for this month's cover story and theme. For in this issue, we kick off UNIX/WORLD Magazine's system administration series.

When we decided to run our systems administration series, we wanted to give you the best. So UNIX/WORLD's own Editor Emeritus, Dr. Rebecca Thomas, an author of *A User Guide to the Unix System* and a noted authority on the Unix system and the C language, jumped to her feet, volunteering to write and oversee the series. Each month, Dr. Thomas will bring you new tips, techniques, and advice on how to keep your Unix system smoothly tuned and purring. And on some occasions she'll have a little help. This month, for instance, experienced 4.2BSD system administrators Nancy Blachman and Phil Ngai tell you the special ins and outs of administering Berkeley Unix systems.

I'd normally go on from here and tell you about the issue, its exciting features and product reviews, and other items of special interest. Only problem is, I've run out of space; soapboxes do that to you. Let it suffice to say that I'm proud of this issue and invite you to enjoy it. Also, if I may, my personal favorite this month is "PCs: From Now On Consider Them Terminal(s)," by *Car and Driver* refugee Dr. Brian Boyle and his cohort John Dunham.

Philip J. Gill
Editor-in-Chief

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

Now Gould offers the widest range of UNIX-based systems in the world.

Gould's PowerSeries™ computers span the performance range. From professional desk-top computers and high-resolution graphics workstations to shared resource and back-end processors with up to twice the performance of a VAX 11/780.™ And, all with the UNIX* operating system to assure interconnectability and software compatibility.

In addition, there's utility. Gould PowerSeries computers are designed for the way you work. Users have all the advantages of a dedicated system plus the lower-cost-per-user option of sharing resources with Gould's offering of standard networking strategies including Ethernet.™

Computer users also need compatibility. Gould's years of expertise with UNIX software have produced the "Compatibility Suite" of application software packages that is compatible and consistent across the entire PowerSeries product line under our Universal Timesharing Executive, UTX,™ a unique combination of Berkeley 4.2 BSD with selected features from AT&T's System V.

And, for your service and support needs, you can rely on Gould's worldwide customer support network.

Put Gould PowerSeries Computers to work for you now.

Gould Inc., Computer Systems Division
Distributed Systems Operation
6901 West Sunrise Boulevard
Fort Lauderdale, FL 33313
1-800-327-9716

*UNIX is a trademark of AT&T Bell Labs
™PowerSeries and UTX are trademarks of Gould Inc.
™Ethernet is a trademark of Xerox Corp.
™VAX is a trademark of Digital Equipment Corporation

Please circle Ad No. 88 on inquiry card.

GOULD
Electronics


HEWLETT PACKARD
Integral Personal Computer

Almost priceless.

Hewlett-Packard presents the first 32-bit UNIX™ system under $5,000.

Introducing the Integral Personal Computer.

It's the newest member of the HP 9000 family of HP-UX workstations. And it delivers the kind of power and flexibility the name implies.

With a 16/32-bit MC68000. A graphics co-processor. And 800KB of standard memory, expandable to 5.8MB.

Its UNIX kernel (HP-UX/RO) is built into ROM. Which means the Integral PC can run most applications *without* an expensive hard disc. Which means more people can afford to run your UNIX software than ever before. But not only can they afford to run it. They can learn to *use* it in minutes. Because the Integral PC is one UNIX system that's easy to use. Its see-and-select user interface, built-in window manager and optional mouse mean even novices can tap the power of your software. No one has to learn a bunch of cryptic commands or go through a lot of expensive training. There's even an entry-level, self-paced tutor disc with every machine.

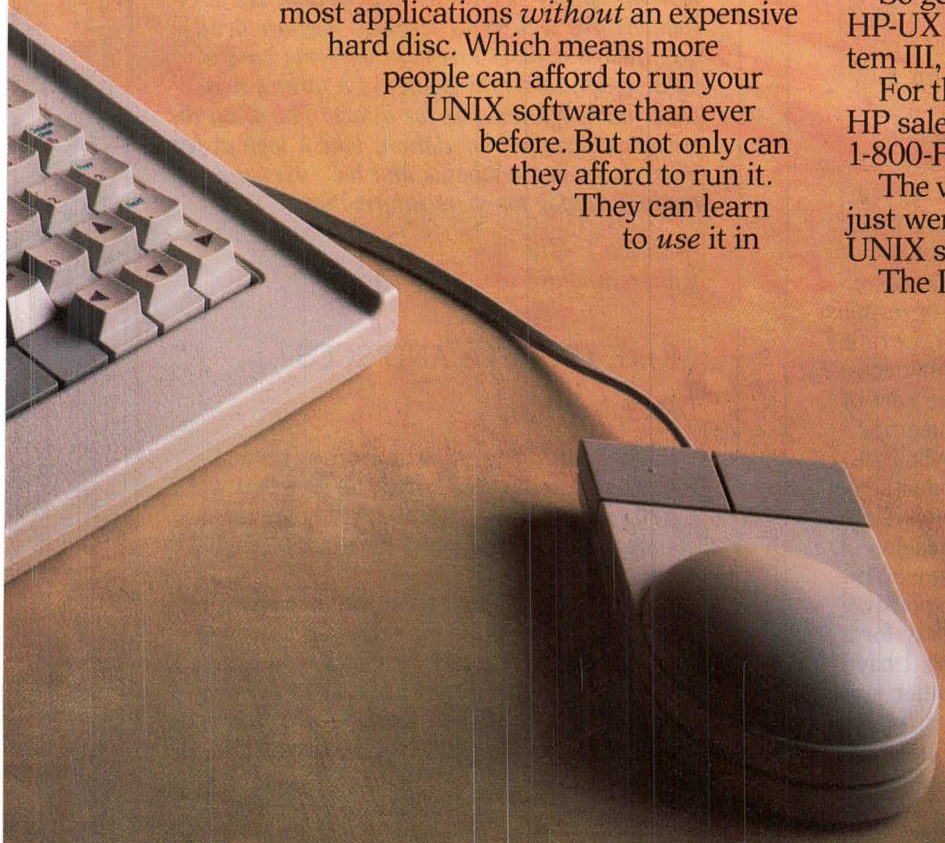
What's more, the Integral PC goes where you need it. The complete system—including our revolutionary ThinkJet printer—packs into a single 25-pound package that takes up less than a cubic foot.

So get started porting your software now. HP-UX is an enhanced version of UNIX System III, including Berkeley 4.2 BSD features.

For the name of the authorized HP dealer or HP sales office nearest you, call toll-free 1-800-FOR-HPPC.

The value of your investment in UNIX software just went up. Because the price of a complete UNIX system just came down.

The Integral Personal Computer. Just $4,995.*



HEWLETT PACKARD

PGO2504

UNIX is a trademark of AT&T Bell Laboratories.
Mouse is optional. *$4,995 is suggested U.S. list price; prices elsewhere may vary.

Please circle Ad No. 93 on inquiry card.

mail

This space is yours. Tell us what we're doing right—or wrong. Raise issues and ideas that other readers may respond to. Propose projects. Reject market trends. Take a stand.

ONE READER RESPONDS

Dear Editor:

I have been a reader of UNIX/WORLD since your third issue, and I must say that thus far it has gotten better issue by issue. Your last issue (Vol. 1, No. 7) was the best of all. Please continue to deal with interesting Unix system features (such as the `uucp` and USENET networks) and with timely aspects of the Unix world in general (such as the job market).

As the issues go by, I hope you will continue to delve into technical issues more deeply. Owners of Unix systems are likely to be more interested in the nuts and bolts of how things work than are owners of computers with less demanding operating systems. I found the "Wizard's Grabbag" particularly interesting, and I hope this sort of "how to do it" article is a regular feature of your magazine.

I do have one quibble with Vol. 1, No. 7, and that is with Stephen Auditore's article. I'm sure that many people are interested in alternate marketing models for software, but Mr. Auditore does your readers a disservice by being so cavalier with his suggestion for a subscription-like model for software.

Software updates are *not* analogous to news. Old news is useless (to all but historians), but software may continue to serve people for years without being updated, provided the particular user's needs do not require the updated capabilities (as is often the case).

I am also curious about the lack of terminal emulation software available under an operating system that is so famous for its "communications capabilities." I am thinking of the sort of program one would use to call Compuserve, The Source, or Western Union's Easylink. A product mentioned in your "New Products" column ("ASCII Communications Utility for Unix Systems") is the sort of thing I am thinking of. But why are these products so rare under the Unix system, and why are they so expensive?

Two thousand dollars is a lot to pay for a product that does no more than others I have that cost only $150 and that run on other standalone and multiuser operating systems (MS-DOS, CP/M, Oasis, etc.). The Unix system does include `cu`, but this utility is next to useless as it does not allow downloading to local disks (unless the text is specially formatted by the host); and its uploading functions make no provision for dealing with remote editors that have turnaround prompts or what have you.

Again, keep up the good work. I own an Altos 586 running Xenix, and I'll be following your magazine closely. In fact, this letter is my first `nroff`-ed document.

Sincerely,

Matthew Rapaport
Petaluma, Calif.

Editor's Reply: Thank you for the compliments. I'd like to address your letter point by point, if I may. First, please take a more careful look at our back issues—"Wizard's Grabbag" is and will continue to be a regular monthly feature. Second, Mr. Auditore's plan for subscription software is not as far out as you think, nor is it really uncommon in the computer business. Many mainframe users rent or license their software on a monthly basis—a form of subscription. As for comparisons with news services, they are available almost without exception on some kind of subscription or rental basis. As a former newspaper man, I know this for a fact. Third, on the subject of terminal emulation, take a look at Jean Yates' Unix Encyclopedia *and the /usr/group's product catalog for more information.*

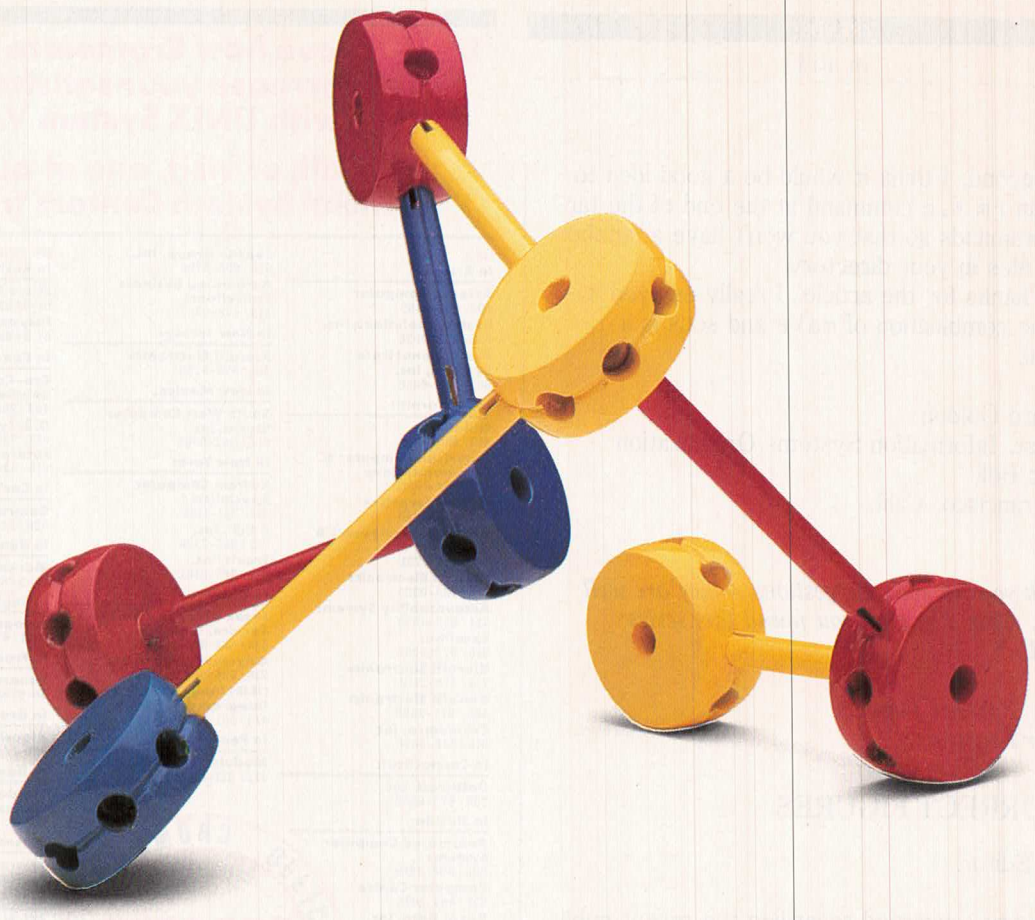
—Philip J. Gill
Editor-in-Chief

MORE ON `make` AND SCCS

Dear Editor:

I enjoyed Dr. Thomas' article on `make` and the SCCS system (Vol. 1, No. 5); the two work together very well. There were, however, a couple of points that could have been emphasized.

First, it wasn't clear to me that `make` took care of doing the `get -p` retrieval. I got the impression from the article that I would have to place those commands in my `makefile` myself. Only when I began working with the SCCS stuff did it become clear that `make` itself had those capabilities.



Networking Software So Simple To Use, It's Child's Play.

What do you think of when someone says "network"? Hardware? Complexity? Money? At TouchStone, we think networks should be simple.

The Connectables Software from TouchStone

The Connectables software — PCworks,™ MacLine,™ and UniHost™ — lets you build a network out of a collection of computers. Without special hardware, unnecessary complexity, or a lot of money.

UNIX + PC + Macintosh

Three great computers with different processors, different operating systems, and different data formats. The Connectables software lets you connect these systems together in whatever way you'd like to use them. Share electronic mail, exchange all types of files, run a program on another system, or use a remote computer's disk or printer. The only hardware you'll need is a standard cable or telephone line and a serial data channel.

It's So Easy.

The Connectables software will take care of all the details — dial the right numbers, ask you what you want to do, and figure out the rest. Move a spreadsheet from your PC to a UNIX™ system, or transfer a text file from the UNIX system to your Macintosh™ for editing. Print the file later on either system, or use another computer to backup your data. With The Connectables software, things that should be easy to do, are easy to do.

THE CONNECTABLES™:
FAMILY OF COMMUNICATION SOFTWARE PACKAGES.

Yesterday, Today, Tomorrow?

Making computers work together shouldn't be a problem. TouchStone's PCworks, UniHost, and MacLine software packages are only the first three members of The Connectables family. TouchStone is committed to the continued development of software products that solve compatibility problems. Even the problems we won't know about until tomorrow.

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

Please circle Ad No. 82 on inquiry card.

Second, I think it would be a good idea to have an `rm *.o` command at the end of the target commands so that you won't have a number of `.o` files in your directory.

Thanks for the article. I really enjoyed it, and the combination of `make` and SCCS is a natural.

Bernard Golden
Analyst, Information Systems Organization
Pacific Bell
San Francisco, Calif.

Thank you for your suggestions—they are well taken. Also, I'm glad you found this series useful.

—Dr. Rebecca Thomas
Editor Emeritus

INCORRECT FIGURES

Dear Editor:

I am very concerned regarding the recent publication of my article "`troff`: A Text Software Boon for Scientific Authors" in the February issue of UNIX/WORLD.

Your production staff inadvertently used the incorrect figures. Being in the publishing business, I do understand how mistakes can happen. Your staff was at all times professional and helpful, and it is truly unfortunate that this mistake happened.

Thank you for all your cooperation and personal attention in this matter.

Sincerely,

Peggy Judd
Associate Project Director—Unix
The American Physical Society
500 Sunnyside Blvd.
Woodbury, NY 11797

Editor's Reply: We here at UNIX/WORLD truly regret that the wrong figures were used. In the interest of serving our readers, we are printing the correct figures in the back of this issue. Furthermore, should you need more information, feel free to contact Ms. Judd.

—Philip J. Gill
Editor-in-Chief

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®

MCBA introduces shrink-to-fit software.

With nine years in minicomputer software, 15,000 installations worldwide and an established reputation in the mini world, MCBA is proudly shrinking its software line.

Down to micro size.

We've taken the impressive power of minicomputer software and made it available for micros. Right now.

Alter the fit? Absolutely.

Alter the functionality, modularity and capability? Not one bit... so to speak.

This new line of serious microcomputer software is by far the most comprehensive, well-tested and sophisticated in the industry today. By whose standards? Thousands of MCBA users who rank it the best in the business.

MCBA's library of 16 integrated manufacturing, distribution and accounting packages can be installed in whatever combination and sequence a user needs for his or her business.

It grows with businesses. No matter what size they are now. Or want to be later.

And MCBA software now runs in RM/COS,® PC-DOS, UNIX™ and UNIX look-alike environments.

In other words, we've tailor-made our newest software to fit micros—as comfortably as it fits user needs.

So whether you're a dealer or a user, find out about it. Call **1-800-MCBA NOW** (toll free outside of California). In California, call **(818) 957-2900.** MCBA's shrink-to-fit software. For growing businesses.

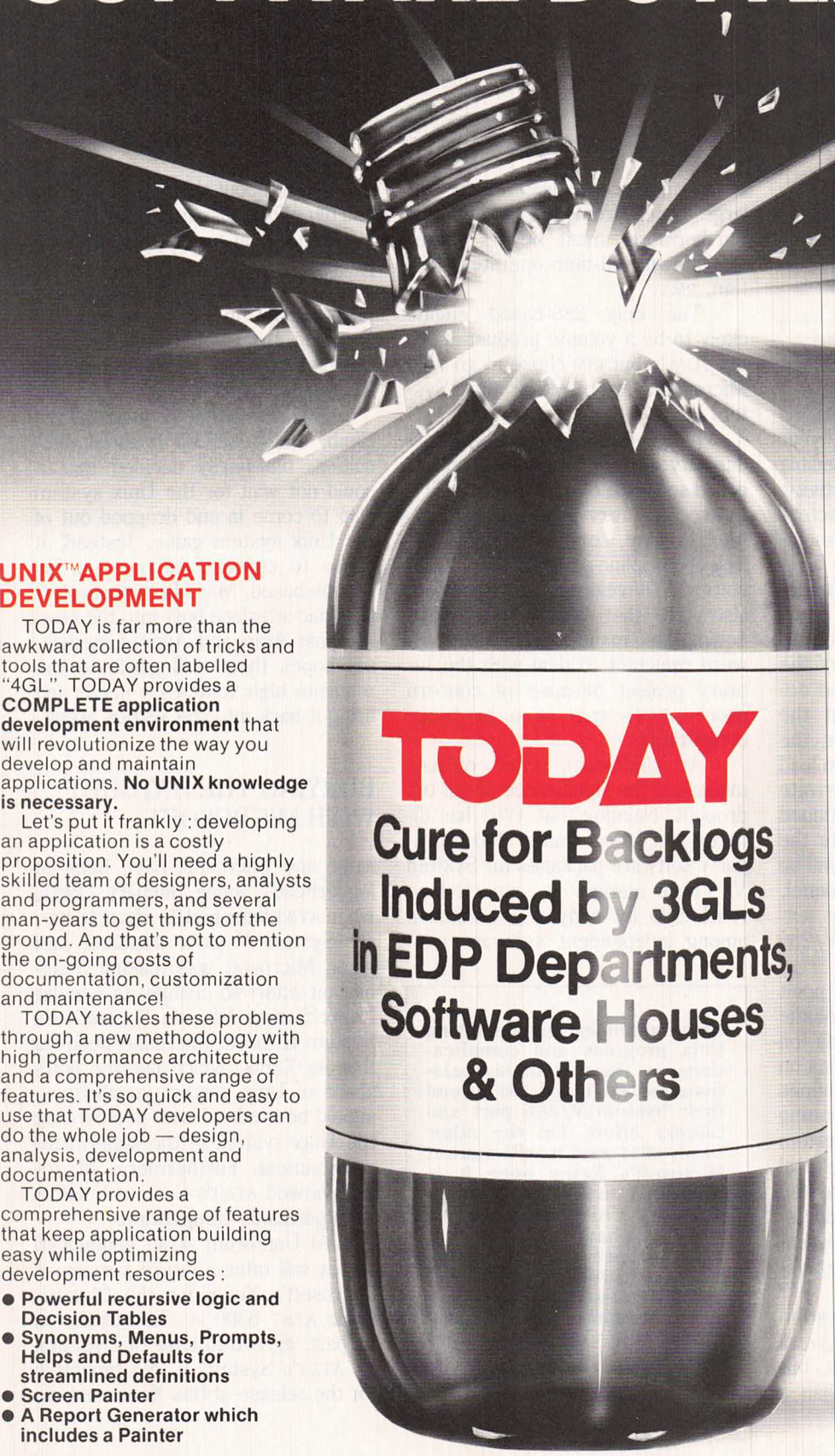
MCBA®

Minicomputer Software for Micros.

2441 Honolulu Avenue, Montrose, California 91020

Also for DEC, Wang, HP, TI, and Perkin-Elmer minis.

BREAK THROUGH THE SOFTWARE BOTTLENECK



UNIX™ APPLICATION DEVELOPMENT

TODAY is far more than the awkward collection of tricks and tools that are often labelled "4GL". TODAY provides a **COMPLETE application development environment** that will revolutionize the way you develop and maintain applications. **No UNIX knowledge is necessary.**

Let's put it frankly : developing an application is a costly proposition. You'll need a highly skilled team of designers, analysts and programmers, and several man-years to get things off the ground. And that's not to mention the on-going costs of documentation, customization and maintenance!

TODAY tackles these problems through a new methodology with high performance architecture and a comprehensive range of features. It's so quick and easy to use that TODAY developers can do the whole job — design, analysis, development and documentation.

TODAY provides a comprehensive range of features that keep application building easy while optimizing development resources :

- **Powerful recursive logic and Decision Tables**
- **Synonyms, Menus, Prompts, Helps and Defaults for streamlined definitions**
- **Screen Painter**
- **A Report Generator which includes a Painter**
- **Push-button Self-documentation**
- **Audit Trails**
- **Source-code security through run-time only configurations**
- **Developed Applications instantly portable across UNIX-based machines**

Because definitions are **Dictionary-based,** any changes are easily made in one central location. A key feature, **"tailoring"** lets you alter an application — perhaps to customize it for a particular site or user — without affecting the original version. If required, applications can be set up as Models (Prototypes) and later enhanced to grow and change with the business. Tailoring versions is the perfect solution for quickly generating multiple applications based on one Model.

TODAY runs under UNIX or UNIX-compatible operating systems on **super-mini down to micro business computers** using any of a range of databases. And if that's not enough, TODAY is backed by 14 man-years of research and development and the confidence of users throughout Australia who are breaking time zones in software development. Watch for our stand at the UNIX Expo in San Francisco, April 24-26, or contact us at :

bbj Computer Services Inc.
2946 Scott Blvd
Santa Clara, CA 95054
Telephone : (408) 727-4464

Please circle Ad No. 44 on inquiry card.

AT&T AND DRI, MICROSOFT, ET AL.

BY OMRI SERLIN

AT&T has begun the new year with an incredible flurry of activity. First, two factors led to the dissolution of the System V Library development effort at Digital Research Inc.: DRI's disappointment with the progress of the Unix system, and the reported decline of interest at AT&T in the Library project. Bruce Weiner, the ex-Zilog staffer DRI recruited to lead the Library project, is now a private consultant. DRI also discontinued work on the System V port for the Intel 286, although that work will be continued by Microport, a Soquel, Calif.-based firm formed by ex-members of the DRI System V/286 development team.

The key problem is DRI's point of view. The company is volume oriented; its success depends on charging small per-copy royalties on products that sell in large volumes (for example, the CP/M operating system). DRI had hoped the System V port it was developing for the Intel 286 would become a widely accepted base for higher-margin products, such as the languages and applications programs planned for the System V Library.

Intel, for which the System V/286 port was done, does have a line of 286-based supermicros, but the product has been less than a smashing success, despite a recent large order from the U.S. Army. Furthermore, most buyers prefer Intel's own real-time operating system, RMX.

The only 286-based engine likely to be a volume product is the IBM PC/AT, but IBM chose to go with Microsoft's Xenix rather than System V.

Another problem has been the difficulty of locating Unix system-based software of consistent quality. In a recent interview, DRI's president, John Rowley, complained about the lack of discipline among Unix system software developers. DRI staffers also felt that hardware original equipment manufacturers (OEMs) were reluctant to deal with the Library project because of concern about AT&T's role as a hardware competitor.

At UniForum, AT&T officials minimized the significance of the DRI dropout, claiming that AT&T has directly recruited some 200 third-party software packages for System V. (That number is expected to reach 500 by midyear.) Confusion among independent software vendors (ISVs) about the duplicate AT&T and DRI efforts also may have figured in the decision to drop the Library project.

AT&T and DRI apparently were unsuccessful in obtaining venture financing that would have allowed the Library effort to continue as an independent entity.

Faced with a slow, long upward climb, and beset by financial difficulties, DRI finally decided that it could not wait for the Unix system ship to come in and dropped out of the Unix system game. Instead, it chose to concentrate on GEM, an MS-DOS-based, Mac-like software ergonomic interface built into the new, low-cost Atari and Apricot models. DRI hopes these will generate the requisite high volumes it needs and bring it back into the MS-DOS arena.

News summary: **Disappointing Unix progress and complications in their business relations led AT&T and DRI to end their System V/286 port and Library effort. On the other hand, AT&T said it will market Microsoft's Xenix once it is converted to System V compatibility. Chiefs of AT&T's two main arms have been replaced, possibly because of disappointing 1984 performance. And the FCC is finally moving to dismantle some of the key provisions of its Computer Inquiry II decision.**

BURYING THE HATCHET WITH MICROSOFT

AT&T and Microsoft have been at loggerheads since January 1983, when AT&T launched its effort to create System V as a standard, just when Microsoft was making a significant effort to change its popular Xenix from a Version 7 base to a System III base. Microsoft believes that its Xenix, which has the lion's share of Unix system installations, should be credited with popularizing the Unix system in the supermicro environment. Furthermore, Microsoft viewed AT&T's System V effort as superfluous competition.

At UniForum, AT&T announced that it will offer a future version of Microsoft's Xenix for the Olivetti-made AT&T 6300 PC. Microsoft, on its part, agreed to base that version on AT&T's System V. No timetable for the release of this Xenix version

UNIX™ & 'C'

HANDS-ON SEMINARS

A Complete Curriculum for: End Users • Management • Applications Staff • Technical Support

COURSES	LONDON	NEW YORK	BOSTON	WASHING-TON, D.C.	SAN FRANCISCO	CHICAGO	DALLAS	LOS ANGELES	TUITION	SEQUENCE TUITION†
UNIX Overview	Apr 9 June 11 Aug 6 Oct 8 Dec 3	Jan 15 May 28 Oct 1	Mar 19 July 30 Dec 17	Jan 15 Mar 19 May 28 July 30 Oct 1 Dec 17	Feb 19 June 18 Oct 15	Mar 12 Aug 20	Feb 19 June 18 Oct 15	Mar 12 Aug 20	$225	$860
UNIX Fundamentals for Non-Programmers*	Apr 10-12 June 12-14 Aug 7-9 Oct 9-11 Dec 4-6	Jan 16-18 May 29-31 Oct 2-4	Mar 20-22 July 31-Aug 2 Dec 18-20	Jan 16-18 Mar 20-22 May 29-31 July 31-Aug 2 Oct 2-4 Dec 18-20	Feb 20-22 June 19-21 Oct 16-18	Mar 13-15 Aug 21-23	Feb 20-22 June 19-21 Oct 16-18	Mar 13-15 Aug 21-23	$735	
UNIX Fundamentals for Programmers*	Apr 15-17 June 17-19 Aug 12-14 Oct 14-16 Dec 9-11	Jan 21-23 June 3-5 Oct 7-9	Mar 25-27 Aug 5-7	Jan 21-23 Mar 25-27 June 3-5 Aug 5-7 Oct 7-9	Feb 25-27 June 24-26 Oct 28-30	Mar 18-20 Aug 26-28	Jan 14-16 Feb 25-27 June 24-26 Oct 28-30	Mar 18-20 Aug 26-28	$735	$1125
Shell as a Command Language*	Apr 18-19 June 20-21 Aug 15-16 Oct 17-18 Dec 12-13	Jan 24-25 June 6-7 Oct 10-11	Mar 28-29 Aug 8-9	Jan 24-25 Mar 28-29 June 6-7 Aug 8-9 Oct 10-11	Feb 28-Mar 1 June 27-28 Oct 31-Nov 1	Mar 21-22 Aug 29-30	Jan 17-18 Feb 28-Mar 1 June 27-28 Oct 31-Nov 1	Mar 21-22 Aug 29-30	$490	
'C' Language Programming*	Apr 22-26 June 24-28 Aug 19-23 Oct 21-25 Dec 16-20	Jan 28-Feb 1 June 10-14 Oct 21-25	Apr 15-19 Aug 12-16	Jan 28-Feb 1 Apr 15-19 June 10-14 Aug 12-16 Oct 21-25	Mar 4-8 July 8-12 Nov 11-15	Apr 29-May 3 Sep 9-13	Jan 21-25 Mar 4-8 July 8-12 Nov 11-15	Apr 29-May 3 Sep 9-13	$1225	
Shell Programming*	Apr 29-30 July 1-2 Sept 2-3 Oct 28-29	Feb 4-5 June 17-18 Oct 28-29	Apr 22-23 Aug 19-20	Feb 4-5 Apr 22-23 June 17-18 Aug 19-20 Oct 28-29	Mar 11-12 July 15-16 Nov 18-19	May 6-7 Sep 16-17	Jan 28-29 Mar 11-12 July 15-16 Nov 18-19	May 6-7 Sep 16-17	$490	$1125
Using Advanced UNIX Commands*	May 1-3 July 3-5 Sept 4-6 Oct 30-Nov 1	Feb 6-8 June 19-21 Oct 30-Nov 1	Apr 24-26 Aug 21-23	Feb 6-8 Apr 24-26 June 19-21 Aug 21-23 Oct 30-Nov 1	Mar 13-15 July 17-19 Nov 20-22	May 8-10 Sep 18-20	Jan 30-Feb 1 Mar 13-15 July 17-19 Nov 20-22	May 8-10 Sep 18-20	$735	
UNIX Internals	May 7-10 July 8-12 Sept 4-6 Oct 30-Nov 1	Feb 11-15 June 24-28 Nov 11-15	Apr 29-May 3 Aug 26-30	Feb 11-15 Apr 29-May 3 June 24-28 Aug 26-30 Nov 11-15	Mar 18-22 July 22-26 Dec 2-6	May 13-17 Sep 23-27	Feb 4-8 Mar 18-22 July 22-26 Dec 2-6	May 13-17 Sep 23-27	$1375	
UNIX Administration*	May 15-17 July 15-17 Sept 18-20 Nov 11-13	Feb 19-21 July 9-11 Nov 19-21	May 7-9 Sep 10-12	Feb 19-21 May 7-9 July 9-11 Sep 10-12 Nov 19-21	Mar 26-28 July 30-Aug 1 Dec 10-12	May 21-23 Oct 1-3	Feb 12-14 Mar 26-28 July 30-Aug 1 Dec 10-12	May 21-23 Oct 1-3	$735	
Advanced 'C' Programming Workshop*	May 20-21 July 22-23 Sept 23-24 Nov 18-19	Feb 25-26 July 15-16 Dec 2-3	May 13-14 Sep 16-17	Feb 25-26 May 13-14 July 15-16 Sep 16-17 Dec 2-3	Apr 15-16 Aug 5-6 Dec 16-17	Jan 14-15 June 3-4 Oct 7-8	Apr 15-16 Aug 5-6 Dec 16-17	Jan 14-15 June 3-4 Oct 7-8	$490	$1125
Advanced 'C' Programming Under UNIX*	May 22-24 July 24-26 Sept 25-27 Nov 20-22	Feb 27-Mar 1 July 17-19 Dec 4-6	May 15-17 Sep 18-20	Feb 27-Mar 1 May 15-17 July 17-19 Sep 18-20 Dec 4-6	Apr 17-19 Aug 7-9 Dec 18-20	Jan 16-18 June 5-7 Oct 9-11	Apr 17-19 Aug 7-9 Dec 18-20	Jan 16-18 June 5-7 Oct 9-11	$735	
Berkeley Fundamentals and 'csh' Shell*	June 3-7 July 29-Aug 2 Sept 30-Oct 4 Nov 25-29	Mar 4-8 July 22-26 Dec 9-13	May 20-24 Sep 23-27	Mar 4-8 May 20-24 July 22-26 Sep 23-27 Dec 9-13	Apr 22-26 Aug 12-16	Jan 28-Feb 1 June 10-14 Oct 21-25	Apr 22-26 Aug 12-16	Jan 28-Feb 1 June 10-14 Oct 21-25	$1225	

*Including hands-on training workshops ™UNIX is a trademark of Bell Laboratories †Savings for consecutive seminar dates

CALL FOR DETAILS ON: ON-SITE SEMINARS • VIDEO-BASED TRAINING • INTERACTIVE VIDEODISC TRAINING

To reserve your seminar space now or for additional information, call: (800) 323-UNIX or in Illinois (312) 987-4082

MANY UNIX-BASED SYSTEMS—ONE UNIX TRAINING COMPANY

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 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.

COMPUTER TECHNOLOGY GROUP

Telemedia, Inc.

310 S. Michigan Ave., Chicago, IL 60604

Please circle Ad No. 115 on inquiry card.

has been announced, although this writer believes this so-called Xenix 5.0 is targeted for release in midyear.

It is unlikely that the present Xenix and System V versions can be made entirely compatible without affecting existing products running on these versions, which differ in binary file formats as well as many other points. AT&T will probably incorporate Xenix as part of a series of extensions to its System V standard. Other extensions may feature real-time support, transaction processing support, 4.2BSD facilities, and the already promised compatibility with the standard defined by /usr/group.

At UniForum AT&T formally announced the availability of a System V Standard document, which defines the minimum subset of system calls and their effect. This subset will be maintained "as is" across future releases, allowing ISVs to develop software that will not be made obsolete by such future versions. UniSoft, the Berkeley, Calif.-based porting house, has been selected to develop a suite of benchmarks that will validate compliance with the standard.

Also at UniForum, Amdahl announced the availability of UTS/V, a System V-based version of its UTS. UTS/V is a software system that runs as a "guest" under the IBM VM operating system. (See UNIX/WORLD's review of UTS in the April 1985 issue.) AT&T and Amdahl officials declined to respond to speculation that AT&T eventually will offer UTS/V directly to IBM mainframe installations. AT&T officials noted, however, that AT&T has a significant number of IBM and Amdahl mainframes in-house. Earlier press reports that AT&T was preparing to offer Amdahl hardware on an OEM basis were vigorously denied by AT&T executives.

EDS AGREEMENT

AT&T is, of course, very interested in extending the scope of the Unix system to mainframes, and the UTS/V move is clearly part of that thrust. Nevertheless, AT&T views the mainframe segment as much more than just another target for the Unix system. Evidence of this surfaced in late January, when AT&T announced a joint marketing agreement with Electronic Data Systems (EDS) to provide customized computer services. EDS, which had $786 million in revenues for its last fiscal year, is a Dallas-based supplier of customized computer system integration, services, and support, principally on IBM mainframes. The company has been acquired by General Motors as part of a massive high-tech diversification effort on GM's part.

EDS will dedicate 150 staffers to the project, under which AT&T and EDS will cooperate in designing, installing, and supporting custom systems involving large-scale computers interfaced to communication networks. Fifty employees will support AT&T Information Systems (AT&T-IS) sales forces in offices around the country. One hundred more will offer support from two centers in Piscataway, N.J.

EDS is interested because it hopes the extensive field presence of AT&T-IS salesmen will help uncover incremental system integration business opportunities for EDS. AT&T, on the other hand, has clearly been disappointed with the lackluster performance of the AT&T-IS sales force; it hopes that EDS' expertise in data processing and system integration, and in selling to business and government, will contribute to more effective selling of its computer gear. However, AT&T carefully noted that the integrated systems to be offered "may include those of other quality companies, in addition to those in AT&T's product line."

EXECUTIVE SHIFTS

The announcement in mid-January that AT&T would be replacing the chief executives of its two main arms—AT&T Information Systems and AT&T Communications—leads to some interesting speculation. Despite denials by AT&T executives, it's hard to avoid the impression that the move reflects disappointment with the performance of these key organizations, an impression re-inforced by the financial results (see below).

Chuck Marshall, who formerly headed AT&T-IS, has been shifted to corporate headquarters and has been given the title of executive vice president. He will be in charge of personnel and external affairs, including regulatory matters. His place was taken by Robert E. Allen, previously executive vice president for corporate administration and finance.

Allen is said to be a protégé of AT&T's chairman, Charles Brown. Some analysts have even suggested the entire move was designed to give Allen operational command duty in preparation for his eventually assuming the chairmanship. Although not likely to happen when Brown retires in June 1986, this could occur at the next shift in leadership.

Morris Tannenbaum, who headed AT&T Communications, has become corporate executive vice president in charge of financial management and strategic planning. His place was taken by Randall L.

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

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

MORE/bsd and MT XINU are trademarks of Mt Xinu Inc., DEC and VAX are trademarks of Digital Equipment Corp., UNIX is a trademark of Bell Laboratories.

Please circle Ad No. 113 on inquiry card.

Tobias, previously executive vice president in charge of regulatory and legislative issues.

AT&T Information Systems is in charge of most of AT&T's unregulated activities, including the marketing of customer-premise equipment (CPE), computers, and information services. Part of AT&T Technologies, it is nominally an "arm's length" subsidiary, formed under the FCC's Computer Inquiry II rules. AT&T Communications is in charge of AT&T's long-distance telephone business, still under FCC regulation.

Part of the reason for moving Chuck Marshall to corporate headquarters is AT&T's desire to speed up the dismantling of the FCC's Computer Inquiry II. AT&T has long viewed CI II, especially the requirement for an "arm's length" relationship with any nonregulated activity, as onerous, wasteful, and especially incongruous following the divestiture agreement.

In early 1984 AT&T petitioned the FCC for relaxation of some of CI II's provisions. The FCC responded in late January 1985 by submitting for industry comment a plan to drop the arm's length requirement. This will permit AT&T to integrate AT&T-IS more fully and to complete the implementation of the line-of-business organization under which AT&T-IS is to receive control of the design and manufacturing of the gear it currently is allowed to merely sell (see "Inside Edge" in the March 1985 issue of UNIX/WORLD).

AT&T completed its first post-divestiture year well under its own (and analysts') expectations. In 1984 the company earned $1.38 billion ($1.25/share) on revenues of $33.19 billion. (AT&T had anticipated $2.02 in per-share earnings.) Brown conceded that the results were disappointing and promised better results in 1985.

Interestingly, most of the divested regional Bell operating companies (RBOCs), which had been expected to do poorly, turned in surprisingly good results.

SHORT NOTES

IBM sold some 70,000 PC/ATs in 1984, according to Dataquest. However, Infocorp believes that just over 21,000 were sold through retail. Further shipments have apparently been suspended pending resolution of the hard disk problem.

In 1984 Big Blue earned $6.58 billion ($10.77/share) on revenues of $45.94 billion, a 14 percent increase. Net income was 20 percent ahead of the 1983 figure. CEO John F. Akers predicts the company will reach $180 billion in revenues in 10 years. This will require IBM to maintain a compound growth rate of over 18 percent a year.

Hydra Computer (Natick, Mass.), the main computer subsidiary of Encore, plans to soon enter the multimicroprocessor (MMP) fray in a big way. The company has been giving presentations to prospective customers over the past several months in preparation for formal system introduction that, as of press time, was scheduled to occur early this year. Speculation is rife that Sperry will surface as a major OEM for Hydra; Encore's C. Gordon Bell was recently named a member of a high-level technical advisory board to the Sperry Corp.

Onyx + IMI (San Jose, Calif.) lost its president, Fred Bialek, who left to join an unidentified startup. Carl Berg, who now becomes CEO, was a co-founder and now holds the major equity position in the company. Herb Martin, who headed Mercator Business Systems when Onyx acquired it in November 1982, becomes president. Onyx, which recently sold off its IMI disk division, was an early supplier of Z8000-based supermicros; Mercator had a line of 8086-based business supermicros.

Sequent Computer (Portland, Ore.) announced three distribution agreements for its Balance 8000, an NS32000-based MMP system. The distributors are First Computer, Westmont, Ill.; Group Three Electronics, Redondo Beach, Calif.; and Peripheral Systems, Winchester, Mass. □

Associate Editor Omri Serlin heads ITOM International, a Los Altos, Calif.-based research and consulting firm. He is the editor/publisher of FT Systems *and* Supermicro, *newsletters that cover developments in the computer industry. Mr. Serlin, who holds bachelor's and master's degrees in electrical engineering, has worked in the computer industry since 1962.*

Attention documentation writers:

You can get great typeset-quality output from your troff, ditroff or other UNIX™-based text files using the new Apple™ LaserWriter, a QMS Lasergrafix™ printer, or the Linotronic™ 300 and Linotron™ 101 typesetters.

So you can produce more attractive, effective manuals, data sheets, and more, faster—and for less money.

If you'd like your next project to take advantage of the next generation of *high* quality, *low* cost output devices, call. We'll send you literature on TRANSCRIPT,™ our conversion software that lets your UNIX system send files to any POSTSCRIPT printer. (The above-named printers and typesetters are POSTSCRIPT driven. We created POSTSCRIPT and TRANSCRIPT.)

TRANSCRIPT sells for $495 and will pay for itself many times over with a single manual.

The revolution in page printing.

ADOBE SYSTEMS INCORPORATED

415-852-0314

Please circle Ad No. 74 on inquiry card.

Trademarks: UNIX—AT&T Bell Labs; Lasergrafix—QMS, Inc.; TRANSCRIPT, POSTSCRIPT—Adobe Systems, Inc.; Apple—Apple Computer, Inc.; Linotronic, Linotron—Allied Corporation.

SYSTEMS ADMINISTRATION: CURES FOR BUSINESS ILLS

BY DR. REBECCA THOMAS

This article is based on A User Guide to the Unix System, *(second edition) by Dr. Rebecca Thomas and Jean Yates. Copyright © 1985 by McGraw-Hill Inc. Used with the permission of Osborne/McGraw-Hill.*

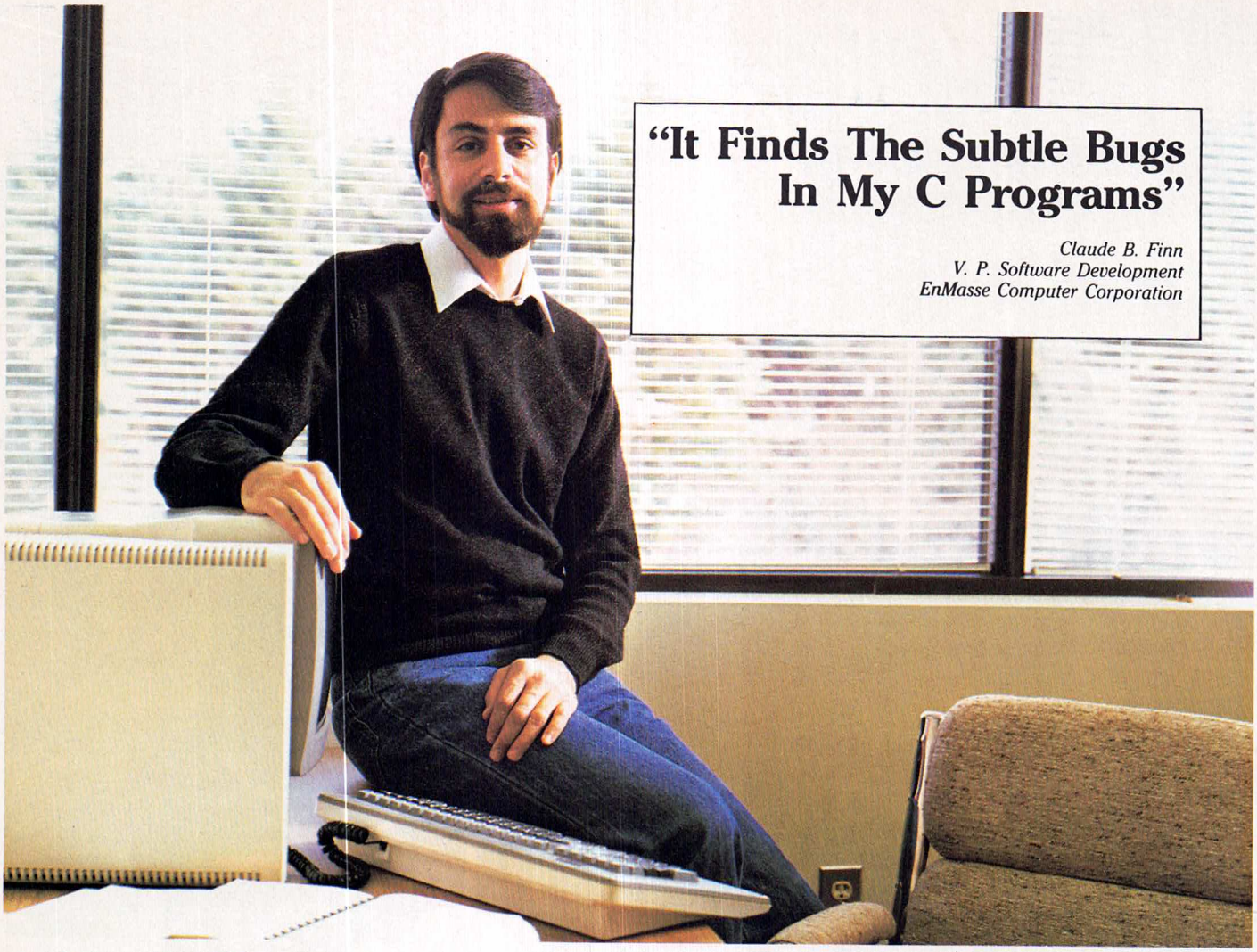
I'll bet you thought system administration was dull, boring, and *unimportant.* Not quite the stuff movies are made of. But have you ever wanted to grab your terminal around the throat because an important file has just "disappeared" from the system? Ah, just as I thought. You didn't take the time (or didn't know how) to back up that certain file that you spent hours working on and that is now raising your blood pressure. Now, I think you begin to get the picture: Perhaps this system administration business is a little more important than you first thought.

The basic problem ahead of you is this: The times are changing. In the old days, when the Unix system ran on DEC minicomputers, a guru or Unix system wizard was generally available on-site to help with tasks such as backing up the system. Today, though, watch out. If you're using the Unix system on a smaller microcomputer system, say a multiuser supermicro, *you* will probably be the one running and maintaining the system.

Beginning with this issue, we'll present a series of articles to help dispel the mystery surrounding administering your Unix system. In this installment, we kick off the series with an overview of what system administration is all about.

System administration generally includes any activity involving setup (configuration) and maintenance of your Unix system. Configuration includes such tasks as establishing and inactivating user accounts, enabling and disabling ports for user log-in, adding additional disks or mounting file systems, or perhaps updating your system with the latest Unix system release.

System maintenance includes the regular tasks of system startup,



"It Finds The Subtle Bugs In My C Programs"

Claude B. Finn
V. P. Software Development
EnMasse Computer Corporation

The SAFE C™ Family Can Literally Cut Software Development Time In Half. For UNIX™ and VAX/VMS.™

"Evasive bugs that use to eat up days — I'm finding them in minutes. Stray pointers, errant array indexes, parameter mismatches, misuse of string functions...I'm using Safe C automatic error detection to find them all."

Claude Finn is one of the many C programmers who have discovered that the Safe C family of software development tools dramatically enhances programmer productivity and improves software reliability and portability. Most Safe C customers have recouped their investment in these tools within the first month of active use. And with the security of Safe C their programmers are sleeping a lot easier!

The Safe C Family includes the Runtime Analyzer, Dynamic Profiler, Standalone Interpreter, English to C Translator and C to English Translator. They may be purchased separately or as a group.

CCA Uniworks, Inc.
Productivity Tools for Programmers
20 William Street • Wellesley, MA 02181

Please circle Ad No. 75 on inquiry card.

For more information or to place an order call our customer representatives at

800-222-0214

In MA (617) 235-2600
or mail this request form today.

Please send me information on:

- ☐ The Safe C Development Tool Family
- ☐ AI Development Tools
- ☐ CCA EMACS Unix/VMS Editor Environment
- ☐ Your complete line of state-of-the-art programming tools
- ☐ Please send license forms

Name _____
Title _____
Company _____
Address _____
City, State, Zip _____
Phone ( ) _____

CCA UNIWORKS, INC.
20 William Street • Wellesley, MA 02181

A Crowntek Company

UW585

file system backup, and system shutdown; emergency tasks such as repairing the file system after a system crash; and less routine tasks such as managing system disk space, checking system security, and obtaining reports of which users did what from the system accounting files.

Who should be responsible for system administration? If you are the sole user of your system, you will probably be administering it. On a larger multiuser system, one user is generally selected for this task. Someone consistent and responsible is the best choice for administrator—not necessarily the local "hotshot" system programmer. A system programmer's time is usually too valuable to be relegated to the more mundane chores of administering the system.

The administrator should have a basic understanding of how the Unix system is organized and how it works. Also essential is knowledge of the Unix file system, of how processes are created and destroyed, and of the common Unix system utilities and the shell command interpreter.

The administrator also must communicate well with users in order to find out their problems and to explain solutions. Administrators of larger systems are frequently asked not only to answer routine questions, but also to educate users on the fine points of using their system.

THE SUPERUSER ACCOUNT

The system administrator must be able to access any file and do any task within the capabilities of the Unix system. The *superuser* account, which has the account name `root`, allows such unrestricted access. Because other system accounts don't allow the necessary access, you must usually be logged on as `root` to do system administration duties.

However, you should be extremely careful when using this account because there are no system access restrictions. For instance, you can change or delete any file or kill any process on the system. Thus, you should carefully consider everything you do *before* you do it. Here are a few guidelines to help maintain your system safely:

> **Someone consistent and responsible is the best choice for administrator—not necessarily the local 'hotshot' programmer.**

First, use superuser powers as little as possible. Don't use this account for your routine work on the system; don't use this account when you are tired or fatigued.

Second, be especially careful when using commands and procedures that could irreparably change or damage the system. In particular, use due caution when using utilities such as `rm`, `mv`, `cp`, `fsck`, and `clri`. I recommend using the interactive option (`-i`) with `rm` (and with `mv` and `cp`, if available) to give yourself another chance in case of a typing mistake or error in judgment.

Third, keep a system log book or notes on-line in "readme" files. Other users may find such logs helpful for solving problems when you're unavailable. You will also find these records useful when tracking difficult or intermittent problems.

In the rest of this article, we highlight different aspects of system administration, topics we'll cover in greater depth in future installments of this series. But first let's consider the steps necessary for making your Unix system operational.

SYSTEM STARTUP AND SHUTDOWN

System startup involves several steps that are similar from one Unix system to another, but the details of each step may vary greatly from system to system. So before you plug in your computer system, consult the documentation that came with it. The details of system startup depend on your particular hardware configuration and system software implementation. The sidebar to this article lists the steps used in starting most Unix systems. Some steps may be automated. For instance, the first four steps are often performed by a read-only memory (ROM)-based program that begins execution as soon as power is applied or the system reset.

I can't overstate how important it is to back up your files regularly. Backing up a file means creating another copy of the data in the file. If the original data is damaged or lost, then you have the duplicate copy to work from. It's best to copy the files onto a storage medium different from your system disk. That way, if the system disk is damaged and the data on it no longer accessible, you still have a copy of your data elsewhere. When the system disk is repaired, you can restore your data from the backup copy.

The system administrator should periodically make copies of

STEPS FOR SYSTEM STARTUP AND SHUTDOWN

System Startup

- You apply power or reset a system that is already running.
- The computer hardware may perform hardware diagnostics, which are usually under direction of a program in ROM (read-only memory).
- The bootstrapping program is read into memory, or a ROM-based procedure is used.
- The bootstrap program reads in and starts the Unix system kernel program to bring the system up in single-user mode.
- You set the system real-time clock if necessary.
- You should check the file system, usually with the `fsck` program. If necessary, you must repair the file system before your Unix system is used.
- You bring the system up multiuser.

System Shutdown

- Log on the system console as the superuser.
- Warn other users that you'll be bringing down the system momentarily.
- Bring the system down to single-user mode.
- Perform any necessary file backup procedures.
- Power off, or reset and reboot.

the entire file system to prevent a major loss of information should a system failure occur. How often the system administrator should back up the system depends on how much file system activity there is. The more quickly files are created or changed, the more often you should back them up. Once a week should be often enough for a full system backup unless you are running a large facility with hundreds of users. Files created or updated (modified) during a work session should be backed up at the end of the work session or at least once a day on a larger system.

You usually can't use your system when a full file system backup is in progress, so the system administrator must perform these backups when demand for the system is lowest, say during the middle of the night or on weekends. Daily incremental backups should take place during the week, either before or after working hours.

Knowing how to shut down your system properly under all circumstances is perhaps the system administrator's most important task. Shutting down the system incorrectly (by pulling the plug, for example) can cause extensive damage; the Unix file system or even the system disk could be ruined. But all may not be lost. Carefully follow the system startup procedure to check and repair the file system properly with the `fsck` program.

As with system startup, the system shutdown procedure involves several steps that are similar from one Unix system to another. However, the details of each step may vary greatly from system to system. The sidebar outlines the general steps for shutdown. Depending on the design of your system, some of these steps may be automated. For instance, your system may have a `shutdown` shell script or program that performs all the steps outlined, except for turning off the power or pressing "Reset" for you.

Knowing how to shut down your system properly under all circumstances is perhaps the system administrator's most important task.

CREATING A NEW FILE SYSTEM AND MANAGING DISK SPACE

You may have to create a file system when you first set up your Unix system or if your present file system becomes corrupted beyond repair. You can frequently extend your existing "base" file system by mounting additional file systems, which effectively extend the file system hierarchical tree.

The first step in creating a file system on a new or previously corrupted disk is to format the disk to contain the file system. *Caution:* The formatting procedure erases any information on the disk. The disk-formatting program writes timing and address information so that the Unix system can locate any particular data block on the disk.

After formatting a disk, you use the `mkfs` command (`newfs` with Berkeley Unix) to create the file system. This command builds all the necessary data structures for the file system on the system disk. You'll need to consult your system documentation or system supplier for the parameters to use with `mkfs` when creating your particular file system configuration.

The complete Bell Unix operating system requires a large amount of disk space, some 7 to 8 Mbytes if you include the on-line documentation. Thus, most Unix systems installed on a microcomputer must have a minimum of 10 Mbytes of storage. However, it won't be long before even this storage fills up, especially if there are multiple users on the system.

When the system disk is full, you can no longer use your Unix system. In general, files accumulate because system users don't remove their unwanted files often enough. Also, some system "log" files grow as information is written to them. Thus, sooner or later the administrator will have to decide on the files to remove and perhaps archive onto backup media.

The Unix system provides several utilities for monitoring disk usage. First use the `df` (disk free) command to determine if enough disk space is available. If not, then notify the users by placing a warning message in the messages-of-the-day file, `/etc/motd`. (The contents of this file are displayed each time a user logs in to the system.)

Ask all users to run `du` to locate their files that are consuming a lot of disk space, and then ask everyone to erase unused files. If the files are important, have them archived off the system. Using the `quot` command, which summarizes disk usage on a per user basis, you (as the administrator) can spot users who aren't cooperating.

You can use the `find` command to locate files that are candidates for removal from the system. Look for large files that haven't been accessed for a while. You can erase some large files such as `a.out` and `core` (used in program development) because they can easily be re-created from the program source code if necessary. If no one has accessed these large files lately, then most likely the programmer probably forgot to remove them when he or she was done with them.

When using the superuser account, you should carefully consider everything you do before you do it. Remember: Look before you leap.

FIRST STEPS

If you are running your Unix system in multiuser mode, one of the first configuration steps will be to enable or perhaps disable terminal lines (or ports) for user log-in. If a terminal line is enabled, a user can log on to the system from that line; if it is disabled, logging-on is prevented. You will want to disable user log-on when you wish to use a terminal line for another purpose, such as sending output to a serial printer or communicating with other systems via a modem.

The procedure for enabling and disabling terminal lines for both Bell Version 7 and Berkeley Unix systems involves editing a file named `/etc/ttys`. For Bell Systems III and V, you must update the file `/etc/inittab`. We'll detail the procedures for each of these systems in a future installment.

Another configuration step is adding or deleting user accounts. When you get your system, several accounts will have already been established, but you will need to add an account entry for each new user. Also, certain special accounts may be useful for performing system-related functions.

Several steps are involved in adding a new account to a Unix system. First, you need to update the password file, `/etc/passwd`, and perhaps the group file, `/etc/group`. The password file contains information important to each account. For instance, one entry defines a *default group* membership for each account. You may define additional group associations using the group file.

The next step is to create a home directory for the new account. Such directories branch from the `/usr` directory in many systems. You might also set up a particular working environment for the new account. You generally do so by installing shell startup files in the home directory. These files are automatically read and the commands therein executed each time the account is logged into.

A user account is not deleted from the system; instead, a user account is inactivated so that any files created by that account user will not pass in ownership to another account user who happens to be assigned the same user identification number. The administrator should back up the files belonging to the old account and then remove them from

the system disk. Finally, delete the home directory for the old account.

Another step in setting up your system will be to configure the line printer spooling system, which allows for background printing of files. Files to be printed are placed in a special *spool* directory. Then a special program known as the *line printer daemon* examines the spool directory and prints the files.

SECURING THE SYSTEM

Computer system security means protecting the computer hardware and the information contained within the system. Threats to system security come not only from outside the computer system but from inside the community of users as well.

Outside threats to system security include such things as unauthorized access to the computer system; unauthorized examination of the computer system's output, whether hard copy or magnetic media; unauthorized tapping of data being transferred between computer systems over phone lines, microwave links, and the like; and damage from fire, electrical power surges and outages, and natural disasters.

Some ways to minimize these problems are to isolate the computer system; properly dispose of printed output and archive magnetic media in a vault; encrypt sensitive data (that is, put it into a code indecipherable to others); and protect the system with fire and flood alarms, electrical surge protectors, etc.

Inside threats are more insidious. They might include destruction of software data (either by mistake or on purpose), examination of sensitive data by unauthorized users, or alteration of sensitive data without detection.

The Bell Unix system provides several safeguards for system security, including password-controlled system access, control of access to individual files, encryption of data files, and system accounting functions for analysis of which users did what. But the Unix system was designed for facile exchange of information between users. As a result, some security loopholes exist in most Unix systems. In a future issue we'll take a closer look at some of these loopholes and how to "plug" them.

A FEW POINTS

If you're installing an updated Unix system, you should consider a few points. If you get a new kernel, your old programs may quit working because the new system may have slightly different system calls or data structures. If so, change the source code as necessary and recompile.

Some Unix system kernels and other special programs write system usage data to special accounting files. Accounting programs and shell scripts on some systems can process this raw data into summary files and reports. These reports describe process resource usage, records of user log-ins, setting the real-time clock, system boots and shutdowns, disk usage for each user, and other information useful to the system administrator.

You can automate the accounting system functions by placing the appropriate commands in files that are executed during system startup, shutdown, or at periodic intervals. Then the accounting system will be easy to maintain.

The related system activity package reports system resource usage. This includes central processing unit (CPU) usage, input/output (I/O) activity, buffer cache usage, system calls invoked, file access activity, and more.

Bell Laboratories provides reference manuals for the Unix system. Volume 1 contains information about many of the Unix system utilities used for system administration tasks. Starting with Bell System V, the system administration utilities are documented in a separate volume, the *Unix System Administrator's Manual.* In addition, the *System Administrator's Guide* describes some maintenance procedures more thoroughly than does the reference manual.

Many systems have the reference manuals available for immediate perusal on-line. However, many microcomputer implementations, especially those with only a 10-Mbyte disk, do not have enough disk space to maintain the manuals.

You see, there is very little mystery surrounding the Unix system after all. By now you should have a better idea of what's involved in administering your Unix system. A system as large and complex as the Unix system requires constant vigilance to maintain it, a task this series will help you accomplish. In future installments of this series, we'll delve even further into the details of system administration. □

Dr. Rebecca Thomas, UNIX/WORLD's Editor Emeritus, is an author of A User Guide to the Unix System, *the second edition of which is now available. She is currently writing a book on Unix system administration.*

Acknowledgment

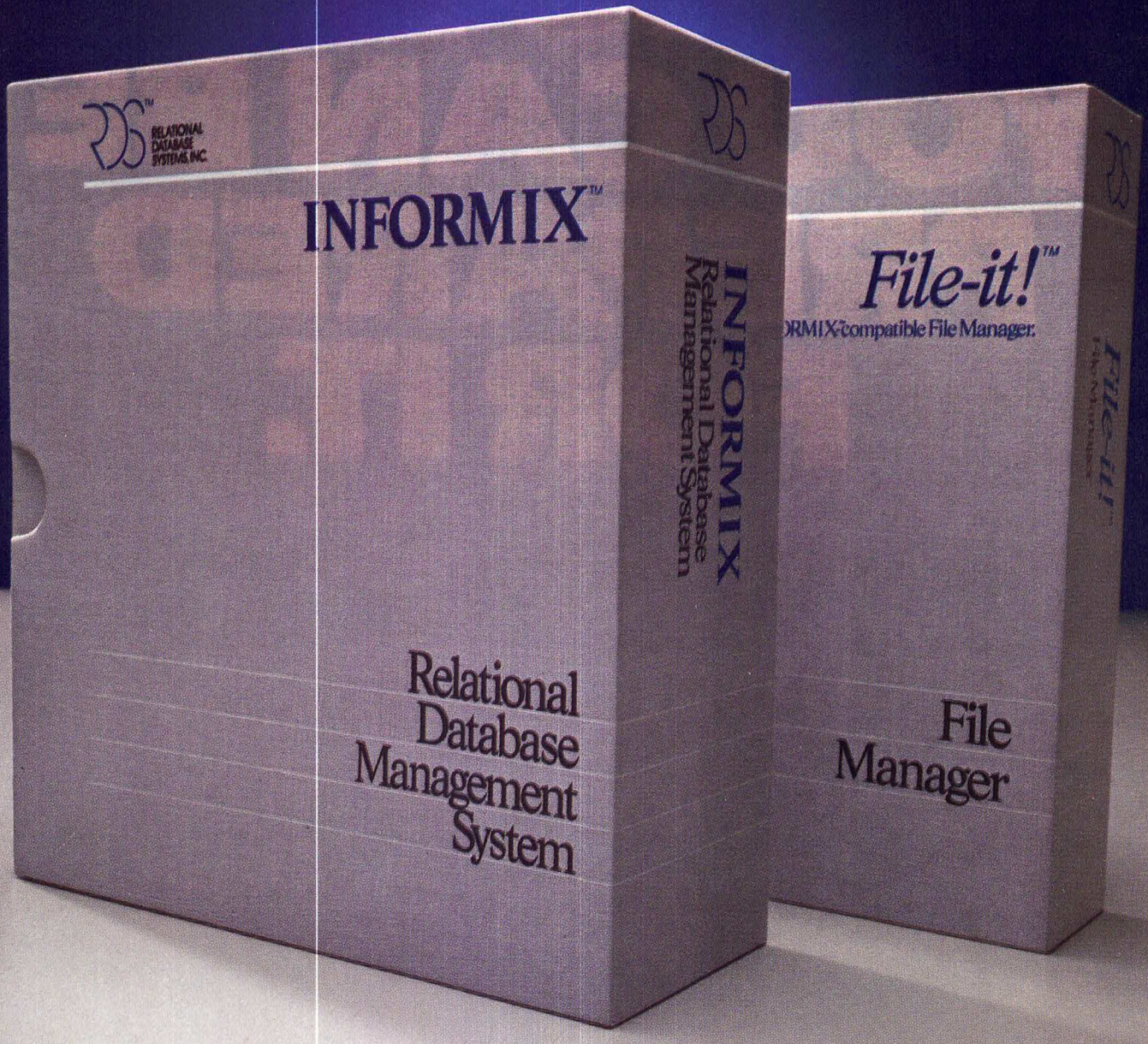
I'd like to thank Rik Farrow and Nancy Blachman for taking their time to discuss various aspects of system administration.

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.

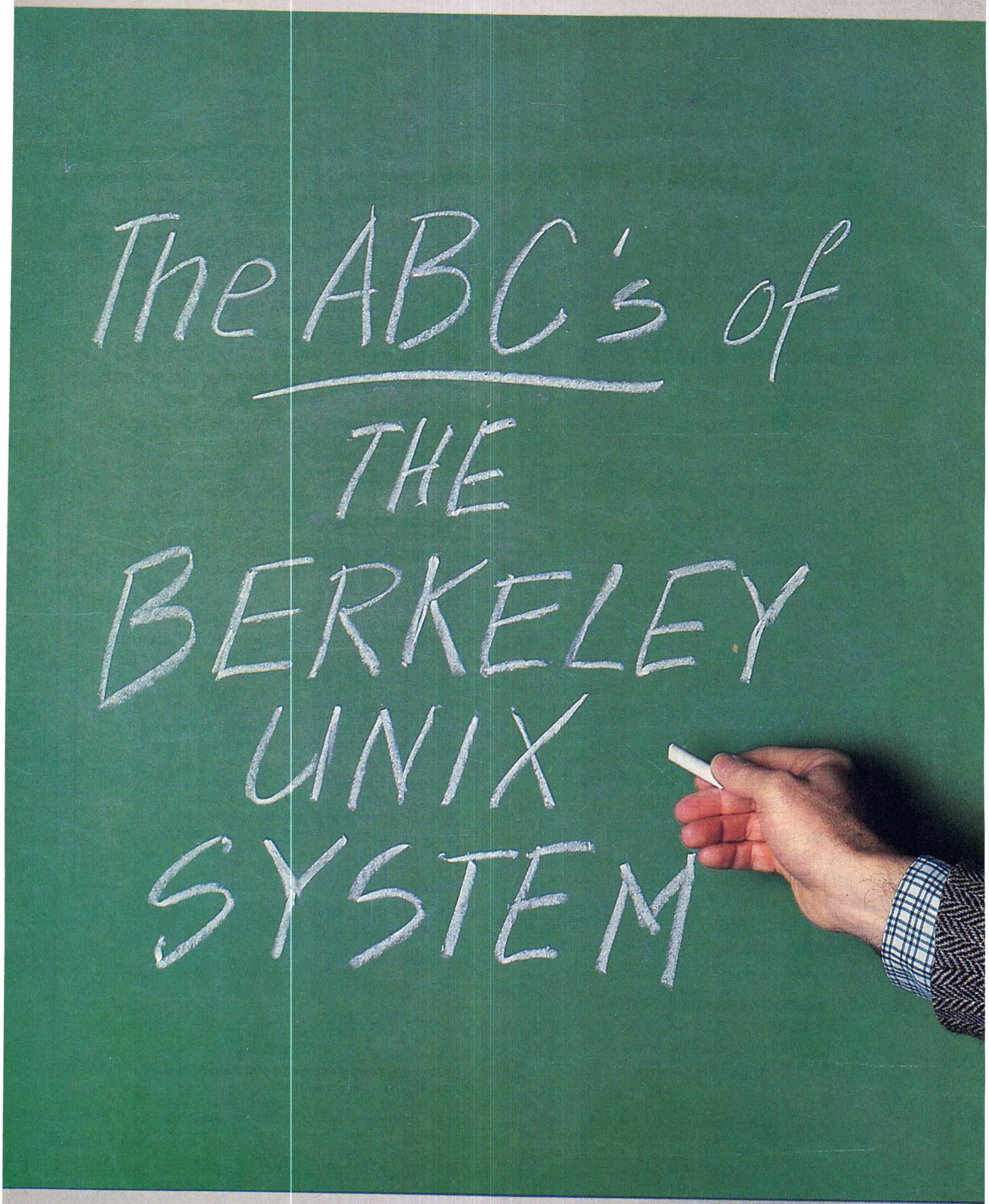
*RDS products are available for the following systems:

- 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


The ABC's of
THE
BERKELEY
UNIX
SYSTEM

Thinking of buying a 4.2BSD system? Or do you already have one and need help? If so, this article will help you avoid the pitfalls.

BY NANCY BLACHMAN
AND PHIL NGAI

You've heard many times before, on the pages of this magazine and elsewhere, that Berkeley 4.2BSD is a different animal from most other versions of the Unix system. So if you are considering buying a 4.2BSD system, or already have one, this article is for you. Over the next several pages we'll discuss installing and maintaining a 4.2BSD system. We'll also explain how to communicate with your users and with the Unix system community at large. Let's get right to it.

Let's start with some of the software on 4.2BSD (Berkeley Software Distribution) such as `sendmail`, `talk`, `script`, and `biff`, which require special attention when you install them.

As distributed on 4.2BSD, `sendmail` does not compile because it assumes the presence of the Source Code Control System (SCCS) utilities, which are not part of the 4.2 distribution. If you don't have Bell System V source code or don't want to install SCCS, then you must change the `sendmail` source. To do this, simply remove the line `define(m4SCCS)dnl`, located at the end of the file `/usr/src/usr.lib/sendmail/md/config.m4`.

If you now recompile and install `sendmail`, your users will see a nonfatal warning message concerning `/dev/console` whenever they send electronic mail. Although this message conveys no problems, you can eliminate it by removing the -DEBUG flag from the assignment to CFLAGS in `/usr/src/usr.lib/sendmail/lib/Makefile`.

If your site's only link to other hosts is through `uucp` connections and if timely delivery of `uucp` mail is important, you will want to remove `-r` from the invocation of `uux` in `/usr/src/usr.lib/sendmail/cf/uucpm.m4`. That is, change the line `A=uux - -r $h!rmail ($u)` to `A=uux - $h!rmail ($u)` and then invoke `make` in that directory.

When `uucp` mail fails, you might have difficulty determining what caused the problem and where it occurred.

Finally, issue the command `cp uucpproto.cf <hostname>.cf`. These changes cause `sendmail` to invoke `uux` in a mode where it immediately calls up the neighboring site to which the mail was routed. Note that a heavily loaded site might prefer to postpone the call until a regularly scheduled time.

`uucp` MAIL PROBLEMS

When `uucp` mail fails, you might have difficulty determining what caused the problem and where it occurred. One way to make it easier to locate the problem is to change the `sendmail` configuration file so that `sendmail` identifies itself as `AMD-MAILER-DAEMON` instead of `MAILER-DAEMON`. Although a change such as this identifies which mailer daemon rejected the mail, `sendmail` has special code to prevent two mailer daemons from rejecting each other's rejections. Such a change breaks that code.

A much better way to make your `sendmail` rejection unique is to change the string `Mail Delivery Subsystem` to `Mail Delivery Subsystem at amd` (for a host named `amd` in this case). This string is in parentheses, designating a comment, so this change should not affect the operation of the electronic mail system.

You can implement this in a quick and dirty way by changing the string in the file `/usr/src/usr.lib/sendmail/src/savemail.c` in the procedure `returntosender` and then recompiling and re-installing `sendmail`. An even better approach, one that makes the code more portable, is to replace the host name (`amd`, for example) with the value that the `gethostname()` system call returns.

As distributed, `sendmail` is paranoid about logging problems during delivery. The `syslog` daemon performs a `sync` operation every time the system accesses it, and it writes the logged information in memory safely out to disk. The benefit of having the log written out as soon as possible is not worth the additional load the `sync` operation causes. Most sites run the `update` daemon, which does a `sync` every 30 seconds anyway.

Therefore, we eliminated one `sync` system call in `syslog` by commenting out the call to `sync()` in `/usr/src/usr.lib/sendmail/aux/syslog.c`. This occurs immediately following the comment `/* let's be paranoid... */`. Also, `syslog` performs a `sync` when it is terminated, and because this doesn't happen often, we left this `sync()` call in the program.

Here is a hint if you make changes to sendmail source: Be sure you are changing Makefile.m4 and not Makefile in the src and aux directories. The Makefile in /usr/src/usr.lib/sendmail uses the m4 macro preprocessor to generate Makefile from Makefile.m4 in those directories.

talk, script, biff, AND cpio

The talk command replaces the write command of earlier Unix system implementations. This command program uses the new interprocess communications (IPC) facilities of 4.2BSD to provide two new features. First, two users on different machines networked together can communicate using talk; second, talk is easier to use. Each user's screen is divided in half, with the locally entered text displayed on the top half of the screen and the remote user's text displayed on the screen's bottom half. Both users can thus type at the same time without mixing their characters.

Sites with only one machine or whose machines aren't networked together might still want to set their systems up to allow interprocess communications. In this case, two steps are necessary to make talk work properly. First, add your system's name as an alias to localhost in /etc/hosts (for example, 127.1 localhost resonex). Second, run the command MAKEDEV pty0 while in the /dev directory.

The 4.2BSD script command is much "cleaner" than that of previous implementations because it uses IPC. script also requires ptys to work, as does the talk program.

The biff system is an accessory to the electronic mail system. (biff is an acronym for "Be notified IF mail arrives and who it is From.") A user enables notification by issuing a biff y command, and from then on a brief message appears on the user's terminal whenever new mail arrives. The comsat daemon does the work. The biff system works only if the word localhost is in /etc/hosts. Our first attempt to make talk work involved substituting resonex for the string localhost; this fixed talk but broke biff. Both localhost and resonex are necessary to run talk and biff (see Figure 1).

The 4.2BSD script command is much 'cleaner' than that of previous implementations because it uses IPC.

To read the contents of AT&T's System III or V distribution tape, you need the cpio command, which is distributed on that tape. You thus have a "Catch-22" situation. We recommend that you get a copy of cpio from another system rather than booting the System V distribution tape to obtain cpio.

Please note: To compile the cpio source on a system running the BSD Unix system, you need to change all occurrences of ushort to unsigned short. And because 4.2BSD does not support FIFOs (First In First Outs), you should remove the code that deals with them from the version of cpio that you run.

```
#
# Resonex Host Database
#
127.1                   localhost resonex
```

FIGURE 1: /etc/hosts

FORMATTING DISKS AND ORGANIZING FILES

If you are concientious, you will format disk drives by using the "severe burn-in" option to format, which locates as many bad sectors as possible on the disk. With this option, the system formats one track at a time by writing the appropriate headers and a test pattern. It then checks the sector by reading and verifying the pattern.

During severe burn-in, the system makes 46 passes using different patterns, which takes considerable time. For instance, formatting a 404-Mbyte Fujitsu Eagle disk drive takes about 13 hours with this option. Although the "severe burn-in" option is advantageous in the long run, it can be painful in the short run because you can't use the system while it is formatting a disk.

When you organize your file system, you should make /usr/spool and its subdirectories a mounted file system if you are on USENET or if you store many unimportant files in this directory. This approach has two benefits. First, you can choose not to back up /usr/spool. Because the dump program backs up in units of file systems and not directories, making /usr/spool a separate file system will avoid "dumping" it. Second, this approach reduces the activity in the /usr file system. Because only file systems that are active during a system crash can become inconsistent, you thus reduce the chances of corrupting /usr.

We also recommend that you make /tmp a mounted file system if you have more than one disk. As an example, place the root file system on the first partition of the first disk and place /tmp on the first partition of the other disk. The first partition is typically too small to be useful for general use, but it is a good size to use for the small /tmp directory. Also, the root file system requires a small amount of free space. Therefore, mounting /tmp on the first partition of the second disk moves /tmp from a crowded root file system to an uncrowded file system (its own).

Besides using file system space more efficiently, this proposal helps balance the load between all the disk drives. Many programs, such as the C compiler and sort, use intermediary files. Because temporary files such as compiler intermediary files are normally stored in tmp, this directory gets more than its share of file system activity.

The root file system is also quite active because many frequently used programs are in /bin and because pipes keep their data in the root directory, /. Therefore, if you have two disks, mounting /tmp on the second disk helps equalize the demand placed on the disks by what are often the system's two most active file systems.

Finally, we recommend that you do not have a real /usr/tmp. Instead, make a symbolic link from /usr/tmp to /tmp, creating one large temporary storage area instead of two half-size areas. By doing so, you can take advantage of the following law of large numbers from probability theory: "As the number of trials increases, the probability that the average number of successes deviates from the expected value by more than any pre-assigned epsilon tends to zero."

Here the law of large numbers tells us that, if there is a probability p that x many programs require y bytes of scratch space, then the probability that $2x$ many programs will require more than $2y$ bytes of scratch space is smaller than p. Most programs use /tmp instead of /usr/tmp, and you should therefore make a link from /usr/tmp to /tmp, instead of the other way around. Less searching of system tables is necessary to locate files this way.

THE dump AND restor BACKUP PROGRAMS

The dump and restor programs supplied with 4.2BSD are a major improvement over previous versions. The former restoration methods were painful for two reasons—first, you had to know the exact destination pathname ahead of time; and second, the restor program's output was a set of files with names corresponding to their inode number. Consequently, if you were restoring a directory hierarchy, you had to move (rename) the files into their proper locations, either manually or with a shell command file or script.

The 4.2BSD restor program has an interactive mode in which you can perform the equivalent of the ls and cd commands. When you have created a list of files to retrieve and have told restor to proceed, it places the files in their proper locations in a directory hierarchy relative to the directory where you performed the retrieval step.

If, for example, you were restoring the /usr directory and ran restor in /usr, the files would be restored in their original places. However, if you restored /usr by running the restor program in /old, the files would be restored relative to /old; /usr/adm/wtmp would be retrieved as /old/adm/wtmp.

Because these backup utilities are so easy to use, it is reasonable to allow users to back up and restore their own files. Such a site would need a "friendly" user community, though, because a user could just as easily retrieve someone else's files, thus violating the file system protection mechanism.

THE 'CRON' TABLE FILE

The "cron" table (basename crontab) is a file of commands that are run periodically by the cron daemon. The table tells cron when to run these commands. Rather than having an entry for each command to be executed, however, we find it more convenient to specify shell scripts for each periodicity.

Figure 2 shows such a "cron" table file. The file cron-weekly is executed weekly, cron-daily daily, cron-semi-hourly 12 times per day, cron-hourly hourly, and cron-bi-hourly twice an hour. Note that cron-semi-hourly is only called 11 times from crontab because cron-daily calls cron-semi-hourly for the twelfth invocation. Similarily, cron-hourly is invoked only 12 times a day from crontab. It is called the other 12 times by cron-semi-hourly. Figure 3 shows examples of the /usr/lib/cron/cron* files that we use.

Having crontab organized as we have described has several advantages over using separate entries for each file to be executed—

```
0    0              * * 1          /usr/lib/cron/cron-weekly
0    0              * * 2,3,4,5,6,7 /usr/lib/cron/cron-daily
0    2,4,6,8,10     * * *          /usr/lib/cron/cron-semi-hourly
0    12,14,16       * * *          /usr/lib/cron/cron-semi-hourly
0    18,20,22       * * *          /usr/lib/cron/cron-semi-hourly
0    1,3,5,7,9      * * *          /usr/lib/cron/cron-hourly
0    11,13,15,17    * * *          /usr/lib/cron/cron-hourly
0    19,21,23       * * *          /usr/lib/cron/cron-hourly
30                * * * *          /usr/lib/cron/cron-bi-hourly
```

FIGURE 2: A TYPICAL `crontab` FILE

namely, `crontab` is short and easy to read. Also `cron` executes commands in the tables sequentially, in the order they are listed in the file. Our method thus avoids poor response time every hour on the hour because `cron` executes only one program at a time.

Normally you would set up `cron` to print the time and date on the system console every 10 minutes to help you determine the approximate time at which console messages were printed. If you have a temperature recorder and the system crashes, for example, correlating the time of the crash with the time the air conditioning failed or with the time of the blackout can be helpful.

The intermixing of time and date with error messages and log-ins on the console can be confusing. The console output will be less confusing if you place several tabs before the time and date and move that information to the far right side of the paper. Figure 4 shows sample console outputs before and after we made such a change.

LOCAL ADDITIONS TO YOUR UNIX SYSTEM

We recommend that you keep local additions and changes to the system in a separate directory, typically `/usr/local` or `/local`. That way, when you update your system, you can easily move the local changes to the new system because one directory contains them all. Also, depending on your file system layout (whether `/usr/local` is a mounted file system), you may not need to back up `/usr/src` because all your local configuration changes are in your local directory and not in `/usr/src`.

We do not recommend that you put a separate manual directory (`man`) in your local directory because there are many programs that assume that the manual pages—for example, `apropos`, `whereis`, `whatis`, `catman`, and `man` itself—are in `/usr/man`.

In addition, most systems come with their own on-line manual pages, which are automatically copied to `/usr/man`. If you install the subsystem in `/usr/local`, you don't need to transport the associated files in `/usr/man` when you install a new release of the operating system. This is so because the `man` pages are also in `/usr/local`.

Note finally that the following subdirectories, among others, are useful to have in your local directory: `bin`, `src`, `lib`, `include`, and `doc`.

MANAGING DISK SPACE

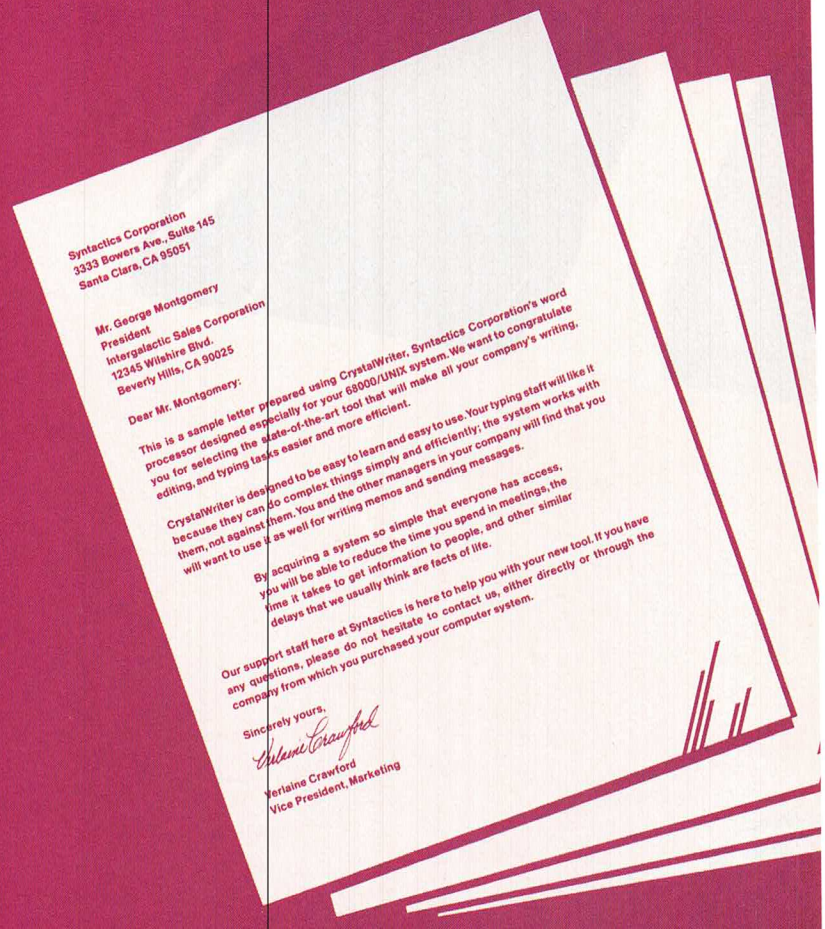
Figure 5 shows a list of the files in the directory `/usr/adm` that grow continually. You should truncate or "age" these files periodically to ensure that they do not take over the entire file system. (Note that you may have additional files associated with special hardware.)

We have a simple shell script (`agelog`) which, when given an argument, `filename`, renames it to `filename.1`. (Figure 6 shows the `agelog` shell script.) If `filename.1` exists, `newlog` first renames it `filename.2` and continues this process until `filename.7` is reached. Any pre-existing `filename.7` is deleted. We run `agelog` every night on `uucp`, `news`, and some other "log" files to keep a week's worth of "log" files one day apart.

We also run a shell script `diskhogs-weekly` (see Figure 7), which generates and posts a report of disk usage every week. The report is sorted by disk usage, with the biggest users at the top of the list, and it is posted to a local newsgroup. This helps determine who is using an unreasonable amount of disk space and takes advantage of peer pressure to maintain disk usage at a reasonable level.

CrystalWriter™

Complete Word Processing for the UNIX™ System



"...one of the best kept secrets in the world of UNIX System application software..."

"...a truly superior package..."

UNIX/WORLD

AVAILABLE NOW, CrystalWriter from SYNTACTICS is the complete word processing system designed to support the first time user– without ignoring the experienced word processor.

For more information on CrystalWriter, call today

800-626-6400
(In California (408) 727 6400)

SYNTACTICS
3333 Bowers Avenue, Suite 145
Santa Clara, CA 95054

SYNTACTICS™

Please circle Ad No. 39 on inquiry card.

UNIX IS A TRADEMARK OF AT&T BELL LABORATORIES. SERIES 32000 IS A TRADEMARK OF NATIONAL SEMICONDUCTOR CORPORATION.

"What's on your mind, Jerry?"

"What's the problem?"

"And, as usual, they need it yesterday."

"Can we do it?"

"What's the risk?"

"I'm worried. Marketing wants us to develop a UNIX based system that's way ahead of everyone else."

"They want more than just another microcomputer. They want a supermini at a micro price."

"Which means we have to develop the hardware and software at the same time. And it's got to have virtual memory and full floating point operations."

"Only if we use the Series 32000 microprocessor family."

"The only risk is if we don't use it."

Right this minute, more than 300 of your competitors are using the Series 32000 to beat you to market.

Of all the 32-bit microprocessors available today, only one delivers the total solution that gives your new products the competitive edge. Ours.

That's why almost half the Fortune 500 companies planning 32-bit products are already using the Series 32000™ family. To say nothing of the hundreds of smaller companies also using Series 32000 in pursuit of new markets. Like your market.

Because our complete Series 32000 microprocessor family is ready today, it reduces the cost and speeds the pace of system development. And because it performs like a supermini and runs true UNIX™ software, Series 32000 will put your new products far ahead of the competition. The only worry you'll face is if you don't use it.

For all the facts, write to:
Series 32000, MS 23-200
National Semiconductor
2900 Semiconductor Drive
P.O. Box 58090
Santa Clara, CA 95052-8090

THE SUPERMINI ON A CHIP

National Semiconductor
We're doing it.

Please circle Ad No. 42 on inquiry card.

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

A. /usr/lib/cron/cron-weekly

```
#! /bin/sh
find /usr/spool/uucp -mtime +14 -exec rm {} \;
find /tmp -mtime +14 -exec rm {} \;
su news < /usr/lib/news/news-weekly
/usr/adm/diskhogs-weekly
```

B. /usr/lib/cron/cron-daily

```
#! /bin/sh
# run this script once a day (at midnight)
/usr/lib/cron/cron-semi-hourly
cd /usr/adm; adm-daily; newsyslog
cd /usr/lib/news; /bin/su news < news-daily
cd /usr/lib/uucp; /bin/su uuclerk < uucp-daily
/usr/bin/calendar -
cd /usr/spool/batch; /usr/bin/find . -mtime +14 -exec rm {} \;
cd /usr/spool/uucp; /usr/bin/find . -mtime +21 -exec rm {} \;
/usr/bin/find / -name 'core' -mtime +7 -exec rm {} \;
/usr/bin/find / -name 'a.out' -atime +7 -exec rm {} \;
```

C. /usr/lib/cron/cron-semi-hourly

```
#! /bin/sh
# run this script every other hour
/usr/lib/cron/cron-hourly
cd /usr/lib/news; /bin/su news < news-hourly
cd /usr/lib/uucp; /bin/su uuclerk < uucp-hourly
```

D. /usr/lib/cron/cron-hourly

```
#! /bin/sh
# run this script every hour
/usr/lib/cron/cron-bi-hourly
cd /usr/lib/uucp; /bin/su uuclerk < uucp-hourly
```

E. /usr/lib/cron/cron-bi-hourly

```
#! /bin/csh
# run this script every 30 minutes
(echo '                    ' `date`; echo '^M') >/dev/console
/usr/adm/adm-bi-hourly
/usr/lib/atrun
```

FIGURE 3: TYPICAL /usr/lib/cron FILE ENTRIES

```
BEFORE

Fri Nov  2 23:00:08 PST 1984
Fri Nov  2 23:10:08 PST 1984
machine check 2: cp tbuf par fault
tbuf par: flushing and returning
mcr0: soft ecc addr 23c4 syn 1
Fri Nov  2 23:20:08 PST 1984
mcr0: soft ecc addr 23c4 syn 1
Fri Nov  2 23:30:08 PST 1984
mcr0: soft ecc addr 23c4 syn 1
hp2g: soft ecc sn382424

AFTER

                                        Fri Nov  2 23:00:08 PST 1984
                                        Fri Nov  2 23:10:08 PST 1984
machine check 2: cp tbuf par fault
tbuf par: flushing and returning
mcr0: soft ecc addr 23c4 syn 1
                                        Fri Nov  2 23:20:08 PST 1984
mcr0: soft ecc addr 23c4 syn 1
                                        Fri Nov  2 23:30:08 PST 1984
mcr0: soft ecc addr 23c4 syn 1
hp2g: soft ecc sn382424
```

FIGURE 4: SAMPLE CONSOLE OUTPUT

```
-rw-r--r--  1 root        193824 Oct 25 23:23 acct
-rw-r--r--  1 uucp         49075 Oct 25 20:11 aculog
-rw-r--r--  1 root          3668 Oct 25 23:00 lastlog
-rw-r--r--  1 root             0 Jul  3 12:05 lpd-errs
-rw-r--r--  1 root         59742 Oct 24 15:40 messages
-rw-r--r--  1 root          4096 Oct 24 15:40 msgbuf
-rw-rw-rw-  1 root         52808 Oct 25 04:30 savacct
-rw-r--r--  1 root          8414 Oct 14 23:02 shutdownlog
-rw-rw-rw-  1 root          3920 Oct 25 04:30 usracct
-rw-r--r--  1 root        395712 Oct 25 23:01 wtmp
```

FIGURE 5: SOME GROWING FILES IN `/usr/adm`

COMMUNICATIONS

Electronic communication provides a way of passing useful information both within a system and outside it. Users can communicate with other users on the system in various ways. You can use the `wall` command, for example, to write to all users. The superuser uses this utility to send messages immediately to all users who are currently logged on.

The `msgs` system keeps only one copy of the message on the system and allows you to update or remove a message easily. On a "vanilla" 4.2BSD system, these messages expire after three weeks, and `cron` removes them.

You can use the Berkeley `Mail` command to send electronic mail to all users or to a subset of users.

The contents of the message of the day file, `/etc/motd`, appear whenever a user logs into the system. If a file named `.hughlogin`

```
#! /bin/sh
# Age log.  Keeps previous days version of file for a week.
if test ! -d OLD
then
        /bin/mkdir OLD
fi
/bin/rm OLD/$1.7
/bin/mv OLD/$1.6  OLD/$1.7
/bin/mv OLD/$1.5  OLD/$1.6
/bin/mv OLD/$1.4  OLD/$1.5
/bin/mv OLD/$1.3  OLD/$1.4
/bin/mv OLD/$1.2  OLD/$1.3
/bin/mv OLD/$1.1  OLD/$1.2
/bin/mv OLD/#1.0  OLD/$1.1
/bin/mv $1    OLD/$1.0
/bin/cp /dev/null $1
/bin/chmod 666    $1
```

FIGURE 6: THE agelog SHELL SCRIPT

```
#! /bin/sh
#  Figure out how much disk usage and then list who is hogging space
cd /u
du -s * | sort -rn |\
 /usr/lib/news/inews -n amd.general -t "Disk space usage for AMD"
```

FIGURE 7: THE diskhog-weekly SHELL SCRIPT

exists in their home directory, however, the display is suppressed.

As mentioned above the talk command is a screen-oriented program that copies characters from your terminal to that of another user. write, an earlier BSD command, also copies lines typed at your keyboard terminal to another terminal. However, unlike talk, it does not split the screen in half to separate what each user has typed—one user's output is mixed in with the other user's output if they type simultaneously.

COMMUNICATION WITH OTHER MACHINES

Because many people using the Unix system might already have solved the same problem now facing you, electronic mail is often the best way to communicate with them. Programmers sometimes work unconventional hours and are hard to reach by phone, but electronic mail is a way around the "telephone tag" problem. It also gives programmers time to research a problem and devise a solution. In addition, electronic mail allows you to send software or bug fixes between hosts quickly and easily. For these reasons, we recommend that you set up uucp between your system and other Unix systems.

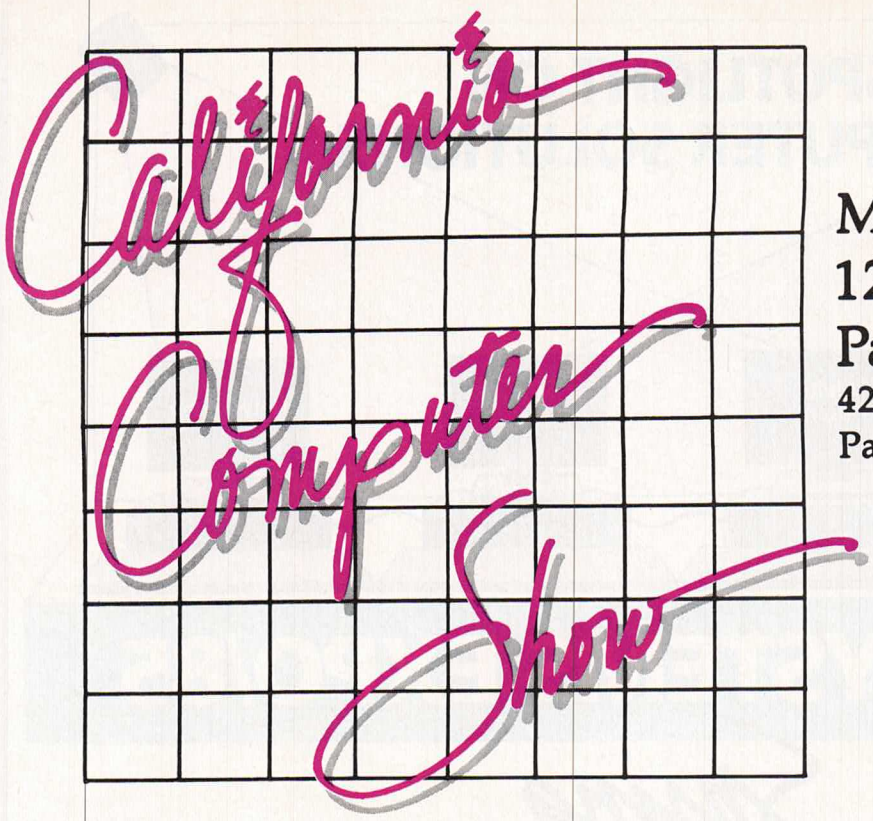
What if you are a new system administrator and do not know people on other sites who can help you? Then you should join USENET (see UNIX/WORLD, Vol. 1, No. 7); it is an electronic bulletin board that requires several "news" utilities for sending or receiving messages. The bulletin board is divided into news groups, such as for a specific host, sets of hosts, geographical region, or worldwide discussion groups.

USENET is a useful source of both technical and nontechnical information. Many of the participants are friendly, open, and willing to share information. This atmosphere probably dates from USENET's beginnings, when most of the sites were at universities and research institutions such as Bell Labs and UC Berkeley.

In this article we have covered some important areas of 4.2BSD system administration, including installation, maintenance, and commu-

nication. There is always some information you have to omit when writing an overview of a subject, but this article should certainly get you started. Keep it handy and refer to it when questions arise. □

Nancy Blackman ({allegra, hplabs, ihnp4, ucbvax!sun} !resonex!nancy) *designs software at Resonex, a Sunnyvale, Calif.-based firm that develops medical-imaging equipment. Phil Ngai* ({ucbvax, ihnp4, decwrl}!amdcad!phil) *received his degree in electrical engineering from M.I.T. and now works as a network engineer at Advanced Micro Devices, also in Sunnyvale, Calif.*



May 14th - 15th, 1985
12:00pm - 6:00pm
Palo Alto Hyatt Hotel
4290 El Camino Real
Palo Alto, California

Northern California's only TWO-DAY event dedicated exclusively to OEMs and sophisticated end users.

Featuring the latest in:

- CAD/CAM Systems
- Graphics Systems
- Engineering Workstations
- Computers and Peripherals
- Data Communications
- Unix

Exhibiting Companies and Product Lines:

- AT & T
- Daisy
- DEC
- Fujitsu
- IBM
- Intel
- Mentor Graphics
- Metheus
- Motorola
- Perkin Elmer
- Sun Microsystems
- Tektronix
- Valid Logic
- Versatec

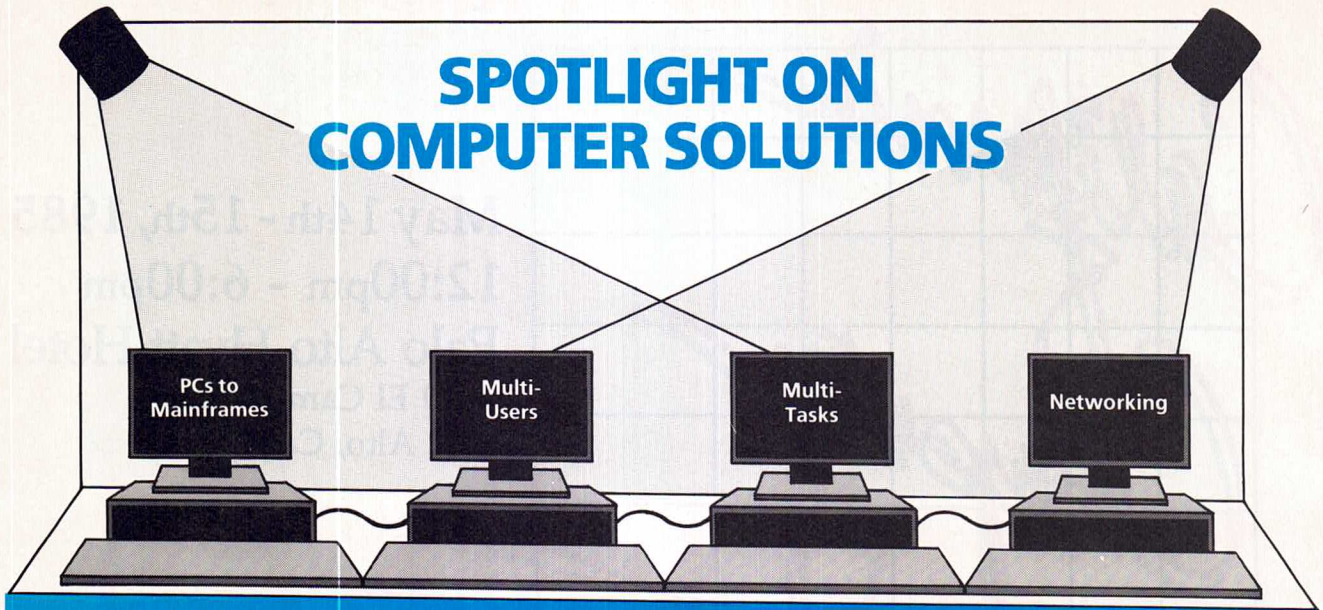
and many more...

Call or write for a complimentary invitation to:

NDN NORM DE NARDI ENTERPRISES
289 So. San Antonio Rd.,Suite #204 • Los Altos, CA 94022 • (415)941-8440

Please circle Ad No. 28 on inquiry card.

SPOTLIGHT ON COMPUTER SOLUTIONS



UNIX* SYSTEMS EXPO/85

Spring

BE THERE!
For Computer Solutions To

- Office/Factory Automation
- Engineering/Scientific Applications
- Text Processing
- Database Access

BE THERE!
To Learn About

- Linking PCs to MainFrames
- Multi-User/Tasking Operations
- Networking
- Clusters/Device Sharing

April 24–26
San Francisco Moscone Center

See exhibits from the leading and most innovative hardware and software companies. Attend the full conference program with over 40 user/marketing oriented sessions.

Don't Miss UNIX Systems Expo/85 for:

- Engineers
- Scientists
- Corporate Users
- VARs
- Software Developers
- Vertical Marketers
- Systems Integrators
- Distributors
- Dealers
- Retailers

An exclusive production of
Computer Faire, Inc./
A Prentice-Hall Company
617/965-8350
415/364-4294

UNIX* SYSTEMS EXPO/85 *Spring*

Please circle Ad No. 147 on inquiry card.

*UNIX is a trademark of AT&T Bell Laboratories

UNIX/WORLD

1 YEAR $36 NOW $18

☐ **YES!** Start my 1-year subscription (12 issues) today. Bill me later for only $18, that's 50% off the newsstand price.

BUY TWO YEARS GET ONE FREE

☐ **YES!** Count me in on your limited offer to buy two years and get one free.

Start my three-year subscription (36 issues) right away and bill me for only $36—the two-year price.

Name

Title/Company

Address

City State Zip

Canadian residents add $6. Other Foreign residents add $20 for International Postage. Both Canadian and Foreign orders must be payable in U.S. Funds on U.S. Banks only.

SPECIAL OFFER!

SEND NO MONEY! MAIL TODAY

UNIX™ is a trademark of AT&T Bell Labs. UNIX/WORLD is not affiliated with AT&T Bell Labs.

EE5D25

Each issue of UNIX*/WORLD brings you ·to-date, comprehensive information available on the most important compu .ry phenomenon since IBM introduced the PC! Whether you have a multiuser syste. or you're just thinking about purchasing one, you can't afford not to investigate UNIX*/WORLD.

EVERY ISSUE CONTAINS:

WIZARD'S GRABBAG—Tips & Techniques that ease the programmer's burden.
NEWS FROM AT&T/BERKELEY—News items from the two poles of UNIX*WORLD: Product releases, updates, major corporate news that affect our professional lives.
INSIDE EDGE—Trends, rumors, speculations from the corridors of Silicon Valley, Route 128, and the aisles of trade shows.
USER SPOTLIGHT—Every issue we focus on individuals or organizations who have contributed significantly to our community through knowledge, inspiration, or innovation.
PLUS. . . More of Bill Joy . . . Bill Tuthill's back with his own column . . . C vs Bourne Shell . . . Hardware price trends . . . UNIX*/WORLD on the top 50 vendors . . . new and expanded tutorials . . . and much, much more!

1 YEAR $36 NOW $18

☐ **YES!** Start my 1-year subscription (12 issues) today. Bill me later for only $18, that's 50% off the newsstand price.

BUY TWO YEARS, GET ONE FREE

☐ **YES!** Count me in on your limited offer to buy two years and get one free.

Start my three-year subscription (36 issues) right away and bill me for $36—the two-year price.

Canadian residents add $6. Other Foreign residents add $20 for International Postage. Both Canadian and Foreign orders must be payable in U.S. Funds on U.S. Banks only.

*UNIX™ is a trademark of AT&T Bell Labs. UNIX/WORLD is not affiliated with AT&T Bell Labs.

Name _____

Title/Company _____

Address _____

City _____ State ____ Zip _____

DE5AH8

UNIX/WORLD

UNIX SYST FOR SMAL

If your employees are shying away from your new Unix system-based computer, then training courses might be just the thing to break the ice.

BY LAWRENCE S. BREWSTER AND GRACE B. MULFORD

Perhaps you're planning to purchase a Unix system-based computer to automate a small company. Or maybe you've already won that battle—you've bought and installed your Unix system supermicro. The only problem is, it's sitting there all shiny and new, its little green and red lights blinking, but nobody's using it.

So maybe it's fair to say, instead, that you have lost the war because your employees are afraid to use your system.

In either case, you should seriously consider your needs for user training. The purpose of this article is to give you some general guidelines that should help you make a better informed decision on the right way for your particular company to get the training it needs.

If you have a small business and a system that supports fewer than 12 users, the system administrator probably will not require as thorough a knowledge of the Unix system as he or she would if you wished to use it for more complex applications.

A deeper knowledge of the Unix system is probably essential if, for example, you are planning to do extensive in-house application program development, if you expect to grow rapidly into the 20- to 50-user category, or if you expect to concentrate much of your multiuser applications on engineering and technical applications.

Small-business executives purchasing their first major computer might want to have a general understanding of the Unix system. Overview courses offer a nontechnical introduction to the system's features and capabilities, its characteristics, and its relevant applications.

More in-depth introductory

EM TRAINING
L BUSINESSES

courses cover fundamentals of the Unix system for users. An introductory class for users acquaints students with the basic operations of the Unix system and with the workings of computer commands and common shell commands.

Fundamental courses for more technical personnel—programmers and data-processing staff, for example—usually include a review of the Unix system, a text editor, commonly used shell commands, and directory hierarchies.

ADVANCED COURSES

Advanced Unix system classes are intended for system programmers, application programmers, and data-processing personnel who have taken fundamental Unix system courses. Topics covered in advanced classes include C programming, advanced C programming, internals of various versions, and shell programming. In the small-business environment, these courses are appropriate when the firm is large enough to have one full-time individual responsible for all operating system maintenance.

C language programming classes (for programmers and application staff) concentrate on a thorough treatment of the C programming language and introduce the student to Unix system interfaces. The course curriculum is often suitable for both Unix system and non-Unix system environments. If you plan to do significant programming, this course can be useful even before you have installed a Unix system.

Delivery media fall into three major categories: public seminars, on-site seminars, and electronic delivery methods.

Advanced C programming requires a deeper understanding of the Unix system, as well as familiarity with at least one other high-level language. Advanced C programming provides professional C language programmers with techniques for advanced coding, interfacing with the operating system, producing portable C code, and sophisticated compiling and loading options. These courses also offer a structure for C application design.

Shell command language can be taught on the user level as well as on the programmer level. Less experience is required of users, but programmers should have active knowledge of a Unix system text editor (to create and correct program text) and the ability to use files within the Unix system.

Classes in Unix System V internals explain the internal design and operation of the Unix system. Teaching this course requires that the trainee's organization have a Unix System V source license.

Other advanced Unix system classes cover specific screen editors, system tools, software devel-

opment, system administration, device drivers, document preparation, and interfaces to application programs with the Unix operating system.

The courses discussed above encompass basic topics. Individual professional training companies (PTCs) may group course progressions and objectives differently. Furthermore, AT&T offers classes that no other training firms do. According to our research, courses such as "Software Development Under the Unix Operating System" and "Internal Unix System Calls and Libraries Using C Programming" are specific to AT&T. PTCs may touch on these topics in their advanced courses, but AT&T devotes an entire course to each of these subjects.

THE BEST TRAINING FOR YOU

Training companies often concentrate on one or two delivery media, or ways in which they give their courses. Delivery media fall into three major categories: public seminars; on-site seminars; and electronic delivery methods, such as video-based instruction and interactive videodiscs. Some training vendors have a unique approach—the User Training Corp., for example, uses a microcomputer and audio tape system.

When you select the best delivery medium for your firm's needs, you should evaluate many key variables, such as the number of students, how often the course will be repeated, the importance of employee time away from the job, and budget. Figure 1 suggests guidelines for selecting a delivery medium.

TRAINING MEDIUM	NUMBER OF STUDENTS		
	1–4	5–29	30+
Live Seminars			
Public	X		
On-Site		X	X
Video-based Training		X	X
Interactive Videodisc			X

FIGURE 1: SOME GUIDELINES FOR SELECTING A USER-TRAINING DELIVERY METHOD.

Public seminars. The advantages of instructor-led classes include individual attention and opportunities to ask questions and clarify points. For firms with only a few students to train, the public seminar is the most cost effective.

There are, however, some disadvantages. Often students in such classes are at different levels of ability. This discrepancy creates problems both for the instructor, who is unsure of where to begin or how fast to go, and for those students who are either way behind or way ahead of the others.

Students are often asked to work in pairs when they receive hands-on training. Although training personnel point out that students benefit from working together, problems arise when students are incompatible because of their ability or pace of learning.

The key factor in determining the cost of training is the number of users versus the cost of the medium.

On-site seminars. Offered on the subscribing firm's property, on-site seminars have many advantages. Students learn on their own system as opposed to one furnished in a public seminar. The trainers are able to provide custom-designed training problems intended to address features and applications specific to the trainee's machine. Furthermore, students can work individually at their own terminals rather than working in pairs. As with any type of instructor-led course, students have the additional advantage of individualized attention.

A big plus for firms still evaluating equipment or that have yet to acquire a Unix-based system is that many training companies will bring equipment to their sites.

Higher cost is the most significant disadvantage of on-site seminars. Base your decision on the number of students to be trained. Trainers usually establish a fixed price for on-site seminars, as well as a class-size limit. However, the cost per student usually justifies the fixed fee for having the trainers come to your site.

Electronic delivery media. Video-based instruction involves purchasing a library of video cassettes accompanied by individual course materials. These videotapes become part of a company's library and thus can be used repeatedly.

A lower cost per user is a primary advantage of this medium. In addition, students can progress at their own pace. This method is ideal

SERIX

ANNOUNCING SYSTEM V UNIX™

...puts your IBM Series/1® ahead of the pack!

SERIX is the high performance CMI version of AT&T's UNIX™ System V operating system with Berkeley 4.1 enhancements ported to the IBM Series/1 minicomputer.

SERIX transforms your Series/1 into an even more powerful, flexible, and convenient processor for general data processing, office automation, communications, and process control. Its advantages are outstanding:

Reduced software costs

Long term growth path
- Software is highly portable
- Provides access to a large, growing software base

More power from the Series/1
- Optimizing C compiler uses native code features
- All code reentrant
- Dynamic memory allocation without fixed partitions

Increased programmer productivity
- Large set of utilities
- Hierarchical file structure
- Pipes, forks, semaphores, and shared data segments

Other CMI Series/1 software
- RM/COBOL™
- UNIFY™ database management system
- ViewComp™ spreadsheet
- vi visual editor
- EDX™- to -SERIX™ conversion kit

CMI Corporation is a Master Value-added Remarketer of IBM Series/1 equipment. Leasing and other financial arrangements are available.
Contact us for further information.

Photographer - Michael Zagaris • UNIX is a trademark of Bell Laboratories • SERIX is a trademark of CMI Corporation • SERIX was developed exclusively for CMI by COSI. • IBM, Series/1, and EDX are trademarks of International Business Machines Corporation • UNIFY is a trademark of North American Technology, Inc. • RM/COBOL is a trademark of Ryan-McFarland Corporation • ViewComp is a trademark of Unicorp Software, Inc.

CMI

A Torchmark Company

CMI Corporation
SERIX Marketing
2600 Telegraph
Bloomfield Hills, MI 48303-2026
(313) 456-0000

TWX: 810-232-1667
Telex: 499-4100 ANS: CMI CORP. BDHS

Member CDLA Member ASCD

Please circle Ad No. 83 on inquiry card.

for firms that cannot allow their employees to leave their jobs for extended periods. You can also choose to train people sporadically over a long period.

The down side of video-based instruction is that the initial purchase can involve a substantial investment. Be sure you intend to commit to the Unix system as an integral part of your business automation. A substantial number of employees should need the training in order to lower the cost per student.

Interactive videodiscs. The interactive videodisc (IVD) training medium is most appropriate for the small business that has a fast-growing employee base and that cannot afford to have many employees out for training simultaneously.

Videodiscs are, however, expensive. Given that this method of instruction is more expensive than video-based instruction, you should have a substantial number of users plus a good possibility for repeat uses of the IVD system.

TRAINING COST AND LENGTH

The key factor in determining the cost of training is the number of users versus the cost of the medium. Public seminars are a cost-effective alternative for the small firm that has few people to train. This is followed by on-site seminars—they may seem more expensive, but if you have enough students, the cost per student can actually be less.

AT&T licenses many of its courses. If you plan to train 30 or more people in one course, this is the most cost efficient. Not only does AT&T train one of your employees to be an instructor, but the firm also provides your company with direct consultation assistance with Unix system developers within AT&T.

Pricing strategies vary, depending on the delivery medium and the credibility of the PTC. The price per day can range from $200 to $300 per day. Note that the average price per day for an on-site seminar is $1400. Prices of an electronic delivery medium, such as videotapes and microcomputer/audio cassette packages, range from $2000 to $6000. Interactive videodiscs can cost as much as $20,000 or $40,000.

The training market for Unix systems has just begun to gain momentum, and locating the right training company for your needs can be difficult.

Course length varies according to the depth of the course, the delivery medium, and the level of the students. Public seminars are, on the average, no more than three days long; some may be as short as one day, while others may be as long as five. The length of the seminar gives a good indication of its depth and quality.

It should be obvious that overviews should be short, while courses for advanced end-users and programmers should be relatively long. Although some of the most advanced courses are two weeks long, our research suggests that students get the most out of an intensive course that is shortened to reduce their time away from the job. On-site seminars usually vary, depending on the clients' needs; they range in length from one day to a week.

Electronic delivery media are most often presented in module format. Each module covers a lesson and can last from half an hour to an hour. Depending on the depth of the unit, a total course can require from 30 minutes to 20 hours.

FINDING THE RIGHT PROGRAM

The training market for Unix systems has just begun to gain momentum, and locating the right training company for your needs can be difficult. As a result, there are several recommended sources to locate training. One of the most logical places is AT&T Technologies—the originators of the Unix system. AT&T offers public seminars at its various training facilities throughout the country, in addition to licensing its courses to both user companies and professional training firms.

There are, however, qualified alternatives to AT&T. We have located at least 30 major PTCs in the U.S. and one in Canada that focus on Unix system training. One of the best places for a recommendation is a Unix system value-added dealer (VAD), value-added reseller (VAR), or OEM (original equipment manufacturer). They often offer training as part of a hardware/software package and might address certain specific applications, as well as make themselves available for future support.

Unix system hardware vendors can also act as a referral source if they do not provide Unix system training. Searching Unix system periodicals for advertisements and articles discussing PTCs could also prove fruitful. Attending Unix sys-

Continued on page 63

First we made the promise . . .

Then we delivered!

UNIX EXPO

There aren't that many trade shows specifically dealing with UNIX, just enough to be confusing. UNIX EXPO . . . the New York Show . . . is the one that **delivered**. Delivered the right audience in the right numbers. Numbers so right that many of last year's exhibitors have already **doubled** the size of their exhibit space for the upcoming show.

UNIX EXPO will continue to deliver. We have a new location in the heart of New York City's business district, an expanded conference program, a proven track record, more exhibit space . . . and the trade show savvy to produce an audience that comes to talk business.

If you sell to any segment of the UNIX environment, make certain that you exhibit in the show with the right stuff.

Call Bob Birkfeld or Don Berey today for the complete story. You won't be disappointed. That's a promise!

September 18, 19, 20, 1985 • New York, N.Y.

T.M.

UNIX EXPO

THE UNIX OPERATING SYSTEM EXPOSITION & CONFERENCE

Produced & Managed by National Expositions Co., Inc.,
14 West 40th Street • New York, NY 10018 • Telephone: 212/391-9111 • Telex: 135401 DIMCOMM

UNIX™ is a registered trademark of Bell Labs UNIX EXPO is not affiliated with Bell Labs

Please circle Ad No. 78 on inquiry card.

Now for $799, a VT-220 with perfect function,

3040 N. First Street, San Jose, CA 95134 (408) 946-3075

compatible terminal fit and form.

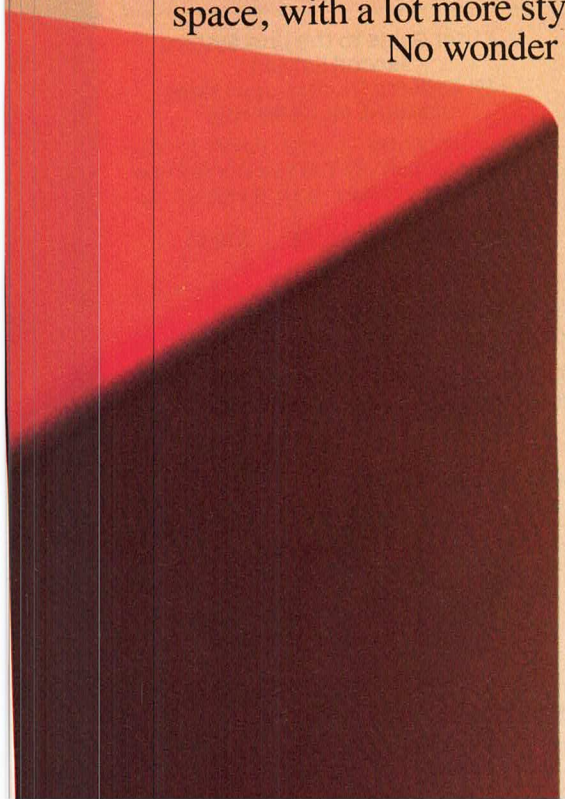
Our WY-85 gives you complete DEC VT-220™ software compatibility in function and fit, plus a form that beats all, for a price that beats all.

Like all our terminals, the WY-85 packs all the ergonomic features you want into an unusually small footprint. The generous 14″ non-glare screen tilts, swivels, and handles a full 132-column format. Even the sculpted, low-profile keyboard adjusts for perfect fit and easier function.

So you get not just emulation, but the full operational compatibility you're looking for, including keyboard layout. For a lot less money, in a lot less space, with a lot more style.

No wonder Wyse terminals are now on more than 150,000 desktops worldwide.

For more information about our full line of computer display products, call the regional office nearest you.



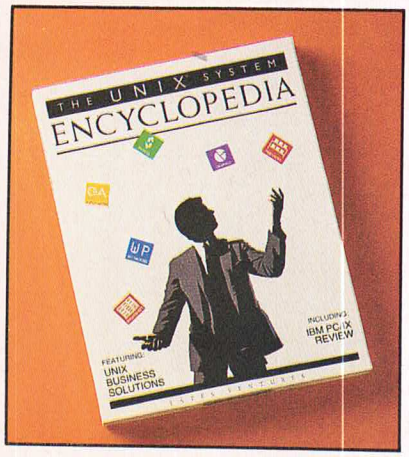
WYSE

Regional Offices:
Northeast (201) 725-5054
Southeast (305) 862-2221
Midwest (313) 471-1565
Southwest (818) 340-2013
Northwest (408) 559-5911

Please circle Ad No. 30 on inquiry card.

THE UNIX/WORLD COLLECTION

COMPLETE YOUR PROFESSIONAL LIBRARY ON THE UNIX SYSTEM!

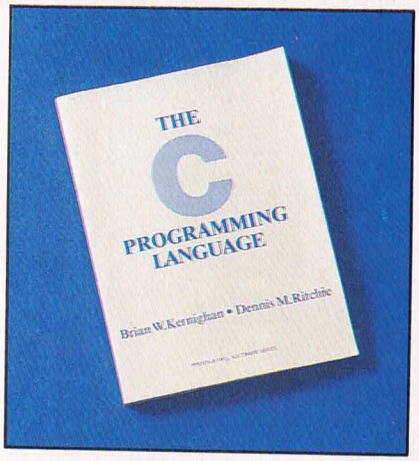


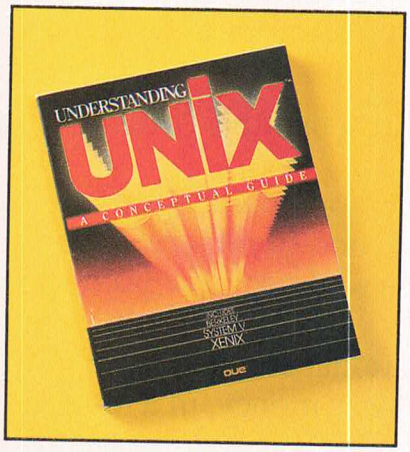
The Unix System Encyclopedia
YATES VENTURES

The UNIX System Encyclopedia is the largest available collection of information about the Unix marketplace. Over 400 pages, with a comprehensive Unix directory. Includes 100 pages of manufacturer specifications, photographs, software reviews, and articles on the Unix system.
PUBLISHER'S LIST PRICE: $34.95

The C Programming Language
Kernighan/Ritchie
PRENTICE-HALL

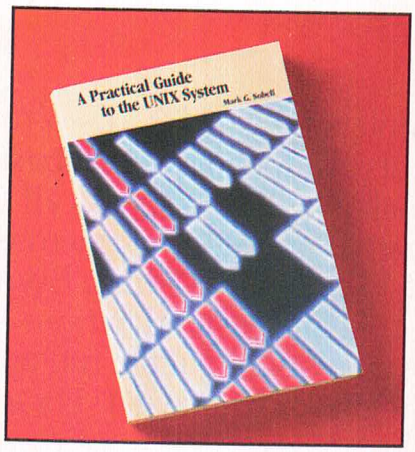
The C Programming Language is an in-depth guide to programming in C. A true classic, it appeals to all levels of expertise.
PUBLISHER'S LIST PRICE: $22.50





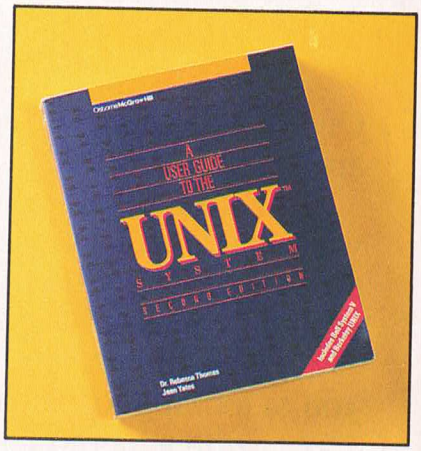
Understanding Unix—
A Conceptual Guide
Groff/Weinberg
QUE PUBLICATIONS

An excellent introduction for managers new to the Unix system, the *Conceptual Guide* introduces you to the reasons for implementing the Unix system.
PUBLISHER'S LIST PRICE: $17.50



A Practical Guide to the Unix System
Mark Sobell
BENJAMIN CUMMINGS GROUP

A Practical Guide to the Unix System is an excellent reference as well as a detailed learning tool. As an added bonus, the author had included a comprehensive overview of Xenix.
PUBLISHER'S LIST PRICE: $21.95



A User Guide to the Unix System
Second Edition
Dr. Rebecca Thomas/Jean Yates
OSBORNE MCGRAW-HILL

Takes up where *A User Guide to the Unix System* left off. The basics of the Unix System for novices, with hands-on tutorials.
PUBLISHER'S LIST PRICE: $18.95

ITEM		QTY	PRICE	TOTAL
User Guide to the Unix Syst. Second. Ed. **Thomas/Yates**	U200			
Practical Guide to the Unix System **Sobell**	U201			
The C Programming Language **Kernighan/Ritchie**	U202			
Understanding Unix, Conceptual Guide **Groff/Weinberg**	U203			
The Unix System Encyclopedia, **Yates Ventures**	U204			
SUBTOTAL				
SAVE! IF YOUR ORDER IS FOR MORE THAN ONE BOOK, **SUBTRACT 20% HERE**				
CA RESIDENTS ADD 6.5% SALES TAX				
SHIPPING AND HANDLING				$2.50
GRAND TOTAL				

SAVE ON ORDERS OF TWO BOOKS OR MORE, TAKE 20 % OFF THE SUGGESTED LIST PRICE!

CREDIT CARD HOLDERS SAVE TIME & MONEY BY CALLING TOLL FREE:
(800) 632-7979 (CA Residents only) or **(800) 227-6703**

Ship Order to:
Name: _____
Company _____
Add. _____
City _____ State _____ Zip _____
Signature _____

☐ Check Enclosed
☐ Please Charge to My:
☐ Mastercard _____
☐ Visa Card _____
Expires _____

ADDRESS ORDERS TO:

UNIX/WORLD Magazine
125 Main Street
Half Moon Bay, CA 94019

One billion byte supermicro. Just think of the possibilities.

Introducing the new, expandable Dual 83/500. A UNIX*-based, 68000-driven supermicro so capable, you'd swear it was a mainframe.

The system already comes with 500 megabytes of Winchester storage. And our patent-pending high-speed SMD disk controller for fast access to data.

But you can increase memory to a massive one billion bytes just by adding a twin drive.

Or take an already sizeable two megabytes of RAM and expand it to six.

Or even double user capacity from 8 to 16. The hardware is already in place.

When it comes to value, no supermicro system offers you more than the 83/500. Because along with the computer, you get a 9-track, 1600 BPI phase-encoded tape drive for reliable disk backup and quick file transfers to other systems.

There's a convenient one megabyte double-sided/double-density floppy disk drive that protects individual files.

And the industry standard UniPlus+™ implementation of AT&T's UNIX System V with Berkeley enhancements. Plus a multi-user license.

All at no extra cost.

And while you're speedily going about processing your data, we're protecting your investment. For free. With a comprehensive 12-month warranty. And a nationwide service network that protects your system whether it's in or out of warranty.

See the system that's redefining the supermicro. The value-packed Dual 83/500. Call or write Dual Systems Corporation, 2530 San Pablo Avenue, Berkeley, CA 94702, (415) 549-3854

At just $65,940 base price, its possibilities are endless.

*UNIX is a trademark of AT&T Bell Labs.
™UniPlus+ is a trademark of UniSoft Corp.

There's only one DUAL

Please circle Ad No. 96 on inquiry card.

COMPANY/LOCATION/PHONE	COURSE TOPICS
Ian D. Allen/ Unit 10, 326 Erb St. West, Waterloo, Ont., Canada N2L 1W3; 519/886-8622	Various levels Unix prog. & software training
AT&T Corporate Education Center/ P.O. Box 1000, Hopewell, NJ 08525; 609/639-4133	Unix System V training
Auxton Computer Enterprises, Inc./ 2 Kilmer Rd., Edison, NJ 08817; 201/572-5075	Complete Unix and C curriculum
BASIS/ 1700 Shattuck Ave., Suite 1, Berkeley, CA 94709; 415/841-1800	Unix training/C programming
Bunker Ramo Training Serv. Group/ 35 Nutmeg Dr., Trumbull, CT 06609; 203/386-2000	Intro. & advanced Unix/intro & adv. C. prog.
Computer Technology Group/ 310 S. Michigan Ave., Chicago, IL 60604; 312/987-4000	C prog./shell prog./Unix fundamentals, internals, etc.
Concentric Associates, Inc./ 22 Netherwood Ave., Plainfield, NJ 07062; 201/866-2880	Complete range of Unix & C courses—custom tailored
Control Data Corp./ 6003 Executive Blvd., Rockville, MD 20852; 301/468-8576	Unix prog./shell prog./other Unix topics/C prog.-features
Cutler-Childs & Associates/ 17101 Edwards Ave., Southfield, MI 48076; 313/569-6161	Unix application training
Daemon Associates/ 1760 Sunset Blvd., Boulder, CO 80302; 303/442-1731	Unix intro./prog. in C/shell programming/screen editor
Delft Consulting Corp./ 432 Park Ave. S., New York, NY 10016; 212/243-8700	Unix & C training/shell prog. higher-level specialists
George Mason University/ School of Bus. Admin. Fairfax, VA 22030; 703/323-2758	Intro. & advanced Unix & C programming, various levels
Human Computing Resources/ 10 St. Mary Street, Toronto, Ont., Canada M4Y 1T9	Unix tech. & mgmt. overviews/C programming
Inst. for Advanced Professional Studies/ 55 Wheeler St., Cambridge, MA 02138; 617/497-2075	C & shell prog./Unix on-line tutorials, inter. & software
International Technical Seminars/ 520 Waller St., San Francisco, CA 94117; 415/621-6415	Complete Unix & C curriculum—advanced Unix courses offered
Lachman Associates/ 645 Blackhawk Dr., Westmont, IL 60559; 312/986-8840	Full curriculum of Unix & C courses
Plum Hall/ One Spruce Ave., Cardiff, NJ 08232; 609/927-3770	Intro.-advanced C prog.; intro. Unix workshop
Productivity Products, Inc./ 37 High Rock Rd., Sandy Hook, NY 06482; 203/426-1875	Object-oriented prog.; objective C prog./Smalltalk
Santa Cruz Operations/ P.O. Box 1900, Santa Cruz, CA 95061; 408/425-7222	Tutorials—Unix, C prog.; dealer training; sys. admin.
Software Kinetics Ltd./ 3 Amberwood Crescent, Nepean, Ont., Canada K2E 7L1; 613/226-6792	Unix & C training & prog./system manager course
Specialized Sys. Consult./ Box 7, Northgate Station, Seattle, WA 98125; 206/367-UNIX	C & Unix reference cards & materials/intro. Unix & C
Structured Methods/ 7 West 18th St., New York, NY 10011; 212/741-7720	Unix workshop & internals/C & shell programming
Telemedia/ 310 S. Michigan Ave., Chicago, IL 60604; 312/987-4000	Unix courses/C & shell prog. shell as command language
Uniq Computer Corp./ 28 S. Water St., Batavia, IL 60510; 312/879-1566	Unix intro./shell prog./sys. administration/C prog.
User Training Corp./ 591 W. Hamilton Ave., Suite 102, Campbell, CA 95008; 408/370-9710	Tutorials—Unix, C prog.; novice and expert levels

DELIVERY MEDIUM(S)	COMMENTS	COST
On-site seminars	Designed for each client; various levels of expertise required.	$500/day median cost
On-site seminars, or at AT&T centers	5 levels: mgr./super.; user; system admin.; applic. devel.; & system programmer.	$200–$1300 per course
On-site & public seminars	Not hardware specific; costs based on instructor days—not number of trainees.	Not given
At BASIS office only	Novice through advanced; uses User Training Corp. equipment, which may be rented.	$154–$200 per course
At training centers or on-site training	Prerequisites vary with course material.	$100 per day per student
Public seminars, Interactive Video	Prerequisites vary; all courses provide hands-on exercises.	$225–$11,000 per course
Custom-designed course manuals	Can license courseware to companies; did all training for AT&T Information Services.	$1250/day consulting
Public & on-site seminars; video cassettes	Not hardware specific; courses given all around country; books also available.	$245–$1095 per course
Unix turnkey system training for client	For Onyx, Altos, IBM hardware; courses tailored to client needs.	$40–$80 per hour
Customized on-site courses	Not hardware specific; courses customized for customers; 20 students per course max.	$750–$1250 per day
On-site courses, may be customized	Not hardware specific or Unix version specific; cost is per class, not per student.	$1500–$2500 per class
Classroom work at univ. & on-site seminars	Contact Dr. Fife; prerequisites vary; two 20-week courses; not hardware specific.	Negotiable; below norm.
In-house, public, and on-site seminars	System V specialists; not hardware specific; customized on-site courses.	$450–$900 per course
Tutorials, public, & on-site seminars	Not hardware specific, but specific machines can be addressed; mgmt. brief; custom train.	$70–$300 per day
On-site & public seminars	Not hardware specific; on-site courses tailored to individual customer.	Not given
On-site seminars, some in-house	Not hardware specific; courses tailored to individual customer's requirements.	$500–$1500 per day
On-site & public seminars	Not hardware specific; computer shipped to sites—hands-on training; prerequisites vary.	$1000–$9000 per course
On-site & public seminars	Requires 16-bit or larger hardware; prerequisites vary depending on course.	$200–$10,000 per course
In-house and on-site seminars	Not hardware specific; User Training Corp. tutorials; custom-designed on-site courses.	$205/month (tutorial)
Customized on-site training	Training is for software customers; courses taught by consulting & product support.	$10,000 for 10 people
Public & on-site seminars (hands on)	Customized on-site courses; on-site costs vary; 15 students max. for on-site course.	$150/day to $100/hr.
On-site & some public seminars	Not hardware specific; some data process. knowledge required; six years Unix training exper.	>$200–$9000 per course
Courses, video-based & interactive video	Not hardware specific; prerequisites vary by course; public & on-site courses also.	$490–$1375 per course
In-house courses; some on-site	Not hardware specific; system manager course discusses DEC; five years teaching experience	$950 per course
Audiodigital course: player + cassette + book	Any hardware w/RS-232 communications; like looking over experienced instructor's shoulder.	$680–$1680 per month

The chart was compiled by David Keith, who has been working as a freelance computer journalist and market researcher for the past two years, and prior to that as editor of a radio science and health newsletter.

tem trade shows can not only provide you with the education you desire but also offer direct contact with Unix system vendors.

The Unix operating system is emerging as a major competitor in the computer operating system market, with the worldwide number of installations totaling nearly 200,000

Advanced C programming requires a deeper understanding of the Unix system and familiarity with at least one other high-level language.

at the end of 1984, according to figures from Yates Ventures.

This growth has created a substantial need for Unix system training. The number of PTCs is growing, and hardware vendors, VARs, and OEMs should address this growing need. Options include offering training themselves, establishing a training program via their dealers, or turning to a third party to advise their customers on where to seek Unix system training best suited for them. □

Larry Brewster is president of L.S. Brewster and Co., an industrial marketing strategy consulting firm located in Boston. Mr. Brewster received his M.B.A. from Harvard Business School. Grace Mulford, an information specialist with the same firm, has co-authored reports on Unix system training and personal computer marketing. She received her B.S. from Boston University.

MICROVAX I LETS YOU START SMALL IN A VERY BIG WAY.

All supermicros let you start out small. Most run into problems, however, when your applications start getting big.

That's why Digital's MicroVAX I™ supermicro is such an intelligent solution. It limits your risk – not your capabilities.

For starters, MicroVAX I is a member of the VAX™ systems family; it provides a full set of VAX resources. That's saying a lot, because VAX systems are the industry standard for 32-bit computing.

MicroVMS,™ our VMS™ operating system for MicroVAX, lets you develop applications for any VAX using the comprehensive set of proven development tools provided by the VAX computing environment.

You get all the VMS editors, debuggers, database managers, screen managers, utilities and communications options. You can use advanced VMS languages – including Pascal, FORTRAN, C, Bliss, COBOL and Ada.® And through the VMS common calling standard, you can combine routines written in different languages within the same program.

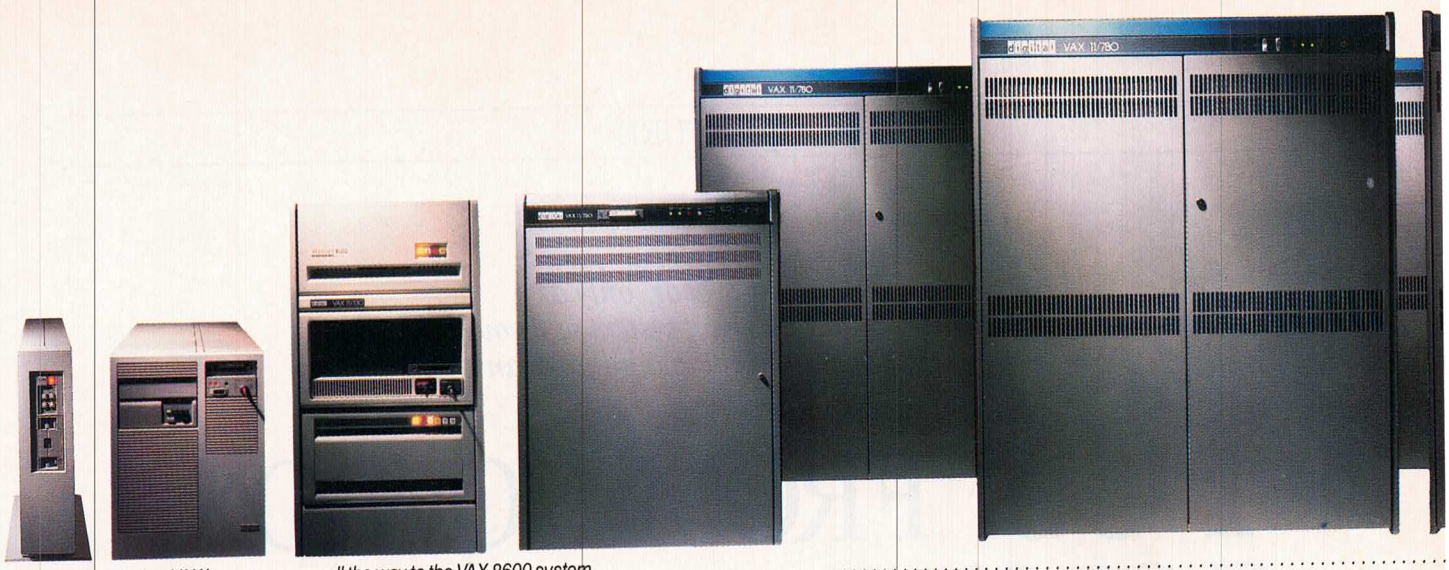
This software sophistication lets you cost-effectively dedicate the MicroVAX I system to highly complex tasks. In areas like process control, signal and image processing, CAD/CAM, mathematical simulation, seismic analysis and automatic test. There's no end to the uses for MicroVAX.

THE BIGGEST GROWTH PATH IN THE INDUSTRY STARTS HERE.

The sky's the limit for applications you develop on MicroVAX. They can run on every VAX in the family – including our top-of-the-line VAX 8600 and VAXcluster™ systems. Through upward expansion and shared resource networking, the computing power at your disposal is virtually inexhaustible. You can go from a supermicro to a mainframe-level system without changing your software. And that's an advantage no other supermicro supplier can match.

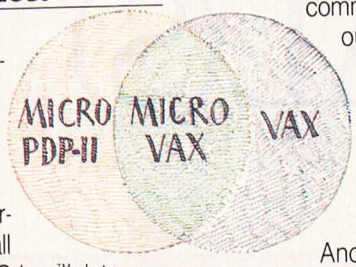
Conversely, you can download VAX applications to MicroVAX I systems. All non-privileged commands and the full VAX instruction set are supported, so your applications will run with little or no modification.

There's even VAXELN™ software, a complete Pascal-based development toolkit, that runs on VMS and MicroVMS. With VAXELN, it's easy to create dedicated realtime applications for MicroVAX and VAX systems. By leaving out the overhead, VAXELN gives you a highly efficient runtime environment.



Starting with MicroVAX I, you can grow all the way to the VAX 8600 system

PART OF A GROWING FAMILY OF SUPERMICROS.



Our 32-bit MicroVAX I systems have a lot in common with our 16-bit MicroPDP-11™ supermicros. They all use the same Q-bus™ data transfer architecture. So they all support the same peripherals.

That's a substantial benefit, because Q-bus is an industry-standard bus, with many options available from Digital and independent vendors. Hundreds of peripheral products are available to make your MicroVAX configurations complete.

Commonality between MicroVAX I and MicroPDP-11 systems is also maintained in file structure, communications options, languages and even many applications. All of which expands your options and provides unlimited opportunities for growth.

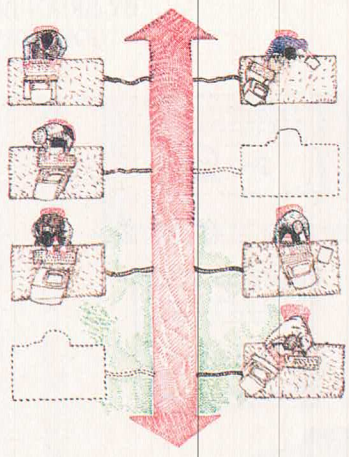
DECNET, YOUR UNIVERSAL NETWORKING STRATEGY.

When you're ready to distribute your applications via MicroVAX I systems, DECnet™ software makes it easy. MicroVAX supports DECnet to give you a complete set of local and wide area networking facilities.

Your options include local communications via our implementation of the most advanced level of the Ethernet local area network available today. And wide area communications via dial up or leased lines, or public packet-switched X.25 networks. Further, you can establish these local and remote systems links transparently to the user and the program.

Digital's networking facilities also let you take advantage of the computing resources of non-Digital systems. The extensive selection of standard protocols available to you with the MicroVAX I system includes 2780/3780, 3271, X.25 and SNA. All these facilities are fully implemented not just at the hardware level, but through software as well.

So if you want to start small,



start big with the MicroVAX I system. You'll get the 4-gigabyte address range, the powerful instruction sets, the incomparable development tools and vast software selection that has made our VAX systems the most popular 32-bit computers in the world. And you'll get it all with the economy and convenience of a supermicro package.

ULTRIX, YOUR TRANSPORTABLE OPERATING SYSTEM.

If you need UNIX® software capabilities, our MicroVAX I system can readily provide them. It supports ULTRIX-32m™ software, our enhanced, fully supported UNIX software for MicroVAX.

ULTRIX-32m includes all Berkeley 4.2 features, in addition to Digital's enhancements that simplify error reporting and recovery.

ULTRIX-32m applications can run throughout the VAX family. Our PDP-11™ and MicroPDP-11 systems families also support a version of ULTRIX software. And that lets you extend a single application across a vast range of systems. You can go from low-cost MicroPDP-11/23 and MicroVAX I systems all the way to our high performance VAX 8600 systems with one, multi-purpose solution. Think how that will extend your reach in your market.

BEST ENGINEERED MEANS ENGINEERED TO A PLAN.

Our MicroVAX I supermicro, like all Digital hardware and software products, is engineered to conform to an overall computing strategy. This means our systems are engineered to work together easily and expand economically. Only Digital provides you with a single, integrated computing strategy, from chips to 32-bit VAX systems and direct from desktop to data center.

If you believe that your company could benefit from our very growth-oriented MicroVAX I system, contact your nearest Digital sales office or a Digital Authorized Industrial Distributor. Or call: 1-800-DIGITAL, ext. 230.

THE BEST ENGINEERED COMPUTERS IN THE WORLD.

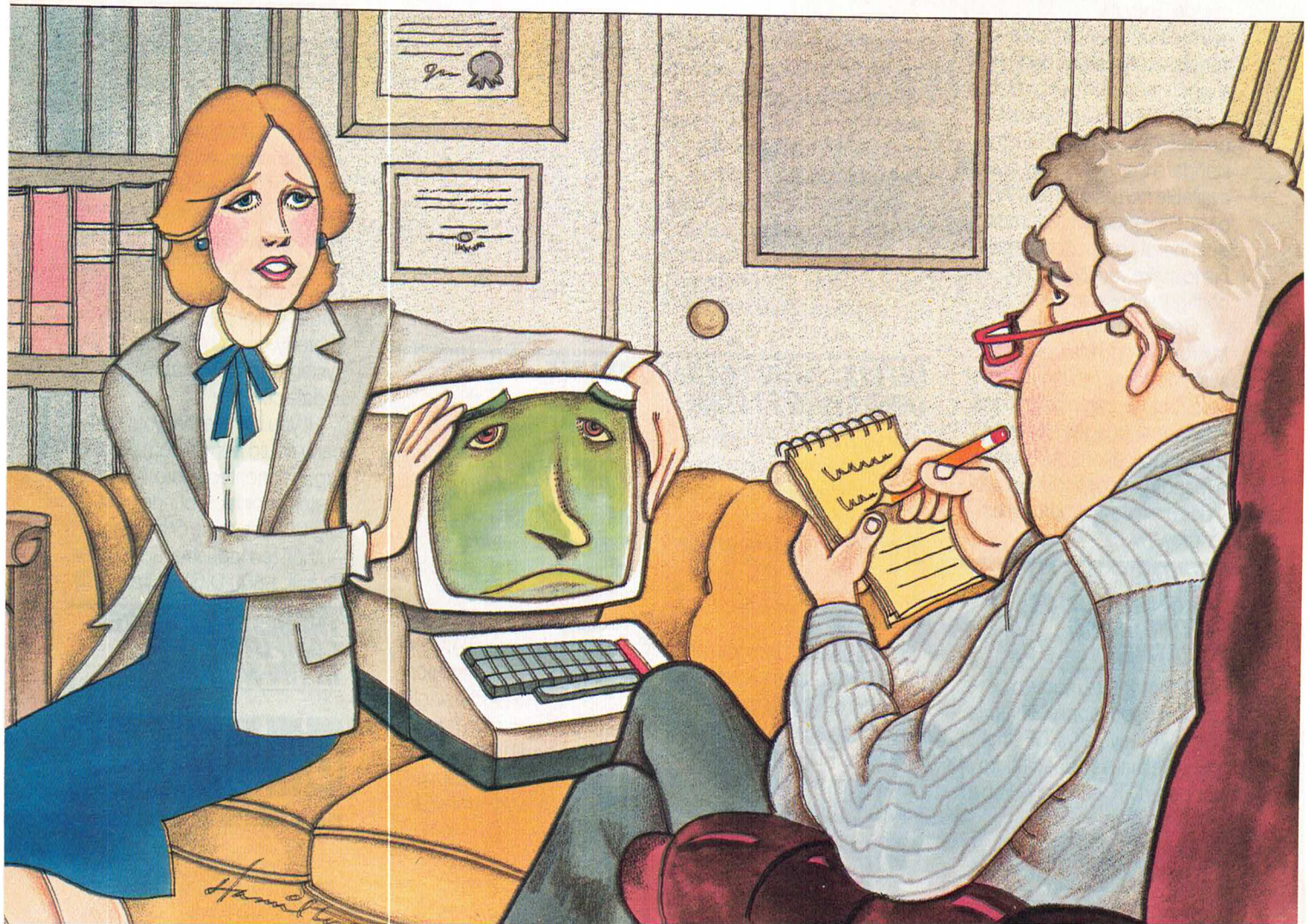
digital™

Please circle Ad No. 34 on inquiry card.

Frankly, my dear, only as a terminal to a Unix system will your PC ever amount to a hill of beans.

PCs: FROM NOW ON CONSIDER THEM TERMINAL(S)

BY BRIAN BOYLE AND
JOHN DUNHAM



Being a Computer Analyst ain't easy. Especially like last Thursday. This blonde brings in one of those cute little PCs. Says it's "just a little slow for its age."

"My Carrot II plays games and prints beautifully—just like the older computers, but . . ." she trailed off, unable to face the ugly truth.

"Can't walk and chew gum at the same time, eh?" says I, in the tactful bedside manner I'd picked up from system programmers trying to relate to end-users. "We call it Single-Tasking Syndrome (STS). It's often associated with a form of mental deficiency known as Underbyte. Usually the victim has good reflexes and a nice smile but an IQ of 80 to 88, roughly an IDIOT/BARELY MORON. And it's congenital," I concluded. Just the facts. "Any questions?"

She shrank into herself but went on bravely. I must be losing my touch, I thought.

"I know he'll never go to the UNIXversity, but perhaps some time-management courses in Concurrent DOS would open a few doors. Maybe he could get a job though the Old PC Network I've heard so much about. . ." her eyes pleaded.

"THAT'S not a question," I snapped, finally infusing her with the precise blend of awe and loathing that makes this job so satisfying. "As for multiple PC networks, if you think he's dangerous on his own, you really think 10 cretins banging away can produce the Theory of Relativity? Smarts ain't additive, lady. Unless you take my advice, your little Carrot will be a vegetable forever!"

"You mean there's hope, Doctor?" she brightened. "Do you mean that my PC's condition isn't really terminal?"

"Precisely the opposite, madam. Only as a terminal to a Unix system will your PC ever amount to a hill of beans."

In reality, PC Disease is not only hereditary but highly contagious—it can be transmitted over a local-area network. Worst of all, it's not a natural condition but the result of a deliberate marketing strategy by IBM. Since August of 1981 the PC has been used as a commercial Market Defoliant (let's call it Agent Blue); it promotes the explosive growth of low-level computer power at the expense of new supermicros that might mature to compete with the established first- and second-generation market of mainframes and minis that IBM has all but cornered.

How dare those quiche-eating PC Eunuchs demean our Real-Unix as being slow, visually unappealing, and downright unsanitary.

The minicomputer wildfire of 1965-1975 took IBM essentially by surprise. (Who could have guessed that anyone playing business with a full DEC would take those little toy computers seriously?) Worse yet, when you're as big as Big Blue, you've got to watch that you don't become your own largest competitor. To avoid cannibalizing its mainframe market, IBM introduced some of the most underpowered iron ever offered; the Systems/III, /32, and /34 are based on a proprietary EBCDIC internal 8-bit processor that just beats a Timex/Sinclair on a good day.

Five years ago, when Apple Corp. fell from the Tree of VisiCalc into the Garden, IBM set out, as usual, in Pursuit of Excellence—wielding an ax at the root of the emerging PC market. The IBM Way—since long before Peters and Waterman ever voiced it—has always been simple: "Stay close to the customer; in fact, sit on him."

In choosing a PC to avoid competition with established minicomputer products, IBM had plenty of choices from its own shelf. It was, at the time, the world's second largest producer of semiconductors, all for consumption by its own product lines. Given that tiny Zilog was able to whip up its own proprietary chip on a few million dollars (and a few million IQ points), IBM would have had no trouble in turning out its own PC chip in short order. But that would have just given the little guys a target to beat on performance—not a tough job because IBM was still forced to protect the soft underbelly of its business minicomputer line.

A quick digression to Business 101 tells us this: RETURN ON INVESTMENT = (PROFIT MARGIN % × MARKET SHARE % × MARKET SIZE $) / INVESTMENT $. As efficient and as pricey as corporate IBM is, it's really tough to attain over 25 percent profit margin before someone—customer or competitors—begins to take note. As unshackled as IBM seems today, it's dangerous to get over 75 percent market share before someone—usually the Justice Department—begins to take notes.

Only the *market size* leaves growing room for a profit worthy of IBM's investment—after all, market and share are limited to a paltry 100 percent. But that means that all the players—software vendors, value-added resellers (VARs), consultants, and so forth—must go after One Big Market. That takes a standard everyone can agree on—a standard merchant chip, a standard commercially available operating system, standard off-the-shelf software, and standard value-added distribution channels.

IBM was already using the Intel 8086 and the fairly sophisticated

UCSD p-System on its DisplayWriter office product, but that's dangerously close to the wimpy power of the 50,000 wimpy System/34s on month-to-month lease. All it would take would be some wise guys like California Software (Anaheim, Calif.) to come up with an RPG/34 emulator, and quicker than you could say "three-month amortization," IBM would have a lot full of trade-ins on some Baby Blues.

If you're looking to set a standard for an entire shortsighted industry to conform (down) to, you couldn't get much lower than MS-DOS 1.0 on an 8088. Even Intel never expected the 4.77 MHz chip with the reduced 8-bit data path and the shrunken 4-byte prefetch buffer to find application except in 8-bit retrofits. And MS-DOS was little more than a quick "productization" of Seattle Software's 16-bit upgrade of CP/M-80. Over time, various features have been tagged on to make it more Unix system-like, so that MS-DOS 3.0 is essentially a camel—a horse designed by committee. In the understated manner of a test pilot about to eject from his overtaxed bird, MS-DOS "has catastrophically exceeded its design limitations."

PC DISEASE

So why is this PC disease so contagious? Why did writing for PC-DOS (let's call it like it is) consume half the software talent of the country just trying to get out that next hit single in time for the Floppy Grammies? How dare those quiche-eating PC-Eunuchs demean our Real-Unix as being slow, visually unappealing, and downright unsanitary?

Basically, they have a point. Much as the authors of this article like Horizon Wordprocessing on our Onyx running Interactive's IS/1, this text was keyed on a Wang OIS shared-logic word-processing system. Our Wang workstation has a 5-MHz Z80 pushing the bits around the screen faster than the eye can see, and there's another Z80 back at the central processing unit (read "file server"), connected by a coaxial cable that could pass for an Ethernet until you stick your nose into the transmission and packetizing protocols.

Let's look at the downsides of this prototypical local-area network. For all intents and purposes, the several-thousand-dollar Wang workstation is a very expensive diskless PC, hooked into an even more expensive star-topology network running proprietary applications software on top of a proprietary operating system.

An example that illustrates this dangerous situation follows: Nearly

a year ago this expensive software stopped accurately logging the date/time of last document update—the things you see in the `ls -l` that comes free with every Unix system. After many pained calls, the Wang field service folks agreed that we had a problem that made it impossible to know when to archive files. They added that we probably should have thought of that when we bought Wang in the first place and locked ourselves into a single vendor whose developers might find maintenance tasks (that is, debugging the side effects of their more creative efforts) too dull to deal with.

For all the complaints about buggy Unix system software, Horizon, Quadratron, CrystalWriter, FinalWord, XED, For:Word—Hell, even Wordstar—are far better written and maintained than that from the Number One office automation company.

Given all these demonic problems, why are we willing to put up with it? Because the ridiculously slow I/O speed of that little pair of wires between the Unix processor and the Unix terminal—even at 19,200 baud—makes us feel like we're being put on hold most of the time. The real magic of the *personal computer* is not that it's much of a computer but that it's personal: The processor is right there behind the screen, waiting to serve only its one true master. For a variety of reasons, it's nice to have your own personal toothbrush rather than time-sharing, if at all economically feasible. But we don't necessarily expect to have our own private bathroom—we accept the time-slicing algorithms normally imposed by multiuser living quarters.

BRAINS AND BRAWN

So how do we get the best of both worlds? What degree of integration is required to have brains and brawn? Several recent Unix system offerings perform the PC-Unix system marriage ceremony with dignity and panache: PC-Works, from TouchStone Software (Seal Beach, Calif.), was one of the first to use an IBM PC as a smart terminal logged on to a Unix system. One amusing solution to the great "Standard Media" controversy in the Unix system environment is TouchStone's method of Unix system software distribution—on a ubiquitous IBM 5¼-inch floppy that is uploaded from your PC's drive to your Unix system hard disk, avoiding the morass of nonstandard media formats our Unix systems are stuck in.

Even on a blindingly fast supermicro such as the Zilog, which could certainly keep up with power typists

on input, there is some slowdown when thousands of characters per second are flashing down the asynchronous lines to several screens at once. (The Unix system has a nice, general I/O procedure for interrupt handling; we pay for that generality on every interrupt.)

PC-Interface, from Locus Computing Corp. (Santa Monica, Calif.)

When you're as big as Big Blue, you've got to watch that you don't become your own largest competitor.

has a recent offering linking AT&T 3Bs to AT&T, IBM or compatible PCs over a high-speed network line—in a star topology. With only 17K of code on the PC side and with about twice as much on the Unix system file-server side, this very smooth implementation can flip back and forth between remote log-in on the Unix system or locally on MS-DOS, therein switching between local files and remote files on the Unix server—faster than the PC/XT's hard disk and much faster than the floppies. The trick is blocked I/O to the PC "terminal" over the net—hundreds of characters at the cost of one interrupt.

Communiqué, from C.O.S.I. (Ann Arbor, Mich.), admits to being a simple but powerful program in the classic Unix system tradition. A collection of (user-modifiable) shell scripts and some ingenious work (with mirrors) not only lets PCs connect to Unix system hosts such as C.O.S.I.'s SERIX on the IBM Series/1, but it lets one PC ship a job to any idle PC on the star network that isn't running a job for its own user.

On a proprietary note, the WorkNet from Altos links multiple Unix system processors together in a distributed file-system configuration. Running under Xenix, it leads us in the direction of PC-to-Unix link in which the PC is also running the Unix system.

The ultimate PC/Unix system integration is, of course, outside this picture. But the PC/IX/AT plans at Interactive and its cousin Uniform Software Systems involve the Connector, a PC-DOS emulator running under the Unix system—on Intel family products only, of course.

OTHER OFFERINGS

Closer to the matter at hand is "Sprouts," also from an Interactive offshoot—so to speak—Slater Towar Inc. (10,000 feet up in Estes Park, Colo.). Sprouts looked good using the little-known graphics capability of the IBM PCjr to generate medium-resolution typographic-quality characters on a truly "distributed" editor—much of the graphic smarts on the Unix system host and the direct bit-pushing on the little PC. Principal author Ned Irons is the same wizard who designed the flashy Ten-Plus interface for Interactive on its PC/IX system.

The mention of fancy graphics directs one's attention toward a newer kind of PC from a company that didn't worry about stealing markets from mainframe and minicomputer giants. The Apple Macintosh, which has a price that qualifies it as a PC (something its forerunners—the Lisa and the Xerox Star—lacked), sets yet another generation in standards for user expectations. Used as a terminal with the Unix system, the Mac or the new AT&T 7300 (code-named "Safari4" at manufacturer Convergent Technologies) could provide at a mere introductory price some of the power of the Office Publishing System-2000 from Interleaf (Cambridge, Mass.) running on the Workstation from Sun Microsystems (Mountain View, Calif.).

As a sideline, Fusion, from Network Research (Santa Monica, Calif.), and Unetix, a networked Unix system look-alike from LanTech (Austin, Texas), offer fairly advanced products that fall outside this specific topic. But watch for their impact on the PC-Unix system connection, particularly on the IBM PC/AT.

As long as we're looking into the future of the PC-Unix-Unix interconnection, our mutual favorite is again from Locus Computing. As with the Unix system itself, which incubated for years in Bells Labs and then in academia before hatching into commercial daylight, the LOCUS (Network) Operating System rattled around in the labs of UCLA professor Jerry Popeck for some time before it saw the light of day in early 1983. It truly offers a foretaste of things to come.

LOCUS is reminiscent of IBM's System R* (or System D) distributed relational database projects, which IBM will probably drop on an unsuspecting world under a

Just think of the Unix system connection as a sort of mobile quality-of-life support system.

TopView-derived interface environment. (And if IBM's recently issued software patents hold up in court, we will just have to sit and watch from the sidelines.)

LOCUS permits explicit remote log-on, explicit or implicit remote file access, and, most remarkably, process creation (forking) and intercommunication (piping) between any number of heterogeneous Unix systems. In the phenomenal number of commercial distributed-database and

specialized-processor resource applications in the market database at NOVON Research (the market research firm with which we are associated), this kind of system has the look of solid platinum.

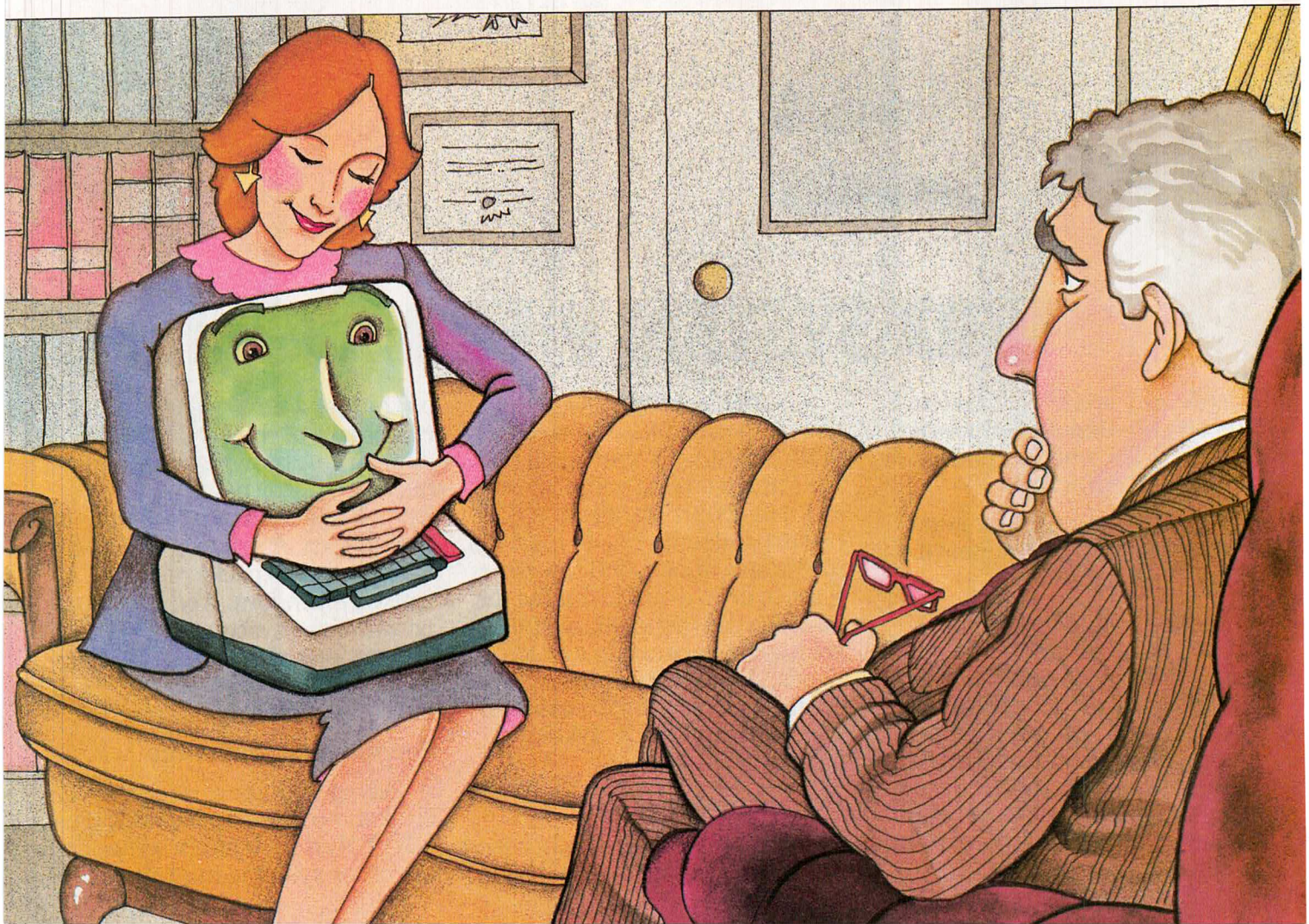
She came back to visit with young Carrot one day last week, and both were dramatically changed since our last meeting. The patient had a light behind his eyes for the first time, and the sparkle was back in hers, only slightly clouded by confusion.

"You've really worked miracles with Junior. He used to seem like such a woodenhead, and now he's as bright a lad as I could ask for. The only problem is that he occasionally seems somewhat distant—even remote—as if his mind were elsewhere. And there's the matter of those wires running out the back of his head. . . ."

"Don't worry your pretty little head about it—leave that to us who know what's best," I chided in the patronizing tone that has earned me animosity from both the National Organization of Women (NOW) and the All-World Society of Holistic Unix Computer Scientists (AW SHUCS). "We've had artificial limbs, kidneys, lungs, and hearts—an artificial brain was bound to come along sooner or later. Just think of the Unix system connection as a sort of mobile quality-of-life support system."

She seemed quite happy when they left together, the sunlight on their hair, the wind at their backs, seeing the day with their own eyes. It sent a small but disturbing voltage spike through my peripheral processing unit. Sometimes I'd give 256K of my main memory just to be in their human shoes. □

Brian Boyle, a UNIX/WORLD contributing editor, is director of research at NOVON Research, a San Francisco-based firm specializing in technical and market research into the next generation of systems and production software. John P. Dunham, program manager for the Computer and Office Products Division of Gnostic Concepts Inc. (San Mateo, Calif.), is a consultant to NOVON Research.



UNIX™ SYSTEM V. FROM AT&T. THE

There's only one UNIX System V. The one created and supported by AT&T. The one fast becoming a universal standard.

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

In more ways than one, it pays to make certain that the products you offer are based on UNIX SystemV. It is the stable foundation on which all of our future enhancements will be developed. It's here to stay because we pledge that our future software enhancements will be compatible with it.

Configure your systems now with products based on UNIX System V and you'll be at the forefront of its establishment as a worldwide standard. AT&T has expanded its role in markets in Europe, Japan, and the Far East to create opportunities for machine-independent, portable solutions to operating system needs. Around the world you can profit from our initiatives when you invest in UNIX SystemV solutions.

On the inside track

If you develop products based on UNIX System V, you'll be in on the latest advances from the custodians of the UNIX System standard. A growing number of microprocessor ports to choose from. Expanded networking capabilities. A growing number of applications packages from software vendors. And, the power of the system that spans generations of machines as well as models and sizes—from micros to mainframes.

Service from the source

Along with its various technical and commercial advantages, UNIX System V offers something the other systems can't match—

SYSTEM V

ONLY

STANDARD BEARER.

the full backing of AT&T.

That includes worldwide support and training programs. Our commitment to work with computer manufacturers and resellers as well as user groups to develop standard products for the future. And for software vendors writing applications programs, a complete service and support program from AT&T.

The one and only from AT&T is UNIX System V—backed by AT&T service and support—designed to make UNIX System V the one for one and all.

For more information on UNIX System V from AT&T, send in the coupon.

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

AT&T

Please send me "UNIX Software." #UW-0500-FF

Mail to: AT&T, Box 967, Madison Square Station
New York, New York 10159

Name_____

Title_____

Department_____

Company_____

Address_____

City_____ State_____ Zip_____

Phone_____

UNIX™ System Licensee ☐ Yes ☐ No ☐ Don't Know

Please circle Ad No. 60 on inquiry card.

10TH ANNIVERSARY

Portland oregon

USENIX Conference and Vendor Exhibition
Marriott Hotel
June 11-14, 1985

USENIX

THE SPONSOR

For the latest in UNIX* applications and research, people look to USENIX, a not-for-profit association of individuals and institutions dedicated to fostering the development and communication and UNIX and UNIX-like systems and C Programming language. USENIX sponsors technical conferences and vendor exhibitions, produces and distributes a newsletter and serves as coordinator of a software exchange for appropriately licensed members.

THE TECHNICAL SESSIONS

The technical sessions emphasize issues of interest to the UNIX community. Suggested topics for formal papers, panel discussions and informal discussions will include the following:

▲ Text Processing ▲ Graphics
▲ Compilers and Software Distribution
▲ Network File Systems
▲ Programming Environments
▲ Standards Activities
▲ Multiprocessing ▲ Window Managers

TUTORIALS

In-depth tutorial seminars will be conducted all day by leading UNIX technical experts. Topics include:

- ▲ UNIX internals
- ▲ Networking
- ▲ Communications
- ▲ Applications
- ▲ C Programming

VENDOR EXHIBITION

Vendors are invited to display advanced technology relevant to the UNIX technical community.

For Complete Conference Information, Call:

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

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

PYRAMID 90x: A RISC SUPERMINI

This month our crazed reviewer went down to the far reaches of Mountain View, Calif., for a long-overdue look at the Pyramid 90x superminicomputer, the first commercial Unix system RISC machine.

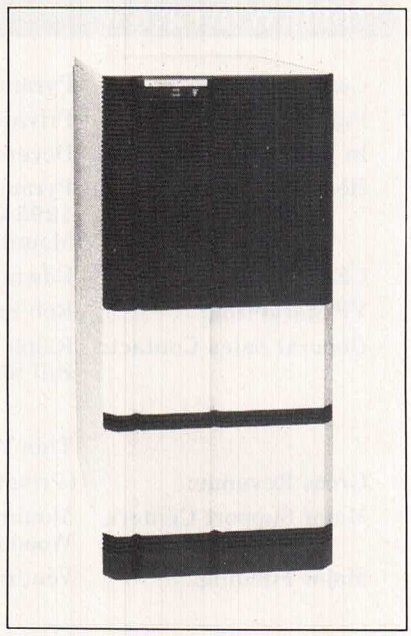
BY BRUCE R. MACKINLAY

The hardware is hot, the software cool—a combo that makes the Pyramid 90x a scorching Unix supermini. The Pyramid 90x is a $150,000 superminicomputer boasting (and showing) performance that is 1½ to 3 times better than that of a VAX 11/780. What makes this a useful and powerful machine is the combination of System V and Berkeley 4.2 Unix.

Anyone who has considered, or is using, a VAX running Ultrix or an IBM 4341 running VM/IX should take a long, hard look at the Pyramid 90x and its new sibling, the 90Mx. In fact, anyone who really believes in the "Keep It Simple, Stupid" (KISS) concept should look at the Pyramid machines. Both the hardware and software reflect a firm commitment to the principles of KISS.

The Pyramid hardware is based upon a reduced instruction set computer (RISC)-like processor that is a full commercial central processing unit (CPU), including floating-point hardware, caching, and all the necessary operating system instructions. The heavy use of input/output (I/O) processors makes the machine ideal for time-sharing. The serial I/O processor is excellent, and the use of an intelligent console processor makes the machine easy to maintain and repair.

For those not familiar with RISC, I'll be explaining what it means in



more technical detail later in this article. For now, let it suffice to say that the aim of a RISC architecture is to boost performance by reducing the number of instructions to the barest minimum possible. Kind of a "less is better" approach to designing computer hardware, you might say.

PYRAMID HARDWARE

What I like the most about the Pyramid is its hardware. The whole design is based upon the RISC machine, developed at UC Berkeley. I like the heavy use of intelligent I/O processors; I like the intelligent console processor; and I like the message-oriented XTEND bus. I really like the 16-channel serial I/O processor based upon a TTL state machine.

To understand my love of the RISC architecture and my immediate liking of the Pyramid 90x, you have to understand my background. Not only did I study computer science at UC Berkeley during the development of the RISC-I, but before that I wrote compilers and interpreters for the PDP-11 and other CISC (complex instruction set computers) machines. I know from personal experience the advantages of RISC. More important than my background is the fact that I am a firm believer in the KISS principles. I love anything that produces good results and keeps things simple. This is the essence of the Pyramid and the RISC architecture.

I mentioned that I like the Pyramid's heavy use of intelligent I/O processors. What I especially like is the 16-channel serial I/O card, containing a custom TTL state machine. This I/O card can handle a full 960 characters per second on *all* 16 channels, with no flow control. This intelligent I/O card contains most of the code normally in the kernel serial I/O driver.

On Unix systems, where the serial I/O is handled by the main processor in the kernel, serial I/O can take up a great deal of processor time, keeping useful work from being done. The other problem with the traditional approach to serial I/O is that the kernel tends to drop characters if the characters are received too fast.

The XTEND bus is a full 32-bit message-based bus with a 32-Mbytes/second bandwidth. The big advantage of a message-based bus is the ease with which it will permit the addition of multiprocessors, allowing them to share memory and I/O devices. The XTEND bus allowed Pyramid to introduce the 90Mx multiprocessor machine so quickly. The traditional weakness of message-based buses is the large overhead that results from message packaging, but this is overcome by the XTEND bus' very high speed.

What I consider to be another key to the success of Pyramid's hardware is the intelligent console processor. This is an M68000-based microcomputer that shares the bus with the CPU and all other devices. This console processor can stop and interrogate any device in the bus, including the CPU.

This processor was initially designed to reduce development time, but it was included in the final hardware configuration when it became clear that it made maintenance and repair a snap. The intelligent console processor used a color graphics terminal to help the operator monitor system resources. What's so nice about it is that you can do usage measurement without slowing down the central processor.

COMPANY OVERVIEW

Company Name:	**Pyramid Technology Corp.**
Public/Private:	**Private**
In Business Since:	**December 1981**
Headquarters:	**Pyramid Technology Corp. 1295 Charleston Road Mountain View, CA 94043**
CEO:	**Edward W. Dolinar, President**
VP Marketing:	**Rob Ragan-Kelley, VP Planning**
General Sales Contact:	**Ralph Reda, VP Sales Bill Shellooe, VP Int'l Sales**
	This Year Last Year
Gross Revenue:	**(Privately held, figures not available)**
Major Support Centers:	**Mountain View, Calif.; Des Plaines, Ill.; Woodbridge, N.J.; Chevy Chase, Md.**
Major Funding:	**Venture capital, $27 million**

REDUCED INSTRUCTION SET COMPUTERS

One of the most interesting things about the Pyramid 90x (and 90Mx) is the application of RISC concepts to a

BENCHMARK MEASUREMENTS

Aim Technology Suite II Pyramid 90x Supermini

Arithmetic Instruction Times (microseconds per op)
(With Floating-Point Hardware)

	short	*long*	*float*	*double*
+ Add	744ns	371ns	6	2
* Multiply	4	3	439ns	524ns
/ Divide	10	11	762ns	722ns

(Without Floating-Point Hardware)

	short	*long*	*float*	*double*
+ Add	744ns	371ns	12	8
* Multiply	7	6	794ns	883ns
/ Divide	10	11	1	1

Memory Loop Access Times (microseconds per byte)

	read	*write*	*copy*
Char type	1	2	3
Short type	705ns	968ns	1
Long type	357ns	359ns	613ns

Input/Output Rates (bytes/sec)

	read	*write*	*copy*
Disk	345K	449K	170K
Pipe			165K
TTY 1		954K	
TTY 1+2		2K	
RAM 1-byte			372K
RAM 4-byte			1630K

Array Subscript References (microseconds)

short[]	long[]
2	2

Function References (microseconds/ref)

0-parameters	1-parameter	2-parameters
funct()	funct(i)	funct(i,i)
6	6	6

Continued on page 78

commercially useful machine. I've already mentioned that RISC stands for "reduced instruction set computer," and it is a concept that seems to reappear every so often in computer architecture.

As I see it, RISC is the application of the KISS principles to computer instruction sets. The very first computers were RISC computers, but as time wore on engineers embellished them, adding instructions for this and instructions for that.

Every so often, someone with clearer vision comes along and says "This is stupid, every new instruction adds to the complexity of the computer and slows down the machine. What we need is a computer with a *simpler* instruction set, using only the instructions that are *truly needed.* We can concentrate on making these instructions faster." This vision is applied, and a new generation of better machines is born.

The Pyramid is a direct outgrowth of the most recent round of visionary thinking. David Patterson of UC Berkeley is credited with the vision—he coined the term "RISC." A RISC machine should be faster and cheaper than the older complex instruction set computers.

The VAX is a good example of what went wrong with the CISC computers. The VAX 11/780 has over a million bits of micro-code, with an instruction for almost everything; it seems as if it should have an instruction to scratch your ears. Consider the VAX instruction to terminate a "Do"-loop. If you use the three instructions (Inc, Test, and Branch), you actually produce faster code. In addition, it is hard to use the "Do"-loop instruction in your code generator. The net effect of all these complex instructions is slower programs and wasted resources.

Patterson—and other re-

Continued

BENCHMARK MEASUREMENTS

Process Forks

(45K bytes)
42 per second

System Kernel Calls
(calls-per-second and microseconds-per-call)

`getpid()` calls:	9K calls/sec or	109 microseconds/call
`sbrk(0)` calls:	126K calls/sec or	8 microseconds/call
`create/close` calls:	33 pairs/sec or	30303 microseconds/pair
`umask(0)` calls:	10K calls/sec or	102 microseconds/call

C Compiler Performance

3001 symbols, maximum
CPU estim: 7.2 sec + 82 lines/sec
Read estim: 1.1 times CPU time

searchers—looked at the complex mess of the CISC machines and decided to try a different (simpler) approach. They measured the usage of different instructions in running programs written in high-level languages (like FORTRAN). The upshot of this investigation was that only a few instructions really mattered. Eighty percent of all instructions reference either constants or local variables within a subroutine, and over 90 percent of all array references are global.

The "Call/Return" from/to subroutines are the most time-consuming operation in a typical program. Patterson also investigated the design limitations of recent VLSI (very large-scale integration) computers. Proceeding from these results, Patterson and a group of researchers at UC Berkeley designed and built what they called the RISC machine.

The four principles of the RISC design are as follows: *(one)* execute one instruction per cycle; *(two)* make all instructions the same size; *(three)* only load and store instructions that can access memory; and *(four)* support high-level languages.

These four principles produce a substantially cheaper machine that compares favorably in performance to the VAX 11/780. This machine also introduces the idea of register windows. A register window is a very clever trick intended to keep local variables on-chip, greatly reducing the need to fetch and store local variables before and after the "Call" statement. This hardware solution to a serious problem not only makes the resulting code faster, but it also makes it easier to write compilers for the RISC machine.

The Pyramid is more like an expansion of the original RISC design. The Pyramid uses the same concepts, like the register windows and 32-bit instructions, but the Pyramid is not a true RISC machine. It violates the RISC rules and has many more instructions than a true RISC machine should. In its defense, I have to point out that the original RISC-I also violated the rules, using two instruction cycles in the "Load" and "Store" instructions.

Additions to the original RISC design were necessary to make the machine viable in the commercial market. The original RISC did not have instructions for operating system calls, process changes, floating and packed arithmetic, and a whole plethora of things that commercial machines must have. Even with these additions, however, the Pyramid 90x is very much a RISC machine.

PYRAMID SOFTWARE

As you can see from the command completeness test, the Pyramid 90x has one of the most complete command lists of any Unix system machine. This is accomplished by combining AT&T System V and Berkeley 4.2. This combination is not just the addition of System V commands to a Berkeley 4.2 Unix, but is a true marriage. You can use the System V version of a command and can later use the Berkeley 4.2 version; both are there.

To make this work, Pyramid introduces a concept of "universes," with one universe for AT&T and another for Berkeley (funny how technology mirrors reality). You can switch from one universe to another any time you want. When reviewing the machine, I stuck mostly to the Berkeley universe, but I played around in the AT&T universe to make sure that it worked as advertised.

The implementation of this dual-universe Unix system is another example of the KISS principle; in fact, I don't think it would work without it. What Pyramid did was to start with a 4.2 kernel (since this is a super-set of System V) and add kernel "Call" entry points for System V. Berkeley 4.2 Unix has a special directory entry called a symbolic

The Power of Expression
The Expression of Power

Now, more than ever, expression of ideas through the printed word is a cultural imperative. Just getting through a day means dealing with massive amounts of information. And we each have a duality of roles — we consume information, we produce it.

As consumers of information, we must make choices. We choose material that appears more interesting, that is credible and clearly presented, that expresses a dimension of power through its appearance.

As producers of written information, we have the burden of creating that dimension. We have the responsibility to insure that our expressed thoughts become conveyed ideas.

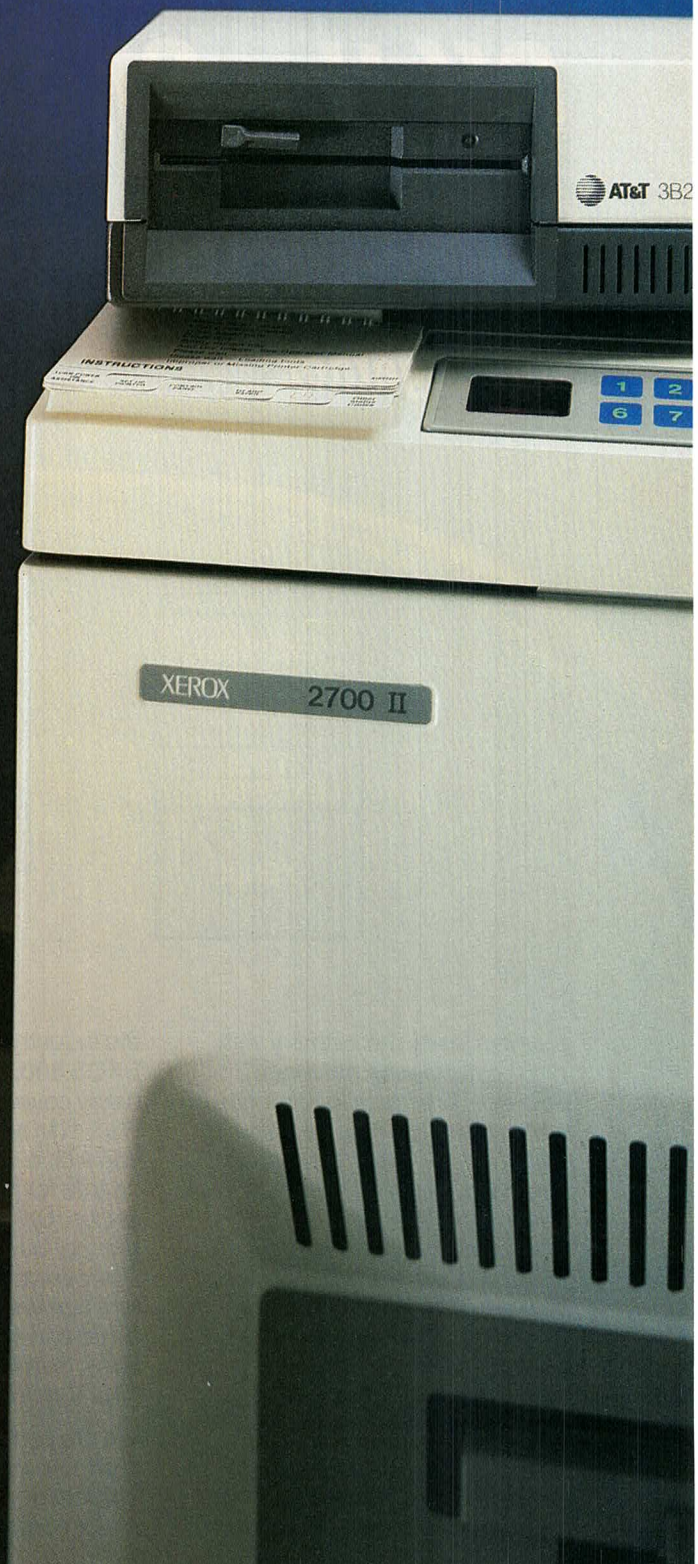
Typesetting offers the solution. With the ability to choose typestyles and sizes, control spacing and position, you can give your message the added impact it needs to cut through the mass of material competing for your readers' attention. Contracts, correspondence, memos, reports, forms, documentation, training materials, everything you produce can carry the extra dimension of power that gets it read and remembered.

Laser xerography provides the means of typesetting in-house, effectively, immediately. VLS, through our *CompWare** product line, provides the means of using laser xerography.

Call now for more information on *DX-2700**, software support for the Xerox 2700* laser printer; our *DWB** release for AT&T's 3B* processors; VAX/UNIX* BSD Laser Printer support; and other *CompWare*products.

We'll show you how to express the power in your corporate publishing.

*DWB and 3B are trademarks of AT&T Technologies; DX-2700 and CompWare are trademarks of VLS, Inc.; Xerox 2700 is a trademark of the Xerox Corporation; UNIX is a trademark of AT&T Bell Labs; VAX is a trademark of Digital Equipment Corporation.

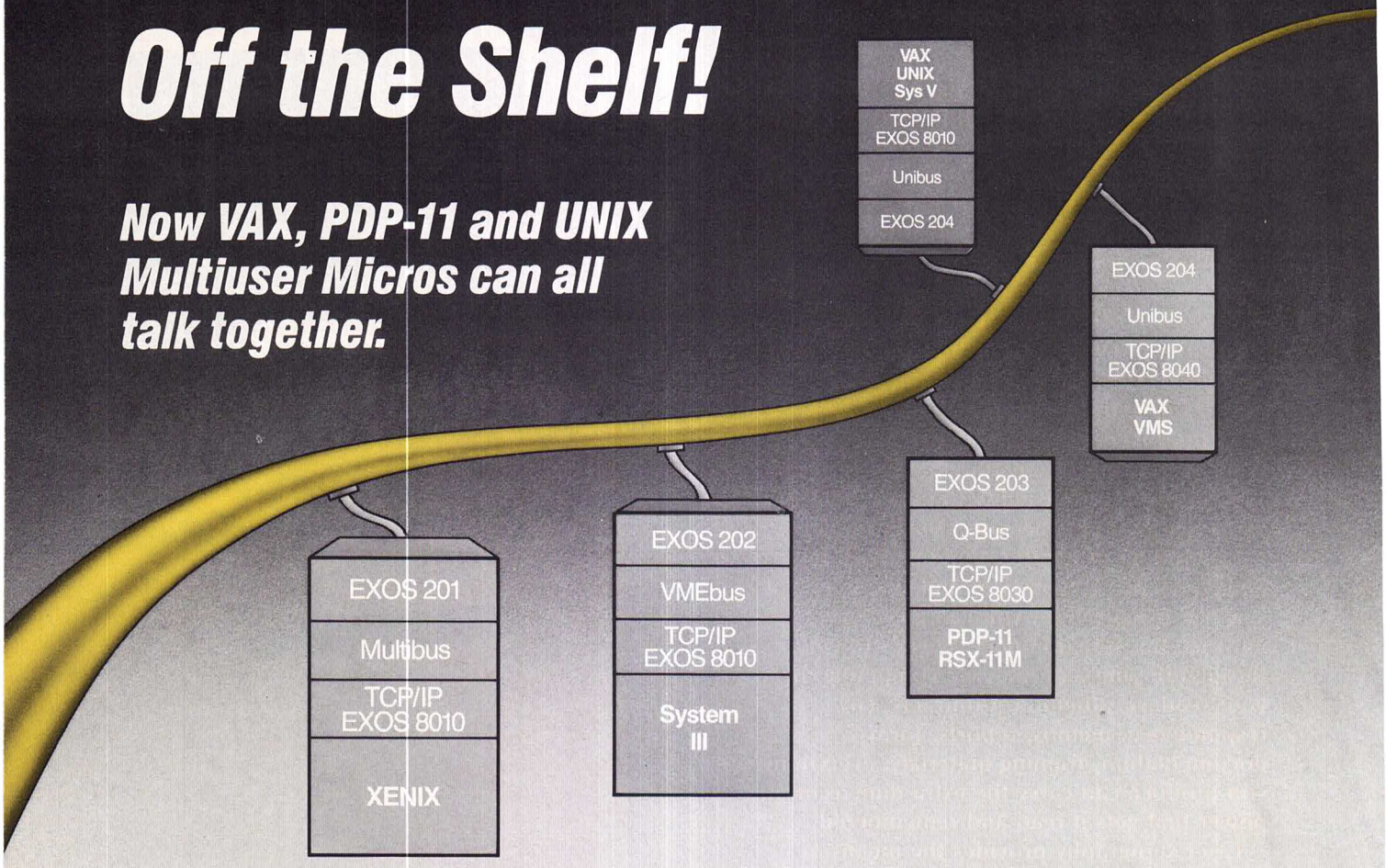


VLS INC. 44000 OLD WARM SPRINGS BLVD. FREMONT, CA 94538 (415) 490-3555

Please circle Ad No. 110 on inquiry card.

Complete Ethernet Solutions. Off the Shelf!

Now VAX, PDP-11 and UNIX Multiuser Micros can all talk together.



Excelan leads the industry with high performance hardware, software, and tools for your networking needs. When we say solutions, we mean everything you need. From front-end processor boards, TCP/IP protocol and application software suite down to our rugged 802.3 transceivers. And we offer the only Ethernet monitor and debugger on the market today.

Excelan pioneered off the shelf networking solutions with our EXOS intelligent front-end processors and our complete, high level protocol software based on the industry standard TCP/IP. Our EXOS 8000 Series software already covers UNIX (any version), RSX-11M and VMS in combination with our EXOS 200 Series boards for Multibus, VMEbus, Q-Bus, Unibus, or your own proprietary bus. And we guarantee interoperability — so your system can talk to computers with B-net and 4.2 BSD implementations, Sun Workstations, VAX (Sys V or VMS) and many more.

We are setting the pace. Our products are already in their third generation, in place, working, in major companies throughout the country. And major OEMs are expediting their design cycles by using our products. Furthermore, our solutions will keep on offering value, because our products have been designed to meet your evolving networking needs.

Call us and talk solutions. We are doing more than anyone else to put you on the right track — and keep you there.

EXCELAN

2180 Fortune Drive
San Jose, CA 95131
(408) 945-9526 TELEX 176610

Please circle Ad No. 17 on inquiry card.

Visit us at DEXPO South, booth #1033.

HARDCOPY Magazine Brings Unix and DEC* Together

Keep abreast of the latest trends in the world of Unix. Track efforts to standardize. Discover how to speed up Unix disk accesses on your DEC systems without spending more money. Learn how to interface Unix and VMS. Stay informed with the latest news on DEC hardware changes. Know about DEC Unix user group meetings. Find out in HARDCOPY.

All The Facts

HARDCOPY gives you the facts you need to be more efficient using Unix on your DEC system. With regular columns like The Unix Connection, The DEC Troubleshooter, Business Software Beat, and monthly articles covering Unix applications, innovations and trade shows, our experts explain how your software and hardware work together, and provide tips to improve your system's performance. Feature articles explore every aspect of DEC computing, from laser printers to application software. So you get Unix — and more — in every issue.

HARDCOPY gives you valuable information about the changing DEC world — with practical ideas that will save you time and money.

Act Now!

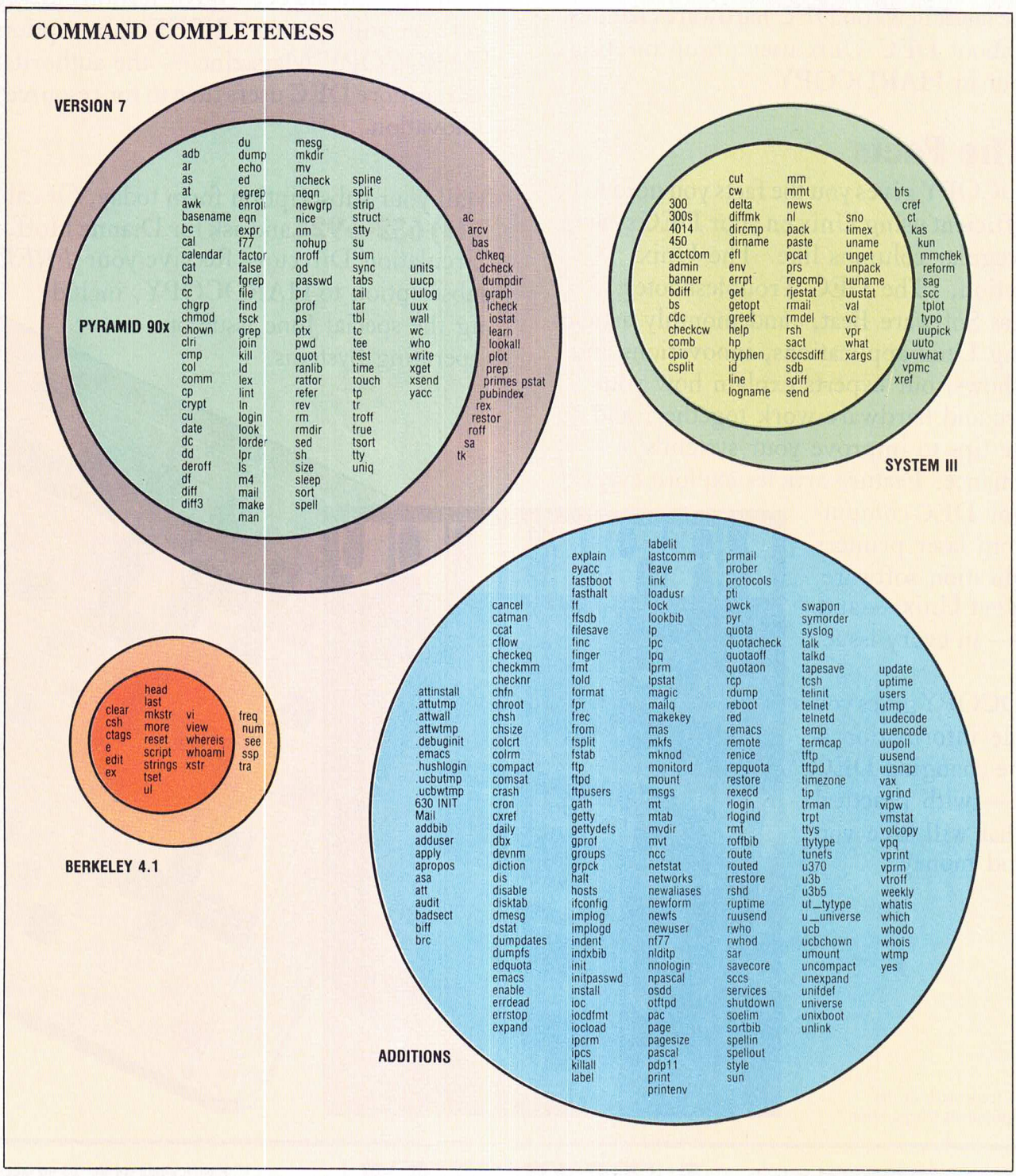
For the keys to more efficient processing using Unix with DEC, subscribe today to the leading DEC-specific publication — HARDCOPY. Complete the attached qualification card today and you will receive a FREE subscription to HARDCOPY Magazine — the authoritative source more DEC users turn to for resources and innovation.

Mail your subscription form today. Or call **(714) 632-6924** and ask for Dianne Hoff, Circulation Director. Receive your *FREE* subscription to HARDCOPY, including the special June issue on Operating Systems.

*DEC is a trademark of the Digital Equipment Corporation.

HARDCOPY
WHERE DEC USERS TURN FOR RESOURCES AND INNOVATION
(714) 632-6924

Please circle Ad No. 40 on inquiry card.


COMMAND COMPLETENESS
VERSION 7
PYRAMID 90x
adb ar as at awk basename bc cal calendar cat cb cc chgrp chmod chown clri cmp col comm cp crypt cu date dc dd deroff df diff diff3
du dump echo ed egrep enroll eqn expr f77 factor false fgrep file find fsck grep join kill ld lex lint ln login look lorder lpr ls m4 mail make man
mesg mkdir mv ncheck neqn newgrp nice nm nohup nroff od passwd pr prof ps ptx pwd quot ranlib ratfor refer rev rm rmdir sed sh size sleep sort spell
spline split strip struct stty su sum sync tabs tail tar tbl tc tee test time touch tp tr troff true tsort tty uniq
units uucp uulog uux wall wc who write xget xsend yacc
ac arcv bas chkeq dcheck dumpdir graph icheck iostat learn lookall plot prep primes pstat pubindex rex restor roff sa tk
SYSTEM III
300 300s 4014 450 acctcom admin banner bdiff bs cdc checkcw comb cpio csplit
cut cw delta diffmk dircmp dirname efl env errpt get getopt greek help hp hyphen id line logname
mm mmt news nl pack paste pcat prs regcmp rjestat rmail rmdel rsh sact sccsdiff sdb sdiff send
sno timex uname unget unpack uuname uustat val vc vpr what xargs
bfs cref ct kas kun mmchek reform sag scc tplot typo uupick uuto uuwhat vpmc xref
BERKELEY 4.1
clear csh ctags e edit ex
head last mkstr more reset script strings tset ul
vi view whereis whoami xstr
freq num see ssp tra
ADDITIONS
.attinstall .attutmp .attwall .attwtmp .debuginit .emacs .hushlogin .ucbutmp .ucbwtmp 630 INIT Mail addbib adduser apply apropos asa att audit badsect biff brc
cancel catman ccat cflow checkeq checkmm checknr chfn chroot chsh chsize colcrt colrm compact comsat crash cron cxref daily dbx devnm diction dis disable disktab dmesg dstat dumpdates dumpfs edquota emacs enable errdead errstop expand
explain eyacc fastboot fasthalt ff ffsdb filesave finc finger fmt fold format fpr frec from fsplit fstab ftp ftpd ftpusers gath getty gettydefs gprof groups grpck halt hosts ifconfig implog implogd indent indxbib init initpasswd install ioc iocdfmt iocload ipcrm ipcs killall label
labelit lastcomm leave link loadusr lock lookbib lp lpc lpq lprm lpstat magic mailq makekey mas mkfs mknod monitord mount msgs mt mtab mvdir mvt ncc netstat networks newaliases newform newfs newuser nf77 nlditp nnologin npascal osdd otftpd pac page pagesize pascal pdp11 print printenv
prmail prober protocols pti pwck pyr quota quotacheck quotaoff quotaon rcp rdump reboot red remacs remote renice repquota restore rexecd rlogin rlogind rmt roffbib route routed rrestore rshd ruptime ruusend rwho rwhod sar savecore sccs services shutdown soelim sortbib spellin spellout style sun
swapon symorder syslog talk talkd tapesave tcsh telinit telnet telnetd temp termcap tftp tftpd timezone tip trman trpt ttys ttytype tunefs u370 u3b u3b5 ut_tytype u_universe ucb ucbchown umount uncompact unexpand unifdef universe unixboot unlink
update uptime users utmp uudecode uuencode uupoll uusend uusnap vax vgrind vipw vmstat volcopy vpq vprint vprm vtroff weekly whatis which whodo whois wtmp yes

A COMPARISON OF PYRAMID 90x AND VAX 11/780

Note: The following ratios reflect the number of times faster that the Pyramid 90x is over the VAX 11/780 running Ultrix. The amounts in "()"s show where the 780 out performs the 90x.

* Numbers not available for either the 780 or the 90x
** This number is meaningless for both the 780 and the 90x

Arithmetic Instruction Times (With Floating-Point Hardware)

	short	*long*	*float*	*double*	
+ Add	2.69	1.62	(.83)	1.50	
* Multiply	(.75)	(.67)	15.95	9.54	
/ Divide	2.10	(.91)	14.44	13.85	Avg.: 4.88

(Without Floating-Point Hardware)

	short	*long*	*float*	*double*	
+ Add	2.69	1.62	(.42)	(.38)	
* Multiply	(.43)	(.33)	8.82	5.66	
/ Divide	2.10	(.91)	11.00	10.00	Avg.: 3.29

Memory Loop Access Times

	right	*write*	*copy*	
Char type	2.00	1.00	(.67)	
Short type	1.11	(.82)	(.78)	
Long type	(.83)	1.12	(.66)	Avg.: 1.00

Input/Output Rates

	right	*write*	*copy*	
Disk	1.71	3.16	2.30	Avg.: 2.39
Pipe			(.68)	Avg.: (.68)
TTY 1		*		
TTY 1+2		*		
RAM 1-byte			(.57)	
RAM 4-byte			(.66)	Avg.: (.21)

Array Subscript References

short[]	long[]	
1.00	1.00	Avg.: 1.00

Continued

link. This allows 4.2 users to link to another directory, making its contents easily available.

Pyramid extended the symbolic link to make a conditionally symbolic link. This is really a two-way branch in the file system. If the universe is a UC Berkeley universe, then the `/bin` directory links to `/.ucbbin`; if it is the AT&T universe, it links to `/.attbin`. In this way the AT&T and UC Berkeley versions of the same command can exist at the same time. This is simple, clean, easy, and much better than versions from competitors.

Pyramid recently announced the addition of Sun's Network File System, which allows a user to read and write directly to a remote disk file as if it were local. (This was the major piece missing from Berkeley 4.2 when it was released.) This news should be very exciting to Sun users, allowing them to use a Pyramid 90x and 90Mx as a host computer. This cooperation between computer companies, although rare, is very wise, and it should make both companies stronger.

Pyramid is also making all its compilers use the same code generator, optimizer, and linker. This makes the generated code compatible. It also has the advantage of making it possible to improve all compilers by making improvements in the common parts. While I was visiting Pyramid, I fooled around with its new optimizer, which is an example of the effect of a good optimizer.

As you can see from Figure 1, a good optimizer can make more difference than floating-point hardware. With both the optimizer and floating-point hardware, the 90x achieved 36K FLOPS (floating-point operations per second). One interesting thing about the optimizer is

A COMPARISON OF PYRAMID 90x AND VAX 11/780

Continued

Function References

0-parameters funct()	1-parameter funct(i)	2-parameters funct(i,i)	
2.83	3.67	4.33	Avg.: 3.61

Process Forks

1.50

System Kernel Calls

`getpid()` calls:	1.68	
`sbrk(0)` calls:	**	
`create/close` calls:	1.10	
`umask(0)` calls:	2.00	Avg.: 1.59

what it did to Suite 2 of the Aim benchmarks. The optimizer removed so much code that the benchmarks were no longer measuring anything.

BENCHMARKS

"Benchmarks, like benches, are for the birds."

Benchmarks are a lot like IQ tests: They measure something, but we don't know if they measure anything of any value. On the Pyramid, the weakness of benchmarks becomes even clearer. There are two problems. First, the buying public wants one number, like an IQ score, to compare machines. Second, as machines become more complex, less Von Neumann, it becomes very hard to measure machines with even a whole suite of tests.

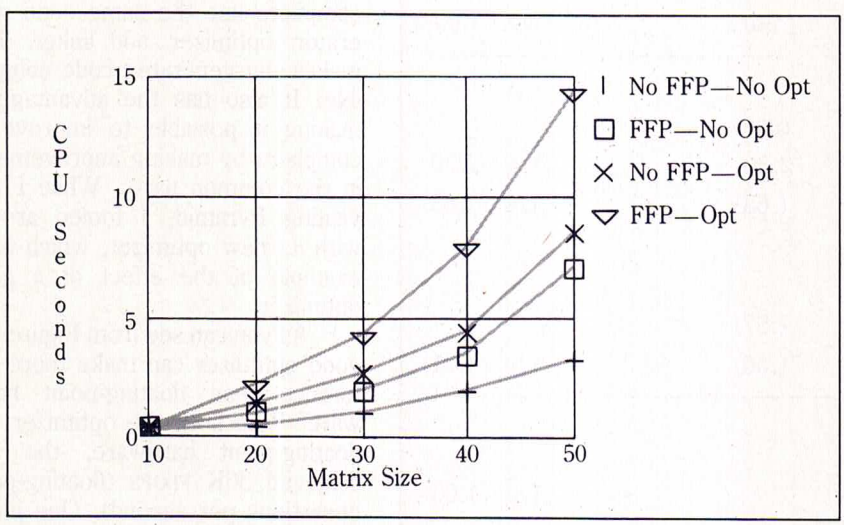


FIGURE 1: TIMES TO INVERT DIFFERENT MATRICES

The IQ test is a very good example of our desire to measure something with one number. Consider the arrogance of trying to measure a person's intelligence with a single number. Yet not only do we do it, but we make serious decisions based upon the one laughable number. We do the same thing with benchmarks. Every time a new machine enters the market, the buying public immediately wants to know how many MIPS (also known as "mythical instructions per second") the machine can perform, even though we all know that MIPS have very little relationship to reality.

Consider the Pyramid 90x. Because of its RISC-like architecture, it is much easier to estimate the number of instructions per second. In essence, the Pyramid performs one instruction in every cycle. With a 125-nanosecond cycle time, this gives it an 8-MIPS computed rating. But when you look closer, you see that it does not execute every instruction in one cycle; the floating-point instructions take longer (not a lot), and the state switch instructions even longer. I don't know how many mythical-instructions-per-second the Pyramid 90x does, but I do know that it is not a 8-MIPS machine.

The Aim benchmarks, which measure different capabilities separately, are a better measure of a machine's performance. The Aim benchmarks give us different numbers for floating point, disk I/O, and so on. However, even with the Aim benchmarks, we don't really know how fast (or slow) the Pyramid is.

One approach is to compare it to a well-known machine like a VAX. We all know how fast (or slow, depending on whether you have one) a VAX 11/780 is. The benchmarks show a VAX 11/780 compared to the Pyramid 90x. As you can see from

the benchmark results, the Pyramid 90x outperforms the VAX 11/780 in a number of ways. The floating point is faster, memory speed is about the same, and the disk rates are faster.

What the benchmarks do not show is that in many application environments the Pyramid should be significantly faster than the VAX 11/780. One important reason for this is Pyramid's use of highly intelligent I/O processors. For the VAX 11/780 to achieve its very high disk rates (though slower than the Pyramid), it must use a much larger amount of the CPU. The benchmark does not show that when the Pyramid is at maximum disk rate, its CPU is free.

There are, however, two other problems with the benchmarks. When I first ran them, the floating-point and `sbrk(0)` speeds were much higher than reasonable. After some discussion with Pyramid, we discovered that their hardware/software handled the special cases in the Aim code in an unusual way.

The floating-point hardware/firmware detects division and multiplication by 1. The Aim benchmarks used this special case to avoid floating-point over- and under-flow. I "munged" with the code, making the division and multiplication by 1.001 and produced reasonable numbers.

The `sbrk(0)` was another problem. The function `sbrk` is a system call requesting memory. On most machines, an `sbrk()` of zero produces a system call but does not allocate memory; on the Pyramid, it does nothing. In other words, on the Pyramid `sbrk(0)` does not measure the system kernel, but it is more like a measurement of Loop time. I considered a patch for this, but could not come up with one. In all, this is a net win for Pyramid, but I did not consider it when comparing it to the VAX 11/780 in the benchmarks.

A BENCHMARK CASE STUDY

Consider Company X (which wishes to keep its name secret for now). Company X grew to a point where it was purchasing a new 11/70 every other month, and soon it would be a new computer every month, then every week. Something drastic had to be done. Because the company's programs were written in DEC BASIC-PLUS, this promised to be a very painful conversion. Company X

HARDWARE OVERVIEW

Model:	**Pyramid 90x**
Price:	**$150,000**
Configuration:	**4MB, 32 users, console OSx, CPU**
First Delivered:	**October 1983**
Related Models:	**Pyramid 90Mx**
PROCESSOR	
CPU:	**90x (RISC)**
Cycle Time:	**125 nanosecond**
Bus:	**XTEND bus (32 Mbytes/sec.)**
Cache Size	**4K-byte instruction cache to 32K-byte data cache**
Min. Memory:	**4 Mbytes**
Max. Memory:	**16 Mbytes**
Floating Point:	**Other / Firm/Hard ware**
STORAGE MEMORY	
Floppy:	**N/A**
Hard Disk:	**160 MB, 415 MB, 300 MB removable**
Backup:	**1600/6250 9-track magnetic tape**
I/O Processor:	**ITP, SSP, IOC**
OTHER HARDWARE	
Serial Ports:	**16 to 128 Async**
Serial I/O Processor:	**Custom State Machine**
LAN medium:	**Ethernet/Co-ax**
Protocols:	**TCP/IP (Network File System)**
STANDARD SOFTWARE	
Unix Version:	**Full System V and 4.2BSD (at the same time)**
Shells:	**Bourne and C**
Libraries:	**GKS graphics library**
Utilities:	**PWB, vi, EMACS, SDB**
Languages:	**C, Pascal, FORTRAN-77, Lisp**

needed a new machine that would handle the load for a long time, and it would need to rewrite its programs into a more common, more portable, language. Clearly this is a job for Super-Unix. The only problem is that Super-Unix was still an infant on Krypton.

The systems department, not having any other choice, decided it would make its decision based upon the performance of different machines (so good so far) and set out to benchmark a number of machines (also good—a rational decision). The head systems programmer and manager of the department measured the current applications programs to see which ones used the most resources, reasoning that if he could find a machine where these "dogs" ran well, then the rest of the programs should do well (again, this sounds reasonable).

After some effort, a number of representative programs were selected, but one of them (we'll call it Program S) was such a dog and was run so often that it was considered the most important. Program S was chosen as the benchmark for Company X's applications.

To make a very long and painful story short, Systems programmer "P" was dispatched to run this program on a number of chosen machines. To do this, he rewrote the program into something more portable (FORTRAN). At this point, a serious mistake was made. The new, portable version of program S used real arrays, whereas the old version (because of the limitation of addressing modes) used "virtual arrays."

In BASIC-PLUS, a "virtual array" is a disk file that behaves like an array. System programmer "P" returned a few weeks later with the good news. "I ran program S on the VAX 11/780, and it ran 20 times faster than on the 11/70. Even better, the migration will be easier because the data files are compatible." Everyone cheered, and Company X invested its very future into a costly conversion to VAX and VAX-FORTRAN.

Now Company X is buying a VAX 11/780 every month (it seemed that disk I/O should have been measured, not program S), and programmer P is older and wiser. The moral is this: "Using random programs as benchmarks produces random results." Carefully tailored programs are better. What Company X should have done was to measure the amounts of disk I/O, floating-point operations, terminal I/O, and system calls under different load conditions.

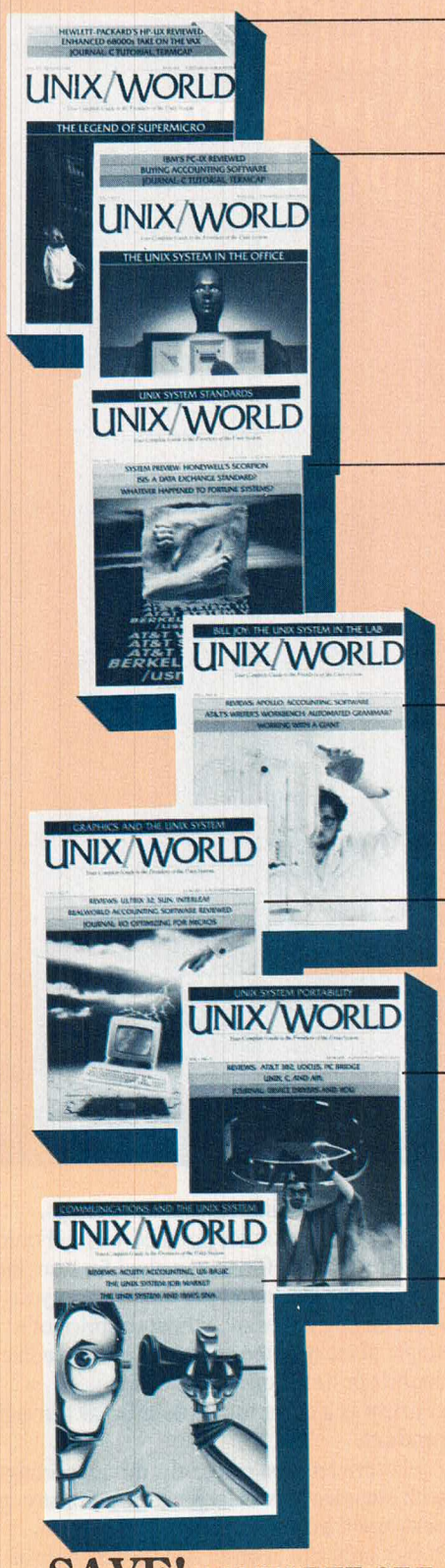
The other moral, even though it contradicts the first, is as follows: "No benchmark will tell you precisely how a new machine will perform with your application in your environment." This is why the Super-Unix system is so important. If the people at Company X had chosen Unix-land over VAX-land, then they could have found the best machine (even if they made a few false starts), and they would never need to place their company's life on the conversion block again. Now, if they can face the truth, they will have to go through another conversion. I hope that they choose the Unix system this time.

CONCLUSION

The Pyramid 90x, at $150,000, is an excellent machine. It embraces the best of the current technology and research. The developers are real risk takers (no pun intended), using RISC architecture and combining Berkeley 4.2 with System V. I found the machine comfortable and complete. Technically, I could find nothing really wrong with it or its design. So my advice is this: If you are in the market for a new supermini, look at the Pyramid. □

Bruce Mackinlay (sun!idi!uworld!novat!brucem) is a frequent contributor to UNIX/WORLD. His last article was a review of the Plexus P/35 (in our April 1985 issue). He is currently vice president of research and development at WMZ/Novatech, Concord, Calif.

THE UNIX/WORLD COLLECTION



VOL. I, NO. 1
The Legend of Supermicro

VOL. I, NO. 2
The Unix System in the Office

VOL. I, NO. 3
Unix System Standards

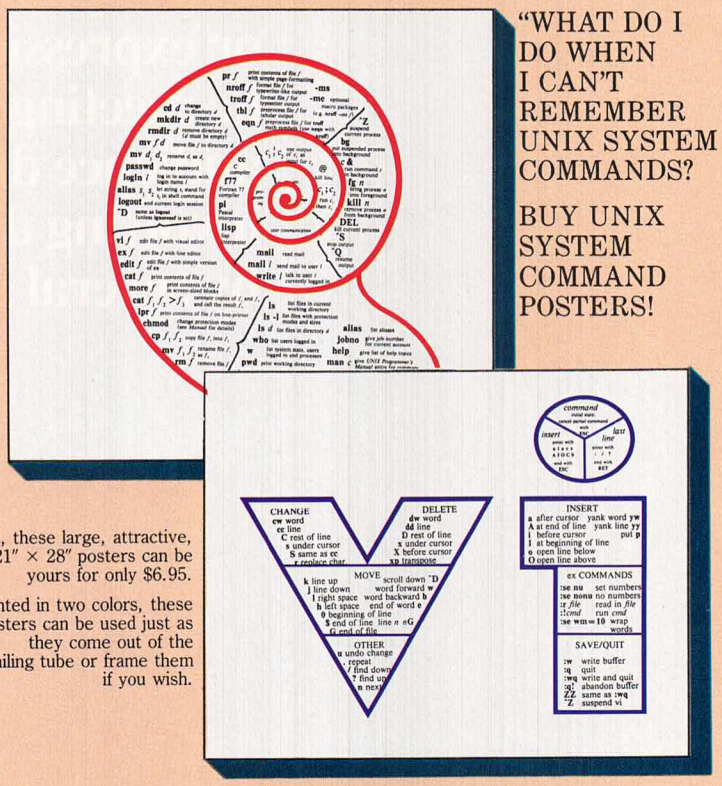
VOL. I, NO. 4
The Unix System in the Lab

VOL. I, NO. 5
Graphics and the Unix System

VOL. I, NO. 6
Unix System Portability

VOL. I, NO. 7
Communications and the Unix System

SAVE! COMPLETE SET OF SEVEN ISSUES ONLY $24.00



"WHAT DO I DO WHEN I CAN'T REMEMBER UNIX SYSTEM COMMANDS?

BUY UNIX SYSTEM COMMAND POSTERS!

Yes, these large, attractive, 21″ × 28″ posters can be yours for only $6.95.

Printed in two colors, these posters can be used just as they come out of the mailing tube or frame them if you wish.

FOR FASTER SERVICE Charge Card Users Call
TOLL-FREE: (800) 632-7979 (800) 227-6703
(California Residents Only)

or mail coupon to: UNIX/WORLD MAGAZINE
125 Main Street
Half Moon Bay, CA 94019

☐ Enclosed is my check (or charge to credit card below) for payment in full including charge for handling and shipping.

Charge Credit Card: ☐ MasterCard ☐ VISA

Exp. Date ___ / ___ Signature _____

Credit Card Acct. No. _____

Name _____ Company _____

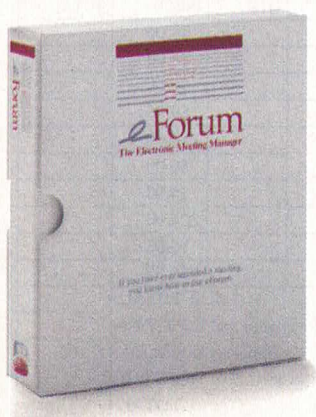
Address _____

City _____ State _____ Zip _____

CODE	ITEM	QTY	PRICE	TOTAL
U101	Vol.1 No.1		$6.00	
U102	Vol.1 No.2		$6.00	
U103	Vol.1 No.3		$6.00	
U104	Vol.1 No.4		$6.00	
U105	Vol.1 No.5		$6.00	
U106	Vol.1 No.6		$6.00	
U107	Vol.1 No.7		$6.00	
U100	Complete Set, No. 1–7		$24.00	
U301	Shell Poster		$6.95	
U300	VI Poster		$6.95	
			SUB-TOTAL	
			California Residents add 6.5% TAX	
			Handling & Shipping	$1.00
			Shipments Outside U.S.A. Add $7.00	
			TOTAL	

BIP55

fo · rum, n. (pl. FORUMS)
1. A public meeting place for open discussion. 2. A medium (as a newspaper) of open discussion or expression of ideas. 3. A public meeting or lecture involving audience discussion. 4. A program involving discussion of a problem by several authorities.



*eForum designed by Marcus Watts, Copyright 1984, Network Technologies International, Inc. (NETI).

Electronic meetings continue the automation of knowledge transfer which started with electronic mail.

Electronic meetings are an extension of the communications revolution which started with electronic mail. It takes seconds to send a letter using electronic mail instead of days via regular mail. Certainly e-mail is a giant step in automating correspondence between two people.

eForum goes yet further to provide immediate communications automation. But for groups. It creates electronic meetings which allow attendees to participate in discussions using the dynamic ebb and flow of points, counterpoints, comments and conclusions just like in-person meetings.

From an economics point-of-view, eForum is the most cost effective method for bringing together the best minds in your company to meet on key issues—without the price of a single plane trip, the aggravation of schedule conflict or time-consuming delay.

eForum is a communications breakthrough product.

eForum lets you create electronic meetings with attendee lists as large as the company staff or as small as a three-person design team.

Not only can eForum handle hundreds of meetings for your company, but, at the same time, limits each participant to only attending meetings to which he belongs.

eForum, n. 1. Low cost electronic meeting system (as in needing no scheduling or travel to attend). v. 1. Automatically organizes, indexes, files and leaves a complete written record of entire meeting. 2. Allows adding more attendees than normal at no extra cost. 3. Gives plenty of time to think before responding. adj. 1. Keeps everyone up-to-date. 2. Doesn't let geographic or time zones determine who can attend the meeting.

The Electronic Meeting Manager

If you have ever attended a meeting, you know how to use eForum.

Simply attend eForum meetings any time convenient for you. Review new discussion materials. eForum keeps track of what you've seen. Enter your comments or new discussion points. Instantaneously, your ideas are available to every member of your eForum group regardless of geographic location. That's productivity.

eForum has the flexibility to fit your communications needs.

- eForum 4000 - a national communications network available with a local phone call from most locations.
- eForum 2000 - UNIX™ based central host software for supermicro and minicomputers.
- eForum WS - software for the IBM PC and compatibles to interact with eForum central host software.

Call 1-800-638-4832 or in Michigan call (313) 994-4030 collect for information on:

- Automating your company's meetings by using General Electric Information Service, the world's largest communications network, to tie together your microcomputers and terminals.
- Creating your own meeting network for your department or company. Software, hardware and leasing available.
- Establishing OEM and VAR agreements to enhance the value of your software or hardware, with the communications power of eForum.

Network Technologies International, Inc.
The Arbor Atrium Building
315 West Huron
Ann Arbor, Michigan 48103

™UNIX is a trademark of AT&T Bell Laboratories
™eForum is a trademark of Network Technologies International, Inc.
(NETI)

Please circle Ad No. 66 on inquiry card.

HORIZON'S LATITUDE ON THE AT&T 3B2/300

This program charts a laudable course between word-processing and spreadsheet applications.

BY HARRY AVANT

Horizon's Latitude integrated word processor and spreadsheet is a beginner's dream. It is easy to set up and use, and it includes extensive support for most popular printers. Fast typists, however, will find that they can outtype the display, although they won't lose any characters.

Latitude is an application program that combines word-processing and limited spreadsheet functions. Produced by Horizon Software Systems Inc. (San Francisco), the program is available for several different Unix-based systems. For this evaluation, I used an AT&T 3B2 Model 300 running Unix System V and a Teletype 5420 terminal. I tested printing features using a C.Itoh 8510A and a NEC 5530 printer. The 3B2 version I tested is intended for both small-business and departmental applications.

Horizon has designed Latitude for users who prepare documents containing both text and tabular (spreadsheet) material. In many cases, Latitude's displays on the terminal screen are very close representations of how the final printed text will appear.

Latitude files use the Unix system hierarchy of directories, but

Horizon's Latitude software runs on the most popular Unix system hardware, including AT&T's 3B2, shown here.

Horizon has adopted an optional file-naming metaphor that makes life easier for inexperienced users. Within each user's home directory, the program opens a file or "File Cabinet." Next, it creates a "drawer" or drawers, located in the file cabinet—this is a subdirectory. The program then creates an actual file, which is a third-level directory. Because you can select available file cabinets, drawers, and individual files from the opening menus, locating specific subdirectories and files is simple.

You can perform a fair amount of file manipulation without being burdened by considerations of operating system nomenclature. For example, one of the available command menus deals with file copies and allows for single keystroke selection of such items as putting a copy of a file in another drawer—that is, copying a file from one directory to another. Other file parameters, such as making a file public or private, are also available.

Backup file protection is also considered, with `archive` and `retrieve` commands vailable. The operation of these programs depends on how the system administrator has set up the system and on what computer is being used.

PRELIMINARY STEPS

Using Latitude requires a few preliminary steps. Unlike `vi`, Latitude is not designed to be invoked with a filename for automatic loading of your text file. After you have logged in and called up Latitude, the "Open a File Cabinet" menu appears. From this menu you can choose to open one of the file cabinets listed on the screen, exit the program, request help, or select a menu of other commands. This first menu lists all files in your current directory, not just those in file cabinets.

After you have chosen a file cabinet, a second menu appears. This menu contains a list of available drawers, and you can then select, create, rename, or delete a drawer.

When you select a drawer, either an existing one or one you have just created, the file menu appears. A listing of all files available in the selected drawer appears, as well as commands for creating a new file, deleting or renaming a file, or opening a file so that you can work on it.

Once you have made a choice from this menu, the program loads your file and the "Edit Command" menu appears. You can now select the `a` (for append) command and begin entering text.

WORD PROCESSING

During text entry, the video screen is divided into three parts. At the top is a panel indicating the various commands available. The center portion (or text panel) provides space for 18 lines of text. A ruler line at the top of the text shows locations of left and right margins and default tab settings. The bottom of the screen contains a prompt line for possible commands and a cursor-position indicator showing current page and line numbers. This last line also shows the name of the file you are currently editing.

During word processing, Latitude operates in either its command mode or its "edit" mode. While in command mode, Latitude interprets keystrokes as program

PRODUCT AND COMPANY OVERVIEW

Price: $595 and up (version as tested, $995)

COMPANY OVERVIEW:

Corporate:

Company name:	**Horizon Software Systems**
Public/private:	**Private**
Year founded:	**1979**
Headquarters:	**China Basin Bldg. 185 Berry St. Suite 4821, San Francisco, CA 94107**

Management:

President:	**Joe Walsh**
Executive VP, Director of planning:	**Paul Miller**
General sales contact:	**Alison Baker (415/543-1199)**
Number of employees:	**12**
Major support center:	**San Francisco**
Major funding:	**Privately funded**
Current ports:	**AT&T 3B2; DEC VAX 730, 750, 780, 785, MicroVAX; Altos 586, 986; Sun-2; IBM-PC, PC-XT, PC-370, PC/AT**

commands; in edit mode, keystrokes are actual text input. Within the edit mode, you can select insert or overstrike "Control-T," which acts as a toggle between the two. To exit from edit mode and to enter command mode or vice versa, you press "Control-C."

Horizon's technical support group truly understands its product and many of the systems on which it is designed to operate.

Latitude contains an adequate number of text-manipulation and formatting commands. Full cursor movement throughout the text by character, word, sentence, and paragraph is available. You can toggle word-wrap or auto-carriage return on or off, and you can also toggle between insert and overstrike modes. The program provides format commands that operate on lines, paragraphs, and pages. Cut and paste for text manipulation, hyphenation help, and command "undo" are also included.

The spelling checker was omitted from the package I received for review. Although Latitude's documentation has references to the spelling checker, it was not implemented in the AT&T version of Horizon's program.

On-line help is available through a set of help files—Horizon calls them the "Explain" feature. You can invoke this feature by typing an `E` as a command response. Explain covers 43 different subjects. When you call it up, a main Explain menu lists each subject, along with a number to type to bring up a particular screen of information. After you have read the information, pressing "Return" will restore your text; if you type `m`, the main Explain menu reappears.

Latitude also offers Memorize, a feature that allows you to commit a command, a series of commands, or even a portion of text to memory for later recall with a single keystroke. Two different Memorize sequences are available, and each can contain up to 100 keystrokes. This feature is useful for storing long and complex command strings.

Latitude's ABBREV function allows you to create a library of abbreviations. Using this feature, you can generate files that contain unabbreviated forms of words or lines of text that you frequently use. Because abbreviation files allow you to create "unabbreviations" of up to several hundred characters, they can be a real timesaver.

Horizon has made provisions for "boilerplating" text additions by use of "resource" files. While you are working with your text file, you can also call up a resource file and extract text from it, placing it into the text you are currently working on. Although only one resource file can be open at a time, closing it and calling up a different one is simple. A resource file can also contain tabular data, allowing you to produce boilerplate tables as well as regular text.

SPREADSHEET FUNCTIONS

The spreadsheet portion, the Latitude Tabulator (or, as Horizon calls it, the "Table Editor"), is designed for creating and manipulating tabular data. You invoke this feature by typing a `t` from the command mode of the word-processing system. The word-processing screen then disappears and is replaced by the Tabulator screen. This screen has a command section at the top, a center area with a cell array of 16 rows by 8 columns, and a command prompt line at the bottom of the display.

Maximum table size is 100 rows by 20 columns, and each cell can contain up to 20 characters. You can place a maximum of 64K bytes in a table. Although this figure is small compared to some spreadsheet programs, Latitude does allow you to have multiple tables within your text.

Available for use with the Table Editor are basic arithmetic functions such as add, subtract, multiply, and divide. Functions such as average, days between dates, minimum and maximum values, and a few logical functions are also present.

The major difference between the Table Editor and a conventional spreadsheet is the lack of more complex mathematical functions, especially the transcendentals. You cannot, for example, compute formulas such as loan payments, which require both exponential and log functions.

Horizon has designed Latitude for users who prepare documents containing both text and tabular (spreadsheet) material.

Like the word-processing portion of Latitude, the spreadsheet also has an Explain feature, which in this case covers 24 different subjects. One major difference between

the word-processing part and the Tabulator is that you cannot move spreadsheets using cut and paste. In order to reposition a spreadsheet, you must move the text around the table.

The Tabulator keeps a running record of how much space is still available for your table. Documentation for the spreadsheet indicates that if you allow the table to fill beyond the 90 percent point (about 57,000 characters), then the Table Editor will malfunction, causing a loss of data. The only warning you have is the display of remaining space. A program running on a computer such as the 3B2 should have a more sophisticated method to prevent data loss.

OTHER FEATURES AND DOCUMENTATION

An integral part of Latitude is its Manager program, which contains two configuration setups. The first allows you to set up automatic program startup for users so that once a user has logged on to the system, Latitude will be loaded.

The second part of the Manager program deals with printer configuration. Latitude supports an extensive number of printers, both dot matrix and daisy wheel. It can configure the system to send output to the printer as single sheet, continuous feed, or into the system's printer spooling function.

Latitude's Enhanced Printer Control (EPC) codes are defined in an EPC code index file. This file allows for special printing features such as boldface, double-strike, superscript, subscript, and so on. Of course, the features supported depend on the capability of your printer. Most terminals can't display special formatting features, such as italics, but seeing a combination of characters before and after a word reminds you that the word will be different when it is printed. Latitude is capable of using most of the special features found on today's printers.

Documentation consists of a single 8-by-9-inch three-ring binder divided into sections. Although the documentation covers essentials and is adequate to get the program running, it is bare bones in many respects. Because Latitude incorporates such extensive on-line help files, you probably won't need to refer to the manual once you've set up the program for the first time. However, I would have appreciated a command summary card. For a small manual, though, Latitude has an extensive, well-done index.

On-line help is available through a set of help files, which Horizon calls the Explain feature.

INSTALLATION AND SUPPORT

Installing the program does not present any problems. The installation section of the documentation is clear and contains a printout of what you should expect to see on the terminal screen during loading.

Installation does require that you be logged on as the superuser (root). This is also the case when you set up auto-log-in features or printer-code selections using the Manager program.

During installation, you can use the `crontab` file to set a time for the removal of files in the "wastebasket" directory and to maintain a spelling history file (if your version of Latitude includes the spelling checker `spell`). When you delete a file from within the program, Latitude places it in the user's wastebasket directory. This means that you can undelete a file as long as you do it before the automatic removal occurs.

Having placed a couple of calls to the technical support group at Horizon, I feel that the staff members truly understand their product and many of the systems on which it is designed to operate.

Latitude is an easy program to learn, and its many menus make it easy to find the command you are looking for. Although display of text cannot keep up with the keystrokes from even a moderately fast (35-words-per-minute) typist, no keystrokes are lost. The Table Editor, although not a powerful spreadsheet, is a handy addition to the word processor. □

Harry Avant evaluates microcomputer hardware and software at the Jet Propulsion Laboratory (Pasadena, Calif.). This article marks his fourth contribution to UNIX/WORLD *Magazine.*

HORIZON RESPONDS

The problem Mr. Avant refers to has been corrected; the tabulator will no longer lose data when overfilled.

Paul Miller
Director of Planning

Editor's Note: *As we were going to press, Horizon officials informed us that the company had changed its name to Laticorp, apparently to emphasize marketing of the Latitude applications program.*

NEW PRODUCTS

ADDITIONAL SOFTWARE AVAILABLE FOR HP's INTEGRAL

Fifteen more software packages are available for The Integral, Hewlett-Packard's portable Unix system-based personal computer.

Besides the 37 previously announced packages, software for The Integral now includes C-ISAM and File-it databases from RDS, the HP-UX-to-MS-DOS Interchange communications package from Oswego Software, Sofgram from Softest, VT/E from P2/i, and LEX Word Processing from HP.

Programming languages include HP's HP-UX BASIC and HP-UX C, STSC's APL, Philon's FAST/BASIC-M and FAST/COBOL, and Oswego Software's MBASIC-HP-UX Basic Xlator.

Engineering software includes Pac Soft's Engineering Survey and Kelix's Flow Network Analysis. Also available is Investor II, a vertical applications package from REMS.

For more information, contact Hewlett-Packard, 3000 Hanover St., Palo Alto, CA 94304; 415/857-1501.

Please circle Reader Service Number 160.

TOM RELEASES MEDICAL GROUP/CLINIC SYSTEM

TOM Software has released the Medical Group/Clinic Management Information System, an integrated package designed for the record-keeping needs of specialty group practices, clinics, and problem-oriented health care facilities.

The modular system produces superbill statements, follow-up statements, and insurance forms for patient charges. It also tracks practice production by physician, procedure type, revenue center, and insurance carrier.

TOM Software has also released the Attorney Business Management System, a six-module integrated business management package that addresses the time-keeping, billing, and accounting needs of small- to medium-sized law firms. It has been ported to a variety of PCs, micros, minis, and small mainframes.

The Legal Time and Billing module, designed for daily use, tracks each attorney's time, expenses, and descriptions of the services performed.

For more information, contact TOM Software, P.O. Box 66596, 127 S.W. 156th, Seattle, WA 98166; 206/246-7022.

Please circle Reader Service Number 161.

RABBIT SOFTWARE INTRODUCES WINDOWS-PLUS

Windows-Plus, which allows users to open up to 14 windows on any asynchronous terminal, is the newest addition to Rabbit Software's Middleware product line. Any Unix system-based application can run in a Windows-Plus window without modification.

Windows-Plus users can conduct multiple sessions or applications at the same time from a single terminal, change window sizes, open new windows, close existing windows, or overlap, enlarge, or diminish any open window.

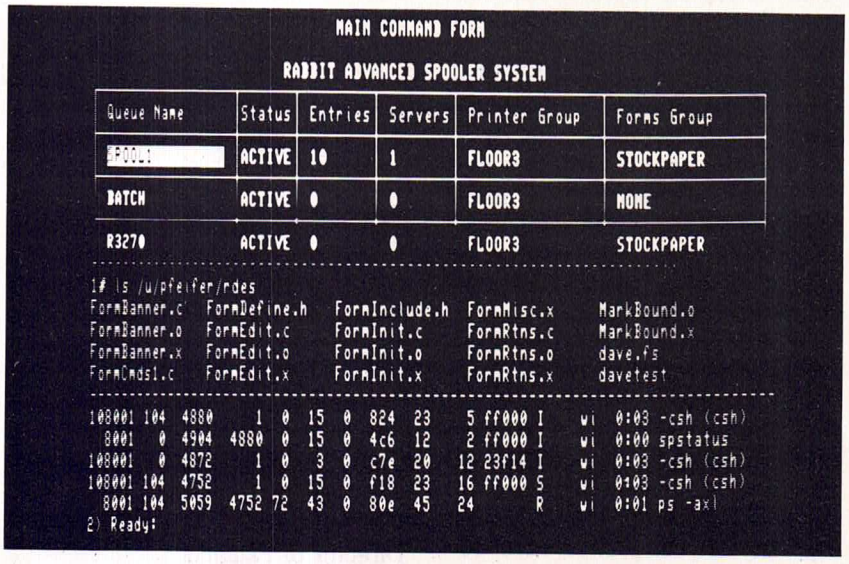
For more information, contact Rabbit Software Corp., One Great Valley Parkway East, Malvern, PA 19355; 215/647-0440.

Please circle Reader Service Number 162.

NEW BASIC VERSION OF ANALOG WORKBENCH

A basic version of the Analog Workbench, intended primarily for users who want to develop their own device libraries and specialized support programs, is now available from Analog Design Tools Inc.

The software consists of a windowing and menu-driven user inter-



Rabbit Software's Windows-Plus software

NAME THE MOST WIDELY USED INTEGRATED OFFICE AUTOMATION SOFTWARE FOR UNIX™ SYSTEMS.

"UNIPLEX II"™

YOU'VE GOT IT!

User satisfaction is the primary reason no other product can make this claim. Already in its second generation, UNIPLEX II offers features designed to meet the requirements of the most demanding user.

The beauty of UNIPLEX II is its simplicity. One personality and one command structure throughout the program provide an ease of use never before experienced with UNIX application software.

UNIPLEX II integrates sophisticated word processing, spreadsheet, and relational database applications into a powerful one-product solution.

UNIPLEX II uses termcap, so it can run on virtually any computer terminal. "Softkeys" allow the user to define function keys which are displayed on the 25th line of most terminals to provide versatility and ease of use.

All this at a price you'd normally pay for a single application software package.

UNIPLEX II is available immediately from UniPress Software, the company that's been at the forefront of quality UNIX software products longer than anyone else.

OEM terms available. Mastercard and Visa accepted.

Call Today! Once you've got it, you'll see why UNIPLEX II is the most widely used integrated office automation software for UNIX-based systems.

Write to: UniPress Software, 2025 Lincoln Hwy., Edison, NJ 08817 or call: **1-800-222-0550 (outside NJ) or 201-985-8000 (in NJ)**; Telex: 709418. Japanese Distributor: Softec, Telephone: 0480 (85) 6565. Swiss Distributor: Modulator SA, Telephone: (031) 59 22 22.

UniPress Software

Your Leading Source for UNIX™ Software.

Please circle Ad No. 77 on inquiry card.

face, circuit drawing and editing, a basic library of device models, and test-simulation programs.

Using the electronic mouse pointing device and a few simple keyboard entries, the designer can construct a circuit, symbolically attach test instruments such as function generators and oscilloscopes, and perform tests. All functions are executed in the workstation computer and are displayed on the multiwindow display screen.

For more information, contact Analog Design Tools, 800 Menlo Ave., Suite 200, Menlo Park, CA 94025; 415/328-0780.

Please circle Reader Service Number 163.

INSPIRATION SYSTEMS PRESENTS INTEGRATED UNIX-BASED SOFTWARE

Inspiration Systems Inc. has unveiled Prevail, a Unix system-based family of software products. The three Prevail product groups include office products—integrated word processing, spreadsheet, database management, and communications modules; database products—fourth-generation database definition and report writer/application generation tools; and applications development products—a Pascal-based application development language.

Hardware currently supported by Prevail includes the DEC VAX series, Sun Microsystems, NCR Tower, and the 3B series computers from AT&T.

Group prices are based on hardware types, which range from supermicro to mini and mainframe computers. OEM and end-user volume discounts are available. Also, individual Prevail components can be assembled in any other combination to suit specific application needs.

For more information, contact Inspiration Systems Inc., Production Plaza, Sewickley, PA 15143; 412/771-4000.

Please circle Reader Service Number 164.

PRICE CUTS FROM APOLLO

Apollo Computer Inc. has cut the prices on its 32-bit DN300/DN320 workstations and related 34- and 70-Mbyte disk subsystems.

The cost of the DN300, with high-resolution graphics, virtual memory, system software, 1 Mbyte of main memory, and a Domain network interface has been reduced by 38 percent to $9900. The price of the DN320, with floating-point hardware, has been reduced by 21 percent to $18,900.

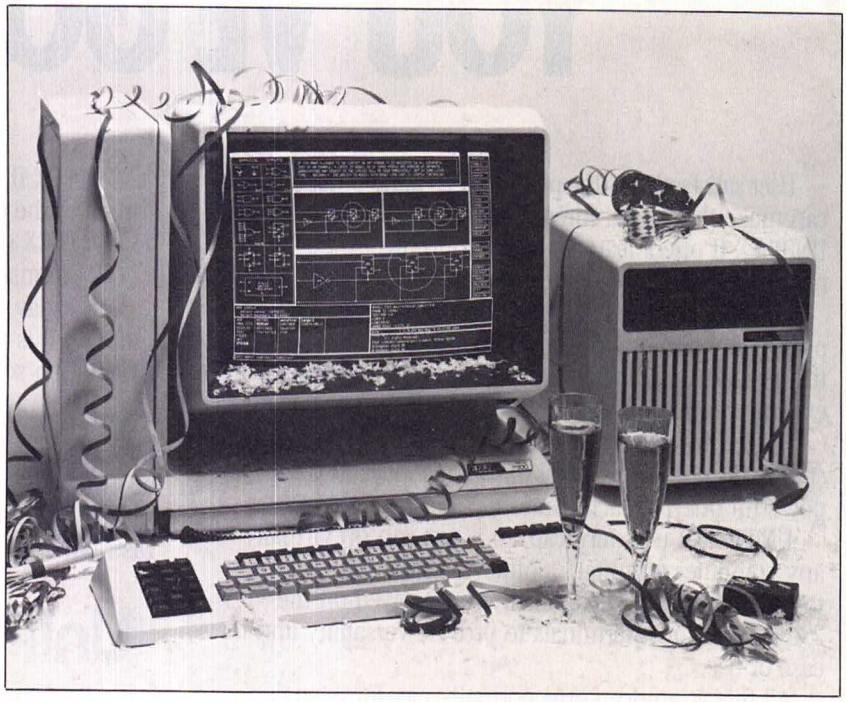
For more information, contact Apollo Computer Inc., 330 Billerica Rd., Chelmsford, MA 01824; 617/256-6600.

Please circle Reader Service Number 165.

LOGICAL ANNOUNCES SOFTSHELL USER INTERFACE

Logical Software Inc. has introduced SoftShell, a full-screen user interface offering an intelligent approach to the Unix system, on a number of multiuser machines. SoftShell is currently available on the following machines: the IBM PC/XT under PC/IX, the PC/AT under Xenix, the DEC Pro-350 under Venix, and the NEC APC III under PC-UX.

SoftShell consists of three programs. Yelp provides structured help information and templates for



Apollo's DN300 and DN320 workstations

NEW RELEASE

UNIPRESS EMACS™

VERSION 2

Another in a series of productivity notes on UNIX™ software from UniPress.

Subject: Multi-window, full screen editor.

Multi-window, full screen editor provides extraordinary text editing. Several files can be edited simultaneously, giving far greater programming productivity than vi. The built-in MLISP programming language provides great extensibility to the editor.

New Features:

- *EMACS is now smaller and faster.*
- *Sun windows with fonts and mouse control are now provided.*
- *Extensive on-line help for all commands.*
- *Overstrike mode option to complement insert mode.*
- *New arithmetic functions and user definable variables.*
- *New manual set, both tutorial and MLISP guide.*
- *Better terminal support, including the option of not using unneeded terminal drivers.*
- *EMACS automatically uses terminal's function and arrow keys from termcap and now handles terminals which use xon/xoff control.*
- *More emulation—TOPS20 for compatibility with other EMACS versions, EDT and simple WordStar™ emulation.*

Features:

- *Multi-window, full screen editor for a wide range of UNIX, VMS™ and MS-DOS™ machines.*
- *"Shell windows" are supported, allowing command execution at anytime during an edit session.*
- *MLISP™ programming language offers extensibility for making custom editor commands! Keyboard and named macros, too.*
- *"Key bindings" give full freedom for defining keys.*
- *Programming aids for C, Pascal and MLISP: EMACS checks for balanced parenthesis and braces, automatically indents and reformats code as needed. C mode produces template of control flow, in three different C styles.*
- *Available for the VAX™ (UNIX and VMS), a wide range of 68000 machines, IBM-PC,™ Rainbow™ 100+, and many more.*

Price:

	Binary	Source
VAX/UNIX		$995
VAX/VMS	$2500	7000
68000/UNIX	395	995
MS-DOS	475	*

*Call for terms

For more information on these and other UNIX software products, call or write:
UniPress Software, Inc.,
2025 Lincoln Hwy.,
Edison, NJ 08817.
Telephone: (201) 985-8000.
Order Desk: (800) 222-0550.
(Outside NJ). Telex: 709418.
Japanese Distributor:
Softec 0480 (85) 6565.
European Distributor:
Modulator SA (031) 59 22 22.

OEM terms available.
Mastercard/Visa accepted.

UniPress Software

Your Leading Source for UNIX Software

Please circle Ad No. 27 on inquiry card.

Unix system commands. LSLS, SoftShell's directory walk program, clarifies the Unix file system by classifying file types and allowing users to annotate directories. The LShell, a full-screen scroll handler, permits users to review a terminal session, editing and re-executing commands as desired.

License fees for binary copies of SoftShell range from $295 to $995, depending on the CPU. OEM and distributor agreements are available.

For more information, contact Logical Software, 17 Mt. Auburn St., Cambridge, MA 02138; 617/864-0137.

Please circle Reader Service Number 166.

HARRIS ANNOUNCES ENGINEERING WORKSTATIONS

A new workstation product line has been introduced by the computer systems division of Harris Corp. The Unix system-based 32-bit Harris Station 10 and Harris Station 20 (HS-10 and HS-20, respectively) feature system architecture for use as either stand-alone workstations or as part of a Harris Station and Harris superminicomputer network.

Both the HS-10 and HS-20 can be linked to Harris superminicomputers via RS-232 serial lines or through Ethernet local-area networking. This capability enables Harris superminicomputers to operate as either compute servers, off-loading computationally intensive tasks from workstations, or as file servers, storing and quickly accessing large files and databases.

Applications software packages include software for engineering, computer-aided design, drafting, graphics, and administration. Application support libraries are callable from C, FORTRAN-77, and Pascal.

Harris' Unix operating system is compatible with AT&T Unix System V, enhanced with real-time and Berkeley extensions.

Prices for the Harris Station 10 start at $32,500; the Harris Station 20 starts at $39,900.

For more information, contact Harris Corp., Computer Systems Division, 2101 W. Cypress Creek Rd., Ft. Lauderdale, FL 33309; 305/973-5125.

Please circle Reader Service Number 167.

HEURIKON OFFERS UNIX DEVELOPMENT WORKSTATION

Heurikon Corp. has introduced the Minibox, a multiuser workstation based on the M68000/68010 microprocessor. The Minibox has a built-in C compiler, six Multibus card slots, single or dual 5¼-inch floppy-disk drives, 30 to 280 Mbytes of Winchester hard-disk storage, and interactive streamer tape drive.

Heurikon's Minibox workstation

The workstation is built around the Heurikon HK68 microcomputer, providing CPU (8 or 10 MHz), quad channel DMA, Winchester interface, streamer tape drive (or definable dip switches), 64K bytes EPROM, 512K to 1 Mbyte RAM, two iSBX expansion connectors, MMU (addresses 16 Mbytes of RAM), four serial I/O ports (expandable to 12), three 16-bit counter timers, and two parallel I/O ports.

The HK68 supports Unix System III and V, with drivers for Ethernet, SDLC/HDLC, and floating-point processors.

For more information, contact Heurikon Corp., 3201 Latham Dr., Madison, WI 53713; 800/356-9602.

Please circle Reader Service Number 168.

COMPUFLEX SUPERMICRO

Compuflex Inc. has introduced the Compuflex 68/10 supermicrocomputer, which provides multitasking capability for up to 32 users. Designed for the OEM and end-user with software development capability, the Compuflex 68/10 has an M68010 16/32-bit microprocessor, a VMEbus, and the Unix System V operating system.

Ten expansion slots are available on the VMEbus. The mass storage devices include a hard disk with up to 100 Mbytes of storage, a 5¼-inch floppy disk drive with up to 1 Mbyte of storage, and a ¼-inch streaming tape with 45 Mbytes of backup.

Users can program in a variety of popular languages such as C, FORTRAN-77, BASIC, COBOL, and Pascal.

The Compuflex 68/10 is available directly from the manufacturer for $19,995. Quantity discounts, special configurations, and interfaces are available.

For more information, contact Compuflex Inc., 2601 E. Chapman Ave., Suite 104, Fullerton, CA 92631; 714/680-4024.

Please circle Reader Service Number 169.

PAM COMPUTER SYSTEMS RELEASES NEW PRODUCT LINE

The Series 3200 product line from PAM Computer Systems incorporates Flex Rx, a prescription pathway designed for user ease and efficiency.

The heart of the system is a 32-bit CPU configured with a minimum of 640K bytes of RAM, but the system can be upgraded to 1 or 2 Mbytes for multiuser applications.

The software for the Series 3200 is written in C, utilizing the Unix system and the Unify Relational Database Management System. Also included with the system are the Horizon word-processing and spreadsheet software packages.

The company offers pre-installation data loading; a 90-day equipment warranty; 30 hours of in-store applications training; and a set of applications manuals with each system.

For more information, contact PAM Computer Systems, 1303 W. Beltline, Suite 201, Carrollton, TX 75006; 214/323-1414.

Please circle Reader Service Number 170.

INTEGRATED SPREADSHEET FOR AT&T 3B SERIES FROM ACCESS TECHNOLOGY

Access Technology Inc. is now shipping 20/20, an integrated spreadsheet combining graphics, database management, project modeling, and

advanced spreadsheet functions, for AT&T's 3B2 and 3B5 computers.

20/20's built-in functions include IRR and NPV, natural recalculation, cell naming, multidimensional consolidation routines, goal-seeking tools, and advanced command file operations. Integrated with the spreadsheet are on-screen graphics, database commands, and a project modeling facility.

20/20 costs $950 for a 3B2 version and $3800 for a 3B5 version, both including documentation, on-line help, self-paced tutorials, and hotline services. 20/20 is available directly from Access Technology or from designated VARs and distributors.

For more information, contact Access Technology, Six Pleasant St., South Natick, MA 01760; 617/655-9191.

Please circle Reader Service Number 171.

OPUS SYSTEMS DEBUTS WITH UNIX CO-PROCESSOR

Opus Systems has introduced the Opus516 Personal Mainframe, a Unix/co-processor subsystem designed to convert an IBM PC to a 32-bit Unix system workstation.

Opus516 consists of a complete port of AT&T Unix System V (Release 2.0) and a 32-bit co-processor for the IBM PC and plug-compatible computers. The product is marketed primarily to OEMs and systems integrators.

Opus516 allows users running software programs on much larger machines to bring them to the desktop level, and it provides 32-bit computing power without sacrificing the standard PC applications users have grown accustomed to. A simple keyboard command allows easy transfer between Unix and DOS, a spokesman said.

Opus516 is based on National Semiconductor's 32016 CPU and includes the 32082 memory management unit and 32081 floating-point unit. Included with Unix System V and its utilities are C and FORTRAN-77 compilers, an assembler, and a debugger.

In OEM quantities, a 1-Mbyte configuration is priced at $3140.

For more information, contact Opus Systems, Suite 120, 960 San Antonio Rd., Suite 120, Los Altos, CA 94022; 415/941-7201.

Please circle Reader Service Number 172.

VIRTUAL MEMORY MICROCOMPUTER FROM SGS

The systems division of SGS Semiconductor Corp. has introduced Samson, a Unix system-based, 16-bit, demand-paged virtual-memory microcomputer, supporting from 10 to 34 ports.

Samson is Multibus compatible, and includes 85 Mbytes of integral disk storage and 67 Mbytes of formatted tape-cartridge capacity. Samson's dual-bus architecture incorporates distributed 16-bit, intelligent I/O boards, enabling software drivers to be executed remotely.

Samson includes the full AT&T Bell Labs System III, with enhancements from SGS and UC Berkeley.

A system for 10 ports is configured with a Multibus, 10 card-slot chassis, 512K bytes of on-board memory, 85 Mbytes integral disk storage, and a 67-Mbyte magnetic tape cartridge. It is priced at $24,900.

For more information, contact SGS USA, 1000 E. Bell Rd., Phoenix, AZ 85022; 602/867-6100.

Please circle Reader Service Number 173.

GENERAL COMMUNICATIONS ANNOUNCES A MULTIUSER SUPERMICRO

General Communications Corp. has introduced the L5, a family of multiuser supermicrocomputers for program development and scientific, industrial, and commercial applications.

The L5 comes with one of two Unix system kernels—the Unix

SGS Systems' Samson virtual memory microcomputer

System V fast kernel or a Unix real-time kernel for data acquisition. Depending on the configuration, the L5 supports 4 to 32 users, has ½ Mbyte to 32 Mbytes of internal memory, and can access from 20 Mbytes to 2½ gigabytes of external storage.

Eventually, HSDX, a wide-area network solution, will provide the L5 with features required by Integrated Services Digital Networks and will eliminate the need for high-speed modems with a TI carrier interface.

A cost under $1000 per user for a mid-range system includes all cabling, complete documentation, and tutorials, as well as several software utilities, including typesetting software for the Hewlett-Packard LaserJet printer.

For more information, contact General Communications Corp., One Main St., Suite 502, Eatontown, NJ 07724; 201/542-6560.

Please circle Reader Service Number 174.

PYRAMID INTRODUCES MULTIPROCESSOR SUPERMINI

Pyramid Technology Corp., manufacturer of the 90x superminicomputer, has introduced a new Unix system-based multiprocessor superminicomputer.

Like the original 90x, the 90Mx is a 32-bit, virtual memory, RISC (Reduced Instruction Set Computer) architecture superminicomputer.

The new system includes up to 16 intelligent terminal processor ports for a maximum of 256 user connections, and slots for eight memory boards, for a maximum of 32 Mbytes of main memory.

The systems range in price from $220,000 to $420,000, with the entry-level 90Mx consisting of OSx, a 415-Mbyte disk, 1600 bpi magnetic

tape drive, black-and-white console, 16 user ports, and 4 Mbytes of main memory.

For more information, contact Pyramid Technology, 1295 Charleston Rd., Mountain View, CA 94043; 415/322-7200.

Please circle Reader Service Number 175.

CYB's NEW MULTIUSER SYSTEMS

Cyb Systems Inc. has announced the addition of two new computers—the WorkMate I and WorkMate II—to its WorkSeries supermicrocomputer product line. The WorkMate I is sold as either a single-user workstation or as a four-user system; the WorkMate II accommodates up to eight users.

The new systems utilize the M68000 microprocessor, Multibus architecture, and industry-standard interfaces. Unix System V is standard on all Cyb computers, with the RM/COS operating system offered as an option.

The four-user WorkMate I consists of 512K bytes of RAM (expandable to 1¼ Mbytes), a 26-Mbyte 5¼-inch Winchester disk drive, a 1-Mbyte floppy-disk drive, 6 serial I/O ports, and Unix System V. A variety of disk options and a floppy tape backup are available for the WorkMate II. Ethernet is optional.

Prices range from under $8000 for a single-user, RM/COS-based WorkMate I, to $15,999 for an eight-user, Unix system-based WorkMate II.

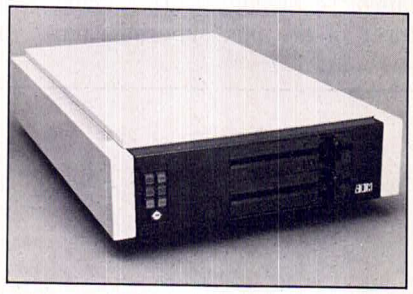
For more information, contact Cyb Systems Inc., 6488 Hwy. 290 East, D-111, Austin, TX 78723; 512/458-3224.

Please circle Reader Service Number 176.

11/73-BASED MICRO/11 AND MICRO/73 SYSTEMS FROM MDB

MDB Systems Inc., a manufacturer of DEC-compatible systems and interface products, has introduced an 11/73-based system.

The MDB Micro/11-73 contains a 20.8-Mbyte Winchester disk subsystem that is RLV22(RL02) software compatible, and a dual 8-inch floppy, 1-Mbyte diskette subsystem that is RX02 media compatible.



MDB's 25-Mbyte SMD controller

The Micro/11-73 systems are supplied with a choice of MDB's dual-size 256K-byte or 512K-byte RAM memory modules or a quad-size 1 Mbyte RAM module.

The basic systems are priced at $13,150 (with 256K-byte RAM), $13,875 (with 512K-bytes), and $15,050 (with 1 Mbyte) in single quantity with OEM discounts.

For more information, contact MDB Systems Inc., 1995 N. Batavia St., Orange, CA 92267; 714/998-6900.

Please circle Reader Service Number 177.

NEW MASSCOMP FORTRAN

Masscomp's newest FORTRAN, Version 3.1, includes a number of support tools for program development, such as a symbolic debugger, new access to the window manager, new access to the Unix system profiler, as well as increased software support for Masscomp's array processor.

The new FORTRAN release includes 100 percent ANSI FORTRAN-77 validation from the Federal Software Testing Center. The interactive multilanguage symbolic debugger, known as `mdb`, lets users correct a program at the level they write it, making it easier to develop applications on Masscomp's system in FORTRAN and C.

Prices range from $25,900 to $150,000, depending on disk, tape, and general configurations.

For more information, contact Masscomp, One Technology Park, Westford, MA 01886; 617/692-6200.

Please circle Reader Service Number 178.

REVISED C COMPILER FROM INFORMATION PROCESSING TECHNIQUES

Information Processing Techniques Corp. has announced a new revision of its C compiler for Data General computers. DG C, a full implementation of Bell Labs' C programming language, is faithful to the Kernighan and Ritchie standard. According to the company, the compiler will run under any mapped Data General operating system.

The primary enhancements to Revision 2.25 of DG C are the addition of an M68000 cross-compiler, 32-bit AOS/VS I/O packet support, and more optimization in generated code size and speed. Debugging features have been added, as have several preprocessor facilities.

Prices range from $5000 to $26,000, depending on configuration and the cross-compilers selected.

For more information, contact

Information Processing Techniques Corp., 1096 East Meadow Circle, Palo Alto, CA 94304; 415/494-7500.

Please circle Reader Service Number 179.

SUPERIOR SOFTWARE RELEASES MODEM/ COMMUNICATION PROGRAM

Superior Software Corp. has introduced Superior MOD/COM, a modem/communication program written in the C programming language, for Unix and Xenix systems.

MOD/COM can send and receive files with error checking/recovery and is compatible with the Christensen XMODEM protocol, allowing transfers to and from many CP/M and MS-DOS systems.

Baud rates and port can be easily changed, and on-screen help facilities are provided. One command line option (-?) will display an overview of the program, command line options, and examples.

License fees are $150 for the first year and $99 for each additional year, which includes any updates available upon renewal. A license for continual use costs $495.

For more information, contact Superior Software Corp., 202 13th St., Suite 206, Augusta, GA 30901; 404/722-0831.

Please circle Reader Service Number 180.

APPLICATION GENERATOR FROM SMI

Software Manufacturers Inc. (SMI) has introduced C-Link, a C application generation system that produces programs for execution under Unix operating systems.

C-Link enables a programmer

to develop programs in the C language by writing in BASIC. It also serves as a teaching aid for programmers learning C and helps experienced C programmers to increase their output.

C-Link accepts programs through keyboard entry or transfer from other systems, producing functionally equivalent programs in C. The resulting C programs are then compiled and linked to form an application system that executes under Unix or Xenix, including version 3.0.

For more information, contact Software Manufacturers Inc., 20720 S. Leapwood Ave., Carson, CA 90746; 213/538-8174.

Please circle Reader Service Number 181.

C CROSS-COMPILER FROM JMI SOFTWARE

JMI Software Consultants has delivered the first C cross-compiler for National Semiconductor's Series 32000 to be hosted on the IBM 370, the company said.

The cross-compiler package includes both a code generator for the Series 32000 (32008, 32016, 32032) and a cross-assembler that produces complete assembly listings, including addresses, object code, and assembler source code. In addition, a cross-linker is provided that facilitates the construction of RAM/ROM combination systems.

The compiler can also produce assembler source code compatible with National Semiconductor's assembler and linker on SYS32/Genix. Floating point support is provided in conjunction with the NS32081 floating point unit.

For more information, contact JMI Software Consultants, P.O. Box 481, Spring House, PA 19477; 215/628-0846.

Please circle Reader Service Number 182.

RTUX EXECUTIVE FROM EMERGE SYSTEMS

Emerge Systems has introduced the RTUX Executive, a real-time support package for use with an unmodified Motorola System V/68 operating system, a derivative of AT&T's Unix System V.

The RTUX package provides support for resident systems mode exception handlers and scheduled real-time processes in single or distributed real-time systems. Interaction between resident programs and scheduled processes is provided by a comprehensive set of RTUX executive service calls and shared memory.

The RTUX Executive provides a real-time enhancement to the general-purpose, interactive, time-sharing capabilities of the System V/68 operating system. No modifications to the system are required to support RTUX. All scheduled real-time processes and resident real-time programs may be written in C. Customizing and programming services are also available.

For more information, contact Emerge Systems, 114 6th Ave., Indialantic, FL 32903; 305/723-0444.

Please circle Reader Service Number 183.

dBASE II-TO-cENGLISH CONVERTER FROM C LINE

C Line Inc. has released its dBase II-to-cEnglish converter, as well as the cEnglish 2.3 version. The converter transforms existing dBase II computer programs into cEnglish, a fourth-generation language with syntax similar to that of dBase II.

More than 400,000 dBase II users can now take advantage of the power of the C programming language through cEnglish without having to program in C, the company said.

cEnglish supports multidimensional arrays and procedure function calls, and it allows escape into direct C programming. The 2.30 release has a uniform user interface to a number of popular database managers.

For more information, contact John Grey, C Line Inc., 20 W. Ontario, Chicago, IL 60610; 312/944-4510.

Please circle Reader Service Number 184.

PRODUCTIVITY APPLICATIONS FOR XENIX 3.0/286 SYSTEMS

Handle Technologies' Office Automation series is now available for machines running Xenix 3.0 and based upon the Intel 80286 microprocessor.

The Office Automation software includes a word processor, spreadsheet, business graphics, and list processing and management, and an 80,000-word spelling checker.

All applications operate in a multiuser or networked configuration. For security, each user operates with a function-key driven, common-user interface that insulates Xenix from the user.

For more information, contact Handle Technologies at 916/581-5227. □

Please circle Reader Service Number 185.

UNIX/WORLD's "New Products" section is provided as a service to our readers. Our selection criteria are based solely on the needs of our readership. If you would like to have your product news considered for publication, please address your correspondence to UNIX/WORLD Magazine, New Products Editor, 444 Castro St., Suite 1220, Mountain View, CA 94041-2017.

THE BOOK THAT STARTED IT ALL

The C Programming Language,
by Brian Kernighan and
Dennis Ritchie
PUBLISHED BY PRENTICE-HALL,
228 PAGES, $22.95

REVIEWED BY RAY SWARTZ

Synergism describes the magic that results when a whole becomes more than the sum of its individual parts. *The C Programming Language,* by Brian Kernighan and Dennis Ritchie, is a book whose synergism has affected the entire world of computers. This book is now so well known and widely discussed that it is simply referred to as "K and R" or, in some circles, "The White Book."

Before the arrival of the C language, the desire to transport system software from one machine to another was largely an unfulfilled dream. Kernighan and Ritchie showed that portability could be designed directly into a language. The ideas embodied in the book have led C to become one of the most popular programming languages today. In addition, even the book's style has served as a model for other language books.

For some C enthusiasts, reviewing the Kernighan and Ritchie book might be likened to discussing the literary merit of the Bible. Yet the demand for C books is growing. It is estimated that over 5000 copies of *The C Programming Language* are sold each month. Thus, its popularity suggests that readers continue to be interested in it. There is also a second reason to review this book: My own sense of completeness tells me that the first C book I review in this column should be the first C book written.

This book is not for first-time programmers; it is written for those already familiar with computers and at least one other structured higher-level programming language, such as Pascal or FORTRAN 77. Yet it is more than simply a description of the C language syntax. The design philosophy and underlying principles of the language are covered as well.

This book is most suited for learning C in the Unix system environment. The authors were instrumental in the development of both C and the Unix operating system, and this book provides a good discussion of the C/Unix system interface. However, most of the book is still relevant even if you aren't using C in a Unix system environment.

HOW THIS BOOK IS WRITTEN

It is often said that the C language is terse. Perhaps, C has this reputation because of the way the K and R book is written. Its compact style can make for slow reading, and some sections will need to be reread. In fact, it is unlikely that anyone can absorb in just one reading all that *The C Programming Language* contains. I constantly discover "new" sentences in it!

Although the authors use words sparingly, they do make liberal use of examples. Complete programs are used to illustrate virtually every point in the book, even minor ones. In several cases, the authors develop an entire application to serve as an extended example of language usage.

Unfortunately, in an attempt to provide nontrivial examples, Kernighan and Ritchie have chosen some obscure ones. For example, they spend eight pages creating a "desk calculator" program that uses Reverse Polish Notation (RPN). Although this section serves as a useful illustration of external variables and their scope, it assumes that the reader understands the nuances of RPN and stack operations such as `push` and `pop`.

On the other hand, the authors have included some valuable examples applicable to everyday problems. For example, they show a shell sort program and a pattern-matching utility that is like the Unix system `grep` command. In addition, several examples contain source code from many of the standard C library functions and from some Unix system commands.

Kernighan and Ritchie include many exercises throughout each chapter for the reader; unfortunately, the authors don't provide answers to them. This can be especially frustrating when the answers aren't obvious (few of them are). However, these exercises do show what the authors believe to be the important points to ponder and work on. [*Editor's Note: Prentice-Hall is publishing a book that contains answers to the K and R exercises.*]

REVIEW HIGHLIGHTS

- Not recommended for beginning programmers.
- Maximum recommendation for all C programmers, especially thouse who program in the Unix system environment.

Continued on page 114

SPRING-TUNING YOUR KERNEL

A little fine-tuning can increase your Unix system's performance. In the first of two parts, our author piques your interest.

BY RIK FARROW

PART ONE

The idea of changing the Unix system kernel strikes fear into the hearts of many system administrators and programmers. But the kernel is the heart of the Unix operating system; it manages system resources, controlling their access by programs. And because the kernel also keeps the information used to manage these resources in tables that are fixed in size, changing the size of these tables can increase (or decrease) the performance of your Unix system.

Your system's kernel can be a fascinating subject upon closer examination. In this two-part series, we will be putting the Unix system kernel under a magnifying glass. In part 1 we'll discuss how the Unix system kernel works, the limits configured into your kernel, drawbacks your particular kernel might have, and reasons for changing, or tuning, your kernel. In part 2 we'll take those of you ready to make the plunge through the steps needed to tune your kernel.

Many companies that distribute the Unix operating system develop their kernels on a large machine, usually a VAX minicomputer, and then download (copy) it to the target machine. Other than trying the kernel out to see if it works, these firms may never *use* the new kernel! They make their money by providing a working Unix system kernel, not necessarily one that is tuned optimally to the particular target machine hardware.

Often, the kernel that's distributed is a "one size fits all" variety, which makes life easiest for the Unix system software vendor. Manufacturers and original equipment manufacturers (OEMs) always have an entry level for their Unix systems: a machine with the minimum memory and disk space needed to run the Unix operating system. The kernel distributed with these systems has been designed so that it works with the minimal system. Thus, it doesn't matter how much memory or disk space you have —you get the same kernel that goes to the minimal machine.

For example, when you get the Unix system for a DEC minicomputer, it comes as a minimal system, *but* it is configurable. Companies that can afford minicomputers will probably also spend the money to have the kernel installed correctly by someone with the expertise.

Certainly, the limits that are built into a "one size fits all" kernel are based on an "ideal world." This

ideal world reflects, at worst, the hardware developer's research and development environment; at best, it is an average of what generally works best. Even after a kernel has been modified to match its hardware and expected load, it still has to live in the "real world."

Thus, even after the vendor has modified the kernel for your particular hardware configuration, you, the purchaser, have to live with what you get: a kernel that has never been used in your particular real-life environment.

WHAT CAN YOU DO?

What you can do depends on many factors. If you are a manufacturer or an OEM, you have control over these configuration limits. You can tweak them so that your system will screech through the benchmarks like nobody's business, but that still may not do much for performance in your working environment.

The average manufacturer is reluctant to give system owners the ability to reconfigure its kernel, justifiably fearing the support nightmare this can cause. After all, not every customer is qualified to change a kernel.

Some companies even sell "reconfiguration licenses" that include the necessary files and minimal documentation, as well as promises that *no* support for changing kernels is available. These companies cannot afford to offer the support of their systems programmers, whose time is too valuable to be used for customer support.

The 4.2BSD Unix system allows these configuration limits to be patched in the kernel. One manufacturer that I spoke with plans to include a menu-driven program that permits "reasonable" changes to these limits. Patching the kernel does not require a configuration license. Also, a menu-driven approach is more acceptable to non-programmer users. However, patching is so far available only for use with 4.2BSD kernels.

Perhaps none of the previous discussion refers to you. You might have a Unix system that works beautifully, one that never runs into these configuration limits; or you may have a system that encounters these limits and you don't know what to do about it. So let's talk some more about these limits.

Built into the Unix system kernel are limits to everything it can do. One of these limits controls the number of files the kernel can open, another restricts the number of processes the kernel can start, another limits the number of pure text files the kernel can invoke, another limits the number of block buffers in the cache, and still another limits the number of file systems the kernel can mount. These limits are systemwide and can represent a resource users compete for. Signs of this competition may show up on the system console as error messages (such as those shown in Figure 1).

Have you ever reached your limits? Here is one scenario for how

```
no more procs
no file
Inode table overflow
```

FIGURE 1: SOME KERNEL ERROR MESSAGES

that can happen. Let's say you are using a Unix system that behaves well with a certain number of users, but right now one more person than usual is on the system. Instead of seeming to be bogged down, the system appears be ignoring the command you just typed. Finally, you get the error message `cannot fork: too many processes`.

Your shell's request, the `fork()` system call, which creates a new process for executing the requested command, has failed. The shell makes five attempts before it even notifies you that there's a problem. If you keep trying, you may eventually be able to execute your command — if the number of processes falls below some limit.

As mentioned above, several other limits exist. One configuration parameter, however, is not a limit in the same sense as are these others; that is, it does not refer to a size limit for a kernel table. This limit refers to the number of block buffers in memory. These buffers are used to increase the performance of disk (block) devices. If you have enough buffers, your file input/output (I/O) will be quicker. It is difficult to measure the effectiveness of your block buffers, although the Berkeley `pstat` command can provide you with a snapshot. I'll explain the significance of these buffers later.

THE PROCESS TABLE

A process is a program. The process might be executing, might be ready to execute in memory, or might be waiting in the swap space. The kernel uses the swap space on the system disk to store a process image when another process must use the memory space. Along with an image of the program code, a process also has data associated with it. The program itself accounts for some of this data, while the current state of the program generates some of it; the kernel creates the rest when the process is created.

The process table holds information about all the processes that are currently "alive." The kernel uses some of this information when deciding the next process to run or to swap out to disk. The process

table also contains the address of the process in memory or in swap space, the identity of its parent process, and the current status of the process. The process table is the data structure the kernel uses to keep track of processes.

As soon as a Unix system is brought up, several processes come alive. `init`, the grandparent of all system processes, and `swapper` will always be around. When the system goes multiuser, `init` runs the `/etc/rc` script. This wakes up various background processes known as daemons, such as `cron` or `update`. And a `getty`, `login`, or shell exists for every port that is enabled for user log-in (in `/etc/ttys` or `/etc/inittab`). Thus, without anyone doing anything, an eight-user system with, for example, two daemons running has already used up 12 process table slots (1+1+8+2).

Each time a user executes a command, the shell requests the creation of a new process via the `fork()` system call. As long as process table slots are available, everything goes well. Continuing with our example, if we let all eight users execute a single command in the foreground, then 20 process table slots are now occupied. Typical system activity, such as executing commands in the background or executing a command from an editor, adds additional processes.

Usually, the number of processes allowed is set conservatively low. One system programmer mentioned that the limit set when his firm received its Unix system port was so low that the system ran out of processes when it had only two users! The recommended size of the process table for 4.2BSD is 20 plus 8 times the maximum expected number of users.

There are, of course, physical limits to how many processes your hardware can handle without becoming bogged down. The amount of memory available to user processes and the amount of swap space are important in determining how many processes can be allowed at one time. Allowing so many processes that the swap space fills up can bring the system to a dangerous deadlock in which nothing can be changed and no process can complete or even be killed.

Another disadvantage to increasing the number of processes allowed has to do with the size of the process table itself. Each entry in the process table is somewhat large: Nine entries take up 1K byte (in System V). The larger the process table, the longer it takes to search through it. Although the search time is short, overall it can represent a significant fraction of processor time because the search is performed repeatedly. Thus, keeping the process table short speeds things up (a little).

If you adjust the size of the process table, you need to balance two things: You want to make the limit large enough so that the process table never runs out of slots, and you want to make it small enough so that you never exhaust your swap space or overpower your hardware. Sizing up your swap space is tricky because processes come in all sizes.

Overpowering your hardware means that the ratio of processes in swap space to processes in memory has become so large that an inordinate amount of system time is spent copying processes in and out of memory. Processes can execute only when they are loaded in memory. The processes waiting in the swap space need to be copied into memory before they can be executed, and processes already in memory that will not be run immediately must be copied out to make room. When too many processes exist, more time is spent copying them between swap space and memory than is spent actually executing any of them. This condition is known as "thrashing."

Both System V and 4.2BSD require a minimum of 512K bytes of memory. This much memory, however, leaves only 200K or 300K bytes for user processes after the kernel is loaded. Then, with as few as two users, the system has to copy processes constantly in and out of memory. This can increase system overhead to the point where thrashing occurs. Perhaps 1 Mbyte would be a more workable "minimum" memory requirement for multiuser operation with these systems.

THE OPEN FILE AND ACTIVE INODE TABLES

Every time a process opens or creates a file (or pipe), the kernel makes an entry in its table of open files. In System V, each table entry is small enough so that there is no reason to restrict the table size because of memory restrictions. But I have seen this table defined to be so small that it was easily overflowed with two users, producing the console message `no file`.

There really is no excuse for a small file table. Because most work is performed with files, a system grinds to a halt when no more files can be opened. Berkeley 4.2BSD bases the size of the file table on the size of the process table and on the expected user load. Based on the Berkeley sizing scheme, the open file table should have space for about 90 entries, as well as 14 for each expected user.

The inode table is related to, but different from, the open file table. An inode is the Unix system data structure that contains all the

information about a file except its name and its contents. It includes the file type, size, number of links, owner, permissions, time of last access, time of last modification, time when it was created, as well as the addresses of the data blocks in the file.

The kernel keeps a memory copy of the "active" inodes in this table for fast access. Every open file, directory file being used by the system, and directory file being used as a user's current directory has an entry in this table. When this table fills up, the following message appears: `Inode table overflow`

The entries in the inode table are somewhat large: Twelve of them take up a little over 1K byte in System V. However, not having enough inode table entries is as bad as not having enough open file table entries. In effect, no new programs can be executed once the inode table fills up, and currently active processes may become blocked.

Most systems come configured with a smaller limit on the number of inodes than on the number of files. This is so because more than one file opening can refer to the same inode. Once again, if you are configuring a system, you never want it to run out of inode table entries. According to the Berkeley scheme for estimating the size of the table, your inode table would have 70 entries as well as 9 for each expected user.

THE PURE TEXT AND MOUNT TABLES

Pure text files are an interesting topic in themselves. They represent programs that have been separated into two pieces: the code and the data area. A pure text program differs from an ordinary program because the data area of the pure text program begins on a memory protection boundary inside the target machine. This approach makes it easy to separate the code part of a program that does not change from the data that does.

When someone invokes a pure text program, the kernel makes an entry in the text table. Then, when another user requests the same program, the kernel checks the table and notices that the program has been loaded into memory already. The kernel modifies the text table to designate that another user is using the text area also, and it creates a new data area for the second user process. Thus, only one memory copy of the program code is required because many processes can use the same code area. But each maintains its own data area.

Only frequently used programs need to be pure text files. The shell, `getty`, `vi`, and the local favorite text editor are strong candidates for pure text files. If there is a heavily used program in your real-world environment, configuring it as a pure text file will improve your system's performance. You can create a pure text program by specifying a particular option to the linking loader, `ld`. Of course, this procedure requires that you have access to the relocatable object code for the program itself.

Of all the limits mentioned here, the limit on the number of text files was enough for the systems I examined. Usually, the size of the text table was 20 or more. The Berkeley algorithm suggests 24 plus the maximum number of users.

The mount table is the kernel's internal record of mounted file systems. This table does not include the root file system because the identity of the root file system is compiled into the kernel. When additional file systems are mounted, the directory on which they are mounted and the major and minor device numbers of the mounted device are added to this table by the kernel.

The mount table is used during directory (pathname) searches. When searching through a pathname, the kernel first scans the mount table for each directory in the path. If it finds the directory name in the table, the kernel continues its search through the pathname in the root directory of the mounted file system corresponding to that mount point directory.

If the directory it is searching for is not a mount point for a file system, the kernel opens the directory that is the parent and locates the inode corresponding to the desired directory.

In other words, during searches for files outside your current directory, the kernel first looks through the mount table each time it encounters a directory name. This procedure makes the mount table one of those frequently scanned spots in the kernel. Obviously, the smaller your mount table, the quicker the kernel can scan it.

Most systems seem to have a reasonable limit on the mount table size. Because you don't want this table to become a large one, a small limit, such as eight entries, is reasonable. In 4.2BSD, the limit is 15. Increasing this limit has side effects that spread throughout the kernel, increasing the sizes of other data areas.

BLOCK BUFFERS

So far, I have talked about the sizes of configurable tables in the kernel that affect system performance. However, another configurable factor looms large in system performance. The Unix system kernel manages a pool of block buffers, also known as a buffer cache. These buf-

fers are used for temporary storage of data while data is transferred to and from the disk.

Also, the data for Unix system pipelines is transferred through the buffer cache. The buffer cache reduces the time you spend waiting for disk transfers by reducing the number of disk accesses. Here is how it works.

When a process makes a request to read from or to write to a disk file, the kernel searches through the buffer cache for a block containing the desired data before it accesses the disk. If it finds the block in the buffer cache, the process can use the data immediately. Otherwise, the process has to wait until the necessary block is in the buffer cache *and* until the kernel restarts the process. The more block buffers there are, the more likely it is that the desired block will be found in the cache.

The entire system shares the buffer cache. The kernel uses the buffers while loading programs (but not while swapping), for searching directories, and other related activities. Programs that manipulate files, such as `cp`, `ed`, and `grep`, use these same buffers. Thus, there is much competition for the available buffers, and processes compete with each other and with the kernel for this important and limited resource.

When the contents of a new disk block must be read into the cache, the kernel uses a "least recently used" algorithm to select the buffer to discard from memory. This approach tends to keep the most useful buffers in the memory cache.

In System V, as in previous Bell Labs' Unix system releases, the number of block buffers is fixed at boot time. You can configure the number of block buffers in systems capable of generating a kernel, but determining how many buffers are enough is difficult. Next, I'll discuss the trade-offs to consider when you select the buffer cache size.

BLACK BOXES

The size of each block buffer is given by the systemwide block-size constant, commonly named BSIZE. In System V the block size is 1K byte. (In earlier Bell Labs' releases, the block size was 512 bytes.) This means that for every additional block buffer configured into a kernel, user memory is reduced by 1K byte. Along with the buffer itself, the kernel needs to maintain information about each block buffer, information that is kept in buffer headers. Each block in the buffer cache has a buffer header. In System V, 23 buffer headers take up almost 1K byte.

The kernel's search through the buffer headers to find a desired block takes time. This is quickly offset, however, by finding a block in the cache, rather than having to access a disk for it.

There is, however, a real problem in determining the optimum size of the cache: The only way to know how often the correct block is found (a hit!) in the buffer cache is to change the kernel so that it keeps track of hits and misses. Because only source licensees can change the kernel, the rest of us are left staring at our Unix system black boxes

scratching our heads and wondering how well our configuration is working.

However, users can make some pragmatic tests. For example, you can generate kernels with differently sized buffer caches and test them with benchmarks. But benchmarks do not a real world make. And remember, if you will, one of my original points—the "one size fits all" kernel that so many systems wind up with. The number of block buffers in a "one size fits all" kernel will be small enough to work with the minimal entry-level system and will be much less than optimal for a fully configured system.

Once again I'm going to borrow from 4.2BSD. The 4.2BSD Unix system provides a choice. It dynamically allocates 10 percent of memory or 16K bytes (whichever is larger) to block buffers at boot time; or a specified number of block buffers can be configured in the kernel. Carrying this example over, for systems with 200K or 300K bytes of user memory, 10 percent translates into 20 to 30 1K-byte blocks added to the kernel memory buffer cache.

You need to test how well a given cache size works thoroughly because so much of the system's performance depends on it. If too much space is devoted to the buffer cache, the system will spend too much time swapping because there is not enough user memory. If too little space is used, the hit rate will go down, the same block will be requested over and over again, and swapping will increase (because processes have to wait until their requested block is back in the cache).

We'll close the first installment in our two-part series here. Be sure to tune in next month, when we'll describe in detail how to tune your kernel. □

A San Francisco resident, Rik Farrow is a freelance technical writer who has written installation and maintenance guides for micro-based Unix systems. His last article for UNIX/WORLD, "All About Device Drivers," appeared in Vol. 1, No. 6.

ADVERTISERS INDEX

If you are interested in products or services advertised in UNIX/WORLD, our advertisers are pleased to send you additional information.

Simply fill out the reader service cards and circle the numbers of advertisers you are interested in. Then mail the card and expect prompt, pertinent information.

In order to correct several errors that appeared in the figures accompanying Peggy Judd's article "troff: A Text Software Boon for Scientific Authors," which appeared in the February issue, we are printing the correct figures here.

FIGURE 1

A complex configuration:

$$f = \frac{\delta}{\hbar\omega} \sum_{m=0}^{\infty} \frac{\alpha^2 m}{2m+1}$$

is produced from the input

```
.EQ  [beginning math material]
f = delta over {hbar omega}
sum from {m = 0} to inf {alpha sup 2 m} over {2m + 1}
.EN  [ending math material]
```

Taking a more complex example, such as a many-line matrix, the coding is similarly simple and logical for the user.

$$f_{3q} = \begin{vmatrix} b_1q^2+b_2q_1^2 & b_3q_1q_2 \\ b_3q_1q_2 & b_1q^2+b_4 \\ b_3q_1q_3 & b_3q_2q_3 \end{vmatrix} > 0$$

is produced from the input

```
.EQ
f sub 3q =
left (
matrix {
ccol {
{b sub 1 q sup 2  +  b sub 2 q sub 1 sup 2} above
{b sub 3 q sub 1 q sub 2} above
{b sub 3 q sub 1 q sub 3}
}
ccol {
{b sub 3 q sub 1 q sub 2} above
{b sub 1 q sup 2  +  b sub 4} above
{b sub 3 q sub 2 q sub 3}
} }
right ) > 0
.EN
```

The use of curly braces logically identifies the parts of the matrix both for the program and for the keyboarder.

FIGURE 2

This figure gives examples of a few entries in a define file and their corresponding output. The first two definitions represent local definitions for a particular paper. The other two are composite characters, and once the definition is established they should be saved in a define file to be used for all future work where the characters might appear.

Define file entries:

```
define ES   'roman Eu sub 2 ( roman SO sub 4 ) sub 3
cdot 8 roman H sub 2 roman O'
define St  'S sup t = (n  + 1)/s'
define <=>  %   "\s− 2\z<h' .3m'\v' .1m'\z\(eq
\h' .6m'\(eq\v' −.1m'\h' −.6m'>\s+2"  %
define apeq2  %   "\v' .18m'\fR\z=\fP
\v' −.18m'\v' −.14m'\(ap\v' .14m' "  %
```

Input for the above entries appears in the left column, and the corresponding output appears on the right.

Input	Output
ES	$Eu_2(SO_4)_3 \cdot 8H_2O$
St	$S' = (n+1)/s$
$<=>$	$\Longleftrightarrow$
$apeq2$	$\cong$

For a character that is a combination of one or more characters, defining the correct amounts of horizontal and vertical motion will depend on the output device being used. This usually requires some testing to perfect the final appearance. The last two definitions above are written for a III VideoComp 500 phototypesetter.

FIGURE 3

For example, a simple table

Spherical analyzer specifications			
Electrode Position	Inner Radius (mm)	Aperture Diameter (mm)	Relative Voltage
A	3.96	3.05	1.0
B	10.54	0.71	0.264
C	14.10	0.71	0.164
D	20.27	0.71	0.060

is obtained from the following input (tabs are designated by a ■)

```
.TS  [designates the beginning of tabular material]
center, box;
c s s s
c c c c
c c c c
c n n n.
Spherical analyzer specifications
Electrode ■ Inner ■ Aperture ■ Relative
Position ■ Radius (mm) ■ Diameter (mm) ■ Voltage
=
A ■ 3.96 ■ 3.05 ■ 1.0
B ■ 10.54 ■ 0.71 ■ 0.264
C ■ 14.10 ■ 0.71 ■ 0.164
D ■ 20.27 ■ 0.71 ■ 0.060
.TE  [designates the end of tabular material]
```

A more complex table, containing math and a block of text, is shown below:

Source	Change in R
Change track-finding efficiency by 5% at worst point, scaled back to no change at 100% efficiency	0.15
λ_2	$-\frac{i}{\sqrt{2}}(Y_3^3 + Y_3^{-3})$
λ_3	$Y_2^0 \quad 0 \le l \le 3$
δ_2	$\frac{1}{\sqrt{2}}(Y_1^2 + Y_1^{-2})$

The input required to produce the table above is shown below (tabs are designated by a ■).

```
.TS
center, box;
c c
l c.
Source ■ Change in $R$
=
T{
Change track-finding efficiency by 5% at worst point, scaled back to no
change at 100% efficiency
T} ■ 0.15
$lambda sub 2$■$- i over {sqrt 2} (Y sub 3 sup 3  +  Y sub 3 sup -3 )$
$lambda sub 3$■$Y sub 2 sup 0 ~ ~ 0 <= l <= 3$
$delta sub 2$■$1 over {sqrt 2} (Y sub 1 sup 2  +  Y sub 1 sup -2 )$
.TE
```

Like the math language, TBL codes are easy to learn. Knowing only a portion of the available table format codes is sufficient for most tables. Referring to the documentation for exceptions to the rule takes care of the rest.

Continued from page 105

WATCH THAT CHAPTER 0

Before discussing the book's contents, I need to point out that the first chapter is numbered zero, as is the first subscript of an array in C. Because of this numbering scheme, it may seem as if I have confused the chapter numbers in the rest of this review.

The first chapter (Chapter 0) is an introduction to both the book and to the C language. It discusses the language's history, its basic design viewpoint, a brief overview of the language, and a description of the chapters that follow. More importantly, it attempts to give the reader an understanding of what C is and the philosophy behind its creation.

The next chapter is called "A Tutorial Introduction." But this name is a misnomer; instead, the chapter should be called "A Brief Overview." Covered in what might be the densest 26 pages I've ever read are data types, looping constructs, character input and output, arrays, functions, argument passing, strings, and the scope of C variables. This section is likely to intimidate anyone not fully prepared for it.

Also included are 23 exercises that seem to suggest that the reader should be able to write significant C programs by page 22. I seriously doubt that most first-time readers, no matter how skilled, could do that. For these reasons, I think this chapter is one of the book's weaknesses.

Chapter 2 takes a more leisurely look at data types, operators, and expressions of the C language. C contains several operators that act like a programming "shorthand." This chapter not only describes most of these operators, but also data types, both numeric and character constants, variable declarations, and conversions between the various data types. In all, this is an excellent chapter with only one minor flaw: the precedence table at the end of the chapter. Of the 13 operators that top the list, only 5 have been discussed.

A SMALL LANGUAGE

C is a small language. It contains less than 30 keywords and 40 operators. Everything else is done by functions, including all input/output (I/O), memory allocation, and string manipulations. You must understand how functions are used to "build" programs in order to use C effectively. This is precisely what Chapter 4, "Functions and Program Structure," does. In addition to functions, it also discusses the scope of C variables—such as using variables globally (extern) or locally (auto, static, and register), variable initialization, recursion, and macro substitution (by the preprocessor).

I suspect that a survey of C programmers would show that pointers were the most difficult aspect of C for them to learn. Yet that same sample would probably identify this concept as one of the most useful. This may explain why Chapter 5, "Pointers and Arrays," is written as it is. The first time I read this chapter, I found that I was getting more confused with each page. It dives right into pointers and variable addresses without telling us why they are important. This approach makes it hard to get a handle on where the chapter is going and why. Five pages later, when the chapter is telling us about the similarity between arrays and pointers, it is easy to lose grasp of the whole point.

Probably because the concept of structures is more concrete, Chapter 6 ("Structures") is more intelligible than the previous one on pointers. However, the authors for some reason use an obscure example—a program that counts C keywords—throughout the chapter. How often do you ever need to count specific words? Aside from this less relevant example, though, the chapter fully and completely covers structures, fields, unions, and typedefs.

Chapter 7 deals with the input and output functions contained in C's standard I/O library. This chapter provides excellent coverage of the formatted display (`printf()`) and formatted input (`scanf()`) functions and accessing files.

CONCLUDING REMARKS

Without question, this is an important book in the computer science field. Indeed, it is a rare book: It not only teaches the C language but also serves as a reference to the language.

Although I don't recommend this book to a beginning programmer, I do strongly recommend it to anyone working with the C language.

This book was published in 1978. In the introduction, the authors write: "Since C is an evolving language that exists on a variety of systems, some of the material in this book may not correspond to the current state of development for a particular system." It is amazing how little C has changed since this book became available over six years ago. That alone is an amazing testament to the book's impact. □

Ray Swartz is the founder and president of Berkeley Decision/Systems Inc., a Santa Cruz, Calif.-based computer consulting and training firm that specializes in the Unix system.

A COLLECTION OF READER OFFERINGS

BY DR. REBECCA THOMAS

We have several interesting contributions for you this month. First, Carl Brandauer submits a few short Bourne shell scripts that provide quick and simple ways to simulate the functionality of the `more` command for Bell System III and V Unix systems. (The `more` program, available with Berkeley Unix systems, allows the display of text a screenful at a time.)

Many System III and V Bell Unix releases don't have the `more` command, but they do have a version of `pr` that pauses after displaying a page. Carl shows us how to simulate `more` with this version of `pr`. A file perusal utility that can display a screenful at a time is a "must" when trying to read text at high terminal display rates. Mr. Brandauer writes. . .

Dear Dr. Thomas:

Attached are a description and listings of several versions of a Bourne shell command I began using shortly after System III was released inside Bell Labs and long before I had heard of `more`. I still use it today since it does not have any of the nastier features of `more`, such as mangling the text or leaving the terminal in unpredictable and strange states.

When you are using a CRT terminal at data rates greater than 1200 baud, there is a need for a command that allows the user to print the contents of a file 22 lines at a time. Even the nimblest fingers cannot hit "Control-S" quickly enough to keep up with the speed at which the screen is printed.

Although a number of compiled utilities exist, such as Berkeley's `more` and Kernighan and Pike's `p` [*Doctor's note:* See page 181 in Kernighan and Pike's *The Unix Programming Environment,* Prentice-Hall, 1984], users of System III and V can do the job much more simply with a short Bourne shell procedure.

The heart of all three procedures described here is found in a new option for the old standby `pr`. Figure 1A shows the contents of an executable shell command file called `sc` (for screen). The familiar options `-t` and `-122` (`1` is the letter "ell") to `pr` eliminate the header and footer and set the page length to 22 lines, respectively. The new option, `-p`, will cause `pr` to pause before beginning each page, prompting by ringing the terminal bell, and waiting for a carriage return, thus printing 22 lines at a time.

Figure 1B shows two ways to use `sc`. The first is self explanatory; the second works because, if no filename argument is specified, `pr` reads from its standard input (output from *command*).

Although `sc` as written will print in 22-line chunks, it has a major drawback. The carriage return needed by `pr` to print the next 22 lines is echoed to the terminal, thus inserting a blank line on the screen that is not in the input. The listing in Figure 2 shows how two more lines of code can avoid this problem. The `stty -echo` command line turns off echoing the new line (the culprit in the first version), and the trap statement ensures that echoing is restored regardless of how the procedure is terminated.

If invoked with multiple file names, the procedures shown will stop printing and prompt between each file, but without displaying the name of the next file. The final ver-

a. Shell script listing

```
$ cat sc
pr -t -122 -p $*
$ []
```

b. Two ways to use `sc`

```
$ sc file1 file2 ...
$ command file1 file2 ... | sc
$
```

FIGURE 1: THE SIMPLEST VERSION OF `sc`

```
$ cat sc
trap 'stty echo; trap 0; exit' 0 1 2 3 15
stty -echo
pr -t -122 -p $*
$
```

FIGURE 2: AN IMPROVED VERSION OF `sc`.

```
$ cat sc
trap 'stty echo; trap 0; exit' 0 1 2 3 15
stty -echo
if test "$#" -gt 0
then
        while test "$1"
        do
                pr -t -122 -p $1
                shift
                if test "$1"
                then
                        echo ">>>>>>> $1 <<<<<<"
                fi
        done
else
        pr -t -122 -p -
fi
$
```

FIGURE 3: THE FINAL VERSION OF `sc`.

sion shows how you can remedy this deficiency. Unfortunately, the remedy complicates things if the command is to work properly, both when given one or more filename arguments and when used at the tail end of a pipe.

Essentially, you must write two commands, one for each case, as well as a conditional statement used to choose between the two. (Note that with System V, `test` is built into the shell, greatly enhancing efficiency.) See Figure 3 for the resulting code. Whether the additional code is worth it is, of course, up to the user.

Carl Brandauer
President of Daemon
Associates Inc.
Boulder, Colo.

TWO CONTRIBUTIONS FROM TORRANCE

Marc Ries offers us two contributions this month. The first is a front-end for the `su` command. His shell script provides additional security-checking and logging functions for the substitute user command. (`su` allows a system user to temporarily assume the identity of another system account without having to log in to that account.)

Dear Dr. Thomas:

Most commercial operating systems, including some enhanced Unix system implementations, provide methods for flagging the presence of privileged users. The following Bourne shell could be used as front-end to the standard Unix `su` program, providing visual reminders of this special status.

The benefit of this shell script is that you may leave out portions so that only the modified prompt string, `PS1`, is passed to the `su` process. You could also expand this shell script, including testing for the existence of a "logfile" before other processing decisions were made. Finally, the system administrator could rename the original `su` program to `newsu`, could rename this shell script to `su`, and could execute `newsu` through this Bourne shell script front-end.

[*Doctor's note:* Figure 4 shows Mr. Ries' shell script.]

Recently, while I was working with the PC/IX operating system, I found that a missing useful tool was the `script` command. This utility, available on many Unix system implementations [*Doctor's note:* `script` is distributed with Berkeley Unix], allows input and output data to be captured in an intermediate file. `script` is especially useful for retaining a "hard copy" of some particular test or evaluation scenario for future use.

Enclosed is the source to the shell script for mimicking the functionality of the standard `script` utility. Although most of the code is used for checking invocation errors, the main shell instructions consist of only two lines. As presented, the `script` command always redirects output to the file `script.txt` and allows only eight arguments to be passed via the command line (see Figure 5).

Marc A. Ries
{ucbvax!trwrb!fcomp!marc}
Software Engineer
Reifer Consultants
Torrance, Calif.

[*Doctor's note:* Mr. Ries' `script` utility also differs from the Berkeley version in another important aspect: The Berkeley version allows logging of an entire terminal session, whereas the submitted script logs output from one command at a time. You must re-invoke it for each command whose output you wish to save on disk.]

```
$ cat ssu
: # SSU: su program, with additional security checking/logging


: # DEFINE
     logdir=.                    # system dependent
     logfile=logsu               # system dependent
     terminal=/dev/console       # system dependent


if test $1                       # ssu called with a user argument?
     then mode=$1                # if so, retain it for future use
     else
     mode=root                   # else, default to user root
fi                               # end if


case $mode in                    # process the user arguments
     root | bin)
          echo "Why are you invoking the su command? "
          read answer            # store response in answer
          set `who am i`         # assign output to positional args
          echo ""                # skip line
          echo "User[$1] on $2 as [$mode] at `date`" >$terminal
          echo "Superuser[$1] on $2 at `date`" >>$logdir/$logfile
          echo $answer >>$logdir/$logfile
          PS1="$mode: "          # use substituted user name in prompt
          readonly PS1           # no change allowed for new prompt
          export PS1             # add to the environment
          exec su $mode          # replace this shell with su process
          ;;

     *adm* | *usr* | *user*)     # look for administrative keyword
          set `who am i`         # assign output to positional args
          echo "User[$1] on $2 as [$mode] at `date`" >$terminal
          PS1="$mode: "          # new prompt
          readonly PS1           # can't change prompt
          export PS1             # add to environment
          exec su $mode          # invoke su
          ;;

     *)                          # anything else
          PS1="$mode: "          # new prompt
          readonly PS1           # can't change prompt
          export PS1             # add to environment
          exec su $mode          # invoke su
          ;;
     esac                        # the end
     $
```

FIGURE 4: A BOURNE SHELL SCRIPT FRONT-END FOR THE su PROGRAM

```
$ cat script
:
# script -- Bourne shell script to mimic "script" under PC/IX
# Note -- doesn't function correctly under XENIX 2.x or with a
#         full-screen editor
#
# -------------- error handling routines follow --------------
trap 'echo "script: Exiting..."; trap 0' 0 1 2 3 4 15
if test "$#" = 0; then cat <<\!
     Usage: "script command [options]" where:

     script      => this program
     command     => program to run
     options     => arguments for the command
     script.txt => file to which output redirected
!
exit 1
fi
echo " script: Session redirected to <script.txt>"
# --------------- main program follows --------------------
echo "$1 $2 $3 $4 $5 $6 $7 $8" >script.txt
tee -a script.txt | $1 $2 $3 $4 $5 $6 $7 $8 | tee -a script.txt
# ----------------- end of script.sh -----------------------
$
```

FIGURE 5: A BOURNE SHELL SCRIPT TO MIMIC THE `script` COMMAND

```
$ cat hunt
# hunt...a script to prode (or, prompt) rusty "find" consumers
#
echo "enter file to find:"
read file
echo "enter directory to begin search:"
read directory
echo "Looking for file named $file...please wait one moment."
find $directory -name $file -print
$
```

FIGURE 6: A BOURNE SHELL FRONT-END FOR THE `find` UTILITY

STEELE'S SHELL SCRIPT

Our last contribution this month comes from Dr. J. Michael Steele. He provides us with a much simpler version of the `refind` shell script published in the March 1985 installment of this column. Dr. Steele's version is appropriate for one of the most common applications of the `find` command—locating files with a specified filename.

Dear Dr. Thomas:

I don't know if you are in the market for reasonably naive shell scripts, but I would like to offer you one that I use almost daily and that illustrates some nice features of `echo` and `read`.

I wrote the script because I am congenitally unable to remember the syntax of `find`. As a side effect, the script shows how to make "friendly" whatever you find unrememberable. (Naturally, a few Unix system philosophy eggs may be broken enroute.)

Enough of my sales pitch; here is the script. I'll bet you'll use it yourself.

[*Doctor's note:* See Figure 6 for the listing.]

Obviously, a serious Unix system user will not want the "warm fuzzy" *Please . . . etc.*, but I stuck it in to illustrate a point.

J. Michael Steele
Professor of Statistics
Princeton University

"Wizard's Grabbag" is a regular feature of UNIX/WORLD. Submit your shell scripts, C programs, or tips and techniques that ease the burdens of system administrators and programmers to "Wizard's Grabbag," UNIX/WORLD, 444 Castro St., Suite 1220, Mountain View, CA 94041. Authors of published entries receive $50 for shell scripts, `awk` scripts, `sed` scripts, `lex`, `yacc`, and C programs, or tips.

Please follow these guidelines for reader contributions: Write your shell scripts, C programs, and other code so that it is portable across different versions of the Unix system. If possible, it should run without change on Bell Version 7, Systems III and V, and Berkeley 4.x. Thus, you should use "universal" Unix utilities such as `who am i` (all systems) in lieu of `whoami` (Berkeley only), and the Bourne shell, if possible, when coding shell scripts.

However, C shell scripts are also welcome because most of our readers now have access to this popular command interpreter. Use the standard I/O library when writing C code. In addition, use the lint syntax checker to eliminate nonportable constructions and compile the code with a portable C compiler such as `pcc` to help ensure portability. Hardware dependencies, such as terminal control sequences, should be eliminated or at least minimized and isolated to one code region or to a separate module. Keep your example as short as possible, say under 100 lines of code. □

ERRATA FOR THE MARCH 1985 WIZARD'S GRABBAG

Please note the following errata for figures that appeared in the March 1985 "Wizard's Grabbag."

Line 33 of Figure 1 page 114 should read—`paths="$response"`.

A single space character should appear after the backquote in line 39.

In line 53 of this same figure the expression on the right-hand side of the assignment statement should be enclosed in backquotes.

Line 68 should read `print= "-print"`.

Figure 2 is incomplete. The following display should have appeared after the line "Select file attributes sought. . ." and before "Exactly 7. . .":

```
.........................File name of: a.out
Other attributes as well (y or n)? y
.................................Owned by:
.........................Group-owned by:
.............Permissions equal (octal):
....Modified more recently than (file):
....Last Modified N days ago; N equals: 7
```

NEWS FROM AT&T

AT&T, MICROSOFT SETTLE DEBATE, DECLARE SYSTEM V 'STANDARD'

At this year's UniForum in Dallas, AT&T and Microsoft laid to rest any further debate concerning System V versus Xenix as the commercial standard. The two companies disclosed that they would work together to provide future compatible releases of System V (from AT&T) and Xenix, which is Microsoft's implementation of AT&T's Unix operating system. This announcement was made as part of an expanded business relationship between the two companies.

Unix System V and Xenix are both multiuser, multitasking operating systems. Microsoft's Xenix, however, is the most widely installed microcomputer operating system derived from the Unix system.

As part of the truce, Microsoft, which had previously disclosed its intentions to make Xenix compatible with System V, re-affirmed its plan to provide a System V version of Xenix, while AT&T announced that it intends to offer Xenix System V on its PC 6300 personal computer.

Microsoft, which worked with AT&T to finalize the AT&T System V Interface Definition, will implement the changes needed to bring Xenix into conformity with that interface definition. This version of Xenix will be upward compatible with the current Xenix release, thus ensuring an upgrade path for current Xenix customers.

Microsoft and AT&T also said that they intend to maintain compatibility between the Xenix V product and the System V Interface Definition on an ongoing basis.

PARAMETERS FOR SYSTEM V VERIFICATION

As part of the two firms' expanded business relationship, AT&T designated Microsoft as the pilot participant for AT&T's System V Verification Service. The service will ultimately be made available to all Unix System V licensees and will be used to help ensure compatibility between different Unix System V-derived implementations.

As a result of Microsoft's early participation in the verification program, Microsoft said it expects that its Xenix V will be the first verified System V product for a microcomputer not owned by AT&T.

AT&T said that, following its verification of Xenix System V, it intends to offer a version of this product for the AT&T PC 6300.

UNISOFT TO DEVELOP SYSTEM V VALIDATION SERVICE

UniSoft Systems, the Berkeley, Calif.-based Unix system porting house, will assist AT&T in developing a verification test software program to validate derivatives of the Unix system developed by independent computer manufacturers, the two firms announced.

The firms said the software program to be developed by UniSoft Systems will determine whether derivatives of Unix System V actually meet the AT&T definition. All original-equipment manufacturers (OEMs) who pass this verification test can claim to have developed a System V operating system. This will ensure that every application software program designed to the System V specifications will run on all validated versions of System V.

AT&T RELEASES TWO PC GRAPHICS PACKAGES

AT&T has introduced two new microcomputer software products designed to enhance the graphics capabilities of personal computers. The programs, the AT&T Paint and Image Processing Software and the AT&T Business Graphics Presentation Software, will run on the AT&T PC 6300 and compatible computers equipped with either the AT&T Video Display Adapter or the AT&T Image Capture Board.

The Video Display Adapter is a high color-resolution graphics board that enables personal computers to display continuous-tone, television-quality graphics on color televisions or special monitors, such as analog RGB (red, green, blue) monitors.

The Image Capture Board is a high color-resolution imaging board for personal computers used to capture and display continuous-tone, television-quality images from a video source, such as a home video camera.

In addition to the drawing capabilities expected from paint software, Paint and Image Processing Software enables the user to perform real-time image processing on electronic photographs taken with the AT&T Image Capture Board.

These functions are similar to those a photographer would perform during retouching, such as adjusting color, adding to or eliminating elements in the photo, combining photos, and cropping.

The Paint and Image Software has functions for color manipulation, text overlay, the merging of multiple images, drawing, and layout. It can also be used to create NAPLPS (North

American Presentation Level Protocol Syntax) frames for videotex services.

The AT&T Business Graphics Presentation Software uses data from popular spreadsheet packages, such as Lotus 1-2-3 and SuperCalc3, to generate a variety of standard graph and chart formats, including two-dimensional and three-dimensional bar and pie charts. It also enables users to design customized formats easily.

This software package is compatible with the Paint and Image Processing Software, permitting the merger of continuous-tone images with charts and graphs. It also can create NAPLPS frames for use in videotex services.

In addition to graph and chart creation, the Business Graphics Presentation Software can be used to generate computerized "slide show" presentations. Users can incorporate electronic photographs taken with the Image Capture Board, pictures created by the Paint and Image Processing Software, and charts created by the Business Graphics Software in a programmed slide show for display on a television or monitor. The slide show can also be sent over telephone lines and filed in another computer for use at a remote location.

Features of the slide show function include automatic or manual display control, a variety of fade-in and fade-out techniques, and keyboard or mouse input.

Both the Paint and Image Processing Software and the Business Graphics Presentation Software require 256K bytes of memory for use with AT&T's Video Display Adapter and 512K bytes of memory for use with the Image Capture Board.

At press time, both products were scheduled to become available during the second quarter of 1985 at computer stores and department stores that sell computers. AT&T has not yet announced retail prices. □

For many reasons, the move from the MS-DOS / PC-DOS world to the Unix system will be an easy crossing. A little history of MS-DOS will explain why.

COMMUNICATING ON UNIX SYSTEMS: THE `mail` AND `write` COMMANDS

BY BILL TUTHILL

Using CP/M or MS-DOS is like rowing across the ocean by yourself—there's nobody around to talk with. On the other hand, using the Unix system is more like taking a cruise on an ocean liner—interesting people might be on the ship with you.

Depending on the hardware configuration, Unix systems are capable of supporting many users, and these machines are often connected to networks of varying sophistication. Users can transmit electronic mail and news across these networks. More immediate connections can be established on a single machine or between machines hooked together by local-area networks such as Ethernet.

Sending a message by electronic mail is the least intrusive (and slowest) way to communicate with another user. Your message is stored in a directory, where the recipient can read it when convenient. In order to send mail to someone on your machine, you have to know the recipient's log-in name. If your message must travel over a network, you need to know the recipient's network address.

The best electronic mail facility I've seen is "Rand-mail," similar to "INmail" on Interactive Systems' Unix. Unfortunately, Rand-mail isn't widely available. The second best is Berkeley `mail`, which is standard on Xenix and which is available as `mailx` on System V. The worst is the regular mail facility on System V, which is similar to the one on Version 7. We'll discuss Berkeley `mail` here since it is the most widely available alternative.

If you have mail when you log in, the system gives you this notification: `You have mail.` To read your messages, simply type `mail` after your shell prompt. The system responds by summarizing your messages as shown in Figure 1.

To read a message, type its number after the ampersand (`&`) prompt. To delete a message, type `d` and then the number of the message. For help, type a question mark. To quit the program, type `q`, and you will be returned to the shell. Messages you have read are placed in a file called `mbox` in your home directory, whereas unread messages go back to the spooling area.

NETWORKS

Connecting your Unix system to another Unix system isn't hard, but it does require study, work, and patience. You'll need a dedicated phone line, a good auto-dial, auto-answer modem, UUCP software for your machine, and a cooperating site that has the same equipment. Normally, one machine calls the other at a fixed time of the day or whenever there is any traffic. The machines exchange files using an error-correcting, packet-based protocol.

If the machine you call is hooked to other systems around the country, then you are probably on UUCPnet. This allows you to send electronic mail all over the world, provided you know the correct path to your destination. This is a point-to-point network, so messages can get lost at any point. Haphazard and mutable, UUCPnet is one foundation of Usenet, an electronic bulletin board administered mostly by its users.

If you're on a large machine with lots of disk space, you could be connected to Usenet, an informal international network for information exchange. Usenet is organized by news groups, of which there are hundreds (some more useful and interesting than others). Don't even think of joining Usenet unless you have at least 15 Mbytes of spare disk space and are willing to tie up your phone all night long.

WRITING TO A TERMINAL

Unix system devices are treated like files. Your terminal, in fact, is treated as a file named `/dev/tty`. All users have a device file associated with their terminal. When you run the `who` command, you can see what device file is associated with what terminal. Figure 2 shows the output of `who`, which indicates that `wnj` is logged in on `/dev/tty01` and that `tut` is logged in on `/dev/tty02`. Suppose I tried to telephone `wnj`, but his phone was busy. I could still reach him by writing to his terminal as shown in Figure 3.

```
% mail
Mail version 2.18. Type ? for help.
"/usr/spool/mail/tut": 3 messages
> 1 leslie Tue Dec 20 21:19 16/473 "software
bugs"
  2 kim    Wed Jan  2 11:55 25/866
"UniForum"
 3 alice   Thu Jan 17 14:07 21/649 "lunch
Monday"
&
```

FIGURE 1: SIMPLE SUMMARY OF MESSAGES

```
%who
wnj tty01 Jan 29 21:40
tut tty02 Jan 29 22:17
```

FIGURE 2: SAMPLE RESPONSE TO `who` COMMAND

The `write` program automatically sends a message to the terminal of the person you specify (in this case `/dev/tty01`, which, as I mentioned above, is `wnj`'s terminal). You can type a message of any length, ending it with a "Control-D," which indicates the end of file. After you type each line, it appears on the recipient's screen. Thus, `wnj` sees the message shown in Figure 4. He could answer by typing: `write tut`.

It is possible to have two-way conversations as long as both parties observe some conventions, such as entering `-o-` (for over) when they're done typing, and `-oo-` (for over and out) when the con-

```
% write wnj
Please stop by when you're off the phone--
Henry is here.
^D
```

FIGURE 3: A SAMPLE MESSAGE USING THE `write` COMMAND

```
Message from tut on tty02 . . .
Please stop by when you're off the phone--
Henry is here.
EOF
```

FIGURE 4: THE MESSAGE AS IT WOULD APPEAR ON THE RECIPIENT'S CRT SCREEN.

versation is complete. However, if both parties type at the same time, the screen becomes garbled.

Needless to say, it can be bothersome to be interrupted by `write` in the middle of important work. This can be a real problem on large systems with unfriendly users. You can always turn off the writing of messages to your terminal by typing: `mesg n`. If you want to turn messages back on (if you need to write someone, for example), type: `mesg y`. To disable messages, the system simply makes your terminal device unwritable by others.

The `write` command is available on all versions of the Unix system. The Berkeley Unix system also includes the `talk` command, which establishes a two-way connection that allows both parties to type at the same time. Characters appear on the other screen as soon as they're typed. The principal advantage of `talk` is that it works between machines hooked together by Ethernet. However, `write` works only when both terminals are connected to the same machine. □

Bill Tuthill is a member of the technical staff at Sun Microsystems, Mountain View, Calif.

MARKETPLACE
CAREER OPPORTUNITIES

Readers who have computer merchandise or services to sell, buy, or trade and who want to be included as a classified advertiser in UNIX/WORLD Magazine should write to: Advertising Department, Tech Valley Publishing, 444 Castro Street, Suite 1220, Mountain View, CA 94041. 415/964-0900. The rate for the classified section is $12 per line. The deadline for ads is the 20th of the 2nd month prior to publication. Please include complete name and address in every ad.

WMZ Consulting Group—System Consultants for total implementation of information systems. WMZ provides design, implementation, training, and ongoing support. We have experts in UNIX on staff and we are not committed to any software or hardware. We sell solutions not iron. Call (415) 945-2070, or write WMZ Consulting Group, 1483 Ashwood Drive, Martinez, CA 94553.

CALENDAR

SOFTWARE REQUIREMENTS, SPECIFICATIONS, AND TESTS
April 30–May 3, 1985
San Diego, Calif.

Learn practical techniques for the successful performance of software requirements, specification, and test phases of micro and minicomputer system development. Contact Ruth Dordick, Integrated Computer Systems, 6305 Arizona Place, P.O. Box 45405, Los Angeles, CA 90045; 800/421-8166.

AT&T, IBM, AND UNIX: DESKTOP DIRECTIONS
May 1, 1985
San Jose, Calif.

In this seminar the people at Yates Ventures will discuss how AT&T and other key multiuser desktop manufacturers will respond to the challenge posed by IBM. Specifically emphasized will be marketing and distribution strategies for Unix systems and supermicros. Contact Glenn Chase at 415/424-8844.

INTERMOUNTAIN—OFFICE AUTOMATION AND COMPUTER SHOW
May 1–3, 1985
Salt Lake City, Utah

This show is intended for users, distributors, and OEMs. Contact Scott Garrett at 801/485-6591.

KANSAS INFORMATION EXPO AND CONFERENCE
May 1–2, 1985
Wichita, Kansas

This new exposition and conference, sponsored by the Wichita chapter of DPMA, is intended for users, distributors, and OEMs. Contact Mark Cramer at 303/696-6100.

BYTE COMPUTER SHOW
May 2–4, 1985
Anaheim, Calif.

This show is intended for engineering management, distributors, and many types of users. Contact Mitch Adelson at 617/449-6600.

UNIX FOR MANAGERS
May 3, 1985
Bellevue, Wash.

This seminar is for managers considering a Unix system for office automation or software development. Contact Kathy Howard, SSC, P.O. Box 7, Northgate Station, Seattle WA 98125; 206/367-UNIX.

COMDEX/SPRING
May 6–9, 1985
Atlanta, Ga.

This show is intended for distributors, OEMs, system integrators, and buyers of software and services. Fifty thousand people attended the last show to view offerings from 850 exhibitors. Contact Aileen Vogt at 617/449-6600.

UNIX: A HANDS-ON INTRODUCTION
May 7–10, 1985
Washington, D.C.
June 4–7, 1985
San Diego, Calif.

This course emphasizes in-class, hands-on exercises using the Unix operating system. Contact Ruth Dordick, Integrated Computer Systems, 6305 Arizona Place, P.O. Box 45405, Los Angeles, CA 90045; 213/417-8888.

BOSTON AREA MICRO SHOW (SPRING)
May 18–19, 1985
Woburn, Mass.

This show is intended for small business, personal computer, and educational users. Ten thousand people and 200 exhibitors attended the last show. Contact Karen Cane at 201/297-2526.

C PROGRAMMING WORKSHOP
May 20–24, 1985
Bellevue, Wash.

This hands-on workshop is designed to teach engineers and programmers how to write and evaluate C programs for specific applications. Contact Kathy Howard, SSC, P.O. Box 7, Northgate Station, Seattle WA 98125; 206/367-UNIX.

To further the education of Unix system users and professionals, UNIX/WORLD each month provides a calendar of events as a service to our readers. Please send information about your seminar, trade show, or course to Calendar Editor, UNIX/WORLD Magazine, 444 Castro St., Suite 1220, Mountain View, CA 94041. All items should include a title, a brief description of the course, seminar, or trade show, the date of the event, the city and state where it will be held, the name of the sponsoring company or organization, and an address and phone number for further information. Please submit all items at least four months in advance. □

Continued from page 128

ripherals to make music, usually in ways totally unintended by the manufacturers. On various models of the IBM chain-type line printers, such as the classic 1403, it was possible for the programmer to precisely control print hammer strike sequencing by sending the appropriate file to the queues for printing.

Depending on how many hammers were striking the paper at approximately the same time, you could generate various musical notes. "She'll Be Coming Around the Mountain When She Comes" was a particular favorite, although it ate up reams of paper in the process of playing. These printer music programs also kept the CE pretty busy trying to keep the printers from wearing out prematurely.

It was also possible to send a particular print line to some of those printers and cause *all* hammers to fire almost simultaneously, which blew a fuse on the earlier models. But that hardly counts as genuine music, unless you're playing the finale of the "William Tell Overture," of course. On today's printers, playing music is usually pretty difficult. The fancy new laser printers don't even give us any significant mechanical actions to watch at all. For shame!

Then there were the programs to play music on tape drives. Yes, through sufficiently creative programming, it was possible to control the stepping motor servo whine to play tunes. Although I've never heard one of these programs in action, I've been assured that they were most impressive.

In fact, it's this type of program that gives us a glimmer of hope. Although most of the tape drives for which those earlier programs were originally written are gone, one of their original authors, a friend of mine, tells me that he is busily working on a new version. His new program is oriented around the modern DEC TU78 tape drive, which is common on many VAXen today. Although the program is still unfinished, he tells me that he's making excellent progress and that he has high hopes for the finished "product." Perhaps he can be talked into releasing it when it's complete.

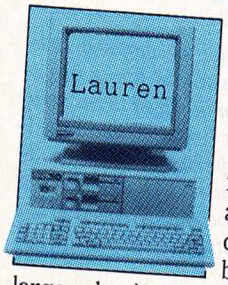
Nobody would claim that the computers and peripherals of yesterday were genuinely more *useful* than the multitude of models available today. The functionality of even today's home computers was undreamed of in the early days of computing. But it seems to me that there was something special about those early machines and the way they functioned, something that has been lost in today's shiny plastic and chrome computers. Maybe those old machines somehow seemed to have more character, more *soul,* if you will. They're antiquated toys by today's standards, but man, those were *real* machines!

```
--Lauren--
UUCP: {decvax, ihnp4,
      seismo, clyde,
      bonnie}!vortex!
      lauren
```

Lauren Weinstein is a computer/telecommunications consultant living in Los Angeles. He has been involved in an array of projects that range from the mundane to the bizarre. He has particular expertise in the fields of computer networking, the Unix system, microcomputer technology, and telecommunications systems.

SHAKE, SPIN, RATTLE, AND ROLL

BY LAUREN WEINSTEIN



Once upon a time, not so very long ago, computers and their associated peripherals didn't always just sit quietly like little bread boxes or large plastic and metal monoliths. Back in the good "old" days, there was more to see and hear from the computing machines than a "power on" light and the whirring of fans.

Somehow, the older machines—even though they were big, sluggish, power hungry, and relatively unsophisticated—were still more fun to watch and hear. Although it was long out of use by the time I arrived, a beautiful old differential analyzer was displayed for many years at the UCLA Mathematical Sciences complex. This massive, programmable calculating machine, consisting of a very long complex of rods, gears, and miscellaneous mechanical devices, was fired up once or twice a year for various events.

It made an incredible racket, but in its day it was just the machine for computing ballistics and other complex calculations. You programmed this beastie with a screwdriver, changing rods and gears. The last time I saw that unit, it had been broken down into boxes of gears and assorted metal parts for shipment to the Smithsonian. At the time, I was convinced they'd never be able to put it back together.

PUNCH, PRINT, AND PROGRAM

If programming with screwdrivers is too primitive for you, consider some other mechanical marvels. The IBM 514 card punch and its associated calculating printer (the 402) were programmed via large "plugboards" that controlled the functioning of their various mechanical linkages. The plugboards were bulky, noisy, and slow, but they got their jobs done.

By the way, the 402 printer had a particular characteristic that gives considerable insight into technical design. The cards on the 402 included a fixed-width alphanumeric field of exactly 43 characters. Why 43? Guess how many characters it takes (including a space between each word) to spell out "International Business Machines Corporation". . . . You guessed it!

Then there were the IBM 082 and 083 card sorters. The cards would go zipping along the sort surface, dropping, as if by magic, into the appropriately programmed slots. Who needed CRT screens and fancy keyboards when you could fling cards around like that? And remember the IBM 026 and 029 keypunches? Hell, lots of the 029 model are still in use today, merrily clanking and clicking away.

Actually, *real* programmers back then preferred to punch their cards on the IBM 024. That model, unlike the later models for sissies, didn't print what you were typing onto the card. Anyone worth his salt could read the punched holes directly anyway, right?

THE RECENT PAST

I suppose that much of the above sounds like ancient history to many of you. Well, even the relatively *recent* past had interesting gadgets you could watch and hear on many computers.

One of the PDP 11/45's best attributes was its dandy display register. You could program your choice of pattern into the register's memory and have it display on the front panel lights of the CPU. Several operating systems were modified to take advantage of this significant feature.

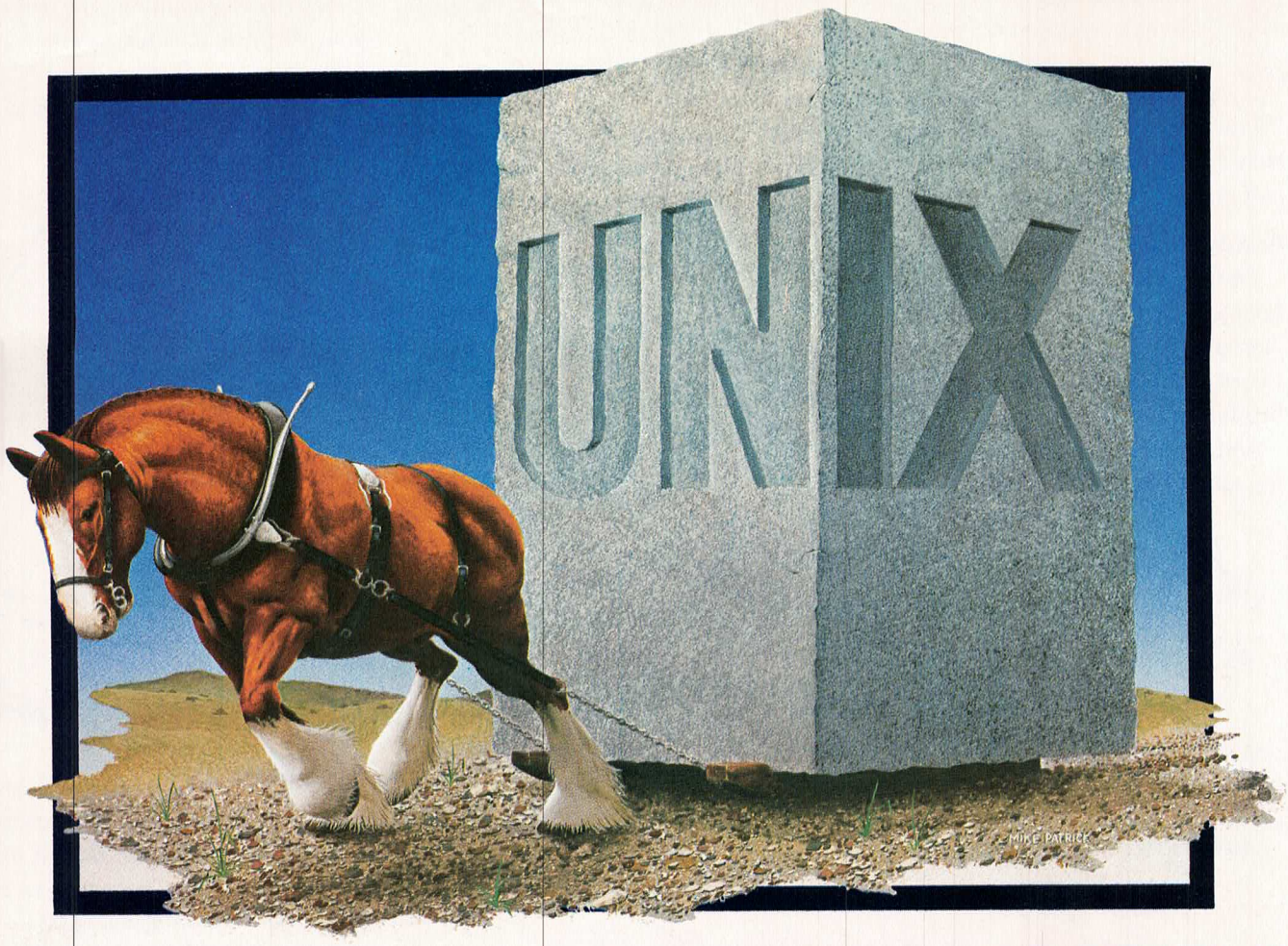
For example, it was common to put a moving "freight train"-type light display into the register from within the operating system itself. Not only could you tell the type of system running on the machine by glancing at the lights (since different systems tended to have different patterns), but you could even tell the relative system load by watching the speed with which the lights moved across the display. Who needed sophisticated system status programs when you had features like that?

But today, they've taken most of our front panel lights away from us. Now we're lucky to have a meager blinking disk select light where we can see it. Ah, the disks! Now, *they* were fun! Remember staring down through the transparent covers and watching those babies spin up a storm? You couldn't help but wonder what would happen if the disk bearing happened to break loose while you were watching, but that was part of the entertainment. They don't even give us clear glass covers anymore. Most of the disks nowadays have opaque covers, and many are completely sealed from human view or access. A true loss.

STRIKING SOUNDS

Even before the advent of complex music synthesizers, some people busily used computers and their pe-

Continued on page 127



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

Dealers— Please circle Ad No. 92 on inquiry card.
End Users— Please circle Ad No. 145 on inquiry card.

Just over a year ago, SofTrak Systems introduced MicroTrak scheduling software, and wrote the first chapter of a classic microcomputer industry success story. Now SofTrak has written a whole new plot for the MicroTrak story.

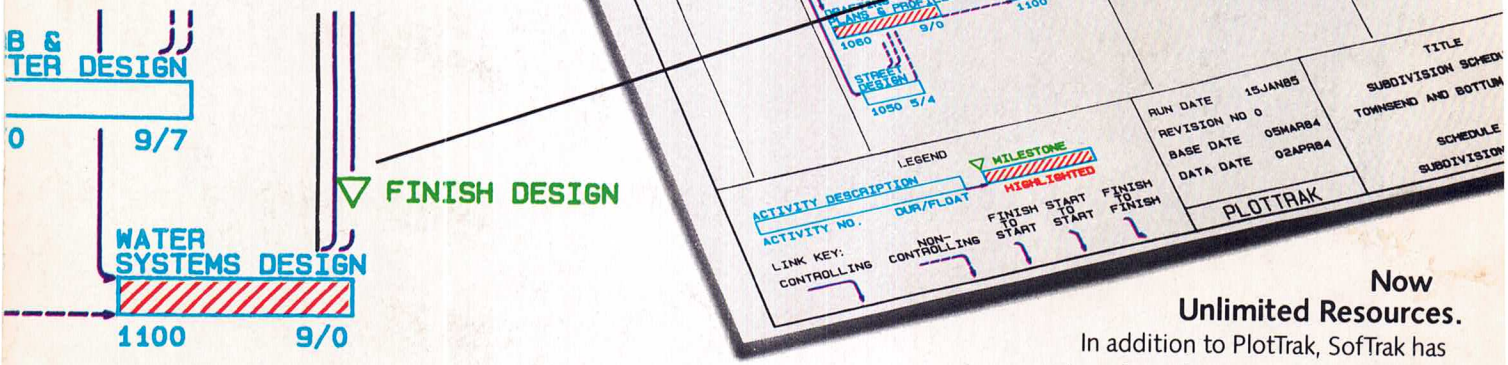
Announcing PlotTrak.

PlotTrak is the new plotter software from SofTrak. Using MicroTrak files, PlotTrak generates a colorful, easy-to-read schedule with many helpful features.

Activity Selection. With PlotTrak, any subset of activities in the overall schedule can be selected for plotting. Plus you can plot items before or after a given date, or items between two dates.

Plot Dimensions. PlotTrak lets you choose paper sizes from 8½ x 11" to 34 x 44," using a daily, weekly or monthly scale. So your plot can be practically any size or scope you need.

Activity Highlighting. Any part of your MicroTrak schedule can be highlighted with PlotTrak. And different pen colors can be used to emphasize specific activities, milestones, highlights and links.



SOFTRAK INTRODUCES AN EXCITING NEW PLOT FOR MICROTRAK.

Now Unlimited Resources.

In addition to PlotTrak, SofTrak has also released a new, upgraded version of MicroTrak featuring unlimited resources. Now MicroTrak can utilize any number of resources for inclusion in your schedule.

Continuing Innovation.

The introduction of PlotTrak and unlimited resources for MicroTrak begins a new chapter for SofTrak Systems. With continued innovation, SofTrak is setting a standard for MS-DOS* and Unix*-based project management software.

FREE 2-WEEK TRIAL.

We want you to give PlotTrak (and MicroTrak, if you're not already using it) the acid test. So we'll send you a diskette and give you two weeks to try it out. We think you'll like it. But if not you can send it back with no obligation. Interested? Call us today:

1-801-531-8550

SOFTRAK

SofTrak Systems, 1977 West North Temple
Salt Lake City, Utah 84122

Please circle Ad No. 100 on inquiry card.

*MS DOS is a registered trademark of Microsoft Corporation. *UNIX is a trademark of AT&T Bell Laboratories.