FROM THE EDITOR
The World Just Keeps Getting More Complicated... 3

USENIX NEWS
President’s Letter ........................................... 4
LISA VIII Conference Reports .................................. 5
Summer ’94 Conference Reports
by JR Oldroyd ............................................ 11
Community News ........................................... 12
Fun Things to do in New Orleans .......................... 12
USENIX Board Meeting Summary
by Ellie Young ............................................ 13

SAGE NEWS
SAGE 1994 System Administrator Profile and Salary Survey
by Elizabeth D. Zwicky & Zanna Knight ............. 16
SAGE Local Group Support
by Elizabeth D. Zwicky .................................. 19
Perl Practicum: A Plea for Clarity
by Hal Pomeranz ......................................... 20
Point/Counterpoint
by George Spielvin & Hal Pomeranz .................... 21
Wherever You Go, There You Are
by Elizabeth D. Zwicky .................................. 23
System Administration Tools Your Vendor
Never Told You About: The Toy
by Elizabeth D. Zwicky .................................. 24
System Administration Tools Your Vendor
Never Told You About: Painkiller
by Hal Pomeranz ......................................... 24

FEATURES
Bash – The GNU shell by Chet Ramey .................... 25
Business Survival Skills 101: The Interview
by Scott Hazen Mueller .................................. 33
Taking Analogies to the Limit
by Michael Chastain ..................................... 36
Results of the International Computer Olympiad
by Don Piele ............................................... 37
Edupage – The News ...................................... 38

STANDARDS REPORTS
An Update on Standards Relevant to USENIX Members by Nick Stoughton ......................... 44

BOOK REVIEWS
The Bookworm by Peter H. Salus ......................... 49
Application Development for Distributed Environments – Reviewed by Bob Birss ....... 51
ATM User-Network Interface Specification
Reviewed by George V. Neville-Neil .............. 51
Building a Successful Software Business –
Reviewed by Gregory C. Depp ......................... 52
Zen and the Art of the Internet, 3rd Edition –
Reviewed by Billy Barron ................................. 53
A DOS User’s Guide to the Internet – E-mail,
Netnews, and File Transfer with UUCP –
Reviewed by Billy Barron ................................. 54

ANNOUNCEMENTS & CALLS
USENIX 1995 Technical Conference .................... 55
Calls for Papers:
Symposium on Mobile and Location-Independent Computing – 1995 .............. 58
UNIX Security .............................................. 60
COOTS ......................................................... 62
LISA ’95 ...................................................... 64

ETC...
Book Publishers Discount Order Forms ................ 66
Membership Order Form .................................... 75
Local User Groups .......................................... 76
Calendar of Events ........................................ 78
SAVE THESE DATES

DECEMBER 8–9, 1994
WORKSHOP ON IEEE MOBILE COMPUTING SYSTEMS AND APPLICATIONS
Sponsored by IEEE TCOS in cooperation with ACM SIGOPS and USENIX
Santa Cruz, California
Program Chair: Darrell Long, University of California, Santa Cruz
Program Chair: M. Satyanarayanan, Carnegie Mellon University

MARCH 12–13, 1995
USENIX/SAGE TUTORIALS ON INTERNET SECURITY, UNIX POWER TOOLS, FIREWALLS, AND THE LAW & COMPUTERS at the Uniforum Exposition & Conference
Dallas Convention Center
Dallas, Texas

APRIL 24–29, 1995
4TH SYSTEM ADMINISTRATION, NETWORKING, AND SECURITY SYMPOSIUM (SANS IV)
Sponsored by The Open Systems Conference Board in cooperation with the USENIX Association's Special Technical Group, SAGE (System Administrators Guild) and The Washington Area UNIX Users Group
Washington, D.C.
Program Chair: Rob Kolstad, BSDI, Inc.

JUNE 26–29, 1995
USENIX CONFERENCE ON OBJECT ORIENTED TECHNOLOGIES (COOTS)
Monterey, California
Program Chair: Vince Russo, Purdue University
Tutorial Program Chair: Doug Lea, SUNY Oswego
Abstracts Due: March 6, 1995
Authors Notified: April 3, 1995
Papers Due: May 15, 1995

JUNE 5–7, 1995
5TH USENIX UNIX SECURITY SYMPOSIUM
Sponsored by the USENIX Association, in cooperation with:
The Computer Emergency Response Team (CERT), PGP WG 11.4, and Uniforum
Salt Lake City, Utah
Abstracts: February 13, 1995
Authors Notified: March 8, 1995
Papers Due: May 1, 1995

SEPTEMBER 18–22, 1995
9TH USENIX SYSTEMS ADMINISTRATION CONFERENCE (LISA IX)
Co-sponsored by SAGE, The System Administrators Guild
Monterey, California
Program Co-chairs: Tina M. Darmohray, Lawrence Livermore National Laboratory & Paul Evans, Synopsis, Inc
Abstracts Due: May 1, 1995
Papers Due: August 1, 1995

PHONE, FAX OR EMAIL US FOR MORE INFORMATION
Please contact: USENIX Conference Office
22672 Lambert St., Suite 613, Lake Forest, CA USA 92630
Phone: +1 714 588 8649; FAX: +1 714 588 9706; Email: conference@usenix.org
For more information about these events, access the USENIX Resource Center on the World Wide Web. The URL is http://www.usenix.org. Or send email to our mailserver at: info@usenix.org
Your message should contain the line: send catalog. A catalog will be returned to you.
FROM THE EDITOR

The World Just Keeps Getting More Complicated

I had one of those “once in a blue moon” experiences last week. A committee of the Colorado Springs Mountain Zoo held a meeting to which I was invited.

The ostensible purpose of the meeting was to work together as a large group of “high-power professionals” to brainstorm designs and ideas for the Zoo’s new “Discovery Center.” An organization with “Humanities and Arts” in its name had a large fund from which to disperse monumental sums of money for projects such as this. I was invited, I was told, to ensure that people did not run off from Star Trek technology, and create inventions whose complexity exceeds that of available technology.

Into the room I strolled. It resembled nothing more than a sixth grade classroom with 30 desks arranged in a horseshoe. Not surprisingly, the walls were decorated with pictures of animals, their keepers, and posters about zoo activities. This was the Zoo’s general purpose activity room, which is used by ages 3 to adult, and is booked solid throughout each day and all year.

Our moderator conducted the group through introductions. Seemed like 2/3 of the participants had Ph.Ds (or D.V.Ms). I noticed that people’s areas of expertise were in fields like anthropology and art history.

Off we went into the world of brainstorming and idea-sharing. I was intrigued to notice that there was no central visionary for this project. There was no Bill Joy or Steve Jobs driving the project from inception to completion. Our goal was to advise some smaller committee that would in turn assimilate the comments and create a proposal.

Imagine my surprise when I was unable to persuade the group to define the problem to be solved during our 8 hours at the zoo. I could not believe that this would not be the first priority. I guess it is like an engineering approach...

We then played a game like Animal Farm (in which each of the various animals wants to be King of the Barnyard). Do we organize the four sections geographically? By ecological niche? By time in history? The geographers insisted that geography is the center of all study, and that we should use their ideas. I am a bad sport in these situations, as it turns out.

By the conclusion, just about everyone felt the day was tremendously valuable. Four different groups designed four different “modules” for the new center – some with over a dozen different museum-style exhibits. After all, we have 60 minutes for the exercise.

I left with a new appreciation for the tremendous difference between the cultures of software engineers and system administrators, and the cultures of those in the more “academic” disciplines. I was not proud of my level of frustration or my internal feelings of intolerance.

I reminded myself, however, that the world is full of all kinds of people – even highly academically successful people – with different ways of approaching problems, creating solutions, and motivations for success. I reckon it was a good thing it happened sooner instead of later, after I moved into the world of the truly bigoted. I hope I’ll be able to work better in that kind of large group in the future. I have this idea that many of my customers (and others) might have unique approaches, too.

RK

USENIX BOARD OF DIRECTORS

If a member wishes to communicate directly with the USENIX Board of Directors, he or she may do so by writing to <board@usenix.org>.

President:
• Stephen C. Johnson <scj@usenix.org>

Vice President:
• Eric Allman <eric@usenix.org>

Secretary:
• Lori Grob <lgrob@usenix.org>

Treasurer:
• Rick Adams <rick@usenix.org>

Directors:
• Tom Christiansen <tchrist@usenix.org>
• Dan Geer <dgeo@usenix.org>
• Andrew Hume <andrew@usenix.org>
• Greg Rose <greg@usenix.org>

Executive Director
• Ellie Young <ellie@usenix.org>

Marketing Director
• Zanna Knight <zanna@usenix.org>

CONFERENCES & SYMPOSIA

• Judith F. DesHamais
  Conference Coordinator
  USENIX Conference Office
  22672 Lambert Street, Suite 613
  Lake Forest, CA 92630
  Telephone: 714/588-8649
  FAX: 714/588-9706
  Email: <conference@usenix.org>

• Peter Mui
  Vendor Displays
  Email: <display@usenix.org>

• Daniel V. Klein
  Tutorial Coordinator
  Telephone: 412/421-2332
  Email: <dv@usenix.org>

WWW URL: http://www.usenix.org

AUTOMATIC INFORMATION SERVER

To receive information electronically about upcoming USENIX symposia & conferences finger <info@usenix.org> and you will be directed to the catalog which outlines all available information about USENIX services.

SUPPORTING MEMBERS

ANDATACO
ASANTE Technologies, Inc.
Frame Technology Corporation
Matsushita Electrical Industrial Co., Ltd.
OTA Limited Partnership
Quality Micro Systems, Inc.
SunSoft Network Products, Sun Microsystems, Inc.
Sybase, Inc.
Tandem Computers, Inc.
UUNET Technologies, Inc.

The closing date for the next issue of <login> is December 14, 1994.
President's Letter
by Steve Johnson
<scj@usenix.org>

KNOCK KNOCK! WHO'S THERE?

I've just returned from a USENIX board meeting where, in a little more than 24 hours, we wrestled with our 1995 budget, staff evaluations, a major housecleaning of our bylaws, future conferences and workshops, and other fun topics. One of the most interesting discussions for me came up in our addressing projects that we might fund in 1995.

Some of the money that we generate through membership and registration fees is returned to the community through “good works” – projects that we feel will benefit society as a whole; so we are less concerned with the return on investment to USENIX. We sponsor a number of activities that involve students from the high school and college level, for example. When we were discussing 1995 projects, one of our board members proposed that we should devote a couple of thousand dollars to projects such as PGP (Pretty Good Privacy) public domain encryption and anonymous mail servers. This proposal was not a complete surprise, since some e-mail had preceded it, but the issue turned out to be much trickier than it first appeared.

Most of the discussion focused around the anonymous mail servers. There are now roughly 50 such, many in countries that have stronger privacy protection laws than the United States. The way this works is that you mail to someone and route the mail by way of one of these servers. When your mail passes through the server, the sender information is stripped off. In some cases, a pseudo sender name is created, and the original routing information is kept so that the receiver can reply to the pseudonym.

Like all new technologies, anonymous mail servers bestow benefits and also provide the opportunity for abuse. Several board members spoke of friends that had received support through anonymous recovery groups and gay support groups. Here, the anonymity allowed people to connect with others and share experiences without concern that they may be incriminating themselves or others. Whistleblowers, who may have become aware of illegal or unethical acts by their government or company, can make these acts known with little fear of reprisal by using anonymous mail servers. Since computers have attacked personal privacy in so many ways, many people felt it was a poetic kind of justice that, in this way, computers might help to preserve privacy as well.

But there is a darker side to this. Anonymity removes one of the basic freedoms in the United States, the right to confront your accuser. Hate-filled people have used these servers to attack others for their religion, politics, and sexual preferences and activities. In some cases these activities have taken a very “moral” tone, suggesting that people send e-mail to a particular (non-anonymous) individual telling them to clean up their (immoral) sexual activity, for example. Many people have attempted to suppress or eliminate such postings when they appear, raising the spectre of censorship. At least one board member spoke very strongly against the whole idea of supporting anonymous mail servers, feeling that the abuses were bad and getting worse, and the benefits were modest and could be obtained in other ways.

In the end, we followed the traditional route of committees and did nothing; the specific actions proposed did not look like they would make much difference to
the state of the world, while the issues raised were deeper than we could comprehend in a few minutes of discussion.

Running through our board discussions were frequent references to “what the members would want.” We could run a workshop, commission a pamphlet, lay a modest amount of money on somebody or something working in this area, or, as some people might prefer, stick to a technical charter and let somebody else do the “soft stuff.” Your suggestions, anonymously or not, can be mailed to board@usenix.org.

LISA VIII Conference Reports

Opening remarks
Dinah McNutt

Keynote Address: The Road to UNIX: A Report from the Fourth Estate
Jack Stanley, Houston Chronicle
Summarized by Robert K. Borowicz
<borowicz@ptsg-austin.sps.mot.com>

The keynote speech, as well as all the Technical Tracks of the 1994 LISA conference, were held in what was previously two 300-seat rooms used for the full-day tutorials. The large stage and twin video screens resembled a Rolling Stones concert rather than a UNIX conference. The growth of LISA was truly evident in the scope of its main meeting hall. In fact, in her opening remarks, Dinah McNutt noted that this was the largest LISA conference ever at 1250 attendees. Dinah gave well-deserved thanks to the USENIX personnel and to her program committee responsible for putting on this year’s conference.

Dinah outlined all the great activities that go on at LISA including the “the Guru is in” sessions, the BOFs, the SAGE board meeting, the poolside reception, and a mystery event Wednesday evening! Dinah then introduced Elizabeth Zwicky, President of SAGE.

Elizabeth presented the SAGE Outstanding Achievement award voted by the board. This year it was awarded to Larry Wall for his efforts with Perl. Attendees listened very intently as Larry accepted the award. Larry commented on his view of Perl and on the accolades he has been receiving on the net. Noting the discomforting referrals to himself as a god, Larry thought it appropriate to give us some sage advice. He urged us to be creative with the tools we’d been given, and called all System Administrators gods. Larry received a standing ovation as he stepped down and made way for the keynote speaker, Jack Stanley.

As Vice President of Operations at the Houston Chronicle, Mr. Stanley oversees over 600 users of Suns, McIntoshes, and PCs. A 35-year career in computing is special, if not rare, in this industry and always leads to truly legendary stories. Jack Stanley is no different.

What I found particularly exciting about Jack’s stories is the nature of his business, newspapers. Newspapers are a threatened media form in the ’90s, and Mr. Stanley illustrated some of the challenges the business faces today.

Mr. Stanley first reminded us why this was a report from the 4th estate. Historically, society has categorized the world’s population into three “estates:” the Clergy, the Nobility, and then all the rest, Commoners. So, we’re somewhere down the ladder of categorization according to Mr. Stanley, and I think we can identify with Mr. Stanley’s perspective.

In the newspaper business, Mr. Stanley supports the cliché of a reporter, the man in a suit, running for a phone to call in a story. With reverence, Mr. Stanley spoke of the “Pipe Organ” device reporters used as their first word processor, the Royal Typewriter. Newspapers also had the Linotype machine. Jack humorously referred to it as being a better editor than teco but not as good as vi.

One of the problems of automating a newspaper was dealing with the antique systems. At one point, the Chronicle upgraded its DEC 11/70s; they added memory. The problems of no interconnectivity and no growth path with proprietary systems were a tremendous issue. The task of pasting little pieces of film together was a daily miracle. In 1987, the newspaper had a vision: to use Open Systems, based in standards. It was a “1000 year solution.” The Chronicle set out to integrate the enterprise.

To implement the vision the newspaper began with Sun 3/60s and then 3/280s, and finally Sparcs. On the network front, they began dealing with the familiar yellow ethernet cable, 400 miles of it! We all laughed when Mr. Stanley mentioned how his vendor suggested putting all (several hundred nodes) on one single segment. Then there were fiber backbone subnets and switched eternets, and more subnets. It was “divide and conquer!” When it became available, the paper looked at NFS, “the great adventure,” Stanley called it.

Next came the task of training the users. What was this UNIX? It sure wasn’t MVS! And there were awk, grep, and vi, SunOS 3.2, 3.5, 4.0.3, UGH! And finally SunOS 4.1.1. The Chronicle developed its own software, Mr. Stanley went on, dubbing it the “Sleazy Adventure.” He used phrases like “joint development project” and “design by committee,” and “programming without standards” to which we all could relate. What became very clear was that these guys had done it all. Where there was no technology, they built it. Hardware and Software solutions that looked feasi-
ble were utilized. And lessons were learned. Mr. Stanley concluded that the future is still in the future for the newspaper business. It is clear that to survive in the '90s and beyond, there is at least one newspaper prepared to do what it takes to get there.

**System Configuration**
*Session chair: Mark Verber, Xerox PARC*

**Central System Administration in a Heterogenous UNIX Environment: GeNUAAdmin**
*Dr. Magnus Harlander, GeNUA mbH*

**Config: A Mechanism for Installing and Tracking System Configurations**
*John P. Rouillard and Richard B. Martin, Dept. of Mathematics and Computer Science, Univ. of Massachusetts, Boston.*

**Towards a High-Level Machine Configuration System**
*Paul Anderson, Dept. of Computer Science, University of Edinburgh*

Summarized by Douglas L. Acker
<acker@se01.wg2.waiti.com>

These sessions dealt with various systems in maintain system configuration, i.e., system files (aliases, hosts, etc.), home files (.kshrc, .xintrc, etc.) and quotas on file servers. The reason for the various systems is to avoid typing errors, to avoid inconsistencies, and to avoid forgetting different options/methods on different systems. Each system focused on a "niche" the authors felt was important. There is overlap but each system handles some issues uniquely or not at all.

**Purchasing a Workstation Installation**
*Peter B. Galvin, Brown University*

Summarized by Ruth Milner
<milner@oaac.nrao.edu>

In 1991, Brown University received an NSF grant to replace an installation of roughly 160 SPARCstation-1's and three servers. Peter's talk gave an overview of the stages in the process along with a number of very useful hints and tips for others having to go through it. This was not a government procurement, so (lucky for them!) they had to deal with fewer rules and restrictions, but much of the information is still relevant even to government sites. Having gone through this myself around the same time as Peter did, I enjoyed hearing about his experience.

The whole process, from initial preparation to delivery of the final pieces of equipment, took two years – not unusual for a procurement of this size. Peter described how Brown University did each of the steps, and in some places also talked about how, with hindsight, they would do things differently next time. He emphasized several points: a knowledgeable committee with good contacts is essential; you must promote your institution to get the vendors interested; technical requirements, evaluation procedures and acceptance criteria must be developed carefully, as well as specifying (and enforcing) realistic penalties for failure to meet these criteria; you must do real tests with relevant benchmarks on systems as close as possible to those bid (if at all possible, create your own benchmarks to reflect your environment); and above all, you must deal with the vendors firmly but fairly: you will almost certainly need to maintain a cordial relationship with them afterwards.

The overhead projections from Peter's talk are available on the Internet via URL http://www.cs.brown.edu/people/pbg

**Automation**
*Session chair: Hal Siern, Sun Microsystems*

Summarized by John P. Rouillard
<rouill@cs.umb.edu>

**OMNICONF – Making OS Upgrades and Disk Crash Recovery Easier**
*Imazu Hideyo, Matsushita Electric*

Imazu Hideyo presented OMNICONF, a process for making "OS upgrades and disk crash recovery easier." He provides a mechanism, written in perl, for storing and imposing configuration files on an installed system. Three main programs are in use. The first, mkprof, is responsible for taking a snapshot (profile) of the system as it stands when first installed. Using this snapshot, the program putconf generates a gnu cpio archive to store the changed files. Once this is done, the cpio file (configuration) can be imposed on the end system using the getconf program. In addition to local use, there is a server portion to this system that can allow the configuration information to be saved on, and retrieved from a remote server, thus protecting the configuration in case of a disk crash.

**Automated Upgrades in a Lab Environment**
*Paul Riddle, University of Maryland, Baltimore County*

Paul Riddle presented a method for automating upgrades in a lab environment. He said that this mechanism isn't limited to lab machines, and could be used for any group of machines with identical software configurations. By booting the workstations in a diskless mode, the local disks on the system are made available for repartitioning and new OS installation. The entire upgrade process is done by calling an upgrade script (written in perl) from /etc/rc on the diskless
Tenwen: The Re-engineering of a Computing Environment
Rémy Evard, Northeastern University

The last talk in this series had less to do with the configuration of a machine than it did with the configuration of a new network. Rémy Evard talked about building a replacement for the network that he inherited when he took his job. In his paper Rémy took us from the first two machines in the new network, through miles of unused and unconnected cables in the machine room to the migration from “oldnet” to newnet. In this journey, he discussed the policies and procedures that were needed for Newnet as well as the tools and user level improvements in service and reliability that went along with this conversion.

Are We There Yet? Evaluating the State of Your Site
Elizabeth D. Zwicky, Silicon Graphics
Summarized by Win Bent
<wbh@skat.usc.edu>

Part of Elizabeth Zwicky’s current job is to evaluate the sysadmin practices at various sites, and in this talk she presented the areas she looks at and what she considers bad, good, and great methods. Her goal is to produce a set of guidelines which result in “FDA-approved” administration: safe and effective.

The areas covered included backup, security, networks, support, applications (availability and support levels), information and resource sharing, maintainability and long-term viability, and human resource issues. In each area, she had about a half-dozen questions she asked of the sysadmins — this being a LISA conference, those of us listening were expected to ask ourselves! For example, in the area of Info and Resource Sharing, she asked: Can users share files easily? Do you have to know someone’s machine name to get files via NFS? In virtually every case, she gave her idea of bad, good, and outstanding answers (I love open-book tests!). One general principle was that more than one person should know what’s going on in complete detail, what I call the “run over by a bus” principle.

The talk was very well received, and people could be seen either smiling proudly or sighing and shaking their heads as they silently answered Elizabeth’s questions. What do YOU do for off-site storage of backups? As always, backups were a major topic in the question-and-answer session after the talk — is this because Elizabeth is a known expert on backup systems, or (perhaps more likely) because this still weighs heavily on our collective minds? Either way, this talk gave an excellent framework for evaluating a site, as well as pointers to making even a good situation better.

The Toolbox
Session Chair: Pat Parseghian, AT&T Laboratories
Summarized by John Rouillard
<rouilf@cs.umb.edu>

Kernel Mucking in Top
William LeFebvre, Argonne National Laboratory

Bill LeFebvre discussed the much-used and very useful monitoring program called top. Top produces much the same information as the Berkeley style ps program, but it updates the information at regular intervals, and show the process using the most cpu. To get this information, it needs to access the internal kernel data structures. Since top runs on 18 different types of OS’s, some mechanism was needed to segment the platform independent parts of top from the OS dependent parts, otherwise the resultant code would become spaghetti. Bill’s solution was to define a set of functions that would be provided to give access to the various kernel parameters that were needed. These “modules” as the function implementations are called allow drop in support for new operating systems simply by writing a module. Bill doesn’t have access to most of the systems on which top runs, he merely gets donated modules from good samaritans on the network.

Handling Passwords with Security and Reliability in Background Processes
Don Libes, National Institute of Standards and Technology

When running background processes that require passwords, you have two choices, store the unencrypted password in a file, or pass the password as a command line argument. Don is the author of the expect tool. He has another mechanism using expect. This uses an expect script to prompt for the password, and then supplies it when the application requests it at some later date. This mechanism is superior to others, for its password only exists in process data space which is more secure than filesystem space, and the password can’t be seen using ps. Don also talked about a number of other uses for expect (and the kibitz expect script) when running programs out of cron or at. There are times when these programs need password prompts, and expect/kibitz allows you to provide the passwords without resorting to risky mechanisms such as passing passwords on the command line.
Network Security Fun

Bill Cheswick, AT&T Bell Laboratories
Summarized by Robert K. Borowicz
<borowicz@ptsg-austin.sps.mot.com>

I have always heard that it’s a good idea to get the audience’s attention when giving a talk. Bill Cheswick chose a unique method for getting our attention at his LISA 94 Invited Talk. He got the power to fail ten minutes into his presentation. Two hours later, the power returned (as did we).

He showed a time-lapse film he made from a vacation at the beach with his family. The video showed Bill and his young son building a sand castle perilously close to the waves of the ocean. As they finished piling the sand on, the tide came in and washed the castle away. His point was that your networks are like this: “Sometimes they die a natural death, and sometimes teenagers come by and kick them over.” Then he gave some highly valuable investment advice based on data he’d compiled. He put up charts of year-long earnings curves of important and not so important companies. He showed us growth curves of Cisco, Wellfleet, and Cabletron and then some simple numbers on Internet growth. We got the picture. He echoed what I’d heard before, the Internet is a new frontier, and as in all new frontiers like the wild west, there are bad guys out there. Maybe I’ve seen too many movies, but I wanted to go get my six gun and strap it on!

Bill then discussed several simple ideas, some or all of which are contained in his book, Firewalls and Internet Security, Repelling the Wily Hacker. He showed us firewall concepts, simple in nature and absolutely essential today. He made a hilariously pragmatic statement on perimeter security: “I have no idea how secure 40,000 nodes are inside AT&T. However, I have a much better idea how secure one is.”

He described the concept of a secure network as that of a piece of candy, one with a soft chewy inside but hard and crunchy outside. In one moment of hysteria, he alluded to the gleeful abandon of being on the internet: “We’re so glad the dog is dancing, we fail to notice that it is rabid.” The room fell apart laughing.

Bill introduced us to the notion of the “doorknob twister.” He may not get in, but he’s there twisting the doorknob. He showed his /etc/inetd.conf file from his SGI machine, pointed out all the daemons run by root in it, and posed the question, “why do they need to run as root?” He wasn’t sure, therefore he shut them off.

Back to the doorknob twister, Bill suggested another interesting idea. One of the most puzzling and frustrating points of network security is that you never know who is out there. Knowing what you’re protecting yourself against helps to prepare you for the task. Bill suggested putting a “Honeypot” or interesting sounding machine out on your network and modifying all of the transport services to be logging services instead. Rather than get the real telnet service, for example, yours would be a very good audit trail back to a logfile or even an Alpha-numeric pager. All of these ideas were simple in concept and all too-often talked about in today’s UNIX world. However, Bill presented them in a humorous and entertaining way, with a serious warning to all in attendance: “If you’re not doing something about network security, you need to be and right now.” I think we heard the message and laughed while we cried.

Software Configuration
Session chair: Paul Evans, Synopsys, Inc.
Summarized by Paul Anderson
<paul@dcs.ed.ac.uk>

Depot-Lite: A Mechanism for Managing Software
John P. Rouillard and Richard B. Martin, University of Massachusetts, Boston

John Rouillard presented a variation of the familiar Depot system which is being used for managing software packages at the University of Boston. In “depot-lite,” the filesystem layout has been changed to group together all files for a particular platform. This allows the system to be adopted more easily by small clusters which do not use an automounter. Software installation is also simpler and some support is provided for multiple simultaneous versions.

Beam: A Tool for Flexible Software Update
Thomas Eirich, Univ. of Erlangen-Nürnberg, IMMD IV

Thomas Eirich described a tool called “Beam” for updating local copies of software packages from a master server. This program updates by “pulling”, using NFS to access the master copies of the files. Beam is written in Perl and is highly configurable, but it appeared to add little to previously reported systems such as “Ifu” and “Nightly”.

Soft: A Software Environment Abstraction Mechanism
Remy Evard and Robert Leslie, Northeastern University

This paper won the award for best student paper. It describes a higher level configuration language for specifying a user’s preferred environment. A program called “Soft” generates startup scripts, for different shells, from the same configuration specification, and these scripts can be cached to improve shell startup times. Changes in the details of a package installation do not normally require changes to the user configuration files and new startup scripts are generated automatically.
The User Environment
Session chair: Trent Hein, XOR Networking Engineering
Summarized by Paul Anderson
<paul@dcs.ed.ac.uk>

**Speeding up UNIX login by Caching the Initial Environment**

*Carl Hauser, Xerox PARC*

Carl Hauser described the Xerox system for configuring the user environment. This is based on a mechanism similar to Furlani's Modules, but the resulting environment is cached to avoid the delays associated with enabling lots of packages. A background process is forked to recompute the environment cache so that logins are always fast at the expense of a slight delay in changes to the configuration.

**The BNR/NT Standard Login**

*Christopher Rath, Bell Northern Research*

The "BNR Standard Login" is a yet another scheme to provide a method for users to configure their environment without the maintenance problems associated with complicated "dot" files. This system has no cache but uses a dedicated C program to interpret the configuration, and provides persistent environment variables for applications.

**Exporting Home Directories on Demand to PCs**

*David Clear, Alan Ibbetson, University of Kent-Canterbury, and Peter Collinson, Hillside Systems*

The final paper of these sessions described some modifications to the PC-NFS daemon which is used to export UNIX home directories to large numbers of PCs across the campus at the University of Kent. The lack of a PC automounter is overcome, and security is improved, by having the pam-fsd locate the appropriate server and authorize it to export just a single home directory to the appropriate PC.

**Building a Successful World Wide Web Server**

*Amy Kreiling, University of North Carolina*

Summarized by Bruce Hamilton
<bruce_A_Hamilton.LAX1B@xerox.com>

Due to vagaries of scheduling, Kreiling's talk was actually a follow-up to her WWW BOF rather than a prelude to it. Kreiling began by offering one hundred hardcopies of her slides, and the crowd descended in an enthusiastic feeding frenzy. By the time you read this, the slides should be available under http://www.sage.usenix.org.

The talk opened with an overview of WWW information resources. Recalcitrant managers can often be persuaded to fund WWW access if you can show them how WWW will meet their specific information needs, such as stock quotations or vendor catalogs.

Next was a review of the four major http servers: NCSA, CERN, GN, and Plexus. Only the CERN server (ftp://info.cern.ch/pub/www/src/) works across a firewall. A Pentium or SPARC 10 should be able to handle roughly 20K http requests per day, according to a table of activity versus processor power which was shown. According to Kreiling, "Webmaster" (server admin) and "Docmaster" (document admin) should be separate jobs, since each can be a heavy workload.

Finally, Kreiling gave an overview of html, html converters, and html editors. She encouraged audience interaction, and gave some pointers to resources. Unfortunately, her home page http://www.cs.unc.edu/~kreiling appears to be behind a firewall, because this reviewer is unable to access it from either Mosaic at Xerox or Netcruiser at Netcom.

You can subscribe to <www-managers@lists.stanford.edu> by mailing to <majordomo@lists.stanford.edu> and putting "subscribe www-managers" in the message body. WAIS-searchable archives of that list are on URL http://www-archive.stanford.edu/mail-archs.html

**The Automation Revolution**

Session chair: Tom Christiansen, Consultant
Summarized by Bruce Hamilton
<bruce_A_Hamilton.LAX1B@xerox.com>

**Automating Printing Configuration**

*Jon Finke, Rensselaer Polytechnic Institute*

Finke's second paper at this LISA applies an RDBMS to yet another problem, generating /etc/printcap for any of several hundred printers.

Source and documentation for the "Printmaster" suite of programs is available under URL ftp://ftp.rpi.edu/pub/lisastate-release/papers. Future directions will be to automate other configuration files such as crontab.

This paper and the needs that inspired it make this reviewer realize how deeply flawed the UNIX printing model is. Why should there be a static "printcap" file on each machine, as opposed to a name service that enumerates network printers, which in turn tell you their properties when you ask?

**Highly Automated Low Personnel System Administration in a Wall Street Environment**

*Harry Kaplan, Sanwa Financial Products Co.*

Kaplan described the development of a system of monitoring utilities combining various commercial and other software to do automatic system recoveries when possible and...
otherwise to notify Sys Admins via pagers. The talk was a fascinating description of the opposing tensions in a real-time trading environment. The SEC wants security, but traders are far more concerned with availability and ease of use. Traders complain when a workstation seems to freeze up for several minutes doing a core dump, but they also complain if you pop up status messages on their displays.

**The Group Administration Shell and the GASH Network Computing Environment**  
**Jonathan Abbey, University of Texas, Austin**

Abbey's paper won the "best paper" award.

The main purpose of GASH is to permit a much finer-grained security model than standard UNIX, so that multiple administrative groups can control different data, even on a single machine. NIS and NIS+ do not have this capability.

Future projects may include using an object database to simplify replication of information, and validating accounts against a personnel database.

Someone in the audience mentioned that MIT/Athena has a similar tool to GASH, but the speaker was not familiar with it.

The GASH software and documentation are available via [http://www.arlut.utexas.edu/ucsd/gash_docs/gash.html](http://www.arlut.utexas.edu/ucsd/gash_docs/gash.html). For information on a GASH mailing list, write to <gash-authors@arlut.utexas.edu>.

**Auspex Users' BOF**  
**Summarized by Ruth Milner**  
<rmliner@aoc.nrao.edu>

About 80 people crowded into the Chamber Room Tuesday evening for the Auspex Users' BOF. The discussion began with a Q&A session based on a list of fourteen questions collected ahead of time from Auspex customer sites and given to Auspex, so answers could be prepared. These ran the gamut from straightforward technical questions to future plans and the availability of new features. There was also a good discussion on how customers are doing backups, and Auspex's plans for providing procedures to make this simpler and more robust. Support for Solaris 2 clients was also discussed. A growing number of Auspex sites have Solaris running on some of their systems, others must wait for key application vendors to finish porting. A detailed report on the Q&A will be posted to the Auspex list out of Princeton; write to <auspex-request@princeton.edu> for info.

This year's discussion was somewhat more confrontational than last year's discussion. The most heated issues were product lifetimes and the pricing of hardware upgrades. While these issues aren't completely resolved, Auspex was willing to listen to customers' arguments (e.g., Auspex's equipment is a larger investment than other vendors' and lifetime is therefore a more critical issue) and is still discussing the policy internally.

**Women in Systems Administration BOF**  
**Co-ordinated by Vicki Brown, Caltech**  
**Summarized by Ruth Milner**  
<rmliner@aoc.nrao.edu>

This BOF was the first of its kind at a LISA conference. A head count early in the BOF revealed twenty-five women and thirteen men in the room. This fluctuated somewhat as time went on.

While no particular conclusions were reached, a number of issues were discussed, including the career paths women had taken into System Administration, and whether there was a "glass ceiling" preventing advancement beyond a certain point. Most people felt that any such barrier had more to do with professional orientation than gender, although there were clear exceptions. One interesting topic was the higher proportion of women at LISA compared to other technical conferences, and whether this is true of SA generally. Several women commented on the reduced proportion of women in undergrad CS programs and applicants for job openings in computing, and what the reasons for this might be (self-esteem/confidence? real obstacles?). Most people felt there is salary equity in the SA field, but without specific knowledge of other people's salaries it is impossible to be sure. Perhaps the SAGE questionnaire will help answer this question.

The difference in communication styles of men and women was a hot topic. A number of people — men as well as women — recommended the book *You Just Don't Understand* by Deborah Tannen, which covers this subject very well. Several women commented that team dynamics and social interaction with their co-workers were where they felt these differences most keenly. This led to the issue of sexual harassment in the workplace. It appears that some organizations are handling this poorly, with the result that men may become wary of saying or doing anything that might conceivably be misconstrued — even to the point of not asking a woman to join a group of her peers for lunch. This is a delicate issue right now; hopefully things will stabilize as awareness becomes more natural rather than a legal mandate.
A SAGE BOF mailing list will be created to discuss these issues further (sage-bof-women, I believe).

**DCE’s Impact on System Administrators**
*Rich Salz, Open Software Foundation*
*Summarized by Douglas L. Acker*
*acker@se01.wg2.waii.com*

DCE is a "somewhat unique" multivendor collaborative effort to provide an infrastructure for secure heterogenous computing. It is accomplished via:

- authenticated RPC
- scalability to large environments (30-40K users)
- multiprotocol

DCE’s origins are in Apollo’s Domain O/S’s Network Computing Service (NCS). It’s still an evolving/maturing system whose impact on system administrators is severe: more subsystems to manage and bugs to work out. The tools are evolving, too.

**Summer ‘94 Conference Reports**
*by JR Oldroyd*
*jr@opal.com*

**Jeff Haemer: A New Object-Oriented Programming Language: sh**

Jeff has developed a set of shell scripts that enable the creation and manipulation of arbitrary object hierarchies. Three basic scripts, “new”, “send”, and “destroy” are used to instantiate and communicate with instances of objects. Object classes are defined using further scripts that entirely encapsulate the object’s characteristics and access methods.

As an example, Jeff showed his “animal” script that defines an animal with the ability to set and determine its name, its favorite food, and the sound it makes. Using this, we saw Pooh Bear’s and Bugs Bunny’s favorite foods, and we had a bunch of dogs barking at each other!

As a more practical example, Jeff illustrated the use of his scripts to define a simple Turing machine object. He has also implemented an object that supports a method to send itself to another instance of the object: a virus.

The basic lesson is that, with a handful of small scripts, and nothing more than a POSIX-compatible shell, it is possible to build an object-manipulation engine. By Jeff’s own admission, the system is somewhat impractical due to serious speed limitations, but it could be used to serve as a tool in teaching basic object oriented concepts.

**Dan Geer: Breaking into Banks**

Dan’s invited talk was a repeat of one he had presented at the SANS III conference earlier in the year. Based on his work as a consultant to major banks and investment houses on Wall Street, Dan was able to observe, consider, and, ultimately make recommendations to replace systems and methodologies used in implementing system and network security.

The talk does not really explain how to break into a bank. Instead, Dan summarized some common problems he had encountered, problems that in some cases had lead to losses of millions of dollars. Even if you are not an investment banker, many of the same principles may apply to you.

Security violations occur because of several things. Human stupidity is one common cause. Too-simple passwords, or none at all, allow intruders in. In one case, Dan reported on a bank in which the “way-in” was not protected at all; it was just hard to find. His lesson here was “Obscurity is not Security.”

By far the most important message of the talk was Dan’s observation that most security violations originate internally. Dan gave example after example of this: from staff mis-interpreting an order to “buy one million” as meaning “$1M worth,” rather than “a million shares worth,” to staff being able to access restricted areas through inadequately protected service elevators. His lesson? Fix internal security and you’ve gone a long way to ensuring effective external security.

Over and over I found myself thinking who could possibly be so stupid as not to see the potential risks in the situations Dan described? At the same time, I also found myself thinking that these scenarios are blatantly obvious when described in a humorous presentation such as this, but are invisible when they’re under your nose and no-one’s pointed them out. If you are concerned about security, it probably pays to bring in an independent consultant to look for those “blind spots” for you.

**Berry Kercheval: ATM**

Berry gave a useful overview of ATM, aimed at the novice, or at least someone with communication and protocol expertise who is unfamiliar with ATM.

The talk started with an explanation of how optical fiber works, and what problems are encountered in attempting to
signal at high speeds over fiber. We’re talking T1, and up, here: 51-622 Mbps being the target speeds.

SONET (Synchronous Optical NETwork) framing techniques were discussed which led to a discussion of small, fixed-size cell units. This is where ATM fits in: ATM (Asynchronous Transfer Mode) is a switching protocol designed to move small (53 octet) data units at high speed. Berry explained how this is done, and described the content of the ATM cell header and what each of the fields is used for.

He moved on to a discussion of switch architectures, and gave examples of the common switches, including some examples of deployment, both in private use, and availability from the public carriers.

The final part of the talk was a discussion of the ATM reference model, and how adaptation layers (AALs) are used to map higher-level protocols, such as IP, onto a sequence of ATM cells. Some of these mappings are documented in RFCs, such as RFC1577 and RFC1483.

All together Berry’s talk was a good intro to ATM for the data-comm person. If you didn’t know what a packet was, you’d have been better off elsewhere.

**John Lin: Probing TCP Implementations**

Douglas Comer (author of probably the most well-read text on TCP/IP) and John Lin’s paper discussed a set of tools for determining operational characteristics of a TCP/IP implementation from an active system.

Their tools measure several things:

• TCP retransmission timeout intervals

• TCP keep-alive mechanisms

• TCP zero window probing

The goal of these experiments was to determine by practical observation of performance how various TCP/IP implementations performed in real use, and to see if performance improvements could be suggested.

The tools operate by monitoring TCP traffic to and from hosts. By injecting fake TCP packets on a network destined for the host under test, it is possible to observe its behavior.

Systems tested were Solaris 2.1, SunOS 4.1.1, SunOS 4.0.3, HP/UX 9.0, and Iris 5.1. The talk showed several graphs of observed behavior, which are duplicated in the proceedings.

In one case (Solaris), the test unearthed an operational error with the TCP implementation.

**Community News**

**David B. Wollner <dwollner@ucsd.edu> announces:**
It’s a boy! **Bentzion David Wollner** arrived on Monday, 9/26/94, at 9:48 am. Vital statistics are 19"., 6lbs. 5 oz. Marcia and Bentzion are both feeling wonderful. Adina and Margalit are thrilled to have a baby brother.

**Scott & Jackie Menter** are thrilled to announce the birth of our son, **Aviel Natan Menter**, on October 4 at 1:19 pm. Aviel weighed in at a hefty 7 lbs. 5 oz. The Brit Mila (circumcision) was held at their house on Tuesday, October 11th.

**Fun Things To Do in New Orleans**

We hope you’ll join us at our 1995 Technical Conference in New Orleans on January 16-20, 1995. See details on the program on page 551.

When you need a break from tutorials, BOFs, invited talks, WiPs, the Vendor Display, and the terminal room, New Orleans has an incredible variety of things to do and see just steps from the hotel. New Orleans is truly a city that never sleeps. The music never stops, you can always get something fabulous to eat, or purchase a tee shirt. Here’s the short list of spots you won’t want to miss. Most are located in the French Quarter, at the heart of the city, where our headquarters Hotel is located.

**What to See and Do in New Orleans**

If New Orleans is known for anything, it’s the music and night life. It’s home to Fats Domino, Wynton Marsalis, and the Neville Brothers, just a few of its better known natives. Music can be heard in the spots listed below, or you can just walk through the French Quarter and listen to street entertainment as the spirit moves you. Many places stay open until the early morning hours.

• **Tipitina’s.** You can hear live entertainment seven days a week, featuring the best of local and regional rhythm & blues, jazz, Cajun, reggae, and rock ‘n roll.

• **Preservation Hall.** The city’s most famous traditional jazz venue offers a different group each night, all masters.
• **544 Club.** Authentic and unforgettable, this club features veteran local musicians performing jazz, R & B, and New Orleans Jam sessions.

• **Maple Leaf Bar.** Open seven nights a week, and acts change every night. Catch Cajun, Creole & Caribbean rhythms.

• **Palm Court Jazz Cafe.** Besides offering tasty local dishes, the Palm Court features jazz seven times a week and offers hundreds of rare recordings for sale.

• **Mid-City Lanes Rock’n’Bowl.** Defying comparison, you can listen to live music and try local specialties between strikes.

Besides its jazz heritage, New Orleans is also known for its rich Creole and Cajun cooking in local dishes such as crawfish etouffee, gumbo, jambalaya, andouille sausage, and red beans and rice. Creole refers to descendants of early French or Spanish settlers while Cajun refers to the descendants of French colonists (Acadians) of Nova Scotia and New Brunswick who were expelled by the British in 1850 and found their way to Southern Louisiana. Creole cooking features more sophisticated, spicy sauces, while Cajun cooking is more hot-peppery and robust. The two terms are sometimes used interchangeably. No matter what you call it, the local cooking is delicious but not for the faint of heart. Here are a few spots to sample it.

• **Pat O’Brien’s.** There’s always a crowd here for fun and beverages. It’s the home of the renowned Hurricane.

• **Emeril’s.** This restaurant was named the best new restaurant by Esquire Magazine in 1990. While not new, it’s still good.

• **K-Paul’s Louisiana Kitchen.** Paul Prudhomme is the Paul in “K-Paul.” He introduced America to Cajun cooking.

• **Café du Monde.** No visit would be complete without a stop at the Café du Monde for beignets and café-au-lait. Open 24 hours, this is the original French Market coffee stand, and the place to stop on your way home from a night on the town.

• **Arnaud’s.** Besides excellent food and a superlative crème brûlée, the mosaic tile floors, ceiling fans and leaded glass windows recall a New Orleans of another era. You can also visit its Mardi Gras museum.

• **Olde N’awlins Cookery.** Zesty local cooking in an informal atmosphere makes this an easy choice.

• **Central Grocery.** You can’t go to New Orleans and not have a muffuletta sandwich; here you get one of the best.

• **Mother’s Restaurant.** If homemade biscuits and grits are what you want, you won’t find them any better.

• **Felix’s Oyster Bar.** Known for oysters on the half shell, but you can find any kind of seafood that you want.

Besides eating and drinking and listening to jazz, New Orleans has a number of unique museums. Here are just a few.

• **New Orleans Historic Voodoo Museum.** Tourists can glimpse voodoo as it is practiced at this museum. It includes a working altar and offers ritual and voodoo walking tours of the French Quarter.

• **New Orleans Jazz Club Collection.** The Jazz Collection tells the story of New Orleans’ music through photos, memorabilia, and instruments owned by musicians such as Louis Armstrong.

• **New Orleans Pharmacy Museum.** The building that houses this museum was built in 1823 for America’s first licensed apothecary. The collection includes hand-blown apothecary jars containing crude drugs and medicinal herbs.

• **Louisiana State Museum’s 1850 House.** This authentic recreation of an ante-bellum townhouse is in the Baroness Pontalba’s famous row house on Jackson Square.

• **Aquarium of the Americas.** This state-of-the-art museum provides unique views of the aquatic worlds of the Caribbean, Amazon Rain forest, Gulf of Mexico, and Mississippi Delta.

---

**USENIX Board Meeting Summary**

*by Ellie Young*
<ellie@usenix.org>

Below is a summary of the actions taken at the regular quarterly meeting of the USENIX Board of Directors which convened in Chicago, Illinois on October 11, 1994.

Attendance: Rick Adams, Eric Allman, Tom Christiansen, Dan Geer, Lori Grob, Andrew Hume, Steve Johnson, Greg Rose, Ellie Young, Judy DesHarnais, Diane DeMartini, Dan Appelman, Elizabeth Zwicky
Budget, Fees, Proposals

Johnson gave an overview of the budget process and noted that while our ability to make projections is better than in the past, it is still difficult to estimate the income generated by the conferences. We budgeted conservatively for overall attendance in 1994, but a number of factors influenced the final figures. Attendance at the tutorials at both general conferences was lower than in the past; overall attendance at the Winter '94 general conference and at the Applications Development symposium was lower than projected; penalty fees were paid for not meeting our room block and cancelling a future booking with the San Francisco Hilton Hotel. On the other hand, additional income was realized from the tutorials offered at UniForum; we experienced some savings in products and projects expenses; and the LISA conference generated more income than expected due to the high percentage of people taking tutorials. Young explained that, depending on attendance at the remaining two events this Fall, USENIX might achieve a surplus of approximately $150,000, which will then be deposited in the Association's Endowment Fund.

The assumptions behind the draft 1995 budget were discussed in depth, particularly the costs for member services, which are subsidized by the income from the conferences. The following funding and fee proposals were discussed and approved:

Membership Dues

Dues were last raised in October, 1992. It was agreed to increase the following classes of dues:

• Individuals: from $65 to $70
• Students: from $20 to $25
• Educational: from $160 to $175
• (Supporting, SAGE, and Corporate dues will remain the same).

Conference Fees

These fees were last increased in 1993. For 1995, tutorial fees at the general conference and symposia will be increased by $20 to $295 for one day and $540 for 2 days; three day technical session registration fees will be increased by $25 to $320 for members and $400 for non-members; and two day technical session registration fees will be increased by $25 to $300 for members; $380 for non-members.

Standards

A proposal to continue funding the USENIX Institutional Representative to the IEEE POSIX and the Standards Report Editor for 1995, with Stephen Walli replacing the outgoing Jeff Haemer as IR, and Nick Stoughton continuing in his role as Report Editor was accepted. Hume and Geer agreed to work on developing a proposal for providing snitch-like reports on X/Open activity, and funds were also allocated for this.

Good Works

A proposal from Kolstad to continue funding the International Computer Problem Solving Competition, and to also fund the US International Computing Olympiad's training camp and travel expenses to have students attend the annual International Olympiad in Informatics was approved.

A proposal from Rose regarding funding Internet-Related activities (to help support Jull Helsingius' anonymity server and to donate funds to Phil Zimmerman's voice encryption project) were discussed. It was agreed that the first will not be funded, and Rose was asked to provide more information concerning the latter project and resubmit a proposal.

The Board was interested in having Young negotiate and present a fuller proposal to fund Carl Malamud's Internet University to cover some of the USENIX conferences.

Expenses Paid by USENIX

It was agreed to change the policy concerning program speakers: For events with an anticipated attendance over 300, one refereed session speaker per paper will receive a complimentary registration.

A new remuneration schedule for tutorial instructors was accepted:

• "New" Instructors (1-3 classes taught) will be paid $2,000 for a full day class or $11/head, whichever is larger.

• "Veteran" instructors (> 4 classes taught) will be paid $3,000 for a full day class or $15/head, whichever is larger.

The SAGE budget (which was approved by the SAGE board at its September meeting) was approved.

Bylaws

Johnson explained that the bylaws had not been reviewed or revised in a very long time, and many statements were no longer accurate. It was also suggested that since revisions
require a two-thirds vote by the Directors and subsequent approval by the membership (which is costly since it involves a first-class mailing to the membership), it was agreed to limit the amount of detail that is put into the bylaws and augment them by referring to the policies document for these details. An overview and discussion of the changes described in the memo from the subcommittee ensued for several hours. Several issues concerning election reform were debated (i.e., raising the number of signatures required to nominate from the floor; whether nominating signatures should be made public and whether the type of nomination should be disclosed on the ballot; and specifying how ballots should be counted.) A revised draft of the bylaws will be presented to the Board at its next meeting.

**Electronic Commerce Workshop**

A proposal from Geer to organize this was accepted. He, Allman, and Young will work on this.

**WWW/Internet Workshop**

It was agreed that we should attempt to do something, but positioning this to be technically interesting and non-competitive with other conferences will be difficult. Young will work with Geer and others to see what might be done.

**1996 Technical Conference**

The proposal from Bob Gray to serve as program chair for the Technical Conference held January 22-26, 1996 in San Diego, was accepted.

**Embedded Systems**

Grob will submit a formal proposal to serve as program chair for a workshop on this topic.

**Free Software/Linux workshop**

Tom Christiansen will look for a program chair to make a proposal on this topic.

**EurOpen**

Zwicky will be attending the EurOpen governing board meeting convening in November in Romania as the USENIX/SAGE representative.

**Marketing Director’s Report**

Knight was instructed to proceed with more surveys of our members and attendees (both current and former) to obtain additional data on which topics should be offered at conferences, and to have more demographic information on our attendees in order to better promote the vendor displays and general conferences. She will also develop a formal proposal to offer member discounts with a computer/peripheral mail order catalog. She will also investigate whether any of the net providers can offer member discounts for service.

**Next Meeting**

It will be held in New Orleans on January 15 and 16, 1995.

---

**DILBERT® by Scott Adams**

I THINK IT WAS FIFTY GIGABITS.

I THINK YOU MEAN MEGABITS.

HA HA HA

HOO-HOO-HA

HA SNORT GIGGLE

HEE-HEE

WE’RE SO FUN-LOVING.

YOU’D THINK ONE OF US WOULD HAVE A FRIEND OUTSIDE OF WORK.

© 1994 United Feature Syndicate, Inc.
SAGE 1994 System Administrator Profile and Salary Survey

by Elizabeth Zwicky, SAGE President, and Zanna Knight, USENIX Marketing Director

As part of our ongoing effort to gain recognition and advancement for system administrators, SAGE conducted a System Administrator Profile survey in September 1994 at the LISA conference in San Diego. 909 participants responded.

This data does not necessarily reflect system administration as a whole, but it should be representative of the LISA attendees. Although 909 surveys represents about 75% participation, it is still a relatively small number of data points. Thus, a failure to find a correlation in this data may reflect inadequate evidence rather than a true lack of correlation (only correlations that achieved a 95% confidence level are shown). The findings here are presented in round numbers to increase comprehensibility and reflect the inherent imprecision of the process. Where numbers are presented and do not add up to 100%, any missing amount represents people who did not answer the question. Where non-respondents are included, rounding error may push totals slightly above 100%.

The survey respondents are mostly male (18% female, 78% male, 4% non-responding), and 65% of them are over 30. Of half of them have bachelor's degrees; one in five have master's degrees, and about 2% have doctorates.

Nearly half of all respondents (47%) had computer science majors, and an additional 12% had computer science minors. In general, they support large complex sites and work for large companies; about three-quarters of them work at sites with more than 100 people, and 29% work for sites with more than 2,000. The largest number of facilities (over a third) are in the 100-500 user range and the 100-500 computer range, and nearly a third of them have six or more operating systems. Two out of three of the respondents have been with their current company less than five years, while nearly half of them have been in system administration more than five years.

According to the survey results, compensation for over 75% of the respondents is above $40,000 annually. Surprisingly, pay correlated only to age and years in system administration; there was no significant correlation to whether the organization was business, educational, or government, or to educational level, job title, job description, or size or complexity of site.

New York City has the highest number of top-earning system administrators, followed by the greater metropolitan areas of San Francisco and Los Angeles. Dallas, Houston, Denver, Chicago, and the Washington DC area all fall between those top cities and the rest of the United States. This appears to reflect the cost of living, in general.

An overwhelming majority of respondents (86%) reported that they work 41-60 hours a week, and one in ten works over 60 hours a week. 43% report that they receive comp time for their extra hours, 16% receive overtime pay, and 32% receive no compensation at all. The pager has become ubiquitous, with two-thirds carrying pagers all or some of the time. Compensation for pager time is rare, however, with only 12% reporting any compensation for either carrying a pager or answering calls.
These results are consistent with the results of last year's survey in most categories. Last year's survey showed approximately the same number of degrees, the same gender and age breakdown, and the same distribution of number of users, computers, and operating systems. Last year's respondents tended to have been in system administration longer, in their current job less long, and to earn less money. An even higher percentage of them were from large employers (45% as opposed to 29% at companies with over 2000 employees). This probably represents skewed data, given that last year's survey data covers only 281 respondents, although it would be pleasant to believe that the salary data, which shows the peak of the bell curve moving from the 40-45 thousand range to the 50-60 thousand range, reflects reality.

Here's a summary of both years' results. All numbers are percentages of all respondents. NR means they did not respond to this question.

### Number of Respondents

<table>
<thead>
<tr>
<th></th>
<th>1993</th>
<th>1994</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>282</td>
<td>909</td>
</tr>
</tbody>
</table>

### Pay Ranges

<table>
<thead>
<tr>
<th></th>
<th>1994</th>
<th>1993</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-25</td>
<td>25-30</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

### Employees In Organization

<table>
<thead>
<tr>
<th>#</th>
<th>-10</th>
<th>25</th>
<th>100</th>
<th>500</th>
<th>1000</th>
<th>2000</th>
<th>+2000</th>
<th>NR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>4</td>
<td>8</td>
<td>12</td>
<td>25</td>
<td>9</td>
<td>9</td>
<td>29</td>
<td>4</td>
</tr>
<tr>
<td>1993</td>
<td>2</td>
<td>2</td>
<td>5</td>
<td>13</td>
<td>8</td>
<td>13</td>
<td>45</td>
<td>12</td>
</tr>
</tbody>
</table>

### Users Supported

<table>
<thead>
<tr>
<th>#</th>
<th>-10</th>
<th>25</th>
<th>100</th>
<th>500</th>
<th>1000</th>
<th>2000</th>
<th>+2000</th>
<th>NR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>1</td>
<td>4</td>
<td>18</td>
<td>35</td>
<td>13</td>
<td>8</td>
<td>17</td>
<td>4</td>
</tr>
<tr>
<td>1993</td>
<td>2</td>
<td>7</td>
<td>21</td>
<td>39</td>
<td>12</td>
<td>6</td>
<td>8</td>
<td>5</td>
</tr>
</tbody>
</table>

### Computers Supported

<table>
<thead>
<tr>
<th>#</th>
<th>-10</th>
<th>25</th>
<th>100</th>
<th>500</th>
<th>1000</th>
<th>2000</th>
<th>+2000</th>
<th>NR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>0</td>
<td>5</td>
<td>13</td>
<td>34</td>
<td>14</td>
<td>9</td>
<td>19</td>
<td>6</td>
</tr>
<tr>
<td>1993</td>
<td>2</td>
<td>4</td>
<td>22</td>
<td>41</td>
<td>11</td>
<td>8</td>
<td>8</td>
<td>4</td>
</tr>
</tbody>
</table>

### OS Versions Supported

<table>
<thead>
<tr>
<th>#</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6+</th>
<th>NR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>1</td>
<td>9</td>
<td>20</td>
<td>18</td>
<td>13</td>
<td>32</td>
<td>7</td>
</tr>
<tr>
<td>1993</td>
<td>4</td>
<td>7</td>
<td>13</td>
<td>15</td>
<td>15</td>
<td>45</td>
<td>1</td>
</tr>
</tbody>
</table>
**Education Level**

<table>
<thead>
<tr>
<th></th>
<th>BA/S</th>
<th>MA/S</th>
<th>PhD</th>
<th>Other</th>
<th>NR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>52</td>
<td>21</td>
<td>2</td>
<td>22</td>
<td>3</td>
</tr>
<tr>
<td>1993</td>
<td>52</td>
<td>23</td>
<td>2</td>
<td>8</td>
<td>14</td>
</tr>
</tbody>
</table>

**Gender**

<table>
<thead>
<tr>
<th></th>
<th>M</th>
<th>F</th>
<th>NR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>73</td>
<td>18</td>
<td>4</td>
</tr>
<tr>
<td>1993</td>
<td>77</td>
<td>22</td>
<td>1</td>
</tr>
</tbody>
</table>

**Years at Current Company**

<table>
<thead>
<tr>
<th></th>
<th>-1</th>
<th>1-2</th>
<th>3-5</th>
<th>6-10</th>
<th>10+</th>
<th>NR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>15</td>
<td>20</td>
<td>33</td>
<td>19</td>
<td>14</td>
<td>0</td>
</tr>
<tr>
<td>1993</td>
<td>16</td>
<td>15</td>
<td>41</td>
<td>19</td>
<td>9</td>
<td>0</td>
</tr>
</tbody>
</table>

**Years in System Administration**

<table>
<thead>
<tr>
<th></th>
<th>-1</th>
<th>1-2</th>
<th>3-5</th>
<th>6-10</th>
<th>10+</th>
<th>NR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>3</td>
<td>13</td>
<td>36</td>
<td>34</td>
<td>11</td>
<td>3</td>
</tr>
<tr>
<td>1993</td>
<td>1</td>
<td>1</td>
<td>19</td>
<td>42</td>
<td>36</td>
<td>1</td>
</tr>
</tbody>
</table>

**Hours Worked Per Week**

<table>
<thead>
<tr>
<th></th>
<th>-32</th>
<th>32-40</th>
<th>41-60</th>
<th>61-80</th>
<th>80+</th>
<th>NR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>2</td>
<td>13</td>
<td>78</td>
<td>7</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1993</td>
<td>1</td>
<td>19</td>
<td>76</td>
<td>3</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Here are some additional results in areas new to this survey.

**Receiving Overtime Pay or Comp Time**

<table>
<thead>
<tr>
<th></th>
<th>Paid</th>
<th>Comp Time</th>
<th>No Pay</th>
<th>N/A</th>
<th>NR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>16</td>
<td>43</td>
<td>32</td>
<td>7</td>
<td>2</td>
</tr>
</tbody>
</table>

**Carry a Pager**

<table>
<thead>
<tr>
<th></th>
<th>Always</th>
<th>Sometimes</th>
<th>Never</th>
<th>NR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>45</td>
<td>23</td>
<td>33</td>
<td>0</td>
</tr>
</tbody>
</table>
Compensated for Carrying a Pager

<table>
<thead>
<tr>
<th>%</th>
<th>When Carrying</th>
<th>When Called</th>
<th>Not Compensated</th>
<th>NR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>5</td>
<td>7</td>
<td>55</td>
<td>33</td>
</tr>
</tbody>
</table>

Degree in Computer Science

<table>
<thead>
<tr>
<th>%</th>
<th>Yes/Major</th>
<th>Yes/Minor</th>
<th>No</th>
<th>NR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>47</td>
<td>12</td>
<td>25</td>
<td>16</td>
</tr>
</tbody>
</table>

SAGE Local Group Support

by Elizabeth D. Zwicky
<zwicky@corp.sgi.com>

The following document on SAGE services for local groups was approved by the board of directors at its most recent meeting. This is the current list of services we offer, though we expect to add new services to the list as time goes by (for instance, we are currently working to define a speaker registry service). Please send your comments and requests to <sage-board@usenix.org>.

SAGE believes that local groups provide important services to the system administration community, and therefore wishes to support them as much as possible. Because we also believe that different formats and organizational structures are appropriate for different local groups, we do not want to specify one particular relationship that makes something a “SAGE local group.” Instead, we provide a list of services that local groups can take advantage of or not as appropriate for their situation.

These services are available to any not-for-profit group that serves system administrators. Except as stated in the requirements for particular services, SAGE makes no restrictions on the nature or name of the group. Please remember that most of the people providing these services are also volunteers with full-time jobs outside of SAGE.

Requests for project funding

SAGE will entertain requests for funding of limited-term projects (booklet publishing, establishment of speaker series, etc.). Calls for such proposals will be periodically announced, though the proposals themselves will be entertained at any time. Requesting groups should identify a SAGE Board member who will be willing to serve as “champion” for the proposal to the Board. SAGE will consider all proposals that benefit system administrators, but may give precedence to proposals that benefit the system administration community as a whole, as opposed to those limited to the local group. Unless the group contains at least one SAGE member, proposals that benefit only the local group will not be considered.

Initial formation

Advice, commiseration, and experience are available from the sage-locals working group and its associated mailing list, which is also a good place to find other people interested in forming a group in your area. SAGE will provide you with the mailing list information for those USENIX/SAGE members in your area who have authorized the release of their information, as long as you agree to use the information only once, and only for the purpose of creating a non-profit group for system administrators.

Publicity

Any local group for system administrators that provides contact information will be listed in ;login:; in SAGE’s WWW server, and any place else that SAGE provides local group information. By default the WWW server has an extremely basic page listing geographic coverage and contact information. Instead, local groups can provide their own page or the URL for their own server. Local groups may also use the sage-announce mailing list to announce their meetings.

Insurance

Many public meeting places require liability insurance for groups that wish to use them. SAGE can make arrangements to extend its liability insurance to local groups which meet the requirements of the insurance company; this will always require that a current SAGE member is a contact for the group and attends the meetings. Other requirements will vary depending on the organization of the local group and its meeting locations.
Perl Practicum: A Plea for Clarity

by Hal Pomeranz  
<pomeranz@netcom.com>

After a brief hiatus, welcome back to the second year of "Perl Practicum." This month, Rob asked me to write a little piece about writing readable Perl. I started out with every intention of doing just that, but the article evolved instead into a piece on writing maintainable Perl. This dismayed me for a bit, until I realized that maintainability was the primary driving force for clarity. If, having been written, one's code never had to be looked at again, the motivation for writing clear code would be greatly reduced (note: not eradicated). That bit of philosophy done with, let us proceed with what amounts to little more than a collection of useful tips I have collected through the years.

A Little About Loops

The first rule is "say what you mean." Compare the following two mechanisms for doing the same thing:

```perl
    grep($array[$_]++, @list);  # BAD!
    
    for (@list) {              # OK
      $array[$_] = 1;
    }
```

I claim the second form is to be preferred since it makes clear what is going on: we are iterating over @list and setting values in $array to be non-zero (the assignment, as opposed to the auto-increment operator, is significant). There are several other, more verbose, ways to rewrite the above "for" loop: decide which form you are most comfortable with, but avoid `grep()` as a loop operator. I generally prefer to optimize for readability over performance.

As another example, consider the following two infinite loop constructs:

```perl
    while (1) {                # BAD!
      ...                     
    }
    
    for (;;) {               # OK
      ...                     
    }
```

The second form tends to be more visually arresting, it alerts the reader that something important is happening.

As long as we are on the subject of loops, let us examine another rule for clear communication, "say it succinctly." The `goto` statement and multi-level break commands are both to be abhorred because they hamper the reader's ability to conceptualize the program flow at a glance. I went look-

Conditional Expressions

Along the lines of "saying what you mean" remember that you can always use `until` instead of `while` and `unless` instead of `if:

```perl
    &usage() unless (@ARGV);
    until ($value > $LIMIT) {
      ...
    }
```

Avoiding extra negation in conditional expressions can be a great aid to clarity. Perl can read like clear prose if you are careful and use informative symbolic names.

With the postfix conditional operators, be careful to put the most important part of the statement up front. This is why we write:

```perl
    open(...) || die ...;         # recommended
    
    die ... unless open(...);    # EVIL!
```

The purpose of the statement is to associate a file handle with a file or process. The `die()` operation is merely a case of exception handling.

Similarly, avoid overloading conditional expressions with operations which actually manipulate program data or have other side effects. Evaluate an expression to take a logical branch in the program flow and then perform your operations.

Parentheses, Functions, and Others

Always err on the side of extra parentheses, though of course too many can cause problems as well. In conditional expressions, "extra" parentheses will help the reader parse the expression. They also help protect the application from maintenance by programmers with a poorer grasp of operator precedence than the author.

Perl is of course extremely forgiving as far as parenthesizing function argument lists. Always parenthesize function arguments. The classic example of the importance of this rule is taken directly from the Camel book:

```perl
    print (1+2)*3, "\n";        # INCORECT!
```

This prints the value expression in parentheses, i.e., the number three without a newline. The statement is syntactically correct (points to you if you figure out exactly what happens in the rest of the line) and the Perl interpreter will not complain, but the output is wildly different from:
print((1+2)*3, "\n"); # CORRECT

which is probably what the author of the code intended.

If you only need a few scattered values out of a list value returned by a function, please avoid assignment to dummy variables. In other words, do:

```perl
($login, $name, $home) = (getpwent)[0,6,7];  # GOOD
```

rather than:

```perl
($login, $dummy, $dummy $dummy, $dummy, $dummy, $name, $home) = getpwent;  # BAD
```

Aside from wasted typing, the second form obscures precisely which information you are interested in manipulating.

### Odds and Ends

It is a good general principle when writing clear code never to rely on default behaviors. Explicitly `undef` your variables or assign them zero values before using them for the first time. This helps to avoid errors introduced through later modifications.

Function defaults are a trickier issue. You can pretty well assume that any Perl function will operate on `$_` or `@_` when given no arguments. This is a nice feature and I use it all the time (too convenient to give up, I suppose). It does, however, make Perl code less than clear to the uninformed reader, and I have had occasions where something unexpected has cropped up because `$_` did not contain what I thought it did. On a more trivial issue, I would like to make a plea for explicitly using the "<" character when opening a file for reading, even though this is the default behavior for `open()`.

Never hard-code pathnames or other constants into your program. Assign these values to variables AT THE TOP of your program. For example, here are the first few lines of an application I wrote to manipulate a remote optical jukebox:

```perl
#!/usr/bin/perl

$jukehost = 'gator';
$sfjukedir = '/rd/juke';
$realjukedir = '/export/jb/jb0';
$localjukedir = '/jukebox';
$remotecmd = '/usr/local/etc/jbadm';
```

When the code is written in this fashion, maintenance becomes a breeze.

Always explicitly close file and directory handles as soon as you finish processing the data. You avoid potential shortage problems, protect your code from interesting side effects caused by later modification, and make your code clearer to the hypothetical external viewer.

### Issues of Convention

The careful reader will note that I have been discussing issues of clarity related to program syntax. Equally important are issues which are not dictated by the language definition, such as your indentation scheme, variable naming conventions, and commenting scheme. These are also the areas where you run into the most religious warfare.

To avoid this morass (for example, everybody I know hates my bracing style), I suggest only one simple rule. Pick a site standard that everybody can live with and stick to it. Even a bad standard is better than no standard at all. If you are forced to maintain code that is developed and used externally to your organization, then maintain whatever conventions pertain to the code as you received it.

For a good starting point, there is a document available on the Internet entitled *Recommended C Style and Coding Standards* (originally from a document prepared by committee at Bell Labs, but modified by Henry Spencer, David Keppel, and Mark Brader). Obtain `/pub/cstyle.tar.Z` from `ftp.cs.washington.edu`.

### Further Study

Please note that everything said above applies pretty well to any language you choose to program in. Certain constructs may or may not be available to you, but clarity should be syntax-independent. For a more in depth treatment of this material, start with a good C style guide and then follow up through any bibliographic information provided in it.

### Point/Counterpoint

### Putting Up With Putting Up

*by George Spellvin*

In `;login: 19.5`, Hal Pomeranz `<pomeranz@netcom.com>` gives some good suggestions on how to cope with Solaris 2.x. While solving the immediate problem of 'how to make it work' is useful, we need to deal with the problems at the source. Why should everyone duplicate effort in after-market workarounds?
SAGE NEWS

Like all OS's, Solaris is far from perfect. It is far from usable for a lot of people. Finding workarounds for some of the problems is a laudable (and indeed, necessary) project. However, we still need to keep the pressure on Sun to fix and improve their products. Vendors must be made accountable for what they produce.

- Enough people beating up on IBM could have saved us from SMIT, but we all shrugged and found workarounds. Now we’re stuck with it.

- Enough people beating up on Sun has finally produced an NIS that approaches a reasonable level of usability for large sites.

- Enough people beating up on HP might just get real disk partitioning in HP-UX. (I know, they’ve been promising it for years.)

If no one tries to change things, the vendors won’t have any reason to do it right. They will just put out product, and we’ll keep finding clever hacks to get around the more obstructive design decisions. Rather than fighting the operating system, wouldn’t you rather use your clever hacks to improve life for your users and yourself?

Call your Sun sales rep. Tell him or her what you don’t like about Solaris. Tell them about the bugs you’ve found. Keep up the pressure until they fix it. Don’t forget though, it isn’t just Sun, do the same for IBM & AIX, HP & HP-UX, DEC & OSF/1, etc.

I’d also like to draw your attention to the standards bodies. How many of you have bothered to get a copy of the POSIX sysadmin standards draft? How many of you have contacted COS/ and told them what you want to see in an OS? How many of you care how hard or easy your job will be in five years? I’ve seen the proposed POSIX standard and it scares me silly. Forget distributed admin if that ever passes.

I’m writing this under a pseudonym. Why? Because a group I was active with, a group that was trying to get Sun to fix Solaris, caused enough commotion that Sun asked my management to intercede on their behalf. If we could get that much attention in only six months, think what kind of a difference all of us working together can make.

We’re all busy, but if we don’t get active now, we’ll have only ourselves to blame later. Don’t put up; get involved. Don’t hack around it; file a bug report and make the vendor fix it. Don’t sit back and let the standards bodies decide for us how we’ll do our jobs; join up and make yourself heard.

A Response to Mr. Spellvin and Others

by Hal Pomeranz
<pomeranz@TFS.COM>

My Solaris article provoked more comment from more people than any other article that I have written for ;login:. I would like to try to reply to some points raised by all, but to George Spellvin’s points in particular.

I believe that George and I actually have more common ground under our feet than may actually be apparent at first glance. I strongly advocate agitating the vendor community to get them to fix some of the system management tragedies that have been foisted upon our professional community in recent years (and those, like POSIX, which are catastrophes waiting to happen). My original article attempted to be extremely strong on the issue of patching your Solaris machine before attempting to get any work done. Those patches would not be available without the efforts of Mr. Spellvin and others who have brought problems to the vendor’s attention.

George is also correct that somebody has to stand up and try to change things. Unfortunately, I think I have hit my quixotic limit. I hate the fact that most commercial UNIX systems no longer are shipped with a bundled C compiler. I do not think that Solaris should try and pretend to have BSD compatibility libraries. The list of things I would like to see changed goes on. I would be extremely happy if somebody else did my dirty work in this regard, but I cannot complain if they do not. In the meantime, I do what I have to do to keep myself and my business going.

Actually, we shouldn’t even be having to fight with the vendors about all this mess. It has been noted by others that the wrong people, depending upon your perspective, are setting the priorities for the next generation of computing devices. Your company wants more cycles and neat new features. Nobody who makes real purchasing decisions in your organization ever says, “Hey, hold on there just a bit fellas! Why don’t you clean up all those outstanding bugs and security holes, make your systems easier to manage, and retune them so I don’t have to buy new hardware to support your new OS, and then we’ll talk about buying your new FUBAR devices?”

To end on a more specific note, one comment I consistently heard was in response to my assertion, “do not try to run Solaris 2.x on anything less than a Sparc Classic with 24MB of RAM – it is just too slow on weaker machines.” Many people told me, “Well I run it on my Sparc [1+]IPCIIPX] with X MB of RAM, and it’s just fine.” I disagree, and, yes, I have run Solaris on those platforms. Your mileage may vary.
Wherever You Go, There You Are

by Elizabeth D. Zwicky
<zwicky@corpsgi.com>

Recently I spent six weeks in Cortaillod, Switzerland, working at SGI's European manufacturing center. People's reactions to this were rather predictable. For one thing, they all wanted to know exactly where Cortaillod was; if you're Swiss, the applicable answer is "About 10 minutes outside of Neuchatel." If you're European, it will probably help to know that Neuchatel is between Lausanne and Basel. If you're American, Neuchatel is about an hour and a half from Geneva. If you didn't know that Geneva was in Switzerland, then about all you need to know is that we're talking about Central Europe here.

Everybody also assumed that I would write a column about the stunning differences between American system administration and Swiss system administration. It's true that I can go on for pages about the differences between Switzerland and America (the icced tea rait alone takes up several screens, and many people of my acquaintance are waiting with bated breath for the moment when I stop saying things like "you don't get much Mexican guitar music in Switzerland, but there are a surprising number of Andean pipe players") but the fact is that system administration is much the same there as it is here. Certainly there are language issues (there's nothing like setting out to read the fine manual only to discover that you have the German version in which only the illustrations make any sense to you), and being in a time zone nine hours away from the center of your industry adds interest to technical support, but, basically, system administration is system administration.

This should not actually come as a surprise to anybody who has held many system administration jobs. I have been slowly moving away from academia into industry, starting from a job in a university, moving to one in a research institute, and finally holding one in a genuine for-profit company. At every stage, people kindly warned me that I was moving into a higher-stress world with a lower standard of intelligence. This is not true. Anybody who thinks the academic world is low-stress has never encountered a graduate student in the final throes of getting a PhD, and anybody who thinks the world of contract research is calm has never truly experienced the full impact of the words "soft money." Your only requirement is to rephrase your job every month or so, but it's not stressful, not really, unless you have some neurotic need to know that you're going to have a job next month and maybe some inkling of what it will be.

As for relative intelligence, it may be true that your average university student is more intelligent than your average company employee, although I wouldn't make any large bets, but in any case it does not follow that your average university student has more common sense, a greater ability to use a computer, or more patience and graciousness when dealing with system administrators. Your average university student tends to be younger, to be more emotional, and to have more spare time. This does not imply more tolerance for delay, however, merely an apparently infinite supply of time in which to think up ways to use computers for purposes other than those for which their purchasers intended them. In moments of crisis, employees tell you they have children at home waiting for them, while students tell you they have homework due, giving you a choice of whether to blight the future of the present generation or the next.

Similarly, the laid-back Californians are not any happier having to wait for their computers to work than the time-conscious Swiss, although admittedly they swear about it in fewer languages. You do get more language misunderstandings in Switzerland (although even in America the computer industry is not completely inhabited by native speakers of American English, particularly in California). Unintuively, the worst of these misunderstandings occur between the Americans and the British. On one occasion I had the humiliating experience of trying to fix a machine I believed was named Bonnie because of the existence of a nearby Clyde. Eventually its owner enlightened me by pointing out that the machine was named after the large purple dinosaur. After that I asked everyone to write down everything for me.

But in general, people vastly overestimate the differences. For instance, people tend to assume that when I talk about my colleagues leaving work because the weather is too good and they want to wind-surf, I'm referring to Californians. As it happens these were Swiss wind-surfing Lake Neuchatel, where the wind is less reliable but you don't need to worry about the tide. (The only Californians I've worked with who were lured away by the weather were going to play golf.) The local wine tastings were also in Switzerland, as was the show where we did the live intercity ATM demo in conjunction with the phone company.

Admittedly, the language situation made setting up the demo more interesting. Especially during an initial problem in which the videoconference application was not connecting all participants to each other, and was connecting different ones for the audio, the video, and the whiteboard.

Imagine, if you will, somebody who speaks only English, who can hear a German speaker with minimal English and a bilingual German and English speaker, who can see but not hear each other and hear but not see him. Now imagine that they are trying to debug multicast routing over ATM. Forty-five minutes passed in the attempt to determine who could see whom and who could hear whom, and success was not achieved until several telephone calls were added in. Unfor-
fortunately, there was a magic moment when one of my colleagues could both see and hear the person who he was trying to reach on the telephone - in fact, he could hear the phone ringing on the far end over the videoconference audio - but the person he was trying to call wasn't answering the phone because he was busy with the videoconference problem. By waving the phone at the video camera, and pleading into the microphone, the caller was able to communicate that he wanted a phone answered to most of the conference participants, but not to the one he was calling. In short order, 5 people in 3 cities were devoted to the problem of getting the phone answered (often one could watch irrelevant people answering their phones), and the caller was clutching his microphone in his teeth in order to type with one hand and hold the telephone with the other. This handily prevented him from swearing at his local audience, which was both steadily growing and laughing hysterically.

In other words, it was just like working here; exciting, maddening, and full of unintentional humor.

System Administration Tools Your Vendor Never Told You About: The Toy
by Elizabeth D. Zwicky
<zwicky@corp.sgi.com>

System administrators need toys for some of the same reasons that doctors need them in their waiting rooms; they keep cranky unhappy people occupied while they wait. It's true that most doctors only keep toys for children, and stick the rest of us with mere unamusing elderly magazines, but then doctors don't have to sit in their own waiting rooms, either, and their patients are generally sick and therefore subdued.

The ideal toy for a system administrator is both amusing and calming for visitors, and serves as stress release for the system administrator. Avoid fragile objects, and anything that makes a noise you're going to object to while frantically debugging things, and anything that shoots water or produces magnetic fields. Legos and their ilk are good, although people tend to rapidly become attached to their creations and become unhappy when you want to take them apart so other people can play. Some people swear by puzzles, but others complain that they either make people more frustrated, or make them refuse to leave until they're done. I have gotten one emphatic vote for a toy called a "Jibba Jabber," which is designed to be throttled.

System Administration Tools Your Vendor Never Told You About: Painkiller
by Hal Pomeranz
<pomeranz@netcom.com>

At the risk of sounding like an Excedrin commercial, I had a headache today. System Administration is a high stress job and I, like many other sufferers, react to stress by developing tension headaches. System Administration is also a customer service job, but I have a hard time being nice and smiling at people when my shoulder, neck, and forehead muscles are tied up in knots. It's at times like this when the 1,000 count bottle of generic pain reliever is a godsend.

Whatever poison you choose (I'm an ibuprofin man myself), buy in bulk and keep it where everybody in the your System Admin group can get at it. Think of it as a group harmony preserving agent. At some places where I've worked, a medical services company provides handy little wall racks that contain boxes of single dose packets. There are two problems with this scheme: (1) personally, it takes several "single dose" packets to satisfy my habit, and (2) you're at the whim of some outside agency as far as restocking goes (somehow their restocking schedules were never built with System Administrators in mind).

Of course, you should strive to develop a chemical free lifestyle. Eat right, get plenty of rest and exercise, get out and do things with your non-work friends, and all of those other things you read in stress reduction pamphlets from your doctor's office. But keep that industrial size bottle of aspirin handy, just in case.
Bash – The GNU shell

by Chet Ramey
<chet@po.cwru.edu>

Introduction

Bash is the shell, or command language interpreter, that will appear in the GNU operating system. The name is an acronym for the “Bourne-Again SHell,” a pun on Steve Bourne, the author of the direct ancestor of the current UNIX shell `/bin/sh`, which appeared in the Seventh Edition Bell Labs Research version of UNIX.

Bash is an `sh`-compatible shell that incorporates useful features from the Korn shell (`ksh`) and the C shell (`csh`), described later in this article. It is ultimately intended to be a conformance implementation of the IEEE POSIX Shell and Utilities specification (IEEE Working Group 1003.2). It offers functional improvements over `sh` for both interactive and programming use. While the GNU operating system will most likely include a version of the Berkeley shell `csh`, Bash will be the default shell. Like other GNU software, Bash is quite portable. It currently runs on nearly every version of UNIX and a few other operating systems – an independently-supported port exists for OS/2, and there are rumors of ports to DOS and Windows NT. Ports to UNIX-like systems such as QNX and Minix are part of the distribution. The original author of Bash was Brian Fox, an employee of the Free Software Foundation. I am the current developer and maintainer, working as a volunteer at Case Western Reserve University.

What’s POSIX, anyway?

POSIX is a name originally coined by Richard Stallman for a family of open system standards based on UNIX. There are a number of aspects of UNIX under consideration for standardization, from the basic system services at the system call and C library level to applications and tools to system administration and management. Each area of standardization is assigned to a working group in the 1003 series.

The POSIX Shell and Utilities standard has been developed by IEEE Working Group 1003.2 (POSIX.2).* It concentrates on the command interpreter interface and utility programs commonly executed from the command line or by other programs. An initial version of the standard has been approved and published by the IEEE, and work is currently underway to update it. There are four primary areas of work in the 1003.2 standard:

• Aspects of the shell’s syntax and command language. A number of special built-ins such as `cd` and `exec` are being specified as part of the shell, since their functionality usually cannot be implemented by a separate executable;

• A set of utilities to be called by shell scripts and applications. Examples are programs like `sed`, `tr`, and `awk`. Utilities commonly implemented as shell built-ins are described in this section, such as `test` and `kill`. An expansion of this section’s scope, termed the User Portability Extension, or UPE, has standardized interactive programs such as `vi` and `mailx`;

• A group of functional interfaces to services provided by the shell, such as the

traditional system() C library function. There are functions to perform shell word expansions, perform filename expansion (globbing), obtain values of POSIX.2 system configuration variables, retrieve values of environment variables (getenv()), and other services;

- A suite of “development” utilities such as c89 (the POSIX.2 version of cc), and yacc.

Bash is concerned with the aspects of the shell’s behavior defined by POSIX.2. The shell command language has of course been standardized, including the basic flow control and program execution constructs, I/O redirection and pipelining, argument handling, variable expansion, and quoting. The special built-ins, which must be implemented as part of the shell to provide the desired functionality, are specified as being part of the shell; examples of these are eval and export. Other utilities appear in the sections of POSIX.2 not devoted to the shell which are commonly (and in some cases must be) implemented as built-in commands, such as read and test. POSIX.2 also specifies aspects of the shell’s interactive behavior as part of the UPE, including job control and command line editing. Interestingly enough, only vi-style line editing commands have been standardized; emacs editing commands were left out due to objections.

While POSIX.2 includes much of what the shell has traditionally provided, some important things have been omitted as being “beyond its scope.” There is, for instance, no mention of a difference between a login shell and any other interactive shell (since POSIX.2 does not specify a login program). No fixed startup files are defined, either – the standard does not mention .profile.

## Basic Bash features

Since the Bourne shell provides Bash with most of its philosophical underpinnings, Bash inherits most of its features and functionality from sh. Bash implements all of the traditional sh flow control constructs (for, if, while, etc.). All of the Bourne shell built-ins, including those not specified in the POSIX.2 standard, appear in Bash. Shell functions, introduced in the SVR2 version of the Bourne shell, are similar to shell scripts, but are defined using a special syntax and are executed in the same process as the calling shell. Bash has shell functions which behave in a fashion upward-compatible with sh functions. There are certain shell variables that Bash interprets in the same way as sh, such as PSI, IFS, and PATH. Bash implements essentially the same grammar, parameter and variable expansion semantics, redirection, and quoting as the Bourne shell. Where differences appear between the POSIX.2 standard and traditional sh behavior, Bash follows POSIX.

The Korn Shell (ksh) is a descendent of the Bourne shell written at AT&T Bell Laboratories by David Korn.* It provides a number of useful features that POSIX and Bash have adopted. Many of the interactive facilities in POSIX.2 have their roots in the ksh: for example, the POSIX and ksh job control facilities are nearly identical. Bash includes features from the Korn Shell for both interactive use and shell programming. For programming. Bash provides variables such as RANDOM and REPLY, the typeset built-in, the ability to remove substrings from variables based on patterns, and shell arithmetic.

RANDOM expands to a random number each time it is referenced; assigning a value to RANDOM seeds the random number generator. REPLY is the default variable used by the read built-in when no variable names are supplied as arguments. The typeset built-in is used to define variables and give them attributes such as readonly. Bash arithmetic allows the evaluation of an expression and the substitution of the result. Shell variables may be used as operands, and the result of an expression may be assigned to a variable. Nearly all of the operators from the C language are available, with the same precedence rules:

```
$ echo $((3 + 5 * 32))
163
```

For interactive use, Bash implements ksh-style aliases and built-ins such as fc (discussed below) and jobs. Bash aliases allow a string to be substituted for a command name. They can be used to create a mnemonic for a UNIX command name (alias del=rm), to expand a single word to a complex command (alias news='xterm -g 80x45 -title trn -e trn -g -S1 -N &'), or to ensure that a command is invoked with a basic set of options (alias ls="/bin/ls -F").

The C shell (csh)†, originally written by Bill Joy while at Berkeley, is widely used and quite popular for its interactive facilities. Bash includes a csh-compatible history expansion mechanism ("! history"), brace expansion, access to a stack of directories via the pushd, popd, and dirs built-ins, and tilde expansion, to generate users’ home directories. Tilde expansion has also been adopted by both the Korn Shell and POSIX.2.

There were certain areas in which POSIX.2 felt standardization was necessary, but no existing implementation provided the proper behavior. The working group invented and standardized functionality in these areas, which Bash implements. The command built-in was invented so that shell functions could be written to replace built-ins; it makes the capabilities of the built-in available to the func-

---

makes the capabilities of the built-in available to the function. The reserved word ‘!” was added to negate the return value of a command or pipeline; it was nearly impossible to express “if not x” cleanly using the sh language. There exist multiple incompatible implementations of the test built-in, which tests files for type and other attributes and performs arithmetic and string comparisons. POSIX considered none of these correct, so the standard behavior was specified in terms of the number of arguments to the command. POSIX.2 dictates exactly what will happen when four or fewer arguments are given to test, and leaves the behavior undefined when more arguments are supplied. Bash uses the POSIX.2 algorithm, which was conceived by David Korn.

**Features not in the Bourne Shell**

There are a number of minor differences between Bash and the version of sh present on most other versions of UNIX. The majority of these are due to the POSIX standard, but some are the result of Bash adopting features from other shells. For instance, Bash includes the new ‘!” reserved word, the command built-in, the ability of the read built-in to correctly return a line ending with a backslash, symbolic arguments to the umask built-in, variable substring removal, a way to get the length of a variable, and the new algorithm for the test built-in from the POSIX.2 standard, none of which appear in sh.

Bash also implements the “$(...)” command substitution syntax, which supersedes the sh "...") construct. The “$(...)” construct expands to the output of the command contained within the parentheses, with trailing newlines removed. The sh syntax is accepted for backwards compatibility, but the "$(...)" form is preferred because its quoting rules are much simpler and it is easier to nest.

The Bourne shell does not provide such features as brace expansion, the ability to define a variable and a function with the same name, local variables in shell functions, the ability to enable and disable individual built-ins or write a function to replace a built-in, or a means to export a shell function to a child process.

Bash has closed a long-standing shell security hole by not using the $IFS variable to split each word read by the shell, but splitting only the results of expansion (ksh and the 4.4 BSD sh have fixed this as well). Useful behavior such as a means to abort execution of a script read with the "!" command using the return built-in or automatically exporting variables in the shell's environment to children is also not present in the Bourne shell. Bash provides a much more powerful environment for both interactive use and programming.

**Bash-specific Features**

This section details a few of the features which make Bash unique. Most of them provide improved interactive use, but a few programming improvements are present as well. Full descriptions of these features can be found in the Bash documentation.

**Startup Files**

Bash executes startup files differently than other shells. The Bash behavior is a compromise between the csh principle of startup files with fixed names executed for each shell and the sh "minimalist" behavior. An interactive instance of Bash started as a login shell reads and executes ~/.bash_profile (the file .bash_profile in the user's home directory), if it exists. An interactive non-login shell reads and executes ~/bashrc. A non-interactive shell (one begun to execute a shell script, for example) reads no fixed startup file, but uses the value of the variable $ENV, if set, as the name of a startup file. The ksh practice of reading $ENV for every shell, with the accompanying difficulty of defining the proper variables and functions for interactive and non-interactive shells or having the file read only for interactive shells, was considered too complex. Ease of use won out here. Interestingly, the next release of ksh will change to reading $ENV only for interactive shells.

**New Built-in Commands**

There are a few built-ins which are new or have been extended in Bash. The enable built-in allows built-in commands to be turned on and off arbitrarily. To use the version of echo found in a user's search path rather than the Bash built-in, enable -n echo suffices. The help built-in provides quick synopses of the shell facilities without requiring access to a manual page. Builtin is similar to command in that it bypasses shell functions and directly executes built-in commands.

Access to a csh-style stack of directories is provided via the pushd, popd, and dirs built-ins. Pushd and popd insert and remove directories from the stack, respectively, and dirs lists the stack contents. On systems that allow fine-grained control of resources, the ulimit built-in can be used to tune these settings.

Ulimit allows a user to control, among other things, whether core dumps are to be generated, how much memory the shell or a child process is allowed to allocate, and how large a file created by a child process can grow. The suspend command will stop the shell process when job control is active; most other shells do not allow themselves to be stopped like that. Type, the Bash answer to which and whence, shows what will happen when a word is typed as a command:
$ type export
export is a shell built-in
$ type -t export
builtin
$ type bash
bash is /bin/bash
$ type cd
cd is a function

cd ()
{
    builtin cd $(1"$@") & & xtitle $HOST: $PWD
}

Various modes tell what a command word is (reserved word, alias, function, built-in, or file) or which version of a command will be executed based on a user's search path. Some of this functionality has been adopted by POSIX.2 and folded into the `command` utility.

### Editing and Completion

One area in which Bash shines is command line editing. Bash uses the `readline` library to read and edit lines when interactive. Readline is a powerful and flexible input facility that a user can configure to individual tastes. It allows lines to be edited using either `emacs` or `vi` commands, where those commands are appropriate. The full capability of `emacs` is not present – there is no way to execute a named command with M-x, for instance – but the existing commands are more than adequate. The `vi` mode is compliant with the command line editing standardized by POSIX.2.

Readline is fully customizable. In addition to the basic commands and key bindings, the library allows users to define additional key bindings using a startup file. The `inputrc` file, which defaults to the file `~/.inputrc`, is read each time readline initializes, permitting users to maintain a consistent interface across a set of programs. Readline includes an extensible interface, so each program using the library can add its own bindable commands and program-specific key bindings. Bash uses this facility to add bindings that perform history expansion or shell word expansions on the current input line.

Readline interprets a number of variables which further tune its behavior. Variables exist to control whether or not eight-bit characters are directly read as input or converted to meta-prefixed key sequences (a meta-prefixed key sequence consists of the character with the eighth bit zeroed, precedes by the meta-prefix character, usually escape, which selects an alternate keymap), to decide whether to output characters with the eighth bit set directly or as a meta-prefixed key sequence, whether or not to wrap to a new screen line when a line being edited is longer than the screen width, the keymap to which subsequent key bindings should apply, or even what happens when readline wants to ring the terminal's bell. All of these variables can be set in the `inputrc` file.

The startup file understands a set of C preprocessor-like conditional constructs which allow variables or key bindings to be assigned based on the application using readline, the terminal currently being used, or the editing mode. Users can add program-specific bindings to make their lives easier. I have bindings that let me edit the value of `PATH` and double-quote the current or previous word:

```bash
# Macros that are convenient for shell
# interaction
$if Bash
# edit the path
"\C-xp": "PATH=$PATH\e\C-e\C-a\ef\C-f"
# prepare to type a quoted word -- insert
# open and close double quotes and move to
# just after the open quote
"\C-xO": "\\"\\"\\C-b"
# Quote the current or previous word
"\C-xq": "\eb\"\ef""
$endif
```

There is a readline command to re-read the file, so users can edit the file, change some bindings, and begin to use them almost immediately.

Bash implements the `bind` built-in for more dynamic control of readline than the startup file permits. `Bind` is used in several ways. In `list` mode, it can display the current key bindings, list all the readline editing directives available for binding, list which keys invoke a given directive, or output the current set of key bindings in a format that can be incorporated directly into an `inputrc` file. In `batch` mode, it reads a series of key bindings directly from a file and passes them to readline. In its most common usage, `bind` takes a single string and passes it directly to readline, which interprets the line as if it had just been read from the `inputrc` file. Both key bindings and variable assignments may appear in the string given to `bind`.

The readline library also provides an interface for word completion. When the completion character (usually TAB) is typed, readline looks at the word currently being entered and computes the set of filenames of which the current word is a valid prefix. If there is only one possible completion, the rest of the characters are inserted directly, otherwise the common prefix of the set of filenames is added to the current word. A second TAB character entered immediately after a non-unique completion causes readline to list the possible completions; there is an option to have the list displayed immediately. Readline provides hooks so that applications can provide specific types of completion before the default filename completion is attempted. This is quite flexible, though it is not completely user-programmable. Bash, for example, can complete filenames, command names (including aliases, built-ins, shell reserved words, shell functions, and executables found in the file system), shell variables, usernames, and hostnames. It uses a set of
heuristics that, while not perfect, is generally quite good at determining what type of completion to attempt.

**History**

Access to the list of commands previously entered (the `command history`) is provided jointly by Bash and the readline library. Bash provides variables (`$HISTFILE`, `$HISTSIZE`, and `$HISTCONTROL`) and the `history` and `fc` built-ins to manipulate the history list. The value of `$HISTFILE` specifies the file where Bash writes the command history on exit and reads it on startup. `$HISTSIZE` is used to limit the number of commands saved in the history. `$HISTCONTROL` provides a crude form of control over which commands are saved on the history list: a value of `ignorespace` means to not save commands which begin with a space; a value of `ignorecase` means to not save commands identical to the last command saved. `$HISTCONTROL` was named `$history_control` in earlier versions of Bash; the old name is still accepted for backwards compatibility.

The `history` command can read or write files containing the history list and display the current list contents. The `fc` built-in, adopted from POSIX.2 and the Korn Shell, allows display and re-execution, with optional editing, of commands from the history list. The readline library offers a set of commands to search the history list for a portion of the current input line or a string typed by the user.

Finally, the `history` library, generally incorporated directly into the readline library, implements a facility for history recall, expansion, and re-execution of previous commands very similar to `csh` ("bang history", so called because the exclamation point introduces a history substitution):

```
$ echo a b c d e
  a b c d e
$ ! f g h i
  echo a b c d e f g h i
  a b c d e f g h i
$ !-2
  echo a b c d e
    a b c d e
$ echo !-2:1-4
  echo a b c d
    a b c d
```

The command history is only saved when the shell is interactive, so it is not available for use by shell scripts.

**New Shell Variables**

There are a number of convenience variables that Bash interprets to make life easier. These include `IGNORE`, which is a set of filename suffixes identifying files to exclude when completing filenames; `HOSTTYPE`, which is automatically set to a string describing the type of hardware on which Bash is currently executing; `command_oriented_history`, which directs Bash to save all lines of a multiple-line command such as a `while` or `for` loop in a single history entry, allowing easy re-editing; and `IGNOREEOF`, whose value indicates the number of consecutive EOF characters that an interactive shell will read before exiting – an easy way to keep yourself from being logged out accidentally. The `auto_resume` variable alters the way the shell treats simple command names: if job control is active, and this variable is set, single-word simple commands without redirections cause the shell to first look for and restart a suspended job with that name before starting a new process.

**Brace Expansion**

Since `sh` offers no convenient way to generate arbitrary strings that share a common prefix or suffix (filename expansion requires that the filenames exist), Bash implements `brace expansion`, a capability picked up from `csh`. Brace expansion is similar to filename expansion, but the strings generated need not correspond to existing files. A brace expression consists of an optional `preamble`, followed by a pair of braces enclosing a series of comma-separated strings, and an optional `postamble`. The preamble is prepended to each string within the braces, and the postamble is then appended to each resulting string:

```
$ echo a[0,1,2]
   ade ace abe
```

As this example demonstrates, the results of brace expansion are not sorted, as they are by filename expansion.

**Process Substitution**

On systems that can support it, Bash provides a facility known as `process substitution`. Process substitution is similar to command substitution in that its specification includes a command to execute, but the shell does not collect the command's output and insert it into the command line. Rather, Bash opens a pipe to the command, which is run in the background. The shell uses named pipes (FIFOs) or the `/dev/fd` method of naming open files to expand the process substitution to a filename which connects to the pipe when opened. This filename becomes the result of the expansion. Process substitution can be used to compare the outputs of two different versions of an application as part of a regression test:

```
$ cmp <(old_prog) <(new_prog)
```

**Prompt Customization**

One of the more popular interactive features that Bash provides is the ability to customize the prompt. Both `$PS1` and `$PS2`, the primary and secondary prompts, are expanded before being displayed. Parameter and variable expansion is
performed when the prompt string is expanded, so any shell variable can be put into the prompt (e.g., $SHELLV, which indicates how deeply the current shell is nested). Bash specially interprets characters in the prompt string preceded by a backslash. Some of these backslash escapes are replaced with the current time, the date, the current working directory, the username, and the command number or history number of the command being entered. There is even a backslash escape to cause the shell to change its prompt when running as root after an su.

Before printing each primary prompt, Bash expands the variable $PROMPT_COMMAND and, if it has a value, executes the expanded value as a command, allowing additional prompt customization. For example, this assignment causes the current user, the current host, the time, the last component of the current working directory, the level of shell nesting, and the history number of the current command to be embedded into the primary prompt:

```bash
$ PS1='\u@\h [\t] \W($SHELLV:\d) $'
```

The string being assigned is surrounded by single quotes so that if it is exported, the value of $SHELLV will be updated by a child shell:

```bash
chet@odin [21:03:44] $ documentation (2:636)$
cd ..
chet@odin [21:03:54] $ src (2:637)$
```

The \$ escape is displayed as "$" when running as a normal user, but as "#" when running as root.

**File System Views**

Since Berkeley introduced symbolic links in 4.2 BSD, one of their most annoying properties has been the "warping" to a completely different area of the file system when "cd." The UNIX kernel treats symbolic links physically. When the kernel is translating a pathname in which one component is a symbolic link, it replaces all or part of the pathname while processing the link. If the contents of the symbolic link begin with a slash, the kernel replaces the pathname entirely; if not, the link contents replace the current component. In either case, the symbolic link is visible. If the link value is an absolute pathname, the user finds himself in a completely different part of the file system.

Bash provides a logical view of the file system. In this default mode, command and filename completion and built-in commands such as cd and pushd which change the current working directory transparently follow symbolic links as if they were directories. The $PWD variable, which holds the shell's idea of the current working directory, depends on the path used to reach the directory rather than its physical location in the local file system hierarchy. For example:

```bash
$ cd /usr/local/bin
$ echo $PWD
/usr/local/bin
$ pwd
/usr/local/bin
$ /bin/pwd
/net/share/sun4/local/bin
$ cd ..
$ pwd
/usr/local
$ /bin/pwd
/net/share/sun4/local
$ cd ..
$ pwd
/usr
$ /bin/pwd
/usr
```

One problem with this, of course, arises when programs that do not understand the shell's logical notion of the file system interpret ".." differently. This generally happens when Bash completes filenames containing ".." according to a logical hierarchy which does not correspond to their physical location. For users who find this troublesome, a corresponding physical view of the file system is available:

```bash
$ cd /usr/local/bin
$ pwd
/usr/local/bin
$ set -o physical
$ pwd
/net/share/sun4/local/bin
```

**Internationalization**

One of the most significant improvements in version 1.13 of Bash was the change to "eight-bit cleanliness." Previous versions used the eighth bit of characters to mark whether or not they were quoted when performing word expansions. While this did not affect the majority of users, most of whom used only seven-bit ASCII characters, some found it confining. Beginning with version 1.13, Bash implemented a different quoting mechanism that did not alter the eighth bit of characters. This allowed Bash to manipulate files with "odd" characters in their names, but did nothing to help users enter those names, so version 1.13 introduced changes to readline that made it eight-bit clean as well. Options exist that force readline to attach no special significance to characters with the eighth bit set (the default behavior is to convert these characters to meta-prefixed key sequences) and to output these characters without conversion to meta-prefixed sequences. These changes, along with the expansion of keymaps to a full eight bits, enable readline to work with most of the ISO-8859 family of character sets, used by many European countries.
POSIX Mode

Although Bash is intended to be POSIX.2 conformant, there are areas in which the default behavior is not compatible with the standard. For users who wish to operate in a strict POSIX.2 environment, Bash implements a POSIX mode. When this mode is active, Bash modifies its default operation where it differs from POSIX.2 to match the standard. POSIX mode is entered when Bash is started with the -p option. This feature is also available as an option to the set built-in, set -o posix. For compatibility with other GNU software that attempts to be POSIX.2 compliant, Bash also enters POSIX mode if the variable $POSIXLY_CORRECT is set when Bash is started or assigned a value during execution. $POSIX_PEDANTIC is accepted as well, to be compatible with some older GNU utilities. When Bash is started in POSIX mode, for example, it sources the file named by the value of $ENV rather than the "normal" startup files, and does not allow reserved words to be aliased.

New Features and Future Plans

There are several features introduced in the current version of Bash, version 1.14, and a number under consideration for future releases. This section will briefly detail the new features in version 1.14 and describe several features that may appear in later versions.

New Features in Bash-1.14

The new features available in Bash-1.14 answer several of the most common requests for enhancements. Most notably, there is a mechanism for including non-visible character sequences in prompts, such as those which cause a terminal to print characters in different colors or in standout mode. There was nothing preventing the use of these sequences in earlier versions, but the readline redisplay algorithm assumed each character occupied physical screen space and would wrap lines prematurely.

Readline has a few new variables, several new bindable commands, and some additional emacs mode default key bindings. A new history search mode has been implemented: in this mode, readline searches the history for lines beginning with the characters between the beginning of the current line and the cursor. The existing readline incremental search commands no longer match identical lines more than once. Filename completion now expands variables in directory names. The history expansion facilities are now nearly completely csh-compatible: missing modifiers have been added and history substitution has been extended. Several of the features described earlier, such as "set -o posix" and $POSIX_PEDANTIC, are new in version 1.14. There is a new shell variable, OSType, to which Bash assigns a value that identifies the version of UNIX it's running on (great for putting architecture-specific binary directives into the $PATH). Two variables have been renamed: $HISTCONTROL replaces $history_control, and $HOSTFILE replaces $hostname_completion_file. In both cases, the old names are accepted for backwards compatibility.

The ksh select construct, which allows the generation of simple menus, has been implemented. New capabilities have been added to existing variables: $auto_resume can now take values of exact or substring, and $HISTCONTROL understands the value ignoreboth, which combines the two previously acceptable values. The dirs built-in has acquired options to print out specific members of the directory stack. The $noLink variable, which forces a physical view of the file system, has been superseded by the -P option to the set built-in (equivalent to set -o physical); the variable is retained for backwards compatibility. The version string contained in $BASH_VERSION now includes an indication of the patch level as well as the "build version."

Some little-used features have been removed: the bye synonym for exit and the $NO_PROMPT_VARS variable are gone. There is now an organized test suite that can be run as a regression test when building a new version of Bash.

The documentation has been thoroughly overhauled: there is a new manual page on the readline library and the info file has been updated to reflect the current version. As always, as many bugs as possible have been fixed, although some surely remain.

Other Features

There are a few features that I hope to include in later Bash releases. Some are based on work already done in other shells.

In addition to simple variables, a future release of Bash will include one-dimensional arrays, using the ksh implementation of arrays as a model. Additions to the ksh syntax, such as varname=( ... ) to assign a list of words directly to an array and a mechanism to allow the read built-in to read a list of values directly into an array, would be desirable.

Given those extensions, the ksh "set -A" syntax may not be worth supporting (the -A option assigns a list of values to an array, but is a rather peculiar special case).

Some shells include a means of programmable word completion, where the user specifies on a per-command basis how the arguments of the command are to be treated when completion is attempted: as filenames, hostnames, executable files, and so on. The other aspects of the current Bash implementation could remain as-is; the existing heuristics would still be valid. Only when completing the arguments to a simple command would the programmable completion be in effect.
FEATURE

It would also be nice to give the user finer-grained control over which commands are saved onto the history list. One proposal is for a variable, tentatively named $HISTIGNORE, which would contain aColon-separated list of commands. Lines beginning with these commands, after the restrictions of $HISTCONTROL have been applied, would not be placed onto the history list. The shell pattern-matching capabilities could also be available when specifying the contents of $HISTIGNORE.

One thing that newer shells such as wksh (also known as diksh) provide is a command to dynamically load code implementing additional built-in commands into a running shell. This new built-in would take an object file or shared library implementing the “body” of the built-in (xxx_builtin/ for those familiar with Bash internals) and a structure containing the name of the new command, the function to call when the new built-in is invoked (presumably defined in the shared object specified as an argument), and the documentation to be printed by the help command (possibly present in the shared object as well). It would manage the details of extending the internal table of built-ins.

A few other built-ins would also be desirable: two are the POSIX.2 getconf command, which prints the values of system configuration variables defined by POSIX.2, and a disown built-in, which causes a shell running with job control active to “forget about” one or more background jobs in its internal jobs table. Using getconf, for example, a user could retrieve a value for $PATH guaranteed to find all of the POSIX standard utilities, or find out how long filenames may be in the file system containing a specified directory.

There are no implementation timetables for any of these features, nor are there concrete plans to include them. If anyone has comments on these proposals, feel free to send me electronic mail.

Reflections and Lessons Learned

The lesson that has been repeated most often during Bash development is that there are dark corners in the Bourne Shell, and people use all of them. In the original description of the Bourne shell, quoting and the shell grammar are both poorly specified and incomplete; subsequent descriptions have not helped much. The grammar presented in Bourne’s paper describing the shell distributed with the Seventh Edition of UNIX is so far off that it does not allow the command who|wc. In fact, as Tom Duff states:

Nobody really knows what the Bourne shell’s grammar is. Even examination of the source code is little help.†

The POSIX.2 standard includes a yacc grammar that comes close to capturing the Bourne shell’s behavior, but it disallows some constructs which sh accepts without complaint—and there are scripts out there that use them. It took a few versions and several bug reports before Bash implemented sh-compatible quoting, and there are still some “legal” sh constructs which Bash flags as syntax errors. Complete sh compatibility is a tough nut.

The shell is bigger and slower than I would like, though the current version is substantially faster than previously. The readline library could stand a substantial rewrite. A handwritten parser to replace the current yacc-generated one would probably result in a speedup, and would solve one glaring problem: the shell could parse commands in “$(...)” constructs as they are entered, rather than reporting errors when the construct is expanded.

As always, there is some chaff to go with the wheat. Areas of duplicated functionality need to be cleaned up. There are several cases where Bash treats a variable specially to enable functionality available another way ($notify vs. set -o notify and $nolinks vs. set -o physical, for instance); the special treatment of the variable name should probably be removed. A few more things could stand removal; the $allow_null_glob_expansion and $glob_dot_filenames variables are of particularly questionable value. The $[...] arithmetic evaluation syntax is redundant now that the POSIX-mandated $(...) construct has been implemented, and could be deleted. It would be nice if the text output by the help built-in were external to the shell rather than compiled into it. The behavior enabled by $command_oriented_history, which causes the shell to attempt to save all lines of a multi-line command in a single history entry, should be made the default and the variable removed.

Availability

As with all other GNU software, Bash is available for anonymous FTP from prep.ai.mit.edu:/pub/gnu and from other GNU software mirror sites. The current version is in bash-1.14.2.tar.gz in that directory. Use Archie to find the nearest archive site. The latest version is always available for FTP from bash.CWRU.Edu:/pub/dist. Bash documentation is available for FTP from bash.CWRU.Edu:/pub/bash.

The Free Software Foundation sells tapes and CD-ROMs containing Bash; send electronic mail to: <gnu@prep.ai.mit.edu> or call +1-617-876-3296 for more information.

Bash is also distributed with several versions of UNIX-compatible systems. It is included as /bin/sh and /bin/bash on several Linux distributions (more about the difference in a moment), and as contributed software in BSD’s BSD/386* and FreeBSD.

The Linux distribution deserves special mention. There are two configurations included in the standard Bash distribution: a “normal” configuration, in which all of the standard features are included, and a “minimal” configuration, which omits job control, aliases, history and command line editing, the directory stack and pushd/popt/dirs, process substitution, prompt string special character decoding, and the select construct. This minimal version is designed to be a drop-in replacement for the traditional UNIX /bin/sh, and is included as the Linux /bin/sh in several packagings.

Conclusion
Bash is a worthy successor to sh. It is sufficiently portable to run on nearly every version of UNIX from 4.3BSD to SVR4.2, and several UNIX workalikes. It is robust enough to replace sh on most of those systems, and provides more functionality. It has several thousand regular users, and their feedback has helped to make it as good as it is today – a testimony to the benefits of free software.

Business Survival Skills
101: The Interview
by Scott Hazen Mueller
<scott@zorch.sf-bay.org>

Introduction
All of us, at some time or another, go through the process of interviewing for a position. For many this is a painful time, fraught with fear and anxiety. For others, it is a chance to strut their stuff in a one-on-one situation. Nearly everyone needs to polish their skills and style for interviewing. I hope to give the reader some hints gleaned from my own reading, a number of successful (and a few otherwise) interviews, and the many times I have interviewed job candidates.

Step 1: Getting in the door
The first and most important step is getting in the door in the first place. Rather than offer the same tired advice available from any placement office, I’ll supply some things I have seen in industry that have made a major difference. First and foremost, spell- and grammar-check and correct your resume. Unless you are applying for a position that requires absolutely no ability to communicate, you must show that you are able to write correctly and clearly when it is important. Getting a job is a very important activity, and a sloppy resume will not win you any interviews.

Second, don’t apply for positions for which you are not qualified. This especially applies to anyone who blanket-replies to all postings in Usenet newsgroups. Not only will you not be considered for the job, you may annoy potential employers who may not have considered you for any position at that firm.

Feel free to add information to your resume that you feel is relevant to a particular position. The rule that you should keep your resume to one page only applies to recent graduates. I expect experienced people to have longer resumes, to document that they have the experience they should have.

No one can guarantee that you will get an interview, but a well-prepared resume can help your odds considerably.

Step 2: Preparing for the interview
Again, I will bypass the advice typically given to interviewees. While it cannot hurt your chances to study your potential employer before the interview, it will not significantly hurt your chances either if you don’t. For one, when asked “Do you know what we do?” you can always respond, “Well, I’m familiar with the firms in your field, but I have not had a chance to use/buy/install your product.” This is a conversational hook that invites the interviewer to tell you about their firm, without your having appeared to be ignorant.

I very much recommend bringing some questions with you to ask the interviewers. Not only do you then impress them with your preparation, you help guarantee that you will leave the interview with vital pieces of information. Many times your questions will be answered in the course of the interview process without your asking them, but it does not hurt to bring a list. Typical questions are:

- What will I be doing?
- Who will I report to?
- What hours will I typically work?
- Will I have to carry a pager or be on call (this does sometimes apply to programmers)?
- What are the medical benefits?
- What are the dental benefits?
How To Buy Good Clothes

There is a lot more to buying good clothing than just taking a suit off a rack and paying for it. Since I don't buy women's businesswear, I will only talk about men's clothing here.

While I won't say you should shop at one for your everyday needs, I strongly suggest you buy your interviewing suit from an upscale clothier such as Nordstrom or another well-respected outfitter. The reason is that unless you're already an excellent dresser, you will want the assistance of the sales staff at such a store. Yes, they are trying to get you to spend money, but this is the sort of time you should plan on doing so. If you need to be fully outfitted (suit, tie, shirt, shoes and belt), plan on spending in the neighborhood of $700, more if you want to buy really good clothes.

There are two important criteria when putting together an ensemble: one, the colors should complement each other, and two, the clothing should fit comfortably. The number one reason why people wearing ties seem on the verge of strangulation is poorly-fitted shirts. If your shirt is an inch or two too small, you too will struggle when you close your collar buttons to put on your tie.

If you don't have good color judgement, but have access to someone who does, bring them with you. You can probably rely on the salesperson, but you might not be comfortable doing so, especially if you don't shop for good clothing very often. At a good clothier, you can usually start in the suit department and have the clerk there not only provide you your suit, but also assist you in the other departments as you obtain your accessories. They will be more than happy to help you color-match a shirt, tie, shoes, and belt with the suit.

Guaranteeing fit is more difficult if you're not used to it, but you can make it easier. Spend some time looking at people in suits and observe how they fit. As a conservative dresser, I have my trousers hemmed so the crease just breaks when I stand straight, but you may prefer them a bit higher; decide what you like and go with that.

FEATURE

- What are the other benefits?
- What are the vacation, holiday and sick leave policies?
- Is there a continuing education program?

Of course, add anything you find personally important.

An often-overlooked piece of information to have ready is your salary requirement. You should keep yourself familiar with typical pay scales for people in your field and region with your level of skills and experience. Also, think about what it will take to get you to change jobs, so that you can answer related questions quickly and easily.

I will echo one other piece of traditional advice: Dress well for your interview! I realize there are many in our field who disdain conservative clothing such as business suits, and there are even a few firms that won't hire a candidate who wears a suit to an interview, but the proper attire for an interview is a suit. If you can't convince yourself of that, at least think of it as the lowest common denominator, the one way you can dress that will be correct in the greatest number of cases. (For some advice on selecting your suit, see the side-bar.) This applies to men and women, though women do have more flexibility in clothing, and a pantsuit or dress would also be appropriate for a woman. Wearing a suit shows that you care enough about the interview to make an effort to show your best side. It also shows that you respect the firm and the people with whom you are meeting.

There are those few firms that conform to a specific alternate dress-code, the "UNIX guru" look (jeans, tee-shirt and sandals), and won't hire a candidate who wears a suit to the interview. If you don't mind this sort of environment, feel free to dress down. If you are not sure about a specific firm, and want to try to tailor your outfit to them, go to their office at about closing time and watch the people who come out. (If no one comes out, leave and do not come back!) Watch the people who get into the more expensive cars, and if none of them is wearing a suit, then you might want to dress more casually to interview with this firm.

Most important, show up on time! The best time to arrive for your interview is 5 minutes or less before the scheduled time. If you arrive late, you have just demonstrated that you cannot manage time well, and whether that judgement is right or wrong, that is what the interviewer will likely think. If you arrive more than a very few minutes early, you will look over-eager, and possibly interrupt and annoy your interviewer. My standard practice is to arrive in the parking lot with sufficient time to tie my tie, check myself over, relax a few minutes, and still walk into the reception area 2-3 minutes before the scheduled time. One of the best ways to make sure you arrive for the interview on time is to go to the interview site before the interview, if you are not familiar with the area. Almost nothing looks worse than getting lost on the way to the interview, especially if it makes you late. If that is not practical for you, make certain you get clear directions, and then when you do go to the interview, allot plenty of time for travel.

Dealing with pre-interview stress is very important. Many people don't deal well with their nerves, and interview poorly as a result. When I was in college, I had some problems with pre-test anxiety. I learned some simple relaxation exercises then, and still apply them now when I interview. If you are nervous, sit someplace comfortable, close your eyes and visualize a pleasant but neutral scene, such as a meadow. Take a deep breath, hold it a second, and exhale.
deeply, pursing your lips slightly as you release the air. Repeat several times. The oxygenation in combination with the relaxing thoughts alleviates many of the symptoms of stress.

You may also have to deal with sweaty palms; I do. My tactic is quite simple: I carry a folded napkin in the pocket of my suit jacket, and I dry my right hand before I meet each new person. This way my hand is dry and warm for the traditional introductory handshake. This symbolic touching is quite important in the interview process, and the impression you make here is nearly as important as the actual interview. If benefits are poorer than your current job, you ought to ask for more money to compensate, though I wouldn’t phrase it quite that way. If on the other hand, the job is one that interests you, or there are other compensatory factors, you can consider taking a lesser increase.

An important corollary is that for most job changes when you are staying within your field, you should expect a pay increase. In Silicon Valley a ten percent increase of current salary is common. At various times I’ve been offered increases ranging from 9% to 20%, with the latter being on my conversion from hourly with overtime to salaried. In nearly all salary discussions, your current salary is a starting point. That does not mean that you might not have to pick a figure out of thin air, but in the absence of other information, it would be reasonable to ask for 110% of your current salary. Best of all is if you can avoid naming a figure first; a good answer to a question about your salary requirement is just to state your current salary, and let them draw the obvious conclusion.

Obviously, if you are not currently working, or are about to stop working involuntarily, you may not have this sort of bargaining position. However, even if you are about to become unemployed, you can still bargain a bit, especially if you do not volunteer that information. Unless your situation is in the local newspaper, you are best off not discussing issues like a pending job cut.

Lastly, employers usually like to know when you will be available to start work for them. Most of the people I know consider two weeks’ notice to be standard, so that is an entirely reasonable answer. Sometimes your new employer will be quite eager for you to come on board, but I would still hold out for two weeks. It’s also not unreasonable to say that you need to give more notice, especially if you have considerable seniority at your job. That only helps to demonstrate that you are important where you currently work, and that in turn helps to convince the employer that you will be important to them, also.

**Step 4: The post-interview**

A variety of different activities occur during the post-interview period. A well-organized employer will usually tell you what to expect, such as “We’ll call you tomorrow and let you know if we’ll be making an offer.” If you are not clear on what happens next, ask. It cannot possibly hurt your chances to indicate that you want to close the loop.

Make sure that you follow through on any commitments you made during the interview. If you promised to supply references, codes samples, or whatever, get them to the employer in a timely fashion. Make sure you have the contact information you need, such as phone numbers and e-mail addresses.

Establish a timeframe with the employer if they don’t supply one. If you don’t hear back from them within the specified time, contact them and ask for an update. Even if you weren’t that interested in the job, don’t let it fall through the cracks; that looks unprofessional. Furthermore, if they come through with

You should have the clerks measure your arm and neck for your shirt, and your inseam and waist for your trousers. Your shirt sleeve length should come just to the base of your palm with your arm held to your waist, and the suit jacket should be about 1/2" shorter. Your collar should close easily and fit closely to your neck without being tight.

When you purchase a belt, you should be able to buckle it in one of the middle holes, and it should not gather your trousers at the waist when you do so. Your shoes should fit snugly; they shouldn’t pinch, but they shouldn’t slip either.

If you only rarely wear ties, you may have some difficulty remembering how to knot one. Many stores have pamphlets showing how to tie one or more of the common knots, which have names like four-in-hand or windsor. I use a knot that I think is a windsor knot, though I won’t swear to it.

Ties are very personal. Don’t let someone talk you into buying a tie you don’t like. I personally detest paisley. (Paisley is a pattern, not a color, and each individual element looks rather like the space blob from the old Star Trek.) If I wouldn’t wear paisley even if it would raise my salary $10K. At any rate, most stores have hundreds of ties, so you should have no difficulty selecting a tie that matches your suit and shirt, and that you like, or at least can stand for a few hours.

An interviewing outfit should be conservative and serious. If you are interviewing at IBM, you should of course wear dark blue, but dark grey, dark brown and black are also appropriate for most interviewing situations. If you are in a position to own multiple good suits, or just plain want to, you can fill out with more adventuresome combinations, though I advise staying away from powder blue unless you’re interviewing for a Lounge Lizard position.

It is also quite acceptable in many situations to wear a jacket and tie instead of an actual suit. A jacket also has the advantage of being more usable in other situations. Follow the same rules with a jacket as you would with a suit, and you should be just fine.
an offer, you can always turn it down, and it is better to turn down an offer you don’t want then to have never gotten the offer at all.

The hard salary bargaining usually comes around this time. If the employer was at all interested in you, they will have established a starting point during the interview, either by getting you to name a figure, or at least by finding out your current salary. The more experience and skills you have, the better you will come out in this bargaining. If you are fresh out of school, or don’t yet have much experience, you won’t be able to push very hard. You can also try to bargain for non-salary perks, such as extra time off, free equipment, and what-have-you.

Finally, it will come down to time for a written offer. Keep in mind that a verbal offer is not binding, so if you are working while you wait for that offer, just bide your time. Do not give notice at your current job until after you have received the written offer. Then, when you have the offer in hand, tell your prospective new employer that you would like a day or two to make your final decision.

You can then give notice to your current employer, and let them know you have the written offer in hand. You can also, if you like, tell them that you haven’t yet formally accepted it, and if both you and your current employer are willing, you may be able to discuss staying on with them. If you or they are not so inclined, you just accept the offer and serve your notice period.

**Conclusion**

No one can guarantee that someone else will have a successful interview. I have had plenty of interviews that were inconclusive, and others where it became clear early on that the employer and I would not be able to come to a meeting of minds on issues such as compensation. However, it is possible to go a long way towards ensuring that you don’t shoot yourself in the foot when interviewing, and thus, give yourself the best chance of actually getting the job.

**Taking Analogies to the Limit**

*by Michael Chastain*

<me@shell.portal.com>

[Michael generously gave us permission to republish this insightful scenario from rec.humor.funny...Ed.]

General Motors doesn’t have a help line for people who don’t know how to drive. Imagine if they did . . .

**HelpLine:** “General Motors HelpLine, how can I help you?”

**Customer:** “I got in my car and closed the door and nothing happened!”

**HelpLine:** “Did you put the key in the ignition slot and turn it?”

**Customer:** “What’s an ignition?”

**HelpLine:** “It’s a starter motor that draws current from your battery and turns over the engine.”

**Customer:** “Ignition? Motor? Battery? Engine? How come I have to know all these technical terms just to use my car?”

**HelpLine:** “General Motors HelpLine, how can I help you?”

**Customer:** “My car ran fine for a week and now it won’t go anywhere!”

**HelpLine:** “Is the gas tank empty?”

**Customer:** “Huh? How do I know?”

**HelpLine:** “There’s a little gauge on the front panel with a needle and markings from ‘E’ to ‘F’. Where is the needle pointing?”

**Customer:** “It’s pointing to ‘E’. What does that mean?”

**HelpLine:** “It means you have to visit a gasoline vendor and purchase some more gasoline. You can install it yourself or pay the vendor to install it for you.”

**Customer:** “What? I paid $12,000 for this car! Now you tell me that I have to keep buying more components? I want a car that comes with everything built in!”

**HelpLine:** “General Motors HelpLine, how can I help you?”

**Customer:** “Your cars suck!”

**HelpLine:** “What’s wrong?”

**Customer:** “It crashed, that’s what wrong!”

**HelpLine:** “What were you doing?”

**Customer:** “I wanted to run faster, so I pushed the accelerator pedal all the way to the floor. It worked for a while and then it crashed and it won’t start now!”

**HelpLine:** “It’s your responsibility if you misuse the product. What do you expect us to do about it?”

**Customer:** “I want you to send me one of the latest version that doesn’t crash any more!”

**HelpLine:** “General Motors HelpLine, how can I help you?”

**Customer:** “Hi, I just bought my first car, and I chose your car because it has automatic transmission, cruise control, power steering, power brakes, and power door locks.”

**HelpLine:** “Thanks for buying our car. How can I help you?”

**Customer:** “How do I work it?”
HelpLine: “Do you know how to drive?”
Customer: “Do I know how to what?”
HelpLine: “Do you know how to drive?”
Customer: “I’m not a technical person. I just want to go places in my car!”

Results of the International Computer Olympiad

by Don Piele
<piele@cs.uwp.edu>

[Ed. Note: USENIX sponsors part of the USA Computing Olympiad, which sends the four best high school programmers in the United States to the IOI (International Olympiads on Informatics), which is similar to the International Math Olympiad, International Chemistry Olympiad, et al.]

The US team finished the 1994 IOI as follows:

- Mahul Patel, USA 156 Gold Medal
- Brian Dean, USA 121 Silver Medal
- Hubert Chen, USA 95 Bronze Medal
- James Ayers, USA 91 Bronze Medal

Our team was one of four that won four medals. No team earned 200 points, since the second day’s problems were very difficult and required solution to three different problems.

Other countries that also placed high in the scoring included:

- Russia: 3 gold and 1 silver
  (the top score of 195 came from Russia)
- China: 3 gold and 1 bronze
- Germany: 2 gold and 1 silver and 1 bronze
- USA: 1 gold, 1 silver, 2 bronze
- Romania: 1 gold, 2 silver and 1 bronze
  (The USA total score was above Romania’s)
- Hungary: 2 gold, 1 bronze

We were all very proud of the team results.

Individual Top scores

- Viktor Bargatchev, Russia 195
- Walter Hoffman, Germany 170
- Roman Elisarov, Russia 167
- Garbor Kovacs, Hungary 165
- Adrian Soviani, Romania 165
- Martin Mares, Czech Republic 162
- Maxim Kouzmin, Bulgaria 161

In Sweden, the weather was ideal for the whole week, 70-75°F and always sunny. It was a very well run olympiad, probably the best so far. Over 180 students participated in the event from 48 countries from Argentina to Vietnam. The closing ceremonies were held at the City Hall of Stockholm, the setting of the Nobel Prize banquet. Our team was very impressed with Sweden and the 6th IOI. It was a fabulous week.
Edupage – The News
<edupage@ivory.educom.com>

[Editor's Note: As part of my ongoing quest to assemble quotes and short reviews of periodical literature, I approached John Gehl <gehl@ivory.educom.edu> at EDUCOM about re-printing his twice-weekly blurb that summarizes information technology items. He generously granted permission! Edupage is a twice-weekly summary of news items on information technology provided as a service by EDUCOM, a consortium of leading colleges and universities seeking to transform education through the use of information technology.

You can subscribe to the full, long-version printed offering of EDUCOM REVIEW. For individual first-time subscribers, a one year trial subscription to EDUCOM REVIEW is $18 (a 70% savings from the regular subscription price). Send check for $18 to EDUCOM Review, 1112 16th NW, Washington, DC 20036. Offer good only in US and Canada.]

Internet Backbone Set For Techno-Overhaul. The old NSFNet, the backbone of the Internet, will be taken out of commission this month as new equipment is phased in. Some users fear the changeover could cause technical glitches, but business users are hopeful that many of the stringent limits on advertising will not apply as forcibly on the new network. (The Internet Letter 8/1/94)

Compaq Puts Warnings On Keyboards. In response to concerns over repetitive stress injuries, Compaq Computer will begin placing warning labels on its keyboards urging users to read their booklet on safety and comfort guidelines for using the machines. Earlier this year, Compaq won what was believed to be the first jury verdict in a case attempting to link keyboard use to wrist injuries. (Wall Street Journal 8/17/94 B6)

Sega Plans 50 High-Tech Theme Parks. Building on the success of its two high-tech theme parks in Japan, Sega Enterprises plans to open up to 50 similar parks in the U.S. by the end of the century. The company says it already has 40 to 50 proposals for partnerships to build the parks. (Miami Herald 8/17/94 C3)

Ferraris Or Edsels On The I-Way? A Wall Street Journal editorial deplores the over-regulation contained in S. 1822, sponsored by Sen. Fritz Hollings, and comes out in favor of a draft bill being circulated by Sen. Bob Dole. On the Hollings bill: "The worst restrictions have to do with 'universal service,' which is so generously defined that companies may be forced to provide video on demand and other entertainment options free to impoverished customers." On the Dole bill: "His draft legislation would even repeal the 1992 Cable Act. Such sweeping deregulation is surely the only way the I-Way is likely to get built." (Wall Street Journal 8/17/94 A12)

America Online To Compete With The Internet. America Online has formed an alliance with publishing giant Simon & Schuster to form the "College Online" interactive computer service which would seek to become a commercial alternative to the Internet. AOL also announced that its subscriber base has passed the 1-million mark. (New York Times 8/17/94 C4)

Software Sales Statistics. First quarter sales figures from the Software Publishers Association indicate that compared to first quarter a year ago sales of programs written for Windows are up 42.9%, for DOS down by 32.3, for Macintosh up 17%. Overall, the biggest jump in a program category was home education software, up 127%; next was entertainment software, up 57%. (Atlanta Journal-Constitution 8/18/94 E2)

Interactive TV: Hit Or Hype? Truly sophisticated interactive television is still a gleam in most multimedia developers' eyes, and is likely to lose out to computer-based online services, according to some. "Communications over PC modem lines will be the growth area; everything else is spurious. Interactive TV will take 20 years to become more than a peripheral industry. Online services is millions of users – today – saying, "This is what we want and we're willing to pay for it.' Interactive TV, so far, is the gods of cable trying to figure out what people might want," says the president of Jupiter Communications, a New York research firm. (Information Week 8/29/94 p.12)

Not-Ready-For-Prime-Time Technologies. Handwriting recognition, voice recognition, navigational interfaces and business card scanners are some of the technologies cited in the August 22 PC Week magazine as too immature to be worth using in the next year. Nevertheless, one firm predicts that sales of voice automation applications will grow 35% every year for the next three years. (Investor's Business Daily 8/25/94 A3)

Holography-Based Storage. Researchers at Stanford University demonstrated a prototype storage device that uses holographic technology to hold massive amounts of computer data in a three-dimensional space the size of two sugar cubes. Hologram technology uses lasers to record images on crystal materials in the form of two-dimensional "pages." A holographic device could store 1,000 times more data than today's computers and retrieve the data 100 to 1,000 times faster. (Washington Post 8/22/94 p.19)

The Road Not Taken – Superhighway Services In Doubt. In a Macworld magazine survey, three-quarters of the 600 respondents said they were unwilling to pay even
$10 a month for any information superhighway service, except in-home learning. (Investor's Business Daily 8/25/94 A3)

Chicago Delayed Again. A report in InfoWorld says that Chicago, Microsoft's next version of its operating system for Windows, will be delayed until April or May of 1995 in order to resolve incompatibility problems with the current system. (Atlanta Journal-Constitution 8/28/94 F10)

"Dylan" Spells Legal Troubles For Apple. Bob Dylan has filed a trademark infringement suit against Apple Computer seeking unspecified damages and a court order to prevent Apple from calling its Dynamic Language software "Dylan." The software was developed about two years ago and is used to create programs for personal digital assistants. The irony is, of course, that Bob Dylan borrowed his own name from poet and author Dylan Thomas. (St. Petersburg Times 8/29/94 p.2)

New Dimensions In Bar Codes. Two-dimensional bar codes are on the horizon, and U.S. citizens will soon find them stamped on driver's licenses, and automobile titles and registrations, analysts predict. The new bar code system stores information in tiny black and white squares in a grid, as opposed to the one-dimensional stripes used now. The market for 2-D bar codes was about $1.2 million in 1993, and is expected to rise to $60 million by 1997. (Investor's Business Daily 8/29/94 A6)

Image Presents Recruiting Problems. Perceptions of computer careers as "nerdy" have caused serious concerns for recruiters in the software development industry in Canada, where a shortage of about 4,000 software professionals currently exists, but few students are taking notice. (Toronto Globe & Mail 8/30/94 C1)

Propeller Heads Need Not Apply. Forgetting the importance of the end user is a key failing of many network managers, says a Yankee Group analyst. "There are too many propeller heads in this business," says one network manager. "Drop the techy talk, don't talk to business executives about how it's done, but instead about what the benefits, cost savings, and strategic advantages are," suggests another. "Remember, you're managing strategic functions rather than a network," offers a third. (Telecommunications 8/4/94 p.31)

Registration Rush @InterNIC. Last month, InterNIC registered more than 2,000 new addresses, roughly double the monthly average earlier this year. Since the organization operates on a first-come-first-served basis, some companies are finding their names have already been assigned elsewhere. Recently, the Internet Co.'s had to return an address using the word "nasdaq" and one with "windows." (Washington Post 9/8/94 B11)

What's In A Name? Microsoft's long-delayed update to Windows 3.1 now has shed its Windy City codename "Chicago" and is now officially called Windows95. (Wall Street Journal 9/8/94 B4)

Turn Off That Computer And Go Outside To Play! Child psychologists are turning their attention to the long-term effects of "too much" PC use by today's computer generation. They suggest parents keep an eye on how their children are using the computer - whether it's just for games, or for more creative activities, and all point out that computer time cuts into kids' need for lots of exercise. (Washington Post 9/5/94 p.20)

Cultural Groups Oppose Sega Channel. Canadian cultural groups and broadcasters are concerned that a Sega video-game channel could undermine principles protecting Canadian ownership and content in broadcasting. (Ottawa Citizen 9/7/94 C7)

Are Computers Harmful To Education? Yale computer scientist David Geelenmeyer says that the computer's potential to do good is modestly greater than a book's in some areas, but that its potential to do harm is vastly greater, across the board. "While we bemoan the decline of literacy, computers discount words in favor of pictures and pictures in favor of video. While we fret about the decreasing cogency of public debate, computers dismiss linear argument and promote fast, shallow romps across the information landscape. While we worry about basic skills, we allow into the classroom software that will do a student's arithmetic or correct his spelling." (New Republic Sept. 19 & 26, 1994 p.14)

Hackers Beware. Forget midnight basketball: the real news in the crime bill is the Computer Abuse Amendments Act of 1994. The amendments significantly increase the chances of successfully prosecuting computer hackers by changing the standard from "intent" to "reckless disregard." In addition, previous laws protected only "federal interest computers" (machines belonging to a government agency or financial services firm). The new rules cover computers "used in interstate commerce," meaning any PC hooked up to the Internet. (Information Week 9/12/94 p.10)

1-2-3: Windows, Mac, DOS. The Software Publishers Association's sales figures for 2nd quarter 1994 are: Windows $1.05 billion, Mac OS $276 million, DOS $235 million. (Atlanta Journal-Constitution 9/15/94)

Freanets Suffer Setback. An archaic 1891 tax rule has slammed the doors on a major avenue of fund-raising for Canadian freanets. Revenue Canada ruled that freanets do not qualify as charitable organizations because they do not provide a location from which they can educate and, while they are vehicles for discussion, they do not offer ther-
Doing Digital Science. "Doing science" on computers – digital science – marks the most fundamental change in scientific methodology since Isaac Newton’s groundwork 350 years ago. In the near future, supercomputers will be used to digitally model how electrons, protons and neutrons can be combined to form new substances, such as drugs, polymers or metals. (Business Week 5/19/94 p.76)

Payday On The Internet. A new company called CyberCash hopes to break the impasse in conducting commercial transactions over the Internet by offering a secure electronic payment system. “We want to make the Internet safe for commerce... and provide safe passage from cyberspace into the banking world,” says the company’s president and co-founder. CyberCash will allow users to ask their banks to set aside money in a “digital purse.” The funds are then transferred to the merchant when an item is purchased. CyberCash would receive a small fee for each transaction. The company is still working on security measures, and must persuade banks that the system protects privacy and is tamper-proof. (Wall Street Journal 9/13/94 B1)

Electronic Achilles Heel. As e-mail is increasingly integrated into corporate culture, businesses are finding that saving messages can cost them a bundle in the courtroom. “e-mail is potentially disastrous for a company,” says a lawyer whose firm often subpoena e-mail during lawsuits. “Attorneys look at e-mail archives as a vast treasure trove,” says another. Experts advise users to routinely delete unnecessary files every 15 to 30 days, and some urge not backing up e-mail at all. “The key is to put a policy in place and tell employees about it,” says a manager at Lockheed. (Information Week 9/12/94 p.12)

Downteching. George Mason University professor Hugh Heclo calls for downteching – “deliberate reductions in the amount and/or sophistication of technology so as to improve performance,” and to counter the tendency of information technology to eliminate the natural limits on “blab” – undirected information pretending to be knowledge. (Atlanta Journal-Constitution 9/11/94 B1)

Secret Formula Posted To Internet. The RSA encryption software RC4 owned by RSA Data Security, Inc., has been anonymously and illegally posted to electronic bulletin boards on the Internet, perhaps compromising the software’s long-term effectiveness. (New York Times 9/17/94 p.17)

Mosaic Speeds Up. The Mosaic Communications company says that their commercial versions of Mosaic, to be available this Fall, will be more than 10 times faster than the public-domain version. (Internet Business Report, Sept. 94, p.1)

Cybernotaries? The American Bar Association is drawing up guidelines for online notaries public. The position is described as an “international computer-based transaction specialist.” (Information Week 9/19/94 p.8)

Schools Ask FCC To Fund On-Ramps. A coalition of education groups has urged the FCC to redirect $300 million in “Consumer Productivity Dividend” funds to building educational and library on-ramps to the information superhighway. The coalition consists of the American Library Association, the Council of Chief State School Officers, the National Association of Secondary School Principals, the National Education Association and the National School Boards Association. (The Heller Report 9/94 p.3)

E-mail Stalking Case. A Michigan man will be tried for pressing his attentions on a woman by e-mail and voice mail after she told him to leave her alone. By e-mail he told her: “You’ve turned my innocent and somewhat foolish love for you into something bad in your own mind.” Her e-mail response: “If you don’t leave me alone, you are going to be sorry. You have been warned.” (New York Times 9/16/94 B11)

Online Insults Cost Money. Santa Rosa Junior College has agreed to pay two female students $15,000 each in compensation for sexual remarks made about them on a male-only online bulletin board. The college admitted no wrongdoing, but has since done away with the men-only and women-only BBSs. (St. Petersburg Times 9/16/94 A3)

Black-Market Chips. The head of the San Jose police department’s high-tech crime unit says that computer chips are the dope of the ’90s. “They’re easier to steal than dope. Worth more money than dope. And for the people who steal them, here’s the best part: Once stolen, they’re almost untraceable.” (New York Times Magazine, 9/18/94 p.54)

Fraud On The Info-Highway. In its first case targeting online advertising, the Federal Trade Commission has charged a California man with making false claims in promoting his credit-repair program on America Online. The FTC alleges the steps he recommended are illegal. (Tampa Tribune 9/15/94 B&F8)

CD Could Be Reduced To A Dot. With a technique that uses an atomic force microscope, Matsushita has demonstrated the basic principles which could eventually increase computer memory storage capacity more than a thousand-fold and allowed the contents of a CD to be stored on a dot – the equivalent of a trillion bits in one square centimeter. However, a practical storage system using the technology is not expected until early in the next century. (New York Times 9/19/94)
Less Staff + More Technology = Longer Workdays. A survey conducted by the magazine Information Week found that 90% of respondents with mobile computing devices say that these devices have extended their workday. However, 60% attribute the longer workday to staff cuts, and say the mobile devices just make their lives more flexible in terms of where they perform this work. (Information Week 9/19/94 p.34)

Campus Connections Card Does It All. MCI Business Markets is now offering a Campus Connections card with a double magnetic strip that accommodates both point-of-sale and debit transactions. The card functions as a college ID card, library card, ATM card and telephone calling card. (The Heller Report 9/94 p.7)

RSA Data Says Don’t Panic. Despite the recent dissemination over the Internet of a software code used to safeguard the privacy of electronic messages, the code’s maker, RSA Data Security, says in no way is the security of existing computer networks threatened. The encryption code RC4 is used in numerous popular software programs, including those of Apple, Lotus and Microsoft. Officials at the U.S. Customs Service and the FBI are investigating the anonymous posting. (Wall Street Journal 9/15/94 B10)

WWW Over e-mail. CERN, the European research group that developed the World-Wide Web, now makes it possible for people to get Web pages via e-mail. Send a message to <listproc@www.cern.ch> and in the body of the message type the Universal Resource Locator for the Web page you want. (Chronicle of Higher Education 9/21/94 A25)

Be A Nuisance, Get Attacked, Get Famous, Publish A Book. Laurence Canter and Martha Siegel, the lawyers who were singed by thousands of flames for using Usenet bulletin boards to widely post ads for their legal services, are publishing a WarperReference book called “How To Make A Fortune On The Information Superhighway.” All the furor about Canter and Siegel (including calls for the book to be stopped) has encouraged the publisher to increase its first printing from: 25,000 to 100,000 copies. (New York Times 9/21/94 B2)

Maxing Out On 800 Numbers. If the growth rate in 800 numbers continues its current climb, it will reach saturation point in two years. A likely solution, according to the Industry Numbering Committee, will be to supplement 800 with 300 or 400 numbers. (Business Week 9/26/94 p.8)

Software Pirates Walk The Plank. Two BBS operators in Montreal and Toronto were fined a total of $22,500 for the illegal distribution of software. The Canadian Alliance Against Software Theft said the ruling sends an important message that piracy will not be tolerated. (Toronto Globe & Mail 9/21/94 B5)

Authors Paid By Connect Time. Authors who used to be paid by the word are to be paid by connect time. A Fountain Valley, California company called Interact will allow authors of books, pictures and programs to place material on their service for downloading at a fixed charge. Modem number is 714-378-4704. Voice number is 800-865-9000. (Atlanta Journal-Constitution 9/22/94 K2)

Mosaic’s Detractors Fear Loss Of Control. Meanwhile, Mosaic’s detractors say the program puts too much control in the hands of readers at the expense of publishers of electronic journals and newspapers. Editors can never be sure that the layout they design will look the same on every reader’s screen, because the program can be individually configured to display different fonts and sizes. This causes particular problems for newspaper editors, who have spent their careers worrying about headline sizes and what stories to place “above the fold.” (Internet World 10/94 p.81)

Record Your Own CD-ROM. Matsushita Electric Industrial Co. has developed a CD-ROM player that can record up to 650 megabytes of data on erasable disks. Prototype players are selling for $1,000 apiece and recordable disks go for $100. Those prices are expected to come down considerably when volume sales go up. (Investor’s Business Daily 9/22/94 A14) [My CD-ROM recorder cost $3,000 and has 11 media (!)]

IP Agenda. The Internet Protocol next-generation group has worked out the details for the next iteration of IP, called IPv6: addressing space will expand from 4 to 16 bytes, to ease the Internet’s impending address crunch; user authentication will assure receivers that a given IP packet comes from the source it claims; an autoconfiguration feature will simplify the translation between the 48-bit MAC addresses used by Ethernet and token ring networks and 32-bit IP addresses; and a 28-bit flow label will flag packets carrying delay-sensitive data such as audio and video. (Data Communications 9/94 p.16)

College BBSs Not Protected Under First Amendment? College administrators across the country are waiting for the final outcome of the Santa Rosa Community College case, where two female students who’d been insulted on a male-only bulletin board were paid $15,000 each, in addition to the male student who told them about the remarks. The college is negotiating with the U.S. Department of Education’s Office for Civil Rights regarding changes in Santa Rosa’s computing policies that would penalize conduct contributing to a “hostile, intimidating, or offensive educational environment.” OCR rejected claims by the college that the remarks were protected under the First Amendment, saying the computer conference was not a public forum, but rather an “educational program.” (Chronicle of Higher Education 9/28/94 A26)
Test Scores Are In The Eye Of The Beholder. Test essays written by hand often are scored higher than those submitted in word-processed versions, concludes a study by the Educational Testing Service. When word-processed essays were converted to handwriting and vice versa, researchers found the hand-written versions of word-processed essays were scored higher than the originals, even though the content was exactly the same. One possible reason is that word-processed essays appear shorter and errors are more obvious. (Chronicle of Higher Education 9/28/94 A25)

Naming Names On The Net. The Internet Network Information Center has clamped down on requests from companies for more than one name on the ‘net, acknowledging “massive problems” of misappropriation of commercial names by unrelated firms and individuals. Some Internet observers say it’s too little, too late. InterNIC “is taking a Band-Aid approach. Unless there is some centralized way of stopping this, it is always going to occur,” says the president of Internet Info. (Wall Street Journal 9/28/94 B16)

Wiretap Legislation Clears Committee. A bill requiring telephone companies to ensure their networks are still accessible for wiretapping by law enforcement agencies was approved by the House Judiciary Committee. A similar measure passed the Senate Judiciary Committee earlier in the week, and the legislation may be voted on before Congress adjourns. The bill requires the federal government to reimburse phone companies for the trouble of retrofitting existing equipment, to the tune of $500 million. Who pays the costs incurred beyond this amount is still up in the air, and the FCC may end up deciding whether government, or industry, or both are responsible. (Wall Street Journal 9/30/94 B5)

Electronic Fix For IP On The Info-Highway. A task force chaired by Patent and Trademark commissioner Bruce Lehman advocates the use of file headers and electronic “envelopes” for strengthening intellectual property protection on the information superhighway. The envelopes would spell out ownership, licensing terms, and conditions pertaining to works available via the NII. (Information Week 10/3/94 p.38)

National Cryptography Policy. The National Research Council is conducting a comprehensive study of national cryptography policy, including such topics as: the availability of cryptography technology to foreign and domestic parties; the competitiveness of U.S. manufacturers and users of such technology; U.S. national security and law enforcement interests; relative merits of various cryptographic technologies; demand for information systems security based on cryptography; the impact of foreign restrictions; the extent to which current policy is adequate for protecting U.S. interests; relative merits of current key escrow implementation schemes; feasible policy options; and recommendations for the process through which all interests are balanced in the formulation of national cryptography policy. Send comments and other correspondence to <crypto@nas.edu>. (NRC Release)

Sprint To Build Fiber-Optic Network. Sprint Canada will spend $100-million to build a high-volume intercity cable link between Toronto, Ottawa and Montreal along Canadian Pacific Railway right-of-ways. (Montreal Gazette 10/4/94 D1)

On The Road To The Global Village. Judy McGrath, creative head of MTV: “I guess my job is to keep us focused on a vision. Is the global thing the MTV vision, is it that we’re not like other TV networks, is it that we are a TV network for youth? What is the vision? Every now and then, I say we need to start stressing the global thing, or whatever, and the staff throws up all over it, which is the right thing to do.” (New Yorker 10/10/94 p.64)

Billing By The Byte. Internet users in Australia face a new system of charges in January that could cost them as much as $1.50 (U.S.) for every megabyte of information they send or receive. Currently, faculty members pay nothing, and universities pay a fixed annual fee based on the operating grant they receive from the federal government. Details are still being worked out, but AARNET says the change is necessary to upgrade the network: “The increase in capacity is in response to the apparently insatiable demand for Internet access in Australia...” says AARNET’s general manager. (Chronicle of Higher Education 10/3/94 A23)

Data Compression Downers. Critics of compressed video say the images produced are too harsh. Analog video provides more subtle gradations, because the technology doesn’t require that picture be interpreted only in terms of 1s and 0s. Reliability can also be a problem. When an analog picture degrades, it does so gradually, often enabling the viewer to continue watching a compromised image. Digital pictures respond to interference with a sudden freeze, or disappear altogether. (Investors Business Daily 10/6/94 A3)

Congress Passes Wiretap Bill. Congress has passed Digital Telephony Legislation (HR 4922/S 2375) which will force phone companies to make digital telephone networks accessible to wiretaps by law enforcement agencies with court-ordered warrants. The bill, which does not apply to online services or to the Internet, also provides increased protection for online personal information and restricts the use of pen registers to track the physical location of individuals. (New York Times 10/9/94 p.15)
Education – The New Killer App? A study, conducted by FIND/SVP and Grunwald Associates, is seeking to define the market for technology-based educational products and services for the home. The project was prompted by findings that almost half of American households with children and/or teenagers own PCs, many including modems and CD-ROM drives. While educational software sales jumped 127% in the first quarter of 1994, compared to the prior year, “Software decision makers are frustrated by the absence of good strategic information on what households want – and what they will pay for,” says Peter Grunwald.

The Continental Perspective. European technology managers approach their jobs differently from the way their American cousins do. “American CIOs tend to be more technology-driven. Europeans are more planning-driven,” says one. Europeans tend to be more cautious, too. In Europe, “they’ll implement the technology if they can control it. It’s a risk-free approach. People in the U.S. might just say, “Go for it,”” says the head of a French consulting company. European geography makes a difference, too. “Europe is more fragmented, so European companies support extensive decentralized processing strategies. That puts much more focus on the overall IT architecture,” says an information systems director. (Information Week 10/10/94 p.192)

Where Do Technology $$$ Go? An Information Week survey of 500 large companies shows the average firm spends 40% of its technology budget on staff, 22% on hardware, and almost 13% each on software and telecommunications. The respondents predicted they’d be spending nearly a fifth of their budget this year on client-server technology. The survey also revealed a lot of turnover in information technology staff – the average CIO tenure is just three years and four months. (Information Week 10/10/94 p.30)

Kindernet. An MIT study predicts the median age of Internet users will drop from 26 to 15 within the next five years. (Bottom Line Personal 10/15/94 p.10) [I find this absolutely astounding... .Ed].

Big Blue’s Superhero. IBM is pitching its own superhero series, “The Adventures of Hyperman,” to TV networks, hoping to snare a Saturday morning slot for the cartoon show based on its new interactive CD-ROM program. Hyperman regularly defeats villains Entrobe and Kid Chaos using basic scientific concepts and help from some of the great scientists of the past. (Business Week 10/17/94 p.6)

The Thinking Man’s Network. Forget the Internet – smart networks developed by companies like AT&T and IBM are the real wave of the future, according to Byte magazine (October ’94). IBM’s Intelligent Communications service will act as a hub for translating communications from one service and medium to another, using software “agents” programmed to select the best mode of communication based on time of day or other instructions. (Investor’s Business Daily 10/13/94 A4)

Stop Squeezing Your Mouse. The Center for Workplace Health Information says that most computer users squeeze a mouse much too hard, and should hold it as if they were holding a small bird. A call to 800-554-4283 will get you a free copy of the Center’s Six Rules for Mouse Use. (Atlanta Journal-Constitution 10/13/94 E2)

Electronic Shopping Gets Safety Net. A San Diego startup company called First Virtual Holdings, backed by Electronic Data Systems and the First USA credit card company, will offer a system that promises shoppers the ability to make purchases over the network without concern for fraud. A consumer wishing to buy a product would send an e-mail message to the merchant over the Internet, furnishing the buyer’s First Virtual account number; the merchant would confirm the account number and notify First Virtual of the transaction; and First Virtual would send a message to the consumer confirming that the request was legitimate. (New York Times 10/15/94 p.19)

Protecting Property On The Internet. As the debate over copyright protection of electronic documents heats up, new encryption programs may pre-empt the need to overhaul the current copyright law by providing a technical fix instead. The Electronic Copyright Management System, developed by the Corporation for National Research Initiatives, the Library of Congress, and DOD’s Advanced Research Projects Agency, will be tested at the University of Illinois later this month. The purchaser of digital data would receive a cryptographic key to decode the document. A cypher would then exist in the document, identifying the owner and the buyer. If the buyer distributes the document on a BBS, for instance, cybercops will be able to trace the transgression back to the offending party. Another approach would turn a document purchased for one-time use into gibberish when transmitted via e-mail. (Business Week 10/17/94 p.212)
STANDARDS

An Update on Standards Relevant to USENIX Members

by Nick Stoughton
USENIX Standards Report Editor
<nick@hoskyns.co.uk>

You know how you get “tuned in” to certain words or phrases because they are relevant to things you do? Well, I tend to notice the word “standard” whenever I see it, which has led me to ponder the various meanings of the word; usually it is being used with a different meaning than International Standards.

Standards can be an accepted or an approved example of something against which another is measured (the Collins dictionary). It also means, in Britain at least, a size that is smaller than large and larger than medium! (I believe that you Americans use “regular” instead of “standard” for this). Clearly, by this measure, POSIX.2 is not standard!

A standard is also a principle of propriety, honesty, and integrity, as in “setting high standards.” Is this POSIX?

An editorial in a British trade weekly recently caught my eye, since it had the word “standards” in the title. For a change, it was really about the sort of standards we are talking about here. So, with permission from the author, John Lamb, and from Computer Weekly, I reproduce some parts:

Mention Standards to most users and you conjure up a mental image of grey committee members arguing over some obscure communications protocol. Packed with suppliers’ representatives, users can be forgiven for regarding many of these committees as cynical efforts to ensure that standards are drawn so widely as to be meaningless.

Remember the ill-fated ACE consortium? Whatever happened to that gathering? Even when the standards body does shuffle together some recommendations, such as the POSIX interface for UNIX, loose specifications render them worse than useless.

All too often the work of standards bodies is undermined by suppliers who refuse to join in. IBM and Microsoft have both played this game with the Object Management Group, although they both finally played ball. Against this background it is not surprising that users are now taking things into their own hands.

[Recently] London hosted a powerful gathering of top names from the oil industry. They want to standardize the software they use for key applications. With the expenditure of billions of dollars, the oilmen have the clout to tame their suppliers.

But Barclays Bank [one of the largest High Street banking companies in the U.K.] has also shown that individual users can build IT strategies around standards. They have taken the bold move of insisting that suppliers adhere to the Open Software Foundation’s DCE specification.

And in the U.S. the network Applications Consortium, a user group with total revenues of $200 billion, is trying to standardise interoperability between software operating systems and network services.

Agreeing on standards is no easy matter: technical complexities, resistance from end-users who have their own pet solutions, and vested commercial interests all conspire against the would-be standards setter.

But the end result in terms of cheaper costs and streamlined systems is worth the effort by the users, if only more would get involved.
Although I do not agree with all of what John has to say, it is encouraging to see someone else recognizing some of the problems we face. It is also interesting to see the "mental image" of us standards geeks as being grey individuals arguing about obscure communications protocols; he clearly has never been to a POSIX meeting!

POSIX may be regarded by some as a loose specification that is worse than useless, but it does have a significant user representation. That’s what I do after all. It is loose in the base specifications, but uses profiles (few of which are yet published) to tighten up for specific uses, such as the POSIX.18 UNIX profile.

It is with considerable regret that we see Jeff Haemer standing down as the USENIX Institutional Representative to POSIX. Jeff has been undertaking this role for the last three years, and has done much to forward the work of POSIX in the interests of the end users. For various, personal, reasons, Jeff feels that he can no longer commit the time needed to this work.

It was Jeff who introduced me to the world of standards, and took me on as Snitch Editor. I owe him a major debt of gratitude, and hope that you will all join me in wishing him well. Jeff has been responsible for keeping POSIX on a sensible course, fighting invention, and ably representing the interests of the majority of UNIX users in the U.S.

Stephe Walli, my predecessor as Snitch Editor, will be rejoining the battle as the USENIX I.R. from January. If you want to contact him, send e-mail to <stephe@usenix.org>. Stephe, you have a hard act to follow, but I know you’ll do it well!

Report on POSIX.5: Ada Bindings

Robbie Robbins <robbie@lfs.loral.com> reports on the July 11-15, 1994 meeting in Nashua, NH:

The primary charter of the POSIX.5 group is to produce Ada language bindings to POSIX standards. The standard for Ada language bindings to 1003.1-1990 was published in 1992 as 1003.5-1992. The working group is now working on three projects:

• P1003.5a – A few problem fixes to the 1003.5-1992 Standard

• P1003.5b, comprising the documents formerly named P1003.20 (the Ada binding to POSIX.1b, the real-time extensions of POSIX) and Mutexes and Condition Variables and Thread I.D.s from P1003.20a (the Ada binding to POSIX.1c, the real-time threads extensions of POSIX.)

• P2003.5, the test assertions document for 1003.5-1992.

In addition, a MOTIF/Ada study group met to consider producing an Ada Binding standard to 1295-1993, the IEEE MOTIF standard.

P1003.5a

There were no meetings on P1003.5a in Nashua. The plan for P1003.5a is to fix the known errors in 1003.5-1992 and rewrite Chapter 6 (Read/Write).

P1003.5b

P1003.5b is the Ada binding to POSIX.1b (the real-time extensions of POSIX) and Mutexes and Condition Variables and Thread I.D.s from P1003.1c (the real-time threads extensions of POSIX).

The first formal ballot on what was then named P1003.20 was conducted on a "thin" binding version: that is, P1003.20 did not repeat the underlying semantics of the POSIX Real-time Extensions draft, which is a C-language interface. The ballot showed that the balloting group favored a self- contained, "thick" binding version instead, which would not require references to the C version. By January, the underlying semantics were imported into the draft. This thickening process in turn exposed problems in the bindings draft. We worked on these problems, and at the July meeting the group reviewed the document and made further changes, and the document was edited for consistency.

Most of the July meeting consisted of issues presentations and fine tuning to 1003.5b. The object was to have a new draft ready by August 1 for ballot, scheduled for the month of September.

The POSIX.5 working group, together with the POSIX.4 working group, is still working to resolve the seven objections to P1003.1c that the POSIX.5 working group submitted in July as a coordination ballot. Five of the objections are considered resolved. The remaining two objections are currently under negotiation between the chairmen of the two working groups, and these negotiations continued during the July meeting. The first objection involves situations where the code of a signal handler needs to ensure that a mutex is locked. The other involves the change brought on by P1003.1c from per-process signal masking to per-task signal masking.

This is still a major issue between the two working groups, and we hope to have it resolved by October.
P2003.5

Kathy Liburdy of Clemson University gave a report on the status of test methods. Sections 3 and 4 are complete in rough draft form and need review. A complete draft of P2003.5 will be ready by the time of the October PASC meetings.

MOTIF/Ada

IEEE PASC approved a Study Group to consider an Ada Binding to the recently approved IEEE Motif Standard (IEEE 1295). This group met in Nashua to discuss what could be standardized. There are several commercial products that provide a MOTIF/Ada binding, thus there is substantial existing practice. However, these products all use Ada83, and the new version of Ada (Ada94) is nearing approval within ISO. Ada94 provides several Object Oriented Programming features that would be very useful in describing a MOTIF interface. As a result, the group decided that standardizing an Ada83 binding would not be very useful. Conversely, there is insufficient commercial practice with Ada94 MOTIF bindings to justify standardization at this time, but it was reasonable to expect Ada94 products "real soon now". In the interim, there are some useful activities that the group could do to prepare for a MOTIF/Ada standard. In particular, the IEEE 1295 document requires some services defined in Xf and Xlib. The C bindings are "standardized" by the X Consortium, but no equivalent document exists for Ada. The group wants to identify the minimum set of Xf/Xlib features needed for MOTIF, so that this set can be included in the MOTIF/Ada binding. The group has asked PASC for permission to continue to meet through Jan 95, with the expectation that there would be sufficient practice to start standardization by the end of 1995.

Report on ISO SC22

John Hill <jhill@bb.unisys.com> reports on the September 19-23, 1994 meeting in The Hague, The Netherlands:

Many of you are aware that there is an international standards development committee whose scope includes computer programming languages and their environments. That committee's formal name is ISO/IEC JTC 1 SC22. Or, SC22 for short. This brief article concerns the outcomes of the most recent meeting of SC22. They met in The Hague, The Netherlands 19-23 September 1994. NNI, the Netherlands national standards body, (analogous to ANSI for the USA) hosted the meeting.

Thirteen delegations participated in the following. They came from Austria, Canada, Denmark, ECMA, Finland, France, Germany, Japan, Netherlands, New Zealand, Switzerland, UK and the US. It happens that the Austrian delegation carried the proxies of both New Zealand and Switzerland.

Eleven working group conveners attended, three other working groups were represented by members of national body delegations. The working group conveners who attended were from WG2-Pascal, WG3-APL, WG4-COBOL, WG5-Fortran, WG9-Ada, WG11-Bindings, WG14-C, WG17-Prolog, WG19-FDTs, WG20-Internationalization, and WG21-C++.

The working groups whose conveners did not attend were WG8-BASIC, WG15-POSIX and WG18-FIMS.

The outcome of these big political meetings is always a set of resolutions. Just so you get an idea of what sort of thing goes on, I reproduce here those resolutions.

SC22 approved 28 resolutions covering a broad range of activities. The following summarizes the key resolutions:

- Divisions of work item were approved for POSIX, APL and Internationalization. This allows for what were previously large, single-document projects to deliver their work in smaller packages.

- Five (5) project editors, all from the US, were approved.

- Eight (8) WG conveners were reappointed. Three from the US received conditional reappointments pending final US nomination approval pending some paperwork.

- SC22 formed a new WG, WG22, for Portable Common Tools Environment (PCTE). France provided a convener, Regis Minot. The UK provided a project editor, John Dawes. SC22 expects Switzerland to propose the European Computer Manufacturers Association (ECMA) as Secretariat.

- SC22 disbanded three WGs: WG2-Pascal, WG8-BASIC, WG18-FIMS. The work on Pascal and BASIC is complete, except for maintenance. The US will provide a project editor for each. The work on FIMS was disbanded due to insufficient participation. SC22 invited the US to submit its FIMS standards for fast-track approval. A likely candidate for this is the COBOL binding.

- SC22 reviewed the JTC1 Chairman's proposals for use of information technology and approved a contribution for the Geneva JTC1 meeting. The text of this contribution is in an attachment at the end of this report.
Please be aware that the document is a response to a proposal that would automate all of JTC1. Consequently, this is a single input of many that JTC1 will take into consideration at its next meeting.

- SC22 requested its chairman to raise, at the SC chairman’s forum, issues concerning publications delays (in ITTF). Essentially, SC22 observed a number of delays in having its standards published. This resolution will deliver some ordinance on the target.

- Canada announced its intent to retain the SC22 secretariat and named Mr. Ed Borkovsky to that position.

This SC22 meeting dealt with a large number of touchy administrative issues, and a few tricky technical matters. As has been typical in the past, the atmosphere was intense but under control. All the national body delegations made effective contributions making for a lively set of debates.

Draft Contribution from ISO/IEC JTC 1 SC22 to JTC 1 on JTC 1’s Strategy for Implementation of Information Technology: SC22 welcomes the initiative to move progressively towards electronic working and appreciates the work involved in producing document JTC1 N3034. It wishes to cooperate in achieving the aggressive time scale proposed. During its discussions to this end, some suggestions were made on which SC22 would welcome further consideration.

SC22 suggests that further consideration be given to the use of electronic mail and FTP in addition to diskettes, and to the use of SGML as a target standard. SC22 would welcome information on the availability and cost of appropriate SGML environments.

SC22 Comments and Suggestions on Document JTC1 N3034. The document discusses the use of electronic formats for document registers and ballot/comment due date list, including distribution by diskette. However, Recommendations #1 and #2 encourage the use of electronic mail. Electronic mail seems reasonable for use of document registers and ballot/comment due date lists in preference to mail distribution of diskettes. (We note parenthetically that customs delays are occasionally encountered in the shipment of diskettes.) The character code set ISO 8859-1 (Latin-1) includes characters for Western European alphabets and cannot be displayed on many existing display screens.

The code set ISO 646 IRV (ASCII+) is a seven-bit code which can be transmitted through existing electronic mail facilities, but some characters will display differently on different display screens in different countries. For example, a document entitled “Problems with ‘£’ character” would display on many Japanese screens as “Problems with ‘¥’ character”.

The most restrictive code set, ISO Invariant 646, contains only the 26 (English) lower case letters, the 26 upper case letters, the ten digits, and the punctuation characters !+%&*()+,-./:<>? and space.

SC22 recommended replacing “ISO 8859-1” by “either ISO 646: IRV or invariant ISO 646.” The advantage of restricting administrative documents to Invariant ISO 646 is that such documents can be assured to display the same on (almost) any existing screen or printer and to transmit unchanged via electronic mail.

SC22 also recommended modifying another recommendation so it reads:

“For work in progress where possible and all future work, Project Editors be required to submit their text (WD, CD, DIS, etc.) in electronic form. When technically possible, Project Editors shall provide also a plain text (ASCII) version for review by National Body experts.”

It should be noted that it is not possible in all cases for Project Editors of work in progress to provide their texts in electronic form so that printed form can be reproduced. For example, printing of the paper copy may involve manual intervention or use of proprietary software, or may be dependent on a specific computer environment.

The ability of working groups to accept and convert electronic documents irrespective of format requires an extensive range of system and software resources. Few, if any, SC22 working groups have such a range of resources available to them. It is not practical to expect working group conveners to make such an extensive investment.

Therefore, the SC22 secretariat is unable to commit to implementation of all of the recommendations by 1995/03/01. A period of investigation and system development is required in order for the secretariat to implement any procedure for electronic distribution of draft standards. It is probably necessary to limit the number of formats that will be accepted and converted.

Report on SRASS: Services for Reliable, Available, Serviceable Systems

Arun Chandra <achandra@vnet.ibm.com> reports on the
July 11-15, 1994 meeting in Nashua, NH:

The POSIX Services for Reliable, Available, and Serviceable Systems (SRASS) study group was finally approved to become a working group at the July 1994 meeting held at Nashua, NH. This study group was formed in October 1993 following two BOF sessions.

The group, originally called the Fault Management Study group, is interested in issues of fault tolerance, high avail-
ability, reliability, serviceability, and maintainability. Vendors in the group include Tandem, Sequoia, Unisys, Texas Instruments, Stratus, and IBM. Users in the group include the Naval Surface Warfare Center, the Naval Air Warfare Center, Jet Propulsion Laboratory, and the Johns Hopkins University Applied Physics Laboratory.

The purpose of the group’s project is to specify a set of interfaces to allow portable access to underlying fault management services by application programs. These underlying services may support detection, logging, and notification of abnormal conditions during system operation; classification and analysis of abnormal conditions for fault diagnosis; and corrective actions for system reconfiguration, reinitialization and recovery.

The SRASS group’s Project Authorization Request (PAR) includes two major areas: Fault Tolerance and Serviceability. IBM’s AIX documentation has been identified as the base document for serviceability issues while documentation from Tandem, Sequoia, and Stratus will serve as the documentation for fault tolerance issues. Also, a number of other existing documents are being used as reference documents.

The SRASS group’s initial scope within POSIX is to define standard APIs which will address fault management (fault tolerance, serviceability). Liaison with other groups will ensure that the APIs standardized by this group will not violate existing standards. Checkpoint/Recovery is an example of the above. The initial list of fault management extensions to the POSIX specifications will comprise APIs for fault detection, fault localization, reconfiguration, fault recovery, error logging, notification, tracing, and dumps. The scope of this effort will be limited only to those APIs which support functionality existing in common practice.

For more information on the group please contact the group Chair, Helmut Roth, <hrath@relay.nswc.navy.mil>, or the group Vice Chair, Arun Chandra, <achandra@vnet.ibm.com>.

---

**Statement of Ownership**

**Management and Circulation, 9/30/94**


<table>
<thead>
<tr>
<th>Extent and nature of circulation</th>
<th>Average no. copies each issue preceding 12 mos.</th>
<th>Actual no. copies of single issue published nearest to filing</th>
</tr>
</thead>
<tbody>
<tr>
<td>A. Total no. copies</td>
<td>6063</td>
<td>6200</td>
</tr>
<tr>
<td>B. Paid circulation, Mail subs.</td>
<td>5526</td>
<td>5744</td>
</tr>
<tr>
<td>C. Total paid circulation</td>
<td>5526</td>
<td>5744</td>
</tr>
<tr>
<td>D. Free distribution</td>
<td>389</td>
<td>350</td>
</tr>
<tr>
<td>E. Total distribution</td>
<td>5915</td>
<td>6094</td>
</tr>
<tr>
<td>F. Copies not distributed</td>
<td>148</td>
<td>106</td>
</tr>
<tr>
<td>G. Total</td>
<td>6063</td>
<td>6200</td>
</tr>
</tbody>
</table>

I certify that the statements made by me above are correct and complete.

Ellie Young, Executive Director
The Bookworm
by Peter H. Salus
<peter@uunet.uu.net>

Everyone knows I read a lot. A little while ago, I was in a mall in suburban Boston and I was stopped by a woman with a clipboard who asked if she could ask me some questions. "Sure," I replied. We went through about three queries (like, did I live in Boston) and then "Do you read novels?" "Yes." "What's the name of the last novel you finished?" "Bleak House." (Scans the piece of paper.)"That's not on my list." "That's not my fault." "Whose is it by?" "Charles Dickens." "Oh." I was not asked any of the remaining questions. (I should mention that my spouse gave me the Oxford complete Dickens nearly two years ago and that I'm working through them in chronological order.) So I can tell we're into the new, not re-reading the old.

A year ago I wrote that I thought I would scream if I received many more books with "Internet" in the title. Many more have arrived. I have (more or less) maintained my composure. But I was really interested in how many there were. So I did a count in Books in Print and Books New and Forthcoming. I got 153 in August. That didn't include, for example, the four volumes of Comer/Stevens or Carl-Mitchell/Quarterman (they have "Internetworking . . ." in their titles) nor the second edition of Rose's Simple Book. I find it staggering. So, much of this month's column is on Net books (Peter's Christmas List is at the end).

The Good, the Insipid, and the Worthless

Yes, Virginia, there are worthwhile books on the Internet. Last March I praised Quarterman and Carl-Mitchell's The Internet Connection. About a month ago, along came Doug Comer's contribution: The Internet Book. Comer's book is "richer" than The Internet Connection in how computer networking in general works; The Internet Connection is still the only book that tells the reader what she/he has to do to get a connection, not how to get an account from a supplier. But Comer's is a really first-rate book. And even if you're an experienced networker, you'll learn something.

Adventures in UNIX Network Applications Programming is destined for the opposite end of one dimension of networking: the applications programmer. Rieken has done a really good job with the chapters on UNIX domain sockets, Internet domain sockets and streams; Weiman has done well on UDP and TCP sockets. This may be the best book on network applications programming since Stevens' in 1990.

How to Manage your Network Using SNMP is a straightforward, practical, realistic piece of work tying together a lot of diffuse information with tool-example in Tel/Tk. It's a solid, useful book. Arick's TCP/IP Companion is supposed to be "user friendly," and make "using TCP/IP easy for everyone." It does not mention Archie, Gopher, Mosaic, or WWW. The "Glossary" has an entry ANSI: American National Standards Institute, but the definition of ASCII mentions the American Standards Institute, which occurs nowhere else. Tch, tch.

Stallings' Network and Internetwork Security is the sort of solid, workmanlike volume we have come to expect from the author. This is a different sort of volume from those by Curry and by Garfinkel and Spafford. It's interesting and worthwhile.
BOOK REVIEWS

Glister’s *Finding it on the Internet* is a straightforward guide to Archie, Veronica, Gopher, WAIS, WWW, Mosaic, and several other browsing tools. It works for me. Better than most.

I don’t think I can say the same thing for *The Internet Business Book* which was clearly designed to be part of the wagon train of Internet books for niche X. The niche here is “Marketing . . . Sales, Advertising . . .” The Glossary contains Archie, Gopher and WWW, but not Mosaic. Shucks.

The real loser of the month, however, is *USENET: Netnews for Everyone*. In the chapter on “The Past,” we’re told that the originators were Truscott and “Elliott.” Ellis, man. No mention of Steve Bellovin. A News, B News, and C News are all mentioned. Such unknown folks as Daniel, Horton, Glickman, Adams, Essick, Thomas, Kolstad, Spencer, or Collyer are not mentioned. Over half the volume is lists of newsgroups. By the time these things get into print, they’re useless. Bah.

Second Editions

Held’s *Complete Modem Reference* is out in a new edition. It’s a decent enough reference work for those of you running PCs. Lots less so if you’re running (say) a SPARC box or a Mac. Though it lists many modems, it lacks either of the two I own, even though it’s well over 100 pages longer than the first edition.

Glister’s *Internet Navigator* is out in its second edition. It’s also put on over 100 pages. Though Glister gets a number of things wrong in his brief history of the net, at least he knows that it’s “Ellis” and he gets a lot of stuff about USENET right. While this isn’t my favorite “introductory” book, it’s better than over 90% of them.

And Some Other Books

*UNIX: An Open Systems Dictionary* is really quite decent. I looked up A, B and C news; they were all there. So were both ASA and ANSI with an appropriate pointer. The authors know that Mike Lesk created UUCP. They think that UUNET was originally called SEISMO, but that may be a toughie.

Wood’s *Information Security Policies Made Easy* isn’t about security. It’s about policies. I found it fascinating, mainly because I hadn’t thought about the issue before. Clearly, for any security effort to be effective, it has to have management support. And the clearest evidence of that support is a set of policies that are widely published and understandable. The book and diskette set is priced to make your management take notice.

*Cyberspace and the Law* is another work off my usual beat, but it’s a good one. In a small paperback, Cavazos and Morin make a wide number of legal questions clear. Privacy, business, copyright, obscenity, pornography, are only a few of the topics that are covered. Appendix B contains the entire text of the “Electronic Communications Privacy Act” (35 pages); Appendix C is the text of the Federal Code on the “Transportation of Obscene Matters” (one paragraph); Appendices E and F contain the Federal and the various State Child Pornography statutes. This is a fine and serious work.

**Peter’s Christmas/Chanukah Recommended List:**

This is the time of year when relatives ask one another what to get for Janet or Irv. Following is the list of my suggestions as to what you might ask for.
1. *sendmail* (Costales, Allman & Rickert); O'Reilly & Associates
2. *Tel and the Tk Toolkit* (Ousterhout); Addison-Wesley
3. *Practical Internetworking with TCP/IP and UNIX* (Carl-Mitchell & Quartenman); Addison-Wesley
4. *The Internet Book* (Comer); Prentice Hall
5. *4.4 BSD Manuals; O'Reilly & Associates*
6. *The Magic Garden Explained* (Goodheart & Cox); Prentice Hall
7. *%!@(: (Frey & Adams); O'Reilly & Associates*
8. *The Design and Evolution of C++* (Stroustrup); Addison-Wesley
9. *Cyberspace and the Law* (Cavazos & Morin); MIT Press
10. *Finding it on the Internet* (Gilster); Wiley

These are not in any particular order. There were a number of other notable books this year, Stevens' on the TCP/IP protocols and Cheswick & Bellovin on firewalls come to mind, as does the second edition of Rose's *Simple Book*. Finally, a last plug for my own *Quarter Century of UNIX*.

Happy hacking in the New Year!

**Application Development for Distributed Environments**


*Reviewed by Bob Birss*

*<bob.birss@eng.sun.com>*

If you’re contemplating “downsizing” or “rightsizing,” you could probably find this book useful, especially Part 5, “Application Development Products,” which is a survey of 4GL's and CASE tools. If you’re new to “client-server” or “distributed” computing, you will find Part 2, “Analysis and Design,” especially useful.

However, there are many gaps and idiosyncrasies in this book. For example, it is surprising that a book of this size, on this subject, published in 1994, spends exactly one page discussing “Testing the Application.” It suggests that Ms. Dewire has spent more time writing about client-server applications than she has testing them, or even developing them. Certainly, everything she says in that one page is true and useful. However, as she says (p. 132), “Testing distributed applications is much more complicated than testing applications developed using traditional structured methods.” Indeed. Were she to be more helpful, she might mention the numerous tools now available that are designed specifically to test client-server applications and their GUI’s.

She might also discuss the design principle that many of us have learned the hard way: client/server applications are best built with GUI’s separated from engines that have programmatic interfaces, in order to minimize the amount of functionality that must be tested through the GUI and maximize the amount that can be tested in more traditional ways. Ms. Dewire also has a very limited discussion of user interface issues: for example, neither “GUI” nor “user interface” appear in the book’s index, and there is no discussion of X Window System user-interface building tools such as X-Designer or UIMX.

This book tells a lot of the story and covers UNIX as well as Windows, OS/2, and the Macintosh, but there is a lot missing. A developer could run into serious trouble if this were his or her only guide to distributed application development.

**ATM User-Network Interface Specification Version 3.0**


**Asynchronous Transfer Mode (Second Edition)**

**Solution for Broadband ISDN**


*Reviewed by George V. Neville-Neil*

*<gmn@netcom.com>*

“ATM is a multiplexing and switching technique which uses fixed sized cells, each tagged with a virtual channel identifier. It is asynchronous in the sense that cells belonging to the same virtual channel will not necessarily appear at period intervals.” The preceding sentence is taken from my notes of a short course on ATM given at the Imperial College of London in the summer of 1993. The instructors, Dr. Ian Leslie and Dr. Derek McAuley (of Cambridge University), were pointing out one of the few concrete statements made about ATM networking technology.

Asynchronous Transfer Mode (ATM) networks are all the rage at the moment. They are the networks that will supposedly make the “Information SuperHighway” possible. The question remains, in many people’s heads, “What is ATM?”

There are now several books on ATM, including the two that I’ve just finished reading. The first book, *ATM User-Network Interface Specification*, is just that, a spec. If you are working on implementing ATM technology standards, either at the hardware or the driver level, then this is THE book for you. It is one of those must-have books, and if you are working on this sort of stuff, you probably already have it.

The authors of this book are The ATM Forum, a consortium of several companies interested in deploying ATM networks in a LAN environment. Until the Forum came into existence,
most of the work on ATM was being done by the International Telecommunications Union (formerly the CCITT) who are mostly interested in deploying long haul networks. Since many computer companies saw ATM as a possible high-bandwidth replacement for current LANs, some of them formed the consortium to push their own agenda and create standards faster than the ITU. This book is the specification that they came up with. It is therefore most applicable for networks that exist within a customer site. It also deals with interfacing with external networks, but it is not the specification that will be used when the phone companies install their own ATM networks.

Following a brief, ten-page introduction, the book works its way up from the Physical Layer (Section 2) through the ATM Layer Specification (Section 3), Interim Local Management Interface (Section 4), and finally UNI (User Network Interface) Signalling (Section 5). The main sections are followed by six annexes and six appendices.

The section on the Physical Layer describes the specs for several accepted physical media that ATM networks will be built from. Section 3, ATM Layer Specification, describes the management of the raw network data. This includes traffic-and congestion-control strategies. Section 4 describes the Interim Local Management strategies. SNMP is used for much of the control and monitoring of the network because the ITU and ANSI, who are specifying network standards for non-LANs consider the specification of monitoring protocols in a LAN environment as a matter “for further study.” The real meat of the spec is in section 5, where User Network Interface Signalling is described. This section talks about addressing, call control (setting up connections), message format, and error conditions – basically, how entities on the network find and communicate with each other.

The six annexes and appendices describe various things that did not make it into the main specification or were thought to be outside of its scope. There is a section that describes what is meant by a buzz-phrase often used with ATM networks: “Quality of Service.” It describes QoS in detail, with subsections on Performance Parameters, Measurement Methods and Factors Affecting ATM QoS Performance Parameters.

ATM User-Network Interface Specification is just that, a spec. It reads like one, very dry, but it is well written and edited. It is not the book for someone just getting into ATM networks who wants a good overview. It is intended as a reference work for those who are implementing ATM technologies. My only criticisms of the book are that the glossary and index are limited.

At approximately thirty dollars, it is fairly cheap for a specification (ITU and ANSI specs can cost hundreds of dollars). If you are an implementor, then this book is for you, if you are a novice, you’ll have to look further.

The second book, Asynchronous Transfer Mode (Solution for Broadband ISDN), discusses ATM networks from an ITU standpoint. This is not surprising as the author works for Alcatel Bell in Belgium.

This book serves better as an introduction/overview of the technology than a specification. The first two chapters spend a good deal of time describing the evolution and current state of the art in communication networks. Chapter 3 discusses the various standards being worked out by the ITU and the ATM Forum. In Chapter 4, Broadband ATM Switching, several different switching technologies are described. Future ATM networks will be built out of many point to point links connected at concentrating switches. Much like the current phone network, ATM switches are the heart of the network. Chapter 5 talks about the Impact of ATM on Terminals and Services. This chapter is mostly a discussion of how services carried by the network affect it and the equipment connected to it. The next chapter, ATM LAN: High Speed and Metropolitan Area Networks, describes ATM running over FDDI, FDDI-II, and DQDB and other similar networks. Chapter 7 devotes itself to Traffic Control in ATM Networks. Finally, Chapter 8 discusses Introduction Strategies for ATM. This last chapter is a vision of how ATM networks will eventually replace most current networking technologies, in the local area as well as the wide area.

ATM Solution for Broadband ISDN serves best as a general, though highly detailed, description of ATM networking technology. It is written in an academic style, and is reasonably easy to read. Each chapter has a bibliography, making it easy for the reader to find even more detailed information. The index is useful for finding general terms and concepts in the main text, but it is not exhaustive. There is no glossary, which is unfortunate as this would make the book perfect for beginners. I would recommend this book to anyone who is interested in ATM networking and has some knowledge of general networking principles.

Building a Successful Software Business


Reviewed by Gregory C. Depp
<depp@world.std.com>

The business of software has undergone tremendous change over the past few years. Formerly, the software professional in most organizations could focus strictly on the technical problem facing him or her, with very little concern about the non-technical problems and issues.

In the '90s, with tightening budgets and schedules, heightened competition, down-sizing, and the movement towards
contracting rather than full-time employment, it is becoming even more important that the software professional understand the “business” of software.

So, if you are a software developer, a software contractor, or a budding entrepreneur who plans to market a software package, *Building a Successful Software Business* provides a valuable overview of not only the building, but also the ongoing maintenance of your business.

Most software developers that I have known over the years, when confronted with terms like “Income Statement,” “Cash Flow,” and “Telemarketing” will usually mumble some remark about “bean counters,” or “suits,” laced with generous use of expletives. Even for these folks, this book is very approachable, readable, and useful. The concepts are clearly defined and outlined. Even the most business adverse technologist should come away with a better understanding of how to add value, by understanding some of the non-technical influences on the software under development.

Radin concentrates on these non-technical aspects of a software business in his book. As he states, there are four critical areas of the business: product development, marketing/sales, finance, and operations. The book addresses the latter three, since these are the areas not well-understood by the average technologist.

Chapter 1, “Ingredients for Success,” provides an introduction to some of the challenges facing the budding entrepreneur, laced with a number of success stories.

Chapters 2-11 discuss in some detail the art and science of marketing and sales. Again, using industry anecdotes as examples, Radin discusses various issues, strategies, and mechanisms for pricing, presenting, distributing, advertising, and selling a software product. For someone without marketing training, these chapters introduce many traditional marketing strategies, with advantages, disadvantages, usage guidelines and the relative cost of each. For those with some training in this area, the book addresses strategies unique to the software industry, such as shareware and how it can be a viable mechanism for marketing a company’s product.

Chapters 12-13 discuss the critical issues surrounding the development, care and feeding of customers. He includes information on training, support, and customer input into the design of a software product. The issues raised in this section are important to all software developers who desire to produce a successful product, as opposed to a technical *tour de force*.

Chapters 14-17 discuss the financial management aspects of running a company. For people who work in very large organizations, this information may be more abstract than for people working in startup organizations, who need to understand the difference between profits and cash flow. As Radin explains, “you can be extremely profitable in the short term, but still go out of business because you run out of cash.” He offers overviews of how to control costs, and raise the average “revenue per employee,” how to manage cash flow, and methods to finance operations.

Chapters 18-22 address various operations issues, such as how to find a business location, attract funding, the business plan, hiring, legal issues, protecting software assets, and going public. Interest in these chapters will be somewhat mixed, depending on the type of organization one is either working in, or anticipates working in. However, some of the information, such as staffing and asset protection is important at all levels of an organization.

The appendix to this book contains a “Business Resource Guide,” giving contact information for distribution channels, corporate partners, trade associations, publications, and a variety of other organizations. For those thinking of starting up a software business, this provides a convenient starting place for acquiring this information.

It is unfortunate, that, short of a few footnoted references, Radin does not provide a bibliography. Clearly, one book can not adequately address all the requirements in building and maintaining a software business. A comprehensive bibliography would significantly enhance this book’s value as a reference source.

In spite of this one drawback, *Building a Successful Software Business* provides an approachable, introduction to the issues, strategies, and techniques for building and maintaining a successful software business. The book is written in a style similar to the “nutshell” handbooks, primarily for the technologist looking for business information. For anyone in the business of software, it is $19.95 well spent!

**Zen and the Art of the Internet, 3rd Edition.**


Reviewed by Billy Barron
<billy@utdallas.edu>

About three years ago, I reviewed the first edition of Brendan Kehoe’s *Zen and the Art of the Internet* (called Zen from here on out). I was very excited about the book. It was free,
good, and the first real Internet book. Still an excellent book, the second edition came out two years ago about the same time as its first competitor was being published. Brendan was even good enough to keep the first edition free.

Since that time, Brendan has been through a lot. He started working for Cygnus Support. Then unfortunately, he was involved in a serious accident. There were concerns about his not fully recovering from the accident. The good news is that, from what I have heard, he has made an amazing recovery.

Now, on to the third edition. The back cover mentions a variety of new topics including MTV, Gopher, and WAIS. Looking around you will find them sprinkled about. However, WAIS is covered in slightly over a page. Gopher and Veronica combined, cover a total of three pages. These important topics need more discussion.

There are some omissions, the most glaring of which is that the Web is not in the index, nor did I see a reference to it. As the Web and Mosaic are a large part of the recent explosion of the Internet, they are important topics to cover.

Another major problem is that some of the information in the book was accurate when originally written, but over time has changed. The new edition does not do a good job of tracking those changes. For example, when I am mentioned in the book, it lists my former job and an old FTP address. I saw several other cases of old information.

In conclusion, while the book was good when originally written, it is quickly becoming dated. I suspect that part of the problem has been related to a lack of time to work due to full-time work and a hospital stay on Brendan’s part. The book has a good solid structure and clear descriptions. If you are looking for the latest information though, I would not recommend buying it.

A DOS User’s Guide to the Internet: E-mail, Netnews, and File Transfer with UUCP


Reviewed by Billy Barron
<billy@utdallas.edu>

A DOS User’s Guide to the Internet: E-mail, Netnews, and File Transfer with UUCP by James Gardner is misleading from start to finish. The first thing I noticed about the book was that the computer in the cover picture was a Commo-
USENIX 1995 Technical Conference

January 16-20, 1995
New Orleans, Louisiana

Tutorial Program*
Monday-Tuesday, January 16-17

Monday Tutorials

Essential UNIX Programming
Richard Stevens, Consultant

UNIX Security: Beyond the Basics
Professor Gene Spafford, COAST Lab Director

Introduction to UNIX Kernel Internals: Data Structures and Algorithms
Dr. Marshall Kirk McKusick, Consultant/Author

IP Network Administration
William LeFebvre, Northwestern University

DCE Remote Procedure Call System (RPC)
Richard Mackey, Open Software Foundation

WIN32: Porting X/Motif Applications to Windows NT
Art Baker, Cydonix Corp.

Hubs, Bridges, and Routers - The Tools of Networking
Dr. Vincent C. Jones, Consultant

Sendmail Inside and Out – Version 8
Eric Allman, University of California, Berkeley

Writing New Motif Widgets
David Shaffer, Integrated Computer Solutions, Inc.

Essentials of Perl Programming
Tom Christiansen, Consultant

Tuesday Tutorials

Firewalls: Achieving Security in an Internet Environment
Tina Darmohray, Lawrence Livermore National Laboratory

Publishing on The World Wide Web
Tom Cappetto and Anne LaVin, MIT

Concurrent Object-Oriented Programming
Douglas C. Schmidt, University of California, Irvine

The Kerberos Approach to Network Security
Daniel Geer and Jon A. Rothlis, OpenVision Technologies

Introduction to Tcl and Tk
John Ousterhout, Sun Microsystems

Selected Topics in Multiprocessing
Nawaf Bitar, Silicon Graphics, Inc.

Topics in System Administration
Trent Hein, XOR Network Engineering, and Evi Nemeth,
University of Colorado, Boulder

UNIX Network Programming
Richard Stevens, Consultant

Microsoft's OLE and the Common Object Model (COM)
David Chappell, Chappell and Associates

The Law and the Internet
Daniel Appelman, Heller, Ehrman, White and McAuliffe

Preliminary Technical Sessions Program

Wednesday-Friday, January 18-20, 1995

Wednesday, January 18

9:00-10:30
Opening Remarks
Peter Honeyman, CITI, University of Michigan

Keynote Address: Ubiquitous Computing
Mark Weiser, Xerox Palo Alto Research Center

11:00-12:30
BSD
Session Chair: John T. Kohl, Atria Software

Portals in 4.4BSD
W. Richard Stevens, Consultant, Jan-Simon Pendry,
Sequent Corporation

*To get complete tutorial descriptions, refer to postings on comp.org.useunix, access the World Wide Web, URL: http://www.useunix.org or send mail to our automatic mailserver at: info@useunix.org. Send in your message the line: send tut95 conferences and the tutorial information will be returned to you. (For complete catalog, send the line: send conferences catalog)
Dynamic Vnodes – Design and Implementation of the Vnode Deallocation Logic
Aju John, Digital Equipment Corporation

Union Mounts in 4.4BSD-Lite
Jan-Simon Pendry, Sequent Corporation, M. Kirk McKusick, Consultant and Author

Invited Talks: Highlights from the 1994: USENIX Symposium on Very High Level Languages
Tom Christiansen, Consultant

LISA, the 8th Systems Administration Conference
Dinah McNutt, Zilker Internet Park, Inc.

2:00-3:30
Mass Store
Session Chair: Charles J. Antonelli, CITI, University of Michigan

Evaluation of Design Alternatives for a Cluster File System
Murthy Devarakonda, Ajay Mohindra, Jill Simoneaux, William H. Tetzlaff, IBM TJ Watson Research Center

Multi-resident AFS: An Adventure in Mass Storage
Jonathan S. Goldick, Kathy Benninger, Christopher Kirby, Christopher Maher, and Bill Zumach, Pittsburgh Supercomputing Center

RAMA: Access to a High-Bandwidth Massively Parallel File System
Ethan L. Miller, University of Maryland, Baltimore County, Randy H. Katz, University of California, Berkeley

Invited Talk: Cryptography
Bruce Schneier, Counterpane Systems

4:00-5:30
Potpourri I
Session Chair: Greg Minshall, Novell, Inc.

Implementing Real Time Packet Forwarding Policies Using Streams
Ian Wakeman, Atanu Ghosh, and Jon Crowcroft, University College London

Scaling the Web of Trust: Combining Kerberos and PGP to Provide Large Scale Authentication
Jeffrey L. Schiller and Derek Atkins, Massachusetts Institute of Technology

Flexible and Safe Resolution of File Conflicts
Puneet Kumar and M. Satyanarayanan, Carnegie Mellon University

Invited Talk: Operating System Measurement
Brad Chen, Harvard University

Thursday, January 19
9:00-10:30
Objects
Session Chair: Richard Draves, Microsoft Research

OODCE: A C++ Framework for the OSF Distributed Computing Environment
John Dilley, Hewlett-Packard Company

Mach-US: UNIX On Generic OS Object Servers
J. Mark Stevenson, Carnegie Mellon University, Daniel P. Julin, ISIS Distributed Systems

Events in an RPC Based Distributed System
Jim Waldo, Ann Wolfrath, Geoff Wyant, Sam Kendall, Sun Microsystems Laboratories, Inc.

Invited Talk: Tcl for Internet Agents
John Ousterhout, Sun Microsystems, Inc.

11:00-12:30
Potpourri II
Session Chair: Lori Grob, Chorus Systèmes

Turning AIX into an MP-capable OS
Jacques Talbot, Bull

A Flash Memory Based File System
Atsuo Kawaguchi, Shingo Nishioka, and Hiroshi Motoda, Hitachi, Ltd.

TRON: Process-Specific File Protection for the UNIX Operating System
Andrew Berman, Virgil Bourassa, and Erik Selberg, University of Washington

Invited Talk: ILU/CORBA Inter-Language Unification
Bill Janssen, Xerox PARC

2:00-3:30
They Come From Palo Alto
Session Chair: Phil Winterbottom, AT&T Bell Laboratories

SIFT – A Tool for Wide-Area Information Dissemination
Tak W. Yan, Hector Garcia-Molina, Stanford University
Performance Implications of Multiple Pointer Sizes
Jeffrey C. Mogul, Joel F. Bartlett, Robert N. Mayo, Amitabh Srivastava, Digital Equipment Corporation, Western Research Laboratory

Idleness is not Sloth
Richard Golding, Hewlett-Packard Laboratories, Peter Bosch, University of Twente, Carl Staelin, Tim Sullivan, John Wilkes, Hewlett-Packard Laboratories

2:00-3:30
Invited Talk: Internet Information Commerce
Nathaniel Borenstein, First Virtual Holdings

4:00-5:30
Work-in-Progress Reports
Session Chair: Peg Schafer

Invited Talk: Cash on the Internet
David Chaum, DigiCash b.v.

Friday, January 20

9:00-10:30
Libraries
Session Chair: Douglas Orr, University of Utah

libckpt: Transparent Checkpointing under UNIX
James S. Plank, Micah Beck, Gerry Kingsley, University of Tennessee, Kai Li, Princeton University

Optimizing the Performance of Dynamically-Linked Programs
W. Wilson Ho, Wei-Chau Chang, and Lilian H. Leung, Silicon Graphics Computer Systems

DP: A Library for Building Portable, Reliable Distributed Applications
David M. Arnow, Brooklyn College

Invited Talk: Economics of the Internet
Hal Varian, University of Michigan

11:00-12:30
File Systems
Session Chair: Noemi Paciorek, Horizon Research

File System Logging versus Clustering: A Performance Evaluation
Margo Seltzer, Keith A. Smith, Harvard University, Hari Balakrishnan, Jacqueline Chang, Sara McMains, Venkata Padmanabhan, University of California, Berkeley

Metadata Logging in an NFS Server
Uresh Vahalia, EMC Corporation, Cary G. Gray, Abilene Christian University, Dennis Ting, Digital Equipment Corporation

Heuristic Cleaning Algorithms for Log-Structured File Systems
Trevor Blackwell, Jeffrey Harris, Margo Seltzer, Harvard University

Invited Talks: Highlights from the 1994 USENIX High-Speed Networking Symposium
Pat Paraseghian, AT&T Bell Laboratories

First Symposium on Operating Systems Design and Implementation (OSDI)
Jay Lepreau, University of Utah

2:00-3:30
Architecture
Session Chair: Bob Gray, US WEST Technologies

The New Jersey Machine-Code Toolkit
Norman Ramsey, Bell Communications Research, Mary F. Fernandez, Princeton University

ATOM: A Flexible Interface for Building High Performance Program Analysis Tools
Alan Eustace and Amitabh Srivastava, Digital Equipment Corporation, Western Research Laboratory

Adaptable Binary Programs
Susan L. Graham, Steven Lucco, Robert Wahbe, University of California, Berkeley

Invited Talk: The Internet at the Turn of the Millennium: How You’ll Use It and Who You’ll Meet There
Pavel Curtis, Xerox Palo Alto Research Center

4:00-5:00
Closing Session
The Second USENIX Symposium on Mobile and Location-Independent Computing will provide a major opportunity for researchers and practitioners in this rapidly growing field to exchange ideas and present results of their work.

The First Mobile Computing Symposium, held in Boston in August 1993, generated a great deal of interest from the UNIX and mobile computing communities. Since that time, mobile computing has become an even hotter topic, with the size, cost, and power requirements of the equipment going down. The FCC has announced a plan to auction radio spectrum for use of mobile devices, and the Internet Engineering Task Force (IETF) is in the process of standardizing protocols for mobile TCP/IP, including roaming capabilities. Mobile computers are the fastest growing segment of the PC market, airlines are scrambling to provide network connectivity on board, and terminal rooms at computer conferences routinely provide network taps for users who bring their own computers.

The 1995 symposium is a single-track symposium offering two days of refereed paper presentations. The symposium will also include panels, Work-in-Progress reports, Birds-of-a-Feather sessions, and a Keynote speaker. Formally reviewed papers, presented during the symposium, will be published in the symposium proceedings. Proceedings will be distributed free to attendees and later will be available for purchase from the USENIX Association.

**PROGRAM COMMITTEE**

- Program Chair: Jim Rees, University of Michigan  
  Jim.Rees@umich.edu
- Norman Adams, Xerox PARC  
  norman@parc.xerox.com
- Dan Duchamp, Columbia University  
  djd@cs.columbia.edu
- Dan Geer, OpenVision Technologies  
  geer@cam.ov.com
- Phil Karn, Qualcomm  
  karn@qualcomm.com
- Jim Kempf, Sun Microsystems  
  james.kempf@eng.sun.com
- Jay Kistler, Digital Equipment Corporation  
  jjk@src.dec.com

**SYMPOSIUM TOPICS**

We seek original and innovative papers about current developments in mobile and location-independent computing. We are especially interested in reports on practical experiences with mobile systems. The Mobile Computing Symposium will address a wide range of issues and ongoing developments, including, but not limited to:

- Applications for the mobile user
Navigation and positioning (GPS, etc.)
Security, especially in wireless environments or when away from home
Caching and disconnected operation of applications and file systems
Communications Protocols, including mobile TCP/IP
Wireless communications (CDPD, CDMA, GSM, Ardis/RAM, cellular modem, etc.), and how they relate to and interact with operating systems and applications
Portable and mobile computing equipment

**Refereed Paper Submissions**
Submission of an extended abstract of 1500–2500 words (9000–15000 bytes or 3–5 pages) is recommended. Shorter abstracts run a significant risk of rejection as there will be little on which the program committee can base an opinion. Extended abstracts should be sent to Jim Rees at the address below.

If you would like to receive detailed guidelines for submission and examples of extended abstracts, you may telephone the USENIX Association office at +1 510 528 8649 or email to mobile2authors@usenix.org.

For administrative reasons (not blind reviewing), each submission should include a separate page or e-mail message giving the title of the paper, the names and affiliations of the authors, and the name of the author who will act as the contact person for the program committee. For the contact person, also include a daytime telephone number, postal address, e-mail address and FAX number if possible.

USENIX symposia, like most symposia and journals, require that papers not be submitted simultaneously to more than one conference or publication and that submitted papers not be previously or subsequently published elsewhere. Papers accompanied by “non-disclosure agreement” forms are not acceptable and will be returned to the author(s) unread. All submissions are held in the highest confidentiality prior to publication in the Proceedings, both as a matter of policy and in accord with the U.S. Copyright Act of 1976.

**For More Program Information**
For questions about refereed paper submissions and other program concerns, contact the Program Chair:

- Jim Rees
  CITI
  University of Michigan
  519 West William
  Ann Arbor, Michigan 48103 USA
- Internet: Jim.Rees@umich.edu
- Telephone: +1 313 763 4174
- Fax: +1 313 763 4434

**Dates for Refereed Paper Submissions**
- Extended abstracts due: January 2, 1995
- Notification to authors: January 23, 1995
- Camera-ready final papers due: March 6, 1995

**For Registration Information**
Materials containing all details of the technical and tutorial programs, registration fees and forms, and hotel information will be available beginning in February 1995. If you wish to receive the registration materials, please contact USENIX at:

- USENIX Conference Office
  22672 Lambert Street
  Suite 613
  Lake Forest, CA 92630
  USA
  Phone: +1 714 588 8649
  Fax: +1 714 588 9706
  Email: conference@usenix.org

For more information about USENIX and its events, access the USENIX Resource Center on the World Wide Web. The URL is http://www.usenix.org. Or send email to our mailserver <info@usenix.org>. Your message should contain the line: send catalog.
ANNOUNCEMENT AND PRELIMINARY CALL FOR PAPERS

SPONSORED BY THE USENIX ASSOCIATION, IN COOPERATION WITH: THE COMPUTER EMERGENCY RESPONSE TEAM (CERT), IFIP WG 11.4, AND UniForum

OVERVIEW
The goal of this symposium is to bring together security practitioners, researchers, system administrators, systems programmers, and others with an interest in computer security as it relates to networks and the UNIX operating system.

This will be a three day, single-track symposium consisting of tutorials, refereed and invited technical presentations, and panel sessions. The first day will be devoted to tutorial presentations. Two days of technical sessions will follow the tutorials.

TUTORIALS
♦ June 5
This one-day tutorial program is designed to address the needs of both technical and management attendees. The tutorials will supply overviews of various security mechanisms and policies. Each will provide specifics to the system and site administrator for implementing numerous local and network security precautions, firewalls, and monitoring systems.

KEYNOTE AND TECHNICAL SESSIONS
♦ June 6-7
The keynote address by Stephen T. Walker, founder and president of Trusted Information Systems, will begin the technical sessions program. Mr. Walker will speak on information security and privacy in computing. Mr. Walker is an electronics engineer and computer systems analyst with over 25 years of experience in system design and program management; particularly extensive is his experience with the design and implementation of large scale computer networks and information systems. He is nationally recognized for his pioneering work on the DoD Computer Security Initiative, the establishment of the National Computer Security Center, and the formation of the Defense Data Network. He is a member of the Computer System Security and Privacy Advisory Board, established by the Computer Security Act of 1987.

The technical sessions program, in addition to presentations of refereed papers, will include invited talks, and possibly panel sessions. There will also be two evenings available for Birds-of-a-Feather sessions (BoFs) and Work-in-Progress Reports (WiPs). The program committee invites you to submit proposals, ideas, or suggestions for these presentations; your suggestions may be submitted to the program chair via email to security@usenix.org or by post to the address given on the following page.

The program committee will formally review and accept papers for presentation at the symposium and publish them in the symposium proceedings. USENIX will provide the proceedings free to technical session attendees; additional copies will be available for purchase from USENIX.
SYMPHOSIUM TOPICS
Presentations are being solicited in areas including but not limited to:
♦ User/system authentication
♦ File system security
♦ Network security
♦ Security and system management
♦ Security-enhanced versions of the UNIX operating system
♦ Security tools
♦ Security incident investigation and response
♦ Computer misuse and anomaly detection

REFERRED PAPER SUBMISSIONS
Submissions must be received by February 13, 1995. Full papers should be
10–15 pages. Instead of a full paper, authors may submit an extended ab-
stract which discusses key ideas. Extended abstracts should be 5–7 pages
long (about 2500–3500 words), not counting references and figures. The
body of the extended abstract should be in complete paragraphs. The object
of an extended abstract is to convince the reviewers that a good paper and
presentation will result. All submissions will be judged on originality, rel-
evance, and correctness.

An individual program committee member will be assigned to shepherd each
accepted submission through preparation of the final paper. The assigned
member will act as a conduit for feedback from the committee to the authors.
Camera-ready final papers are due May 1, 1995.

Please accompany each submission by a cover letter stating the paper title
and authors along with the name of the person who will act as the contact to
the program committee. Please include a surface mail address, daytime and
evening phone number, and, if available, an email address and fax number
for the contact person.

If you would like to receive detailed guidelines for submission and examples
of extended abstracts, you may telephone the USENIX Association office at
+1 510 528 8649, or email to securityauthors@usenix.org or to the program
chair.

The USENIX UNIX Security conference, like most conferences and jour-
nals, requires that papers not be submitted simultaneously to another
conference or publication and that submitted papers not be previously or
subsequently published elsewhere. Papers accompanied by "non-disclosure agreement" forms are not acceptable and will be returned to the author(s)
unread. All submissions are held in the highest confidentiality prior to pub-
lication in the Proceedings, both as a matter of policy and in accord with the
U.S. Copyright Act of 1976.

WHERE TO SUBMIT
Please send one copy of a full paper or an extended abstract to the program
committee via one of the following methods. All submissions will be ac-
nowledged.
♦ Preferred method: email (PostScript or ASCII) to
securitypapers@usenix.org
♦ Alternate method: postal delivery to Fred Avolio, Trusted Information
Systems, 3660 Washington Road, Glenwood, MD 21738
♦ Phone: +1 301 854 6889
♦ Fax: +1 301 854 5363

Program Chair:
Fred Avolio, Trusted
Information Systems, Inc.
Steve Bellovin,
AT&T Bell Laboratories
Bill Cheswick,
AT&T Bell Laboratories
Ed DeHart, CERT
Ed Gould, Digital Equipment
Corporation
Marcus Ranum, Trusted
Information Systems, Inc.
Jeff Schiller, MIT
Gene Spafford, COAST
Laboratory, Purdue University

FOR REGISTRATION
INFORMATION
Materials containing all details of the technical and tutorial programs, registration fees and
forms, and hotel information will be available beginning in April 1995. If you wish to re-
ceive the registration materials, please contact USENIX at:
♦ USENIX Conference Office
22672 Lambert Street
Suite 613
Lake Forest, CA 92630
USA
Phone: +1 714 588 8649
Fax: +1 714 588 9706
Email: conference@usenix.org

For more information about
USENIX and its events, access
the USENIX Resource Center
on the World Wide Web. The
URL is http://www.usenix.org.
Or send email to our mailserver
at: info@usenix.org. Your mes-
 sage should contain the line:
send catalog. A catalog will be
returned to you.

OCTOBER 1994;login: 61
The COOTS conference is designed to be a showplace for advanced development work in object-oriented technologies. The conference will emphasize research and experience derived from efforts to use object-oriented techniques to build complex systems that meet real world needs.

The COOTS conference will begin with two days of tutorials. The tutorial program will offer a selection of tutorials from among several tracks. We expect tutorial topics to include:

- distributed object systems (CORBA, etc.)
- object-oriented network programming
- alternative object-oriented languages
- advanced techniques in memory management
- efficient and effective class design

If you are interested in offering a tutorial, proposals are due December 1, to Doug Lea, tutorial program chair (dl@g.oswego.edu).

Two days of technical sessions will follow the tutorials. Proceedings of the conference will be published by USENIX and will be provided free to technical session attendees; additional copies will be available for purchase from USENIX.

Like the USENIX C++ Conferences and Advanced Topics Workshops from which it is derived, COOTS will emphasize the advanced engineering aspects of object technology. While papers covering work in C++ are encouraged, the conference is broader in scope than its ancestors and invites submissions describing results and work in other object-oriented or object-based languages.

**CONFERENCE TOPICS**

We seek papers describing original work concerning the design, implementation, and use of object-oriented technologies. Questions regarding a topic's relevance may be addressed to the program chair via electronic mail to russo@cs.purdue.edu.

Potential topics include:

- work on object-oriented programming languages (C++, Modula-3, Eiffel, etc.)
- implementations of commercial object infrastructures (CORBA, NextStep, OLE-II, SOM/DSOM, etc.)
- interface description languages
- distributed object systems
- unique applications of and experiences with object-oriented technologies

**REFEREED PAPER SUBMISSIONS**

Submissions must be received by March 6, 1995. Full papers should be 10 to 15 pages. Instead of a full paper, authors may submit an extended abstract which discusses key ideas. Extended abstracts should be 5-7 pages long (about 2500-3500 words), not counting references and figures. The body of the extended abstract should be complete paragraphs. The object of an extended abstract is to convince the reviewers that a good paper and presentation will result. While, by acceptance of extended abstracts, we intend to stimulate industrial participation, submission of extended abstracts by academics is in no way discouraged.
All submissions will be judged on originality, relevance, and correctness. Each accepted submission will be assigned a member of the program committee to act as its shepherd through the preparation of the final paper. The assigned member will act as a conduit for feedback from the committee to the authors. Camera-ready final papers are due May 15, 1995.

Please accompany each submission with a cover letter stating the paper title and authors along with the name of the person who will act as the contact to the program committee. Please include a surface mail address, daytime and evening phone number, and, if available, an email address and fax number for the contact person.

If you would like to receive detailed guidelines for submission and examples of extended abstracts, you may telephone the USENIX Association office at +1-510-528-8649, or email to cootsauthors@usenix.org or to the program committee chair.

The COOTS conference, like most conferences and journals, requires that papers not be submitted simultaneously to another conference or publication and that submitted papers not be previously or subsequently published elsewhere. Papers accompanied by “non-disclosure agreement” forms are not acceptable and will be returned to the author(s) unread. All submissions are held in the highest confidentiality prior to publication in the Proceedings, both as a matter of policy and in accord with the U.S. Copyright Act of 1976.

WHERE TO SUBMIT
Please send one copy of a full paper or an extended abstract to the program committee via one of the following methods. All submissions will be acknowledged.
• Preferred Method: email (Postscript or ASCII) to cootspapers@usenix.org
• Alternate Method: postal delivery to USENIX COOTS Conference
  c/o Dr. Vincent F. Russo
  Department of Computer Sciences
  Purdue University
  West Lafayette, IN 47907 USA
  Telephone: +1 317 494 6008

For more information about USENIX and its events, access the USENIX Resource Center on the World Wide Web. The URL is http://www.usenix.org. Or send email to our mailserv er at: info@usenix.org. Your message should contain the line: send catalog. A catalog will be returned to you.
ANNOUNCEMENT & CALL FOR PARTICIPATION

The 9th Systems Administration (LISA) Conference, sponsored by USENIX and SAGE, is widely recognized as the leading technical conference for system administrators. Historically, LISA stood for "Large Installation Systems Administration," back in the days when having a large installation meant having over 100 users, over 100 systems, or over one giga-byte of disk storage. Today, the scope of the LISA conference includes topics of interest to system administrators from sites of all sizes and kinds. What the conference attendees have in common is an interest in solving problems that cannot be dealt with simply by scaling up well-understood solutions appropriate to a single machine or a small number of workstations on a LAN.

The theme for this year's conference is "New Challenges," which includes such emerging issues as integration of non-UNIX and proprietary systems and networking technologies, distributed information services, network voice and video teleconferencing, and managing very complex networks. We are particularly interested in technical papers that reflect hands-on experience, describe fully implemented and freely distributable solutions, and advance the state of the art of system administration as an engineering discipline.

TUTORIAL PROGRAM

Monday and Tuesday, September 18–19, 1995

The two-day tutorial program offers up to five tracks of full and half-day tutorials. Tutorials offer expert instruction in areas of interest to system administrators of all levels, from novice through senior. Topics are expected to include networking, advanced system administration tools, Solaris and BSD administration, Perl programming, firewalls, NIS, DNS, Sendmail, and more.

To provide the best possible tutorial offerings, USENIX continually solicits proposals for new tutorials. If you are interested in presenting a tutorial at this or other USENIX conferences, please contact the tutorial coordinator, Daniel V. Klein:

Phone: +1 415 421 0285; FAX: +1 415 421 2332; E-mail: dvk@usenix.org

TECHNICAL SESSIONS

Wednesday through Friday, September 20–22, 1995

The three days of technical sessions consist of two parallel tracks. The first track is dedicated to presentations of refereed technical papers. The second track will accommodate invited talks, panels and Works-in-Progress (WIP) sessions.

CONFERENCE TOPICS

Papers addressing the following topics are particularly timely; papers addressing other technical areas of general interest are equally welcome.

• Your plans for the year 2000
• Deployment of new networking technologies
• Coping with the commercialization of the Internet
• Support models in use at your site
• Dealing with differences in UNIX implementations — migration and interoperability among BSD, SVR4, OSF and others
• Integration of UNIX-based with non-UNIX-based and proprietary systems and networking technologies (Mac, NT and DOS PCs)
• Application of emerging technologies (Mbone, Mosaic) to system administration
• Administration and security of distributed information services (WAIS, gopher, WWW) and network voice and video teleconferencing (Mbone)
• Experience supporting mobile and location-independent computing
• Experience with large (1000+ machine) networks, especially networks of SVR4-based systems
• Real-world experience with implementations of proposed system administration standards
• Unusual applications of commercial system administration software packages
• Application of operational planning techniques to system administration including measurements and metrics, continuous process improvement, automation, and increasing productivity
• File migration, archival storage & backup systems in extremely large environments
• Innovative tools and techniques that have worked for you
Managing high-demand and high-availability environments
Migrating to new hardware and software technologies
Administration of remote sites that have no technical experts
Supporting MIS organizations on UNIX
Real-world experiences with emerging procedural/ethical issues — e.g., developing site policies, tracking abusers, and implementing solutions to security problems
Networking non-traditional sites (libraries, museums, K-12)

**Refereed Paper Submissions**

An extended abstract is required for the paper selection process. Full papers are not acceptable at this stage; if you send a full paper, you must also include an extended abstract. “Extended” means 2–5 pages.

Include references to establish that you are familiar with related work, and, where possible, provide detailed performance data to establish that you have a working implementation or measurement tool.

Submissions will be judged on the quality of the written submission, and whether or not the work advances the state of the art of system administration. For more detailed author instructions and a sample extended abstract, send e-mail to: lisa9authors@usenix.org or call the USENIX office at +1 510 528 8649.

Note that LISA, like most conferences and journals, requires that papers not be submitted simultaneously to more than one conference or publication and that submitted papers not be previously or subsequently published elsewhere. Papers accompanied by “non-disclosure agreement” forms are not acceptable and will be returned unread. All submissions are held in the highest confidence prior to publication in the conference proceedings, both as a matter of policy and as protected by the U.S. Copyright Act of 1976.

Authors of an accepted paper must provide a final paper for publication in the conference proceedings. At least one author of each accepted paper presents the paper at the conference. Final papers are limited to 20 pages, including diagrams, figures and appendices, and must be in troff, ASCII, or LaTeX format. We will supply you with instructions. Papers should include a brief description of the site, where appropriate.

Conference proceedings, containing all refereed papers and materials from the invited talks, will be distributed to attendees and will also be available from USENIX following the conference.

**Where to Send Submissions**

Please submit extended abstracts for the refereed paper track by two of the following methods. All submissions will be acknowledged.

- E-mail to: lisa9papers@usenix.org; FAX to: +1 510 548 5738; Mail to: LISA 9 Conference, USENIX Association, 2560 Ninth Street, Suite 215, Berkeley CA USA 94710
- To discuss potential submissions, and for inquiries regarding the content of the conference program, contact the program co-chairs at lisa9chair@usenix.org or at:
  - Tina M. Darmohray, Livermore National Laboratory, PO Box 808 L-510, Livermore CA USA 94550. +1 510 423 5999; FAX: +1 510 422 7869; E-mail: tmd@usenix.org
  - Paul Evans, Synopsys, Inc., 700 East Middlefield Road, Mountain View CA USA 94043. +1 415 694 1855; FAX: +1 415 965 8637; E-mail: ple@usenix.org

**Invited Talks Track**

If you have a topic of general interest to system administrators, but that is not suited for a traditional technical paper submission, please submit a proposal for a second track presentation to the invited talk (IT) coordinators at itlisa@usenix.org or to:

- Laura de Leon, Hewlett-Packard. +1 415 857 5605; FAX: +1 415 857 5686; E-mail: deleon@hp.com
- Peg Schafer, BBN. +1 617 873 2626; FAX: +1 617 873 4265; E-mail: peg@bbn.com

**Program Committee**

- Program Co-chair: Tina Darmohray, Lawrence Livermore National Laboratory
- Program Co-chair: Paul Evans, Synopsys, Inc.
- Paul Anderson, University of Edinburgh
- Kim Carney, Massachusetts Institute of Technology
- Rob Kolstad, Berkeley Software Design, Inc.
- Bryan McDonald, SRI International
- Marcus Ranum, Trusted Information Systems, Inc.
- John Schimmel, Silicon Graphics, Inc.

**For Registration Information**

All details of the technical and tutorial programs, registration fees and forms, and hotel information will be available in July, 1995. If you wish to receive the registration materials, please contact USENIX:

- USENIX Conference Office
  22672 Lambert Street, Suite 613
  Lake Forest, CA 92630 USA
  Phone: +1 714 588 8649
  Fax: +1 714 588 9706
  E-mail: conference@usenix.org

For more information about USENIX and its events, access the USENIX Resource Center on the World Wide Web. The URL is http://www.usenix.org. Or send e-mail to our mailserver at: info@usenix.org. Your message should contain the line: send catalog. A catalog will be returned to you.

**Vendor Display**

Wed. & Thurs., Sept. 20–21, 1995

Well-informed vendor representatives will demonstrate products and services at the informal display. If your company would like to participate, please contact:

- Zanna Knight, +1 510 528 8649
  FAX: +1 510 548 5738
  E-mail: display@usenix.org
The Internet Navigator, 2ed  
Paul Gilster  
1-05260-4 $24.95 member price: $21.20  
# of copies:  

Advanced Topics in UNIX  
Ronald Leach  
1-03663-3 $24.95 member price: $21.20  
# of copies:  

Introduction to Client Server Systems  
Paul Renaud  
1-57774-X $34.95 member price: $29.70  
# of copies:  

Portable UNIX  
Douglas Topham  
1-57926-2 $14.95 member price: $12.71  
# of copies:  

UNIX, Self-Teaching Guide  
George Leach  
1-57924-6 $19.95 member price: $16.95  
# of copies:  

Object Oriented Programming with Turbo C++  
Keith Weiskamp  
1-52466-2 $24.95 member price: $21.20  
# of copies:  

Obfuscated C and Other Mysteries  
Don Libes  
1-57805-3 $39.95 member price: $33.96  
# of copies:  

Finding It On the Internet  
Paul Gilster  
1-03857-1 $19.95 member price $16.95  
# of copies:  

Internationalization: Developing Software for Global Markets  
Tuoc Luong  
1-07661-9 (pub. date: 1/95)  
# of copies:  

Adventures in UNIX Network Applications Programming  
Bill Rieken  
1-52858-7 $39.95 member price: $33.96  
# of copies:  

UNIX Shell Programming, 3e  
Lowell Jay Arthur  
1-59941-7 $29.95 member price: $25.45  
# of copies:  

The UNIX Command Reference Guide  
Kaare Christian  
1-85580-4 $32.95 member price: $28.01  
# of copies:  

Berkeley UNIX: A Simple & Comprehensive Guide  
James Wilson  
1-61582-X $40.95 member price: $34.80  
# of copies:  

Please send all orders to:  
John Wiley & Sons, Inc.  
Attn: Karen Cooper, Special Sales  
605 Third Avenue  
New York, NY 10158  
Phone: (212) 850-6789  
Fax: (212) 850-6142  

Payment enclosed, plus sales tax  
VISA  Mastercard  
American Express  
Card No.  
Name  
Firm  
Address  
City/State/Zip  
Signature  
(order invalid unless signed)
Prentice Hall PTR is pleased to recommend the following titles to USENIX members...

**USENIX members receive a 15% discount on orders**

  (15105-0)  List: $48.00  Members: $40.80

- **Object-Oriented Modeling and Design**, James Rumbaugh, 0-13-629841-9
  (62984-0)  List: $54.00  Members: $45.90

- **Zen and the Art of the Internet, Third Edition**, Brendan Kehoe, 0-13-121492-6
  (12149-1)  List: $23.95  Members: $20.36

- **The Magic Garden Explained**, Bernard Goodheart/James Cox, 0-13-098138-9
  (09813-7)  List: $38.00  Members: $32.30

  (47224-1)  List: $61.33  Members: $52.13

- **SCO Performance Tuning Handbook**, Gina Muciovich/David Simons, 0-13-102690-9
  (10269-9)  List: $42.00  Members: $35.70

- **Object-Oriented Programming**, Peter Coad/Jill Nicola, 0-13-032616-X
  (03261-5)  List: $48.00  Members: $40.90

  (47422-1)  List: $53.00  Members: $45.35

  (47423-9)  List: $53.00  Members: $45.05

  (09294-0)  List: $50.00  Members: $42.50

- **The Standard C Library**, PJ Plauger, 0-13-131509-9
  (13150-8)  List: $37.80  Members: $32.13

- **All About Administering the NIS+, Second Edition**, Rick Ramsey, 0-13-309576-2
  (30957-5)  List: $42.00  Members: $35.70

  (17725-3)  List: $55.00  Members: $46.75

- **Networking Operations on UNIX SVR4**, Mike Padavano, 0-13-613555-2
  (61355-4)  List: $50.00  Members: $42.50

- **Solaris Porting Guide**, Sunsoft ISV Engineering
  0-13-030396-8
  (03039-5)  List: $52.00  Members: $44.20

- **Multiprocessor System Architectures**, Ben Catanzaro
  0-13-089137-1
  (08913-6)  List: $44.00  Members: $37.40

- **The HP-UX System Administrator's"How To" Book**, Marty Poniatowski, 0-13-099821-4
  (09982-0)  List: $32.00  Members: $27.20

- **UNIX System V Performance Management**, edited by Phyllis Eve Bregman and Sally A. Browning
  0-13-016429-1
  (01642-8)  List: $29.95  Members: $25.45

  (01256-7)  List: $39.00  Members: $33.15

---

**HERE'S HOW TO ORDER:**

**CALL** 800-880-6818

**OR WRITE:**
CompuBooks
Route 1, Box 271-D,
Cedar Creek, TX 78612

**OR INTERNET:**
70007.1333@CompuServe.com
(600) 777-CBKB (GO CBK on CompuServe)

**WE SHIP ANYWHERE!**

**FOR MORE INFORMATION, OR QUANTITY ORDERS, PLEASE CALL 201-592-2657**
A UNIQUE OFFER ON THE BEST IN UNIX FOR USENIX MEMBERS

- **THE INTERNET GUIDE FOR NEW USERS**
  D. Dern
  hardcover, 016510-6, $40.00, member price $32.00
  paperback, 016511-4, $27.95, member price $22.36

- **INTERNET FOR EVERYONE**
  R. Wiggins
  hardcover, 067018-8, $29.95, member price $23.96
  paperback, 067019-6, $45.00, member price $36.00

- **THE ESSENTIAL INTERNET INFORMATION GUIDE**
  J. Manger
  707905-1, paperback, $27.95, member price $22.36

- **THE INFORMATION BROKERS HANDBOOK, Second Edition**
  S. Rugge
  911878-x, paperback, $34.95, member price $27.96
  Available December 1994

- **SAA AND UNIX: IBM's Open System Strategy**
  M. Killen
  034607-0, $40.00, member price $32.00

- **A STUDENT'S GUIDE TO UNIX**
  H. Hahn
  025511-3, paperback, $28.00, member price $22.40

- **UNIX DEVELOPER'S TOOL KIT**
  K. Leininger
  911836-4, $65.00, member price $52.00

- **UNIX SECURITY: A Practical Tutorial**
  N. Arnold
  002560-6, $24.95, member price $19.96

- **THE UNIX AUDIT: Using UNIX to Audit UNIX**
  M. Grottola
  025127-4, $32.95, member price $26.36

- **UNIX: A Database Approach**
  S. Das
  015745-6, $29.95, member price $23.96
  Available November 1994

---

I am a member of USENIX Association. Please send me the books I have indicated at the member special rate. I have added $3.00 postage and handling for the first book ordered, $1.00 for each additional book, plus my local sales tax.

Check or money order is enclosed—payable to McGraw-Hill, Inc.

Charge my [ ] Visa  [ ] Mastercard  [ ] Discover  [ ] Amex

Bill & Ship To:

Name__________________________

Street__________________________

City, State, Zip__________________

Daytime Phone #________________

Account #_______________________

Expiration Date__________________

Signature_______________________

USENIX Membership #____________

If I am not completely satisfied, I will return the book(s) within 15 days for a full refund or credit. Satisfaction unconditionally guaranteed. Prices subject to change without notice. We can only accept orders within the continental USA.
Commemorating 25 years of UNIX

This year UNIX marks its 25th anniversary. We think its something to celebrate. After all, we owe our livelihoods to UNIX. Over the years it's given us plenty to explore, learn, teach, and talk about. It's enabled us to do good work—work that's filled with adventure, fun, and creative challenge. We couldn't ask for much more than that. Except, of course, for a little cake and ice cream.

You might like to celebrate with these new titles from O'Reilly: Managing Internet Information Services, The Mosaic Handbook (with software for three platforms, including Microsoft Windows, X, and the Macintosh), X User's Tools, Motif Tools, PGP: Pretty Good Privacy, and Multi-Platform Code Management. Call or write to us at the numbers below for details. Mention code ALOG.

O'REILLY & ASSOCIATES, INC.
103A Morris Street, Sebastopol, CA 95472 FAX: 707-829-0104
Credit card orders: 1-800-889-8969 Weekdays 6AM-6PM PST
Online orders: order@ora.com
For inquiries or to request a copy of our catalog: 1-800-998-9938
UNIXWORLD is a monthly magazine written for people who integrate, resell, manage, and program commercial computer systems to provide solutions based on interoperable networks and applications. Topics for features include the effective use of UNIX and other open systems to downsize and distribute applications. The magazine also provides a useful mix of product reviews, case studies, industry profiles and analysis, and technical tutorials.

The Benefits of UNIXWORLD:

Programming Tips - helps the reader become educated in UNIX and OPEN SYSTEMS computing. Provides real-world application information the reader can use everyday.

Unbiased Hardware and Software Reviews - helps the reader recommend and justify purchasing decisions.

Clear Editorial - so both beginners and experts can get full benefit from the publication. Product Trends and Technical Advances - to aid professionals in developing company-wide systems strategies.

Profiles on U.S. and International Companies - provides the reader with a truly global perspective of the marketplace and the players. Helps vendors assess and monitor the competition.

Special Rates:

<table>
<thead>
<tr>
<th></th>
<th>DOMESTIC</th>
<th>CANADIAN</th>
<th>OTHER INTERNATIONAL (Air Delivery)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Yr.</td>
<td>$15.00</td>
<td>$21.00</td>
<td>1 Yr. $48.00</td>
</tr>
<tr>
<td>2 Yrs.</td>
<td>$24.00</td>
<td>$36.00</td>
<td>2 Yrs. $90.00</td>
</tr>
<tr>
<td>3 Yrs.</td>
<td>$30.00</td>
<td>$48.00</td>
<td>3 Yrs. $130.00</td>
</tr>
</tbody>
</table>

Be wise subscribe today!

To place your order please call:
Kelly Kendall at 1-800-327-6850 or fax your order to 609/426-7055
When it comes to Open Computing, the solution is UniForum.

UniForum '95
The Official Conference and Exposition for Open Computing Solutions
March 14–16, 1995
Dallas Convention Center • Dallas, Texas

REGISTER BY FAX!
Call 617-449-5554, enter Code 70 and have your fax number ready — we'll fax back your registration form within 24 hours!

NEW!
UNIFORUM ON-LINE...for the latest information!
Internet
World Wide Web URL: http://www.uniforum.org

Now managed by The Interface Group, the producer of COMDEX.
Sponsored by UniForum, The International Association of Open Systems Professionals.
What is Open Systems and where can you find the information you need to keep up in today’s competitive open systems environment?

Find out by joining UniForum:
The International Association of Open Systems Professionals.

The Advantages of Membership:
Your annual membership is a resource treasure chest. Among the many valuable benefits you'll receive are:

- UniForum Monthly: The Magazine for Open Systems Professionals
- UniNews - the authoritative voice of UniForum - now on-line
- The Open Systems Products Directory - comprehensive, accurate, indispensable
- Reduced fees at all UniForum conferences and seminars
- Technical and Standards publications: timely, reliable and useful
- Discounts on many outstanding industry publications
- Discounts on leading commercial hardware and software products
- Discounts on courses offered by leading training organizations
- Discounts on exclusive new shareware selections
- Services such as discounts on car rentals, travel and mail list rentals
- Eligibility to serve on UniForum committees and the Board of Directors

---

**UNIFORUM MEMBERSHIP APPLICATION FORM (please print)**

<table>
<thead>
<tr>
<th>Name</th>
<th>MR/MS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td></td>
</tr>
<tr>
<td>Company</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>Mail Stop</td>
<td></td>
</tr>
<tr>
<td>City</td>
<td>State</td>
</tr>
<tr>
<td>Country</td>
<td>Telephone (______)</td>
</tr>
<tr>
<td>E-mail (where we will send UniNews)</td>
<td>Fax (______)</td>
</tr>
</tbody>
</table>

**General Membership**

$125

Includes UniForum Monthly magazine, UniNews electronic newsletter, Open Systems Products Directory, technical publications, recruitment advertising service, and discounts on UniForum conference registration and shareware CD-ROMs, purchases of software and hardware, educational seminars and special classes, additional UniForum publications and other industry publications, computer books, CD-ROMs and training services.

** Overseas postage:

<table>
<thead>
<tr>
<th>Amount</th>
<th>$</th>
</tr>
</thead>
<tbody>
<tr>
<td>air $100 or surface $60 (no additional postage necessary for Canada, Mexico or Puerto Rico)</td>
<td></td>
</tr>
</tbody>
</table>

TOTAL AMOUNT ENCLOSED $______

**Method of Payment:**

Select one:  
☑ Check payable to UniForum  
☐ Money Order in US Dollars

☐ MasterCard  
☑ Visa  
☐ American Express

Credit card number  Expiration date

Signature

*Payments must be included in U.S. dollars. All checks must be drawn on a U.S. bank. Credit card orders can be accepted by telephone.

** Overseas surface delivery takes up to six weeks.

Send application and payment to: UniForum, 2901 Tasman Drive, Suite 205, Santa Clara, CA 95054-1100

Tel: (408) 986-8840  (800) 255-5620  Fax: (408) 986-1645

---

There's Never Been a Better Time for an Outstanding Value

As the unstoppable move to open systems gathers even more momentum you need a strong and independent association that can help you achieve your professional goals. An association that can magnify your voice, which can educate, inform and inspire. You need and deserve an association that provides value for your money. In short you need UniForum. With a value this good, the real question might be, "Can you afford to be without it?" Join today!
For the first time in one volume the collected Whitepapers from MOSES: The Massive Open Systems Environment Standard is now available from UniForum.

The MOSES Project is a collective effort by large open systems shops to define the operational production environment needed to ensure the highest levels of system availability and reliability. The Whitepapers are aimed primarily at MIS professionals who operate large data centers, especially those in UNIX environments. The contributors are all experts who wrote the selections as part of their efforts to improve their own shops and to create a dialogue on the best practices for perfecting the UNIX data center. The Whitepapers, while written by and for UNIX operations, are largely applicable to any operating system or database.

The MOSES Whitepapers are arranged as 12 stand-alone chapters that can be used individually or as a collection to provide data center operators with the guidelines they need to perform their duties more efficiently.

The Whitepapers include:

- General Introduction
  John C. Black, US WEST NewVector Group, Inc.
- Service Level Support Agreements
  Bill Holt, US WEST NewVector Group, Inc.
- Event Monitoring and Alarming
  Steve Starck, US WEST NewVector Group, Inc.
- Backup and Recovery Requirements
  Mike Farnham, Burlington Coat Factory
- Change Management
  Michelle L. Trout, Sequent Computer Systems
- Benchmark Standards/Metrics and Capacity Planning
  Paul McKenney, Sequent Computer Systems
- Data Archiving Requirements
  Ron Hawkins, Millipore
- Job Scheduling
  Dan Poulin, Millipore
- Interface and Infrastructure Requirements
  Bill Hughes, Sequent Computer Systems
- Security
  Barry Harness, British Telecom
- Disaster Planning/Recovery
  Scott McMahan, Oracle Corporation
- UNIX System Administration Hiring Process
  Bill Holt, US WEST NewVector Group, Inc.
MEMBERSHIP INFORMATION

Any individual or institution may become a member of the USENIX Association by filling out an application form and paying the appropriate annual fee.

There are five classes of membership:

Student: $25
Open to any full-time student at an accredited educational institution. A copy of the current student I.D. card must be provided.

Individual: $70
Open to any individual or institution. Individual Members may vote.

Corporate: $325
Corporate Membership is open to any individual or institution.

Educational: $175
Educational Membership is open to accredited educational institutions.

Supporting: $1000
Open to any individual or institution that wants to support the Association to a greater degree than through the Corporate Membership fee.

Corporate, Educational and Supporting members have one designated representative who receives all services available to Individual Members, plus copies of the proceedings from all conferences and symposia that are held during the term of membership.

SAGE, the System Administrators Guild

The System Administrators Guild (SAGE) is a Special Technical Group within the USENIX Association, devoted to the furtherance the profession of systems administration. SAGE brings together system administrators for professional development, for the sharing of problems and solutions, and to provide a common voice to users, management, and vendors on topics of system administration.

A number of working groups within SAGE are focusing on special topics such as conferences, local organizations, professional and technical standards, policies, systems and network security, publications, and education. USENIX and SAGE will work jointly to publish technical information and sponsor conferences, tutorials, and local groups in the systems administration field.

To become a SAGE member you must be a member of USENIX as well.

MEMBERSHIP APPLICATION

New ☐ Renewal ☐

Name ____________________________________________________________

Address __________________________________________________________

City ___________________________ State ________ Zip ______________ Country ___________________________

Phone ___________________________ email address: ___________________________

☐ $25 Student (full-time) (with copy of I.D. card) ☐ $175 Educational Institution ☐ $25 - SAGE (Additional)

☐ $70 Individual ☐ $325 Corporate ☐ $1000 Supporting

PAYMENT OPTIONS

☐ My total amount for membership is $_________.

☐ Check enclosed payable to USENIX Association. ☐ Purchase order enclosed (Educational and Corporate members only).

☐ Please charge my: ☐ Visa ☐ MasterCard ☐ American Express

Account # ___________________________ Exp. Date ___________________________

Signature __________________________________________________________

Outside the U.S.A.? Please make your payment in U.S. currency by one of the following:

* Charge (Visa, MasterCard, or foreign equivalent)
* International postal money order
* Check - issued by a local branch of a U.S. Bank

$24 of your annual membership dues is for a one-year subscription to the newsletter, ;login:, and $30 of your dues is for a one-year subscription to the journal, Computing Systems.

USENIX Mailing List

☐ I do not want my address made available to other members.

☐ I do not want my address made available for commercial mailings.

OCTOBER 1994 ;login: 75
The Association will support local user groups by doing a mailing to assist in the formation of a new group and publishing information on local groups in ;login:. At least one member of the group must be a current member of the Association. Send additions and corrections to: <login@usenix.org>.

## California

### Fresno:
The Central California UNIX Users Group consists of a uucp-based electronic mailing list to which members may post questions or information. For connection information:
- Educational and governmental institutions: Brent Auernheimer (209) 278-2573, <brent@CSUFresno.edu> or <cufres!brent>
- Commercial institutions or individuals:
  - Gordon Crumal (209) 251-2648 <cufres!gordon>

### Orange County
Meets the 2nd Monday of each month
- UNIX Users Association of Southern California
  - Dave Close (714) 434-7359 <dchloe@alumni.caltech.edu>
  - New Horizons Computer Learning Center
    - 1231 E. Dyer Rd., Suite 140
    - Santa Ana, CA 92705
    - (714) 438-9440

## Florida

### Coral Springs:
- S. Shaw McQuinn (305) 344-8686
  - 8557 W. Sample Road
  - Coral Springs, FL 33065

### Melbourne:
Meets the 3rd Monday of every month.
- Space Coast UNIX User's Group
  - Steve Lindsey (407) 242-4766 <lindsey@vnet.ibm.com>

### Orlando:
Meets the 3rd Thursday of each month.
- Central Florida UNIX Users Group
  - Mikkel Manitius (407) 444-8448 <mkel@aaa.com>

### Western:
Meets 1st Thursday of each month.
- Florida West Coast UNIX Users Group
  - Richard Martino (813) 536-1776
  - Tony Becker (813) 799-1836 <mcrsys!tony>
  - Ed Gallizzi, Ph.D. (813) 864-8272 <e.gallizzi@compmail.com>
  - Jay Ts (813) 979-9169 <utnet!pdl!tscs!mtrant!jan>
  - Dave Lewis (407)242-4372 <dhil@ccd.harris.com>

## Georgia

### Atlanta:
Meets on the 1st Monday of each month in White Hall, Emory University.
- Atlanta UNIX Users Group
  - P.O. Box 12241
  - Atlanta, GA 30355-2241
  - Mark Landry (404) 365-8108

## Kansas or Missouri

### Kansas City UNIX Users Group (KCUUG)
P.O. Box 412622
Kansas City, MO 64141
(816) 891-1093 <richj@northcs.cps.com>

## Michigan

### Detroit/Ann Arbor
Meets on the 2nd Thursday of each month in Ann Arbor.
- Southeastern Michigan Sun Local Users Group and Nameless UNIX Users Group
  - Steve Simmons office: (313)769-4086
  - home: (313) 426-8981 <sccs@lokkur.dexter.mi.us>

## Minnesota

### Minneapolis/St. Paul:
Meets the 1st Wednesday of each month.
- UNIX Users of Minnesota
  - 17130 Jordan Court
  - Lakeville, MN 55044
  - Robert A. Monio (612) 220-2427 <pessutil@dmnshq.mn.org>

## Missouri

### St. Louis:
- St. Louis UNIX Users Group P.O. Box 2182 St. Louis, MO 63158
  - Terry Linhardt (314) 772-4762 <uunet!galiss!terry>

## Nebraska

### Omaha: Meets monthly.
- /usr/group/nebraska
  - P.O. Box 31012
  - Omaha, NE 68132
  - Phillip Allendorfer (402) 423-1400
LOCAL USER GROUPS

New England

Northern:
Meets monthly at different sites.
- Peter Schmitt (603) 646-2085
  Kiewit Computation Center
  Dartmouth College
  Hanover, NH 03755
  <peter.schmitt@dartmouth.edu>

New Jersey

Princeton:
Meets monthly.
- Princeton UNIX Users Group
  Mercer County Community
  College
  1200 Old Trenton Road
  Trenton, NJ 08690
  Peter J. Holsberg (609) 586-4800
  <mcce!pjh>

New Mexico

Albuquerque:
ASIGUNIX meets every 3rd Wednesday of each month.
- Phil Hertz 505/275-0466.

New York

New York City:
Meets every other month in Manhattan.
- Unigroup of New York City
  G.P.O.
  Box 1931
  New York, NY 10116
  <unigroup@murphy.com>
  Bob Young (212) 490-8470

Oklahoma

Tulsa:
Meets 2nd Wednesday of each month.
- Tulsa UNIX Users Group, $USR
  Stan Mason (918) 560-5329
  <tulsi sx@mason@drd.com>
  Mark Lawrence (918) 743-3013
  <mark@drd.com>

Texas

Austin:
Meets 3rd Thursday of each month.
- Capital Area Central Texas UNIX Society (CACTUS)
  P.O. Box 9786

Austin, TX 78766-9786
Tom Painter (512) 258-7321
<president@caucus.org>

System Administration Groups

Back Bay LISA (BBLISA)
New England forum covering all aspects of system and network administration, for large and small installations. Meets monthly, at MIT in Cambridge, MA. For information, contact:
- J. R. Oldroyd (617)227-563
  <jr@opal.com>
- Mailing list subscription:
  <requests:bblisasquest@cs.umb.edu>
- Mailing list postings:
  <bblisa@cs.umb.edu>
- For current calendar of events:
  finger <bblisa@cs.umb.edu>

Bay LISA
Meets 3rd Thursday of each month in Mountain View, CA. For more information, please contact:
  <baylisa-info@baylisa.org> or
  Bryan McDonald,
  BayLISA President
  <bigmac@baylisa.org>
  P.O. Box 64369
  Sunnyvale, CA 94088-4369
  (415) 859-3246

$GROUPNAME (New Jersey)

$GROUPNAME is an organization in New Jersey formed to facilitate information exchange pertaining to the field of UNIX system administration. For more information, send email to:
  Majordomo@Warren. MENTORG.COM or
  Tom Limoncelli
  <tom_limoncelli@warren.mentorg.com>

New York Systems Administrators (NYSA)
Meets 2nd Monday of each month.
  <nysa-request@esm.com>
  914/472-3635

North Carolina System Administrators Group
The North Carolina System Administrators Group meets on the 2nd Monday each month around the Research Triangle Park area.
- Amy Kreiling (919) 962-1843
  <kreiling@cs.unc.edu>
- William E. Howell (919) 962-1717
  <howell@cs.unc.edu>
# CALENDAR OF EVENTS

This is a combined calendar of conferences, symposia, and standards meetings related to the UNIX operating system. If you have an event that you wish to publicize, please contact <login@usenix.org>. Please provide your information in the same format as below.

* = events sponsored by the USENIX Association.

## 1994

### December

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>5-9</td>
<td>IETF, San Jose, CA</td>
</tr>
<tr>
<td>8-9*</td>
<td>IEEE Mobile Computing Systems &amp; Applications, Santa Cruz, CA</td>
</tr>
<tr>
<td>10-15</td>
<td>DECUS, Anaheim, CA</td>
</tr>
</tbody>
</table>

### 1995

#### January

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>9-13</td>
<td>IEEE 1003, Fort Lauderdale, FL</td>
</tr>
<tr>
<td>16-20*</td>
<td>USENIX, New Orleans, LA</td>
</tr>
</tbody>
</table>

#### March

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>13-17</td>
<td>UniForum, Dallas, TX</td>
</tr>
<tr>
<td>21-23</td>
<td>AFUU Unix '95 Conference Paris, France</td>
</tr>
<tr>
<td>28-31</td>
<td>Fifth Conference on Computers, Freedom and Privacy (ACM)</td>
</tr>
</tbody>
</table>

#### April

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-7</td>
<td>IETF, Danvers, MA</td>
</tr>
<tr>
<td>10-11*</td>
<td>2nd Mobile and Location–Independent Computing, Ann Arbor, MI</td>
</tr>
<tr>
<td>10-14</td>
<td>IEEE 1003</td>
</tr>
<tr>
<td>10-14</td>
<td>3rd International World Wide Web Conference, Darmstadt, Germany</td>
</tr>
<tr>
<td>24-29*</td>
<td>SANS IV, Washington, DC</td>
</tr>
</tbody>
</table>

#### May

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-5</td>
<td>IEEE 5th Workshop on Hot Topics in Operating Systems Orcas Island, WA</td>
</tr>
<tr>
<td>6-11</td>
<td>DECUS, Washington, DC</td>
</tr>
</tbody>
</table>

#### June

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>5-7</td>
<td>UNIX Security, Salt Lake City, UT</td>
</tr>
<tr>
<td>26-29*</td>
<td>COOTS, Monterey, CA</td>
</tr>
</tbody>
</table>

#### July

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>10-14</td>
<td>IEEE 1003</td>
</tr>
</tbody>
</table>

## 1996

### January

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>22-26*</td>
<td>USENIX, San Diego, CA</td>
</tr>
</tbody>
</table>

### February

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>14-16</td>
<td>UniForum, San Francisco, CA</td>
</tr>
</tbody>
</table>

### May

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>18-24</td>
<td>DECUS, Orlando, FL</td>
</tr>
</tbody>
</table>

### August

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>6-11</td>
<td>Siggraph, Los Angeles, CA</td>
</tr>
<tr>
<td>13-17</td>
<td>Interex 95, Toronto, Canada</td>
</tr>
<tr>
<td>14-18</td>
<td>Computers in Context, Aarhus, Denmark</td>
</tr>
</tbody>
</table>

### September

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>18-22*</td>
<td>LISA '95, Monterey, CA</td>
</tr>
<tr>
<td>19-21</td>
<td>UNIX Expo, New York City</td>
</tr>
</tbody>
</table>

### October

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>9-13</td>
<td>IEEE 1003</td>
</tr>
<tr>
<td>15-19</td>
<td>OOPSLA, Austin, TX</td>
</tr>
</tbody>
</table>

### November

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-8</td>
<td>DECUS, San Francisco, CA</td>
</tr>
</tbody>
</table>

### December

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-6</td>
<td>JUS UNIX Fair, Tokyo, Japan</td>
</tr>
</tbody>
</table>

### December 1996

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-8</td>
<td>Interex 96, San Diego, CA</td>
</tr>
</tbody>
</table>

### November

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>16-22</td>
<td>DECUS, Anaheim, CA</td>
</tr>
</tbody>
</table>

### December

<table>
<thead>
<tr>
<th>Date</th>
<th>Event Details</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>JUS UNIX Fair, Tokyo, Japan</td>
</tr>
</tbody>
</table>
Integrate Macs with Sun, SGI, & HP

Server to the Macs

K-AShare/K-FS

Share Resources seamlessly
Macintosh® and UNIX® workstation users share files effortlessly. K-AShare™ brings UNIX-resident files to Macintosh computers; K-FS™ brings Macintosh files to UNIX workstations. Cross-platform applications like Frame, Photoshop and Illustrator can read and write the same files from either system.

Better than an AppleShare file server
UNIX workstations running K-AShare provide higher performance than dedicated AppleShare servers while allowing Mac users to continue using the familiar Mac interface.

And, the UNIX workstation simultaneously gets other important jobs done for you.

K-Spool

More printing options
Macintoshes and UNIX systems talk to all PostScript printers and image setters regardless of how they're connected. Each system continues to use its own familiar printing interface.

Increased productivity
Macs finish printing more quickly, and UNIX workstations have access to many more printing devices. K-Spool also brings the power of SGI's impressario™ to your Macs.